Cisco Nexus 9000 シリーズ NX-OS システム管理構成ガイド、リリース 10.4(x)

最終更新：2025 年 8 月 15 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 REFERENCED IN THIS DOCUMENTATION ARE SUBJECT TO CHANGE WITHOUT NOTICE. EXCEPT AS MAY OTHERWISE BE AGREED BY CISCO IN WRITING, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENTATION ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

The Cisco End User License Agreement and any supplemental license terms govern your use of any Cisco software, including this product documentation, and are located at: https://www.cisco.com/c/en/us/about/legal/cloud-and-software/software-terms.html. Cisco product warranty information is available at https://www.cisco.com/c/en/us/products/warranty-listing.html. US Federal Communications Commission Notices are found here https://www.cisco.com/c/en/us/products/us-fcc-notice.html.

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 products and features described herein as in development or available at a future date remain in varying stages of development and will be offered on a when-and if-available basis. Any such product or feature roadmaps are subject to change at the sole discretion of Cisco and Cisco will have no liability for delay in the delivery or failure to deliver any products or feature roadmap items that may be set forth in this document.

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.

The documentation set for this product strives to use bias-free language. For the purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

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: https://www.cisco.com/c/en/us/about/legal/trademarks.html. 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)

© 2023 –2024 Cisco Systems, Inc. All rights reserved.

目 次

Trademarks ?

はじめに：

はじめに xxix

対象読者 xxix

表記法 xxix

Cisco Nexus 9000 シリーズ スイッチの関連資料 xxx

マニュアルに関するフィードバック xxx

通信、サービス、およびその他の情報 xxxi

Cisco バグ検索ツール xxxi

マニュアルに関するフィードバック xxxi

第 1 章

新機能および変更された機能に関する情報 1

新機能および変更された機能に関する情報 1

第 2 章

概要 11

ライセンス要件 11

サポートされるプラットフォーム 12

Cisco NX-OS デバイスのコンフィギュレーション方式 12

CLI または XML 管理インターフェイスで設定する 13

Cisco DCNM での設定 13

ネットワーク タイム プロトコル 13

Cisco Discovery Protocol 13

セッションマネージャ 14

スケジューラ 14

SNMP 14

オンライン診断 14
オンボード障害ロギング 14
SPAN 15
ERSPAN 15
LLDP 15
MPLS ストリッピング 15
sFlow 15
SMU 15
仮想デバイス コンテキスト 16
トラブルシューティング機能 16

第 3 章 2 ステージ コンフィギュレーション コミット 17
2 段階構成のコミットについて 17
注意事項と制約事項 18
2 ステージ コンフィギュレーション コミット モードでの設定 19
2ステージコンフィギュレーション コミット モードの中止 26
コミット ID の表示 26
ロールバック機能 27
現在のセッション設定の表示 27

第 4 章 スイッチ プロファイルの設定 29
スイッチ プロファイルの概要 29
スイッチ プロファイル：コンフィギュレーション モード 30
コンフィギュレーション同期モード 30
スイッチ プロファイル モード 30
スイッチ プロファイル インポート モード 30
コンフィギュレーションの検証 30
相互排除チェック 31
マージ チェック 31
スイッチ プロファイルを使用したソフトウェアのアップグレードとダウングレード 31
スイッチ プロファイルの注意事項および制約事項 32

スイッチ プロファイルの設定 34
スイッチ プロファイルのコマンドの追加または変更 36
スイッチ プロファイルのインポート 38
vPC トポロジでの設定のインポート 40
ピア スイッチの分離 41
スイッチ プロファイルの削除 41
ミューテックスとマージの失敗の手動修正 42
スイッチ プロファイル設定の確認 43
スイッチ プロファイルの設定例 43
ローカルおよびピア スイッチでのスイッチ プロファイルの作成... 43
同期ステータスの確認 46
実行中のコンフィギュレーションの表示 47
ローカルとピア スイッチ間のスイッチ プロファイルの同期の表示 47
ローカルおよびピア スイッチでの確認とコミットの表示 48
ローカルおよびピア スイッチ間の成功および失敗した同期の表示 49
スイッチ プロファイル バッファの表示 50
設定のインポート 51
ファブリック エクステンダのストレート型トポロジでの Cisco NX-OS リリース 7.0(3)I2(1) 以降への移行 53
Cisco Nexus 9000 シリーズ スイッチの交換 53
設定の同期 55
Cisco Nexus 9000 シリーズ スイッチのリブート後の設定の同期化 55
mgmt0 インターフェイスの接続が失われた場合の設定の同期化 55
グローバル コンフィギュレーション モードでレイヤ 2 からレイヤ 3 への不注意によるポート モードの変更を元に戻す 55

第 5 章 周波数の同期の設定 57
周波数同期化について 57
外部 PRC ソースを使用した Hybrid SyncE-PTP 58
タイミング ソース 58
タイミング入力 58

タイミング出力 59
タイミング ソース選択ポイント 59
同期イーサネット（SyncE）のライセンス要件 60
周波数同期のガイドラインと制限事項 60
周波数の同期の設定 61
周波数の同期の有効化 61
インターフェイスの周波数の同期の設定 63
周波数の同期の設定の確認 66

第 6 章 **PTP の設定 71**
PTP について 71
PTP オフロード 72
PTP デバイス タイプ 73
クロック 73
PTP プロセス 74
PTP の ITU-T 電気通信プロファイル 76
Telecom Profile G.8275.1 76
Telecom Profile G.8275.2 77
PTP のハイ アベイラビリティ 79
PTP の注意事項および制約事項 79
PTP のデフォルト設定 86
PTP の設定 87
PTP のグローバルな設定 87
PTP GM の構成 92
インターフェイスでの PTP の設定 93
ユニキャストモードでの PTP の設定 100
IPv4 または IPv6 向けユニキャスト モードの構成 100
動的ロールの割り当て 101
マスター ロールの割り当て 102
スレーブ ロールの割り当て 104
ユニキャスト送信元アドレスの設定 106

PTP テレコム プロファイルの設定 107
グローバル PTP テレコム プロファイルの設定 107
PTP テレコム プロファイル 8275-1 のインターフェイスの構成 110
PTP テレコム プロファイル 8275-2 のインターフェイスの構成 114
PTP プロファイルのデフォルト 118
PTP 通知の設定 120
PTP 混合モード 122
PTP インターフェイスがマスター ステートを維持する設定 122
PTP ユニキャスト ネゴシエーションの有効化 124
拡張マルチキャスト スケール 127
タイムスタンプ タギング 128
タイムスタンプ タギングの設定 128
TTAG マーカー パケットと時間間隔の設定 129
PTP 設定の確認 132
PTP テレコム プロファイル設定の確認 133
PTP の設定例 138
その他の参考資料 140
関連資料 140

第 7 章 GPS の設定 141
GPS について 141
GPS に関する注意事項と制限事項 142
グランドマスター クロックの GPS の構成 142
GPS 構成の検証 143

第 8 章 GNSS の構成 145
GNSS について 145
GNSS の注意事項と制約事項 146
GNSS レシーバーの有効化 146
GNSS 構成の検証 148

第 9 章 NTP の設定 151
NTP の詳細 151
NTP アソシエーション 152
時間サーバとしての NTP 152
クロック マネージャ 152
高可用性 153
仮想化のサポート 153
NTP の前提条件 153
NTP の注意事項と制約事項 153
NTP のデフォルト設定 156
NTP の設定 156
NTP の有効化または無効化 156
正規の NTP サーバとしてのデバイスの設定 157
NTP サーバおよびピアの設定 158
NTP 認証の設定 160
NTP アクセス制限の設定 162
NTP ソース IP アドレスの設定 164
NTP ソース インターフェイスの設定 165
NTP ロギングの設定 166
NTP の設定確認 167
NTP の設定例 167
その他の参考資料 169
関連資料 169
MIB 169

第 10 章 CDP の設定 171
CDP について 171
VTP 機能のサポート 172
高可用性 173
仮想化のサポート 173

CDP の注意事項と制約事項 173
CDP のデフォルト設定 173
CDP の設定 174
CDP のグローバルな有効化または無効化 174
インターフェイス上での CDP の有効化または無効化 175
CDP オプション パラメータの設定 176
CDP コンフィギュレーションの確認 177
CDP のコンフィギュレーション例 178

第 11 章 システムメッセージロギングの設定 179
システム メッセージ ロギングの詳細 179
Syslogサーバ 180
セキュアな Syslog サーバ 180
システム メッセージ ロギングの注意事項および制約事項 181
システム メッセージ ロギングのデフォルト設定 182
システムメッセージロギングの設定 182
ターミナル セッションへのシステム メッセージ ロギングの設定 182
Syslog メッセージの送信元 ID の設定 185
ファイルへのシステム メッセージの記録 186
モジュールおよびファシリティ メッセージのロギングの設定 188
syslog サーバの設定 191
セキュアな Syslog サーバの設定 193
CA 証明書の設定 194
CA 証明書の登録 195
UNIX または Linux システムでの syslog サーバの設定 197
ログ ファイルの表示およびクリア 198
システム メッセージ ロギングの設定確認 199
繰り返されるシステム ロギング メッセージ 200
システム メッセージ ロギングの設定例 201
その他の参考資料 201
関連資料 201

第 12 章 **Smart Call Home の設定** 203

Smart Call Home の概要 203

Smart Call Home - 概念 204

宛先プロファイル 204

Smart Call Home アラート グループ 205

Smart Call Home のメッセージ レベル 208

Smart Call Home の取得 209

データベース マージの注意事項 210

高可用性 210

仮想化のサポート 210

Smart Call Home の前提条件 211

Smart Call Home の注意事項および制約事項 211

Smart Call Home のデフォルト設定 211

Smart Call Home の設定 212

連絡先情報の設定 213

宛先プロファイルの作成 215

宛先プロファイルの変更 217

アラート グループと宛先プロファイルのアソシエート 219

アラート グループへの show コマンドの追加 220

電子メール サーバの設定 221

HTTP を使用したメッセージ送信のための VRF 設定 223

HTTP プロキシ サーバの設定 224

定期的なインベントリ通知の設定 226

重複メッセージ抑制のディセーブル化 227

Smart Call Home のイネーブル化またはディセーブル化 228

Call Home メール転送用の SMTP-AUTH の設定 229

送信元インターフェイスの構成 232

Smart Call Home 設定のテスト 233

Smart Call Home 設定の確認 234

Smart Call Home の設定例 234

その他の参考資料 236
イベント トリガ 236
メッセージ フォーマット 238
ショート テキスト メッセージ フォーマット 238
共通のイベント メッセージ フィールド 239
アラート グループ メッセージ フィールド 241
リアクティブおよびプロアクティブ イベント メッセージのフィールド 241
インベントリ イベント メッセージのフィールド 242
ユーザが作成したテスト メッセージのフィールド 243
フル テキスト形式での syslog アラート通知の例 243
XML 形式での syslog アラート通知の例 246
MIB 249

第 13 章 Session Manager の設定 251
セッション マネージャについて 251
高可用性 252
セッション マネージャの前提条件 252
Session Manager の注意事項および制約事項 252
Session Manager の設定 252
セッションの作成 253
セッションでの ACL の設定 253
セッションの確認 254
セッションのコミット 255
セッションの保存 255
セッションの廃棄 255
Session Manager 設定の確認 255
Session Manager のコンフィギュレーション例 256
その他の参考資料 256
関連資料 256

第 14 章 スケジューラの設定 257

スケジューラについて 257
リモート ユーザ認証 258
ログ 258
高可用性 258
スケジューラの前提条件 258
スケジューラの注意事項および制約事項 259
スケジューラのデフォルト設定 259
スケジューラの設定 260
スケジューラの有効化または無効化 260
スケジューラ ログ ファイル サイズの定義 261
リモート ユーザ認証の設定 261
ジョブの定義 263
ジョブの削除 264
タイムテーブルの定義 265
スケジューラ ログ ファイルの消去 267
スケジューラの設定確認 268
スケジューラの設定例 268
スケジューラ ジョブの作成 268
スケジューラ ジョブのスケジューリング 268
ジョブ スケジュールの表示 268
スケジューラ ジョブの実行結果の表示 269

第 15 章 SNMP の設定 271
SNMP について 271
SNMP 機能の概要 271
SNMP 通知 272
SNMPv3 273
SNMPv1、SNMPv2、SNMPv3 のセキュリティ モデルおよびセキュリティ レベル 274
ユーザベースのセキュリティ モデル 275
CLI および SNMP ユーザの同期 276
グループベースの SNMP アクセス 278

SNMP および Embedded Event Manager 279
マルチ インスタンス サポート 279
SNMP のハイ アベイラビリティ 279
SNMP の仮想化サポート 280
SNMP の注意事項および制約事項 280
SNMP のデフォルト設定 283
SNMP の設定 283
SNMP ユーザーの構成 283
ハッシュ化されたパスワードをオフラインで生成する 285
SNMP メッセージ暗号化の適用 286
SNMPv3 ユーザに対する複数のロールの割り当て 287
SNMP コミュニティの作成 287
SNMP 要求のフィルタリング 288
SNMP 通知レシーバの設定 289
SNMP 通知用の発信元 インターフェイスの設定 290
通知ターゲット ユーザの設定 292
VRF を使用する SNMP 通知レシーバの設定 293
帯域内ポートを使用してトラップを送信するための SNMP 設定 295
SNMP 通知のイネーブル化 296
インターフェイスでのリンク通知のディセーブル化 305
インターフェイスの SNMP ifIndex の表示 306
TCP による SNMP のワンタイム認証の有効化 306
SNMP スイッチのコンタクト（連絡先）およびロケーション情報の指定 307
コンテキストとネットワーク エンティティ間のマッピング設定 308
SNMP のディセーブル化 309
SNMP サーバ カウンタ キャッシュ更新タイマーの管理 310
AAA 同期時間の変更 311
SNMP ローカル エンジン ID の設定 312
SNMP の設定の確認 313
SNMP エンティティ 315
SNMP の設定例 315

その他の参考資料 317
関連資料 317
RFC 317
MIB 317

第 16 章 RMON の設定 319
RMON について 319
RMON アラーム 320
RMON イベント 320
RMON のハイ アベイラビリティ 321
RMON の仮想化サポート 321
RMON の注意事項と制約事項 321
RMON のデフォルト設定 321
RMON の設定 322
RMON アラームの設定 322
RMON イベントの設定 323
RMON 設定の確認 324
RMON の設定例 325
その他の参考資料 325
MIB 325

第 17 章 オンライン診断の設定 327
オンライン診断について 327
ブートアップ診断 327
ランタイムまたはヘルス モニタリング診断 329
オンデマンド診断 338
高可用性 338
仮想化のサポート 339
オンライン診断の注意事項と制約事項 339
オンライン診断のデフォルト設定 340
オンライン診断の設定 341

起動診断レベルの設定 341
診断テストのアクティブ化 342
オンデマンド診断テストの開始または中止 343
診断結果のシミュレーション 344
診断結果の消去 345
オンライン診断設定の確認 345
オンライン診断のコンフィギュレーション例 346

第 18 章 Embedded Event Manager の設定 347
EEM について 347
ポリシー 348
イベント文 349
アクション文 350
VSH スクリプト ポリシー 351
環境変数 351
EEM イベント相関 351
高可用性 351
仮想化のサポート 352
EEM の前提条件 352
EEM の注意事項と制約事項 352
EEM のデフォルト設定 354
EEM の設定 354
環境変数の定義 354
CLI によるユーザ ポリシーの定義 355
イベント文の設定 357
アクション文の設定 363
VSH スクリプトによるポリシーの定義 365
VSH スクリプト ポリシーの登録およびアクティブ化 365
ポリシーの上書き 366
メモリのしきい値の設定 367
EEM パブリッシャとしての syslog の設定 369

EEM の設定確認 370
EEM の設定例 371
イベント ログの自動収集とバックアップ 372
拡張ログ ファイルの保持 373
すべてのサービスの拡張ログ ファイル保持のイネーブル化 373
すべてのサービスの拡張ログ ファイル保持の無効化 373
単一サービスの拡張ログファイル保持の有効化 374
拡張ログ ファイルの表示 376
ログ統計ごとのグローバル ディクショナリの表示 376
単一サービスに対する拡張ログファイル保持の無効化 377
トリガーベースのイベント ログの自動収集 379
トリガーベースのログ ファイルの自動収集の有効化 379
ログプロファイル YAML ファイル 379
自動収集 YAML ファイル 380
コンポーネントあたりの自動収集の量の制限 387
自動収集ログ ファイル 387
サポートされているログ収集の例 391
トリガーベースのログ収集の確認 391
トリガーベースのログ ファイル生成の確認 392
ローカル ログ ファイルのストレージ 392
最近のログ ファイルのローカル コピーの生成 392
外部ログ ファイルのストレージ 395

第 19 章 MAC 移動ポリシーの構成 397
MAC 移動ポリシーについて 397
MAC 移動ポリシーの注意事項と制約事項 397
MAC 移動ポリシーの構成 398
MAC 移動ポリシーの構成の確認 399

第 20 章 VSH セッションの端末ロック 401
VSH セッションの端末ロック 401

第 21 章 オンボード障害ロギングの設定 405

OBFL の概要 405

OBFL の前提条件 406

OBFL の注意事項と制約事項 406

OBFL のデフォルト設定 406

OBFL の設定 406

OBFL 設定の確認 409

OBFL のコンフィギュレーション例 410

その他の参考資料 411

関連資料 411

第 22 章 SPAN の設定 413

SPAN の概要 413

SPAN ソース 413

送信元ポートの特性 414

SPAN 宛先 415

宛先ポートの特性 415

SPAN セッション 415

ローカライズされた SPAN セッション 416

SPAN 切り捨て 416

ACL TCAM リージョン 416

高可用性 416

SPAN の前提条件 417

SPAN の注意事項および制約事項 417

Cisco Nexus 3000 プラットフォーム スイッチの SPAN の制限 422

Cisco Nexus 9200 プラットフォーム スイッチの SPAN の制限事項（9232E-B1 を除く） 422

Cisco Nexus 9300 プラットフォーム スイッチの SPAN の制限事項 423

Cisco Nexus 9500 プラットフォーム スイッチの SPAN の制限事項 426

Cisco Nexus 9800 および 9232E-B1 プラットフォーム スイッチの SPAN の注意事項と制限事項 429

SPAN のデフォルト設定 429
SPAN の設定 430
SPAN セッションの設定 430
UDF ベース SPAN の設定 434
SPAN 切り捨ての設定 437
異なる LSE スライス間のマルチキャスト Tx トラフィックの SPAN の設定 438
SPAN から CPU への構成 439
はじめに 439
注意事項と制約事項 440
SPAN から CPU への構成 441
SPAN セッションのシャットダウンまたは再開 442
SPAN 設定の確認 444
SPAN のコンフィギュレーション例 444
SPAN セッションのコンフィギュレーション例 444
単一方向 SPAN セッションの設定例 445
SPAN ACL の設定例 446
UDF ベース SPAN の設定例 446
SPAN 切り捨ての設定例 447
LSE スライス間のマルチキャスト Tx SPAN の設定例 448
その他の参考資料 449
関連資料 449

第 23 章 **ERSPAN の設定** 451
ERSPAN について 451
ERSPAN 送信元 451
ERSPAN の宛先 452
ERSPAN セッション 452
ローカライズされた ERSPAN セッション 453
ERSPAN の切り捨て 453
ERSPAN の前提条件 453
ERSPAN の注意事項および制約事項 453

デフォルト設定 459
ERSPAN の設定 459
ERSPAN 送信元セッションの設定 459
ERSPAN セッションのシャットダウンまたはアクティブ化 463
ERSPAN ACL の設定 465
ERSPAN ACL 構成の確認 468
UDF ベース ERSPAN の設定 469
ERSPAN 切り捨ての設定 471
ERSPAN 宛先セッションの設定 473
ERSPAN 設定の確認 476
ERSPAN の設定例 477
IPv6 経由の ERSPAN 送信元セッションの設定例 477
単一方向 ERSPAN セッションの設定例 477
ERSPAN ACL の設定例 477
マーカー パケットの設定例 478
UDF ベース ERSPAN の設定例 479
ERSPAN 切り捨ての設定例 480
IPv4 上の ERSPAN 接続先セッションの構成例 480
IPv6 上の ERSPAN 接続先セッションの構成例 481

第 24 章 **LLDP の構成 483**
LLDP について 483
DCBXP について 484
高可用性 485
仮想化のサポート 485
LLDP に関する注意事項および制約事項 486
LLDP のデフォルト設定 487
LLDP の構成 488
LLDP をグローバルに有効化または無効化する 488
インターフェイス上での LLDP の有効化または無効化 489
DCBXP 出力キューイングの構成 491

DCBXP プロトコル バージョンの設定 492
物理インターフェイスごとの複数の LLDP ネイバー 493
LLDP マルチネイバー サポートのイネーブル化またはディセーブル化 494
ポート チャネル インターフェイスでの LLDP サポートの有効化または無効化 496
LLDP オプション パラメータの設定 499
LLDP 設定の確認 500
LLDP の設定例 501

第 25 章 NetFlow の設定 503

第 26 章 混合モードで構成する 537
混合モードについて 537
混合モードに関する注意事項と制限事項 537
混合モード：ユース ケース 539
ユースケース：機能分析がすでに展開されたスイッチ 539
ユースケース：すでに機能 NetFlow が展開されたスイッチ 539
ユースケース：どちらの機能も構成されていないスイッチ 540
混合モード構成の検証 542
混合モードの表示例 543

第 27 章 sFlow の設定 545
sFlow 545
sFlow エージェント 545
sFlow の前提条件 546
sFlow の注意事項および制約事項 546
sFlow のデフォルト設定 549
sFlow の設定 550
sFlow の有効化 550
サンプリング レートの設定 551
最大サンプリング サイズの設定 552
カウンタのポーリング間隔の設定 553

最大データグラム サイズの設定 554
sFlow コレクタ アドレスの設定 555
sFlow コレクタ ポートの設定 556
sFlow エージェント アドレスの設定 557
sFlow サンプリング データ ソースの設定 558
sFlow 拡張 BGP（Gateway）の設定 559
sFlow 設定の確認 560
sFlow 統計情報のモニタリングとクリア 561
sFlow の設定例 561
その他の参考資料 562
関連資料 562

第 28 章 TAP アグリゲーション 563
TAP アグリゲーションについて 563
ネットワーク TAP 563
TAP アグリゲーション 564
TAP 集約の注意事項と制約事項 564
MPLS ストリッピングについて 567
MPLS ストリッピングに関する注意事項と制限事項 567
TAP アグリゲーションの設定 569
ライン カードの TAP 集約のイネーブル化 569
TAP 集約ポリシーの設定 570
TAP アグリゲーション ポリシーのインターフェイスへのアタッチ 572
プロバイダー VLAN で選択的 Q-in-Q を構成する 573
TAP アグリゲーションの設定の確認 575
TAP アグリゲーションの設定例 575
MPLS ストリッピングの設定 576
MPLS ストリッピングの有効化 576
VLAN タグの着信ポートの設定 577
MPLS ラベルの追加と削除 579
宛先 MAC アドレスの設定 580

MPLS ラベル エージングの設定 581
MPLS ストリッピング設定の確認 581
MPLS ストリッピング カウンタおよびラベル エントリのクリア 583
MPLS ストリッピングの設定例 583
その他の参考資料 584
関連資料 584

第 29 章 重複排除の構成 585
重複排除について 585
注意事項と制約事項 586
重複排除の構成 587

第 30 章 MPLS アクセス リストの構成 589
MPLS アクセス リストの構成 589
MPLS アクセス リスト構成の検証 590
MPLS アクセス リストの構成例 590

第 31 章 Nexus Data Broker のヘッダ ストリッピング機能の構成 591
Nexus Data Broker の ヘッダー ストリッピングの紹介 591
ヘッダー ストリッピングに関する注意事項と制限事項 593
Nexus Data Broker の VXLAN および iVXLAN ヘッダー ストリッピング 594
Nexus Data Broker – VXLAN および iVXLAN ヘッダ ストリッピングについて 594
ストリップ VXLAN および iVXLAN をサポートされている PID 595
VXLAN および IVXLAN ヘッダー ストリップに関する注意事項と制限事項 595
Nexus Data Broker 終了の構成 596
VXLAN および iVXLAN ヘッダー ストリップの構成例 599
Nexus Data Broker の ERSPAN ヘッダー ストリッピング 600
ERSPAN ヘッダ ストリッピングについて 600
ERSPAN ヘッダをストリッピングするためにサポートされる PID 600
ERSPAN ヘッダ ストリッピングに関する注意事項と制限事項 600
ERSPAN ヘッダ ストリッピングの設定 601

ERSPAN ヘッダ ストリッピングの設定例 603
ERSPAN ヘッダ ストリッピングの設定の確認 603
Nexus Data Broker の GRE ヘッダ ストリッピング 604
NDB GRE ヘッダー ストリッピングについて 604
NDB GRE ヘッダー ストリッピングに関する注意事項と制限事項 604
GRE ヘッダー ストリップ機能の CLI 605
出力ポートと入力ポートの構成 605
Nexus Data Broker の MPLS ヘッダ ストリッピング 606
NDB MPLS ヘッダー ストリッピングについて 607
NDB MPLS ヘッダー ストリッピングに関する注意事項と制限事項 607
MPLS ヘッダー ストリップ機能のコマンド 609
出力ポートと入力ポートの構成 610

第 32 章 グレースフル挿入と削除の設定 613
グレースフル挿入と削除について 613
プロファイル 614
スナップショット 616
GIR の注意事項と制限事項 616
GIR ワークフロー 617
メンテナンス モード プロファイルの設定 617
通常モード プロファイルの設定 619
スナップショットの作成 621
スナップショットへの show コマンドの追加 623
グレースフル削除のトリガー 625
グレースフル挿入のトリガー 628
メンテナンス モードの強化 629
GIR 設定の確認 631
GIR の設定例 632

第 33 章 ソフトウェア メンテナンス アップグレードの実行 635
SMU について 635

パッケージ管理 636
パッケージのアクティブ化と非アクティブ化の影響 637
SMU の前提条件 637
SMU の注意事項と制約事項 638
Cisco NX-OS のソフトウェア メンテナンス アップグレードの実行 639
パッケージ インストールの準備 639
Cisco.com からの SMU パッケージ ファイルのダウンロード 640
ローカル ストレージ デバイスまたはネットワーク サーバへのパッケージ ファイルのコピー 641
パッケージの追加とアクティブ化 644
アクティブなパッケージ セットのコミット 646
パッケージの非アクティブ化と削除 647
SMU インストールのリロードなしオプション 649
高度な SMU インストール方法 655
単一の TAR ファイルを使用した複数の SMU パッケージのインストール 655
新しい NX-OS ソフトウェア イメージのインストールの一部としての SMU パッケージのインストール 656
機能 RPM のダウングレード 657
インストール ログ情報の表示 659
Guest Shell Bash のソフトウェア メンテナンス アップグレードの実行 659
その他の参考資料 661
関連資料 661

第 34 章 コンフィギュレーションの置換の実行 663
コンフィギュレーションの置換とコミットタイムアウトについて 663
概要 664
コンフィギュレーションの置換の利点 665
コンフィギュレーションの置換に関する注意事項と制限事項 666
コンフィギュレーションの置換の推奨ワークフロー 668
コンフィギュレーションの置換の実行 670
コンフィギュレーションの置換の確認 673

コンフィギュレーションの置換の例 673

第 35 章 ロールバックの設定 681

ロール バックについて 681

システム チェックポイントの自動生成 682

高可用性 682

仮想化のサポート 683

ロールバックの前提条件 683

ロールバックの注意事項と制約事項 683

ロールバックのデフォルト設定 684

ロールバックの設定 684

チェックポイントの作成 684

ロールバックの実装 685

ロールバック コンフィギュレーションの確認 686

ロールバックの設定例 687

その他の参考資料 687

関連資料 687

第 36 章 候補構成の完全性チェック 689

候補構成について 689

候補構成の完全性チェックの注意事項と制限事項 689

候補構成の完全性チェックの実行 690

完全性チェックの例 691

第 37 章 安全に消去するを実行します 693

安全に消去する（Secure Erase）機能に関する情報 693

安全な消去を実行するための前提条件 694

安全な消去の注意事項と制約事項 694

安全な消去の設定 694

付録 A： Cisco NX-OS システム管理でサポートされている IETF RFC 705

Cisco NX-OS システム管理でサポートされている IETF RFC 705

付録 B：

Embedded Event Manager システム イベントおよび設定例 707

EEM システム ポリシー 707

EEM イベント 711

EEM ポリシーの設定例 712

CLI イベントの設定例 712

インターフェイス シャットダウンのモニタリング 712

モジュール パワーダウンのモニタリング 713

ロールバックを開始するトリガーの追加 713

メジャーしきい値を上書き（無効化）する設定例 713

メジャーしきい値に達したときにシャットダウンを防ぐ方法 713

One Bad センサーの無効化 714

複数の不良センサーを無効にする方法 714

モジュール全体の上書き（無効化） 714

複数のモジュールおよびセンサーの上書き（無効） 715

1 つのセンサーを有効にして、すべてのモジュールの残りのセンサーをすべて無効にする方法 715

複数のセンサーを有効にして、すべてのモジュールの残りのセンサーをすべて無効にする方法 715

1 つのモジュールのすべてのセンサーを有効にして、残りのモジュールのすべてのセンサーを無効にする方法 716

モジュールのセンサーを組み合わせて有効にして、残りのモジュールのすべてのセンサーを無効にする方法 716

ファントレイ取り外しのためのシャットダウンを上書き（無効化）するコンフィギュレーション例 717

1 つまたは複数のファン トレイ取り外しのためのシャットダウンの上書き（無効） 717

指定したファン トレイを取り外すためのシャットダウンの上書き（無効） 717

指定した複数のファン トレイを取り外すためのシャットダウンの上書き（無効化） 717

1 つを除くすべてのファンを取り外すためのシャットダウンの上書き（無効） 718

ファン トレイの指定したセットを除くファン トレイを取り外すためのシャットダウンの上書き（無効） 718

ファン トレイのセットから 1 台を除くすべてのファン トレイを取り外すためのシャットダウンの上書き（無効） 718
補足ポリシーを作成するコンフィギュレーション例 719
ファン トレイが存在しないイベントの補足ポリシーの作成 719
温度しきい値イベントの補足ポリシーの作成 719
電力のバジェット超過ポリシーの設定例 719
モジュールのシャットダウン 719
指定された一連のモジュールのシャットダウン 720
シャットダウンするモジュールを選択する設定例 720
デフォルトでシャットダウンに非上書きモジュールを選択するポリシーの使用 720
シャットダウンに非上書きモジュールを選択するパラメータ置き換えの使用 720
活性挿抜イベントのコンフィギュレーション例 721
ユーザ syslog を生成するコンフィギュレーション例 721
Syslog メッセージをモニタする設定例 721
SNMP 通知の設定例 722
SNMP OID のポーリングによる EEM イベントの生成 722
イベント ポリシーのイベントへの応答で SNMP 通知を送信 722
ポート トラッキングの設定例 722
EEM によって EEM ポリシーを登録する設定例 723

付録 C： **Cisco NX-OS システム管理の設定制限 727**
Cisco NX-OS システム管理の設定制限 727

はじめに

この前書きは、次の項で構成されています。

- 対象読者 （xxix ページ）
- 表記法 （xxix ページ）
- Cisco Nexus 9000 シリーズ スイッチの関連資料 （xxx ページ）
- マニュアルに関するフィードバック （xxx ページ）
- 通信、サービス、およびその他の情報 （xxxi ページ）

対象読者

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

表記法

コマンドの説明には、次のような表記法が使用されます。

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

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

例では、次の表記法を使用しています。

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

Cisco Nexus 9000 シリーズ スイッチの関連資料

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

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

マニュアルに関するフィードバック

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

通信、サービス、およびその他の情報

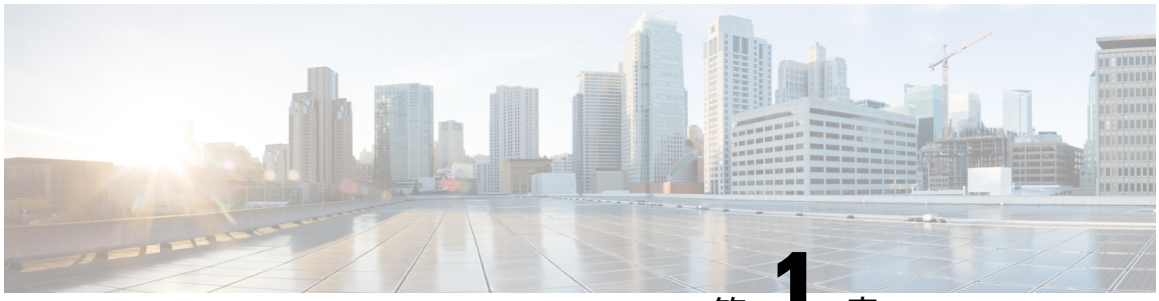
- シスコからタイムリーな関連情報を受け取るには、Cisco Profile Manager でサインアップしてください。
- 重要な技術によりビジネスに必要な影響を与えるには、Cisco Services にアクセスしてください。
- サービス リクエストを送信するには、Cisco Support にアクセスしてください。
- 安全で検証済みのエンタープライズクラスのアプリケーション、製品、ソリューション、およびサービスを探して参照するには、Cisco DevNet [英語] にアクセスしてください。
- 一般的なネットワーキング、トレーニング、認定関連の出版物を入手するには、Cisco Press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

Cisco バグ検索ツール

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

マニュアルに関するフィードバック

シスコのテクニカルドキュメントに関するフィードバックを提供するには、それぞれのオンラインドキュメントの右側のペインにあるフィードバックフォームを使用してください。



第 1 章

新機能および変更された機能に関する情報

• 新機能および変更された機能に関する情報 （1 ページ）

新機能および変更された機能に関する情報

表 *1* : 新機能および変更された機能

特長	説明	変更が行われたリリース	参照先
SNMP	Cisco Nexus N9K-C9364C-H1 プラットフォームスイッチで SNMP のサポートが追加されました。	10.4(3)F	SNMP の注意事項および制約事項 （280 ページ）
sFlow	Cisco Nexus N9K-C9364C-H1 プラットフォームスイッチで sFlow のサポートが追加されました。	10.4(3)F	sFlow の注意事項および制約事項 （546 ページ）
セキュア消去のサポート	Cisco Nexus N9K-C9364C-H1 スイッチでセキュア消去のサポートが追加されました。	10.4(3)F	安全な消去の注意事項と制約事項 （694 ページ）
SPAN	Cisco Nexus 93400LD-H1 プラットフォームスイッチで SPAN のサポートが追加されました。	10.4(2)F	SPAN の注意事項および制約事項 （417 ページ）

新機能および変更された機能に関する情報

特長	説明	変更が行われたリリース	参照先
ERSPAN	Cisco Nexus 93400LD-H1 プラットフォームスイッチで ERSPAN のサポートが追加されました。	10.4(2)F	ERSPAN の注意事項および制約事項 （453 ページ）
CDP	Cisco Nexus 9232E-B1 プラットフォームスイッチで CDP のサポートが追加されました。	10.4(2)F	CDP の注意事項と制約事項 （173 ページ）
LLDP	Cisco Nexus 9232E-B1 プラットフォームスイッチで LLDP のサポートが追加されました。	10.4(2)F	LLDP に関する注意事項および制約事項 （486 ページ）
出力 NetFlow	入力 NetFlow の既存のサポートに加えて、出力 NetFlow のサポートが追加されました	10.4(1)F	出力 NetFlow について （525 ページ） NetFlow の設定 （503 ページ） 入力 NetFlow について （503 ページ）
ロギング 2.0 の機能拡張	追加コンポーネントの ipqosmgr、aclqos、cfs、ethport、feature-mgr、icam、interface manager、lacp、m2rib、mfdm、nbm、ngoam、nve、port-channel、qos、sla_responder、sla_sender、sla_twamp,smm、spm、sysmgr、および vpc に対する、デフォルトのブロガー自動収集のサポートが追加されました。	10.4(1)F	EEM の注意事項と制約事項 （352 ページ） コンポーネントごとの自動収集の作成または削除 （381 ページ） サポートされているログ収集の例 （391 ページ）

特長	説明	変更が行われたリリース	参照先
N9K-C9408 での G.8275.1 テレコム プロファイルの PTP クラス C サポート	Cisco Nexus 9408 プラットフォーム スイッチでの G.8275.1 テレコム プロファイルの PTP クラス C のサポートが追加されました	10.4(1)F	PTP の注意事項および制約事項 （79 ページ） PTP プロファイルのデフォルト （118 ページ）
N9K-C9332D-H2R の SyncE のサポート	Cisco N9K-C9332D-H2R スイッチの SyncE のサポートが追加されました	10.4(1)F	周波数同期のガイドラインと制限事項 （60 ページ）
N9K-C9332D-H2R での G.8275.1 テレコム プロファイルの PTP メディア プロファイルとクラス C のサポート	Cisco Nexus 9332D-H2R プラットフォーム スイッチでの PTP メディア プロファイルおよびクラス C G.8275.1 テレコム プロファイルのサポートが追加されました。	10.4(1)F	PTP の注意事項および制約事項 （79 ページ） PTP プロファイルのデフォルト （118 ページ）
1G ポートでの ASIC ベース PTP のサポート	Cisco Nexus 931806C-FX3 および 93108TC-FX3P プラットフォーム スイッチの 1G PTP 対応ポートの ASIC ベースのタイムスタンプのサポートが追加されました。	10.4(1)F	PTP の注意事項および制約事項 （79 ページ） PTP のグローバルな設定 （87 ページ）

新機能および変更された機能に関する情報

特長	説明	変更が行われたリリース	参照先
PTP	次のスイッチおよびライン カードでの PTP のサポートが追加されました。 • Cisco Nexus 9804 プラットフォーム スイッチ • Cisco Nexus X98900CD-A および X9836DM-A ラインカードと Cisco Nexus 9808 および 9804 スイッチ	10.4(1)F	PTP の注意事項および制約事項 （79 ページ） PTP プロファイルのデフォルト （118 ページ）
PTP	Cisco Nexus C9348GC-FX3 スイッチのサポートが追加されました	10.4(1)F	PTP の注意事項および制約事項 （79 ページ）
Generic Online Diagnostics（GOLD）	次のラインカードおよびスイッチで GOLD のサポートが追加されました。 • Cisco Nexus 9804 および N9K-C9332D-H2R プラットフォーム スイッチ • Cisco Nexus X98900CD-A ライン カード • Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ	10.4(1)F	オンライン診断の注意事項と制約事項 （339 ページ） ランタイムまたはヘルス モニタリング診断 （329 ページ）

特長	説明	変更が行われたリリース	参照先
集約アドレスルートを取り消す BGP 分離モード	集約アドレスルートを取り消すための BGP 分離モードに関する情報が追加されました。	10.4(1)F	メンテナンスモードの強化 （629 ページ） GIR の設定例 （632 ページ）
EPLD バージョンの SNMP OID	Cisco Nexus 9300-FX/FX2/FX3/GX プラットフォーム スイッチおよび Nexus 9508 スイッチの SNMP を使用して EPLD ファームウェア バージョンを表示するためのサポートが追加されました。	10.4(1)F	SNMP の注意事項および制約事項 （280 ページ） SNMP エンティティ （315 ページ）
重複排除	Cisco Nexus 9300 FX2/FX3/GX/GX2 プラットフォームスイッチの重複排除機能のサポートが追加されました。	10.4(1)F	重複排除について （585 ページ）
自動収集の機能拡張	予期しないプロトコルフラップ時にデフォルトで自動データ収集を行うための BGP、OSPF、ISIS、および BFD プロトコルをサポートするための自動収集機能拡張の詳細が追加されました。	10.4(1)F	プロトコルフラップの自動収集の作成 （384 ページ）

新機能および変更された機能に関する情報

特長	説明	変更が行われたリリース	参照先
CDP	次のラインカードおよびスイッチで CDP のサポートが追加されました。 • Cisco Nexus 9804 プラットフォーム スイッチ • Cisco Nexus X98900CD-A ライン カード • Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ	10.4(1)F	CDP の注意事項と制約事項 （173 ページ）
LLDP	次のラインカードおよびスイッチで LLDP のサポートが追加されました。 • Cisco Nexus 9804 プラットフォーム スイッチ • Cisco Nexus X98900CD-A ライン カード • Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ	10.4(1)F	LLDP に関する注意事項および制約事項 （486 ページ）

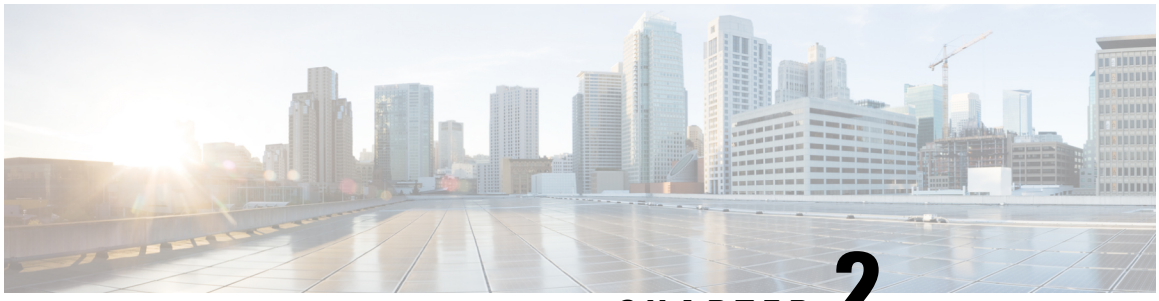
特長	説明	変更が行われたリリース	参照先
SNMP	次のラインカードおよびスイッチでGOLDのサポートが追加されました。 • Cisco Nexus 9804 および N9K-C9332D-H2R プラットフォーム スイッチ • Cisco Nexus X98900CD-A ライン カード • Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ	10.4(1)F	SNMP の注意事項および制約事項 （280 ページ）
sFlow	次のラインカードおよびスイッチでGOLDのサポートが追加されました。 • Cisco Nexus 9804 および N9K-C9332D-H2R プラットフォーム スイッチ • Cisco Nexus X98900CD-A ライン カード • Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ	10.4(1)F	sFlow の注意事項および制約事項 （546 ページ）

新機能および変更された機能に関する情報

特長	説明	変更が行われたリリース	参照先
セキュア消去のサポート	Cisco N9K-C9332D-H2R スイッチでのセキュア消去のサポートが追加されました	10.4(1)F	安全な消去の注意事項と制約事項 （694 ページ）
入力 NetFlow	Cisco N9K-C9332D-H2R スイッチでの入力 NetFlow のサポートが追加されました	10.4(1)F	入力 NetFlow に関する注意事項および制約事項 （507 ページ）
SPAN	次のラインカードおよびスイッチでの SPAN のサポートが追加されました。 • Cisco Nexus 9804 プラットフォーム スイッチ • Cisco Nexus 9332D-H2R プラットフォーム スイッチ • Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ	10.4(1)F	SPAN の注意事項および制約事項 （417 ページ） Cisco Nexus 9800 および 9232E-B1 プラットフォーム スイッチの SPAN の注意事項と制限事項 （429 ページ） SPAN 切り捨ての設定 （437 ページ） SPAN 設定の確認 （444 ページ）

特長	説明	変更が行われたリリース	参照先
ERSPAN	次のラインカードおよびスイッチでのERSPANのサポートが追加されました。 • Cisco Nexus 9804 プラットフォーム スイッチ • Cisco Nexus 9332D-H2R プラットフォーム スイッチ • Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ	10.4（1）F	ERSPAN の注意事項および制約事項 （453 ページ） ERSPAN 切り捨ての設定 （471 ページ） ERSPAN 設定の確認 （476 ページ）

新機能および変更された機能に関する情報



CHAPTER 2

概要

この章では、Cisco NX-OS デバイスのモニタや管理に使用できるシステム管理機能について説明します。

- ライセンス要件 （11 ページ）
- サポートされるプラットフォーム （12 ページ）
- Cisco NX-OS デバイスのコンフィギュレーション方式 （12 ページ）
- ネットワーク タイム プロトコル （13 ページ）
- Cisco Discovery Protocol （13 ページ）
- セッションマネージャ （14 ページ）
- スケジューラ （14 ページ）
- SNMP （14 ページ）
- オンライン診断 （14 ページ）
- オンボード障害ロギング （14 ページ）
- SPAN （15 ページ）
- ERSPAN （15 ページ）
- LLDP （15 ページ）
- MPLS ストリッピング （15 ページ）
- sFlow （15 ページ）
- SMU （15 ページ）
- 仮想デバイス コンテキスト （16 ページ）
- トラブルシューティング機能 （16 ページ）

ライセンス要件

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

サポートされるプラットフォーム

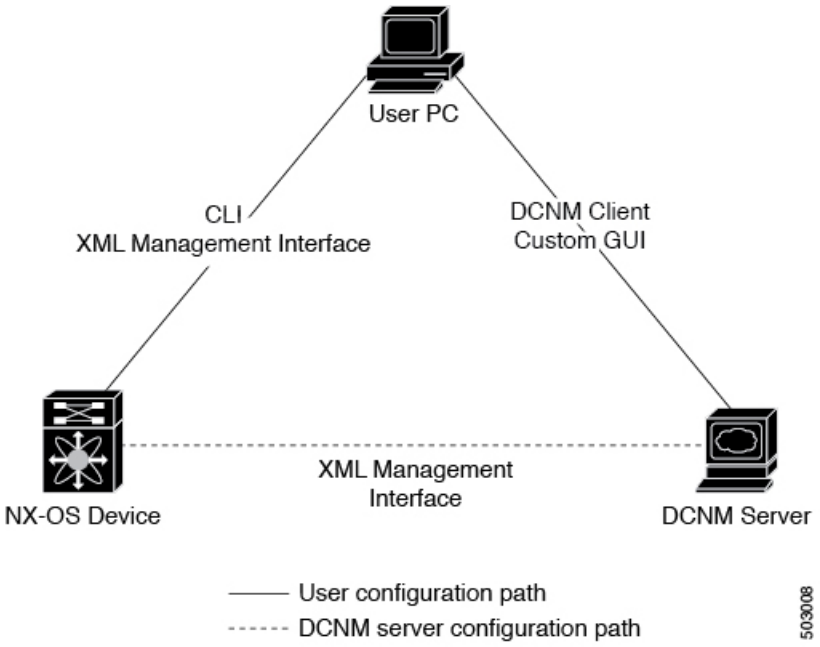
Cisco NX-OS リリース 7.0(3)I7(1) 以降では、Nexus スイッチ プラットフォーム サポート マトリクスに基づいて、選択した機能をさまざまな Cisco Nexus 9000 および 3000 スイッチで使用するために、どの Cisco NX-OS リリースが必要かを確認してください。

Cisco NX-OS デバイスのコンフィギュレーション方式

デバイスは、直接ネットワーク コンフィギュレーション方式またはCiscoデータ センター ネットワーク管理（DCNM）サーバが提供する Web サービスを使用して設定できます。

次の図は、ネットワーク ユーザが使用できるデバイスのコンフィギュレーション方式を示します。

図 *1 : Cisco NX-OS* デバイスのコンフィギュレーション方式



この表に、コンフィギュレーション方式と詳しい説明が記載されているマニュアルを示します。

表 *2:* コンフィギュレーション方式および参考資料

設定方法	ドキュメント
セキュア シェル（SSH）セッション、Telnet セッション、またはコンソール ポートからの CLI	
Cisco DCNM クライアント	*Cisco DCNM* 基本ガイド

CLI または XML 管理インターフェイスで設定する

次のように SSH からコマンドライン インターフェイス（CLI）または XML 管理インターフェイスを使用して、Cisco NX-OS デバイスを設定できます。

- SSH セッション、Telnet セッション、またはコンソール ポートから CLI：SSH セッション、Telnet セッション、またはコンソール ポートを使用してデバイスを設定できます。SSH ではデバイスへの安全な接続が提供されます。詳細については、『Cisco Nexus 9000 シリーズ NX-OS 基本設定ガイド』を参照してください。
- SSH を介して XML 管理インターフェイス：XML 管理インターフェイスを使用してデバイスを設定できます。これは、CLI 機能を補完する NETCONF プロトコルに基づくプログラム方式です。詳細については、『*Cisco NX-OS XML* 管理ユーザ ガイド』を参照してください。

Cisco DCNM での設定

Cisco DCNM クライアントを使用して Cisco NX-OS デバイスを設定できます。Cisco DCNM クライアントはユーザのローカル PC 上で動作し、Cisco DCNM サーバの Web サービスを使用します。Cisco DCNM サーバでは XML 管理インターフェイスを使用してデバイスを設定します。Cisco DCNM クライアントの詳細については、『Cisco DCNM Fundamentals Guide』を参照してください。

ネットワーク タイム プロトコル

ネットワーク タイム プロトコル（NTP）は、分散している一連のタイム サーバとクライアント間で 1 日の時間を同期させ、ネットワーク内のデバイスから受信するシステム ログなどの時間関連の情報を相互に関連付けることができます。

Cisco Discovery Protocol

Cisco Discovery Protocol（CDP）を使用して、デバイスに直接接続されているすべてのシスコ製機器を検出し、情報を表示できます。CDP は、ルータ、ブリッジ、アクセス サーバ、コミュニケーション サーバ、スイッチを含む、シスコ製のあらゆる機器で動作します。CDP は、メ

ディアにもプロトコルにも依存せず、ネイバー デバイスのプロトコル アドレスを収集し、各デバイスのプラットフォームを検出します。CDPの動作はデータリンク層上に限定されます。異なるレイヤ 3 プロトコルをサポートする 2 つのシステムで相互学習が可能です。

セッションマネージャ

Session Managerを使用すると、コンフィギュレーションを作成し、すべて正しく設定されていることを確認および検証したあとでバッチ モードで適用できます。

スケジューラ

スケジューラを使用すると、データの定期的なバックアップや quality of service（QoS）ポリシーの変更などのジョブを作成し、管理できます。スケジューラでは、ジョブを指定された時間に一度だけ、または定期的な間隔で実行するなど、ニーズに合わせて開始できます。

SNMP

簡易ネットワーク管理プロトコル（SNMP）は、SNMP マネージャとエージェント間の通信用メッセージフォーマットを提供する、アプリケーションレイヤプロトコルです。SNMP では、ネットワーク内のデバイスのモニタリングと管理に使用する標準フレームワークと共通言語が提供されます。

オンライン診断

Cisco Generic Online Diagnostics（GOLD）では、複数のシスコ プラットフォームにまたがる診断操作の共通フレームワークを定義しています。オンライン診断フレームワークでは、中央集中システムおよび分散システムに対応する、プラットフォームに依存しない障害検出アーキテクチャを規定しています。これには共通の診断CLIとともに、起動時および実行時に診断するための、プラットフォームに依存しない障害検出手順が含まれます。プラットフォーム固有の診断機能は、ハードウェア固有の障害検出テストを行い、診断テストの結果に応じて適切な対策を実行できます。

オンボード障害ロギング

永続ストレージに障害データを記録するように、デバイスを設定できます。あとで記録されたデータを取得して表示し、分析できます。この On-Board Failure Logging（OBFL: オンボード障害ロギング）機能は、障害および環境情報をモジュールの不揮発性メモリに保管します。この情報は、障害モジュールの分析に役立ちます。

SPAN

イーサネット スイッチド ポート アナライザ（SPAN）を設定すると、デバイスの入出力トラフィックをモニタできます。SPAN の機能を使用すると、送信元ポートから宛先ポートへのパケットを複製できます。

ERSPAN

Encapsulated Remote Switched Port Analyzer（ERSPAN）は、IP ネットワークでミラーリングされたトラフィックを転送するために使用します。ERSPAN は異なるスイッチ上の送信元ポート、送信元VLAN、および宛先をサポートし、ネットワーク上にある複数のスイッチのリモート モニタリングを可能にします。

ERSPAN 送信元セッションを設定するには、送信元ポートまたは VLAN のセットを、宛先 IP アドレス、ERSPAN ID 番号、および仮想ルーティングおよび転送（VRF）名に対応付けます。（VRF）名に対応付けます。

LLDP

リンク層検出プロトコル（LLDP）はベンダーに依存しない、単一方向のデバイス ディスカバリ プロトコルです。このプロトコルでは、ネットワーク上の他のデバイスにネットワーク デバイスから固有の情報をアドバタイズできます。このプロトコルはデータリンク層で動作するため、異なるネットワーク層プロトコルが稼働する 2 つのシステムで互いの情報を学習できます。LLDP はグローバルに、またはインターフェイスごとにイネーブルにすることができます。

MPLS ストリッピング

MPLS ストリッピングは、MPLS ラベルをパケットから除去する機能を提供し、非 MPLS 対応ネットワーク モニタリング ツールでパケットをモニタできるようにします。

sFlow

サンプリングされたフロー（sFlow）では、スイッチとルータを含むデータ ネットワークのリアルタイム トラフィックをモニタし、中央データ コレクタにサンプルデータを転送できます。

SMU

ソフトウェア メンテナンス アップグレード（SMU）は、特定の障害の修正を含むパッケージ ファイルです。SMU は、直近の問題に対処するために作成され、新しい機能は含まれていません。SMU は、メンテナンス リリースの代わりになるものではありません。直近の問題に対

する迅速な解決策を提供します。SMU で修正された障害は、メンテナンス リリースにすべて統合されます。

仮想デバイス コンテキスト

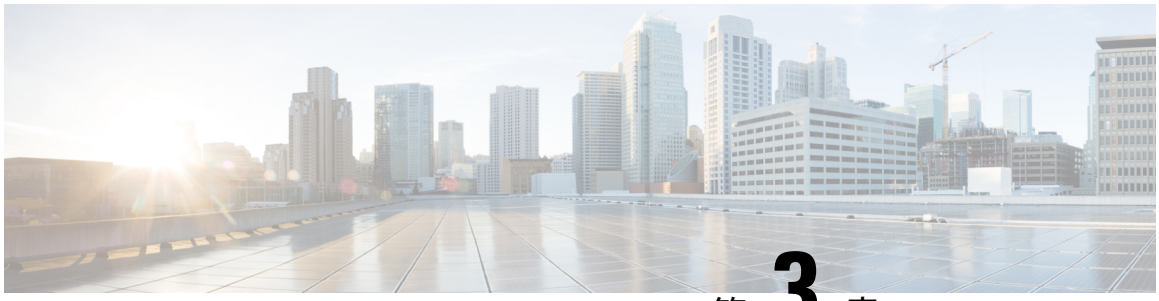
Cisco NX-OS では、仮想デバイスをエミュレートする Virtual Device Context（VDCs）に、OS およびハードウェア リソースを分割できます。Cisco Nexus 9000 シリーズ スイッチは、現在のところ、複数の VDC をサポートしていません。すべてのスイッチ リソースはデフォルト VDC で管理されます。

トラブルシューティング機能

Cisco NX-OS には ping、traceroute、Ethanalyzer、Blue Beacon 機能など、さまざまなトラブルシューティング ツールが揃っています。

サービスで障害が発生すると、システムは障害の原因を判定するために使用できる情報を生成します。次の情報ソースが使用可能です。

- サービスの再起動によって、LOG_ERR レベルの Syslog メッセージが生成されます。
- Smart Call Home サービスがイネーブルになっている場合は、サービスの再起動によって Smart Call Home イベントが生成されます。
- SNMP トラップがイネーブルになっている場合、サービスが再起動されると、SNMP エージェントはトラップを送信します。
- サービスの障害がローカル モジュール上で発生した場合は、そのモジュール内で **show processes log** コマンドを入力することで、イベントのログを表示できます。プロセスのログは、スーパーバイザのスイッチオーバーまたはリセット後も保持されます。
- サービスの障害が発生すると、システムのコア イメージ ファイルが生成されます。最新のコア イメージを表示するには、アクティブなスーパーバイザ上で**show cores** コマンドを入力します。スーパーバイザのスイッチオーバーおよびリセットが生じると、コア ファイルは保持されません。ただし、**system cores** コマンドを入力し、Trivial File Transfer Protocol（TFTP）のファイル転送ユーティリティを使用して、コア ファイルを外部サーバへエクスポートするようシステムを設定できます。
- CISCO-SYSTEM-MIB には、コアのテーブルが含まれています（cseSwCoresTable）。



第 3 章

2ステージコンフィギュレーションコミット

この章では、Cisco NX-OS デバイス上で 2 ステージ コンフィギュレーション コミット モードを有効にする方法について説明します。

この章は、次の項で構成されています。

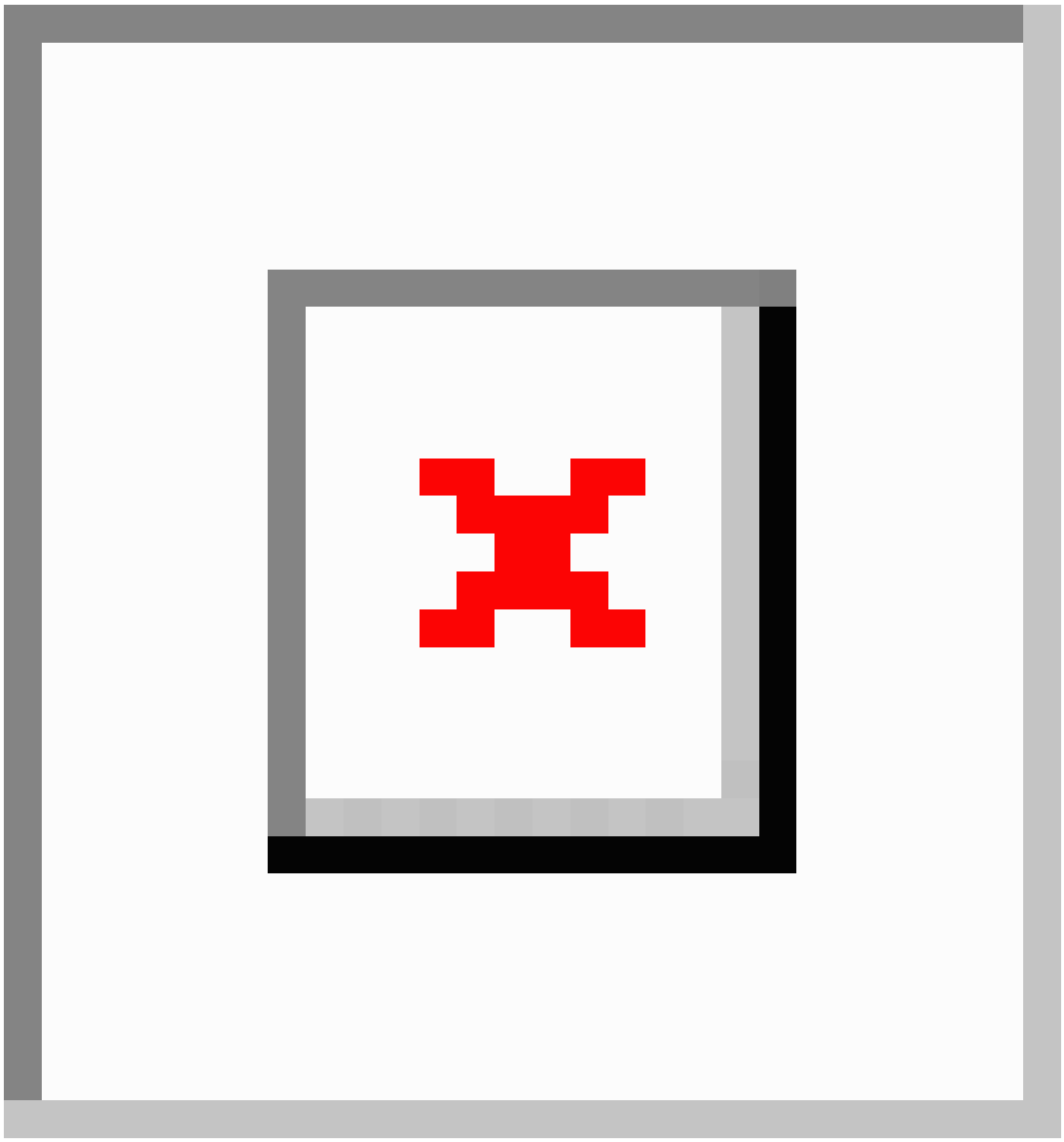
- 2 段階構成のコミットについて （17 ページ）
- 注意事項と制約事項 （18 ページ）
- 2 ステージ コンフィギュレーション コミット モードでの設定 （19 ページ）
- 2ステージコンフィギュレーション コミット モードの中止 （26 ページ）
- コミット ID の表示 （26 ページ）
- ロールバック機能 （27 ページ）
- 現在のセッション設定の表示 （27 ページ）

2 段階構成のコミットについて

インタラクティブセッションでは、コマンドを実行するとコマンドが実行され、実行コンフィギュレーションが変更されます。この動作は、1 ステージコンフィギュレーション コミットと呼ばれます。確認コミットまたは2段階の設定コミットでは、設定の変更がステージング データベースに保存されます。これらの変更は、**commit** コマンドを実行するまで実行コンフィギュレーションに影響しません。この 2 段階のプロセスにより、ターゲット コンフィギュレーション セッションが作成されます。このコンフィギュレーションでは、スイッチの実行状態にコミットする前に、設定の変更、編集、および確認を行うことができます。永続的にコミットする前に、指定した期間の変更をコミットすることもできます。**commit** コマンドを実行しないと、指定した時間が経過してもスイッチは以前の設定に戻ります。コミットが成功すると、コミット ID、ユーザ名、およびタイムスタンプを含むコミット情報を表示できます。

次の図に、2 段階の設定コミット プロセスを示します。

図 *2: 2* 段階でのコミット コンフィギュレーション プロセス



注意事項と制約事項

2 段階設定コミットには、次の注意事項および制限事項があります。

- この機能は、ユーザ インタラクティブ セッションの CLI インターフェイスでのみサポートされます。
- 機能関連のコンフィギュレーション コマンドを実行する前に、**feature** コマンドを使用して機能を有効にし、**commit** コマンドを使用してコミットします。

- 2 段階設定コミット モードは、メンテナンス モード、スケジューラ モード、仮想モードなどの他のモードをサポートしていません。
- 2 段階設定コミット モードの場合は、1 段階設定コミット モードで異なるセッションから同時に設定を編集しないでください。
- 変更を確定する前に、**show configuration** コマンドを使用して設定を確認します。
- Show configuration には、段階的な設定が表示されます。
  - 実際の違いが表示されます。つまり、同じコマンドの yes および no 形式は空の設定になります。
  - 設定を無効にするには、正確な no 形式の cli を発行することを推奨します。

    例：「ip address x」設定を無効にするには、「no ip address」ではなく「no ip address x」を指定する必要があります。
  - インターフェイス レイヤ変更コマンド（switchport / no switchport）は明示的に発行する必要があります。
  - コミットを試行する前に、セッション内の無効な設定をユーザが手動で削除する必要があります。手動で削除できなかった場合は、セッションをクリアして新しいセッションを開始します。
- 検証に失敗した場合は、コミットして編集します。
- コミットが失敗すると、設定は以前の設定にロールバックされます。
- コミットしない設定は、スイッチをリロードした後は保存されません。
- この機能は、NX-API、EEM、PPM、および Netconf でのコミットをサポートしていません。
- 一度にアクティブにできる 2 段階設定コミット セッションは 1 つだけです。

2 ステージ コンフィギュレーション コミット モードでの設定

2 ステージ コンフィギュレーション コミット モードで機能を有効にするには、次の手順を実行します。

（注） この手順では、例として BGP 機能を有効にします。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure dual-stage**
	例：
	`switch# configure dual-stage`
	`switch(config-dual-stage)#`
		新しいターゲット コンフィギュレーション セッションを作成します。
		（注）
		ターゲット コンフィギュレーションは、実行コンフィギュレーションのコピーではありません。ターゲット コンフィギュレーションには、そのターゲット コンフィギュレーション セッションで入力されたコンフィギュレーション コマンドだけが含まれます。

ステップ 2	**feature** *feature_name*
	例：
	`switch(config-dual-stage)# feature bgp`
	`switch(config-dual-stage)#`
		機能を有効にします。
		（注）
		• 2 ステージ コンフィギュレーション コミット モードを開始する前でも、この機能を有効にできます。
		• 機能が有効になっていない場合は、機能関連のコマンドを組み合わせて使用することはできません。

ステップ 3	**commit** [**confirmed** *seconds*]
	例：
	`switch(config-dual-stage-router)# commit confirmed 30`
	`Verification Succeeded.`
	`Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer.`
	`Please avoid other configuration changes during this time.`
	
	`Configuration committed by user 'admin' using Commit ID : 1000000001`
	
	`switch(config-dual-stage)#`
	`switch(config-dual-stage)# commit`
	`Confirming commit for trial session.`
	`switch(config-dual-stage)#`
	例：
	`switch(config-dual-stage)# hostname example-switch`
	`switch(config-dual-stage)# commit`
	`Verification Succeeded.`
	
	`Proceeding to apply configuration. This might`
		実行コンフィギュレーションに変更をコミットします。
		• **confirmed**：実行コンフィギュレーションに変更をコミットします。
		• 秒: グローバル コンフィギュレーション モードで、最低 30 秒間、最大 65535 秒間の試験稼働のためにコンフィギュレーションをコミットします。
		（注）
		トライアル期間を入力する場合は、commit コマンドを実行して設定を確認します。commit コマンドを実行しないと、トライアル期間後に以前の設定に戻ります。

	コマンドまたはアクション	目的
	take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time. Configuration committed by user 'admin' using Commit ID : 1000000002 example-switch(config-dual-stage)#	
ステップ 4	例： switch(config-dual-stage)# router bgp 64515.46 switch(config-dual-stage-router)# switch(config-dual-stage-router)# router-id 141.8.139.131 switch(config-dual-stage-router)#	このコンフィギュレーション モードでサポートされている機能関連のコマンドを実行します。
ステップ 5	**show configuration** 例： switch(config-dual-stage-router)# show configuration ! Cached configuration ! router bgp 64515.46 router-id 141.8.139.131	ターゲット コンフィギュレーションの内容を表示します。 （注） このコマンドは、デュアルステージ コンフィギュレーション モードでのみ実行できます。
ステップ 6	**commit** [**confirmed** *seconds*] 例： switch(config-dual-stage-router)# commit Verification Succeeded. Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time. Configuration committed by user 'admin' using Commit ID : 1000000003	実行コンフィギュレーションに変更をコミットします。
ステップ 7	（任意） **show configuration commit** [**changes**] *commit-id* 例： switch(config-dual-stage-router)# show configuration commit changes 1000000003 *** /bootflash/.dual-stage/1000000003.tmp Fri Mar 19 10:59:00 2021 --- /bootflash/.dual-stage/1000000003 Fri Mar 19 10:59:05 2021 *************** *** 378,383 **** --- 378,385 ---- line console line vty boot nxos bootflash:/nxos64.10.1.1.44.bin + router bgp 64515.46 + router-id 141.8.139.131 xml server timeout 1200	コミット関連情報を表示します。 最後の 50 個のコミットまたは予約済みディスク領域に保存されたコミット ファイルのみが保存されます。予約済みディスク領域は20MBです。スイッチをリロードすると、すべてのコミット セッションが削除されます。ただし、コミット ID は削除されません。また、これらのコミット ID は、書き込み、消去、およびリロードの際にも削除されません。 指定したコミットの現在のセッションの変更のみを表示するには、**show configuration commit changes** *commit-id* コマンドを使用します。 指定したコミットの完全な構成と、いくつかのクラス マップ ポリシーが表示されます。これらのクラ

	コマンドまたはアクション	目的
	no priority-flow-control override-interface mode off **例：** switch(config-dual-stage)# show configuration commit 1000000003 feature bgp router bgp 64515.46 router-id 141.8.139.131 . . .	ス マップ ポリシーは、新しいポリシーではなく、非表示のポリシーです。非表示のポリシーを表示するには、 **show run all** コマンドを使用します。
ステップ 8	（任意） **save configuration** *filename* **例：** switch(config-dual-stage)# save configuration bootflash:test.cfg	ターゲット コンフィギュレーションは、実行コンフィギュレーションにコミットすることなく、独立したファイルに保存できます。 **（注）** • ターゲットコンフィギュレーションファイルは、後でロード、変更、またはコミットできます。ファイルはブートフラッシュに保存されます。 • 保存したコンフィギュレーション ファイルを表示するには、**show configuration file***filename* コマンドを実行します。 • ユーザ固有の情報の一部は、ユーザ ロールに基づいてマスクされます。 • デュアルステージモードで保存された設定は暗号化されたファイルであり、**#show configuration file <>** を使用してのみ表示でき、**#show file <>** は使用できません。
ステップ 9	（任意） **load** *filename* **例：** switch (config-dual-stage)# show configuration ! Cached configuration switch (config-dual-stage)# load test.cfg switch (config-dual-stage-router)# show configuration ! Cached configuration ! router bgp 1 switch(config-dual-stage-router)#	保存したターゲットコンフィギュレーションをロードします。ファイルをロードした後、ファイルを変更したり、実行コンフィギュレーションにコミットしたりできます。変更を保存するには、**save configuration** *filename* コマンドを使用します。 **save configuration** *filename* コマンドのみを使用して保存したターゲットコンフィギュレーションをロードできます。
ステップ 10	（任意） **clear configuration** **例：** switch(config-dual-stage)# show configuration ! Cached configuration	コンフィギュレーションセッションを終了せずに、ターゲット コンフィギュレーションに加えられた変更をクリアします。コミットされていない設定変更は削除されます。

	コマンドまたはアクション	目的
	```! router bgp 64515.46 router-id 141.8.139.131 switch (config-dual-stage)# clear configuration switch (config-dual-stage)# show configuration ! Cached configuration switch (config-dual-stage)#```	
ステップ 11	**end** 例： ```switch(config-dual-stage-if)# end Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]```	グローバル デュアル コンフィギュレーション モードを終了します。 設定変更をコミットせずにコンフィギュレーション セッションを終了すると、変更内容を保存するか、変更を破棄するか、または操作をキャンセルするように指示されます。 • はい：設定変更をコミットしてから、コンフィギュレーション モードを終了します。 • いいえ：設定変更をコミットせずに、コンフィギュレーション モードを終了します。 • キャンセル：設定変更をコミットせずに、コンフィギュレーション モードに留まります。 （注） • 確認コミット タイマーの実行中に終了することを選択した場合は、同じオプションが表示されます。終了を選択した場合、トライアル設定はすぐにロールバックされます。 • タイマーが期限切れになる前にデフォルト セッションがタイムアウトした場合、トライアル設定はセッションを終了する前にロールバックします。この場合、警告メッセージが表示されます。
ステップ 12	**show configuration dual-stage sessions** 例： ```switch(config-dual-stage)# show configuration dual-stage sessions SNo. Session Line User Date ---- ---------- ---------- ---------- ---------- 1 8671-17101913 /dev/ttyS0 admin Wed Feb 17 10:56:00 2021 switch(config-dual-stage)# end switch# show configuration dual-stage sessions There are no active dual stage sessions```	コンフィギュレーション セッションを開始する前に、進行中のその他のコンフィギュレーションセッションがないか確認する必要があります。シングル ユーザのみがデュアル ステージ コンフィギュレーション モードを開始できます。したがって、新しいセッションを開始する前に、前のセッションを終了する必要があります。最大 32 のインタラクティブ VSH セッションがあり、show コマンドはデュアル ステージ セッションの PID と回線情報を表示します。

	コマンドまたはアクション	目的
	`switch#`	（注） デュアル ステージ モードは、システムの準備完了後にのみアクセスできます。

ステップ 13	**clear configuration commits diskspace**
	例：
	`Southlake-2# clear configuration commits diskspace ?`
	`<1-20971> Number of Kilo Bytes of disk space to free`
	
	`Southlake-2# clear configuration commits diskspace 100`
	`Deleting 7 rollback points from '1000005557' to '1000005563'`
	`101 KB of disk space will be freed.`
	`Continue with deletion (yes/no)? [no] y`
	`Southlake-2#`
		EXEC モードまたは管理 EXEC モードで clear configuration commits コマンドを入力することにより、最も古い設定の commitID を削除できます。clear configuration commit コマンドの後ろには、解放するディスク スペースの量または削除する commitID の数を指定する必要があります。最も古い一連の commitID を削除して指定したディスク スペースを空けるには、ディスクスペース キーワードと再要求するキロバイト数の後ろに clear configuration commits コマンドを入力します。

ステップ 14	**clear configuration commits oldest**
	例：
	`switch(config-dual-stage)# clear configuration commits oldest 10`
	`Deleting 10 rollback points '1000000030' to '1000000039'`
	`125 KB of disk space will be freed.`
	`Continue with deletion (yes/no)? [no] n`
		最も古い方からの指定した回数分の commitID を削除するには、最も古いキーワードと削除する commitID 数の後ろに clear configuration commits コマンドを入力します。

ステップ 15	**Show configuration failed**
	例：
	`switch(config-dual-stage-if)# commit`
	`Verification Succeeded.`
	
	`Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer.`
	`Please avoid other configuration changes during this time.`
	`Failed to commit one or more configuration items.`
	`Commit Failed, Rolling back ...`
	`switch(config-dual-stage)#`
	`switch(config-dual-stage)# show configuration failed`
	`` `config terminal` ``
	`` `router bgp 100 ` ``
	`` `neighbor 2.2.2.2 ` ``
	`` `bfd ` ``
	`Syntax error while parsing 'bfd '`
	
	`` `neighbor 3.3.3.3 ` ``
	`` `bfd ` ``
	`Syntax error while parsing 'bfd '`
	
	`` `interface port-channel23 ` ``
		設定変更は、コミット操作中に意味的に検証され、検証が成功すると実際のバックエンド コミットが開始されます。コミット中に 1 つ以上の設定エントリが失敗すると、メッセージが表示されます。失敗したコンフィギュレーションのエラー メッセージと説明を表示するには、show configuration failed コマンドを入力します。これにより、最後のコミットで失敗した設定ブロックが表示されます。設定ブロックは、設定コンテキストを保持します。

	コマンドまたはアクション	目的

	`bfd `
	Syntax error while parsing 'bfd '
	
	`end`
	`end`
	switch(config-dual-stage)#	

ステップ 16	**show configuration failed noerrors**
	例：
	switch(config-dual-stage)# show configuration failed noerror
	router bgp 100
	  neighbor 2.2.2.2
	  bfd
	  neighbor 3.3.3.3
	  bfd
	interface port-channel23
	  bfd
	switch(config-dual-stage)#
		失敗したコンフィギュレーションブロックのエラー設定（説明なし）のみを表示するには、show configuration failed noerrors コマンドを入力します。

ステップ 17	**load configuration failed commit** 例： switch(config-dual-stage)# load configuration failed commit switch(config-dual-stage-if)# sh configuration ! Cached configuration ! router bgp 100 neighbor 2.2.2.2 bfd ! interface port-channel23 bfd switch(config-dual-stage-if)#	コミット中にルータが検証失敗メッセージを表示した場合、設定変更は失われません。ターゲット設定を変更し、再度コミットできます。設定変更をコミットしようとして、コンフィギュレーションが失敗したというメッセージがルータから表示された場合、その設定変更内容は失われません。デュアルステージ コンフィギュレーション モード留まっている間に、sいっぱい下コンフィギュレーション ブロックをターゲット コンフィギュレーションにリロードし、エラーを修正して、変更内容をコミットできます。 失敗した設定をロードするには、load configuration failed commit コマンドを入力します。コンフィギュレーションを回復して、修正、コミットするか、またはファイルに保存した後であれば、コンフィギュレーションが失われることはありません。ロード中、構文的に誤った設定は無視されることに注意してください。「show configuration」を使用してターゲット設定を表示できます。

2ステージコンフィギュレーション コミット モードの中止

コンフィギュレーション セッションを破棄すると、コミットされていない変更内容は破棄され、コンフィギュレーション セッションが終了します。設定変更は、警告なしに削除されます。

```
switch(config-dual-stage)# router bgp 1
switch(config-dual-stage-router)# neighbor 1.2.3.4
switch(config-dual-stage-router-neighbor)# remote-as 1
switch(config-dual-stage-router-neighbor)# show configuration
! Cached configuration
!
router bgp 1
neighbor 1.2.3.4
remote-as 1
switch(config-dual-stage-router-neighbor)# show run bgp

!Command: show running-config bgp
!Running configuration last done at: Wed Mar 17 16:17:40 2021
!Time: Wed Mar 17 16:17:55 2021

version 10.1(2) Bios:version
feature bgp


switch(config-dual-stage-router-neighbor)# abort
switch# show run bgp

!Command: show running-config bgp
!Running configuration last done at: Wed Mar 17 16:18:00 2021
!Time: Wed Mar 17 16:18:04 2021

version 10.1(2) Bios:version
feature bgp

switch#
```

コミット ID の表示

コミットが成功するたびに、コミット ID が syslog に表示されます。システムに保存されるコミット ID の総数は、設定サイズと使用可能なディスク領域によって異なります。ただし、任意の時点で保存されるコミット ID の最大数は 50 です。

最後の 50 のコミット ID に関する情報を表示するには、**show configuration commit list** コマンドを使用します。各エントリに、設定変更をコミットしたユーザ、コミットの実行に使用された接続、およびコミット ID のタイムスタンプが表示されます。

```
switch# show configuration commit list
SNo. Label/ID     User     Line         Client     Time Stamp
~~~~ ~~~~~~~~~~~~ ~~~~~~~~ ~~~~~~~~~~~~ ~~~~~~~~~~ ~~~~~~~~~~~~~~~~~~~~~~~~
1    1000000001   admin    /dev/ttyS0   CLI        Wed Jul 15 15:21:37 2020
2    1000000002   admin    /dev/ttyS0   Rollback   Wed Jul 15 15:22:15 2020
```

```
3    1000000003   admin    /dev/pts/0   CLI        Wed Jul 15 15:23:08 2020
4    1000000004   admin    /dev/pts/0   Rollback   Wed Jul 15 15:23:46 2020
```

ロールバック機能

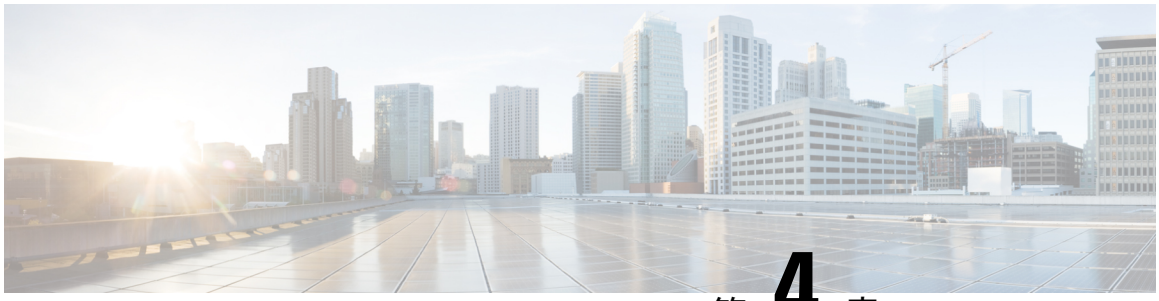
以前に成功したコミットのいずれかに設定をロールバックできます。**rollback configuration** コマンドを使用して、最後の 50 のコミットのいずれかにロールバックします。

```
switch# rollback configuration to ?
1000000015
1000000016
1000000017

:
:

switch#

Each commit ID acts as a (checkpoint or) rollback point. You can rollback to any given
commit ID. When you roll back the configuration to a specific rollback point, you undo
all configuration changes made during the session identified by the commitID for that
rollback point, and you undo all configuration changes made after that point. The
rollback process rolls back the configuration and commits the rolled-back configuration.
The rollback process also creates a new rollback point (commit ID)so that you can roll
back the configuration to the previous configuration.


switch(config-dual-stage)# rollback configuration to 1000000002
Rolling back to commitID :1000000002
ADVISORY: Rollback operation started...
Modifying running configuration from another VSH terminal in parallel
is not recommended, as this may lead to Rollback failure.

Configuration committed by rollback using Commit ID : 1000000004
switch(config-dual-stage)#
```

現在のセッション設定の表示

show configuration コマンドを使用して、現在のコンフィギュレーション セッションを表示できます。このコマンドは、デュアル ステージ モードでのみサポートされます。コミットが失敗すると、セッション設定はクリアされます。

```
switch(config-dual-stage-cmap)# show configuration
! Cached configuration
!
class-map type control-plane match-any copp-s-ipmcmiss
class-map type control-plane match-any copp-s-l2switched
class-map type control-plane match-any copp-s-l3destmiss
switch(config-dual-stage-cmap)#

If there is no configuration, the following message appears:

switch(config-dual-stage)# show configuration
! Cached configuration
switch(config-dual-stage)# commit
No configuration changes to commit.
switch(config-dual-stage)#
```



第 4 章

スイッチ プロファイルの設定

この章では、Cisco Nexus 9000 シリーズ スイッチでスイッチ プロファイルを設定する方法を説明します。

- スイッチ プロファイルの概要 （29 ページ）
- スイッチ プロファイルの注意事項および制約事項 （32 ページ）
- スイッチ プロファイルの設定 （34 ページ）
- スイッチ プロファイルのコマンドの追加または変更 （36 ページ）
- スイッチ プロファイルのインポート （38 ページ）
- vPC トポロジでの設定のインポート （40 ページ）
- ピア スイッチの分離 （41 ページ）
- スイッチ プロファイルの削除 （41 ページ）
- ミューテックスとマージの失敗の手動修正 （42 ページ）
- スイッチ プロファイル設定の確認 （43 ページ）
- スイッチ プロファイルの設定例 （43 ページ）

スイッチ プロファイルの概要

複数のアプリケーションは、ネットワーク内のデバイス間で整合性のある設定が必要です。たとえば、仮想ポート チャネル（vPC）のコンフィギュレーションを同じにする必要があります。コンフィギュレーションが一致しない場合、エラーやコンフィギュレーション エラーが生じる可能性があります。その結果、サービスが中断することがあります。設定の同期（config-sync）機能では、1 つのスイッチ プロファイルを設定し、設定を自動的にピア スイッチに同期させることができます。

スイッチ プロファイルには次の利点があります。

- スイッチ間でコンフィギュレーションを同期化できます。
- 2 つのスイッチ間で接続が確立されると、コンフィギュレーションがマージされます。
- どのコンフィギュレーションを同期化するかを完全に制御できます。
- マージチェックおよび相互排除チェックを使用して、ピア全体でコンフィギュレーションの一貫性を確保します。

- verify 構文および commit 構文を提供します。
- 既存の vPC 設定をスイッチ プロファイルに移行できます。

スイッチ プロファイル：コンフィギュレーション モード

スイッチ プロファイル機能には、次のコンフィギュレーション モードがあります。

- コンフィギュレーション同期モード（config-sync）
- スイッチ プロファイル モード（config-sync-sp）
- スイッチ プロファイル インポート モード（config-sync-sp-import）

コンフィギュレーション同期モード

コンフィギュレーション同期化モード（config-sync）を使用してスイッチプロファイルを作成できます。

スイッチ プロファイル モード

スイッチプロファイルモード（config-sync-sp）では、後でピア スイッチと同期化されるスイッチ プロファイル一時バッファに、サポートされているコンフィギュレーション コマンドを追加できます。スイッチ プロファイル モードで入力するコマンドは、**commit** コマンドを入力するまで実行されません。コマンドを入力すると、コマンドの構文が検証されますが、**commit** コマンドを入力したときにコマンドが正常に実行される保証はありません。

スイッチ プロファイル インポート モード

スイッチ プロファイル インポート モード（config-sync-sp-import）では、既存のスイッチ設定を実行コンフィギュレーションからスイッチプロファイルインポートし、どのコマンドをプロファイルに含めるかを指定できます。このオプションは、スイッチプロファイルをサポートしていないCisco NX-OS リリースからサポートしているリリースにアップグレードする場合に特に役立ちます。

スイッチ プロファイル インポート モードを使用して実行コンフィギュレーションから必要な設定をインポートし、スイッチ プロファイルまたはグローバル コンフィギュレーション モードで追加の変更を行う前に変更を確定することを推奨します。そうしないと、インポートが危険にさらされ、現在のインポートセッションを放棄してプロセスを再実行する必要が生じる場合があります。詳細については、「スイッチ プロファイルのインポート （38 ページ）」を参照してください。

コンフィギュレーションの検証

2種類のコンフィギュレーション検証チェックを使用して、スイッチプロファイルエラーを識別できます。

- 相互排除チェック

- マージ チェック

相互排除チェック

コンフィギュレーション コマンドの相互排除は、config-sync およびグローバル コンフィギュレーション モードでのコマンドの重複を避けるために適用されます。スイッチ プロファイルの設定をコミットすると、相互排除（mutex）チェックがローカル スイッチとピア スイッチ（設定されている場合）で実行されます。両方のスイッチで障害が報告されない場合、コミットは受け入れられ、実行コンフィギュレーションにプッシュされます。

スイッチ プロファイルに含まれるコマンドは、スイッチ プロファイル外に設定できます。

mutex チェックがエラーを識別すると、mutex の障害として報告され、手動で修正する必要があります。詳細は、ミューテックスとマージの失敗の手動修正 （42 ページ）を参照してください。

相互排除ポリシーには、次の例外が適用されます。

- インターフェイス コンフィギュレーション：インターフェイス コンフィギュレーションは、競合しない限り、スイッチプロファイルと実行コンフィギュレーションのそれぞれに部分的に含まれることができます。
- shutdown/no shutdown
- System QoS

マージ チェック

マージ チェックは、コンフィギュレーションを受信する側のピア スイッチで実行されます。マージチェックは、受信したコンフィギュレーションが、受信側のスイッチにすでに存在するスイッチ プロファイル コンフィギュレーションと競合しないようにします。マージ チェックは、確認プロセスまたはコミット プロセス中に実行されます。エラーはマージ エラーとして報告され、手動で修正する必要があります。詳細は、ミューテックスとマージの失敗の手動修正 （42 ページ）を参照してください。

1 つまたは両方のスイッチがリロードされ、コンフィギュレーションが初めて同期化される際には、マージ チェックによって、両方のスイッチのスイッチ プロファイル コンフィギュレーションが同じであることが検証されます。スイッチ プロファイルの相違はマージ エラーとして報告され、手動で修正する必要があります。

スイッチプロファイルを使用したソフトウェアのアップグレードとダウングレード

スイッチ プロファイルをサポートする Cisco NX-OS リリースからスイッチ プロファイルをサポートしない Cisco NX-OS リリースにダウングレードする場合、スイッチ プロファイルを削除する必要があります。

旧リリースからスイッチ プロファイルをサポートする Cisco NX-OS リリースにアップグレードする場合、実行コンフィギュレーション コマンドの一部をスイッチ プロファイルに移動することができます。詳細は、スイッチ プロファイル インポート モード （30 ページ）を参照してください。

バッファされた（コミットされていない）設定が存在する場合でもアップグレードを実行できますが、コミットされていないコンフィギュレーションは失われます。

スイッチ プロファイルの注意事項および制約事項

スイッチ プロファイルの注意事項および制約事項

- Cisco NX-OS リリース 9.3(3) 以降、**mtu** コマンドは、インターフェイス コンフィギュレーション モードでスイッチ プロファイル コンフィギュレーション モードを介してサポートされます。
- スイッチ プロファイルは Cisco Nexus 9300 シリーズ スイッチでのみサポートされます。Cisco Nexus 9500 シリーズ スイッチは、スイッチ プロファイルをサポートしていません。
- mgmt0 インターフェイスを使用してのみ設定同期化をイネーブルにできます。
- 仮想ピアリンク環境で config-sync を使用する場合は、次の制限事項に注意してください。
  - 仮想ピア リンクで config-sync セッションを開始するには、ピア スイッチ間で管理 IP アドレスの代わりにループバック IP アドレスを設定します。
  - マルチシャーシ EtherChannel トランク （MCT） 設定と仮想ピア リンク設定の間で設定の同期を実行することはできません。この config-sync 操作はサポートされていません。
- 同じスイッチ プロファイル名で同期されたピアを設定する必要があります。
- スイッチ プロファイル設定で使用可能なコマンドを、設定スイッチ プロファイル モード（config-sync-sp）で設定できます。
- サポートされているスイッチ プロファイル コマンドは、vPC コマンドに関連します。
- 1 つのスイッチプロファイルセッションのみを一度に進行できます。別のセッションの開始を試みると失敗します。
- スイッチ プロファイル セッションの進行中は、グローバル コンフィギュレーション モードから実行されたサポートされているコマンドの変更はブロックされます。
- **commit** コマンドを入力し、ピア スイッチに到達可能である場合、設定は、両方のピア スイッチに適用されるか、いずれのスイッチにも適用されません。コミットの障害が発生した場合、コマンドは、スイッチ プロファイル バッファに残ります。その場合、必要な修正をし、コミットを再試行します。
- コンフィギュレーション同期（**config-sync**）モードは、コンフィギュレーション ターミナル モード（**config t**）と同等の L2 モードです。config-sync は、スイッチプロファイルを使

用して、ピア スイッチと同じスイッチの **config t** モードを更新します。**switch-profile** モードでの同期の問題を防ぐために、現在の CLI コマンドを上書きまたは置換する前に、各 CLI コマンドの後にコミット アクションを実行することを推奨します。

たとえば、**CLI_command_A** を上書きして **CLI_command_B** に変更する場合は、まず **CLI_command_A** をコミットしてから、**CLI_command_B** を設定し、別のコミット アクションを実行します。

```
switch# conf sync
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-sync)# switch-profile test
Resyncing db before starting Switch-profile.Re-synchronization of switch-profile db
 takes a few minutes...
Re-synchronize switch-profile db completed successfully.
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)#
switch(config-sync-sp)# int e 1/3
switch(config-sync-sp-if)# switchport trunk allowed vlan 100-150
switch(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch(config-sync)#
switch(config-sync)# switch-profile test
Resyncing db before starting Switch-profile.Re-synchronization of switch-profile db
 takes a few minutes...
Re-synchronize switch-profile db completed successfully.
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)#
switch(config-sync-sp)# int e 1/3
switch(config-sync-sp-if)# switchport trunk allowed vlan 45-90
switch(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch(config-sync)# end
switch#
```

- レイヤ 3 コマンドはサポートされていません。

config-sync 機能には、次の注意事項と制約事項があります。

- スイッチ プロファイル モードで作成されるポート チャネルは、グローバル コンフィギュレーション（config terminal）モードを使用して設定することはできません。
- ポート チャネルをグローバル コンフィギュレーション モードで作成した場合は、メンバー インターフェイスを含むチャネル グループも、グローバル コンフィギュレーション モードを使用して作成する必要があります。
- スイッチ プロファイル モードで設定されたポート チャネルには、スイッチ プロファイルの内部と外部どちらからもメンバーにすることができます。

- メンバ インターフェイスをスイッチ プロファイルにインポートする場合は、そのメンバ インターフェイスに対応するポート チャネルがスイッチ プロファイル内に存在する必要があります。
- グローバル レベルでの「no system default switchport」設定の場合、port-channel の下の「switchport」コマンドも相互排除と見なされます。

スイッチ プロファイルの設定

ローカル スイッチでスイッチ プロファイルを作成および設定し、同期に含まれる 2 番目のスイッチを追加することができます。

スイッチプロファイルは、各スイッチで同じ名前を使用して作成する必要があります。また、スイッチは互いにピアとして設定する必要があります。同じアクティブなスイッチプロファイルが設定されたスイッチ間で接続が確立されると、スイッチプロファイルが同期化されます。

手順

ステップ 1 **configure terminal**

例：

```
switch# configure terminal
switch(config)#
```

グローバル コンフィギュレーション モードを開始します

ステップ 2 必須: **cfs ipv4 distribute**

例：

```
switch(config)# cfs ipv4 distribute
```

ピア スイッチ間の Cisco Fabric Services（CFS）配信を有効にします。

ステップ 3 必須: **config sync**

例：

```
switch(config)# config sync
switch(config-sync)#
```

コンフィギュレーション同期モードを開始します。

ステップ 4 必須: **switch-profile** *name*

例：

```
switch(config-sync)# switch-profile abc
switch(config-sync-sp)#
```

スイッチ プロファイルを設定し、スイッチ プロファイルの名前を設定し、スイッチ プロファイル コンフィギュレーション モードを開始します。

ステップ 5 必須: [**no**] **sync-peers destination** *ip-address*

例：

```
switch(config-sync-sp)# sync-peers destination 10.1.1.1
```

スイッチ プロファイルにスイッチを追加します。宛先 IP アドレスは、同期するスイッチの IP アドレスです。

このコマンドの **no** 形式でスイッチ プロファイルから指定のスイッチを削除します。

（注）
コミットが完了する前に、ピア スイッチがスイッチプロファイル ステータス「In sync」を表示するまで待機する必要があります。

ステップ 6 必須: Cisco Nexus 3164Q スイッチの場合のみ、次の手順を実行します。

a) **interface** *type slot/port*

例：

```
switch(config-sync-sp)# interface ethernet 1/1
switch(config-sync-sp-if)#
```

スイッチ プロファイル 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。

b) **switchport**

例：

```
switch(config-sync-sp-if)# switchport
```

レイヤ 3 インターフェイスをレイヤ 2 インターフェイスに変更します。

c) **exit**

例：

```
switch(config-sync-sp-if)# exit
switch(config-sync-sp)#
```

スイッチ プロファイル インターフェイス コンフィギュレーション モードを終了します。

d) **commit**

例：

```
switch(config-sync-sp)# commit
```

現在の設定をコミットします。

（注）
コミットが完了する前に、スイッチプ ロファイルのステータスが「In sync」と表示されていることを確認します。

ステップ 7 （任意） **end**

例：

```
switch(config-sync-sp)# end
switch#
```

スイッチ プロファイル コンフィギュレーション モードを終了し、EXEC モードに戻ります。

ステップ 8 （任意） **show switch-profile** *name* **status**

例：

```
switch# show switch-profile abc status
```

ローカル スイッチのスイッチ プロファイルおよびピア スイッチ情報を表示します。

ステップ 9 （任意） **show switch-profile** *name* **peer** *ip-address*

例：

```
switch# show switch-profile abc peer 10.1.1.1
```

スイッチ プロファイルのピアの設定を表示します。

ステップ 10 （任意） **copy running-config startup-config**

例：

```
switch# copy running-config startup-config
```

実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

スイッチ プロファイルのコマンドの追加または変更

ローカルおよびピア スイッチでスイッチ プロファイルを設定したら、スイッチ プロファイルにサポートされているコマンドを追加し、コミットする必要があります。

追加または変更されたコマンドは、**commit** コマンドを入力するまでバッファに格納されます。コマンドは、バッファリングされた順序で実行されます。特定のコマンドに順序の依存関係がある場合（たとえば、QoS ポリシーは適用前に定義する必要がある）、その順序を維持する必要があります。そうしないとコミットに失敗する可能性があります。**show switch-profile** *name* **buffer** コマンド、**buffer-delete** コマンド、**buffer-move** コマンドなどのユーティリティ コマンドを使用して、バッファを変更し、入力済みのコマンドの順序を修正できます。

手順の概要

1. **config sync**
2. **switch-profile** *name*
3. *command*
4. （任意） **show switch-profile** *name* **buffer**
5. **verify**
6. **commit**
7. （任意） **end**
8. （任意） **show switch-profile** *name* **status**
9. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	必須: **config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 2	必須: **switch-profile** *name*
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		スイッチプロファイルを設定し、スイッチプロファイルの名前を設定し、スイッチ プロファイル コンフィギュレーション モードを開始します。

ステップ 3	必須: *command*
	例：
	`switch(config-sync-sp)# interface Port-channel100`
	`switch(config-sync-sp-if)# speed 1000`
	`switch(config-sync-sp-if)# interface Ethernet1/1`
	`switch(config-sync-sp-if)# speed 1000`
	`switch(config-sync-sp-if)# channel-group 100`
	`switch(config-sync-sp-if)# exit`
	`switch(config-sync-sp)#`
		スイッチ プロファイルにコマンドを追加します。

ステップ 4	（任意） **show switch-profile** *name* **buffer** 例： `switch(config-sync-sp)# show switch-profile abc buffer`	スイッチ プロファイル バッファ内のコンフィギュレーション コマンドを表示します。
ステップ 5	必須: **verify** 例： `switch(config-sync-sp)# verify`	スイッチ プロファイル バッファ内のコマンドを確認します。
ステップ 6	必須: **commit** 例： `switch(config-sync-sp)# commit`	スイッチ プロファイルにコマンドを保存し、ピアスイッチと設定を同期します。このコマンドは、次のことも行います。 • mutex チェックとマージ チェックを起動し、同期を確認します。 • ロールバック インフラストラクチャでチェックポイントを作成します。 • スイッチプロファイル内の任意のスイッチでアプリケーション障害がある場合は、すべてのスイッチでロール バックを実行します。

	コマンドまたはアクション	目的
		• チェックポイントを削除します。

ステップ 7	（任意） **end**
	例：
	`switch(config-sync-sp)# end`
	`switch#`
		スイッチプロファイルコンフィギュレーションモードを終了し、EXEC モードに戻ります。

ステップ 8	（任意） **show switch-profile** *name* **status** 例： `switch# show switch-profile abc status`	ローカルスイッチのスイッチプロファイルのステータスとピア スイッチのステータスを表示します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

スイッチ プロファイルのインポート

インポートするコマンドのセットに基づいてスイッチ プロファイルをインポートできます。

始める前に

コマンドをスイッチ プロファイルにインポートする前に、スイッチ プロファイル バッファが空であることを確認します。

手順の概要

1. （任意） ステップ 4 でインポートするインターフェイスを設定します。
2. **config sync**
3. **switch-profile** *name*
4. **import** [**interface** *interface port*/*slot* | **running-config**]
5. **commit**
6. （任意） **abort**
7. （任意） **end**
8. （任意） **show switch-profile**
9. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	（任意）ステップ 4 でインポートするインターフェイスを設定します。
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)# switchport`
	`switch(config-if)# switchport mode trunk`
	`switch(config-if)# switchport trunk allowed vlan 12`
	`switch(config-if)# speed 10000`
	`switch(config-if)# spanning-tree port type edge trunk`
	`switch(config)# end`
	`switch#`
		コンフィギュレーション同期モードを開始します。

ステップ 2	**config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 3	必須: **switch-profile** *name*
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		スイッチプロファイルを設定し、スイッチプロファイルの名前を設定し、スイッチ プロファイル コンフィギュレーション モードを開始します。

ステップ 4	必須: **import** [**interface** *interface port*/*slot* | **running-config**]
	例：
	`switch(config-sync-sp)# import interface ethernet 1/2`
	`switch(config-sync-sp-import)#`
		インポートするコマンドを識別し、スイッチ プロファイル インポート モードを開始します。次のオプションを使用できます。
		• オプションを指定せずに **import** コマンドを入力すると、選択したコマンドがスイッチプロファイルに追加されます。
		• **import interface** オプションは、指定されたインターフェイスでサポートされるコマンドを追加します。
		• **running-config** オプションでは、サポートされるシステムレベル コマンドを追加します。
		（注）
		新しいコマンドがインポート中に追加されると、スイッチ プロファイルが保存されていないままになり、スイッチはスイッチ プロファイル インポート モードのままになります。

	コマンドまたはアクション	目的
ステップ 5	必須: **commit** 例： `switch(config-sync-sp-import)# commit`	コマンドをインポートし、スイッチプロファイルにコマンドを保存します。
ステップ 6	（任意） **abort** 例： `switch(config-sync-sp-import)# abort`	インポート プロセスを中止します。

ステップ 7	（任意） **end**
	例：
	`switch(config-sync-sp-import)# end`
	`switch#`
		スイッチプロファイルインポートモードを終了し、EXEC モードに戻ります。

ステップ 8	（任意） **show switch-profile** 例： `switch# show switch-profile`	スイッチ プロファイル コンフィギュレーションを表示します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

vPC トポロジでの設定のインポート

2 スイッチ vPC トポロジで設定をインポートできます。

（注） 次の手順の詳細については、この章の該当する項を参照してください。

1. 両方のスイッチで、同じ名前を持つスイッチ プロファイルを設定します。
2. 両方のスイッチに設定を個別にインポートします。

（注） 両方のスイッチで、スイッチプロファイルに移動された設定が同じであることを確認します。同じでない場合、マージ チェックの障害が発生する場合があります。

3. **sync-peer destination** コマンドを入力してスイッチを設定します。
4. 適切な **show** コマンドを入力して、スイッチプロファイルが同一であることを確認します。

ピア スイッチの分離

スイッチ プロファイルを変更するためにピア スイッチを分離できます。このプロセスは、設定の同期をブロックしたり、設定をデバッグしたり、設定同期機能が同期しなくなった状況から回復したりする場合に使用できます。

ピア スイッチを分離するには、スイッチ プロファイルからピア接続をブレークし、スイッチ プロファイルにピア スイッチを追加する必要があります。

（注） 次の手順の詳細については、この章の該当する項を参照してください。

1. 両方のスイッチでスイッチ プロファイルからピア スイッチを削除できます。
2. **no sync-peers destination** コマンドをスイッチ プロファイルに追加し、両方のスイッチで変更をコミットします。
3. 必要なトラブルシューティング設定を追加します。
4. show running switch-profile が両方のスイッチで同一であることを確認します。
5. **sync-peers destination** *ip-address* コマンドを両方のスイッチに追加して、変更をコミットします。
6. ピアが同期中であることを確認します。

スイッチ プロファイルの削除

スイッチ プロファイルを削除できます。

手順の概要

1. **config sync**
2. **no switch-profile** *name* {**all-config** | **local-config**}
3. （任意） **end**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**config sync** 例：	コンフィギュレーション同期モードを開始します。

	コマンドまたはアクション	目的
	`switch# config sync` `switch(config-sync)#`	

ステップ 2	必須: **no switch-profile** *name* {**all-config** | **local-config**}
	例：
	`switch(config-sync)# no switch-profile abc local-config`
	`switch(config-sync-sp)#`
		次の手順に従って、スイッチプロファイルを削除します。
		• **all-config**：ローカル スイッチおよびピア スイッチのスイッチプロファイルを削除します。ピア スイッチが到達可能でない場合は、ローカル スイッチ プロファイルだけが削除されます。
		• **local-config**：スイッチ プロファイルおよびローカル コンフィギュレーションを削除します。
		（注）
		スイッチ プロファイルを削除する前に、**resync-database** を実行することを推奨します。
		`switch(config-sync)# resync-database`

ステップ 3	（任意） **end**
	例：
	`switch(config-sync-sp)# end`
	`switch#`
		スイッチプロファイル コンフィギュレーション モードを終了し、EXEC モードに戻ります。

ステップ 4	（任意） **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。このコマンドを入力すると、config-sync 機能がピア スイッチで同じ動作をトリガします。

ミューテックスとマージの失敗の手動修正

ミューテックスとマージの障害が発生した場合は、手動で修正できます。

（注） ピア スイッチで競合が発生している場合は、ピア スイッチの分離 （41 ページ） の手順に従ってそのスイッチの問題を修正します。

1. スイッチ プロファイル インポート モードを使用して、問題のコマンドをスイッチ プロファイルにインポートします。
2. 必要に応じて動作を変更します。

スイッチ プロファイル設定の確認

スイッチ プロファイルの関する情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show switch-profile *name*	スイッチ プロファイル中のコマンドを表示します。
show switch-profile *name* **buffer**	スイッチプロファイル中のコミットされていないコマンド、移動されたコマンド、削除されたコマンドを表示します。
show switch-profile *name* **peer** *ip-address*	ピア スイッチの同期ステータスが表示されます。
show switch-profile *name* **session-history**	最後の 20 のスイッチ プロファイル セッションのステータスを表示します。
show switch-profile *name* **status**	ピア スイッチのコンフィギュレーション同期ステータスを表示します。
show running-config switch-profile	ローカル スイッチのスイッチ プロファイルの実行コンフィギュレーションを表示します。
show startup-config switch-profile	ローカル スイッチのスイッチ プロファイルのスタートアップ コンフィギュレーションを表示します。

スイッチ プロファイルの設定例

ローカルおよびピア スイッチでのスイッチ プロファイルの作成...

次に、ローカルおよびピア スイッチで正常にスイッチ プロファイル設定を作成する例を示します。これには QoS ポリシー（vPC ピアリンクおよびスイッチ プロファイル中の vPC）の設定が含まれます。

1. ローカルおよびピア スイッチで CFS 配信を有効にし、スイッチの管理インターフェイスなど、同期するスイッチの宛先 IP アドレスを設定します。

```
—Local switch-1#---
switch-1# configure terminal
switch-1(config)# cfs ipv4 distribute
switch-1(config)# interface mgmt 0
switch-1(config-if)# ip address 30.0.0.81/8

—Peer switch-2#--
switch-2# configure terminal
switch-2(config)# cfs ipv4 distribute
switch-2(config)# interface mgmt 0
```

```
switch-2(config-if)# ip address 30.0.0.82/8
```

2. ローカルおよびピア スイッチで新しいスイッチ プロファイルを作成します。

```
—Local switch-1#---
switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# sync-peers destination 30.0.0.82
switch-1(config-sync-sp)# end

—Peer switch-2#--
switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# sync-peers destination 30.0.0.81
switch-1(config-sync-sp)# end
```

3. スイッチプロファイルが、ローカルおよびピア スイッチで同じであることを確認します。

```
switch-1(config-sync-sp)# show switch-profile status

switch-profile : A
----------------------------------------------------------

Start-time: 843992 usecs after Wed Aug 19 17:00:01 2015
End-time: 770051 usecs after Wed Aug 19 17:00:03 2015

Profile-Revision: 1
Session-type: Initial-Exchange
Session-subtype: Init-Exchange-All
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 30.0.0.82
Sync-status: In sync
Status: Commit Success
Error(s):
```

4. ローカル スイッチでスイッチ プロファイルにコンフィギュレーション コマンドを追加します。コマンドがコミットされたときに、コマンドがピア スイッチに適用されます。

```
switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface port-channel 10
switch-1(config-sync-sp-if)# switchport
switch-1(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch-1(config-sync)# switch-profile A
```

```
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface port-channel 10
switch-1(config-sync-sp-if)# switchport mode trunk
switch-1(config-sync-sp-if)# switchport trunk allowed vlan 10
switch-1(config-sync-sp-if)# spanning-tree port type network
switch-1(config-sync-sp-if)# vpc peer-link
switch-1(config-sync-sp-if)# switch-profile switching-mode switchname
switch-1(config-sync-sp-if)# show switch-profile buffer

switch-profile : A
-----------------------------------------------------------
Seq-no Command
-----------------------------------------------------------
1 interface port-channel10
1.1 switchport mode trunk
1.2 switchport trunk allowed vlan 10
1.3 spanning-tree port type network
1.4 vpc peer-link

switch-1(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface ethernet 2/1
switch-1(config-sync-sp-if)# switchport mode trunk
switch-1(config-sync-sp-if)# switchport trunk allowed vlan 10
switch-1(config-sync-sp-if)# spanning-tree port type network
switch-1(config-sync-sp-if)# channel-group 10 mode active
```

5. バッファリングされたコマンドを表示します。

```
switch-1(config-sync-sp-if)# show switch-profile buffer

switch-profile : A
-----------------------------------------------------------
Seq-no Command
-----------------------------------------------------------
1 interface Ethernet2/1
1.1 switchport mode trunk
1.2 switchport trunk allowed vlan 10
1.3 spanning-tree port type network
1.4 channel-group 10 mode active
```

6. スイッチ プロファイルのコマンドを検証します。

```
switch-1(config-sync-sp-if)# verify
Verification Successful
```

7. スイッチ プロファイルにコマンドを適用し、ローカルとピア スイッチ間の設定を同期させます。

```
—Local switch-2#--
switch-1(config-sync-sp)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
```

```
Commit Successful
switch-1(config-sync)# end

switch-1# show running-config switch-profile

switch-profile A
sync-peers destination 30.0.0.82

interface port-channel10
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
vpc peer-link

interface Ethernet2/1
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
channel-group 10 mode active


—Peer switch-2#--
switch-2# show running-config switch-profile

switch-profile A
sync-peers destination 30.0.0.81

interface port-channel10
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
vpc peer-link

interface Ethernet2/1
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
channel-group 10 mode active
```

同期ステータスの確認

次に、ローカルとピア スイッチ間の同期ステータスを確認する例を示します。

```
switch-1# show switch-profile status

switch-profile : A
-------------switch-1-----------------------------------------------

Start-time: 912776 usecs after Wed Aug 19 17:03:43 2015
End-time: 868379 usecs after Wed Aug 19 17:03:48 2015

Profile-Revision: 4
Session-type: Commit
Session-subtype: -
Peer-triggered: No
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):
```

```
Peer information:
----------------
IP-address: 30.0.0.82
Sync-status: In sync
Status: Commit Success
Error(s):
```

実行中のコンフィギュレーションの表示

次に、ローカル スイッチでスイッチ プロファイルの実行コンフィギュレーションを表示する方法の例を示します。

```
—— PEER SWITCH-1 ——
switch-1# show running-config switch-profile

switch-profile A
 sync-peers destination 30.0.0.82

 interface port-channel10
 switchport mode trunk
 switchport trunk allowed vlan 10
 spanning-tree port type network
 vpc peer-link

 interface Ethernet2/1
 switchport mode trunk
 switchport trunk allowed vlan 10
 spanning-tree port type network
 channel-group 10 mode active
switch-1#

—— PEER SWITCH-2 ——
switch-2# show running-config switch-profile

switch-profile A
 sync-peers destination 30.0.0.81

 interface port-channel10
 switchport mode trunk
 switchport trunk allowed vlan 10
 spanning-tree port type network
 vpc peer-link

 interface Ethernet2/1
 switchport mode trunk
 switchport trunk allowed vlan 10
 spanning-tree port type network
 channel-group 10 mode active
switch-2#
```

ローカルとピア スイッチ間のスイッチ プロファイルの同期の表示

次に、2 台のピア間の最初の正常な同期を表示する例を示します。

```
switch1# show switch-profile sp status

Start-time: 491815 usecs after Mon Jul 20 11:54:51 2015
End-time: 449475 usecs after Mon Jul 20 11:54:58 2015
```

```
Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: No
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.52
Sync-status: In Sync.
Status: Commit Success
Error(s):


switch2# show switch-profile sp status

Start-time: 503194 usecs after Mon Jul 20 11:54:51 2015
End-time: 532989 usecs after Mon Jul 20 11:54:58 2015

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.51
Sync-status: In Sync.
Status: Commit Success
Error(s):
```

ローカルおよびピア スイッチでの確認とコミットの表示

次に、ローカル スイッチおよびピア スイッチで正常に確認とコミットを実行する例を示します。

```
switch1# config sync
switch1(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch1(config-sync-sp)# interface Ethernet1/1
switch1(config-sync-sp-if)# description foo
switch1(config-sync-sp-if)# exit
switch1(config-sync-sp)# verify
Verification Successful
switch1(config-sync-sp)# commit
Commit Successful
switch1(config-sync)# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.52
  interface Ethernet1/1
    description foo
```

```
switch1(config-sync)# show switch-profile sp status

Start-time: 171513 usecs after Wed Jul 20 17:51:28 2015
End-time: 676451 usecs after Wed Jul 20 17:51:43 2015

Profile-Revision: 3
Session-type: Commit
Peer-triggered: No
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.52
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch1(config-sync)#


switch2# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.51
  interface Ethernet1/1
    description foo
switch2# show switch-profile sp status

Start-time: 265716 usecs after Mon Jul 20 16:51:28 2015
End-time: 734702 usecs after Mon Jul 20 16:51:43 2015

Profile-Revision: 3
Session-type: Commit
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.51
Sync-status: In Sync.
Status: Commit Success
Error(s):
```

ローカルおよびピア スイッチ間の成功および失敗した同期の表示

次に、ピア スイッチでスイッチ プロファイルの同期ステータスを設定する例を示します。最初の例は正常な同期を示し、2 番目の例はピアの到達不能な状態を示します。

```
switch1# show switch-profile sp peer

switch1# show switch-profile sp peer 10.193.194.52
Peer-sync-status           : In Sync.
```

```
Peer-status                : Commit Success
Peer-error(s)              :
switch1#


switch1# show switch-profile sp peer 10.193.194.52
Peer-sync-status           : Not yet merged. pending-merge:1 received_merge:0
Peer-status                : Peer not reachable
Peer-error(s)              :
```

スイッチ プロファイル バッファの表示

次に、スイッチ プロファイル バッファの設定、バッファ移動、バッファ削除を設定する例を示します。

```
switch1# config sync
switch1(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch1(config-sync-sp)# vlan 101
switch1(config-sync-sp-vlan)# ip igmp snooping querier 10.101.1.1
switch1(config-sync-sp-vlan)# exit
switch1(config-sync-sp)# mac address-table static 0000.0000.0001 vlan 101 drop
switch1(config-sync-sp)# interface Ethernet1/2
switch1(config-sync-sp-if)# switchport mode trunk
switch1(config-sync-sp-if)# switchport trunk allowed vlan 101
switch1(config-sync-sp-if)# exit
switch1(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
1       vlan 101
1.1       ip igmp snooping querier 10.101.1.1
2       mac address-table static 0000.0000.0001 vlan 101 drop
3       interface Ethernet1/2
3.1       switchport mode trunk
3.2       switchport trunk allowed vlan 101

switch1(config-sync-sp)# buffer-move 3 1
switch1(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
1       interface Ethernet1/2
1.1       switchport mode trunk
1.2       switchport trunk allowed vlan 101
2       vlan 101
2.1       ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop

switch1(config-sync-sp)# buffer-delete 1
switch1(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
2       vlan 101
2.1       ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop

switch1(config-sync-sp)# buffer-delete all
```

```
switch1(config-sync-sp)# show switch-profile sp buffer
```

設定のインポート

次に、インターフェイス コンフィギュレーションをインポートする例を示します。

```
switch# show running-config interface Ethernet1/3

!Command: show running-config interface Ethernet1/3
!Time: Wed Jul 20 18:12:44 2015

version 7.0(3)I2(1)

interface Ethernet1/3
  switchport mode trunk
  switchport trunk allowed vlan 1-100

switch# config sync
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1

switch(config-sync-sp)# import interface Ethernet1/3
switch(config-sync-sp-import)# show switch-profile sp buffer
-----------------------------------------------------------
Seq-no  Command
-----------------------------------------------------------
1       interface Ethernet1/3
1.1       switchport mode trunk
1.2       switchport trunk allowed vlan 1-100

switch(config-sync-sp-import)# verify
Verification Successful
switch(config-sync-sp-import)# commit
Commit Successful
```

次に、実行コンフィギュレーションにサポートされるコマンドをインポートする例を示します。

```
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# import running-config
switch(config-sync-sp-import)# show switch-profile sp buffer
-----------------------------------------------------------
Seq-no  Command
-----------------------------------------------------------
1       logging event link-status default
2       vlan 1
3       interface port-channel 3
3.1       switchport mode trunk
3.2       vpc peer-link
3.3       spanning-tree port type network
4       interface port-channel 30
4.1       switchport mode trunk
4.2       vpc 30
4.3       switchport trunk allowed vlan 2-10
5       interface port-channel 31
5.1       switchport mode trunk
5.2       vpc 31
5.3       switchport trunk allowed vlan 11-20
6       interface port-channel 101
```

```
6.1       switchport mode fex-fabric
6.2       fex associate 101
7       interface port-channel 102
7.1       switchport mode fex-fabric
7.2       vpc 102
7.3       fex associate 102
8       interface port-channel 103
8.1       switchport mode fex-fabric
8.2       vpc 103
8.3       fex associate 103
9      interface Ethernet1/1
10      interface Ethernet1/2
11      interface Ethernet1/3
12      interface Ethernet1/4
12.1      switchport mode trunk
12.2      channel-group 3
13      interface Ethernet1/5
13.1      switchport mode trunk
13.2      channel-group 3
14      interface Ethernet1/6
14.1      switchport mode trunk
14.2      channel-group 3
15      interface Ethernet1/7
15.1      switchport mode trunk
15.2      channel-group 3
16      interface Ethernet1/8
17      interface Ethernet1/9
17.1      switchport mode trunk
17.2      switchport trunk allowed vlan 11-20
17.3      channel-group 31 mode active
18      interface Ethernet1/10
18.1      switchport mode trunk
18.2      switchport trunk allowed vlan 11-20
18.3      channel-group 31 mode active
19      interface Ethernet1/11
20      interface Ethernet1/12
...
45      interface Ethernet2/4
45.1      fex associate 101
45.2      switchport mode fex-fabric
45.3      channel-group 101
46      interface Ethernet2/5
46.1      fex associate 101
46.2      switchport mode fex-fabric
46.3      channel-group 101
47      interface Ethernet2/6
47.1      fex associate 101
47.2      switchport mode fex-fabric
47.3      channel-group 101
48      interface Ethernet2/7
48.1      fex associate 101
48.2      switchport mode fex-fabric
48.3      channel-group 101
49      interface Ethernet2/8
49.1      fex associate 101
...
89      interface Ethernet100/1/32
90      interface Ethernet100/1/33
91      interface Ethernet100/1/34
92      interface Ethernet100/1/35
93      interface Ethernet100/1/36
...
105     interface Ethernet100/1/48
```

ファブリック エクステンダのストレート型トポロジでの Cisco NX-OS リリース 7.0(3)I2(1) 以降への移行

この例では、ファブリック エクステンダのアクティブ/アクティブ トポロジまたはストレート型 トポロジで Cisco NX-OS リリース 7.0(3)I2(1) 以降に移行するために使用するタスクを示します。タスクの詳細については、この章の該当する項を参照してください。

1. 両方のスイッチで設定が同じであることを確認します。
2. 両方のスイッチで、同じ名前を持つスイッチ プロファイルを設定します。
3. 両方のスイッチのすべての vPC ポート チャネルについて、**import interface port-channel** *x-y*, **port-channel** *z* コマンドを入力します。
4. **show switch-profile** *name* **buffer** コマンドを入力し、すべての設定が両方のスイッチで正しくインポートされていることを確認します。
5. バッファを編集して不要な設定を削除します。
6. 両方のスイッチで **commit** コマンドを入力します。
7. **sync-peers destination** *ip-address* コマンドを入力して、両方のスイッチでピア スイッチを設定します。
8. **show switch-profile** *name* **status** コマンドを入力して、両方のスイッチが同期状態であることを確認します。

Cisco Nexus 9000 シリーズ スイッチの交換

Cisco Nexus 9000 シリーズ スイッチを交換する場合、交換するスイッチで次の設定手順を実行し、既存の Cisco Nexus 9000 シリーズ スイッチと同期する必要があります。この手順は、ハイブリッド ファブリック エクステンダのアクティブ/アクティブトポロジとファブリック エクステンダ ストレート型トポロジで実行できます。

1. ピア リンク、vPC、アクティブ/アクティブ、またはストレート型のトポロジ ファブリック ポートを交換用スイッチに接続しないでください。
2. 交換するスイッチを起動します。スイッチは設定なしで起動します。
3. 交換スイッチを設定します。
   - 実行コンフィギュレーションがオフラインで保存された場合は、手順 4 ～ 8 に進み、設定を適用します。
   - 実行コンフィギュレーションがオフラインで保存されなかった場合で、設定同期機能がイネーブルの場合、ピア スイッチから実行コンフィギュレーションを取得できます（ローカルおよびピア スイッチでのスイッチ プロファイルの作成... （43 ページ） の手順 1 および 2 を参照してください。その後、手順 9 から開始します）。

- いずれの条件にも当てはまらない場合は、手動で設定を追加し、以下の手順 9 に進みます。

4. 設定同期機能を使用している場合は、コンフィギュレーション ファイルを編集し、**sync-peer** コマンドを削除します。
5. mgmt ポート IP アドレスを設定し、コンフィギュレーション ファイルをダウンロードします。
6. 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。
7. **show running-config** コマンドを入力して、コンフィギュレーションが正しいことを確認します。
8. 交換スイッチが動作していない間に、ピア スイッチでスイッチ プロファイルの設定が変更された場合、スイッチ プロファイルでこれらの設定を適用して、**commit** コマンドを入力します。
9. vPC トポロジに含まれるすべてのファブリック エクステンダ ストレート型トポロジ ポートをシャットダウンします。
10. ファブリック エクステンダ ストレート型トポロジ ファブリック ポートを接続します。
11. ファブリック エクステンダ ストレート型トポロジ スイッチがオンラインになるまで待ちます。
12. 既存スイッチの vPC のロールプライオリティが、交換スイッチよりも上位であることを確認します。
13. ピア リンク ポートをピア スイッチに接続します。
14. スイッチ vPC ポートを接続します。
15. すべてのファブリック エクステンダ ストレート型 vPC ポートで、**no shutdown** コマンドを入力します。
16. 交換スイッチにあるすべての vPC スイッチおよびファブリック エクステンダ がオンラインになり、トラフィックに中断がないことを確認します。
17. 設定同期機能を使用している場合、手順 3 で有効にされなかった場合は、sync-peer の設定をスイッチ プロファイルに追加します。
18. コンフィギュレーション同期機能を使用している場合、**show switch-profile** *name* **status** コマンドを使用し、両方のスイッチが同期されるようにします。

設定の同期

Cisco Nexus 9000 シリーズ スイッチのリブート後の設定の同期化

スイッチプロファイルを使用して新しい設定がピア スイッチでコミットされている中でCisco Nexus 9000 シリーズ スイッチがリブートする場合、 これらの手順に従いリロード後にピア スイッチを同期します。

1. 両方のスイッチでスイッチ プロファイルからピア スイッチを削除できます。
2. **no sync-peers destination** コマンドをスイッチ プロファイルに追加し、両方のスイッチで変更をコミットします。
3. 欠落または変更されたコマンドを追加します。
4. show running switch-profile が両方のスイッチで同一であることを確認します。
5. **sync-peers destination** *ip-address* コマンドを両方のスイッチに追加して、変更をコミットします。
6. ピアが同期中であることを確認します。

mgmt0 インターフェイスの接続が失われた場合の設定の同期化

mgmt0 インターフェイスの接続が失われ、設定変更が必要な場合は、スイッチ プロファイルを使用して、両方のスイッチに設定変更を適用します。mgmt0インターフェイスへの接続が復元されると、両方のスイッチが同期されます。

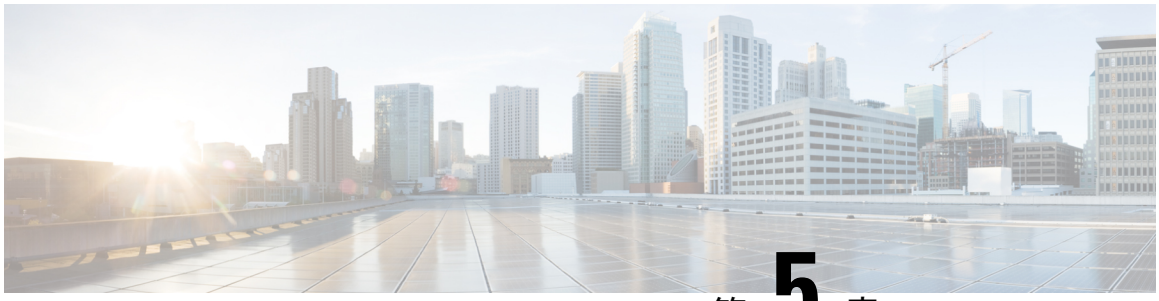
このシナリオで設定変更が1台のスイッチのみで実行された場合、マージは、mgmt0インターフェイスが起動し、設定が他のスイッチに適用されたときに成功します。

グローバル コンフィギュレーション モードでレイヤ 2 からレイヤ 3 への不注意によるポート モードの変更を元に戻す

config-syncモードでインポートされたポートに関連する設定は、グローバル コンフィギュレーション モードで設定しないでください。通常、そのような試みはconfig-sync 機能によって拒否され、mutex 警告が表示されます。ただし、mutex チェックの制限により、config-sync モードでレイヤ 2 として設定されたポートが、グローバル コンフィギュレーション モードでレイヤ3（スイッチポートなし）に変更された場合、config-sync機能は検出および防止できません。その結果、config-sync モードがグローバル コンフィギュレーション モードと同期しなくなる可能性があります。この場合は、次の手順に従って変更を元に戻します。

1. 両方のスイッチでスイッチ プロファイルからピア スイッチを削除できます。
2. **no sync-peers destination** コマンドをスイッチ プロファイルに追加し、両方のスイッチで変更をコミットします。
3. 現在のインターフェイス設定をインポートします。
4. 必要な変更を加えてコミットします。

5. show running switch-profile が両方のスイッチで同一であることを確認します。
6. **sync-peers destination** *ip-address* コマンドを両方のスイッチに追加して、変更をコミットします。
7. ピアが同期中であることを確認します。



第 5 章

周波数の同期の設定

この章では、Cisco NX-OS デバイスで周波数の同期を設定する方法について説明します。

この章は、次の項で構成されています。

- 周波数同期化について （57 ページ）
- 同期イーサネット（SyncE）のライセンス要件 （60 ページ）
- 周波数同期のガイドラインと制限事項 （60 ページ）
- 周波数の同期の設定 （61 ページ）

周波数同期化について

次世代ネットワークは、ネットワーク全体に高精度の周波数を配信する機能を提供する必要があります。これは、周波数同期化と呼ばれます。高精度周波数は、回線エミュレーションやセル タワー周波数参照などのアプリケーションに必要です。TDM の ITU 仕様への準拠を実現するには、差分方式の回線エミュレーションが使用される必要があります。これには、エミュレートされた回線の両端で、既知で共通の精密周波数基準が必要です。

たとえば、ネットワーク内の2つのノード間のパケット遅延を正確に計算するために、異なるネットワーク デバイス間で時刻を正確に同期することが望ましい場合もあります。

次第に、SDH および SONET 機器はイーサネット機器と置き換えられつつあります。これは、周波数の同期機能がイーサネット ポートを介して必要になってきたためです。同期イーサネット（SyncE）は、既知で共通の精密周波数基準の PHY レベルの周波数の配布を提供します。

SyncE リンクを維持するには、一連の処理メッセージが必要です。これらのメッセージは、ノードが常に最も信頼できるソースからタイミングを取得していることを確認し、SyncE リンクのクロック制御に使用されているタイミング ソースの品質に関する情報を転送します。イーサネットを介した同期ステータス メッセージ（SSM）のトランスポート チャネルを提供する単純なプロトコルは、ITU 標準 G.8264 およびその関連する推奨事項に記載されています。

各タイミング ソースには、関連付けられている品質レベル（QL）があり、クロックの精度が提供されます。この QL 情報は、Ethernet Synchronization Messaging Channel（ESMC）上の SSM を介してネットワーク全体に送信されます。これにより、デバイスは同期のための利用可能で最適なソースを認識できます。推奨ネットワーク同期の流れを定義して、タイミング ループを防止するために、各ルータの特定のタイミング ソースにプライオリティ値を割り当てることが

できます。QL 情報およびユーザ割り当てのプライオリティ レベルを組み合わせることにより、ITU 標準 G.781 に従って SyncE のクロック制御に使用するタイミング ソースを各ルータが選択できるようになります。

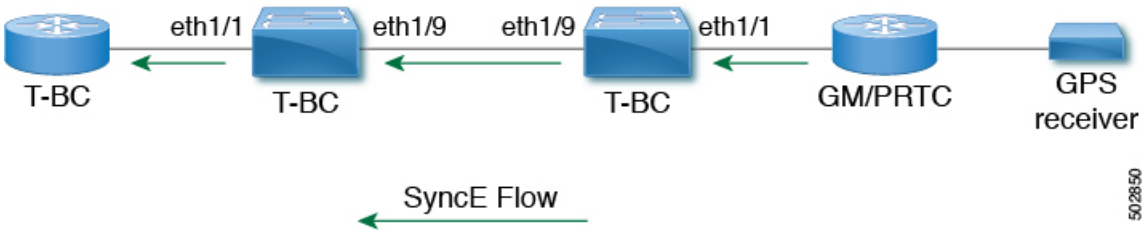
SyncE は時刻情報を伝送しません。時刻同期は、PTP などのパケットベースのテクノロジーを使用して実現されます。GNSS/GPS などのクロック ソースを使用して、正確な時刻と周波数をネットワークに注入できます。ネットワーク内の各スイッチは、時刻のソースと頻度のソースを選択し（または、可能かつ望ましい場合は両方に同じ送信元を選択し）、パケットベースのプロトコルを使用して時刻情報をピアに渡すことができます。時刻情報には QL に相当するものがないため、設定を使用して時刻の異なるソースを選択できます。

外部 PRC ソースを使用した Hybrid SyncE-PTP

Cisco NX-OS リリース 9.3(5) 以降では、ハイブリッド SyncE-PTP トポロジがサポートされ、回線エミュレーションとセル タワー周波数参照に必要なエンドツーエンド ネットワークの高精度周波数を実現します。

次の図に、外部タイミング ソースを、テレコム境界クロック（T-BC）のタイミング ソースを提供するグランドマスター/プライマリ基準時間クロック（GM / PRTC）として示します。

図 *3 :* 外部 *PRC* ソースを使用した *Hybrid SyncE-PTP*



タイミング ソース

以下に説明するように、システム/ネットワークにタイミング クロック信号を入力するさまざまなタイミング ソースと、システムからタイミング クロック信号を出力するタイミング ソースがあります。

タイミング入力

入力クロック信号は、プラットフォーム ハードウェアから、GPS / GNSS などのタイミング ソースからの入力、内部発振器からの入力、SyncE 対応インターフェイスの回線からの回復、または Precision Time Protocol（PTP）などのタイミング オーバー パケットから受信できます。

プラットフォームに依存しない（PI）ソフトウェアは、それぞれに関連付けられた品質レベル（QL）と優先度レベルを含む、これらすべての入力のデータベースを保持します。プライオリティ レベルは設定によって制御され、QL 値はさまざまな方法で取得できます。

- SyncE 対応インターフェイスは、イーサネット低速プロトコル（ESMC）を介して SSM を受信します。

- GPS および GNSS では、プラットフォーム依存（PD）ソフトウェアによって維持される QL が修正され、PI 機能に通知されます。
- PTP は、プラットフォーム API を介して周波数同期 PI ソフトウェアに QL を伝達します。
- デフォルトの QL 値は、タイミング コネクタおよび内部発振器の PD レイヤで定義できます。
- タイミング ソースの QL を定義する設定を行うことができます。

可能な入力ソース：

- 内部発振器
- 回復済み SyncE クロック
- 外部クロック 1588/PTP
- 外部クロック（GPS）
- 内部クロック（GNSS）

タイミング出力

プラットフォーム ハードウェアには、SyncE からのタイミング クロック出力や GPS の有効なインターフェイスなど、クロック信号用の出力が多数あります（現在はサポートされていません）。

ソフトウェアは、これらの出力を駆動するために使用されるクロック信号に関連付けられた QL 情報を含む、これらのすべての出力をデータベースに保持します。QL 情報には、QL 値、ステップ削除カウンタ、発信元クロック ID、および発信元クロックから現在のクロックまでのパスに関する情報を含む一連のフラグが含まれます。QL 値は、入力で説明したのと同じ方法で送信されます（つまり、SyncE インターフェイスは ESMC SSM を送信します）。

可能な出力ソース：

- SyncE
- 1588/PTP：パケット出力は、PTP ソフトウェアで個別に処理されます。

タイミング ソース選択ポイント

システム全体でタイミングクロックを同期するさまざまな段階で、プラットフォームは、使用可能なタイミングクロックのいずれをさらに処理するかを選択する可能性があります。これらの選択ポイントは、システムを通過するタイミングクロック信号のフローを定義し、最終的には、タイミング出力に使用する入力タイミング ソースを全体的に決定します。

各プラットフォームでのこれらの選択ポイントの設定方法はハードウェアに依存しますが、プラットフォーム独立（PI）レイヤは、任意のプラットフォーム選択ポイントハードウェアを柔軟に表すことができる汎用選択ポイント抽象化を定義し、各プラットフォームがどの選択ポイントを持つか、また接続方法を定義できます。PI コードは、これらの選択ポイントを制御し、

タイミング ソースに関する必要な情報を追跡および配信し、プラットフォーム依存（PD）レイヤと対話して、各段階でのPD選択の結果を検出します。

PI タイミング ソース選択ポイント：

- 選択可能なタイミング入力：プラットフォーム選択ポイントのハードウェアで選択可能な多数のタイミング クロック入力を使用できます。可用性および関連する QL 情報と優先順位は PI ソフトウェアによって追跡されます。PI ソフトウェアは、使用可能な入力を、関連する品質レベルと優先順位とともに全体的な順序で PD レイヤに通知します。
- プラットフォーム固有の選択：プラットフォーム レイヤは、PI から取得した情報、およびその他のプラットフォーム レイヤの決定（クロック信号のハードウェアレベル認定など）に基づいて、使用する入力を決定します。実際の決定は、PD ソフトウェアで行う（およびハードウェアにプログラムする）ことも、ハードウェア自体で決定して PD ソフトウェアに戻すこともできます。
- 選択されたタイミング ソース出力：プラットフォームは、選択されたクロック信号を選択ポイントからの出力として渡します。PD レイヤは、使用可能な入力のステータスと、選択された入力を PI ソフトウェアに通知します。

プラットフォーム レイヤは、選択ポイントが何であるかを定義し、それらが潜在的な入力、相互、および潜在的な出力に接続される方法を定義します。PD で定義された選択ポイントのそれぞれで、プラットフォームは PI ソフトウェアとやり取りする方法を選択して、その特定のハードウェアを PI ソフトウェアに表すことができます。ハードウェアは、各選択ポイントでクロッキング認定を実行する必要はありません。各選択ポイントは、ハードウェアが複数の入力を選択する場所を表し、1 つまたは複数の入力からのクロックを転送します。

スイッチ スーパーバイザ上の SyncE の選択ポイント タイプは 1 つだけサポートされます。これは T0 および 1588 選択ポイントと呼ばれます。T0 選択ポイントは、SyncE DPLL のソースとその選択を表します。1588 の選択ポイントは、1588 の Assist DPLL のソースとその選択を表します。

同期イーサネット（SyncE）のライセンス要件

製品	ライセンス要件
Cisco NX-OS	SyncE にはアドオン ライセンスが必要です。NX-OS ライセンス方式の詳細について Licensing Guide』を参照してください。

周波数同期のガイドラインと制限事項

周波数同期には、次のガイドラインと制限事項があります。

- Cisco NX-OS リリースを通じて周波数同期（SyncE）機能をサポートする Cisco Nexus スイッチのリストについては、『Nexus Switch Platform Support Matrix』を参照してください。

- SyncE は物理インターフェイスだけでサポートされます。
- 任意の時点で、SyncE 選択入力について最大 4 つのイーサネット インターフェイスをモニタできます。
- PHYの各クワッド ポート グループは、1 つの基準クロックを提供します。
- 各クワッド ポート グループから 1 つのイーサネット インターフェイスのみを SyncE 入力として設定できます（ポートグループごとに 1 つの基準クロック）。SyncE 出力に制限はありません。
- SyncEは、ポートチャネルのメンバーインターフェイスで明示的にイネーブルにする必要があります。ポートチャネルのメンバー インターフェイスが SyncE 送信元としてロックされている場合、SyncE が有効になっている他のメンバーインターフェイスで DNU を送信する機能は、グローバルコマンド **fsync transmit dnu lag-members** によって制御されます。
- BC モードの G.8275.1 ハイブリッド プロファイルのみがサポートされます。
- このリリースの認定光学部品のリストについては、『Cisco Optics Compatibility Matrix』を参照してください。

（注） GLC-TE が SFP として使用されている場合、SyncE は 1G ではサポートされません。

- Cisco NX-OS リリース 10.3(2)F 以降、GPS と GNSS は Cisco Nexus 93180YC-FX3S スイッチでサポートされます。

周波数の同期の設定

周波数の同期の有効化

周波数同期を有効にし、スイッチの品質レベルを設定し、ESMC 拡張 TLV のクロック ID を特定し、ソフトウェア アップグレードの ESMCピア タイムアウトを設定するには、次の手順を使用します。

手順の概要

1. **configure terminal**
2. [**no**] **feature frequency-synchronization**
3. [**no**] **fsync quality itu-t option** { **1** | **2 generation** { **1** | **2** }
4. **fsync clock-identity** *mac-address* | **no fsync clock-identity**
5. [**no**] **fsync esmc peer receive timeout** { **0** | *value*}
6. [**no**] **fsync transmit dnu lag-members**

7. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **feature frequency-synchronization**
	例：
	`switch(config)# feature frequency-synchronization`
	`switch(config)#`
		スイッチの周波数の同期を有効にします。

ステップ 3	[**no**] **fsync quality itu-t option** { **1** | **2 generation** { **1** | **2** } }
	例：
	`switch(config)# fsync quality itu-t option 1`
	`switch(config)#`
		スイッチの品質レベルを指定します。デフォルトは **option 1** です。
		• **option 1**：DNU、EEC1、PRC、PRTC、SEC、SSU-A、SSU-B、eEEC および ePRTC が含まれます。
		• **option 2 generation 1**：DUS、EEC2、PRS、PRTC、RES、SMC、ST2、ST3、ST4、STU、eEEC、ePRTC が含まれます。
		• **option 2 generation 2**：DUS、EEC2、PROV、PRS、PRTC、SMC、ST2、ST3、ST3E、ST4、STU、TNC、eEEC および ePRTC が含まれます。
		（注）
		ここで設定される品質オプションは、インターフェイス周波数の同期コンフィギュレーション モードの **quality receive** および **quality transmit** コマンドで指定された品質オプションと一致する必要があります。

ステップ 4	**fsync clock-identity** *mac-address* | **no fsync clock-identity**
	例：
	`switch(config)# fsync clock-identity AB:CD:EF:12:34:56`
	`switch(config)#`
		イーサネット同期メッセージチャネル（ESMC）拡張 TLV に使用するクロック ID を指定します。クロック ID が設定されていない場合、システムはデフォルトの VDC MAC アドレスを使用します。

	コマンドまたはアクション	目的

ステップ 5	[**no**] **fsync esmc peer receive timeout** { **0** | *value*}
	 例：
	 `switch(config)# fsync esmc peer receive timeout 120`
	 `switch(config)#`
		ISSU 中の ESMC ピア受信タイムアウトを指定します。
		 0 を指定すると、ESMC ピア受信タイムアウトが無効になります。
		 値は ESMC 受信タイムアウト（秒単位）です。120 ー 600 の値を入力します。デフォルトは 120 です。
		 このコマンドは、ESMC コントロールプレーン、つまり選択が、*value* の期間のソフトウェア アップグレード中に削除されないようにします。

ステップ 6	[**no**] **fsync transmit dnu lag-members**
	 例：
	 `switch(config)# fsync transmit dnu lag-members`
	 `switch(config)#`
		SyncE は、ポートチャネルのメンバー インターフェイスで明示的に有効にする必要があります。ポートチャネルのメンバーインターフェイスが SyncE 送信元としてロックされている場合、SyncE が有効になっている他のメンバー インターフェイスで DNU（Do Not Use）QL を送信する機能は、このコマンドによって制御されます。
		 有効で、スイッチのクロックを駆動しているインターフェイスがポートチャネルの一部である場合、SyncE がそのインターフェイスで有効になっていると、ポートチャネルのメンバーも DNU QL を送信します。
		 無効にすると、システムは、クロックを駆動するインターフェイスと同じポートチャネルにあるかどうかに関係なく、選択した送信元の QL をすべてのインターフェイスで駆動します。

ステップ 7	（任意） **copy running-config startup-config**
	 例：
	 `switch(config)# copy running-config startup-config`
	 `switch(config)#`
		実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

インターフェイスの周波数の同期の設定

特定のインターフェイスで周波数同期を設定するには、次の手順を実行します。

始める前に

この手順は、同じインターフェイスでの PTP テレコム プロファイルの設定とともに、「ハイブリッド PTP」プラットフォームに必要なインターフェイス設定を構成します。インターフェ

イス PTP テレコム プロファイル設定の詳細については、PTP テレコム プロファイル 8275-1 のインターフェイスの構成 （110 ページ） を参照してください。

デバイスで周波数同期がグローバルに有効になっていることを確認します（グローバル コンフィギュレーション コマンド **feature frequency-synchronization** による）。

手順の概要

1. **configure terminal**
2. [**no**] **interface ethernet** *slot* / *port*
3. [**no**] **frequency synchronization**
4. [**no**] **selection input**
5. [**no**] **ssm disable**
6. [**no**] **quality** { **receive** | **transmit** } { **exact** | **highest** | **lowest** } **itu-t option** *ql-option ql*
7. [**no**] **priority** *value*
8. [**no**] **wait-to-restore** *minutes*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	[**no**] **interface ethernet** *slot* / *port*
	例：
	`switch(config)# interface ethernet 1/5`
	`switch(config-if)#`
		周波数同期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	[**no**] **frequency synchronization**
	例：
	`switch(config-if)# frequency synchronization`
	`switch(config-if-freqsync)#`
		インターフェイスの周波数の同期をイネーブルにして、インターフェイス周波数の同期 コンフィギュレーション モードを開始します。システムは、クロッキング送信に使用する周波数信号を選択しますが、入力としてのインターフェイスの使用をイネーブルにはしません。
		（注）
		このコマンドの **no** 形式は、周波数同期コンフィギュレーション モードでコンフィギュレーションが存在しない場合にのみ機能します。

ステップ 4	[**no**] **selection input** 例：	選択アルゴリズムに渡すタイミング ソースとしてインターフェイスを指定します。

	コマンドまたはアクション	目的
	`switch(config-if-freqsync)# selection input` `switch(config-if-freqsync)#`	

ステップ 5	[**no**] **ssm disable**
	例：
	`switch(config-if-freqsync)# ssm disable`
	`switch(config-if-freqsync)#`
		ESMC パケットの送信をディセーブルにします。受信した ESMC パケットはすべて無視されます。

ステップ 6	[**no**] **quality** { **receive** | **transmit** } { **exact** | **highest** | **lowest** } **itu-t option** *ql-option ql*
	例：
	`switch(config-if-freqsync)# quality receive exact itu-t option 1 PRC`
	`switch(config-if-freqsync)#`
		選択アルゴリズムで使用する前に、SSM で受信または送信した品質レベル（QL）値を調整します。各タイミング ソースには、関連付けられている QL があり、これらはクロックの精度を提供します。この QL 情報は、Ethernet Synchronization Messaging Channel（ESMC）上の SSM を介してネットワーク全体に送信されます。これにより、デバイスは同期のための利用可能で最適なソースを認識できます。
		• **exact** *ql*：受信した値に関係なく、正確な QL を指定します。ただし、受信した値が DNU の場合を除きます。
		• **highest** *ql*：受信した QL の上限を指定します。受信した値がこの指定された QL よりも大きい場合、この QL が代わりに使用されます。
		• **lowest** *ql*：受信した QL の下限を指定します。受信した値がこの指定された QL よりも小さい場合、DNU が代わりに使用されます。
		（注）
		このコマンドで指定された品質オプションは、**quality itu-t option** コマンドでグローバルに設定された品質オプションとマッチしている必要があります。

ステップ 7	[**no**] **priority** *value*
	例：
	`switch(config-if-freqsync)# priority 100`
	`switch(config-if-freqsync)#`
		インターフェイスの周波数のソースのプライオリティを設定します。プライオリティは、クロック選択アルゴリズムで同じ QL がある 2 つのソース間から選択するために使用されます。値は、1（最高プライオリティ）から 254（最低プライオリティ）の範囲で設定します。デフォルト値は 100 です。
		（注）
		このコマンドは、**selection input** が設定されている場合にのみ有効です。

	コマンドまたはアクション	目的

ステップ 8	[**no**] **wait-to-restore** *minutes*
	例：
	`switch(config-if-freqsync)# wait-to-restore 0`
	`switch(config-if-freqsync)#`
		インターフェイスの周波数同期の復元待機時間を分単位で設定します。 *minutes* は、インターフェイスが初期化されてから同期に使用されるまでの時間です。有効値の範囲は、0 ～ 12 です。デフォルト値は 5 です。
		（注）
		このコマンドは、**selection input** が設定されている場合にのみ有効です。

周波数の同期の設定の確認

周波数の同期の設定タスクが完了したら、このリファレンスを使用して設定エラーがないことを確認して、設定を確認します。

show frequency synchronization configuration-errors

このコマンドの出力には、周波数同期設定のエラーが表示されます。

次の例は、グローバル **quality itu-t option** とインターフェイス **quality receive itu-t option** 間の不一致を示しています。

```
switch# show frequency synchronization configuration errors

Elysian2(config)# show frequency synchronization configuration errors
Ethernet1/9
    quality receive exact itu-t option 1 PRC
* The QL that is configured is from a different QL option set than is
configured globally.


!Command: show running-config fsync_mgr all
!Running configuration last done at: Mon Feb 10 06:06:15 2020
!Time: Mon Feb 10 06:09:18 2020

version 9.3(5) Bios:version 00.04
feature frequency-synchronization

fsync quality itu-t option 2 generation 1 << must be the same as interface
fsync clock-identity 0
fsync esmc peer receive timeout 120

interface Ethernet1/9
  frequency synchronization
    selection input
    ssm disable
    quality receive exact itu-t option 1 PRC << must be the same as global
    priority 100
    wait-to-restore 0

interface Ethernet1/13
  frequency synchronization
    selection input
    ssm disable
```

```
    quality receive exact itu-t option 1 PRC
    priority 110
    wait-to-restore 0
```

show running-config fsync_mgr

このコマンドの出力には、デバイスの現在の周波数同期設定が表示されます。

show running-config fsync_mgr コマンドの出力例を次に示します。

```
switch# show running-config fsync_mgr

!Command: show running-config fsync_mgr
!Running configuration last done at: Mon Jun 29 13:49:34 2020
!Time: Mon Jun 29 13:50:51 2020

version 9.3(5) Bios:version 01.01
feature frequency-synchronization


interface Ethernet1/9
  frequency synchronization
    selection input
    priority 99
    wait-to-restore 0

interface Ethernet1/13
  frequency synchronization
    selection input
    ssm disable
    quality receive exact itu-t option 1 PRC
    wait-to-restore 0
```

show frequency synchronization interface brief

このコマンドの出力には、設定済みの周波数同期があるすべてのインターフェイスが表示されます。入力として指定されたソースには、フラグ（Fl）列に「S」があります。入力として指定されていないソースには「S」が表示されません。

show frequency synchronization interface brief コマンドの出力例を次に示します。

```
switch# show frequency synchronization interface brief

Flags:  > - Up                D - Down              S - Assigned for selection
        d - SSM Disabled      x - Peer timed out    i - Init state
        e - SSM Enabled       s - Output squelched
Fl    Interface          QLrcv QLuse Pri QLsnd Output driven by
===== ================== ===== ===== === ===== ======================
>S    Eth1/9             PRC   PRC   100 PRC   Eth1/13
>Sds  Eth1/13            n/a   PRC   100 n/a   Eth1/13
```

show frequency synchronization interface ethernet

このコマンドの出力には、個々の（ユーザが選択した）インターフェイスと関連する周波数同期情報が表示されます。

show frequency synchronization interface ethernet *slot* / *port* コマンドの出力例を次に示します。

```
switch# show frequency synchronization interface ethernet 1/9
```

```
Interface State:UP
Assigned as input for Selection
  Wait-to-restore time 0 minute(s)
  SSM Enabled
    Peer Up for 00:07:01, last SSM received 0.307s ago
    Peer has come up 4 times and timed out 1 times
    ESMC SSMs        Total  Information     Event    DNU/DUS
      Sent:           1097         1088         9         83
      Received:        823          816         7        155
  Input:
    Up
    Last received QL: PRC
    Effective QL: PRC, Priority: 100
    Originator clock ID: ffffffffebfa543
    SyncE steps: 1, eSyncE steps: 1
    Not all steps run eSyncE; Chain of extended ESMC data is broken
    Supports frequency
  Output:
    Selected source: Eth1/13
    Selected source QL: PRC
    Effective QL: PRC
    Originator clock ID: ffffffffebfa863
    SyncE steps: 1, eSyncE steps: 1
    Not all steps run eSyncE; Chain of extended ESMC data is broken
  Next selection points:
```

show frequency synchronization selection（PTP Profile 8275-1 あり）

このコマンドの出力には、システム内のさまざまな選択ポイントの詳細ビューが表示されます。

（注） 次に、PTP プロファイル 8275-1 が設定されている場合の出力例を示します。

show frequency synchronization selection *slot* / *port* コマンドの出力例を次に示します。

```
switch# show frequency synchronization selection
==============
Selection point: System Clock (T0) Selector (3 inputs, 1 selected)
  Last programmed 18.898s ago, and selection made 8.621s ago
  Next selection points
    Node scoped   :
  Uses frequency selection
  Used for local line interface output
  S  Input                     Last Selection Point         QL  Pri  Status
  == ========================  ========================  =====  ===  ===========
  11 Ethernet1/9               n/a                         PRC   99  Locked
     Ethernet1/13              n/a                         PRC  100  Available
     Internal0[1]              n/a                         SEC  255  Available
==============
Selection point: IEEE 1588 Clock Selector (3 inputs, 1 selected)
  Last programmed 18.898s ago, and selection made 18.626s ago
  Next selection points
    Node scoped   :
  Uses frequency selection
  S  Input                     Last Selection Point         QL  Pri  Status
  == ========================  ========================  =====  ===  ===========
     Ethernet1/9               n/a                         PRC   99  Unmonitored
     Ethernet1/13              n/a                         PRC  100  Unmonitored
```

```
  21 Internal0[1]              n/a                          SEC  255  Freerun    <<
```

show frequency synchronization selection（PTP Profile 8275-1 なし）

このコマンドの出力には、システム内のさまざまな選択ポイントの詳細ビューが表示されます。

（注） 次に、PTP プロファイル 8275-1 が設定されていない場合の出力例を示します。

show frequency synchronization selection *slot* / *port* コマンドの出力例を次に示します。

```
switch# show frequency synchronization selection==============
Selection point: System Clock (T0) Selector (3 inputs, 1 selected)
  Last programmed 00:03:04 ago, and selection made 00:02:54 ago
  Next selection points
    Node scoped   :
  Uses frequency selection
  Used for local line interface output
  S  Input                     Last Selection Point         QL  Pri  Status
  == ========================  ========================  =====  ===  ===========
  11 Ethernet1/9               n/a                         PRC   99  Locked
     Ethernet1/13              n/a                         PRC  100  Available
     Internal0[1]              n/a                         SEC  255  Available
==============
Selection point: IEEE 1588 Clock Selector (3 inputs, 1 selected)
  Last programmed 00:03:04 ago, and selection made 3.296s ago
  Next selection points
    Node scoped   :
  Uses frequency selection
  S  Input                     Last Selection Point         QL  Pri  Status
  == ========================  ========================  =====  ===  ===========
     Ethernet1/9               n/a                         PRC   99  Unmonitored
     Ethernet1/13              n/a                         PRC  100  Unmonitored
  21 Internal0[1]              n/a                         SEC  255  Holdover   <<
```

show esmc counters all

このコマンドの出力には、送受信された ESMC SSM のカウンタが表示されます。

show esmc counters all コマンドの出力例を次に示します。

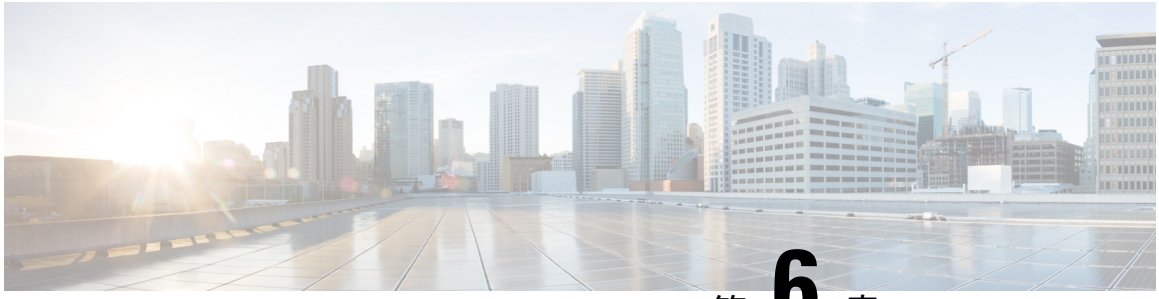
```
ESMC Packet Counters of Interface Ethernet1/1:
    ESMC SSMs        Total  Information     Event    DNU/DUS
      Sent:              0            0         0          0
      Received:          0            0         0          0

ESMC Packet Counters of Interface Ethernet1/5:
    ESMC SSMs        Total  Information     Event    DNU/DUS
      Sent:              0            0         0          0
      Received:          0            0         0          0

ESMC Packet Counters of Interface Ethernet1/9:
    ESMC SSMs        Total  Information     Event    DNU/DUS
      Sent:           7685         7683         2          0
      Received:       7688         7682         6         19

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

show esmc counters interface ethernet

このコマンドの出力には、特定のインターフェイスで送受信された ESMC SSM のカウンタが表示されます。

show esmc counters interface ethernet *slot* / *port* コマンドの出力例を次に示します。

```
ESMC Packet Counters of Interface Ethernet1/9:
    ESMC SSMs        Total  Information     Event    DNU/DUS
      Sent:           7955         7953         2          0
      Received:       7958         7952         6         19

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



第 6 章

PTP の設定

この章では、Cisco NX-OS デバイスで高精度時間プロトコル（PTP）を設定する方法について説明します。

この章は、次の項で構成されています。

- PTP について （71 ページ）
- PTP の注意事項および制約事項 （79 ページ）
- PTP のデフォルト設定 （86 ページ）
- PTP の設定 （87 ページ）
- PTP ユニキャスト ネゴシエーションの有効化 （124 ページ）
- 拡張マルチキャスト スケール （127 ページ）
- タイムスタンプ タギング （128 ページ）
- PTP 設定の確認 （132 ページ）
- PTP の設定例 （138 ページ）
- その他の参考資料 （140 ページ）

PTP について

PTP は、ネットワークに分散したノード間で時刻同期を行うプロトコルで、IEEE 1588 に定義されています。PTP を使用すると、イーサネット ネットワークを介して 1 マイクロ秒未満の精度で、分散したクロックを同期できます。さらに、PTP のハードウェア タイムスタンプ機能は、ERSPAN タイプ III ヘッダのタイムスタンプ情報を提供します。この情報は、エッジ スイッチ、集約スイッチ、およびコア スイッチ間のパケット遅延の計算に使用できます。

PTP システムは、PTP および非 PTP デバイスの組み合わせで構成できます。PTP デバイスには、オーディナリ クロック、境界クロック、およびトランスペアレント クロックが含まれます。非PTPデバイスには、通常のネットワーク スイッチやルータなどのインフラストラクチャ デバイスが含まれます。

PTPは、システムのリアルタイムPTPクロックが相互に同期する方法を指定する分散プロトコルです。これらのクロックは、グランドマスタークロック（階層の最上部にあるクロック）を持つマスター/スレーブ同期階層に編成され、システム全体の時間基準を決定します。同期は、タイミング情報を使用して階層のマスターの時刻にクロックを調整するメンバーと、PTP タイ

ミングメッセージを交換することによって実現されます。PTPは、PTPドメインと呼ばれる論理範囲内で動作します。

PTP は次の機能をサポートしています。

- マルチキャストおよびユニキャストPTP転送：マルチキャスト転送モードでは、PTPはデバイス間の通信に IEEE 1588 標準に従ってマルチキャスト宛先 IP アドレス 224.0.1.129 を使用します。送信元 IP アドレスの場合、PTP ドメインでユーザが設定可能なグローバル IPアドレスを使用します。ユニキャストトランスポートモードでは、PTPはインターフェイスで設定可能な設定可能なユニキャスト送信元および宛先 IPア ドレスを使用します。ユニキャスト モードとマルチキャスト モードの両方で、PTP は UDP ポートを使用します。イベント メッセージには319、デバイス間の一般的なメッセージ通信には 320 を使用します。
- PTP マルチキャスト設定は、L2 または L3 の物理インターフェイスでのみサポートされます。L3物理インターフェイスでのみサポートされるユニキャストPTP設定。PTPは、ポートチャネル、SVI、トンネルなどの仮想インターフェイスではサポートされません。
- IP over UDP over PTP カプセル化：PTP は、IP 上のトランスポート プロトコルとして UDP を使用します。ユニキャスト モードとマルチキャスト モードの両方で、PTP はイベント メッセージに UDP ポート 319 を使用し、デバイス間の一般的なメッセージ通信に 320 を使用します。L2 カプセル化モードは、 ではサポートされていません。
- PTP プロファイル：PTP はデフォルト（1588）、AES67、および SMPTE 2059-2 プロファイルをサポートします。すべての同期要求間隔と遅延要求間隔が異なります。デフォルト プロファイルの詳細については、IEEE 1588 を参照してください。AES67 および SMPTE 2059-2 の詳細については、それぞれの仕様を参照してください。
- パス遅延測定：マスターとスレーブのデバイス間の遅延を測定する遅延要求および応答メカニズムをサポートします。ピア遅延要求および応答メカニズムは、 ではサポートされていません。
- メッセージ間隔：デバイス間でアナウンス、同期、および遅延要求メッセージを送信する必要がある間隔を設定できます。
- ベスト マスター クロック（BMC）の選択：BMC アルゴリズムは、1588 仕様に従って受信したアナウンス メッセージに基づいて、PTP 対応インターフェイスのマスター、スレーブ、およびパッシブ状態を選択するために使用されます。

PTP オフロード

この機能により、ライン カードに PTP 機能が分散され、システムでサポートされる PTP セッション数のスケーリングが可能になります。この機能は、9700-EX、9700-FX、9636C-R、9636Q-R、9624D-R2、および 9636C-RX ライン カードを搭載した Cisco Nexus 9500 プラットフォーム スイッチで使用できます。

PTP デバイス タイプ

PTP デバイス タイプは設定可能で、クロック タイプの設定に使用できます。

クロック

次のクロックは、一般的な PTP デバイスです。

オーディナリ クロック

エンド ホストと同様に、単一の物理ポートに基づいてネットワークと通信します。オーディナリ クロックはグランドマスター クロックとして動作できます。

境界クロック

通常、複数の物理ポートがあり、各ポートはオーディナリ クロックのポートのように動作します。ただし、各ポートはローカル クロックを共有し、クロックのデータ セットはすべてのポートに共通です。各ポートは、境界クロックのその他すべてのポートから使用可能な最善のクロックに基づいて、個々の状態を、マスター（それに接続されている他のポートを同期する）またはスレーブ（ダウンストリーム ポートに同期する）に決定します。同期とマスター/スレーブ階層の確立に関するメッセージは、境界クロックのプロトコル エンジンで終了し、転送されません。

トランスペアレント クロック

通常のスイッチやルータなどのすべての PTP メッセージを転送しますが、スイッチでのパケットの滞留時間（パケットがトランスペアレント クロックを通過するために要した時間）と、場合によってはパケットの入力ポートのリンク遅延を測定します。トランスペアレント クロックはグランドマスター クロックに同期する必要がないため、ポートの状態はありません。

次の 2 種類のトランスペアレント クロックがあります。

エンドツーエンド トランスペアレント クロック

PTP メッセージの滞留時間を測定し、PTP メッセージまたは関連付けられたフォローアップ メッセージの修正フィールドの時間を収集します。

ピアツーピア トランスペアレント クロック

PTP メッセージの滞留時間を測定し、各ポートと、リンクを共有する他のノードの同じように装備されたポートとの間のリンク遅延を計算します。パケットの場合、この着信リンクの遅延は、PTP メッセージまたは関連付けられたフォローアップ メッセージの修正フィールドの滞留時間に追加されます。

（注） PTP は境界クロック モードのみで動作します。シスコでは、スイッチに接続された、同期を必要とするクロックが含まれるサーバを使用して、グランドマスター クロック（10 MHz）アップストリームを配置することを推奨します。

エンドツーエンド トランスペアレント クロック モードとピアツーピア トランスペアレント クロック モードはサポートされません。

グランドマスタークロック

単一の PTP ドメインでは、グランドマスター（GM）ノードは PTP ネットワーク全体のプライマリ クロック ソースとして機能します。GM ノードのプライマリ ソースは、内部 GNSS システムまたは外部 GNSS システムのいずれかから取得されます。GM ノードは他の PTP ノードから時間または周波数を同期できません。つまり、GM ノードはスレーブ ポートを持つことができず、すべてのポートはマスター ロールとしてのみ機能します。

クロック モード

IEEE 1588 規格は、PTP をサポートするデバイスが 1 ステップと 2 ステップで動作するための 2 つのクロックモードを指定しています。

1 ステップ モード：

1 ステップ モードでは、クロック同期メッセージに、マスター ポートがメッセージを送信した時刻が含まれます。ASIC は、同期メッセージがポートを出るときにタイムスタンプを追加します。1 ステップ モードで動作するマスター ポートは、Cisco Nexus 9508-FM-R および 9504-FM-R ファブリック モジュールおよび Cisco Nexus 9636C-R、9636Q-R、9624D-R2、および 9636C-RX ライン カードで使用できます。

スレーブ ポートは、同期メッセージの一部として送信されるタイムスタンプを使用します。

2 ステップ モード：

2 ステップ モードでは、同期メッセージがポートを出た時刻は後続のフォローアップ メッセージで送信されます。これは、デフォルトのモードです。

PTP プロセス

PTP プロセスは、マスター/スレーブ階層の確立とクロックの同期の 2 つのフェーズで構成されます。

PTP ドメイン内では、オーディナリ クロックまたは境界クロックの各ポートが、次のプロセスに従ってステートを決定します。

- 受信したすべての（マスター ステートのポートによって発行された）アナウンス メッセージの内容を検査します
- 外部マスターのデータ セット（アナウンス メッセージ内）とローカル クロックで、優先順位、クロック クラス、精度などを比較します

- 自身のステートがマスターまたはスレーブのいずれであるかを決定します

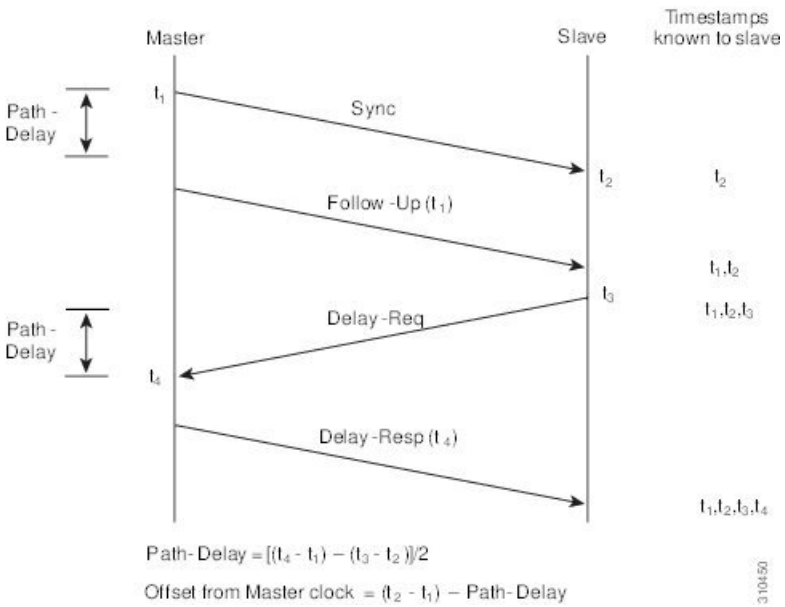
オーディナリ クロックと境界クロックは、**Sync**、**Delay_Req**、**Follow_Up**、**Delay_Resp** イベント メッセージを使用してタイミング情報を生成し、伝えます。

これらのメッセージは、次のシーケンスで送信されます。

1. マスターが、スレーブに **Sync**メッセージを送信し、それが送信された時刻（t1）を記録します。1 ステップ **Sync** メッセージの場合、メッセージはマスターから送り出された時刻を示します。2 ステップ メッセージの場合、この時刻は、後続の **Follow-Up** イベントメッセージで送信されます。
2. スレーブは、**Sync** メッセージを受信し、受信した時刻 （t2）を記録します。
3. マスターはスレーブに対し、タイムスタンプ t1 を、**Follow_Up** イベント メッセージに埋め込むことにより送信します。
4. スレーブはマスターに対し、**Delay_Req** メッセージを送信し、送信した時刻 t3 を記録します。
5. マスターは **Delay_Req** メッセージを受信し、受信した時刻、t4 を記録します。
6. マスターはスレーブに対し、タイムスタンプ t4 を、**Delay_Resp** メッセージに埋め込むことによって送信します。
7. このシーケンスの後、スレーブは 4 つすべてのタイムスタンプを所有します。これらのタイムスタンプを使用して、マスターに対するスレーブ クロックのオフセットと、2 つのクロック間のメッセージの平均伝達時間を計算できます。

   次の図は、タイミング情報を生成して通信する PTP プロセスのイベント メッセージを示しています。

図 *4 : PTP* プロセス



PTP の ITU-T 電気通信プロファイル

Cisco NX-OS ソフトウェアは、ITU-T 勧告の定義に従って、PTP の ITU-T 電気通信プロファイルをサポートしています。プロファイルは、特定のアプリケーションにのみ適用可能な PTP 設定オプションで構成されます。

IEEE 1588-2008 標準に基づいて PTP を異なるシナリオに組み込むために、個別のプロファイルを定義することができます。電気通信プロファイルは、IEEE 1588-2008 標準で定義されているデフォルトの動作とはいくつかの点で異なります。主要な相違点については、以降の項で説明します。

次の項では、PTP でサポートされている ITU-T 電気通信プロファイルについて説明します。

Telecom Profile G.8275.1

シスコの Telecom Profile G.8275.1 機能は、ITU-T *G.8275.1* をサポートします。これは、ネットワーク標準からの完全なタイミングサポートによる、フェーズ/時間同期用の高精度時間プロトコル Telecom プロファイルです。G.8275.1 プロファイルは、PTP プロトコルに参加しているすべてのネットワーク デバイスとの電気通信ネットワークにおける時刻およびフェーズの同期要件を満たしています。SyncE を使用した G.8275.1 プロファイルは、時刻およびフェーズの同期の周波数安定性を向上させます。

G.8275.1 プロファイルの特徴は次のとおりです。

- 同期モデル：G.8275.1 プロファイルは、ホップバイホップ同期モデルを採用しています。マスターからスレーブへのパス内の各ネットワーク デバイスは、ローカル クロックをアップストリーム デバイスに同期させ、ダウンストリーム デバイスに同期を提供します。
- クロック選択：G.8275.1 プロファイルでは、同期用のクロックを選択する代替 BMCA も定義され、ネットワーク内のすべてのデバイスのローカル ポートのポート状態がプロファイル用に定義されています。BMCA の一部として定義されているパラメータは次のとおりです。
  - クロック クラス
  - クロック精度
  - オフセット調整されたログのバリアンス
  - 優先順位 2
  - ローカル優先度
  - クロック ID
  - 削除されるステップ
  - ポート ID
- ポート状態の決定：ポート状態は、代替 BMCA に基づいて選択されます。

- 代替 BMCA：推奨で定義されている代替 BMCA データセット比較アルゴリズムに従います。ITU-T G.8275.1/Y.1369.1：ノードの GM を選択します。
- パケット レート：アナウンス パケットの公称パケット レートは、Sync/Follow-Up および Delay-Request/Delay-Response パケットの場合、それぞれ毎秒 8 パケットおよび毎秒 16 パケットです。
- 転送メカニズム：G.8275.1 プロファイルは、イーサネット PTP 転送メカニズムのみをサポートします。
- モード：G.8275.1 プロファイルは、マルチキャスト モードでのみデータ パケットの転送をサポートします。転送は、転送可能または転送不可能なマルチキャスト MAC アドレスに基づいて行われます。
- クロック タイプ：G.8275.1 プロファイルは、次のクロック タイプをサポートしています。
  - Telecom Grandmaster（T-GM）：他のネットワーク デバイスにタイミングを提供し、ローカル クロックを他のネットワーク デバイスと同期させません。
  - Telecom Time Slave Clock（T-TSC）：スレーブ クロックは、ローカル クロックを別の PTP クロックに同期させますが、他のネットワーク デバイスには PTP 同期を提供しません。
  - Telecom Boundary Clock（T-BC）は、ローカル クロックを T-GM またはアップストリーム T-BC クロックに同期させ、タイミング情報をダウンストリーム T-BC または T-TSC クロックに提供します。

（注） Telecom Boundary Clock（T-BC）は、Cisco NX-OS Release 9.3（5）でサポートされている唯一のクロック タイプです。

- ドメイン番号：G.8275.1 プロファイル ネットワークで使用できるドメイン番号は 24 ～ 43 です。デフォルトのドメイン番号は 24 です。

Telecom Profile G.8275.2

シスコの Telecom Profile G.8275.2 機能は、ITU-T *G.8275.2* をサポートします。これは、ネットワーク標準からの完全なタイミング サポートによる、フェーズ/時間同期用の高精度時間プロトコル Telecom プロファイルです。G.8275.2 は、フェーズまたは時刻の同期が必要な電気通信ネットワークで使用するための PTP プロファイルです。ネットワーク内の各デバイスが PTP プロトコルに参加する必要はないという点で、G.8275.1 とは異なります。また、G.8275.2 は、ユニキャスト モードで IPv4 および IPv6 経由の PTP を使用します。

G.8275.2 プロファイルは、ネットワークからの部分的なタイミング サポートに基づいています。したがって、G.8275.2 を使用するノードは直接接続する必要はありません。G.8275.2 プロファイルは、時間とフェーズの正確な同期を必要とするモバイル セルラー システムで使用されます。たとえば、第 4 世代（4G）の移動体通信技術で使用されます。

LTE-TDD、LTE-A CoMP、LTE MBSFN、ロケーション ベースのサービスなどの今後のテクノロジーでは、eNodeB（基地局デバイス）はフェーズと時間で正確に同期する必要があります。各ノードに GNSS システムを配置すると、コストがかかるだけでなく、脆弱性も生じます。G.8275.2 プロファイルは、これらの新しいテクノロジーの同期要件を満たしています。

PTPポート

ポートは、そのロールをダイナミックに変更するように構成できます。ポートにロールが割り当てられていない場合は、BMCA に基づいてプライマリ、パッシブ、または下位のロールをダイナミックに担当できます。

G.8275.2 では、PTP ポートは特定の物理インターフェイスに関連付けられておらず、ループバック（仮想）インターフェイスに関連付けられています。PTP ポートからのトラフィックは、ルーティングの決定に基づいて任意の物理インターフェイスを介してルーティングされます。ダイナミック ポートの場合、構成できるクロック ソースは 1 つだけです。

代替 BPCA

G.8275.2 の BPCA（ベスト プライマリ クロック アルゴリズム。ベスト マスター クロック アルゴリズム（BMCA [RFC 7273]）とも呼ばれる）の導入は、デフォルトの PTP プロファイルの導入とは異なります。G.8275.2 の導入は、代替ベスト プライマリ クロック アルゴリズム（ABPCA）を指定します。ABPCA は、各デバイスが同期するクロックを選択し、ローカル ポートのポート状態を決定するために使用されます。

BPCA の G.8275.2 実装には、次の考慮事項が適用されます。

- PrimaryOnly：ポートごとの属性。PrimaryOnly はポートの状態を定義します。この属性が true の場合、ポートは従属状態になりません。
- プライオリティ 1：プライオリティ 1 は、このプロファイルでは常に静的であり、128 に設定されます。プライオリティ 1 は BPCA では使用されません。
- プライオリティ 2：プライオリティ 2 は構成可能な値で、その範囲は 0 ～ 255 です。
- ローカル プライオリティ：ローカル プライオリティは、指定されたクロックにプライオリティを設定するために、クロック ポートでローカルに構成されます。デフォルト値は 18、有効値の範囲は 1 ～ 255 です。

G.8275.2 プロファイルの使用に関する制約事項

- G.8275.2 プロファイルでは、PTP は none モード（デフォルト）でのみサポートされます。
- G.8275.2 PTP クロックには、（複数の PTP ポートを介して）冗長クロック ソースを構成できます。ただし、G.8275.2 PTP クロックは、BMCA によって選択された 1 つのクロック ソースにのみ同期します。
- G.8275.2 には、クロックのパフォーマンス分析およびネットワーク制限に関する推奨事項はありません。

PTP のハイ アベイラビリティ

PTP のステートフル リスタートはサポートされません。リブート後またはスーパーバイザ スイッチオーバー後に、実行コンフィギュレーションが適用されます。ハイ アベイラビリティの詳細については、『Cisco Nexus 9000 シリーズ NX-OS ハイ アベイラビリティおよび冗長性ガイド』を参照してください。

PTP の注意事項および制約事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

PTP 用 Cisco Nexus 9000 シリーズスイッチの注意事項と制約事項は次のとおりです。

- PTP が正常に機能するには、最新の SUP およびラインカードの FPGA バージョンを使用する必要があります。FPGAのアップグレードについては、リリースノートのランディングページにアクセスし、「FPGA / EPLDアップグレードリリースノート（NX-OSモードスイッチ）」セクションに移動して、ご使用のソフトウェアバージョンのFPGA / EPLDアップグレードリリースノートを参照してください。https://www.cisco.com/c/en/us/support/switches/nexus-9000-series-switches/products-release-notes-list.html「インストール ガイドライン」のトピックを参照してください。
- Cisco NX-OS リリース 9.3(3) 以降では、PTP は Cisco Nexus 93360YC-FX2 および 93216TC-FX2 スイッチでサポートされています。
- Cisco NX-OS リリース 9.3(5) から、N9K-C93180YC-FX3S プラットフォーム スイッチでは、PTP G.8275.1 Telecom プロファイルがサポートされています。
- Cisco NX-OS リリース 9.3(5) から、 N9K-C93180YC-FX3P プラットフォーム スイッチでは、PTP がサポートされています。ただし、syncE はサポートされていません。
- Cisco NX-OS リリース 9.3(7) から、N9K-C93180YC-FX3S プラットフォーム スイッチでは、PTP G.8275.1 Telecom プロファイル がサポートされています。
- Cisco NX-OS リリース 10.2(1)F 以降では、PTP プロファイル 8275-1 で ing-sup（入力スーパーバイザ TCAM リージョンのサイズ）を 768 に明示的にカービングする必要はありません。
- PTPv1 転送と機能 VMCT1を同時に有効にすることはサポートされていません。
- PTP テレコム プロファイルには次の注意事項と制約事項があります。
  - PTP テレコム プロファイルは、Cisco Nexus 93180YC-FX3S スイッチ と N9K-C93180YC-FX3 スイッチでのみサポートされます。

- デフォルトでは、毎秒 1 パルス（1PPS）の出力が有効になっています。UTC/SMB ポートは出力モード です。1PPS 出力はサポートされていないことに注意してください。
- 25G 以上のポート速度では、PTP クラス B のみがサポートされます。
- Telecom Boundary Clock（T-BC）のみがサポートされます。
- シスコの Telecom Profile G.8273.2 機能は、ITU-T *G.8273.2*：通信境界クロックおよび通信時間スレーブ クロックのタイミング特性標準に準拠しています。ただし、1 PPS 出力が PTP と整合していないことを除きます。

（注） 時刻および PTP GM は、Cisco NX-OSリリース 9.3(5) ではサポートされていません。

- Cisco NX-OSリリース 9.3(5) 以降、PTP は Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされます。
- Cisco NX-OSリリース9.3(5)以降、PTPコマンドのCLI動作は次のように変更されました。
  - ほとんどの PTP コマンドは、同じコマンドを再度適用してもエラーを返しません。
  - ほとんどのPTPコマンドは、「no」コマンドとして入力されたパラメータを検証しません。たとえば、現在設定されているコマンドが「ptp sync interval -3」の場合、「no ptp sync interval -1」はその否定として受け入れられます。
- PTP はネットワークごとに 1 つのドメインに制限されます。
- ユーザ データグラム プロトコル（UDP）上の PTP 転送がサポートされます。 PTP over Ethernet は、Nexus 9300-FX3 プラットフォーム スイッチでのみサポートされています。
- PTPはマルチキャスト通信をサポートします。PTPはユニキャスト通信もサポートしていますが、ユニキャスト モードはオプションです。
- PTP は境界クロック モードをサポートします。エンドツーエンド トランスペアレント クロック モードとピアツーピア トランスペアレント クロック モードはサポートされません。
- PTP はポートチャネル メンバー ポートで有効にできます。
- スレーブ ポートから受信したすべての管理メッセージは、すべてのPTP対応ポートに転送されます。スレーブ ポートから受信した管理メッセージは処理されません。
- PTPは、Cisco Nexus 92348GC-Xプラットフォーム スイッチではサポートされていません。
- タイムスタンプ タギング（TTAG）は、次のプラットフォーム スイッチでサポートされています。
  - Cisco Nexus 9200 プラットフォーム スイッチ：Cisco NX-OS リリース 7.0(3)I6(1) 以降

- Cisco Nexus 9364C：Cisco NX-OS リリース 7.0(3)I7(2) 以降
- Cisco Nexus 9332C：Cisco NX-OS リリース 9.2(3) 以降
- Cisco Nexus 9300-EX プラットフォーム スイッチ：Cisco NX-OS リリース 7.0(3)I6(1) 以降
- Cisco Nexus 9300-FX プラットフォーム スイッチ：Cisco NX-OS リリース 7.0(3)I7(3) 以降
- Cisco Nexus 9300-FX2 プラットフォーム スイッチ：Cisco NX-OS リリース 9.3(3) 以降
- Cisco Nexus 9300-FX3 および -GX プラットフォーム スイッチ：Cisco NX-OS リリース 9.3(5) 以降
- -EX/-FX ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチ

- RACL を使用して PTP 制御パケットを照合するには、L3 インターフェイスで PIM を有効にします。
- Cisco Nexus 9000 シリーズ スイッチに PTP を設定する場合は、clock protocol ptp vdc 1コマンドを使用して、PTP を使用するようにクロック プロトコルを設定します。
- PTP は、100G 9408PC ライン カードおよび 100G M4PC 汎用拡張モジュール（GEM）を除き、すべての Cisco Nexus 9000 シリーズおよび 3164Q ハードウェアでは利用できません。
- Cisco NX-OS リリース 9.3(3) 以降、Cisco Nexus 9504-FM-R プラットフォーム スイッチでは PTP が利用できます。
- PTP correction-range、PTP correction-range logging、および PTP mean-path-delay コマンドは、Cisco Nexus 9508-R ライン カードでサポートされます。
- Cisco Nexus 31108PC-V および 31108TC-V スイッチの場合、100 Gの速度で動作するポートでは PTP はサポートされません。
- Cisco Nexus 9000 シリーズ スイッチでは、マスター PTP ポートで操作の混合非ネゴシエート モードがサポートされます。つまり、スレーブ クライアントがユニキャスト遅延要求 PTP パケットを送信すると、Cisco Nexus 9000 はユニキャスト遅延応答パケットで応答することを意味します。また、スレーブ クライアントがマルチキャスト遅延要求PTP パケットを送信すると、Cisco Nexus 9000 はマルチキャスト遅延応答パケットで応答します。混合非ネゴシエートモードが機能するには、BC デバイスの ptp 送信元 *IP* アドレス設定で使用される送信元 IP アドレスが、BC デバイスの物理または論理インターフェイスでも設定されている必要があります。推奨されるベストプラクティスは、デバイスのループバック インターフェイスを使用することです。
- Cisco NX-OSリリース9.2(1) 以降では、Cisco Nexus 9732C-EX、9736C-EX、および 97160YC-EX ライン カードが PTP オフロードをサポートしています。
- Cisco NX-OSリリース 9.3(1) からリリース 7.0(3)I7 にダウングレードする際には、その前に、PTP オフロードを設定解除する必要があります。Cisco NX-OSリリース7.0(3)I7 の場

合、PTP オフロードは、9636PQ、9564PX、9464PX、および 9536PQ ライン カード上の Cisco Nexus 9000 プラットフォーム スイッチではサポートされません。

- Cisco Nexus 93108TC-EX および 93180YC-EX スイッチは、混合モードおよびユニキャスト モードでの PTP をサポートします。Cisco Nexus 9396 スイッチは PTP 混合モードをサポートします。
- 同期間隔 -3での PTP は、Cisco Nexus 9508-R ファミリ ライン カードでのみサポートされます。より高い同期間隔はサポートされません。
- PTP ユニキャストはデフォルトの VRF でのみサポートされます（PTP ユニキャストはオフロード モードではサポートされません）。
- PTP は、ステートフル高可用性ではサポートされません。
- PTP は、管理インターフェイスではサポートされません。
- PTPは、PTP メッセージを配信するための混合モードをサポートします。これは、接続されたクライアントから受信した遅延要求メッセージのタイプに基づいて Cisco Nexus デバイスが自動的に検出するものなので、設定は不要です。
- ワンステップ PTP は、Cisco Nexus 9000-R シリーズ プラットフォーム スイッチでのみサポートされます。
- PTP は、FEX インターフェイスではサポートされません。
- PTP 対応ポートは、ポート上で PTP をイネーブルにしない場合、PTP パケットを識別せず、これらのパケットにタイムスタンプを適用したり、パケットをリダイレクトしたりしません。
- 9636C-R、9636C-RX、または 9636Q-R ライン カードを搭載した Cisco Nexus 9504 および 9508プラットフォームスイッチでは、マスターポートはワンステップ モードで動作できます。
- PTP ワンステップ モードは、9636C-R、9636C-RX、9624D-R2、または 9636Q-R ライン カードを搭載した Cisco Nexus 9504 および 9508 プラットフォーム スイッチの PTP オフロードモードでのみサポートされます。Cisco NX-OS リリース 9.3(3) 以降では、ワンステップ モードが設定されると、PTP オフロードが自動的に有効になります。
- PTP が有効になっているトポロジで、GrandMaster デバイスにプロファイルが設定され、冗長 GrandMaster がネットワークに展開されている場合、GrandMaster のプロファイルを変更するには、最初にスイッチへの GrandMaster に設定されているポートをシャットダウンし、プロファイルを変更してから、ポートを再度有効にする必要があります。例えば、AES7 プロファイルから SMPTE プロファイルに、またはその逆の移動です。
- 各ポートは、サポートされている任意の PTP プロファイルを使用して個別に構成できます。異なる PTP プロファイルは、インターフェイス上で共存できます。デフォルトの 1588 と SMPTE-2059-2 または AES67 プロファイルの組み合わせがサポートされています。ただし、SMPTE-2059-2 と AES67 プロファイルの組み合わせは、同じインターフェイスではサポートされていません。

- PTP は N9K-C92348GC-X スイッチではサポートされていません。
- Cisco NX-OS リリース 10.1(2) 以降、PTP (IEEE 1588) は、C9504-FM-G および N9K-C9508-FM-G ファブリック モジュールと共に使用される N9K-C9700-GX ラインカード、および N9K-C9700-EX および N9K-C9700-FX ラインカードでサポートされます。
- Cisco NX-OSリリース10.1(2) 以降では、N9K-X9624D-R2 ライン カードで PTP がサポートされます。
- Cisco NX-OS リリース 10.2(1q)F 以降、PTP は N9K-C9332D-GX2B プラットフォーム スイッチでサポートされます。ただし、PTP は 1/33 および 1/34 ポートではサポートされません。
- Cisco NX-OS リリース 10.2(1) 以降、PTP IPv6 トランスポートは N9K-C93180YC-FX3S プラットフォームでサポートされます。
- QoS TCAM リージョンの入力 SUP [ingress-sup] は、動作するために PTP IPv6 トランスポートで 768 以上に設定する必要があります。
- Cisco NX-OS リリース 10.2(1)F 以降、ユニキャスト ネゴシエーションは、N9K-C93180YC-FX3S プラットフォームのデフォルト プロファイルで IPv4 および IPv6 アドレスに対してサポートされます。
- プラットフォーム スイッチはクラス B でのみサポートされ、クラス C のサポートを満たしません。
- 8275.2 には CLI プロファイル コマンドはありません。これは、APTS がサポートされている場合にのみ追加されます。このリリースの機能は、デフォルトモードでのみ動作します。
- Cisco NX-OS リリース 10.2(2)F 以降では、PTP IPv6 UDP トランスポート機能が Cisco Nexus 9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(2)F 以降では、PTP ユニキャスト ネゴシエーション機能もまた、Cisco Nexus 9300-EX、9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでもサポートされます。
- Cisco NX-OS リリース 10.2(2)F 以降では、1G ポートのジッター修正を使用した PTP 機能が Cisco N9K-C93108TC-FX3P プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.2(2)F 以降では、PTPv1 および v2 共存機能が Cisco Nexus 9300-GX、9300-GX2、および 9300-GX3 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、PTP は次のプラットフォームの次のポートではサポートされていません。
  - N9K-C9364D-GX2A：PTP は 1/65 および 1/66 ポートではサポートされていません
  - N9K-C9348D-GX2A：PTP は 1/49 および 1/50 ポートではサポートされていません
- Cisco NX-OS リリース 10.2(3)F 以降、スイッチあたり最大 2000 のセカンダリ デバイスの PTP サポート機能により、スイッチごとに 2000 のマルチキャスト セカンダリ デバイスに

より、ポートあたり最大 100 のマルチキャスト セカンダリ デバイスをサポートするオプションが提供され、システム全体で最大 100 のマルチキャスト セカンダリ デバイスがサポートされます。この機能は、すべての Cisco Nexus 9000-FX2 および 9000-FX3 プラットフォーム スイッチでサポートされています。

- Cisco NX-OS リリース以降では、IPv6 マルチキャスト構成を介した PTP の送受信範囲がサポートされています。
- Cisco NX-OS リリース 10.3(1)F 以降、PTP メディア プロファイルとワンステップ モードは Cisco Nexus 9808 プラットフォーム スイッチでサポートされます。このプラットフォーム スイッチに関するいくつかの注意事項と制約事項を次に示します。
  - IPv4 トランスポートのみがサポートされています
  - 1 ステップのマルチキャスト PTP のみがサポートされます
  - ユニキャストやユニキャスト ネゴシエーションなどの他の PTP 機能はサポートされていません。
- Cisco NX-OS リリース 10.3(2)F 以降、メディア プロファイル上の PTP は、Cisco Nexus 9408 プラットフォーム スイッチでサポートされています。このプラットフォーム スイッチに関するいくつかの注意事項と制約事項を次に示します。
  - 1 ステップのマルチキャスト PTP のみがサポートされます。
  - ユニキャストやユニキャスト ネゴシエーションなどの他の PTP 機能はサポートされていません。
- Cisco NX-OS リリース 10.3(2)F 以降、PTP GM 機能は Cisco Nexus N9K-C93180YC-FX3 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(3)F 以降、アンカー DR は Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。次の注意事項および制約事項が適用されます。
  - この機能により、ITU-T 高精度時間プロトコル（PTP）プロファイルを使用し、ネットワークからの部分的なタイミング サポート（ユニキャスト モード）によって、フェーズ/時間配信を行うことができます。
  - APTS がサポートされているため、G.8275.2 のコマンドが追加されました。
  - この機能は、複数のユニキャスト フォロワーもサポートします。ユニキャスト フォロワーのうち、1つはアクティブユニキャストフォロワーとして選択され、残りの設定済みユニキャスト フォロワーはパッシブとして機能します。
  - このリリース以降、ユニキャストポートを使用したダイナミックロールがサポートされています。
  - ダイナミック PTP ユニキャスト ポートの場合、ユニキャスト ネゴシエーション機能を有効にする必要があります。

- Cisco NX-OS リリース 10.4(1)F 以降、高精度時間プロトコル（PTP）は Cisco Nexus C9348GC-FX3PH スイッチではサポートされていません。
- Cisco NX-OS リリース 10.4(1)F 以降、高精度時間プロトコル（PTP）は Cisco Nexus C9348GC-FX3 スイッチでサポートされています。

（注） PTP インターフェイス全体で PTP 状態が安定するように、COPP 制限を増やします。構成の詳細については『*Cisco Nexus 9000* シリーズ *NX-OS* セキュリティ構成ガイド』の「*CoPP* の構成例」の項を参照してください。

- Cisco NX-OS リリース 10.4(1)F 以降、1G PTP での ASIC ベースのタイム スタンプ処理に対応したポートが、N9K-C931806C-FX3 と N9K-C93108TC-FX3P プラットフォーム スイッチでサポートされています。したがって、1G ポートと 10G ポートの両方が ASIC モードで動作できるようになりました。ただし、すべての 1G ポートが PHY モード（デフォルト）または ASIC モードである必要があります。この機能を有効にするために、グローバル コマンドが導入されました。
- Cisco NX-OS リリース 10.4(1)F 以降、G.8275.1 Telecom プロファイル機能の PTP クラス C サポートは、Cisco Nexus 9408 および N9K-C9332D-H2R プラットフォーム スイッチでサポートされます。このプラットフォーム スイッチに関するいくつかの注意事項と制約事項を次に示します。
  - プラットフォーム スイッチは、クラス C サポートを提供します。
  - ワンステップ モードのみがサポートされます
  - イーサネットトランスポートがサポートされています
  - システムでクラス C レベルの修正が行われるまでに最大 30 分かかる場合があります
  - G.8275.1 Telecom プロファイルは、次を除きサポートされます。
    - Telecom トランスペアレント クロック（T-TC）
    - Telecom グランドマスター（T-GM）
- Cisco NX-OS リリース 10.4(1)F 以降、PTP v1 パケット転送機能は、Cisco Nexus N9K-C9408 および N9K-C9332D-H2R プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、メディア プロファイル上の PTP は、N9K-C9332D-H2R プラットフォーム スイッチでサポートされています。このプラットフォーム スイッチに関するいくつかの注意事項と制約事項を次に示します。
  - 1 ステップのマルチキャスト PTP のみがサポートされています。
  - ユニキャストはサポートされていません。
  - PTP はインターフェイス Ethernet 1/33 および Ethernet 1/34 ではサポートされません。

- Cisco NX-OS リリース 10.4(1)F 以降、PTP は次のスイッチおよびライン カードでサポートされています。
  - Cisco Nexus 9804 スイッチには、次の注意事項と制限事項が適用されます。
    - IPv4 トランスポートのみがサポートされています。
    - メディア プロファイルのみがサポートされています。
    - サポートされているクロック分散モードのみが 1 ステップです。2 ステップ モードはサポートされません。
    - ユニキャストやユニキャスト ネゴシエーションなどの他の PTP 機能はサポートされていません。
  - Cisco Nexus X98900CD-A および X9836DM-A ラインカードと Cisco Nexus 9808 および 9804 スイッチ

PTP のデフォルト設定

次の表に、PTP パラメータのデフォルト設定を示します。

表 *3:* デフォルトの *PTP* パラメータ

パラメータ	デフォルト
PTP	ディセーブル
PTP バージョン	2
PTP ドメイン	0
クロックをアドバタイズする場合、PTP プライオリティ 1 値	255
クロックをアドバタイズする場合、PTP プライオリティ 2 値	255
PTP アナウンス間隔	1 ログ秒
PTP アナウンス タイムアウト	3 アナウンス間隔
PTP 遅延要求間隔	• 0 ログ秒 • Cisco Nexus 3232C、3264Q、および 9500 プラットフォーム スイッチの場合、-1 ログ秒

パラメータ	デフォルト
PTP 同期間隔	• – 2 ログ秒 • Cisco Nexus 3232C、3264Q、および 9500 プラットフォーム スイッチでは -3 ログ秒
PTP VLAN	gPTP はデフォルトの VLAN 1 だけをサポートし、他のユーザ設定 VLAN はサポートしません。

PTP の設定

PTP のグローバルな設定

デバイスで PTP をグローバルにイネーブルまたはディセーブルにできます。また、ネットワーク内のどのクロックがグランドマスターとして選択される優先順位が最も高いかを判別するために、さまざまな PTP クロック パラメータを構成できます。

（注） PTP が正常に機能するには、最新の SUP および LC FPGA バージョンを使用する必要があります。FPGAのアップグレードについては、リリースノートのランディングページにアクセスし、「FPGA / EPLDアップグレードリリースノート（NX-OSモードスイッチ）」セクションに移動して、ご使用のソフトウェアバージョンのFPGA / EPLDアップグレードリリースノートを参照してください。https://www.cisco.com/c/en/us/support/switches/nexus-9000-series-switches/products-release-notes-list.html「インストール ガイドライン」のトピックを参照してください。

（注） 1 ステップ モードまたは 2 ステップ モードに関係なく、PTP プロトコルによって更新されるローカル クロックのクロック プロトコル PTP vdc1 を常に設定する必要があります。設定は、**show running-config clock_manager** コマンドを使用して確認できます。

手順の概要

1. **configure terminal**
2. [**no**] **feature ptp**
3. （任意） [**no**] **ptp scale-on-1G**
4. [**no**] **ptp device-type** [**generalized-ptp** |**boundary-clock** |**ordinary-clock-grandmaster**]
5. [**no**] **ptp source** {<*ipv4 address*> | <*ipv6 address*>} [**vrf** <*vrf-name*>]
6. （任意） [**no**] **ptp domain** *number*
7. （任意） [**no**] **ptp offload**

8. （任意） [**no**] **ptp clock-operation one-step**
9. （任意） [**no**] **ptp priority1** *value*
10. （任意） [**no**] **ptp priority2** *value*
11. [**no**] **ptp management**
12. （任意） [**no**] **ptp delay tolerance** { **mean-path** | **reverse-path** } *variation*
13. （任意） **ptp forward-version1**
14. （任意） **ptp unicast-negotiation**
15. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **feature ptp** 例： `switch(config)# feature ptp`	デバイス上で PTP をイネーブルまたはディセーブルにします。 （注） スイッチの PTP をイネーブルにしても、各インターフェイスの PTP はイネーブルになりません。 dot1x（**feature dot1x**）または NV オーバーレイ（**feature nv overlay**）のいずれかの機能のみが設定されていることを確認します。これらの機能が設定されると、ダイナミック ifacl ラベルが予約されます。ただし、使用可能なダイナミック ifacl ラベル ビットは 2 つだけです。これらの機能の両方がすでに設定されている場合、ダイナミック ifacl ラベルは PTP で使用できず、機能を有効にすることはできません。
ステップ 3	（任意） [**no**] **ptp scale-on-1G** 例： `switch(config)# ptp scale-on-1G`	1G ポートの ASIC でタイムスタンプをイネーブルにします（デフォルトでは、PHY モードはイネーブルです）。 （注） グローバル CLI（ptp scale-on-1G）を有効または無効にする前に、各 1G インターフェイスで PTP を無効にしてください。そうでない場合、ロールバック構成と構成の置換の両方が失敗します。これは、

	コマンドまたはアクション	目的
		PHY ベースのタイム スタンプを ASIC ベースのタイム スタンプに移行する場合とその逆の両方に必要です。グローバル CLI は N9K-C93108TC-FX3P にのみ適用されます。
ステップ 4	[**no**] **ptp device-type** [**generalized-ptp** |**boundary-clock** |**ordinary-clock-grandmaster**] 例： `switch(config)# ptp device-type generalized-ptp`	デバイス タイプを gPTP または境界クロックあるいは通常のクロック グランドマスターとして設定します。 （注） この **generalized-ptp** オプションは、Cisco NX-OS リリース 7.0(3)PTP0(15) 以降の -R シリーズ ライン カード。 （注） Cisco NX-OS リリース 10.3(2)F 以降、**通常のクロック グランドマスター**は、Cisco Nexus N9K-C93180YC-FX3 プラットフォーム スイッチでのみ使用できます。詳細については、PTP GM の構成 （92 ページ）を参照してください。
ステップ 5	[**no**] **ptp source** {<*ipv4 address*> | <*ipv6 address*>} [**vrf** <*vrf-name*>] 例： `switch(config)# ptp source 10.10.10.1`	マルチキャスト PTP モードのすべての PTP パケットに、送信元 IPv4/IPv6 アドレスを設定します。 IP アドレスがデフォルト以外の VRF にある場合は、**vrf** パラメータを使用して VRF を構成します。 インターフェイスで PTP IPv4/IPv6 トランスポートを有効にする前に、対応するソース アドレス（IPv4/IPv6）が必要です。 （注） IPv6 ソースは、10.2(1)F リリース以降の Cisco Nexus 93180TC-FX3S スイッチでサポートされます。 Cisco NX-OS Release 10.2(2)F 以降では、このオプションは、Cisco Nexus 9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでも使用できます。
ステップ 6	（任意） [**no**] **ptp domain** *number* 例： `switch(config)# ptp domain 1`	このクロックで使用するドメイン番号を構成します。PTP ドメインを使用すると、1 つのネットワーク上で、複数の独立した PTP クロッキング サブドメインを使用できます。 指定できる数の範囲は 0 ～ 127 です。

	コマンドまたはアクション	目的
ステップ 7	（任意）[**no**] **ptp offload** 例： `switch(config)# ptp offload`	一部のタイマーをライン カードにオフロードすることで、PTP セッションの数を増やします。 この手順は、1 ステップ モードでは必須であり、2 ステップ モードではオプションです。 （注） dot1x（**feature dot1x**）と NV オーバーレイ（**feature nv overlay**）のどちらの機能も設定されていないことを確認します。これらの機能が設定されると、ダイナミック ifacl ラベルが予約されます。ただし、使用可能なダイナミック ifacl ラベル ビットは 2 つだけです。これらの機能のいずれかがすでに設定されている場合、ダイナミックifaclラベルはPTPオフロードに使用できず、機能を有効にすることはできません。PTP（**feature ptp**）は 1 つの ifacl ラベルを消費することに注意してください。 （注） Cisco NX-OS リリース 9.3(3) 以降、9636C-R、9636C-RX、または 9636Q-R ライン カードを搭載した Cisco Nexus 9504 および 9508 プラットフォーム スイッチは、1 ステップのクロック動作でのみオフロードをサポートします。PTPオフロードは、ワンステップ クロック動作が有効または無効になると、自動的に有効または無効になります。
ステップ 8	（任意）[**no**] **ptp clock-operation one-step** 例： `switch(config)# ptp clock-operation one-step`	PTP クロック動作を 1 ステップ モードに設定します。この場合、タイムスタンプ メッセージは同期メッセージの一部として送信されます。このモードでは、フォローアップ メッセージは送信されません。
ステップ 9	（任意）[**no**] **ptp priority1** *value* 例： `switch(config)# ptp priority1 1`	このクロックをアドバタイズするときに使用する priority1 の値を設定します。この値はベスト マスター クロック選択のデフォルトの基準（クロック品質、クロック クラスなど）を上書きします。低い値が優先されます。 *value* の範囲は 0 ～ 255 です。 （注） スイッチが外部グランドマスタークロックと同期するには、ローカル スイッチの PTP 優先順位の値を外部グランド マスター クロック の優先順位の値よりも大きく設定する必要があります。

	コマンドまたはアクション	目的
ステップ 10	（任意） [**no**] **ptp priority2** *value* 例： `switch(config)# ptp priority2 1`	このクロックをアドバタイズするときに使用する priority2 の値を構成します。この値は、デフォルトの基準では同等に一致する 2 台のデバイスのうち、どちらを優先するかを決めるために使用されます。たとえば、priority2 値を使用して、特定のスイッチが他の同等のスイッチよりも優先されるようにすることができます。 *value* の範囲は 0 ～ 255 です。 （注） スイッチが外部グランドマスタークロックと同期するには、ローカル スイッチの PTP 優先順位の値を外部グランド マスター クロック の優先順位の値よりも大きく設定する必要があります。

ステップ 11	[**no**] **ptp management**
	例：
	`switch(config)# ptp management`
	`switch(config-ptp-profile)#`
		PTP 管理パケットのサポートを設定します。このコマンドは、デフォルトでイネーブルになっています。
		no：管理パケットのサポートを無効にします。

ステップ 12	（任意） [**no**] **ptp delay tolerance** { **mean-path** | **reverse-path** } *variation*
	例：
	`switch(config)# ptp delay tolerance mean-path 50.5`
	`switch(config)#`
		PTP 遅延平均パス/リバース パスの許容差の変動を設定します。
		mean-path：PTP BMC アルゴリズムによって計算された平均パス遅延（MPD）のスパイクを無視します。
		reverse-path：PTP BMC アルゴリズムによって計算された（t4-t3）のスパイクを無視します。
		variation:：スパイクの許容度を定義するパーセンテージ。単一の 10 進数の数値を使用します。範囲は 1.0ー100.0 です。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 13	（任意） **ptp forward-version1**
	例：
	`switch(config)# ptp forward-version1`
	`switch(config)#`
		転送ルールに基づいてすべての PTPv1 パケットを転送するようにスイッチを設定します。
		（注）
		このコマンドを有効にしない場合、すべての PTPv1 パケットが CPU に渡され、最終的にドロップされます。

	コマンドまたはアクション	目的
		このコマンドは、Cisco NX-OS リリース 9.3(6) 以降でサポートされます。
ステップ 14	（任意） **ptp unicast-negotiation**	この構成は 10.2(1)F で導入され、93180YC-FX3S でサポートされます。 Cisco NX-OX リリース 10.2(2)F 以降、この構成は Cisco Nexus 9300-EX、9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでサポートされています。 有効にすると、すべての PTP ユニキャスト セッションがネゴシエート モードに移行します。 詳細については、「PTP ユニキャスト ネゴシエーション」のセクションを参照してください。
ステップ 15	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

PTP GM の構成

Cisco NX-OS 10.3(2)F リリース以降、PTP GM 機能をサポートするために新しく追加された CLI は次のとおりです。

手順の概要

1. [**no**] **ptp device-type ordinary-clock-grandmaster**
2. （任意） **ptp utc-offset** <*leap-seconds*> [**next-leapevent** <*date*> <*time*> **new-offset** <*new-leap-seconds*>]
3. **clock protocol gnss**
4. 外部 GNSS の構成については、「グランドマスター クロックの *GPS* 構成」の項を参照してください。内部 GNSS 構成については、「GNSS レシーバーの有効化 （146 ページ）」セクションを参照してください。

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	[**no**] **ptp device-type ordinary-clock-grandmaster** 例：	デバイス タイプを通常のクロック グランドマスターとして構成します。

	コマンドまたはアクション	目的
	`switch(config)# ptp device-type ordinary-clock-grandmaster`	
ステップ 2	（任意） **ptp utc-offset** <*leap-seconds*> [**next-leapevent** <*date*> <*time*> **new-offset** <*new-leap-seconds*>] 例： `switch(config)# ptp utc-offset 37 next-leap-event 2022-04-30 23:59:59 new-offset 38`	PTP GM 機能の **utc-offset** 値を設定します。デフォルト値は 37 秒です。 • **utc-offset**: GM クロックに追加する UTC オフセット値。 うるう秒 : うるう秒の値、範囲は 0 ～ 125 秒です。デフォルト値は 37 秒です。 **next-leapevent**: うるう秒が変わる時刻を設定するオプションのキーワード。 *date* : うるう秒の値が YYYY-MM-DD 形式で変化する日付。 *time* : うるう秒の値が HH:MM:SS 形式で変化する時刻 **new-offset** : 上記の構成された **utc-offset** 値の後に使用される新しい UTC オフセット値。 *new-leap-seconds* : 新しいうるう秒の値、範囲は 0～125 秒です。構成 **utc-offset** 値の±1のみです。
ステップ 3	**clock protocol gnss** 例： `switch(config)# clock protocol gnss`	デバイスのシステム クロックへの GNSS 時刻 (ToD) 同期を有効にします。
ステップ 4	外部 GNSS の構成については、「グランドマスタークロックの *GPS* 構成」の項を参照してください。内部 GNSS 構成については、「GNSS レシーバーの有効化 （146 ページ）」セクションを参照してください。	

インターフェイスでの PTP の設定

PTP をグローバルにイネーブルにしても、デフォルトで、サポートされているすべてのインターフェイス上でイネーブルになりません。PTP インターフェイスは個別にイネーブルに設定する必要があります。

始める前に

スイッチ上でグローバルに PTP をイネーブルにし、PTP 通信の送信元 IP アドレスを設定したことを確認します。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. [**no**] **ptp**
4. （任意） **ptp transport** {**ethernet** | **ipv4** | **ipv6** }
5. （任意） **ptp transmission** {**multicast** | **unicast** [**negotiation-schema** <*schema-name*>]}
6. （任意） **ptp role** { **dynamic** | **master** | **slave** }
7. （任意） [**no**] **ptp master** {<*ipv4-addr*> | <*ipv6-addr*>} { **negotiation-schema** <*schema-name*>}
8. （任意） [**no**] **ptp slave** {<*ipv4-addr*> | <*ipv6-addr*>}
9. （任意） **ptp peer** *ipv4-address* [**vrf** *vrf-name*]
10. **ptp multicast master-only**
11. （任意） **ptp ucast-source** {**<ipv4-addr> | <ipv6-addr>**} [**vrf** <*vrf-name*>]
12. （任意） [**no**] **ptp announce** {**interval** *log-seconds* | **timeout** *count*}
13. （任意） [**no**] **ptp delay-request minimum interval** *log-seconds*
14. （任意） [**no**] **ptp delay-request minimum interval** [**aes67-2015** | **smpte-2059-2**] *log-seconds*
15. （任意） [**no**] **ptp sync interval** *log-seconds*
16. （任意） [**no**] **ptp sync interval** [**aes67-2015** | **smpte-2059-2**] *log-seconds*
17. （任意） [**no**] **ptp vlan** *vlan-id*
18. （任意） **ptp destination-mac non-forwardable rx-no-match accept**
19. （任意） **show ptp brief**
20. （任意） **show ptp port interface** *interface slot/port*
21. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**interface ethernet** *slot/port*
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		PTP を有効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	[**no**] **ptp** 例： `switch(config-if)# ptp`	インターフェイスで PTP をイネーブルまたはディセーブルにします。

	コマンドまたはアクション	目的

ステップ 4	（任意） **ptp transport** {**ethernet** | **ipv4** | **ipv6** }
	例：
	`switch(config-if)# ptp transport ipv4`
	`switch(config-if)#`
	`switch(config-if)# ptp transport ipv6`
	`switch(config-if)#`
		PTP パケットの送信に使用されるトランスポート メカニズムを指定します。
		ethernet：PTP パケットは Eth フレーム（Eth / ptp）でのみ伝送されます。このオプションは、Cisco Nexus 93180YC-FX3S スイッチの PTP Telecom Profile でのみ使用できます。
		ipv4：PTP パケットは IPv4 で伝送されます。これがデフォルトの設定です。
		ipv6：PTP パケットは IPv6 で伝送されます。このオプションは、10.2(1)F リリース以降の Cisco Nexus 93180YC-FX3S スイッチで使用できます。 Cisco NX-OS Release 10.2(2)F 以降では、このオプションは、Cisco Nexus 9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでも使用できます。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 5	（任意） **ptp transmission** {**multicast** | **unicast** [**negotiation-schema** <*schema-name*>]}
	例：
	`switch(config-if)# ptp transmission multicast`
	`switch(config-if)#`
		インターフェイスで使用される PTP 伝送方式を設定します。
		multicast：PTP は、デバイス間の通信に IEEE 1588 標準に従ってマルチキャスト宛先 IP アドレス 224.0.1.129 を使用します。これがデフォルトの設定です。
		unicast：PTP メッセージは特定の PTP ピア ノードにユニキャストされます。
		negotiation schema <*schema-name*>：このオプションは、ユニキャスト ネゴシエーションがグローバルに有効になっている場合に使用でき、インターフェイスで使用するネゴシエーション スキーマを設定できます。
		このオプションは、Cisco NX-OS リリース 10.2(1)F 以降の Cisco Nexus 93180YC-FX3S スイッチで使用できます。 Cisco NX-OS Release 10.2(2)F 以降では、このオプションは、Cisco Nexus 9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでも使用できます。
		（注）

	コマンドまたはアクション	目的
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 6	（任意） **ptp role** { **dynamic** | **master** | **slave** }
	例：
	`switch(config-if)# ptp role dynamic`
	`switch(config-if)#`
		インターフェイスの PTP ロールを設定します。
		dynamic：ベスト マスター クロック アルゴリズム（BMCA）がロールを割り当てます。これは、デフォルト PTP プロファイルのデフォルト設定であり、G.8275.1 PTP プロファイルでのみ許可される設定です。
		Cisco NX-OS リリース 10.3(3)F 以降では、PTP 伝送がユニキャストであり、PTP ロールがダイナミックに設定されている場合は、PTP ユニキャスト ネゴシエーションを有効にしてください。
		master：マスター クロックは、インターフェイスの PTP ロールとして割り当てられます。
		slave：スレーブ クロックがインターフェイスの PTP ロールとして割り当てられます。
		Cisco NX-OS リリース 10.3(3)F 以降、PTP ロールの **slave**は、複数のユニキャスト ポートで構成できます。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 7	（任意） [**no**] **ptp master** {<*ipv4-addr*> | <*ipv6-addr*>} { **negotiation-schema** <*schema-name*>}
	例：
	`switch(config-if)# ptp master 10.10.10.1`
	`switch(config-if)#`
		（任意）インターフェイスの PTP ロールが「slave」に設定されている場合に、マスター クロックの IP アドレスを設定します。
		negotiation-schema ：これは、ユニキャスト ネゴシエーションがグローバルに有効になっている場合に、マスターの特定のネゴシエーション スキーマを設定するために使用できます。このオプションは、Cisco NX-OS リリース 10.2(1)F 以降の Cisco Nexus 93180YC-FX3S スイッチで使用できます。Cisco NX-OS Release 10.2(2)F 以降では、このオプションは、Cisco Nexus 9300-EX、9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでも使用できます。
		（注）

	コマンドまたはアクション	目的
		このコマンドは、ユニキャスト マスターを設定し、伝送がユニキャストに設定されている場合に使用されます。 このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。 IPv6 は、Cisco NX-OS リリース 10.2(1)F 以降の Cisco Nexus 93180YC-FX3S でサポートされます。 Cisco NX-OS Release 10.2(2)F 以降では、IPv6 は Cisco Nexus 9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでもサポートされます。

ステップ 8	（任意） [**no**] **ptp slave** {<*ipv4-addr*> | <*ipv6-addr*>}
	例：
	`switch(config-if)# ptp slave 10.10.10.2`
	`switch(config-if)#`
		（任意）インターフェイスの PTP ロールが「master」に設定されている場合に、マスター クロックの IP アドレスを設定します。
		（注）
		このコマンドは、ユニキャスト スレーブを設定し、伝送がユニキャストに設定されている場合に使用されます。
		
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。
		
		IPv6 は、Cisco NX-OS リリース 10.2(1)F 以降の Cisco Nexus 93180YC-FX3S でサポートされます。 Cisco NX-OS Release 10.2(2)F 以降では、IPv6 は Cisco Nexus 9300-FX、9300-FX2、9300-GX、および 9300-GX2 プラットフォーム スイッチでサポートされます。

ステップ 9	（任意） **ptp peer** *ipv4-address* [**vrf** *vrf-name*]
	例：
	`switch(config-if)# ptp peer 10.10.10.2`
	`switch(config-if)#`
		インターフェイスの PTP ロールが動的に設定されている場合に、ピア クロックの IP アドレスを設定します。
		（注）
		• 構成できるピアは 1 つだけです。
		• このコマンドは、ユニキャスト ピアを構成し、伝送がユニキャストに設定されていて、ロールが動的に設定されている場合に使用されます。
		• このコマンドは、Cisco NX-OS Release 10.3(3)F で導入されました。

	コマンドまたはアクション	目的

ステップ 10	**ptp multicast master-only**
	例：
	`switch(config)# ptp multicast master-only`
	`switch(config)#`
		インターフェイスの PTP ロールとして割り当てられるマスター クロックを設定します。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) で廃止され、将来のリリースではサポートされません。必要に応じて、ステップ 4 ー 8 のコマンドを使用してください。

ステップ 11	（任意） **ptp ucast-source** {**<ipv4-addr> | <ipv6-addr>**} [**vrf** <*vrf-name*>] 例： `switch(config)# ptp ucast-source 10.1.1.40`	（任意）ユニキャスト メッセージの送信元 IP アドレスを設定します。 *ipv4-address*：ユニキャスト送信元の IPv4 アドレス。トランスポートが IPv4 に設定されている場合に使用されます。 *ipv6-address*：ユニキャスト送信元の IPv6 アドレス。これは、トランスポートが IPv6 に設定されている場合に使用されます。 **vrf** *vrf-name*：hello メッセージに使用される VRF の名前。 （注） IPv6 は、Cisco NX-OS リリース 10.2(1)F 以降の Cisco Nexus 93180YC-FX3S でサポートされます。Cisco NX-OS Release 10.2(2)F 以降では、IPv6 は Cisco Nexus 9300-FX、9300-FX2、9300-GX、および9300-GX2 プラットフォーム スイッチでもサポートされます。
ステップ 12	（任意） [**no**] **ptp announce** {**interval** *log-seconds* | **timeout** *count*} 例： `switch(config-if)# ptp announce interval 3`	インターフェイス上の PTP アナウンス メッセージ間の間隔またはタイムアウトがインターフェイスで発生する前の PTP 間隔の数を設定します。 PTP アナウンス間隔の範囲は 0 ～ 4 ログ秒で、間隔のタイムアウトの範囲は 2 ～ 4 間隔です。
ステップ 13	（任意） [**no**] **ptp delay-request minimum interval** *log-seconds* 例： `switch(config-if)# ptp delay-request minimum interval -1`	ポートがマスター ステートの場合に PTP 遅延メッセージ間で許可される最小間隔を設定します。 範囲は log（–1）ー log（6）秒です。ここで、log（–1）は毎秒 2 フレームです。
ステップ 14	（任意） [**no**] **ptp delay-request minimum interval** [**aes67-2015** | **smpte-2059-2**] *log-seconds* 例：	ポートがマスター ステートの場合に PTP 遅延メッセージ間で許可される最小間隔を設定します。

	コマンドまたはアクション	目的
	`switch(config-if)# ptp delay-request minimum interval aes67-2015-1`	**表 *4 : PTP* 遅延要求の最小間隔の範囲とデフォルト値** オプション / 範囲 / デフォルト値 **aes67-2015** / -4 ～ 5 ログ秒 / 0 ログ秒 **smpte-2059-2** / -4 ～ 5 ログ秒 / 0 ログ秒 **aes67-2015** または **smpte-2059-2** オプションなし / –1 ～ 6 ログ秒（ここで、–1 = 2 フレーム毎秒） / 0 ログ秒
ステップ 15	（任意） [**no**] **ptp sync interval** *log-seconds* **例：** `switch(config-if)# ptp sync interval 1`	インターフェイス上の PTP 同期メッセージの送信間隔を設定します。 範囲は、log（–3）～log（1）秒です。メディア関連のプロファイル情報については、『メディア ソリューション ガイド向け Cisco NX-OS IP ファブリック』を参照してください。
ステップ 16	（任意） [**no**] **ptp sync interval** [**aes67-2015** | **smpte-2059-2**] *log-seconds* **例：** `switch(config-if)# ptp sync interval aes67 1`	インターフェイス上の PTP 同期メッセージの送信間隔を設定します。 **表 *5 : PTP* 同期間隔の範囲とデフォルト値** オプション / 範囲 / デフォルト値 **aes67-2015** / -4 ～ 1 ログ秒 / – 2 ログ秒 **smpte-2059-2** / -4 ～ -1 ログ秒 / -2 ログ秒 **aes67-2015** または **smpte-2059-2** オプションなし / -3 ～ 1 ログ秒 / -2 ログ秒
ステップ 17	（任意） [**no**] **ptp vlan** *vlan-id* **例：** `switch(config-if)# ptp vlan 1`	PTP をイネーブルにするインターフェイスの VLAN を指定します。インターフェイスの 1 つの VLAN でイネーブルにできるのは、1 つの PTP のみです。 指定できる範囲は 1 ～ 4094 です。

ステップ 18	（任意） **ptp destination-mac non-forwardable rx-no-match accept**
	例：
	`switch(config-if)# ptp destination-mac non-forwardable rx-no-match accept`
	`switch(config-if)#`
		転送不能な宛先 MAC アドレス パケットを受け入れ、応答します。これらの宛先 MACアドレスは、GM クロック、PTP マスター クロック、および PTP スレーブ クロック間で交換される PTP メッセージで使用されます。

	コマンドまたはアクション	目的
		このコマンドはCisco NX-OS リリース 9.3(5) 以降でサポートされ、Cisco Nexus 93180YC-FX3S スイッチのみでサポートされます。
ステップ 19	（任意） **show ptp brief** 例： `switch(config-if)# show ptp brief`	PTP のステータスを表示します。
ステップ 20	（任意） **show ptp port interface** *interface slot/port* 例： `switch(config-if)# show ptp port interface ethernet 2/1`	PTP ポートのステータスを表示します。
ステップ 21	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

ユニキャストモードでの PTP の設定

IPv4 または IPv6 向けユニキャスト モードの構成

従来の PTP メッセージは、PTP マルチキャスト メッセージを受信できるノードに配信されます。（たとえば、**announce**、**sync**、**delay_req**、**delay_resp** および **follow_up**）。ユニキャストモードでは、すべて のPTP メッセージが特定の PTP ノードにのみ配信されます。マルチキャストアドレスは使用されません。ユニキャスト モードでは、マスター/スレーブ ロールを設定し、対応するピア スレーブ/マスター IP アドレスを割り当てることができます。

スレーブ ユニキャスト ポートには最大 8 個のマスター IP を設定でき、マスター ポートには最大 64 個のスレーブ IP を設定でき、すべてのポートで最大 256 個のスレーブ IP を設定できます。ユニキャスト スレーブ IP とユニキャスト マスター IP を設定するには、次のコマンドを使用します。ユニキャスト パケットは、これらの IP との間でのみ送受信されます。他の IP から受信したパケットは無視されます。

Cisco NX-OS リリース 10.2(1)F 以降の場合：

```
IPv4 config
interface Ethernet1/34
ptp
ptp transport ipv4
ptp transmission unicast
ptp role master
ptp slave 10.10.10.2
ptp ucast-source 10.10.10.1

interface Ethernet1/35
ptp
ptp transport ipv4
ptp transmission unicast
```

```
ptp role slave
ptp master 10.10.10.1
ptp ucast-source 10.10.10.2
```

IPv6 config

```
interface Ethernet1/34
ptp
ptp transport ipv6
ptp transmission unicast
ptp role master
ptp slave 2012:a1:0:0:0:0:0:2
ptp ucast-source 2012:a1:0:0:0:0:0:1

interface Ethernet1/35
ptp
ptp transport ipv6
ptp transmission unicast
ptp role slave
ptp master 2012:a1:0:0:0:0:0:1
ptp ucast-source 2012:a1:0:0:0:0:0:2
```

Cisco NX-OS リリース 9.3(5) 以降の場合：

```
switch(config-if)# ptp
switch(config-if)# ptp transmission unicast
switch(config-if)# ptp role master
switch(config-if)# ptp slave 10.10.10.2

switch(config-if)# ptp
switch(config-if)# ptp transmission unicast
switch(config-if)# ptp role slave
switch(config-if)# ptp master 10.10.10.1
```

Cisco NX-OS リリース 9.3(4) 以前の場合：

```
switch(config-if)# ptp transport ipv4 ucast master
switch(config-if-ptp-master)# slave ipv4 10.10.10.2

switch(config-if)# ptp transport ipv4 ucast slave
switch(config-if-ptp-slave)# master ipv4 10.10.10.1
```

動的ロールの割り当て

動的ロールを割り当てるには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. [**no**] **ptp**
4. **ptp transmission unicast**
5. **ptp role dynamic**
6. **ptp peer** *ipv4-address*

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**interface ethernet** *slot/port*
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		PTP を有効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	[**no**] **ptp**
	例：
	`switch(config-if)# ptp`
	`switch(config-if)#`
		インターフェイスで PTP をイネーブルまたはディセーブルにします。
		（注）
		このコマンドは、以下のユニキャスト構成コマンドをインターフェイスに適用する前に必要となるものです。

ステップ 4	**ptp transmission unicast**
	例：
	`switch(config-if)# ptp transmission unicast`
	`switch(config-if)#`
		インターフェイスで使用される PTP 伝送方式を設定します。

ステップ 5	**ptp role dynamic**
	例：
	`switch(config-if)# ptp role dynamic`
	`switch(config-if)#`
		インターフェイスの PTP ロールを設定します。
		dynamic：動的クロックがインターフェイスの PTP ロールとして割り当てられます。

ステップ 6	**ptp peer** *ipv4-address*
	例：
	`switch(config-if)# ptp peer 10.10.10.1`
	`switch(config-if)#`
		インターフェイスの PTP ロールが動的に設定されている場合に、ピア クロックの IP アドレスを設定します。

マスター ロールの割り当て

マスター ロールを割り当てるには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. [**no**] **ptp transport ipv4 ucast master**

4. **slave ipv4** *<IP_address>*
5. [**no**] **ptp**
6. **ptp transmission unicast**
7. **ptp role master**
8. **ptp slave** *ipv4-address*

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**interface ethernet** *slot/port*
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		PTP を有効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。
		（注）
		このコマンドを設定した後、Cisco NX-OS リリース 9.3(5) 以降の場合は、ステップ 5 に進みます。Cisco NX-OS リリース 9.3(4) 以前の場合は、ステップ 3 に進みます。

ステップ 3	[**no**] **ptp transport ipv4 ucast master**
	例：
	`switch(config-if)# ptp transport ipv4 ucast master`
	`switch(config-if-ptp-master)#`
		特定のポート（レイヤ 3 インターフェイス）で PTP マスターをイネーブルにします。マスター サブモードでは、スレーブ IPv4 アドレスを入力できます。

ステップ 4	**slave ipv4** *<IP_address>*
	例：
	`switch-1(config)# interface ethernet 1/1`
	`switch-1(config-if)# ptp transport ipv4 ucast master`
	`switch-1(config-if-ptp-master)# slave ipv4 1.2.3.1`
	`switch-1(config-if-ptp-master)# slave ipv4 1.2.3.2`
	`switch-1(config-if-ptp-master)# slave ipv4 1.2.3.3`
	`switch-1(config-if-ptp-master)# slave ipv4 1.2.3.4`
	`switch-1(config-if-ptp-master)#`
		スレーブ IPv4 アドレスを入力します。マスターごとに最大 64 個の IP アドレスを使用できますが、実際の数は同期間隔の設定に応じて変わります。マスターは、これらのスレーブ アドレスにのみ、アナウンス、同期、フォローアップ、および delay_resp を送信します。スレーブ IP が到達可能であることを確認する必要があります。
		（注）
		Cisco NX-OS リリース9.3 (4) 以前の場合は、これで手順は終了です。

ステップ 5	[**no**] **ptp** 例：	インターフェイスで PTP をイネーブルまたはディセーブルにします。

	コマンドまたはアクション	目的
	`switch(config-if)# ptp` `switch(config-if)#`	（注） 9.3(5) 以降では、このコマンドは、以下のユニキャスト コンフィギュレーション コマンドをインターフェイスに適用する前に必要です。

ステップ 6	**ptp transmission unicast**
	例：
	`switch(config-if)# ptp transmission unicast`
	`switch(config-if)#`
		インターフェイスで使用されるPTP伝送方式を設定します。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 7	**ptp role master**
	例：
	`switch(config-if)# ptp role master`
	`switch(config-if)#`
		インターフェイスの PTP ロールを設定します。
		master：マスタークロックは、インターフェイスの PTP ロールとして割り当てられます。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 8	**ptp slave** *ipv4-address*
	例：
	`switch(config-if)# ptp slave 10.10.10.2`
	`switch(config-if)#`
		インターフェイスの PTP ロールが「master」に設定されている場合に、スレーブ クロックの IP アドレスを設定します。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

スレーブ ロールの割り当て

スレーブ ロールを割り当てるには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. [**no**] **ptp transport ipv4 ucast slave**
4. **master ipv4** *<IP_address>*
5. [**no**] **ptp**
6. **ptp transmission unicast**
7. **ptp role slave**
8. **ptp master** *ipv4-address*

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**interface ethernet** *slot/port*
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		PTP を有効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。
		（注）
		このコマンドを設定した後、Cisco NX-OS リリース 9.3(5) 以降の場合は、ステップ 5 に進みます。Cisco NX-OS リリース 9.3(4) 以前の場合は、ステップ 3 に進みます。

ステップ 3	[**no**] **ptp transport ipv4 ucast slave**
	例：
	`switch(config-if)# ptp transport ipv4 ucast slave`
	`switch(config-if-ptp-slave)#`
		特定のポート（レイヤ 3 インターフェイス）で PTP スレーブをイネーブルにします。スレーブ サブモードでは、ユーザーはマスター IPv4 アドレスを入力できます。

ステップ 4	**master ipv4** *<IP_address>*
	例：
	`switch-1(config)# interface ethernet 1/1`
	`switch-1(config-if)# ptp transport ipv4 ucast slave`
	`switch-1(config-if-ptp-slave)# master ipv4 4.4.4.1`
	`switch-1(config-if-ptp-slave)# master ipv4 4.4.4.2`
	`switch-1(config-if-ptp-slave)# master ipv4 4.4.4.3`
		マスター IPv4 アドレスを入力します。
		（注）
		Cisco NX-OS リリース9.3 (4) 以前の場合は、これで手順は終了です。

ステップ 5	[**no**] **ptp**
	例：
	`switch(config-if)# ptp`
	`switch(config-if)#`
		インターフェイスで PTP をイネーブルまたはディセーブルにします。
		（注）
		このコマンドは、9.3(5) 以降で、以下のユニキャスト コンフィギュレーション コマンドをインターフェイスに適用する前に必要となるものです。

ステップ 6	**ptp transmission unicast**
	例：
	`switch(config-if)# ptp transmission unicast`
	`switch(config-if)#`
		インターフェイスで使用される PTP 伝送方式を設定します。
		（注）

	コマンドまたはアクション	目的
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 7	**ptp role slave**
	例：
	`switch(config-if)# ptp role slave`
	`switch(config-if)#`
		インターフェイスの PTP ロールを設定します。
		slave：スレーブ クロックがインターフェイスの PTP ロールとして割り当てられます。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 8	**ptp master** *ipv4-address*
	例：
	`switch(config-if)# ptp master 10.10.10.1`
	`switch(config-if)#`
		インターフェイスの PTP ロールが「slave」に設定されている場合、マスター クロックの IP アドレスを設定します。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ユニキャスト送信元アドレスの設定

（注） Cisco NX-OS リリース 9.3(4) までのすべてのリリースで、インターフェイスの PTP 設定がユニキャストからマルチキャストまたはユニキャスト スレーブからユニキャスト マスターに変更された場合は、ユニキャスト送信元アドレスを再設定する必要があります。

Cisco NX-OS リリース 9.3(5) 以降では、インターフェイスの PTP 設定がユニキャストからマルチキャストまたはユニキャスト スレーブからユニキャスト マスターに変更された場合、ユニキャスト送信元アドレスを再設定する必要はありません。

ユニキャスト送信元アドレスを設定するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot/port*
3. [**no**] **ptp ucast-source** {<*ipv4-addr*> | <*ipv6-addr*>} [**vrf** <*vrf-name*>

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**interface ethernet** *slot/port*
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		PTP を有効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	[**no**] **ptp ucast-source** {<*ipv4-addr*> | <*ipv6-addr*>} [**vrf** <*vrf-name*>
	例：
	`switch(config-if)# ptp ucast-source 10.10.10.20`
	`switch(config-if)#`
		インターフェイス レベルごとに PTP 送信元アドレスを設定します。この IP アドレスは、ユニキャスト PTP メッセージにのみ使用されます。PTP ユニキャスト送信元 IP アドレスが到達可能である必要があります。
		ipv4-address：ユニキャスト送信元の IPv4 アドレス。トランスポートが IPv4 に設定されている場合に使用されます。
		ipv6-address：ユニキャスト送信元の IPv6 アドレス。これは、トランスポートが IPv6 に設定されている場合に使用されます。
		vrf*vrf-name*：IP アドレスがデフォルト以外の VRF にある場合は、vrf パラメータを使用して VRF を構成します。

PTP テレコム プロファイルの設定

グローバル PTP テレコム プロファイルの設定

この手順では、クロックとその設定を含む PTP テレコム プロファイルを、周波数に合った ITU-T テレコム プロファイルと一致するように設定する手順を説明します。

始める前に

QoS TCAM リージョンの入力 SUP [ingress-sup] は、768 以上に設定する必要があります。手順は以下のとおりです。

1. **show hardware access-list tcam region** コマンドを使用して、TCAM リージョンを確認します。

2. 入力 SUP リージョンが 768 以上に設定されていない場合は、**hardware access-list tcam region ing-sup 768** コマンドを使用して入力 SUP TCAM リージョンを設定します。実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーし（**copy running-config startup-config**）、スイッチをリロードします。

手順の概要

1. **configure terminal**
2. **feature ptp**
3. **ptp profile** { **default** | **8275-1** | **8275-2** }
4. プロファイルのデフォルト：**mode** { **hybrid** | **non-hybrid** | **none** }
5. **exit**
6. **ptp source** *ip-address*
7. プロファイルのデフォルト：**ptp priority1** *value*
8. プロファイルのデフォルト：**ptp priority2** *value*
9. **ptp pdelay-req-interval** *value*
10. プロファイルのデフォルト：**ptp domain** *value*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	必須: **feature ptp**
	例：
	`switch(config)# feature ptp`
	`switch(config)#`
		グローバル PTP 機能をイネーブルにします。

ステップ 3	必須: **ptp profile** { **default**	**8275-1**	**8275-2** }
			例：
			`switch(config)# ptp profile 8275-1`
			`switch(config-ptp-profile)#`
				PTP プロファイルをイネーブルにし、PTP プロファイル コンフィギュレーション モードを開始します。このコマンドのプロファイル タイプでサポートされるコマンドの詳細については、次を参照してください：
				（注）
				8275-1 および 8275-2 は PTP テレコム プロファイル構成をサポートします。

	コマンドまたはアクション	目的
		Cisco NX-OS リリース 9.3(5) では、Cisco Nexus 93180YC-FX3S スイッチのみが、このコマンドのどちらかのオプションをサポートします。

ステップ 4	プロファイルのデフォルト：**mode** { **hybrid** | **non-hybrid** | **none** }
	例：
	`switch(config)# mode hybrid`
	`switch(config-ptp-profile)#`
		スイッチの PTP 動作モードを設定します。
		hybrid：SyncE ソースは PTP ソースとして機能します。
		default：local/1588 クロックは PTP ソースとして機能します。
		（注）
		• このコマンドは、**ptp profile** コマンドが設定されると自動的に設定されます。設定値は変更できません。詳細については、「ステップ 3（108 ページ）」を参照してください。
		• 8275-2 プロファイルがサポートする唯一のモードは none です。

ステップ 5	**exit**
	例：
	`switch(config-ptp-profile)# exit`
	`switch(config)#`
		PTP プロファイル コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。

ステップ 6	**ptp source** *ip-address*
	例：
	`switch(config)# ptp source 10.10.10.20`
	`switch(config)#`
		マルチキャスト PTP モードのすべての PTP パケットに、送信元 IPv4 アドレスを設定します。

ステップ 7	プロファイルのデフォルト：**ptp priority1** *value*
	例：
	`switch(config)# ptp priority1 128`
	`switch(config)#`
		このクロックをアドバタイズするときに使用する priority1 の値を設定します。このクロックをアドバタイズするときに使用する priority1 の値を設定します。低い値が優先されます。
		（注）
		このコマンドは、**ptp profile 8275-1 or 8275-2** グローバル コマンドが設定されると自動的に設定されます。設定値は変更できません。「ステップ 3（108 ページ）」を参照してください。

ステップ 8	プロファイルのデフォルト：**ptp priority2** *value*
	例：
	`switch(config)# ptp priority2 128`
	`switch(config)#`
		このクロックをアドバタイズするときに使用する priority2 の値を設定します。このクロックをアドバタイズするときに使用する priority1 の値を設定します。低い値が優先されます。

	コマンドまたはアクション	目的
		デフォルト：128 範囲：0 ー 255 （注） このコマンドは、**ptp profile 8275-1** または **8275-2** グローバル コマンドが設定されると自動的に構成されます。ステップ 3 （108 ページ） を参照してください。

ステップ 9	**ptp pdelay-req-interval** *value*
	例：
	`switch(config)# ptp pdelay-req-interval 0`
	`switch(config)#`
		ピア遅延要求間隔を設定します。
		value：範囲は 0 ～ 5 です。

ステップ 10	プロファイルのデフォルト：**ptp domain** *value*
	例：
	`switch(config)# ptp domain 24`
	`switch(config)#`
		PTP クロック ドメイン値を指定します。G.8275.1 プロファイルで許可されるドメイン番号の範囲は 24 ー 43 です。デフォルトは 24 です。G.8275.2 プロファイルで許可されるドメイン番号の範囲は 44 ー 63 です。デフォルトは 44 です。
		（注）
		このコマンドは、**ptp profile 8275-1** または **8275-2** グローバル コマンドが設定されると自動的に構成されます。「ステップ 3 （108 ページ）」を参照してください。

PTP テレコム プロファイル 8275-1 のインターフェイスの構成

この手順では、インターフェイスの PTP テレコム プロファイル 8275-1 を構成する手順を説明します。

（注） この手順で説明する一部のコマンドは、**ptp profile 8275-1** グローバル コマンドが設定され、インターフェイスでPTPが有効になっている場合に自動的に有効になり、設定されます。詳細については、「グローバル PTP テレコム プロファイルの設定 （107 ページ）」を参照してください。

始める前に

この手順は、インターフェイスでの周波数同期の設定とともに、「ハイブリッドPTP」プラットフォームに必要なインターフェイス設定を構成します。インターフェイスの周波数の同期化の設定の詳細については、インターフェイスの周波数の同期の設定 （63 ページ） を参照してください。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot* / *port*
3. [**no**] **ptp**
4. プロファイルのデフォルト： **ptp transport ethernet**
5. プロファイルのデフォルト： **ptp transmission multicast**
6. プロファイルのデフォルト： **ptp role dynamic**
7. （任意） **ptp destination-mac non-forwardable rx-no-match accept**
8. プロファイルのデフォルト：**ptp cost** *value*
9. プロファイルのデフォルト：**ptp delay-request minimum interval** *log-seconds*
10. プロファイルのデフォルト：**ptp announce interval** *log-seconds*
11. プロファイルのデフォルト：**ptp sync interval** *log-seconds*
12. （任意） [**no**] **ptp announce timeout** *count*
13. （任意） [**no**] **ptp profile-override**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface ethernet** *slot* / *port*
	例：
	`switch(config)# interface ethernet 1/5`
	`switch(config-if)#`
		PTP テレコム プロファイル パラメータを設定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	[**no**] **ptp**
	例：
	`switch(config-if)# ptp`
	`switch(config-if)#`
		インターフェイスで PTP を有効にします。

ステップ 4	プロファイルのデフォルト： **ptp transport ethernet**
	例：
	`switch(config-if)# ptp transport ethernet`
	`switch(config-if)#`
		PTP パケットの送信に使用されるトランスポート メカニズムを指定します。**ethernet** については、PTP パケットは Eth フレーム（Eth / ptp）でのみ伝送されます。
		（注）
		このコマンドは、**ptp profile 8275-1** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-1** コマンドの詳細については、グローバル PTP

	コマンドまたはアクション	目的
		テレコム プロファイルの設定 （107 ページ） を参照してください。

ステップ 5	プロファイルのデフォルト： **ptp transmission multicast**
	例：
	`switch(config-if)# ptp transmission multicast`
	`switch(config-if)#`
		インターフェイスで使用される PTP 伝送方式を設定します。**multicast** に関して、IEEE 1588 標準に従って、PTP はデバイス間の通信にマルチキャスト宛先 IP アドレス 224.0.1.129 を使用します。
		（注）
		このコマンドは、**ptp profile 8275-1** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-1** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定 （107 ページ） を参照してください。

ステップ 6	プロファイルのデフォルト： **ptp role dynamic**
	例：
	`switch(config-if)# ptp role dynamic`
	`switch(config-if)#`
		インターフェイスの PTP ロールを設定します。**dynamic** では、ベスト マスター クロック アルゴリズム（BMCA）がロールを割り当てます。
		（注）
		このコマンドは、**ptp profile 8275-1** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-1** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定 （107 ページ） を参照してください。

ステップ 7	（任意） **ptp destination-mac non-forwardable rx-no-match accept**
	例：
	`switch(config-if)# ptp destination-mac non-forwardable rx-no-match accept`
	`switch(config-if)#`
		転送不能な宛先 MAC アドレス パケットを受け入れ、応答します。これらの宛先 MACアドレスは、GM クロック、PTP マスター クロック、および PTP スレーブ クロック間で交換される PTP メッセージで使用されます。

ステップ 8	プロファイルのデフォルト：**ptp cost** *value*
	例：
	`switch(config-if)# ptp cost 128`
	`switch(config-if)#`
		BMCA の最適なマスター クロックの選択で使用される値を設定します。標準に記載されているすべてのパラメータが同じ場合、このローカル プライオリティが使用されます。
		（注）
		このコマンドは、**ptp profile 8275-1** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-1** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定 （107 ページ） を参照してください。

	コマンドまたはアクション	目的
ステップ 9	プロファイルのデフォルト：**ptp delay-request minimum interval** *log-seconds* 例： `switch(config-if)# ptp delay-request minimum interval -4`	ポートがマスター ステートの場合に PTP 遅延メッセージ間で許可される最小間隔を設定します。 （注） このコマンドは、**ptp profile 8275-1** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-1** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定 （107 ページ） を参照してください。
ステップ 10	プロファイルのデフォルト：**ptp announce interval** *log-seconds* 例： `switch(config-if)# ptp announce interval -3`	インターフェイス上の PTP アナウンス メッセージ間の間隔またはタイムアウトがインターフェイスで発生する前の PTP 間隔の数を設定します。 （注） このコマンドは、**ptp profile 8275-1** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-1** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定 （107 ページ） を参照してください。
ステップ 11	プロファイルのデフォルト：**ptp sync interval** *log-seconds* 例： `switch(config-if)# ptp sync interval -4`	インターフェイス上の PTP 同期メッセージの送信間隔を設定します。 （注） このコマンドは、**ptp profile 8275-1** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-1** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定 （107 ページ） を参照してください。
ステップ 12	（任意） [**no**] **ptp announce timeout** *count* 例： `switch(config-if)# ptp announce timeout 3`	タイムアウトがインターフェイスで発生する前の PTP 間隔の数を設定します。 PTP アナウンスのタイムアウト間隔の範囲は 2 ～ 4 です。

ステップ 13	（任意） [**no**] **ptp profile-override**
	例：
	`switch(config-if)# ptp profile-override`
	`switch(config-if)#`
		デフォルトで [無効（Disabled）] になっており、有効にすると、このインターフェイス設定で次のコマンドを変更できます。
		• **ptp transport**
		• **ptp announce interval**
		• **ptp delay-request minimum interval**
		• **ptp sync interval**

	コマンドまたはアクション	目的
		• **ptp cost** （8275-1 プロファイルのみ） （注） 有効にすると、グローバル PTP プロファイルが変更されても、コマンドへの変更はデフォルトにリセットされません。**ptp profile-override** を削除すると、インターフェイスの PTP 設定がグローバル プロファイルに対応するデフォルト値にリセットされます。

PTP テレコム プロファイル 8275-2 のインターフェイスの構成

この手順では、インターフェイスの PTP テレコム プロファイル 8275-2 を構成する手順を説明します。

（注） この手順で説明する一部のコマンドは、**ptp profile 8275-2** グローバル コマンドが設定され、インターフェイスでPTPが有効になっている場合に自動的に有効になり、設定されます。詳細については、「グローバル PTP テレコム プロファイルの設定 （107 ページ）」を参照してください。

始める前に

この手順は、インターフェイスでの周波数同期の設定とともに、「ハイブリッドPTP」プラットフォームに必要なインターフェイス設定を構成します。インターフェイスの周波数の同期化の設定の詳細については、インターフェイスの周波数の同期の設定 （63 ページ） を参照してください。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot* / *port*
3. [**no**] **ptp**
4. プロファイルのデフォルト： **ptp transport ipv4**
5. プロファイルのデフォルト： **ptp transmission unicast**
6. プロファイルのデフォルト： **ptp role dynamic**
7. （任意） **ptp destination-mac non-forwardable rx-no-match accept**
8. プロファイルのデフォルト：**ptp cost** *value*
9. プロファイルのデフォルト：**ptp delay-request minimum interval** *log-seconds*
10. プロファイルのデフォルト：**ptp announce interval** *log-seconds*
11. プロファイルのデフォルト：**ptp sync interval** *log-seconds*
12. （任意） [**no**] **ptp announce timeout** *count*

13. （任意）[**no**] **ptp profile-override**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface ethernet** *slot* / *port*
	例：
	`switch(config)# interface ethernet 1/5`
	`switch(config-if)#`
		PTP テレコム プロファイル パラメータを設定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	[**no**] **ptp**
	例：
	`switch(config-if)# ptp`
	`switch(config-if)#`
		インターフェイスで PTP を有効にします。

ステップ 4	プロファイルのデフォルト： **ptp transport ipv4**
	例：
	`switch(config-if)# ptp transport ipv4`
	`switch(config-if)#`
		PTP パケットの送信に使用されるトランスポート メカニズムを指定します。
		（注）
		このコマンドは、**ptp profile 8275-2** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-2** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定 （107 ページ） を参照してください。

ステップ 5	プロファイルのデフォルト： **ptp transmission unicast**
	例：
	`switch(config-if)# ptp transmission unicast`
	`switch(config-if)#`
		インターフェイスで使用される PTP 伝送方式を設定します。**unicast** の場合、PTP は構成されたマスター/スレーブ/ピア IP を使用します。
		（注）
		このコマンドは、**ptp profile 8275-2** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-2** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定 （107 ページ） を参照してください。

	コマンドまたはアクション	目的

ステップ 6	プロファイルのデフォルト： **ptp role dynamic**
	例：
	`switch(config-if)# ptp role dynamic`
	`switch(config-if)#`
		インターフェイスの PTP ロールを設定します。**dynamic** では、ベスト マスター クロック アルゴリズム（BMCA）がロールを割り当てます。
		（注）
		このコマンドは、**ptp profile 8275-2** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-2** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定（107 ページ） を参照してください。

ステップ 7	（任意） **ptp destination-mac non-forwardable rx-no-match accept**
	例：
	`switch(config-if)# ptp destination-mac non-forwardable rx-no-match accept`
	`switch(config-if)#`
		転送不能な接続先MACアドレスのパケットを受け入れ、応答します。これらの宛先 MACアドレスは、GM クロック、PTP マスター クロック、および PTP スレーブ クロック間で交換される PTP メッセージで使用されます。

ステップ 8	プロファイルのデフォルト：**ptp cost** *value*
	例：
	`switch(config-if)# ptp cost 128`
	`switch(config-if)#`
		BMCA の最適なマスター クロックの選択で使用される値を設定します。標準に記載されているすべてのパラメータが同じ場合、このローカル プライオリティが使用されます。
		（注）
		このコマンドは、**ptp profile 8275-2** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-2** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定（107 ページ） を参照してください。

ステップ 9	プロファイルのデフォルト：**ptp delay-request minimum interval** *log-seconds* 例： `switch(config-if)# ptp delay-request minimum interval -4`	ポートがマスター ステートの場合に PTP 遅延メッセージ間で許可される最小間隔を設定します。 （注） このコマンドは、**ptp profile 8275-2** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-2** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定（107 ページ） を参照してください。
ステップ 10	プロファイルのデフォルト：**ptp announce interval** *log-seconds* 例： `switch(config-if)# ptp announce interval -3`	インターフェイス上の PTP アナウンス メッセージ間の間隔またはタイムアウトがインターフェイスで発生する前の PTP 間隔の数を設定します。 （注）

	コマンドまたはアクション	目的
		このコマンドは、**ptp profile 8275-2** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-2** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定（107 ページ） を参照してください。
ステップ 11	プロファイルのデフォルト：**ptp sync interval** *log-seconds* **例：** `switch(config-if)# ptp sync interval -4`	インターフェイス上の PTP 同期メッセージの送信間隔を設定します。 **（注）** このコマンドは、**ptp profile 8275-2** global コマンドが設定されると自動的に設定されます。**ptp profile 8275-2** コマンドの詳細については、グローバル PTP テレコム プロファイルの設定（107 ページ） を参照してください。
ステップ 12	（任意）[**no**] **ptp announce timeout** *count* **例：** `switch(config-if)# ptp announce timeout 3`	タイムアウトがインターフェイスで発生する前の PTP 間隔の数を設定します。 PTP アナウンスのタイムアウト間隔の範囲は 2 ～ 4 です。

ステップ 13	（任意）[**no**] **ptp profile-override**
	例：
	`switch(config-if)# ptp profile-override`
	`switch(config-if)#`
		デフォルトで [無効（Disabled）] になっており、有効にすると、このインターフェイス設定で次のコマンドを変更できます。
		• **ptp transport**
		• **ptp announce interval**
		• **ptp delay-request minimum interval**
		• **ptp sync interval**
		• **ptp cost**（8275-2 プロファイルのみ）
		（注）
		有効にすると、グローバル PTP プロファイルが変更されても、コマンドへの変更はデフォルトにリセットされません。**ptp profile-override** を削除すると、インターフェイスの PTP 設定がグローバル プロファイルに対応するデフォルト値にリセットされます。

PTP プロファイルのデフォルト

次の表に、global コマンド **ptp profile** の設定時に自動的に設定されるコマンドの範囲とデフォルト値を示します。影響を受けるグローバル コマンドの範囲を、設定されたプロファイルで許可されている範囲を超えて変更することはできません。ただし、インターフェイス モードでは、**ptp profile-override**コマンドが設定されている場合は変更できます。

（注） Cisco NX-OS リリース 9.3(5) では、Cisco Nexus 93180YC-FX3S スイッチのみがこのコマンドのいずれかのオプションをサポートします。

デフォルトプロファイル（デフォルト PTP）の構成は、1 秒あたり 4 パケット（pps）の間隔で動作します。しかし Cisco Nexus 9408、9808、9804と N9K-C9332D-H2R プラットフォーム スイッチ上では、メディア プロファイル、最小同期と遅延間隔とデフォルトでは、8 pps サポートされます。時刻送信元と同期する場合、同期および遅延要求間隔でサポートされる最小レートは 8 pps です。

表 *6:* 範囲とデフォルト値

パラメータ	範囲またはコンフィギュレーション モード	デフォルトプロファイルでサポートされる値の範囲	デフォルト プロファイルのデフォルト値	8275-1 プロファイルでサポートされる値の範囲	8275-1 プロファイルのデフォルト値	8275-2 プロファイルでサポートされる値の範囲	8275-2 プロファイルのデフォルト値	インターフェイスで設定された「ptp profile override」の値の範囲（デフォルトは設定されたプロファイルに基づく）
モード	グローバル	none	none	ハイブリッド	ハイブリッド	なし	なし	変更なし
domain	グローバル	0 ～ 63	0	24 ～ 43	24	44 ～ 63	44	変更なし
priority1	グローバル	0 ～ 255	255	128	128	128	128	変更なし
priority2	グローバル	0 ～ 255	255	0 ～ 255	128	0 ～ 255	128	変更なし

パラメータ	範囲またはコンフィギュレーション モード	デフォルトプロファイルでサポートされる値の範囲	デフォルト プロファイルのデフォルト値	8275-1 プロファイルでサポートされる値の範囲	8275-1 プロファイルのデフォルト値	8275-2 プロファイルでサポートされる値の範囲	8275-2 プロファイルのデフォルト値	インターフェイスで設定された「ptp profile override」の値の範囲（デフォルトは設定されたプロファイルに基づく）
コスト	インターフェイス	設定不能	設定不能	0 ～ 255	128	0 ～ 255	128	0 ～ 255
トランスポート	インターフェイス	ipv4、ipv6	ipv4	イーサネット	イーサネット	ipv4、ipv6	ipv4	ethernet、ipv4
transmission	インターフェイス	multicast、unicast	multicast	multicast	multicast	ユニキャスト	unicast	変更なし
役割	インターフェイス	dynamic、master、slave	ダイナミック	ダイナミック	ダイナミック	動的、マスター、スレーブ	ダイナミック	変更なし
アナウンス間隔	インターフェイス	0 ～ 4 0 ～ 4（aes67） -3ー1（smpte2059-2）	1	-3	-3	-3 ～ 0	-3	-3 ～ 4 0 ～ 4（aes67） -3ー1（smpte2059-2）
delay-request minimum interval	インターフェイス	-1 ～ 6 -4 ー 5（aes67） -4 ー 5（smpte2059-2）	0	-4	-4	-4 ～ 0	-4	-4 ～ 6 -4 ー 5（aes67） -4 ー 5（smpte2059-2）
同期間隔	インターフェイス	-3 ～ -1 -4 ー 1（aes67） -7 ー 0（smpte2059-2）	-2	-4	-4	-4 ～ 0	-4	-4 ー 1 -4 ー 1（aes67） -7 ー 0（smpte2059-2）

PTP 通知の設定

始める前に

次の重要な PTP イベントの通知を有効化、無効化、およびカスタマイズできます。

- グランド マスター（GM）クロックの変更
- 親クロックの変更
- ポートの PTP ステートの変更
- 高 PTP クロック修正

通知は、PTPから受信した情報に基づいてDMEインフラストラクチャによって生成されます。

手順の概要

1. [**no**] **ptp notification type gm-change**
2. [**no**] **ptp notification type parent-change**
3. [**no**] **ptp notification type port-state-change** [**category** { **all** | **master-slave-only** }] [**interval** { **immediate** | *seconds* [**periodic-notification** { **disable** | **enable** }] }]
4. [**no**] **ptp notification type high-correction** [**interval** { *seconds* [**periodic-notification** { **disable** | **enable** }] | **immediate** }]
5. （任意） [**no**] **ptp correction-range** { *nanoseconds* | **logging** }

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	[**no**] **ptp notification type gm-change**
	例：
	`switch(config)# ptp notification type gm-change`
	`switch(config)#`
		PTP グランド マスター クロックが変更された場合に、変更通知を送信するようにシステムを設定します。

ステップ 2	[**no**] **ptp notification type parent-change**
	例：
	`switch(config)# ptp notification type parent-change`
	`switch(config)#`
		PTP の親クロックが変更された場合に、変更通知を送信するようにシステムを設定します。

ステップ 3	[**no**] **ptp notification type port-state-change** [**category** { **all** | **master-slave-only** }] [**interval** { **immediate** | *seconds* [**periodic-notification** { **disable** | **enable** }] }] 例：	ポート ステート変更イベントが発生した場合に通知を送信するようにシステムを設定します。 • **category**：通知を送信するために必要な状態変更を指定します。

	コマンドまたはアクション	目的
	`switch(config)# ptp notification type port-state-change category master-slave-only` `switch(config)#`	• **all**：すべてのポート状態の変更が報告されます。 （注） **all** オプションを使用すると、多くの通知が表示されます。 • **master-slave-only**：マスター スレーブ状態との間のポート状態の変更のみが報告されます。 • **interval** *seconds*：ポート状態変更通知は、設定された間隔（1 ー 300 秒、粒度は 1 秒）で送信されます。 • **periodic-notification**：設定された間隔の間にポート ステートの変更が発生していない場合でも、定期的な通知を送信するかどうかを決定します。 **disable**：ポート状態変更通知は、現在の状態が以前に報告された状態と同じでない場合にのみ報告されます。設定された定期的な間隔中の中間状態の変更は無視されます。たとえば、ポートが時刻 X で MASTER であり、DISABLED に変更されてから X + periodic-interval が発生するまでに MASTER に戻る場合、その間のイベントは通知されません。 **enable**：ポート ステート変更通知は、ポート ステートの変更に関係なく、設定された間隔で送信されます。 • **interval immediate**：ポートの状態変化通知は、状態が変化すると送信されます。

ステップ 4	[**no**] **ptp notification type high-correction** [**interval** { *seconds* [**periodic-notification** { **disable** | **enable** }] | **immediate** }]
	例：
	`switch(config)# ptp notification type high-correction interval immediate`
	`switch(config)#`
		PTP 高補正イベントが発生した場合に高補正通知を送信するようにシステムを設定します。高修正イベントは、修正が **ptp correction-range** コマンドで設定された値を超えた場合です（次のオプションの手順を参照）。
		• **interval** *seconds*：設定された間隔（1 ー 300 秒、精度 1 秒）で高修正通知が送信されます。

	コマンドまたはアクション	目的
		• **periodic-notification**：設定された間隔中に高度な修正が行われなかった場合でも、定期的な通知を送信するかどうかを決定します。 **disable**：設定された定期的な間隔の間に高補正イベントが発生した場合にのみ通知を送信します。これがデフォルトの設定です。 **enable**：設定された定期的な間隔の間に高修正イベントの数に関係なく通知を送信します。そのようなイベントがない場合、ペイロードは定期的な間隔の間にゼロ修正イベントを示します。 • **interval immediate**：高度な修正イベントが発生するとすぐに通知を送信します。 高修正通知には、次の属性が含まれます。 • highCorrectionCount • lastHighCorrectionTime • lastHighCorrectionValue

ステップ 5	（任意）[**no**] **ptp correction-range** { *nanoseconds* | **logging** }
	例：
	`switch(config)# ptp correction-range 200000`
	`switch(config)#`
		超過すると、PTP 高補正が発生したことを示すしきい値を設定します。範囲は 10 ー 1000000000 です。デフォルト値は 100（マイクロ秒の 10 倍）です。

PTP 混合モード

PTP は、接続されたクライアントから受信した **delay_req** メッセージのタイプに基づいて、Cisco Nexus デバイスによって自動的に検出される PTP メッセージを配信するための混合モードをサポートします。このモードでは、スレーブがユニキャスト メッセージで **delay_req** を送信すると、マスターもユニキャスト **delay_resp** メッセージで応答します。

PTP インターフェイスがマスター ステートを維持する設定

この手順では、エンドポイントによってポートがスレーブ ステートに移行するのを防ぐ方法について説明します。

始める前に

- スイッチ上でグローバルに PTP をイネーブルにし、PTP 通信の送信元 IP アドレスを設定したことを確認します。
- PTP をグローバルにイネーブルにしても、デフォルトで、サポートされているすべてのインターフェイス上でイネーブルになりません。PTP インターフェイスは個別にイネーブルに設定する必要があります。

手順の概要

1. switch # **configure terminal**
2. switch(config) # **interface ethernet** *slot/port*
3. switch(config-if) # **ptp**
4. switch(config-if) # **ptp multicast master-only**
5. **ptp role master**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	switch # **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	switch(config) # **interface ethernet** *slot/port*	PTP をイネーブルにするインターフェイスを指定し、インターフェイス構成モードを開始します。
ステップ 3	switch(config-if) # **ptp**	インターフェイスで PTP をイネーブルまたはディセーブルにします。 （注） このコマンドを設定した後、Cisco NX-OS リリース 9.3(5) 以降の場合は、ステップ 5 に進みます。Cisco NX-OS リリース 9.3(4) 以前の場合は、ステップ 4 に進みます。
ステップ 4	switch(config-if) # **ptp multicast master-only**	マスター ステートを維持するようにポートを設定します。 （注） このコマンドは、Cisco NX-OS リリース 9.3(4) 以前でサポートされています。Cisco NX-OS リリース 9.3(5) 以降では廃止されています。 Cisco NX-OS リリース9.3 (4) 以前の場合は、これで手順は終了です。

	コマンドまたはアクション	目的
ステップ 5	**ptp role master**	マスターステートを維持するようにポートを設定します。 （注） このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

例

この例では、インターフェイス上に PTP を設定し、インターフェイスがマスター ステートを維持するように設定する方法を示しています。

```
switch(config)# show ptp brief

PTP port status
-----------------------------------
Port                   State
----------------  ----------------
Eth1/1               Slave
switch(config)# interface ethernet 1/1
switch(config-if)# ptp multicast master-only
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_GM_CHANGE: Grandmaster clock has changed
 from 60:73:5c:ff:fe:62:a1:41 to 58:97:bd:ff:fe:0d:54:01 for the PTP protocol
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_STATE_CHANGE: Interface Eth1/1 change from
 PTP_BMC_STATE_SLAVE to PTP_BMC_STATE_PRE_MASTER
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_TIMESYNC_LOST: Lost sync with  master clock
2001 Jan  7 07:50:07 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_STATE_CHANGE: Interface Eth1/1 change from
 PTP_BMC_STATE_PRE_MASTER to PTP_BMC_STATE_MASTER
```

PTP ユニキャスト ネゴシエーションの有効化

PTPユニキャスト送信を有効にすることは、ユニキャストネゴシエーションを使用するための前提条件です。

Cisco NX-OS 10.2(1)F リリース以降、新しく追加された CLI は次のとおりです。

手順の概要

1. **switch (config-ptp-ucast-negotiation)# schema** <*schema-name*>
2. （任意） **switch (config-ptp-ucast-nego-schema)# announce interval** <*log-seconds*>
3. （任意） **switch (config-ptp-ucast-nego-schema)# sync interval** <*log-seconds*>
4. **switch (config-ptp-ucast-nego-schema)# delay-response interval** <*log-seconds*>
5. **switch (config-ptp-ucast-nego-schema)# announce duration** <*seconds*> [**renew-offset** <*seconds*>]
6. **switch (config-ptp-ucast-nego-schema)# sync duration** <*seconds*> [**renew-offset** <*seconds*>]
7. **switch (config-ptp-ucast-nego-schema)# delay response duration** <*seconds*> [**renew-offset** <*seconds*>]

8. **switch (config-ptp-ucast-nego-schema)# announce interval range** <*minimum-log-val*> <*maximum-log-val*>
9. **switch (config-ptp-ucast-nego-schema)# sync interval range** <*minimum-log-val*> <*maximum-log-val*>
10. **switch (config-ptp-ucast-nego-schema)# delay-response interval range** <*minimum-log-val*> <*maximum-log-val*>
11. **switch (config-ptp-ucast-nego-schema)# announce duration range** <*minimum-seconds*> <*maximum-seconds*>
12. **switch (config-ptp-ucast-nego-schema)# sync duration range** <*minimum-seconds*> <*maximum-seconds*>
13. **switch (config-ptp-ucast-nego-schema)# delay-response duration range** <*minimum-seconds*> <*maximum-seconds*>
14. **show ptp unicast-negotiation** [*interface ethernet slot/port*]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**switch (config-ptp-ucast-negotiation)# schema** <*schema-name*>	デフォルト スキーマは、ユニキャスト ネゴシエーションが有効になっているときに作成され、PTPユニキャストが有効になっているすべてのインターフェイスと、現在設定されているマスター IP に適用されます。 スキーマ名は最大で 31 文字にできます。
ステップ 2	（任意） **switch (config-ptp-ucast-nego-schema)# announce interval** <*log-seconds*>	PTP アナウンス メッセージの間隔を設定します。 範囲は- 3ー0 です。 デフォルト値は 1 です。
ステップ 3	（任意） **switch (config-ptp-ucast-nego-schema)# sync interval** <*log-seconds*>	PTP 同期メッセージの間隔を構成します。 範囲は -4 ー 0です。 デフォルト値は 3 です。
ステップ 4	**switch (config-ptp-ucast-nego-schema)# delay-response interval** <*log-seconds*>	ポートがマスター状態のとき、PTP遅延メッセージ間で許可されている間隔を設定します。 範囲は -4 ー 0 です。 デフォルト値は-2です。
ステップ 5	**switch (config-ptp-ucast-nego-schema)# announce duration** <*seconds*> [**renew-offset** <*seconds*>]	（任意）アナウンス セッションの期間を設定します。 **renew-offset**<*seconds*>:

	コマンドまたはアクション	目的
		これは、スレーブがセッションの更新要求を送信する時間を設定するために使用できます。デフォルト値は 10 です。つまり、セッションの有効期限の 10 秒前に更新要求を送信します（許可期間）。 指定できる範囲は 60 ～ 1000 です。 デフォルト値は 300 です。
ステップ 6	**switch (config-ptp-ucast-nego-schema)# sync duration** <*seconds*> [**renew-offset** <*seconds*>]	（任意）同期セッションの期間を設定します。 **renew-offset**<*seconds*>: これは、スレーブがセッションの更新要求を送信する時間を設定するために使用できます。デフォルト値は 10 です。つまり、セッションの有効期限の 10 秒前に更新要求を送信します（許可期間）。 指定できる範囲は 60 ～ 1000 です。 デフォルト値は 300 です。
ステップ 7	**switch (config-ptp-ucast-nego-schema)# delay response duration** <*seconds*> [**renew-offset** <*seconds*>]	（任意）遅延応答セッションの期間を設定します。 **renew-offset**<*seconds*>: これは、スレーブがセッションの更新要求を送信する時間を設定するために使用できます。デフォルト値は 10 です。つまり、セッションの有効期限の 10 秒前に更新要求を送信します（許可期間）。 指定できる範囲は 60 ～ 1000 です。 デフォルト値は 300 です。
ステップ 8	**switch (config-ptp-ucast-nego-schema)# announce interval range** <*minimum-log-val*> <*maximum-log-val*>	（任意）スレーブからのアナウンス間隔要求の値の許容範囲を設定します。 minimum-log-val のデフォルトは -3 です。 maximum-log-val のデフォルトは 0 です。
ステップ 9	**switch (config-ptp-ucast-nego-schema)# sync interval range** <*minimum-log-val*> <*maximum-log-val*>	（任意）スレーブからの同期間隔要求の許容範囲を設定します。 minimum-log-val のデフォルトは -4 です。 maximum-log-val のデフォルトは 0 です。
ステップ 10	**switch (config-ptp-ucast-nego-schema)# delay-response interval range** <*minimum-log-val*> <*maximum-log-val*>	（任意）スレーブからの遅延応答間隔要求の許容範囲を設定します。 minimum-log-val のデフォルトは -4 です。 maximum-log-val のデフォルトは 0 です。

	コマンドまたはアクション	目的
ステップ 11	**switch (config-ptp-ucast-nego-schema)# announce duration range** <*minimum-seconds*> <*maximum-seconds*>	（任意）スレーブからのセッション継続時間要求の値の許容範囲を設定します。 minimum-seconds のデフォルトは 60 です。 maximum-seconds のデフォルトは 1000 です。
ステップ 12	**switch (config-ptp-ucast-nego-schema)# sync duration range** <*minimum-seconds*> <*maximum-seconds*>	（任意）スレーブからの同期セッション期間要求の値の許容範囲を設定します。 minimum-seconds のデフォルトは 60 です。 maximum-seconds のデフォルトは 1000 です。
ステップ 13	**switch (config-ptp-ucast-nego-schema)# delay-response duration range** <*minimum-seconds*> <*maximum-seconds*>	（任意）スレーブからの遅延応答セッション期間要求の値の許容範囲を設定します。 minimum-seconds のデフォルトは 60 です。 maximum-seconds のデフォルトは 1000 です。
ステップ 14	**show ptp unicast-negotiation** [*interface ethernet slot/port*]	ユニキャスト ネゴシエーションのステータスを表示します。

拡張マルチキャスト スケール

この機能は、デバッグ機能が非常に制限されている場合でも、PTPマルチキャストセカンダリデバイスのより高いスケーリングが必要な特定の展開シナリオでのみ使用されます。

この機能には、次の制限があります。

- PTPスレーブの数が多いということは、PTP制御パケットレートが非常に高いことを意味します。その結果、銅率を適切に上げる必要があります。コントロール プレーン ポリシングの構成の詳細については、cisco.com の *Cisco Nexus 9000* シリーズ *NX-OS* セキュリティ構成ガイドの適切なバージョンを参照してください。
- PTP デバッグは、PTP 内部 syslog などのさまざまな内部ログとともに、**no ptp debug all** コマンドを使用して完全に無効にする必要があります。その結果、問題をデバッグする機能が非常に制限されます。
- PTP セカンダリ ポートは、スケーリングされた PTP プライマリ ポートのいずれともハードウェア MAC（ポート fifo）を共有しないようにすることをお勧めします。さらに、ハードウェア MAC ごとに 2 つ以下のプライマリ ポートを有効にする必要があります。特定のスイッチのポートのハードウェア MAC は、次のコマンドを使用して確認できます。

  show interface hardware-mappings
- まれに、修正がミリ秒の範囲に急上昇することがあります。

次のコマンドを実行して、PTP マルチキャスト セカンダリ デバイスのスケーリングを有効にします。

ptp enhanced-client-scale

上記のコマンドのステータスを表示するには、次のコマンドを実行します。

switch# **show run ptp | grep enhanced**

タイムスタンプ タギング

タイムスタンプ タギング機能は、リモート デバイスでパケットが到達したときに正確な時間情報を提供し、実際の時間を追跡できるようにします。パケットは、PTP を使用してナノ秒の精度で切り捨てられ、タイムスタンプが付けられます。Cisco Nexus Data Broker とともにスイッチの TAP 集約機能を使用すると、SPAN を使用してネットワーク トラフィックをコピーし、トラフィックをフィルタリングしてタイムスタンプを付け、記録および分析のために送信できます。

タイムスタンプ タギングの設定

（注） 9636C-R、9636C-RX、および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 スイッチでは、タイムスタンプ タギングの設定はサポートされていません。

（注）
- VXLAN EVPN マルチサイト展開で ttag 機能を使用する場合は、クラウドに接続する BGW の DCI インターフェイスで ttag が削除されていることを確認します（**ttag-strip**）。詳細に説明すると、ttagが、ether-type 0x8905をサポートしないNexus 9000以外のデバイスに接続されている場合、ttagの除去が必要です。
- DCI の BGW バックツーバック モデルでは ttag の削除は必要ありません。
- Cisco Nexus 9800 スイッチは、ether-type 0x8905 パケットのルーティングをサポートしていません。

始める前に

PTP オフロードがグローバルに有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. **interface** *type slot/port*
3. [**no**] **ttag**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 2/2`
	`switch(config-if)#`
		指定したインターフェイスに対してインターフェイス コンフィギュレーション モードを開始します。

ステップ 3	[**no**] **ttag** 例： `switch(config-if)# ttag`	レイヤ 2 またはレイヤ 3 出力インターフェイスでタイムスタンプ タギングを設定します。

TTAG マーカー パケットと時間間隔の設定

ttag タイムスタンプ フィールドは、マーカー パケットに 48 ビットのタイムスタンプを付加します。この 48 ビットのタイムスタンプは、人間の読み取りやすい ASCII ベースのタイムスタンプではありません。この48ビットのタイムスタンプを人間が読み取れるようにするために、ttagマーカーパケットを使用して、48ビットのタイムスタンプ情報をデコードするための追加情報を提供できます。

フィールド	位置（バイト：ビット）	長さ	定義
Magic		16	デフォルトでは、このフィールドには A6A6 と表示されます。これにより、パケットストリーム上の ttag-marker パケットを識別できます。
バージョン		8	バージョン番号。デフォルトのバージョンは 1 です。
精度		16	このフィールドは、48 ビットのタイムスタンプサイズの粒度を表します。デフォルトの値は04で、これは 100 ピコ秒つまり 0.1 ナノ秒を表します。

UTc_offset		8	ASIC と UTC クロック間の utc_offset 値です。デフォルト値は 0 です。
Timestamp_hi		32	48 ビットの ASIC ハードウェア タイムスタンプの上位 16 ビットです。 （注） 64 ビットの ASIC ハードウェア タイムスタンプを取得するために、[Timestamp_hi] および [Timestamp_lo] フィールドに [Correction_hi] および [Correction_lo] を追加します。
Timestamp_lo		32	48 ビットの ASIC ハードウェア タイムスタンプの下位 32 ビットです。 （注） 64 ビットの ASIC ハードウェア タイムスタンプを取得するために、[Timestamp_hi] および [Timestamp_lo] フィールドに [Correction_hi] および [Correction_lo] を追加します。
UTC sec		32	Cisco Nexus 9000 シリーズ スイッチの CPU クロックに基づく UTC タイムスタンプの秒の部分です。
UTC sec		32	Cisco Nexus 9000シリーズスイッチのCPUクロックに基づく UTC タイムスタンプのナノ秒の部分です。
予約済み		32	将来的な使用のために予約されています。

Correction_hi		32	Cisco Nexus 9000 シリーズ スイッチの累積 PTP 補正の上位 32 ビット。 （注） 64 ビットの ASIC ハードウェア タイムスタンプを取得するために、[Timestamp_hi] および [Timestamp_lo] フィールドに [Correction_hi] および [Correction_lo] を追加します。
Correction_lo		32	Cisco Nexus 9000 シリーズ スイッチの累積 PTP 補正の下位 32 ビット。 （注） 64 ビットの ASIC ハードウェア タイムスタンプを取得するために、[Timestamp_hi] および [Timestamp_lo] フィールドに [Correction_hi] および [Correction_lo] を追加します。
署名（Signature）		32	デフォルト値は 0xA5A5A5A5 です。これにより、マーカーパケットの前方検索が可能になり、UTC タイムスタンプへの参照が提供されるため、クライアント ソフトウェアはその参照 UTC を使用して、各パケット ヘッダーの 32 ビットのハードウェア タイムスタンプを回復できます。
パッド		8 64	これは、ttag-marker の位置wo合わせを4バイト境界に変換するための位置合わせバイトです。

始める前に

PTP オフロードがグローバルにイネーブル化されていることを確認します。

手順の概要

1. **configure terminal**
2. **ttag-marker-interval** *seconds*

3. **interface** *type slot/port*
4. [**no**] **ttag-marker enable**
5. **ttag-strip**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**ttag-marker-interval** *seconds* 例： `switch(config-if)# ttag-marker-interval 90`	スイッチが ttag-marker パケットを発信ポートに送信するまでの秒数を設定します。これはスイッチのグローバル設定です。デフォルトでは、ttag-marker パケットを 60 秒ごとに送信します。seconds の範囲は 1 ～ 25200 です。

ステップ 3	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 2/2`
	`switch(config-if)#`
		指定したインターフェイスに対してインターフェイス コンフィギュレーション モードを開始します。

ステップ 4	[**no**] **ttag-marker enable** 例： `switch(config-if)# ttag-marker enable`	ttag-marker パケットを発信ポートに送信します。
ステップ 5	**ttag-strip** 例： `switch(config-if)# ttag-strip`	インターフェイスの出力パケットから TTAG を削除します。

PTP 設定の確認

次のいずれかのコマンドを使用して、設定を確認します。

表 *7: PTP Show* コマンド

コマンド	目的
show ptp brief	PTP のステータスを表示します。
show ptp clock	ローカル クロックのプロパティ（クロック ID など）を表示します。

コマンド	目的
show ptp clock foreign-masters-record	PTP プロセスが認識している外部マスターの状態を表示します。外部マスターごとに、出力に、クロック ID、基本的なクロック プロパティ、およびクロックがグランドマスターとして使用されているかどうかが表示されます。
show ptp corrections	最後の数個の PTP 修正を表示します。
show ptp counters [**all** | **interface ethernet** *slot/port*]	すべてのインターフェイスまたは指定したインターフェイスの PTP パケットカウンタを表示します。
show ptp parent	PTP の親のプロパティを表示します。
show ptp port interface ethernet *slot/port*	スイッチの PTP ポートのステータスを表示します。
show ptp time-property	PTP クロック プロパティを表示します。
show running-config ptp [**all**]	PTP の実行コンフィギュレーションを表示します。
clear ptp counters [**all** | **interface ethernet** *slot/port*]	特定のインターフェイスまたは PTP が有効になっているすべてのインターフェイスで送受信されるすべての PTP メッセージをクリアします。
show ptp detail	各 PTP ポートに接続されているすべてのピアのリストが表示され、ロールが静的か動的かが示されます。

PTP テレコム プロファイル設定の確認

PTPテレコムプロファイルの設定タスクを実行した後、ここでの説明に基づいて、設定を確認してください。

show running-config ptp all

このコマンドの出力には、PTPテレコムプロファイルのグローバル設定とインターフェイス設定が表示されます。

show running-config ptp all コマンドの出力例を次に示します。

```
switch# show running-config ptp all
!Command: show running-config ptp all
!Running configuration last done at: Fri Feb 21 20:09:55 2020
!Time: Fri Feb 21 21:10:19 2020
```

```
version 9.3(5) Bios:version 01.00
feature ptp

ptp profile 8275-1
  mode hybrid
ptp source 0.0.0.0
ptp device-type boundary-clock
ptp priority1 128
ptp priority2 10
ptp pdelay-req-interval 0
no ptp notification type parent-change
no ptp notification type gm-change
no ptp notification type high-correction
no ptp notification type port-state-change
ptp correction-range 100000
no ptp correction-range logging
ptp management
ptp mean-path-delay 1000000000
ptp domain 24
ttag-marker-interval 60


interface Ethernet1/1
  ptp
  no ptp profile-override
  ptp destination-mac non-forwardable rx-no-match accept
  ptp transport ethernet
  ptp transmission multicast
  ptp role dynamic
  ptp cost 128
  ptp delay-request minimum interval -4
  ptp announce interval -3
  ptp sync interval -4
  ptp announce timeout 3

interface Ethernet1/6
  ptp
  no ptp profile-override
  ptp destination-mac non-forwardable rx-no-match accept
  ptp transport ethernet
  ptp transmission multicast
  ptp role dynamic
  ptp cost 128
  ptp delay-request minimum interval -4
  ptp announce interval -3
  ptp sync interval -4
  ptp announce timeout 3

interface Ethernet1/7
  ptp
  no ptp profile-override
  ptp destination-mac non-forwardable rx-no-match accept
  ptp transport ethernet
  ptp transmission multicast
  ptp role dynamic
  ptp cost 128
  ptp delay-request minimum interval -4
  ptp announce interval -3
  ptp sync interval -4
  ptp announce timeout 3

interface Ethernet1/8
  ptp
  no ptp profile-override
```

```
ptp destination-mac non-forwardable rx-no-match accept
ptp transport ethernet
ptp transmission multicast
ptp role dynamic
ptp cost 128
ptp delay-request minimum interval -4
ptp announce interval -3
ptp sync interval -4
ptp announce timeout 3
```

（注） **show running-config ptp all** コマンドの出力には、すべての PTP 設定済みインターフェイスの完全なリストが表示されます。

show ptp parent

このコマンドの出力には、PTP の親プロパティが表示されます。

show ptp parent コマンドの出力例を次に示します。

```
switch# show ptp parent
PTP PARENT PROPERTIES

Parent Clock:
Parent Clock Identity: 10:b3:d6:ff:fe:bf:a8:63
Parent Port Number: 0
Observed Parent Offset (log variance): N/A
Observed Parent Clock Phase Change Rate: N/A

Grandmaster Clock:
Grandmaster Clock Identity: 10:b3:d6:ff:fe:bf:a8:63
Grandmaster Clock Quality:
        Class: 248
        Accuracy: 254
        Offset (log variance): 65535
        Priority1: 128
        Priority2: 10
```

show ptp corrections

このコマンドの出力には、各 PTP スレーブ ポートの直近 2000 件までの修正の詳細が表示されます。

show ptp corrections コマンドの出力例を次に示します。

```
switch# show ptp corrections
PTP past corrections
--------------------------------------------------------------------------------
Slave Port            SUP Time                 Correction(ns)    MeanPath Delay(ns)
----------  ------------------------------  ------------------  ------------------
Eth1/3      Thu Feb 20 22:51:02 2020 861523           4           260
Eth1/3      Thu Feb 20 22:51:02 2020 735961           4           260
Eth1/3      Thu Feb 20 22:51:02 2020 610170           4           268
Eth1/3      Thu Feb 20 22:51:02 2020 483106           0           280
Eth1/3      Thu Feb 20 22:51:02 2020 355745           0           280
Eth1/3      Thu Feb 20 22:51:02 2020 229924          -4           268
Eth1/3      Thu Feb 20 22:51:02 2020 104819          -4           268
Eth1/3      Thu Feb 20 22:51:01 2020 979604           8           272
```

show ptp clock

このコマンドの出力には、ローカル クロックのプロパティ（クロック ID など）が表示されます。

show ptp clock コマンドの出力例を次に示します。

```
switch# show ptp clock
PTP Device Type : boundary-clock
PTP Device Encapsulation : NA
PTP Source IP Address : 0.0.0.0
Clock Identity : 10:b3:d6:ff:fe:bf:a8:63
Clock Domain: 24
Slave Clock Operation : Unknown
Master Clock Operation : Two-step
Slave-Only Clock Mode : Disabled
Number of PTP ports: 35
Priority1 : 128
Priority2 : 10
Clock Quality:
        Class : 248
        Accuracy : 254
        Offset (log variance) : 65535
Offset From Master : 0
Mean Path Delay : 0
Steps removed : 0
Correction range : 100000
MPD range : 1000000000
Local clock time : Wed Feb 26 17:08:34 2020
Hardware frequency correction : NA
PTP Clock state               : Free-Run
```

show ptp brief

このコマンドの出力には、設定されたポートごとの PTP クロック状態が表示されます。

show ptp brief コマンドの出力例を次に示します。

```
switch# show ptp brief
PTP port status
-----------------------------------
Port                  State
--------------------- ------------
Eth1/1                Slave
Eth1/6                Disabled
Eth1/7                Disabled
Eth1/8                Disabled
Eth1/10               Master
Eth1/11               Disabled
Eth1/12               Disabled
Eth1/13               Master
Eth1/14               Disabled
Eth1/15               Disabled
Eth1/16               Disabled
Eth1/17               Disabled
Eth1/18               Disabled
Eth1/19               Disabled
Eth1/20               Disabled
Eth1/21               Disabled
Eth1/22               Disabled
Eth1/23               Disabled
Eth1/24               Disabled
Eth1/25               Disabled
```

```
Eth1/26               Disabled
Eth1/27               Disabled
Eth1/28               Disabled
Eth1/29               Disabled
Eth1/30               Disabled
Eth1/31               Disabled
Eth1/32               Disabled
Eth1/33               Disabled
Eth1/34               Disabled
Eth1/35               Disabled
Eth1/36               Disabled
Eth1/37               Disabled
Eth1/38               Disabled
Eth1/39               Disabled
Eth1/40               Disabled
```

show ptp clock foreign-masters record

このコマンドの出力には、PTPプロセスが認識している外部マスターの状態が表示されます。出力には、外部マスターごとにクロック ID、基本的なクロック プロパティ、およびクロックがグランドマスターとして使用されているかどうかが表示されます。

show ptp clock foreign-master-record コマンドの出力例を次に示します。

```
switch# show ptp port status
P1=Priority1, P2=Priority2, C=Class, A=Accuracy,
OSLV=Offset-Scaled-Log-Variance, SR=Steps-Removed
GM=Is grandmaster

---------   ----------------------  ---  ----  ----  ---  -----  --------
Interface         Clock-ID          P1   P2    C     A    OSLV   SR
---------   ----------------------  ---  ----  ----  ---  -----  --------

Eth1/1    00:00:00:00:00:00:00:01    128  128   6     33   65535  0    GM
```

show ptp detail

このコマンドの出力には、各PTPポートに接続されているすべてのピアのリストが表示され、ロールが静的か動的かが示されます。

```
switch(config-if)# show ptp detail
PTP port details:
    Flags:   (S) Static, (D) Dynamic
    Connected Peer details: IP  clock-id

Port             State         Transmission     Transport        Flags
====             =====         ============     =========        =====
Eth1/27/1        Slave         Unicast          IPV4             (S)
Connected Peers:
    10.10.1.36            00:fd:22:ff:fe:8d:1a:a5-0x1901

Eth1/27/2        Passive       Unicast          IPV4             (D)
Connected Peers:
    10.10.2.36            00:00:00:00:00:00:00:00-0x0

Eth1/27/3        Master        Multicast        IPV4             (D)
Connected Peers:
```

PTP の設定例

次に、デバイス上で PTP をグローバルに設定し、PTP 通信用の送信元 IP アドレスを指定し、クロックの優先レベルを設定する例を示します。

```
switch# configure terminal
switch(config)# feature ptp
switch(config)# ptp source 10.10.10.1
switch(config)# ptp priority1 1
switch(config)# ptp priority2 1
switch(config)# show ptp brief
PTP port status
-----------------------
Port State
------- --------------
switch(config)# show ptp clock
PTP Device Type: Boundary clock
Clock Identity : 0:22:55:ff:ff:79:a4:c1
Clock Domain: 0
Number of PTP ports: 0
Priority1 : 1
Priority2 : 1
Clock Quality:
  Class : 248
  Accuracy : 254
  Offset (log variance) : 65535
Offset From Master : 0
Mean Path Delay : 0
Steps removed : 0
Local clock time:Mon Dec 22 14:13:24 2014
```

次に、インターフェイス上で PTP を設定し、アナウンス、遅延要求、および同期メッセージの間隔を設定する例を示します。

```
switch# configure terminal
switch(config)# interface Ethernet 1/1
switch(config-if)# ptp
switch(config-if)# ptp announce interval 3
switch(config-if)# ptp announce timeout 2
switch(config-if)# ptp delay-request minimum interval smpte-2059-2 -3
switch(config-if)# ptp sync interval smpte-2059-2 -3
switch(config-if)# no shutdown
switch(config-if)# show ptp brief
PTP port status
-----------------------
Port State
------- --------------
Eth2/1 Master
switch(config-if)# show ptp port interface ethernet 2/1
PTP Port Dataset: Eth2/1
Port identity: clock identity: 0:22:55:ff:ff:79:a4:c1
Port identity: port number: 1028
PTP version: 2
Port state: Master
Delay request interval(log mean): 4
Announce receipt time out: 2
Peer mean path delay: 0
```

```
Announce interval(log mean): 3
Sync interval(log mean): 1
Delay Mechanism: End to End
Peer delay request interval(log mean): 0
```

個の例では、マスター/スレーブ ロールを設定し、対応するピア スレーブ/マスター IP アドレスを割り当てる方法を示します。

```
For Cisco NX-OS Release 9.3(5) and later:
switch# configure terminal
switch(config)# interface ethernet 1/1
switch(config-if)# ptp
switch(config-if)# ptp transmission unicast
switch(config-if)# ptp role master
switch(config-if)# ptp slave 10.1.1.2
switch(config-if)# ptp ucast-source 11.0.0.1
switch(config-if)# ip address 11.0.0.1/24
switch(config-if)# no shutdown

switch# configure terminal
switch(config)# interface ethernet 1/1
switch(config-if)# ptp
switch(config-if)# ptp transmission unicast
switch(config-if)# ptp role slave
switch(config-if)# ptp master 10.1.1.2
switch(config-if)# ptp ucast-source 11.0.0.1
switch(config-if)# ip address 11.0.0.1/24
switch(config-if)# no shutdown

For Cisco NX-OS Release 9.3(4) and earlier:
switch-1(config)# interface ethernet 1/1
switch-1(config-if)# ptp transport ipv4 ucast master
switch-1(config-if-ptp-master)# slave ipv4 1.2.3.1
switch-1(config-if-ptp-master)# slave ipv4 1.2.3.2
switch-1(config-if-ptp-master)# slave ipv4 1.2.3.3
switch-1(config-if-ptp-master)# slave ipv4 1.2.3.4
switch-1(config-if-ptp-master)#

switch-1(config-if)# ptp transport ipv4 ucast slave
switch-1(config-if-ptp-slave)# master ipv4 4.4.4.1
switch-1(config-if-ptp-slave)# master ipv4 4.4.4.2
switch-1(config-if-ptp-slave)# master ipv4 4.4.4.3


switch-1(config-if-ptp-slave)# ptp ucast-source 9.9.9.9


switch-1(config-if)# sh running-config ptp

!Command: show running-config ptp
!Time: Tue Feb  7 17:37:09 2017

version 7.0(3)I4(6)
feature ptp

ptp source 1.1.1.1


interface Ethernet1/1
  ptp transport ipv4 ucast master
    slave ipv4 1.2.3.1
    slave ipv4 1.2.3.2
```

```
    slave ipv4 1.2.3.3
    slave ipv4 1.2.3.4

interface Ethernet1/2
  ptp transport ipv4 ucast slave
    master ipv4 4.4.4.1
    master ipv4 4.4.4.2
    master ipv4 4.4.4.3
  ptp ucast-source 9.9.9.9

switch-1(config-if)#
```

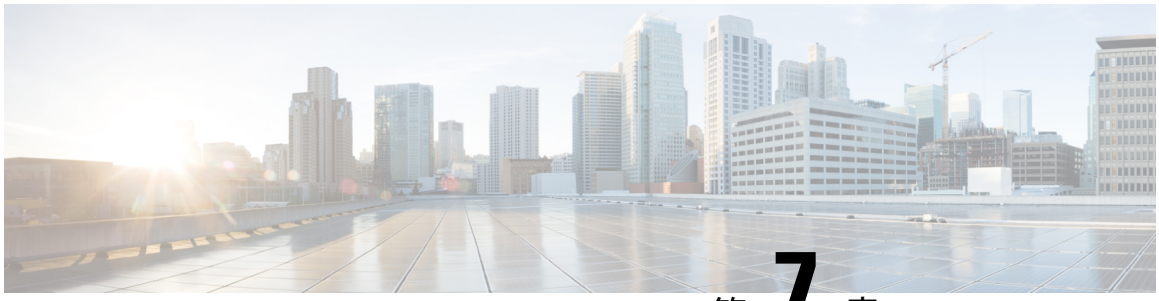
次に、マスター ポートまたはスレーブ ポートでクロック動作モードで PTP を設定する例を示します。

```
PLTFM-A(config)# show ptp clock
PTP Device Type : boundary-clock
PTP Device Encapsulation : layer-3
PTP Source IP Address : 1.1.1.1
Clock Identity : 74:26:ac:ff:fe:fd:de:ff
Clock Domain: 0
Slave Clock Operation : One-step
Master Clock Operation : One-step
Slave-Only Clock Mode : Disabled
Number of PTP ports: 142
Priority1 : 200
Priority2 : 200
Clock Quality:
        Class : 248
        Accuracy : 254
        Offset (log variance) : 65535
Offset From Master : -32
Mean Path Delay : 105
Steps removed : 1
Correction range : 200
MPD range : 100
Local clock time : Wed Jul  3 18:57:23 2019
Hardware frequency correction : NA
```

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
1588 IEEE	1588 IEEE 標準



第 7 章

GPS の設定

この章は、GPS 入力と構成方法を説明します。

この章は、次の項で構成されています。

- GPS について （141 ページ）
- GPS に関する注意事項と制限事項 （142 ページ）
- グランドマスター クロックの GPS の構成 （142 ページ）
- GPS 構成の検証 （143 ページ）

GPS について

ルータは、外部のクロックおよびタイミング ソースから 1 PPS、10 MHz、および ToD 信号を受信できます。3 つの入力は Sync-2 インターフェイスとして結合され、外部タイミング ソースまたは GPS 入力を形成します。

GPS 前面パネルのコネクタの詳細は次のとおりです。

- ToD：入力としての RS422 フォーマット
- 1PPS：入力としての RS422 または DIN コネクタ
- 10MHz：入力としての DIN コネクタ

GPS 入力が開始されるのは、3 つすべての信号（1PPS、10MHz、ToD）がアップの場合のみです。

（注） イーサネット インターフェイスとは異なり、Sync-2 インターフェイスは QL を送受信できません。Sync-2 インターフェイスに QL 値を割り当てていることを確認します。

デフォルトでは、1PPS および 10MHz は出力モードになっています。ToD の出力モードは設定できません。

GPS に関する注意事項と制限事項

GPS には、次の注意事項と制限事項があります。

- Cisco NX-OS リリース 10.3(2)以降、GPS 入力は Cisco Nexus 93180YC-FX3S スイッチでのみサポートされますが、次の制限事項があります。
  - ToD 出力はサポートされていません。
  - RS422 ToD 入力はサポートされていません。
    - RS422 TOD 入力のない GPS は、周波数と位相の同期にのみ使用できます。

グランドマスター クロックの GPS の構成

この手順を使用して、 GPS を入力として有効にします。

手順の概要

1. **configure terminal**
2. **clock-interface sync 1/1**
3. **gps-input tod-format** *value* **pps-input** *value*
4. **frequency synchronization**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**clock-interface sync 1/1**
	例：
	`switch(config)# clock-interface sync 1/1`
	`switch(config-clk-if)# port-parameters`
	`switch(config-clk-params)#`
		GPS クロック 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	**gps-input tod-format** *value* **pps-input** *value* 例： `switch(config-clk-params)# gps-input tod-format ntp4 pps-input ttl`	GPS を入力として有効にし、PPS 入力のソースと ToD 入力の形式を構成します。 サポートされている ToD 形式は次のとおりです。

	コマンドまたはアクション	目的
	`switch(config-clk-params)# exit` `switch(config-clk-if)#`	**着信 ToD 入力** / **書式** cisco / ASCII 形式 ntp4 / NTP Type4 形式 サポートされている PPS 入力形式は次のとおりです。 **着信 PPS 入力** / **書式** RS-422 / RS422 形式 TTL / TTL 形式

ステップ 4	**frequency synchronization**
	例：
	`switch(config-clk-if)# frequency synchronization`
	`switch(config-clk-freqsync)# selection input`
	`switch(config-clk-freqsync)# wait-to-restore 0`
	`switch(config-clk-freqsync)# exit`
	`switch(config-clock-if)#`
		GPS 上で周波数同期を構成します。

GPS 構成の検証

GPSの構成タスクが完了したら、このリファレンスを使用して構成エラーがないことをチェックして、構成を確認します。

show clock-interface

このコマンドの出力には、GPS クロックの詳細が表示されます。

show frequency synchronization 選択コマンドの出力例を次に示します。

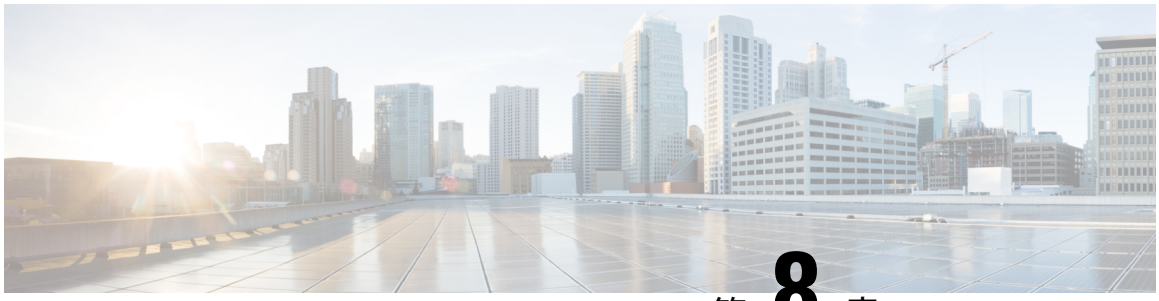
```
switch(config-clk-if)# sh frequency synchronization selection
==============
Selection point: System Clock (T0) Selector (2 inputs, 1 selected)
Last programmed 00:53:56 ago, and selection made 00:53:35 ago
Next selection points
Node scoped :
Uses frequency selection
Used for local line interface output
S Input Last Selection Point QL Pri Status
== ======================== ======================== ===== === ===========
Internal0[1] n/a SEC 255 Available
11 Sync1[1] n/a PRC 100 Locked
```

```
==============
Selection point: IEEE 1588 Clock Selector (2 inputs, 1 selected)
Last programmed 00:53:56 ago, and selection made 00:53:55 ago
Next selection points
Node scoped :
Uses frequency selection
S Input Last Selection Point QL Pri Status
== ======================== ======================== ===== === ===========
21 Internal0[1] n/a SEC 255 Holdover
Sync1[1] n/a PRC 100 Unmonitored
==============
switch(config-clk-if)#
```

フロント パネルのタイミング LED が緑色の場合は、GPS が構成されており、1PPS、ToD、および 10M の入力が有効であることを示します。

以下は、デバイスへの入力に基づく GPS タイミングのデフォルトの LED ステータスです。

TIMING	消灯	GPS 設定、および GPS ポートがダウンしています。一日内の時刻（ToD）、1PPS、および 10-MHz ポートがプロビジョニングされていないか、または無効です。
	オレンジ	ToD、1PPS、10-MHz 信号が無効です。
	緑	GPS ポートが稼働しています。ToD、1PPS、10-MHz 信号が有効です。



第 8 章

GNSS の構成

この章では、Cisco NX-OS デバイス上で Global Navigation Satellite System（GNSS）を構成する方法について説明します。

この章は、次の項で構成されています。

- GNSS について （145 ページ）
- GNSS の注意事項と制約事項 （146 ページ）
- GNSS レシーバーの有効化 （146 ページ）
- GNSS 構成の検証 （148 ページ）

GNSS について

GNSS 受信機は、GPS、Galileo、GLONASS、BeiDou および QZSS の L1 周波数 1551MHz ～ 1614MHz、標準位置サービス、および Coarse Acquisition コードで動作するように設計されています。受信機には、外部の GNSS アンテナに接続して、GNSS 衛星信号を自動的に取得し、最大 32 個の GNSS 衛星を追跡し、位置、速度、方位、時間を計算するために必要なすべての回路が含まれます。内部システムでの使用向けに、正確な 1 パルス/秒（PPS）と安定した 10 MHz 周波数出力を提供します。

受信機は、GNSS 衛星を捕捉すると、自動的に自己測量を開始します。測量が完了すると、受信機は「Over-Determined」タイミング モードに切り替わります。このモードでは、自己調査からの基準位置がメモリに保持され、受信機はクロック エラーとクロック バイアスのみを解決します。受信機は、位置と時刻の両方の受信機自律完全性監視機能（T-RAIM）を提供します。これにより、受信機は位置変更を自己決定したり、タイミング ソリューションに誤った情報を提供している衛星を削除したりすることができます。

（注） 信号は GNSS ポート（アンテナ ポート）から送信されません。

全地球航法衛星システム（GNSS）では、それぞれ特定の疑似ランダムノイズ（PRN）番号を持つさまざまな衛星コンステレーションをサポートしています。さまざまなGNSSシステムでサポートされる PRN 範囲は次のとおりです。

- GPS - PRN 1 ～ 32 のみをサポート
- GLONASS：PRN 65 ～ 96 のみをサポート
- BeiDou：PRN 201 ～ 237 のみをサポート
- Galileo：PRN 97 ～ 133 のみをサポート
- QZSS：PRN 193 ～ 195 のみをサポート
- SBAS：PRN 33 ～ 54 のみをサポート

GNSS の注意事項と制約事項

GNSS には、次の注意事項と制約事項があります。

- Cisco NX-OS リリース 10.3(2)F 以降、GNSS は Cisco Nexus 93180YC-FX3 スイッチでサポートされます。

GNSS レシーバーの有効化

この手順を使用して、GNSS レシーバーを入力として有効にします。

手順の概要

1. **configure terminal**
2. **feature frequency-synchronization**
3. **gnss-receiver sync 1/2**
4. **frequency synchronization**
5. （任意） **constellation** <*type*>
6. （任意） **cable-delay compensation**
7. （任意） **elevation threshold**
8. （任意） **snr threshold**
9. （任意） **pdop threshold**
10. （任意） **1pps polarity**
11. （任意） **anti-jam disable**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**feature frequency-synchronization**
	例：
	`switch# feature frequency-synchronization`
	`switch(config)#`
		機能の周波数同期を有効にします。

ステップ 3	**gnss-receiver sync 1/2**
	例：
	`switch(config)# gnss-receiver sync 1/2`
	`switch(config-gnss-if)# no shutdown`
	`switch(config-gnss-if)#`
		GNSS レシーバーを入力として有効にします。

ステップ 4	**frequency synchronization**
	例：
	`switch(config-gnss-if)# frequency synchronization`
	`switch(config-gnss-freqsync)# selection input`
	`switch(config-gnss-freqsync)# wait-to-restore 0`
	`switch(config-gnss-freqsync)# exit`
		GNSS レシーバーの周波数同期を設定します。

ステップ 5	（任意） **constellation** <*type*>
	例：
	`switch(config-gnss)# constellation gps`
	`switch(onfig-gnss)#`
		GNSSモジュールを構成して、任意の衛星を自動的に追跡したり、次の表に示す特定のコンステレーションを明示的に使用するように構成したりできます。デフォルトのコンステレーション構成は**AUTO**です：
		• 自動
		• GPS
		• GALILEO
		• BEIDOU
		• QZSS
		• GLONASS

ステップ 6	（任意） **cable-delay compensation**
	例：
	`switch(config-gnss)# cable-delay compensation 0`
	`switch(onfig-gnss)#`
		アンテナ ケーブルによって生じる遅延を設定します。長いケーブル配線では、この遅延が大きくなる可能性があります。範囲は –1000000、+1000000 ナノ秒です。

	コマンドまたはアクション	目的
		（注） ケーブルの遅延は、ケーブル 1 メートルあたり約 5.9 ナノ秒です。

ステップ 7	（任意） **elevation threshold**
	例：
	`switch(config-gnss)# elevation threshold 10`
	`switch(onfig-gnss)#`
		修正の最低衛星高度を構成します。範囲は 0～90° です。デフォルト値は 10° です。

ステップ 8	（任意） **snr threshold**
	例：
	`switch(config-gnss)# snr threshold 10`
	`switch(onfig-gnss)#`
		キャリア対ノイズ比 (C/No) を設定します。範囲は 0～15db-Hz です。デフォルト値は 0.0 C/No です。
		（注）
		Over-Determined Clock にのみ適用されます。自己調査中は適用されません。

ステップ 9	（任意） **pdop threshold**
	例：
	`switch(config-gnss)# pdop threshold 6`
	`switch(onfig-gnss)#`
		精度の位置希釈 (PDOP) は、位置決定の信頼レベルを示します。低い DOP 値は高い信頼性レベルを示し、高い DOP 値は低い信頼性レベルを示します。範囲は 0 ～ 10 です。デフォルト値は、6 です。

ステップ 10	（任意） **1pps polarity**
	例：
	`switch(config-gnss)# 1pps polarity positive`
	`switch(onfig-gnss)#`
		GNSS レスーバーの極性を設定します。プラスにもマイナスにも設定できます。デフォルトのオプションはプラスです。

ステップ 11	（任意） **anti-jam disable**
	例：
	`switch(config-gnss)# anti-jam disable`
	`switch(onfig-gnss)#`
		GNSS 信号に干渉する、またはアンテナ LNA またはレシーバーのフロント エンドを飽和させる外部 RF 送信元によってレシーバー機能が妨害されたときに発生する妨害を有効または無効にします。値：[Enable] または [Disable]
		（注）
		有効にすると、Over-Determined Clock モードでの修正に最低 2 つのサテライトが必要です。

GNSS 構成の検証

GNSS の構成タスクが完了したら、このリファレンスを使用して構成エラーがないことを確認して、設定を確認します。

gnss レシーバーを表示

このコマンドの出力には、GNSS 構成の詳細が表示されます。

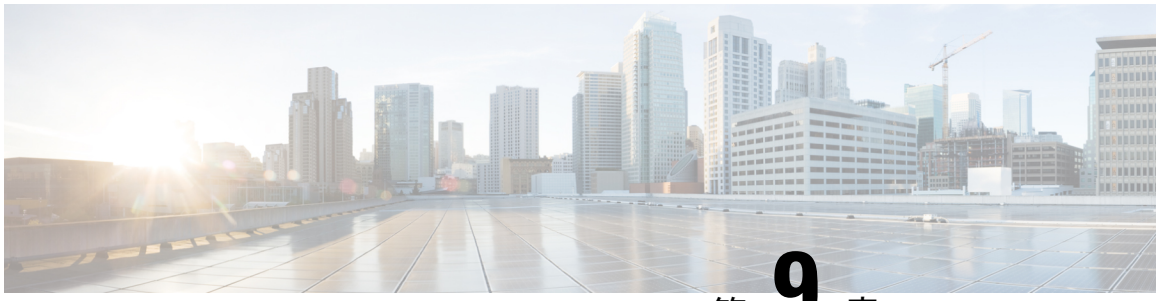
show gnss-receiver コマンドの出力例を次に示します。

```
switch(config-gnss-if)# sh gnss-receiver
GNSS-receiver SYNC 01/02
Status: Available, Up
Position: 41:11:660 N 74:0:0 W -12.805 m
Time: 1648046422 (UTC offset: 18s)
Firmware version: 1.7
Lock Status: Phase Locked, Receiver Mode: 3D-fix
Survey Progress: 100, Holdover Duration: 0
Major Alarm: Not used
Minor Alarm: Antenna shorted
Anti-jam: Disabled, Cable-delay compensation: 1000
1PPS polarity: Negative
PDOP: 0.000, HDOP: 0.000, VDOP: 0.000, TDOP: 0.001
Constellation: Auto, Satellite Count: 7
Satellite Thresholds:
SNR - 0 dB-Hz, Elevation - 0 degrees, PDOP - 5, TRAIM - 1 us
Satellite Info:
PRN   Channel Acquisition Ephemeris  SV     Signal
No.   No.     Flag        Flag       Type   Strength   Elevat'n  Azimuth
--- ------- ----------- --------- ----------- -------- -------- --------
11     0      On          On         GPS     0.036     0.076      0.024
28     1      On          On         GPS     0.036     0.025      0.272
1      2      On          On         GPS     0.037     0.089      0.002
19     3      On          On         GPS     0.037     0.036      0.151
14     5      On          On         GPS     0.036     0.019      0.045
17     6      On          On         GPS     0.037     0.025      0.314
23     7      On          On         GPS     0.037     0.014      0.178
switch(config-gnss-if)#
```

周波数同期の選択を表示

show frequency synchronization selection コマンドの出力例を次に示します。

```
switch(config-gnss-if)# sh frequency synchronization selection
==============
Selection point: System Clock (T0) Selector (2 inputs, 1 selected)
Last programmed 00:53:56 ago, and selection made 00:53:35 ago
Next selection points
Node scoped :
Uses frequency selection
Used for local line interface output
S Input Last Selection Point QL Pri Status
== ======================= ======================= ===== === ===========
Internal0[1] n/a SEC 255 Available
11 GNSS2[1] n/a PRC 100 Locked
==============
Selection point: IEEE 1588 Clock Selector (2 inputs, 1 selected)
Last programmed 00:53:56 ago, and selection made 00:53:55 ago
Next selection points
Node scoped :
Uses frequency selection
S Input Last Selection Point QL Pri Status
== ======================= ======================= ===== === ===========
21 Internal0[1] n/a SEC 255 Holdover
GNSS2[1] n/a PRC 100 Unmonitored
==============
switch(config-gnss-if)#
```



第 9 章

NTP の設定

この章では、Cisco NX-OS デバイスでネットワーク タイム プロトコル（NTP）を設定する方法について説明します。

この章は、次の項で構成されています。

- NTP の詳細 （151 ページ）
- NTP の前提条件 （153 ページ）
- NTP の注意事項と制約事項 （153 ページ）
- NTP のデフォルト設定 （156 ページ）
- NTP の設定 （156 ページ）
- NTP の設定確認 （167 ページ）
- NTP の設定例 （167 ページ）
- その他の参考資料 （169 ページ）

NTP の詳細

ネットワーク タイム プロトコル（NTP）は、分散している一連のタイム サーバとクライアント間で 1 日の時間を同期させ、複数のネットワーク デバイスから受信するシステム ログや時間関連のイベントを相互に関連付けられるようにします。NTP ではトランスポート プロトコルとして、ユーザ データグラム プロトコル（UDP）を使用します。すべての NTP 通信は UTC を使用します。

NTP サーバは通常、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 クロックなどの正規の時刻源から時刻を受信し、ネットワークを介してこの時刻を配信します。NTP はきわめて効率的で、毎分 1 パケット以下で 2 台のマシンを相互に 1 ミリ秒以内に同期します。

NTP ではストラタム（stratum）を使用して、ネットワーク デバイスと正規の時刻源の距離を表します。

- ストラタム 1 のタイム サーバは、信頼できる時刻源に直接接続されます（無線時計や原子時計または GPS 時刻源など）。
- ストラタム 2 の NTP サーバは、ストラタム 1 のタイム サーバから NTP を使用して時刻を受信します。

同期の前に、NTP は複数のネットワーク サービスが報告した時刻を比較し、1 つの時刻が著しく異なる場合は、それが Stratum 1 であっても、同期しません。Cisco NX-OS は、無線時計や原子時計に接続できず、ストラタム 1 サーバとして動作することはできないため、インターネット上で利用できるパブリック NTP サーバを使用することを推奨します。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、Cisco NX-OS では、NTP によって時刻が同期されていなくても、NTP で同期されているものとして時刻を設定できます。

（注） NTP ピア関係を作成して、サーバで障害が発生した場合に、ネットワーク デバイスを同期させて、正確な時刻を維持するための時刻提供ホストを指定できます。

デバイス上の時刻は重要な情報であるため、NTP のセキュリティ機能を使用して、不正な時刻を誤って（または悪意を持って）設定できないように保護することを強く推奨します。その方法として、アクセス リストベースの制約方式と暗号化認証方式があります。

NTP アソシエーション

N TP アソシエーションは、次のいずれかになります。

- ピア アソシエーション：デバイスが別のデバイスに同期するか、別のデバイスをそのデバイスに同期させることができます。
- サーバ アソシエーション：デバイスは、サーバに同期します。

設定する必要があるのはアソシエーションの片側だけです。他方のデバイスは自動的にアソシエーションを確立できます。

時間サーバとしての NTP

Cisco NX-OS デバイスでは、時刻を配信するために NTP を使用できます。他のデバイスからタイム サーバとして設定できます。デバイスを正規の NTP サーバとして動作するよう設定し、外部の時刻源と同期していないときでも時刻を配信させることもできます。

クロック マネージャ

クロックはさまざまなプロセス間で共有する必要のあるリソースです。NTP などの複数の時刻同期プロトコルが、システムで稼働している可能性があります。

クロック マネージャを使用して、システム内のさまざまなクロックを制御するプロトコルを指定できます。プロトコルを指定すると、システム クロック更新が開始します。クロック マネージャの設定の詳細については『Cisco Nexus 9000 シリーズ NX-OS 基本設定ガイド』を参照してください。

高可用性

NTP はステートレス リスタートをサポートします。リブート後またはスーパーバイザ スイッチオーバー後に、実行コンフィギュレーションが適用されます。ハイアベイラビリティの詳細については、『Cisco Nexus 9000 シリーズ NX-OS ハイ アベイラビリティおよび冗長性ガイド』を参照してください。

NTP ピアを設定すると、NTP サーバ障害の発生時に冗長性が得られます。

仮想化のサポート

NTP は Virtual Routing and Forwarding（VRF）インスタンスを認識します。NTP サーバおよび NTP ピアに対して特定の VRF を設定していない場合、NTP はデフォルトの VRF を使用します。VRF の詳細については、『Cisco Nexus 9000 シリーズ NX-OS ユニキャスト ルーティング設定ガイド』を参照してください

NTP の前提条件

NTP の前提条件は、次のとおりです。

- NTP を設定するには、NTP が動作している 1 つ以上のサーバに接続できなければなりません。

NTP の注意事項と制約事項

NTP に関する設定時の注意事項および制約事項は、次のとおりです。

- NTP サーバ機能はサポートされます。
- デフォルト以外の VRF で名前ベースの NTP サーバ（FQDN）を設定する前に、その特定の VRF で DNS サーバを設定する必要があります。オプションを使用してグローバルコンフィギュレーションモードからDNSサーバを設定する場合、その名前ベースのNTPサーバ設定は実行コンフィギュレーションに追加されません。**use-vrf** この方法を使用して NTP サーバを設定しようとした場合は、コマンドの **no** バージョンを使用して NTP 設定を削除し、その VRF の下に DNS サーバを追加してから、VRF に名前ベースの NTP サーバを追加する必要があります。構成された DNS サーバーは到達可能であり、照会されたときに NTP サーバーの FQDN の正しい IP を返す必要があります。
- 使用するクロックの信頼性が確実な場合（つまり、信頼できる NTP サーバのクライアントである場合）に限り、別のデバイスとの間にピア アソシエーションを設定することを推奨します。
- 単独で設定したピアは、サーバの役割を担いますが、バックアップとして使用する必要があります。サーバが 2 台ある場合、いくつかのデバイスが一方のサーバに接続し、残りの

デバイスが他方のサーバに接続するように設定できます。その後、2 台のサーバ間にピアアソシエーションを設定すると、信頼性の高い NTP 構成になります。

- サーバが 1 台だけの場合は、すべてのデバイスをそのサーバのクライアントとして設定することを推奨します。
- 設定できる NTP エンティティ（サーバおよびピア）は、最大 64 です。
- VRF で NTP を設定する場合は、NTP サーバおよびピアが、設定された VRF を介して相互にアクセスできることを確認します。
- ネットワーク全体の NTP サーバおよび Cisco NX-OS デバイスに、NTP 認証キーを手動で配信します。
- スイッチをエッジ デバイスとして使用して NTP を利用したい場合は、**ntp access-group** コマンドを使用して必要なエッジ デバイスにのみ NTP をフィルタリングすることを推奨します。
- システムに **ntp passive**、**ntp broadcast client**、または **ntp multicast client** コマンドが設定されている場合、対称アクティブの着信パケット、ブロードキャスト パケット、マルチキャスト パケットを NTP が受信する際に、送信者と同期させるための一時的なピア アソシエーションを設定できます。

（注） 上記コマンドのいずれかを有効にする前に必ず **ntp authenticate** bを指定してください。そうしないと、上記のパケット タイプのいずれかを送信する任意のデバイス（悪意のある攻撃者に制御されたデバイスを含む）とデバイスが同期される可能性があります。

- **ntp authenticate** コマンドが指定されている場合、対称アクティブ パケット、ブロードキャスト パケット、マルチキャスト パケットが受信されても、**ntp trusted-key** グローバル コンフィギュレーション コマンドで指定された認証キーの 1 つがパケットで運ばれていない限り、システムとピアの同期は行われません。
- **ntp access-group** コマンドなど他の方法で、デバイスの NTP サービスと非承認ホストとの通信防止の措置が取られている場合を除き、非承認のネットワーク ホストとの同期を避けるには、**ntp passive**、**ntp broadcast client**、**ntp multicast client** コマンドを指定した段階で随時 **ntp authenticate** コマンドを指定する必要があります。
- The **ntp authenticate** コマンドは、**ntp server** および **ntp peer** コンフィギュレーション コマンドで設定されたピア アソシエーションを認証しません。**ntp server** および **ntp peer** アソシエーションを認証するには、**key** キーワードを指定します。
- 1 つの NTP アクセス グループに最大 4 つの IP ACL を設定できます。IPv4 および IPv6 ACL がサポートされています。
- インバンド ポートでパケット フラッディングが発生すると、NTPD による CPU 使用率が 90% を超える可能性があります。NTPD によるこの高い CPU 使用率を克服するには、カ

スタム CoPP ポリシーを使用して、NTP への着信トラフィックをレート制限します。コントロール プレーン ポリシングの詳細については、cisco.com の『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』の関連バージョンの「Configuring Control Plane Policing」の章を参照してください。

（注） 推奨されるレート制限は、ポリシー **CIR** フィールドの場合は 1000 kbps、**BC** フィールドの場合は 64,000 バイトです。

- Cisco NX-OS リリース 10.1(1) 以降、Cisco Nexus 9000 スイッチはストラタム 14 および 15 と同期しません。
- Cisco NX-OS リリース 10.1(1) 以降、NTP バージョン 4（NTPv4）は Nexus スタンドアロンスイッチでサポートされます。
- Cisco NX-OS リリース 10.3(3)F 以降、RFC 8573 標準に沿って、NTP セキュリティは、認証キーのタイプ 6 暗号化サポートとともに AES128CMAC 認証メカニズムによって強化されています。次の注意事項と制限事項が適用されます。
  - この機能には、パスワードをタイプ 0、タイプ 7、またはタイプ 6 として設定するオプションがあります。
  - 構成できる一意のキーの最大数は 1024 で、範囲は 1 ～ 65535 です。
  - タイプ 6 認証を機能させるには、**feature password encryption aes** とともに、構成する新しいタイプ 6 キーを生成するために使用したものと同じプライマリ（マスター）キーをデバイスに構成します。
  - **encryption re-encrypt obfuscated** コマンドを使用して再暗号化を適用すると、タイプ 6 以外のすべての NTP パスワードがタイプ 6 に再暗号化されます。
  - コマンドは、NTP に設定されているすべてのタイプ 6 パスワードを削除します。**encryption delete type6**
  - **encryption decrypt type6** コマンドは、既存の構成済みのタイプ 6 パスワードを復号化します。
  - AES128CMAC/タイプ 6 でサポートされるバージョンから非 AES128CMAC/タイプ 6 でサポートされるバージョンに ISSD を実行するには、タイプ 6 キーの設定を解除してから ISSD を実行します。
  - プログラム（restconf/Netconf など）でキーチェーンを構成する場合は、encryptType と keyString を指定することをお勧めします。指定しない場合、キー チェーン インフラは、欠落しているプロパティのすでに使用可能な（またはデフォルトの）値を使用して keyString を構成します。
  - 欠落しているプロパティを構成する必要がある場合は、両方のピア ルータで同じ手順を実行する必要があります。

NTP のデフォルト設定

次の表に、NTP パラメータのデフォルト設定を示します。

パラメータ	デフォルト
NTP	イネーブル
NTP 認証	ディセーブル
NTP アクセス	イネーブル
NTP ロギング	無効化

NTP の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

NTP の有効化または無効化

NTP をイネーブルまたはディセーブルにできます。NTP はデフォルトでイネーブルです。

手順の概要

1. **configure terminal**
2. [**no**] **feature ntp**
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **feature ntp** 例：	NTP を有効または無効にします。

	コマンドまたはアクション	目的
	`switch(config)# feature ntp`	
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

正規の NTP サーバとしてのデバイスの設定

デバイスを正規の NTP サーバとして動作するよう設定し、既存のタイム サーバと同期していないときでも時刻を配信させることができます。

手順の概要

1. **configure terminal**
2. [**no**] **ntp master** [*stratum*]
3. （任意） **show running-config ntp**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **ntp master** [*stratum*] 例： `switch(config)# ntp master`	正規の NTP サーバとしてデバイスを設定します。 NTP クライアントがこれらの時間を同期するのと別の階層レベルを指定できます。指定できる範囲は 1 ～ 15 です。
ステップ 3	（任意） **show running-config ntp** 例： `switch(config)# show running-config ntp`	NTP コンフィギュレーションを表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

NTP サーバおよびピアの設定

NTP サーバおよびピアを設定できます。

始める前に

使用している NTP サーバと、そのピアの IP アドレスまたはドメイン ネーム システム（DNS）名がわかっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **ntp server** {*ip-address* | *ipv6-address* | *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*]
3. [**no**] **ntp peer** {*ip-address* | *ipv6-address* | *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*]
4. （任意） **show ntp peers**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **ntp server** {*ip-address* | *ipv6-address* | *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*] 例： `switch(config)# ntp server 192.0.2.10`	1 つのサーバと 1 つのサーバ アソシエーションを形成します。 NTP サーバとの通信で使用するキーを設定するには、**key** キーワードを使用します。*key-id* 引数の範囲は 1 ～ 65535 です。 サーバをポーリングする最大および最小の間隔を設定するには、 **maxpoll** および **minpoll** キーワードを使用します。 *max-poll* および *min-poll* 引数の範囲は4ー16（2の累乗として設定されます。つまり、実質的に16ー65536秒）で、デフォルト値はそれぞれ 6 と 4 です（*maxpoll* デフォルト = 64秒、*minpoll* デフォルト= 16秒）。 このサーバをデバイスの優先 NTP サーバにするには、 **prefer** キーワードを使用します。

	コマンドまたはアクション	目的
		指定された VRF を介して通信するように NTP サーバを設定するには、**use-vrf** キーワードを使用します。*vrf-name* 引数には、**default**、**management**、または大文字と小文字が区別される最大 32 文字の任意の英数字文字列を指定できます。 （注） NTP サーバとの通信で使用するキーを設定する場合は、そのキーが、デバイス上の信頼できるキーとして存在していることを確認してください。
ステップ 3	[**no**] **ntp peer** {*ip-address* | *ipv6-address* | *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*] 例： `switch(config)# ntp peer 2001:0db8::4101`	1 つのピアと 1 つのピア アソシエーションを形成します。複数のピア アソシエーションを指定できます。 NTP ピアとの通信で使用するキーを設定するには、**key** キーワードを使用します。*key-id* 引数の範囲は 1 ～ 65535 です。 サーバをポーリングする最大および最小の間隔を設定するには、 **maxpoll** および **minpoll** キーワードを使用します。 *max-poll* および *min-poll* 引数の範囲は4ー16（2の累乗として設定されます。つまり、実質的に 16ー131072 秒）で、デフォルト値はそれぞれ 6 と 4 です（*maxpoll* デフォルト＝64秒、*minpoll* デフォルト= 16秒）。 デバイスに対して対象の NTP ピアを優先にするには、**prefer** キーワードを使用します。 指定された VRF を介して通信するように NTP ピアを設定するには、**use-vrf** キーワードを使用します。*vrf-name* 引数には、**default**、**management**、または大文字と小文字が区別される最大 32 文字の任意の英数字文字列を指定できます。
ステップ 4	（任意） **show ntp peers** 例： `switch(config)# show ntp peers`	設定されたサーバおよびピアを表示します。 （注） ドメイン名が解決されるのは、DNS サーバが設定されている場合だけです。 DNS/ネームサーバが IPv4 と IPv6 の両方を解決する場合、NX-OS では IPv6 アドレスが優先されます。
ステップ 5	（任意） **copy running-config startup-config** 例：	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

	コマンドまたはアクション	目的
	`switch(config)# copy running-config startup-config`	

NTP 認証の設定

ローカル ロックを同期させる時刻源を認証するようデバイスを設定できます。NTP 認証をイネーブルにすると、**ntp trusted-key** コマンドによって指定されたいずれかの認証キーを時刻源が保持している場合のみ、デバイスはその時刻源と同期します。デバイスは、認証チェックに失敗したすべてのパケットをドロップし、それらのパケットでローカルクロックがアップデートされないようにします。NTP 認証はデフォルトでディセーブルになっています。

始める前に

この手順で指定する予定の認証キーによって、NTP サーバが設定されていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **ntp authentication-key** *number* {**md5** | **aes128cmac**} *password string encryption-type*
3. **ntp server** *ip-address* **key** *key-id*
4. （任意） **show ntp authentication-keys**
5. [**no**] **ntp trusted-key** *number*
6. （任意） **show ntp trusted-keys**
7. [**no**] **ntp authenticate**
8. （任意） **show ntp authentication-status**
9. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **ntp authentication-key** *number* {**md5** | **aes128cmac**} *password string encryption-type* 例：	認証キーを定義します。認証キーの範囲は1～65535です。 デバイスが時刻源と同期するのは、時刻源がこれらの認証キーのいずれかを持ち、**ntp trusted-key** *number*

	コマンドまたはアクション	目的

	`switch(config)# ntp authentication-key 42 md5 aNiceKey`
	`switch(config)# ntp authentication-key 21 md5 JDYk3pp/Fuv0zWyVSRhS6EDERSSsp1uA7s57dvdsx g74ndf02lEI9dF6WX6Z78/5R8qPmSRRrDUDtCcUlZ XDUrf0ErodS3ikPQA= 6`
	`switch(config)# ntp authentication-key 12 aes128cmac JDYkzj4NojJdSkQPvBhFvAO9xCSVwj2iRGvShNSg ER4JwMBMtUEibfqkscgZ4+/iTdDmeCRW9SGWLxKb 3Xk5g8pz4bR7Iu1a7QA= 6`
		コマンドによってキー番号が指定されている場合だけです。
		md5 または aes128cmac 認証方式を選択できます。
		ユーザーが同じプライマリ（**マスター**）キーから生成されたタイプ 6 キーを使用している場合、ユーザーが**機能パスワード暗号化 aes** を有効にするまで、デバイスは時刻源に同期しません。
		タイプ 0 およびタイプ 7 の暗号化タイプの場合、最大長は 32 文字です。リリース 10.3(3)F までは、15 文字（英数字）でした。タイプ 6 暗号化タイプの場合、最大文字数は 128 文字です。

ステップ 3	**ntp server** *ip-address* **key** *key-id* **例**： `switch(config)# ntp server 192.0.2.1 key 1001`	1 つのサーバと 1 つのサーバ アソシエーションを形成します。 NTP サーバとの通信で使用するキーを設定するには、**key** キーワードを使用します。*key-id* 引数の範囲は 1 ～ 65535 です。 認証を必須とする場合は、**key** キーワードを使用する必要があります。**ntp server** または **ntp peer** コマンドで **key** キーワードを指定しない場合、認証なしでの動作が続けられます。
ステップ 4	（任意） **show ntp authentication-keys** **例**： `switch(config)# show ntp authentication-keys`	設定済みの NTP 認証キーを表示します。
ステップ 5	[**no**] **ntp trusted-key** *number* **例**： `switch(config)# ntp trusted-key 42`	1 つ以上のキー（ステップ 2 で定義されているもの）を指定します。デバイスを時刻源と同期させるには、未設定のリモート シンメトリック、ブロードキャスト、およびマルチキャストの時刻源を NTP パケット内に入力する必要があります。trusted key の範囲は 1 ～ 65535 です。 このコマンドにより、デバイスが、信頼されていない時刻源と誤って同期する、ということが防止されます。
ステップ 6	（任意） **show ntp trusted-keys** **例**： `switch(config)# show ntp trusted-keys`	設定済みの NTP の信頼されているキーを表示します。

	コマンドまたはアクション	目的
ステップ 7	[**no**] **ntp authenticate** 例： `switch(config)# ntp authenticate`	ntp passive、ntp broadcast client、および ntp multicast で認証を有効または無効にします。NTP 認証はデフォルトでディセーブルになっています。
ステップ 8	（任意） **show ntp authentication-status** 例： `switch(config)# show ntp authentication-status`	NTP 認証の状況を表示します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

NTP アクセス制限の設定

アクセス グループを使用して、NTP サービスへのアクセスを制御できます。具体的には、デバイスで許可する要求のタイプ、およびデバイスが応答を受け取るサーバを指定できます。

アクセス グループを設定しない場合は、すべてのデバイスに NTP アクセス権が付与されます。何らかのアクセス グループを設定した場合は、ソース IP アドレスがアクセス リストの基準をパスしたリモート デバイスに対してだけ、NTP アクセス権が付与されます。

- **match-all** キーワードがない場合、パケットは permit が見つかるまでアクセス グループに対して（以下に示す順で）評価されます。permit が検出されない場合、パケットはドロップされます。
- **match-all** キーワードがある場合、パケットはすべてのアクセス グループに対して（以下に示す順で）評価され、最後に成功した評価（ACL が設定されている最後のアクセス グループ）に基づいてアクションが実行されます。
- peer：クライアント、対称アクティブ、対称パッシブ、サービス、コントロール、およびプライベート パケット（すべてのタイプ）を処理
- serve：クライアント、コントロール、およびプライベート パケットを処理
- serve-only：クライアント パケットだけを処理
- query-only：コントロールおよびプライベート パケットだけを処理

アクセス グループは次の順で評価されます：

1. peer（すべてのパケット タイプ）
2. serve（クライアント、コントロール、およびプライベート パケット）
3. serve-only（クライアント パケット）または query-only（コントロールおよびプライベート パケット）

serve-only または query-only の ACL 処理は、NTP パケット タイプによって異なります。

手順の概要

1. **configure terminal**
2. [**no**] **ntp access-group match-all** | {{**peer** | **serve** | **serve-only** | **query-only** }*access-list-name*}
3. （任意） **show ntp access-groups**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **ntp access-group match-all** | {{**peer** | **serve** | **serve-only** | **query-only** }*access-list-name*}
	例：
	`switch(config)# ntp access-group match-all`
	`switch(config)# ntp access-group peer peer-acl`
	`switch(config)# ntp access-group serve serve-acl`
		NTP のアクセスを制御し、基本の IP アクセス リストを適用するためのアクセス グループを作成または削除します。
		設定ピアの遅延 ACL 1ルールに NTP が一致する場合、ACL 処理は停止し、次のアクセス グループ オプションへと継続しません。
		• **peer** キーワードは、デバイスが時刻要求と NTP 制御クエリーを受信し、アクセス リストで指定されているサーバと同期するようにします。
		• **serve** キーワードは、アクセス リストに指定されているサーバからの時刻要求と NTP 制御クエリーをデバイスが受信できるようにしますが、指定されたサーバとは同期しないようにします。
		• **serve-only** キーワードは、デバイスがアクセス リストで指定されたサーバからの時刻要求だけを受信するようにします。
		• **query-only** キーワードは、デバイスがアクセス リストで指定されたサーバからの NTP 制御クエリーのみを受信するようにします。
		• **match-all** キーワードを使用すると、アクセス グループ オプションが、制限の最も緩いものから最も厳しいもの、peer、serve、serve-only、

	コマンドまたはアクション	目的
		query-only の順序でスキャンされるようにできます。着信パケットが peer アクセス グループの ACL に一致しない場合、パケットは serve アクセス グループに送信され、処理されます。パケットが serve アクセス グループの ACL に一致しない場合、serve-only アクセス グループに送られ、これが継続されます。 （注） **match-all** キーワードは、Cisco NX-OS リリース 7.0(3)I6(1) 以降で利用可能なもので、Cisco Nexus 9000 シリーズ スイッチと、Cisco Nexus 3164Q、31128PQ、3232C、および 3264Q スイッチでサポートされています。 • *access-list-name* 変数は、NTP アクセスグループの名前です。名前は、特殊文字を含む、最大 64 文字の英数字ストリングで指定できます。
ステップ 3	（任意） **show ntp access-groups** 例： `switch(config)# show ntp access-groups`	NTP アクセス グループのコンフィギュレーションを表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

NTP ソース IP アドレスの設定

NTP は、NTP パケットが送信されたインターフェイスのアドレスに基づいて、すべての NTP パケットにソース IP アドレスを設定します。特定のソース IP アドレスを使用するよう NTP を設定できます。

手順の概要

1. **configure terminal**
2. [**no**] **ntp source** *ip-address*
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **ntp source** *ip-address* 例： `switch(config)# ntp source 192.0.2.1`	すべての NTP パケットにソース IP アドレスを設定します。*ip-address* には IPv4 または IPv6 形式を使用できます。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

NTP ソース インターフェイスの設定

特定のインターフェイスを使用するよう NTP を設定できます。

手順の概要

1. **configure terminal**
2. [**no**] **ntp source-interface** *interface*
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **ntp source-interface** *interface* 例： `switch(config)# ntp source-interface ethernet 2/1`	すべての NTP パケットに対してソース インターフェイスを設定します。サポートされているインターフェイスのリストを表示するには、**?** キーワードを使用します。

	コマンドまたはアクション	目的
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

NTP ロギングの設定

重要な NTP イベントでシステム ログを生成するよう、NTP ロギングを設定できます。NTP ロギングはデフォルトでディセーブルになっています。

手順の概要

1. **configure terminal**
2. [**no**] **ntp logging**
3. （任意） **show ntp logging-status**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **ntp logging** 例： `switch(config)# ntp logging`	重要な NTP イベントでシステム ログを生成することをイネーブルまたはディセーブルにします。NTP ロギングはデフォルトでディセーブルになっています。
ステップ 3	（任意） **show ntp logging-status** 例： `switch(config)# show ntp logging-status`	NTP ロギングのコンフィギュレーション状況を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

NTP の設定確認

NTP 設定を表示するには、次のタスクのうちのいずれかを実行します。

コマンド	目的
show ntp access-groups	NTP アクセス グループのコンフィギュレーションを表示します。
show ntp authentication-keys	設定済みの NTP 認証キーを表示します。
show ntp authentication-status	NTP 認証の状況を表示します。
show ntp logging-status	NTP のロギング状況を表示します。
show ntp peer-status	すべての NTP サーバおよびピアのステータスを表示します。
show ntp peers	すべての NTP ピアを表示します。
show ntp rts-update	RTS アップデートの状況を表示します。
ntp ソースを表示する	設定済みの NTP ソース IP アドレスを表示します。
show ntp source-interface	設定済みの NTP ソース インターフェイスを表示します。
show ntp statistics {**io** | **local** | **memory** | **peer** {**ipaddr** {*ipv4-addr* | *ipv6-addr*} | **name** *peer-name*}}	NTP 統計情報を表示します。
show ntp trusted-keys	設定済みの NTP の信頼されているキーを表示します。
show running-config ntp	NTP 情報を表示します。

NTP セッションをクリアするには、**clear ntp session** コマンドを使用します。

NTP 統計情報を消去するには、**clear ntp statistics** コマンドを使用します。

NTP の設定例

次に、NTP パケット内で認証キー 42 を提示している時刻源とだけ同期するようデバイスを構成する md5 の例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp authentication-key 42 md5 aNiceKey
switch(config)# ntp server 192.0.2.105 key 42
switch(config)# ntp trusted-key 42
switch(config)# ntp authenticate
switch(config)# copy running-config startup-config
```

```
[########################################] 100%
switch(config)#
```

次に、NTP パケット内で認証キー 12 を提示している時刻源とだけ同期するようデバイスを構成する aes128cmac の例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp authentication-key 12 aes128cmac password 0/6/7
switch(config)# ntp server 192.0.2.105 key 12
switch(config)# ntp trusted-key 12
switch(config)# ntp authenticate
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

次に、以下の制約事項のある NTP アクセス グループの設定の例を示します。

- peer の制約事項は、「peer-acl」というアクセス リストの条件を満たす IP アドレスに適用されます。
- serve の制約事項は、「serve-acl」というアクセス リストの条件を満たす IP アドレスに適用されます。
- serve-only の制約事項は、「serve-only-acl」というアクセス リストの条件を満たす IP アドレスに適用されます。
- query-only の制約事項は、「query-only-acl」というアクセス リストの条件を満たす IP アドレスに適用されます。

```
switch# configure terminal
switch(config)# ntp peer 10.1.1.1
switch(config)# ntp peer 10.2.2.2
switch(config)# ntp peer 10.3.3.3
switch(config)# ntp peer 10.4.4.4
switch(config)# ntp peer 10.5.5.5
switch(config)# ntp peer 10.6.6.6
switch(config)# ntp peer 10.7.7.7
switch(config)# ntp peer 10.8.8.8
switch(config)# ntp access-group peer peer-acl
switch(config)# ntp access-group serve serve-acl
switch(config)# ntp access-group serve-only serve-only-acl
switch(config)# ntp access-group query-only query-only-acl
switch(config)# ip access-list peer-acl
switch(config-acl)# 10 permit ip host 10.1.1.1 any
switch(config-acl)# 20 permit ip host 10.8.8.8 any
switch(config)# ip access-list serve-acl
switch(config-acl)# 10 permit ip host 10.4.4.4 any
switch(config-acl)# 20 permit ip host 10.5.5.5 any
switch(config)# ip access-list serve-only-acl
switch(config-acl)# 10 permit ip host 10.6.6.6 any
switch(config-acl)# 20 permit ip host 10.7.7.7 any
switch(config)# ip access-list query-only-acl
switch(config-acl)# 10 permit ip host 10.2.2.2 any
switch(config-acl)# 20 permit ip host 10.3.3.3 any
```

（注） 単一の ACL グループのみが適用される場合、他の ACL カテゴリに関連するすべてのパケットは拒否され、設定された ACL グループに関連するパケットのみが処理されます。これについては、以下のシナリオで説明します。

- serve ACL が設定されている場合、クライアント、コントロール、およびプライベート パケットのみが処理され、他のすべてのパケットは拒否されます。
- serve-only ACL が設定されている場合、クライアント パケットのみが処理され、他のすべてのパケットは拒否されます。

複数の ACL が設定されている場合、以下のシナリオで説明されている処理の順序に従います。

- serve と serve-only の両方が、match-all が構成されていない同じ IP アドレスに対して構成されていて、IP が serve-acl で許可され、serve-only で拒否されている場合、クライアント、コントロール、プライベート パケットはその IP に対して許可されます。

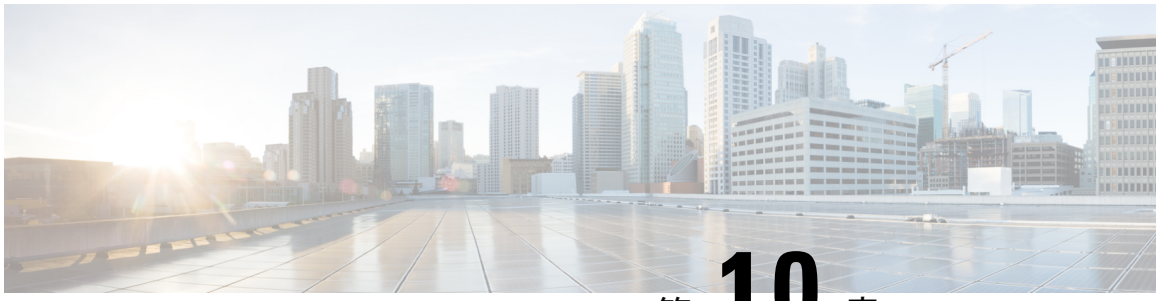
その他の参考資料

関連資料

関連項目	マニュアル タイトル
クロック マネージャ	『Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide』

MIB

MIB	MIB のリンク
NTP に関連する MIB	サポートされている MIB を検索およびダウンロ 次の URL にアクセスしてください。 https://cisco.github.io/cisco-mibs/supportlists/nexus9 Nexus9000MIBSupportList.html



第 10 章

CDP の設定

この章では、Cisco NX-OS デバイス上で Cisco Discovery Protocol（CDP）を設定する方法について説明します。

この章は、次の項で構成されています。

- CDP について （171 ページ）
- CDP の注意事項と制約事項 （173 ページ）
- CDP のデフォルト設定 （173 ページ）
- CDP の設定 （174 ページ）
- CDP コンフィギュレーションの確認 （177 ページ）
- CDP のコンフィギュレーション例 （178 ページ）

CDP について

Cisco Discovery Protocol（CDP）は、ルータ、ブリッジ、アクセス サーバ、コミュニケーション サーバ、スイッチを含め、シスコ製のあらゆる機器で動作する、メディアにもプロトコルにも依存しないプロトコルです。CDP を使用すると、デバイスに直接接続されているすべてのシスコ デバイスの情報を検出して表示できます。

CDP はネイバー デバイスのプロトコル アドレスを収集し、各デバイスのプラットフォームを検出します。CDP の動作はデータリンク層上に限定されます。異なるレイヤ 3 プロトコルをサポートする 2 つのシステムで相互学習が可能です。

CDP が設定された各デバイスは、マルチキャスト アドレスに定期的にアドバタイズメントを送信します。各デバイスは、SNMP メッセージを受信できるアドレスを少なくとも 1 つアドバタイズします。アドバタイズメントには保持時間情報も含まれます。保持時間は、受信デバイスが CDP 情報を削除するまでに保持する時間の長さを表します。アドバタイズメントまたはリフレッシュ タイマーおよびホールド タイマーを設定できます。

CDP Version-2（CDPv2）では、接続デバイスとの間でネイティブ VLAN ID またはポート デュプレックス ステートが一致していないインスタンスを追跡できます。

CDP では、次の Type-Length-Value（TLV）フィールドがアドバタイズされます。

- デバイス ID

- アドレス
- ポート ID
- 機能
- バージョン
- プラットフォーム
- ネイティブ VLAN
- 全二重/半二重
- MTU
- SysName
- SysObjectID
- 管理アドレス
- Physical Location
- VTP

すべての CDP パケットに VLAN ID が含まれます。レイヤ 2 アクセス ポート上で CDP を設定した場合、そのアクセス ポートから送信される CDP パケットには、アクセス ポートの VLAN ID が含まれます。レイヤ 2 トランク ポート上で CDP を設定した場合は、そのトランク ポートから送信される CDP パケットに、トランク ポート上で許可設定されている最小の VLAN ID が含まれます。トランク ポートは、そのトランク ポートの許可 VLAN リストに指定されている VLAN ID であれば、どの VLAN ID が含まれている CDP パケットでも受信できます。VLAN の詳細については、『Cisco Nexus 9000 シリーズ NX-OS レイヤ 2 スイッチング設定ガイド』を参照してください。

VTP 機能のサポート

次の条件に当てはまる場合、CDP は VLAN トランキング プロトコル（VTP）の type-length-value（TLV）フィールドを送信します。

- CDP バージョン 2 がイネーブルになっています。
- VTP 機能がイネーブルになっています。
- VTP ドメイン名が設定されています。

show cdp neighbors detail コマンドを使用すると、VTP 情報を参照できます。

高可用性

Cisco NX-OS は、CDP のステートフルおよびステートレス両方のリスタートとスイッチオーバーをサポートします。ハイ アベイラビリティの詳細については、『*Cisco Nexus 9000* シリーズ *NX-OS* ハイ アベイラビリティおよび冗長性ガイド』を参照してください。

仮想化のサポート

Cisco NX-OS は、CDP のインスタンスを 1 つサポートします。

CDP の注意事項と制約事項

CDP に関する設定時の注意事項および制約事項は、次のとおりです。

- 接続数が 256 のハブにポートを接続した場合、CDP はポートあたり最大 256 のネイバーを検出できます。
- デバイス上で CDP をイネーブルにする必要があります。イネーブルにしておかないと、インターフェイス上で CDP をイネーブルにできません。
- CDP を設定できるのは、物理インターフェイスおよびポート チャネル上に限られます。
- Cisco NX-OS リリース 10.3(1)F 以降、CDP は Cisco Nexus 9808 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、CDP は次のライン カードおよびスイッチでサポートされます。
  - Cisco Nexus 9804 プラットフォーム スイッチ
  - Cisco Nexus X98900CD-A ライン カード
  - Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ
- Cisco NX-OS リリース 10.4(2)F 以降、CDP は Cisco Nexus 9232E-B1 プラットフォーム スイッチでサポートされます。

CDP のデフォルト設定

次の表に、CDP パラメータのデフォルト設定を示します。

パラメータ	デフォルト
CDP	グローバルおよびすべてのインターフェイスでイネーブル
CDP version	バージョン 2

パラメータ	デフォルト
CDP device ID	シリアル番号
CDP timer	60 秒
CDP hold timer	180 秒

CDP の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があります。

CDP のグローバルな有効化または無効化

CDP はデフォルトで有効になっています。CDP をディセーブルにしてから、もう一度イネーブルにできます。

インターフェイス上で CDP をイネーブルにするには、先にデバイス上で CDP をイネーブルにしておく必要があります。CDP がグローバルなディセーブルになっているときに、特定のインターフェイス上で CDP をイネーブルにしても、これらのインターフェイス上で CDP がアクティブになることはなく、エラー メッセージが戻ります。

手順の概要

1. **configure terminal**
2. [**no**] **cdp enable**
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal switch(config)#`	グローバル設定モードを開始します。
ステップ 2	[**no**] **cdp enable** 例： `switch(config)# cdp enable`	デバイス全体で CDP 機能を有効または無効にします。デフォルトでは有効。

	コマンドまたはアクション	目的
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

インターフェイス上での CDP の有効化または無効化

CDP はデフォルトで、インターフェイス上でイネーブルです。インターフェイス上で CDP をディセーブルにできます。

CDP がグローバルなディセーブルになっているときに、特定のインターフェイス上で CDP をイネーブルにしても、これらのインターフェイス上で CDP がアクティブになることはなく、エラー メッセージが戻ります。

手順の概要

1. **configure terminal**
2. **interface** *interface slot/port*
3. [**no**] **cdp enable**
4. （任意） **show cdp interface** *interface slot/port*
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**interface** *interface slot/port*
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		インターフェイス設定モードを開始します。

ステップ 3	[**no**] **cdp enable** 例： `switch(config-if)# cdp enable`	このインターフェイスでCDPを有効または無効にします。デフォルトでは有効。 （注） CDP がデバイス上でグローバルに有効になっていることを確認します。

	コマンドまたはアクション	目的
ステップ 4	（任意） **show cdp interface** *interface slot/port* 例： `switch(config-if)# show cdp interface ethernet 1/2`	インターフェイスの CDP 情報を表示します。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

CDP オプション パラメータの設定

この手順でオプションのコマンドを使用して CDP を変更できます。

手順の概要

1. **configure terminal**
2. （任意） **cdp advertise** {**v1** | **v2**}
3. （任意） **cdp format device-id** {**mac-address** | **serial-number** | **system-name**}
4. （任意） **cdp holdtime** *seconds*
5. （任意） **cdp timer** *seconds*
6. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal switch(config)#`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	（任意） **cdp advertise** {**v1**	**v2**} 例： `switch(config)# cdp advertise v1`	デバイスがサポートする CDP のバージョンを設定します。デフォルトは v2 です。
ステップ 3	（任意） **cdp format device-id** {**mac-address**	**serial-number**	**system-name**} 例： `switch(config)# cdp format device-id mac-address`	CDP デバイス ID を設定します。オプションは次のとおりです。 • **mac-address**：シャーシの MAC アドレスを指定します。

	コマンドまたはアクション	目的
		• **serial-number**：シャーシのシリアル番号/組織固有識別子（OUI） • **system-name**：システム名または完全修飾ドメイン名 デフォルトは **system-name** です。
ステップ 4	（任意） **cdp holdtime** *seconds* 例： `switch(config)# cdp holdtime 150`	CDP ネイバー情報を削除するまでに保持する時間を設定します。範囲は 10 ～ 255 秒です。デフォルト値は 180 秒です。
ステップ 5	（任意） **cdp timer** *seconds* 例： `switch(config)# cdp timer 50`	CDP がネイバーにアドバタイズメントを送信するリフレッシュ タイムを設定します。範囲は 5 ～ 254 秒です。デフォルトは 60 秒です。
ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

CDP コンフィギュレーションの確認

CDP 設定を表示するには、次のタスクのうちのいずれかを実行します。

コマンド	目的
show cdp all	CDP がイネーブルになっているすべてのインターフェイスを表示します。
show cdp entry {**all** | **name** *entry-name*}	CDP データベース エントリを表示します。
show cdp global	CDP グローバル パラメータを表示します。
show cdp interface *interface slot*/*port*	CDP インターフェイスのステータスを表示します。
show cdp neighbors {**device-id** | **interface** *interface slot*/*port*} [**detail**]	CDP ネイバーのステータスを表示します。
show cdp interface *interface slot*/*port*	インターフェイスの CDP トラフィック統計を表示します。

インターフェイスの CDP 統計情報を消去するには、**clear cdp counters** コマンドを使用します。

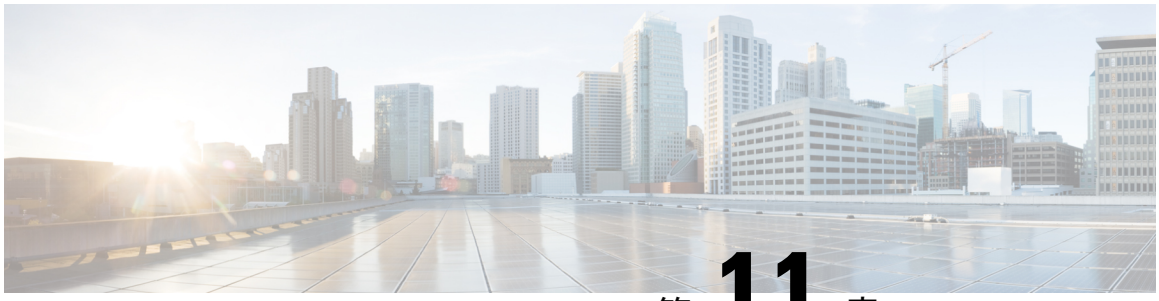
1 つまたはすべてのインターフェイスの CDP キャッシュを消去するには、**clear cdp table** コマンドを使用します。

show cdp neighbors detail コマンドを（**show cdp neighbors** コマンドの代わりに）使用することを推奨します。**show cdp neighbors** コマンドが表示するのは、プラットフォーム名の 13 文字だけです。完全なプラットフォーム名を表示するには、**show cdp neighbors detail** コマンドを使用します。

CDP のコンフィギュレーション例

CDP 機能を有効にして、リフレッシュ タイマーおよびホールド タイマーを設定する例を示します。

```
configure terminal
cdp enable
cdp timer 50
cdp holdtime 100
```



第 11 章

システムメッセージロギングの設定

この章では、Cisco NX-OS デバイス上でシステム メッセージ ロギングを設定する方法について説明します。

この章は、次の内容で構成されています。

- システム メッセージ ロギングの詳細, on page 179
- システム メッセージ ロギングの注意事項および制約事項 （181 ページ）
- システム メッセージ ロギングのデフォルト設定, on page 182
- システムメッセージロギングの設定 （182 ページ）
- システム メッセージ ロギングの設定確認, on page 199
- 繰り返されるシステム ロギング メッセージ （200 ページ）
- システム メッセージ ロギングの設定例 （201 ページ）
- その他の参考資料 （201 ページ）

システム メッセージ ロギングの詳細

システムメッセージロギングを使用して宛先を制御し、システムプロセスが生成するメッセージの重大度をフィルタリングできます。端末セッション、ログ ファイル、およびリモート システム上の Syslog サーバへのロギングを設定できます。

システムメッセージのフォーマットおよびデバイスが生成するメッセージの詳細については、『Cisco NX-OS System Messages Reference』を参照してください。

デフォルトでは、デバイスはターミナル セッションにメッセージを出力し、ログ ファイルにシステム メッセージをログ記録します。

次の表に、システムメッセージで使用されている重大度を示します。重大度を設定する場合、システムはそのレベル以下のメッセージを出力します。

Table 8: **システム メッセージの重大度**

レベル	説明
0：緊急	システムが使用不可

レベル	説明
1：アラート	即時処理が必要
2：クリティカル	クリティカル状態
3：エラー	エラー状態
4：警告	警告状態
5：通知	正常だが注意を要する状態
6：情報	単なる情報メッセージ
7：デバッグ	デバッグ実行時にのみ表示

デバイスは重大度 0、1、または 2 のメッセージのうち、最新の 100 メッセージを NVRAM ログに記録します。NVRAM へのロギングは設定できません。

メッセージを生成したファシリティと重大度に基づいて記録するシステム メッセージを設定できます。

Syslogサーバ

syslog サーバは、syslog プロトコルに基づいてシステム メッセージを記録するリモート システム上で動作します。IPv4 または IPv6 の Syslog サーバを最大 8 つ設定できます。

ファブリック内のすべてのスイッチで syslog サーバの同じ設定をサポートするために、Cisco Fabric Services（CFS）を使用して syslog サーバ設定を配布できます。

Note 最初のデバイス初期化時に、メッセージが syslog サーバに送信されるのは、ネットワークの初期化後です。

セキュアな Syslog サーバ

Cisco NX-OS リリース 9.2(1) 以降では、リモート ロギング サーバへのセキュアな TLS トランスポート接続をサポートするように Syslog サーバを設定できます。さらに、相互認証の設定によって NX-OS スイッチ（クライアント）のアイデンティティを強化することができます。NX-OS スイッチの場合、この機能は TLSv1.1 および TLSv1.2 をサポートします。

セキュアな Syslog サーバの機能では、デバイス認証および暗号化を提供するために TCP/TLS トランスポートおよびセキュリティ プロトコルを使用します。この機能を使用すると、（クライアントとして機能している）Cisco NX-OS デバイスが、ロギングにセキュアな接続をサポートする（サーバとして機能している）リモート Syslog サーバに対してセキュアな暗号化されたアウトバウンド接続を確立できるようになります。認証と暗号化により、この機能では、セキュリティ保護されていないネットワーク上でもセキュアな通信を実現できます。

システムメッセージロギングの注意事項および制約事項

システム メッセージ ロギングには次の設定上の注意事項と制約事項があります。

- システム メッセージは、デフォルトでコンソールおよびログ ファイルに記録されます。
- syslog サーバに到達する前に出力されるシステム メッセージ（スーパーバイザ アクティブ メッセージやオンライン メッセージなど）は、syslog サーバに送信できません。
- Cisco では、すべてのプロセスのログ レベルをデフォルトのまま維持することを推奨しています。レベルを上げて高い値に設定すると、お客様向けではないsyslogメッセージが表示される可能性があります。これらのメッセージは、誤ったアラームを生成する可能性があり、通常は TAC による短期的なトラブルシューティングの目的で使用されます。Cisco では、デフォルトよりも上のレベルの syslog メッセージをサポートしていません。
- Syslog の制限により、securePOAP pem ファイル名の文字長は 230 文字に制限されていますが、セキュア POAP は pem ファイル名として 256 文字の長さをサポートしています。
- Cisco NX-OS リリース 9.2(1) 以降では、リモート ロギング サーバへのセキュアな TLS トランスポート接続をサポートするようにSyslogサーバを設定できます。この機能は、TLS v1.1 および TLS v1.2 をサポートします。
- Cisco NX-OS リリース 10.2(4)M 以降、TLS v1.3 が Cisco Nexus 9000 シリーズ プラットフォーム スイッチでサポートされています。
- セキュアなsyslogサーバがインバンド（非管理）インターフェイスを介して到達できるようにするには、CoPP プロファイルに調整が必要な場合があります。特に、複数のロギング サーバが設定されている場合、および短時間で多数の syslog が生成される場合（ブートアップや設定アプリケーションなど）。
- このガイドラインは、ユーザ定義の永続ロギング ファイルに適用されます。

  syslogコマンド**logging logfile** では、永続的な場所（/logflash/log）と非永続的な場所（/log）の両方でログファイルを設定できます。

  デフォルトのログファイルには「messages」という名前が付けられ、バックアップ ファイル（存在する場合）とともに、**delete /log/** または **delete logflash:/log/** コマンドでもこのファイルは messages.1、messages.2、messages.3、messages.4 を削除できません。

  カスタム名のログファイル（**logging logfile** *file-name severity*）を設定するためのプロビジョニングがありますが、このカスタム名のファイルは削除操作によって削除できます。この場合、syslog ロギングは機能しません。

  たとえば、カスタム名のログファイルが設定され、同じファイルが削除操作によって削除されます。これは意図的な削除操作であるため、syslogメッセージをカスタムログファイルに記録するには、コマンド **logging logfile** *file-name severity* を使用してカスタム ログファイルを再設定する必要があります。この設定が実行されるまで、syslog ロギングは実行できません。

- 通常、syslog にはローカル タイム ゾーンが表示されます。ただし、NGINX などの一部のコンポーネントでは、ログが UTC タイム ゾーンで表示されます。

システム メッセージ ロギングのデフォルト設定

次の表に、システム メッセージ ロギング パラメータのデフォルト設定を示します。

Table 9: デフォルトのシステム メッセージ ロギング パラメータ

パラメータ	デフォルト
コンソール ロギング	重大度 2 でイネーブル
モニタ ロギング	重大度 5 でイネーブル
ログ ファイル ロギング	重大度 5 のメッセージ ロギングがイネーブル
モジュール ロギング	重大度 5 でイネーブル
ファシリティ ロギング	イネーブル
タイムスタンプ単位	秒
Syslog サーバ ロギング	ディセーブル
Syslog サーバ設定の配布	無効化

システムメッセージロギングの設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

ターミナル セッションへのシステム メッセージ ロギングの設定

重大度に基づいて、コンソール、Telnet、および SSH セッションにメッセージを記録するようにデバイスを設定できます。

デフォルトでは、ターミナル セッションでロギングはイネーブルです。

Note コンソールのボー レートが 9600 ボー（デフォルト）の場合、現在の Critical（デフォルト）ロギング レベルが維持されます。コンソール ロギング レベルを変更しようとすると、必ずエラー メッセージが生成されます。ロギング レベルを上げる（Critical よりも上に）には、コンソールのボー レートを 38400 ボーに変更する必要があります。

SUMMARY STEPS

1. **terminal monitor**
2. **configure terminal**
3. [**no**] **logging console** [*severity-level*]
4. (Optional) **show logging console**
5. [**no**] **logging monitor** [*severity-level*]
6. (Optional) **show logging monitor**
7. [**no**] **logging message interface type ethernet description**
8. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**terminal monitor** **Example:** `switch# terminal monitor`	デバイスがコンソールにメッセージを記録できるようにします。

ステップ 2	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 3	[**no**] **logging console** [*severity-level*] **Example:** `switch(config)# logging console 3`	指定された重大度とそれより上位の重大度のメッセージをコンソールセッションに記録するように、デバイスを設定します。小さい値は、より高い重大度を示します。重大度は 0 ～ 7 の範囲です。 • 0：緊急 • 1：アラート • 2：クリティカル • 3：エラー • 4：警告 • 5：通知

	Command or Action	Purpose
		• 6：情報 • 7：デバッグ 重大度が指定されていない場合、デフォルトの 2 が使用されます。**no** オプションは、メッセージをコンソールにログするデバイスの機能をディセーブルにします。
ステップ 4	(Optional) **show logging console** **Example:** `switch(config)# show logging console`	コンソール ロギング設定を表示します。
ステップ 5	[**no**] **logging monitor** [*severity-level*] **Example:** `switch(config)# logging monitor 3`	デバイスが指定された重大度とそれより上位の重大度のメッセージをモニタに記録できるようにします。小さい値は、より高い重大度を示します。重大度は 0 ～ 7 の範囲です。 • 0：緊急 • 1：アラート • 2：クリティカル • 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ 設定は Telnet および SSH セッションに適用されます。 重大度が指定されていない場合、デフォルトの 2 が使用されます。**no** オプションは、メッセージを Telnet および SSH セッションにログするデバイスの機能をディセーブルにします。
ステップ 6	(Optional) **show logging monitor** **Example:** `switch(config)# show logging monitor`	モニタ ロギング設定を表示します。
ステップ 7	[**no**] **logging message interface type ethernet description** **Example:**	システム メッセージ ログ内で、物理的なイーサネット インターフェイスおよびサブインターフェイスに対して説明を追加できるようにします。この説明

	Command or Action	Purpose
	`switch(config)# logging message interface type ethernet description`	は、インターフェイスで設定された説明と同じものです。 **no** オプションは、物理イーサネット インターフェイスのシステム メッセージ ログ内のインターフェイス説明の印刷をディセーブルにします。
ステップ 8	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

Syslog メッセージの送信元 ID の設定

リモート syslog サーバに送信される syslog メッセージにホスト名、IP アドレス、またはテキスト文字列を付加するように Cisco NX-OS を設定できます。

手順の概要

1. **configure terminal**
2. **logging origin-id** {**hostname** | **ip** *ip-address* | **string** *text-string*}
3. （任意） **show logging origin-id**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	必須: **logging origin-id** {**hostname**	**ip** *ip-address*	**string** *text-string*} 例： `switch(config)# logging origin-id string n9k-switch-abc`	リモート syslog サーバに送信される syslog メッセージに追加するホスト名、IPアドレス、またはテキスト文字列を指定します。

ステップ 3	（任意） **show logging origin-id**
	例：
	`switch(config)# show logging origin-id`
	`Logging origin_id : enabled (string: n9k-switch-abc)`
		リモート syslog サーバに送信される syslog メッセージに付加される、設定済みのホスト名、IP アドレス、またはテキスト文字列を表示します。

	コマンドまたはアクション	目的
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ファイルへのシステム メッセージの記録

システム メッセージをファイルに記録するようにデバイスを設定できます。デフォルトでは、システム メッセージは /logflash/log/*logfilename* に記録されます。

手順の概要

1. **configure terminal**
2. [**no**] **logging logfile** *logfile-name severity-level* [**persistent threshold** *percent* | **size** *bytes*]
3. **logging event** {**link-status** | **trunk-status**} {**enable** | **default**}
4. （任意） **show logging info**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	[**no**] **logging logfile** *logfile-name severity-level* [**persistent threshold** *percent* | **size** *bytes*]
	例：
	`switch(config)# logging logfile my_log 6`
	`switch(config)# logging logfile my_log 6 persistent threshold 90`
		非永続的または永続的なログファイルパラメータを設定します。
		logfile-name：システム メッセージの保存に使用するログ ファイルの名前を設定します。デフォルトのファイル名は「message」です。
		severity-level：ログに記録する最小の重大度レベルを設定します。小さい値は、より高い重大度を示します。デフォルトは 5 です。範囲は 0 ー 7 です。
		• 0：緊急
		• 1：アラート
		• 2：クリティカル

	コマンドまたはアクション	目的

		• 3：エラー
		• 4：警告
		• 5：通知
		• 6：情報
		• 7：デバッグ
		
		persistent threshold *percent*：オプションで、永続ログ ファイルのしきい値パーセンテージを設定します。範囲は 0 ～ 99 です。
		
		（注）
		persistent threshold を 0（ゼロ）に設定すると、永続しきい値機能が無効になり、しきい値 syslog は生成されません。
		
		percent は、永続ファイルのパーセントしきい値サイズを設定します。しきい値サイズに達すると、アラート通知メッセージがログに記録されます。永続ログファイルの使用率が 100％ に達すると、システムは別の syslog メッセージ通知を送信します。既存のログ ファイルのバックアップファイルが作成され、設定されたしきい値のパーセンテージが適用される、新しいログファイルへの書き込みが開始されます。最大で、新しい方から合計 5 つのバックアップファイルが保持されます。5ファイルを超えると、システムは最も古いものからファイルを削除します。
		
		（注）
		永続的ロギングは、システム対応の機能です。ログファイルは /logflash/log/[filename] にあります。
		
		次の show コマンドの出力は、永続ログ ファイル機能をサポートしています。
		• **show logging info**
		• **show logging**
		
		出力には、永続ログについての次のような情報が含まれます。
		`Logging logflash: enabled (Severity: notifications)(threshold percentage: 99)`
		`Logging logfile: enabled`

	コマンドまたはアクション	目的
		`Name - messages: Severity - notifications Size - 4194304` **size** *bytes*：オプションとして、最大ファイル サイズを指定します。範囲は 4096 ー 4194304 バイトです。
ステップ 3	**logging event** {**link-status** | **trunk-status**} {**enable** | **default**} 例： `switch(config)# logging event link-status default`	インターフェイス イベントをロギングします。 • **link-status**：すべての UP/DOWN メッセージおよびCHANGEメッセージをログに記録します。 • **trunk-status**：すべてのトランク ステータス メッセージをロギングします。 • **enable**：ポート レベルのコンフィギュレーションを上書きしてロギングをイネーブルにするよう、指定します。 • **default**：ロギングが明示的に設定されてないインターフェイスで、デフォルトのロギング設定を使用するよう、指定します。
ステップ 4	（任意） **show logging info** 例： `switch(config)# show logging info`	ロギング設定を表示します。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

モジュールおよびファシリティ メッセージのロギングの設定

モジュールおよびファシリティに基づいて記録するメッセージの重大度およびタイムスタンプの単位を設定できます。

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **logging module** [*severity-level*]
3. (Optional) **show logging module**
4. [**no**] **logging level** *facility severity-level*
5. (Optional) **show logging level** [*facility*]
6. (Optional) [**no**] **logging level** *ethpm*
7. [**no**] **logging timestamp** {**microseconds** |**milliseconds** |**seconds**}
8. (Optional) **show logging timestamp**

9. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	[**no**] **logging module** [*severity-level*] **Example:** `switch(config)# logging module 3`	指定された重大度またはそれ以上の重大度であるモジュール ログ メッセージをイネーブルにします。重大度は 0 ～ 7 の範囲です。 • 0：緊急 • 1：アラート • 2：クリティカル • 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ 重大度が指定されていない場合、デフォルトの 5 が使用されます。**no** オプションを使用すると、モジュール ログ メッセージがディセーブルになります。
ステップ 3	(Optional) **show logging module** **Example:** `switch(config)# show logging module`	モジュール ロギング設定を表示します。
ステップ 4	[**no**] **logging level** *facility severity-level* **Example:** `switch(config)# logging level aaa 2`	指定された重大度またはそれ以上の重大度である指定のファシリティからのロギング メッセージをイネーブルにします。重大度は 0 ～ 7 の範囲です。 • 0：緊急 • 1：アラート • 2：クリティカル

	Command or Action	Purpose
		• 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ 同じ重大度をすべてのファシリティに適用するには、**all** ファシリティを使用します。デフォルト値については、**show logging level** コマンドを参照してください。 **no** オプションを使用すると、指定されたファシリティのロギング重大度がデフォルトのレベルにリセットされます。ファシリティおよび重大度を指定しなかった場合、すべてのファシリティがそれぞれのデフォルト重大度にリセットされます。
ステップ 5	(Optional) **show logging level** [*facility*] **Example:** `switch(config)# show logging level aaa`	ファシリティごとに、ロギング レベル設定およびシステムのデフォルト レベルを表示します。ファシリティを指定しなかった場合は、すべてのファシリティのレベルが表示されます。
ステップ 6	(Optional) [**no**] **logging level** *ethpm* **Example:** **switch(config)# logging level ethpm ?** <0-7> 0-emerg;1-alert;2-crit;3-err;4-warn;5-notif;6-inform;7-debug link-down Configure logging level for link down syslog messages link-up Configure logging level for link up syslog messages **switch(config)#logging level ethpm link-down ?** error ERRORS notif NOTICE **(config)# logging level ethpm link-down error ?** <CR> **(config)# logging level ethpm link-down notif ?** <CR> **switch(config)#logging level ethpm link-up ?** error ERRORS notif NOTICE **(config)# logging level ethpm link-up error ?**	レベル 3 のイーサネット ポート マネージャ リンクアップ/リンクダウン syslog メッセージのロギングを有効にします。 **no** オプションを使用すると、イーサネット ポート マネージャの syslog メッセージにデフォルトのロギング レベルが使用されます。

	Command or Action	Purpose
	<CR> **(config)# logging level ethpm link-up notif ?** <CR>	
ステップ 7	[**no**] **logging timestamp** {**microseconds** |**milliseconds** |**seconds**} **Example:** `switch(config)# logging timestamp milliseconds`	ロギング タイムスタンプ単位を設定します。デフォルトでは、単位は秒です。 **Note** このコマンドは、スイッチ内で保持されているログに適用されます。また、外部のロギング サーバには適用されません。
ステップ 8	(Optional) **show logging timestamp** **Example:** `switch(config)# show logging timestamp`	設定されたロギング タイムスタンプ単位を表示します。
ステップ 9	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

syslog サーバの設定

Note シスコは、管理仮想ルーティングおよび転送（VRF）インスタンスを使用するサーバとして、syslog サーバを設定することを推奨します。VRF の詳細情報については、『Cisco Nexus 9000 シリーズ NX-OS ユニキャスト ルーティング設定ガイド』を参照してください。

システム メッセージを記録する、リモート システムを参照する syslog サーバーを最大で 8 台設定できます。

Note Cisco NX-OS リリース 10.3(2)F までは、ユーザーが特定のデフォルト値を入力すると、logging server コマンドの実行中の構成にそれらのデフォルト値がランダムまたは一貫性なく表示されていました。ただし、Cisco NX-OS リリース 10.3(2)F 以降では、実行中の構成には常にデフォルト以外の値のみが表示されます。

たとえば、以前のリリースで、特定のユーザー入力に対し、実行中の構成が `logging server 1.1.1.1 port 514 facility local7 use-vrf default` という値を表示していたような場合、Cisco NX-OS リリース 10.3(2)F 以降では、同じ入力に対し、実行中の構成は `logging server 1.1.1.1` という値のみを表示します。デフォルト ポート、デフォルト ファシリティ（local7）、デフォルト VRF などのデフォルト値が実行中の構成で表示されないことに注意してください。

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **logging server** *host* [*severity-level* [**use-vrf** *vrf-name*]]
3. **logging source-interface loopback** *virtual-interface*
4. (Optional) **show logging server**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	[**no**] **logging server** *host* [*severity-level* [**use-vrf** *vrf-name*]]
	Example:
	`switch(config)# logging server 192.0.2.253`
	Example:
	`switch(config)# logging server 2001::3 5 use-vrf red`
		指定されたホスト名、IPv4 または IPv6 アドレスで Syslog サーバーを構成します。**use-vrf** キーワードを使用すると、メッセージ ロギングを VRF の特定の Syslog サーバーに限定できます。 **use-vrf** *vrf-name* キーワードは、VRF 名のデフォルトまたは管理値を示します。デフォルト VRF は、デフォルトで管理 VRF です。ただし、**show-running** コマンドはデフォルトの VRF をリストしません。重大度は 0 ～ 7 の範囲です。
		• 0：緊急
		• 1：アラート
		• 2：クリティカル
		• 3：エラー
		• 4：警告
		• 5：通知
		• 6：情報
		• 7：デバッグ
		デフォルトの発信ファシリティは local7 です。
		no オプションは、指定したホストのロギング サーバを削除します。
		この最初の例では、ファシリティ local 7 のすべてのメッセージを転送します。2 番目の例では、重大度

	Command or Action	Purpose
		が 5 以下のメッセージを、VRF red の指定された IPv6 アドレスに転送します。
ステップ 3	Required: **logging source-interface loopback** *virtual-interface* **Example:** `switch(config)# logging source-interface loopback 5`	リモート Syslog サーバの送信元インターフェイスをイネーブルにします。 *virtual-interface* 引数の範囲は 0 ～ 1023 です。
ステップ 4	(Optional) **show logging server** **Example:** `switch(config)# show logging server`	Syslog サーバ設定を表示します。
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

セキュアな Syslog サーバの設定

手順の概要

1. **configure terminal**
2. [**no**] **logging server** *host* [*severity-level* [**port** *port-number*][**secure**[**trustpoint client-identity** *trustpoint-name*]][**use-vrf** *vrf-name*]]
3. （任意） **logging source-interface** *interface name*
4. （任意） **show logging server**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	[**no**] **logging server** *host* [*severity-level* [**port** *port-number*][**secure**[**trustpoint client-identity** *trustpoint-name*]][**use-vrf** *vrf-name*]] 例：	指定されたホスト名、あるいは IPv4 または IPv6 アドレスで Syslog サーバを設定します。必要に応じて、CA によって署名されるクライアント アイデンティティ証明書をインストールし、trustpoint

	コマンドまたはアクション	目的

	`switch(config)# logging server 192.0.2.253 secure`
	例：
	`switch(config)# logging server 2001::3 5 secure trustpoint client-identity myCA use-vrf red`
		client-identity オプションを使用することで相互認証を適用できます。
		セキュアなTLS接続のデフォルト宛先ポートは6514です。

ステップ 3	（任意） **logging source-interface** *interface name* 例： `switch(config)# logging source-interface lo0`	リモートSyslogサーバの送信元インターフェイスをイネーブルにします。
ステップ 4	（任意） **show logging server** 例： `switch(config)# show logging server`	Syslog サーバ設定を表示します。secure オプションを設定する場合、出力のエントリにトランスポート情報が含まれるようになります。デフォルトでは、secure オプションが設定されていない場合、トランスポートは UDP です。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

CA 証明書の設定

セキュアな Syslog 機能のサポートには、トラストポイントの設定によってリモート サーバを認証する必要があります。

手順の概要

1. **configure terminal**
2. [**no**] **crypto ca trustpoint** *trustpoint-name*
3. **crypto ca authenticate** *trustpoint-name*
4. （任意） **show crypto ca certificate**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

	コマンドまたはアクション	目的

ステップ 2	[**no**] **crypto ca trustpoint** *trustpoint-name*
	例：
	`switch(config)# crypto ca trustpoint winca`
	`switch(config-trustpoint)#`
		トラストポイントを設定します。
		（注）
		トラストポイントの設定の前に ip domain-name を設定する必要があります。

ステップ 3	必須: **crypto ca authenticate** *trustpoint-name* 例： `switch(config-trustpoint)# crypto ca authenticate winca`	トラストポイントの CA 証明書を設定します。
ステップ 4	（任意） **show crypto ca certificate** 例： `switch(config)# show crypto ca certificates`	設定されている証明書/チェーンと、関連付けられているトラストポイントを表示します。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	デバイスのリロード後にトラストポイントが持続されるように、実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーします。

CA 証明書の登録

NX-OS スイッチ（クライアント）が識別するようリモートサーバによって要求される相互認証では、ピア認証が必須であるため、これは証明書をスイッチに登録するための追加設定です。

手順の概要

1. **configure terminal**
2. **crypto key generate rsa label** *key name* **exportable modules 2048**
3. [**no**] **crypto ca trustpoint** *trustpoint-name*
4. **rsakeypair** *key-name*
5. **crypto ca trustpoint** *trustpoint-name*
6. [**no**] **crypto ca enroll** *trustpoint-name*
7. **crypto ca import** *trustpoint-name* **certificate**
8. （任意） **show crypto ca certificates**
9. **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	必須: **crypto key generate rsa label** *key name* **exportable modules 2048** 例： `switch(config-trustpoint)# crypto key generate rsa label myKey exportable modulus 2048`	RSA キーペアを設定します。デフォルトでは、Cisco NX-OS ソフトウェアは 1024 ビットの RSA キーを作成します。

ステップ 3	[**no**] **crypto ca trustpoint** *trustpoint-name*
	例：
	`switch(config)# crypto ca trustpoint myCA`
	`switch(config-trustpoint)#`
		トラストポイントを設定します。
		（注）
		トラストポイントの設定の前に ip domain-name を設定する必要があります。

ステップ 4	必須: **rsakeypair** *key-name* 例： `switch(config-trustpoint)# rsakeypair myKey`	トラストポイント CA に生成されたキーペアを関連付けます。
ステップ 5	**crypto ca trustpoint** *trustpoint-name* 例： `switch(config)# crypto ca authenticate myCA`	トラストポイントの CA 証明書を設定します。
ステップ 6	[**no**] **crypto ca enroll** *trustpoint-name* 例： `switch(config)# crypto ca enroll myCA`	CA に登録するスイッチのアイデンティティ証明書を生成します。
ステップ 7	**crypto ca import** *trustpoint-name* **certificate** 例： `switch(config-trustpoint)# crypto ca import myCA certificate`	CA によって署名されたアイデンティティ証明書をスイッチにインポートします。
ステップ 8	（任意） **show crypto ca certificates** 例： `switch# show crypto ca certificates`	設定されている証明書またはチェーンと、関連付けられているトラストポイントを表示します。
ステップ 9	必須: **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

	コマンドまたはアクション	目的

UNIX または Linux システムでの syslog サーバの設定

/etc/syslog.conf ファイルに次の行を追加して、UNIX または Linux システム上に syslog サーバを設定できます。

```
facility.level <five tab characters> action
```

次の表に、設定可能な syslog フィールドを示します。

表 *10 : syslog.conf* の *syslog* フィールド

フィールド	説明
Facility	メッセージの作成者。auth、authpriv、cron、daemon、kern、lpr、mail、mark、news、syslog、user、local0 ～ local7 です。アスタリスク（*）を使用するとすべてを指定します。これらのファシリティ指定により、発信元に基づいてメッセージの宛先を制御できます。 （注） ローカル ファシリティを使用する前に設定をチェックします。
Level	メッセージを記録する最小重大度。debug、info、notice、warning、err、crit、alert、emerg です。アスタリスク（*）を使用するとすべてを指定します。none を使用するとファシリティをディセーブルにできます。
Action	メッセージの宛先。ファイル名、前に @ 記号を加えたホスト名、ユーザをカンマで区切ったリスト、またはすべてのログイン ユーザを表すアスタリスク（*）を使用できます。

手順の概要

1. /etc/syslog.conf ファイルに次の行を追加して、ファイル /var/log/myfile.log に local7 ファシリティのデバッグ メッセージを記録します。
2. シェル プロンプトで次のコマンドを入力して、ログ ファイルを作成します。
3. 次のコマンドを入力して、システム メッセージ ロギング デーモンが myfile.log をチェックして、新しい変更を取得するようにします。

手順の詳細

手順

ステップ 1 /etc/syslog.conf ファイルに次の行を追加して、ファイル /var/log/myfile.log に local7 ファシリティのデバッグ メッセージを記録します。

例：

```
debug.local7 var/log/myfile.log
```

ステップ 2 シェル プロンプトで次のコマンドを入力して、ログ ファイルを作成します。

例：

```
$ touch /var/log/myfile.log
$ chmod 666 /var/log/myfile.log
```

ステップ 3 次のコマンドを入力して、システム メッセージ ロギング デーモンが myfile.log をチェックして、新しい変更を取得するようにします。

例：

```
$ kill -HUP ~cat /etc/syslog.pid~
```

ログ ファイルの表示およびクリア

ログ ファイルおよび NVRAM のメッセージを表示したり消去したりできます。

SUMMARY STEPS

1. **show logging last** *number-lines*
2. **show logging logfile duration** *hh:mm:ss*
3. **show logging logfile last-index**
4. **show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]
5. **show logging logfile** [**start-seqn** *number*] [**end-seqn** *number*]
6. **show logging nvram** [**last** *number-lines*]
7. **clear logging logfile** [**persistent**]
8. **clear logging nvram**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	Required: **show logging last** *number-lines* **Example:** `switch# show logging last 40`	ロギング ファイルの最終行番号を表示します。最終行番号には 1 ～ 9999 を指定できます。
ステップ 2	**show logging logfile duration** *hh:mm:ss* **Example:** `switch# show logging logfile duration 15:10:0`	入力された時間内のタイム スタンプを持つログ ファイルのメッセージを表示します。
ステップ 3	**show logging logfile last-index** **Example:** `switch# show logging logfile last-index`	ログファイルの最後のメッセージのシーケンス番号を表示します。
ステップ 4	**show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*] **Example:** `switch# show logging logfile start-time 2013 oct 1 15:10:0`	入力されたスパン内にタイム スタンプがあるログ ファイルのメッセージを表示します。終了時間を入力しないと、現在の時間が使用されます。月の時間フィールドには 3 文字を、年と日の時間フィールドには数値を入力します。
ステップ 5	**show logging logfile** [**start-seqn** *number*] [**end-seqn** *number*] **Example:** `switch# show logging logfile start-seqn 100 end-seqn 400`	シーケンス番号の範囲内である、発生したメッセージを表示します。終了シーケンス番号を指定しなかった場合は、ログ ファイルの、開始番号から最後のメッセージまでのメッセージが表示されます。
ステップ 6	**show logging nvram** [**last** *number-lines*] **Example:** `switch# show logging nvram last 10`	NVRAM のメッセージを表示します。表示される行数を制限するには、表示する最終行番号を入力できます。最終行番号には 1 ～ 100 を指定できます。
ステップ 7	**clear logging logfile** [**persistent**] **Example:** `switch# clear logging logfile`	ログ ファイルの内容をクリアします。 **persistent**：永続的な場所から、ログファイルの内容をクリアします。
ステップ 8	**clear logging nvram** **Example:** `switch# clear logging nvram`	NVRAM の記録されたメッセージをクリアします。

システム メッセージ ロギングの設定確認

システム メッセージ ロギングの設定情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show logging console	コンソール ロギング設定を表示します。
show logging info	ロギング設定を表示します。
show logging last *number-lines*	ログ ファイルの末尾から指定行数を表示します。
show logging level [*facility*]	ファシリティ ロギング重大度設定を表示します。
show logging logfile duration *hh:mm:ss*	入力された時間内のタイム スタンプを持つログ ファイルのメッセージを表示します。
show logging logfile last-index	ログファイルの最後のメッセージのシーケンス番号を表示します。
show logging logfile [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]	開始日時と終了日時に基づいてログファイルのメッセージを表示します。
show logging logfile [**start-seqn** *number*] [**end-seqn** *number*]	シーケンス番号の範囲内である、発生したメッセージを表示します。終了シーケンス番号を指定しなかった場合は、ログ ファイルの、開始番号から最後のメッセージまでのメッセージが表示されます。
show logging module	モジュール ロギング設定を表示します。
show logging monitor	モニタ ロギング設定を表示します。
show logging nvram [**last** *number-lines*]	NVRAM ログのメッセージを表示します。
show logging server	Syslog サーバ設定を表示します。
show logging timestamp	ロギング タイムスタンプ単位設定を表示します。

繰り返されるシステム ロギング メッセージ

システム プロセスはロギング メッセージを生成します。生成される重大度レベルを制御するために使用されるフィルタによっては、多数のメッセージが生成され、その多くが繰り返されます。

ロギング メッセージの量を管理するスクリプトの開発を容易にし、**show logging log** コマンドの出力の「フラッディング」から繰り返されるメッセージを排除するために、繰り返されるメッセージをロギングする次の方法が使用されます。

以前の方法では、同じメッセージが繰り返された場合、デフォルトでは、メッセージ内でメッセージが再発生した回数が示されていました。

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
```

```
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting
Port
Identity is 00:1c:73:ff:ff:ee:f6:e5
2019 Mar 11 13:43:15 Cisco-customer last message repeated 242 times
```

新しいメソッドは、繰り返しメッセージの最後に繰り返し回数を追加するだけです。

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting
Port
Identity is 00:1c:73:ff:ff:ee:f6:e5

2019 Mar 11 13:43:15 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting
Port
Identity is 00:1c:73:ff:ff:ee:f6:e5 (message repeated 242 times)
```

システム メッセージ ロギングの設定例

システム メッセージ ロギングのコンフィギュレーション例を示します。

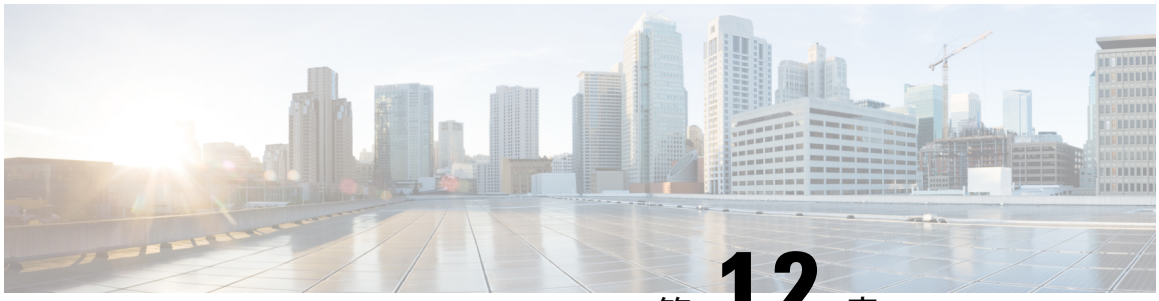
```
configure terminal
 logging console 3
 logging monitor 3
 logging logfile my_log 6
 logging module 3
 logging level aaa 2
 logging timestamp milliseconds
 logging server 172.28.254.253
 logging server 172.28.254.254 5 facility local3
 copy running-config startup-config
```

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
システム メッセージ	『*Cisco NX-OS System Messages Reference*』

関連資料



第 12 章

Smart Call Home の設定

この章では、Cisco NX-OS デバイスの Smart Call Home 機能を設定する方法について説明します。

この章は、次の内容で構成されています。

- Smart Call Home の概要, on page 203
- Smart Call Home - 概念 （204 ページ）
- Smart Call Home の前提条件, on page 211
- Smart Call Home の注意事項および制約事項 （211 ページ）
- Smart Call Home のデフォルト設定, on page 211
- Smart Call Home の設定 （212 ページ）
- Smart Call Home 設定の確認, on page 234
- Smart Call Home の設定例 （234 ページ）
- その他の参考資料 （236 ページ）

Smart Call Home の概要

Smart Call Home により、重要なシステム ポリシーについて電子メールベースの通知が提供されます。豊富なメッセージ フォーマットから選択できるので、ポケットベル サービス、標準 E メール、または XML ベースの自動解析アプリケーションとの最適な互換性が得られます。この機能を使用して、ネットワーク サポート エンジニアやネットワーク オペレーション センターを呼び出せます。また、Cisco Smart Call Home サービスを使用して、TAC でケースを自動的に生成することもできます。

Smart Call Home には、次の機能があります。

- 関連する CLI コマンド出力の実行および添付が自動化されます。
- 次のような、複数のメッセージ フォーマット オプションがあります。
  - ショート テキスト：ポケットベルまたは印刷形式のレポートに最適。
  - フルテキスト：人間が判読しやすいように完全にフォーマットされたメッセージ情報です。

- XML：Extensible Markup Language（XML）および Adaptive Messaging Language（AML）XML Schema Definition（XSD）を使用する、調和の取れた判読可能なフォーマット。AML XSD は Cisco.com の Web サイトで公開されています。XML 形式は、Technical Assistance Center とのやり取りの中でも使用されます。

- 複数のメッセージ宛先への同時配信が可能。各宛先プロファイルには最大 50 件の電子メール宛先アドレスを設定できます。

Smart Call Home - 概念

このセクションでは、Smart Call Home に関連するいくつかの概念について説明します。

宛先プロファイル

宛先プロファイルには、次の情報が含まれます。

- 1 つ以上のアラート グループ：アラートの発生時に、特定の Smart Call Home メッセージを送信するアラートのグループ。
- 1 つまたは複数の電子メール宛先：この宛先プロファイルに割り当てられたアラート グループによって生成された Smart Call Home メッセージの受信者リスト。
- メッセージ フォーマット：Smart Call Home メッセージのフォーマット（ショート テキスト、フル テキスト、または XML）。
- メッセージ重大度：Cisco NX-OS が宛先プロファイル内のすべての電子メール アドレスに対して Smart Call Home メッセージを生成するまで、アラートが満たす必要がある Smart Call Home 重大度。アラートの Smart Call Home 重大度が、宛先プロファイルに設定されたメッセージ重大度よりも低い場合、Cisco NX-OS はアラートを生成しません。

定期メッセージを日別、週別、月別で送信するコンポーネント アラート グループを使用して、定期的なコンポーネント アップデート メッセージを許可するよう宛先プロファイルを設定することもできます。

Cisco NX-OS は、次の定義済み宛先プロファイルをサポートします。

- CiscoTAC-1：XML メッセージ フォーマットの Cisco-TAC アラート グループをサポートします。このプロファイルは、callhome@cisco.com という E メール コンタクト、最大メッセージサイズ、およびメッセージ重大度 0 で設定済みです。このプロファイルのデフォルト情報はどれも変更できません。
- full-text-destination：フル テキスト メッセージ フォーマットをサポートします。
- short-text-destination：ショート テキスト メッセージ フォーマットをサポートします。

Smart Call Home アラート グループ

アラート グループは、すべての Cisco Nexus デバイスでサポートされる Smart Call Home アラートの定義済みサブセットです。アラート グループを使用すると、定義済みまたはカスタム宛先プロファイルに送信する一連の Smart Call Home アラートを選択できます。Smart Call Home アラートが宛先プロファイルにアソシエートされたいずれかのアラート グループに属する場合、およびアラートで、Smart Call Home メッセージ重大度が宛先プロファイルに設定されているメッセージ重大度と同じか、それ以上である場合のみ、デバイスは Smart Call Home アラートを宛先プロファイルの電子メールの宛先に送信します。

次の表に、サポートされるアラート グループと、アラート グループ用に生成された Smart Call Home メッセージに含まれるデフォルトの CLI コマンド出力を示します。

Table 11: アラート グループおよび実行されるコマンド

アラート グループ	説明	実行されるコマンド
Cisco-TAC	Smart Call Home 宛ての、他のアラート グループからのすべてのクリティカル アラート。	アラートを発信するアラート グループに基づいてコマンドを実行します。
設定	設定に関連した定期的なイベント。	**show module** **show version**
診断	診断によって生成されたイベント。	**show diagnostic result module all detail** **show diagnostic result module** *number* **detail** **show hardware** **show logging last 200** **show module** **show sprom all** **show tech-support gold** **show tech-support ha** **show tech-support platform** **show version**

アラート グループ	説明	実行されるコマンド
EEM	EEMによって生成されるイベント	**show diagnostic result module all detail** **show diagnostic result module** *number* **detail** **show module** **show tech-support gold** **show tech-support ha** **show tech-support platform**
環境	電源、ファン、および温度アラームなどの環境検知要素に関連するイベント。	**show environment** **show logging last 200** **show module** **show version**
インベントリ	装置がコールドブートした場合、またはFRUの取り付けまたは取り外しを行った場合に示されるコンポーネントステータス。このアラートは重要でないイベントであり、情報はステータスおよび使用権に使用されます。	**show inventory** **show license usage** **show module** **show sprom all** **show system uptime** **show version**
ライセンス	ライセンスおよびライセンス違反に関連するイベント	**show logging last 200**

アラート グループ	説明	実行されるコマンド
ラインカード ハードウェア	標準またはインテリジェント スイッチング モジュールに関連するイベント。	**show diagnostic result module all detail** **show diagnostic result module** *number* **detail** **show hardware** **show logging last 200** **show module** **show sprom all** **show tech-support ethpm** **show tech-support gold** **show tech-support ha** **show tech-support platform** **show version**
スーパーバイザ ハードウェア	スーパーバイザ モジュールに関連するイベント。	**show diagnostic result module all detail** **show hardware** **show logging last 200** **show module** **show sprom all** **show tech-support ethpm** **show tech-support gold** **show tech-support ha** **show tech-support platform** **show version**
Syslog port group	syslog PORT ファシリティによって生成されるイベント	**show license usage** **show logging last 200**

アラート グループ	説明	実行されるコマンド
システム	装置の動作に必要なソフトウェア システムの障害によって生成されたイベント。	**show diagnostic result module all detail** **show hardware** **show logging last 200** **show module** **show sprom all** **show tech-support ethpm** **show tech-support gold** **show tech-support ha** **show tech-support platform**
テスト	ユーザが作成したテスト メッセージ	**show module** **show version**

Smart Call Home は、syslog の重大度を、syslog ポート グループ メッセージの対応する Smart Call Home の重大度に対応させます。

特定のイベントが発生し、Smart Call Home メッセージを含む **show** 出力を送信した場合に、追加の **show** コマンドを実行するために、定義済みのアラート グループをカスタマイズできます。

show コマンドは、フル テキストおよび XML 宛先プロファイルにのみ追加できます。ショート テキスト宛先プロファイルは、128 バイトのテキストに制限されているため、追加の **show** コマンドをサポートしていません。

Smart Call Home のメッセージ レベル

Smart Call Home を使用すると、緊急度に基づいてメッセージをフィルタリングできます。各定義済みまたはユーザ定義宛先プロファイルを、0（最小緊急度）～9（最大緊急度）までの Smart Call Home しきい値と関連付けることができます。デフォルトは 0（全メッセージを送信）です。

syslog 重大度は、Smart Call Home メッセージ レベルにマッピングされています。

Note Smart Call Home と Syslog は異なる重大度を使用します（次の表を参照）。Smart Call Home は、メッセージ テキストで syslog メッセージ レベルを変更しません。

次の表に、各 Smart Call Home メッセージ レベルのキーワードと、syslog ポート アラート グループの対応する syslog レベルを示します。

Table 12: 重大度と *syslog* レベルのマッピング

Smart Call Home レベル	キーワード	Syslog レベル	説明
9	Catastrophic	該当なし	ネットワーク全体に壊滅的な障害が発生しています。
8	Disaster	該当なし	ネットワークに重大な影響が及びます。
7	Fatal	緊急（0）	システムが使用不可能な状態。
6	Critical	アラート（1）	クリティカルな状況で、すぐに対応する必要があります。
5	Major	重要（2）	重大な状態。
4	Minor	エラー（3）	軽微な状態。
3	警告	警告（4）	警告状態。
2	通知	通知（5）	基本的な通知および情報メッセージです。他と関係しない、重要性の低い障害です。
1	標準	情報（6）	標準状態に戻ることを示す標準イベントです。
0	Debugging	デバッグ（7）	デバッグ メッセージ。

Smart Call Home の取得

シスコと直接サービス契約を結んでいる場合は、Smart Call Home サービスに登録できます。Smart Call Home は、Smart Call Home メッセージを分析し、背景説明と推奨措置を提供します。既知の問題、特にオンライン診断障害については、TAC に Automatic Service Request が作成されます。

Smart Call Home には、次の機能があります。

- 継続的なデバイス ヘルス モニタリングとリアルタイムの診断アラート。
- Smart Call Home メッセージの分析。必要に応じて、自動サービス要求（詳細な診断情報が含まれる）が作成され、該当する TAC チームにルーティングされるため、問題解決を高速化できます。
- セキュアなメッセージ転送が、ご使用のデバイスから直接、または HTTP プロキシサーバやダウンロード可能な転送ゲートウェイ（TG）を経由して行われます。TG 集約ポイントは、複数のデバイスをサポートする場合またはセキュリティ要件によって、デバイスをインターネットに直接接続できない場合に使用できます。

- あらゆる Smart Call Home デバイスの Smart Call Home メッセージおよび推奨事項、インベントリ情報、設定情報への Web アクセス。この機能によって、関連するフィールドの注意事項、セキュリティ勧告、および廃止情報にアクセスできます。

登録には次の情報が必要です。

- デバイスの SMARTnet 契約番号
- 電子メール アドレス
- お使いの Cisco.com ID

データベース マージの注意事項

2 つの Smart Call Home データベースをマージする場合は、次の注意事項に従ってください。

- マージされるデータベースには、次の情報が含まれます。
  - マージ側デバイスからの全宛先プロファイルのスーパーセット。
  - 宛先プロファイルの E メール アドレスとアラート グループ。
  - マージ側デバイスにあるその他の設定情報（メッセージスロットリング、定期的なインベントリなど）。
- 宛先プロファイル名は、マージするデバイス内で重複しないようにしてください。コンフィギュレーションが異なっても、同じ名前は使用できません。プロファイル名が重複している場合、重複するプロファイルの 1 つを削除する必要があります。そうしなければマージ処理が失敗します。

高可用性

ステートフルおよびステートレスの両方のリスタートが、Smart Call Home でサポートされます。

仮想化のサポート

Smart Call Home のインスタンスが 1 つサポートされます。次の URL から、Smart Call Home の Web サイトでお客様の連絡先を登録できます。https://supportforums.cisco.com/community/netpro/solutions/smart_services/smartcallhome

callhome send および **callhome test** コマンドを使用して Smart Call Home をテストできます。

Smart Call Home は Virtual Routing and Forwarding（VRF）を認識します。特定の VRF を使用して Smart Call Home SMTP サーバに接続するように Smart Call Home を設定できます。

Smart Call Home の前提条件

Smart Call Home には、次の前提条件があります。

- 電子メール アドレスにメッセージを送信するには、まず電子メール サーバを設定する必要があります。HTTP を使用してメッセージを送信するには、HTTPS サーバにアクセスでき、Cisco Nexus デバイスに有効な証明書がインストールされている必要があります。
- デバイスは電子メール サーバまたは HTTPS サーバと IP 接続している必要があります。
- まず、コンタクト名（SNMP サーバのコンタクト）、電話番号、および住所情報を設定する必要があります。この手順は、受信メッセージの送信元を判別するために必要です。
- Smart Call Home サービスを使用する場合、設定中のデバイスに対応している現在のサービス契約が必要です。

Smart Call Home の注意事項および制約事項

Smart Call Home には、次の注意事項および制限事項があります。

- IP 接続がない場合、またはプロファイル宛先への仮想ルーティングおよびフォワーディング（VRF）インスタンス内のインターフェイスがダウンしている場合、デバイスは Smart Call Home メッセージを送信できません。
- Smart Call Home はあらゆる SMTP サーバで動作します。
- Smart Call Home には最大 5 個までの SMTP サーバを設定できます。
- Link up/down syslog メッセージは、Smart Call Home メッセージまたはアラート通知をトリガーしません。
- 住所、顧客 ID、サイト ID などの Smart Call Home コマンドを設定する場合は、これらのコマンドをセミコロン区切りでグループ化するのではなく、個別のコマンドとして設定する必要があります。
- Cisco NX-OS リリース 10.2(3)F 以降、SMTP-AUTH は、Cisco Nexus 9000 シリーズ プラットフォーム スイッチでのセキュアな Call Home メール転送でサポートされています。
- Callhome は、**ip http source-interface** コマンドを使用した送信元インターフェイスの指定をサポートしていません。callhome で **source-interface** コマンドを構成して、送信元インターフェイスを指定することができます。

Smart Call Home のデフォルト設定

このテーブルは、Smart Call Home パラメータのデフォルト設定を示します。

Table 13: デフォルトの *Smart Call Home* パラメータ

パラメータ	デフォルト
フルテキスト フォーマットで送信するメッセージの宛先メッセージ サイズ	2,500,000
XML フォーマットで送信するメッセージの宛先メッセージ サイズ	2,500,000
ショートテキスト フォーマットで送信するメッセージの宛先メッセージ サイズ	4000
ポートを指定しなかった場合の SMTP サーバ ポート	25
プライオリティを指定しなかった場合の SMTP サーバのプライオリティ	50
プロファイルとアラート グループのアソシエート	フルテキスト宛先プロファイルおよびショートテキスト宛先プロファイルの場合はすべて。CiscoTAC-1 宛先プロファイルの場合は cisco-tac アラート グループ
フォーマット タイプ	XML
Smart Call Home のメッセージ レベル	0（ゼロ）
HTTP プロキシ サーバの使用	無効であり、プロキシ サーバは設定されていません。

Smart Call Home の設定

（注） Cisco NX-OS コマンドは Cisco IOS コマンドと異なる場合があるので注意してください。

次の順序で Smart Call Home 設定を行うことを推奨します。

1. 連絡先情報の設定 （213 ページ）
2. 宛先プロファイルの作成 （215 ページ）
3. アラート グループと宛先プロファイルのアソシエート （219 ページ）
4. （オプション）

   アラート グループへの show コマンドの追加 （220 ページ）

5. Smart Call Home のイネーブル化またはディセーブル化 （228 ページ）
6. （任意）
   送信元インターフェイスの構成 （232 ページ）
7. （オプション）
   Smart Call Home 設定のテスト （233 ページ）

連絡先情報の設定

Smart Call Home には、電子メール、電話番号、住所の各情報を指定する必要があります。契約 ID、カスタマー ID、サイト ID、およびスイッチ プライオリティ情報を任意で指定できます。

これらの Smart Call Home コマンドは、セミコロン区切りでグループ化するのではなく、個別のコマンドとして設定する必要があります。

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server contact** *sys-contact*
3. **callhome**
4. **email-contact** *email-address*
5. **phone-contact** *international-phone-number*
6. **streetaddress** *address*
7. (Optional) **contract-id** *contract-number*
8. (Optional) **customer-id** *customer-number*
9. (Optional) **site-id** *site-number*
10. (Optional) **switch-priority** *number*
11. **commit**
12. (Optional) **show callhome**
13. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**snmp-server contact** *sys-contact* **Example:**	SNMP sysContact を設定します。

	Command or Action	Purpose
	`switch(config)# snmp-server contact personname@companyname.com`	

ステップ 3	**callhome**
	Example:
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart Call Home コンフィギュレーション モードを開始します。

ステップ 4	**email-contact** *email-address* **Example:** `switch(config-callhome)# email-contact admin@Mycompany.com`	デバイスの主要責任者の電子メール アドレスを設定します。 *email-address* には、電子メールアドレスの形式で、最大 255 の英数字を使用できます。 **Note** 任意の有効な電子メール アドレスを使用できます。アドレスには、空白を含めることはできません。
ステップ 5	**phone-contact** *international-phone-number* **Example:** `switch(config-callhome)# phone-contact +1-800-123-4567`	デバイスの担当者の電話番号を国際電話フォーマットで設定します。*international-phone-number* は、最大 17 文字の英数字で、国際電話フォーマットにする必要があります。 **Note** 電話番号には、空白を含めることはできません。番号の前にプラス（+）プレフィックスを使用します。
ステップ 6	**streetaddress** *address* **Example:** `switch(config-callhome)# streetaddress 123 Anystreet st. Anytown,AnyWhere`	デバイスの主要責任者の住所を空白の含まれる英数字ストリングとして設定します。 *address* には、最大 255 の英数字を使用できます。スペースを使用できます。
ステップ 7	(Optional) **contract-id** *contract-number* **Example:** `switch(config-callhome)# contract-id Contract5678`	サービス契約からこのデバイスの契約番号を設定します。 契約番号は、最大255文字の英数字を自由なフォーマットで指定できます。
ステップ 8	(Optional) **customer-id** *customer-number* **Example:** `switch(config-callhome)# customer-id Customer123456`	サービス契約からこのデバイスのカスタマー番号を設定します。 カスタマー番号は、最大255文字の英数字を自由なフォーマットで指定できます。
ステップ 9	(Optional) **site-id** *site-number* **Example:**	このデバイスのサイト番号を設定します。

	Command or Action	Purpose
	`switch(config-callhome)# site-id Site1`	*site-number* は、最大 255 文字の英数字を自由なフォーマットで指定できます。
ステップ 10	(Optional) **switch-priority** *number* **Example:** `switch(config-callhome)# switch-priority 3`	このデバイスのスイッチ プライオリティを設定します。 指定できる範囲は 0 ～ 7 です。0 は最高のプライオリティを、7は最低のプライオリティを示します。デフォルト値は 7 です。
ステップ 11	**commit** **Example:** `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 12	(Optional) **show callhome** **Example:** `switch(config-callhome)# show callhome`	Smart Call Home コンフィギュレーションの概要を表示します。
ステップ 13	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

What to do next

宛先プロファイルを作成します。

宛先プロファイルの作成

ユーザ定義宛先プロファイルを作成し、メッセージ フォーマットを設定できます。

手順の概要

1. **configure terminal**
2. **callhome**
3. **destination-profile** *name*
4. **destination-profile** *name* **format** {**XML** | **full-txt** | **short-txt**}
5. **commit**
6. （任意） **show callhome destination-profile** [**profile** *name*]
7. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**callhome**
	例：
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**destination-profile** *name* 例： `switch(config-callhome)# destination-profile Noc101`	新しい宛先プロファイルを作成します。名前は、最大 31 文字の英数字で指定できます。
ステップ 4	**destination-profile** *name* **format** {**XML** | **full-txt** | **short-txt**} 例： `switch(config-callhome)# destination-profile Noc101 format full-txt`	プロファイルのメッセージフォーマットを設定します。名前は、最大31 文字の英数字で指定できます。
ステップ 5	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 6	（任意） **show callhome destination-profile** [**profile** *name*] 例： `switch(config-callhome)# show callhome destination-profile profile Noc101`	1 つまたは複数の宛先プロファイルに関する情報を表示します。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

次のタスク

1 つの宛先プロファイルに 1 つまたは複数のアラート グループを関連付けます。

宛先プロファイルの変更

定義済みまたはユーザ定義の宛先プロファイルの次の属性を変更できます。

- 宛先メール アドレス：アラートの送信先となる実際のアドレス（トランスポート メカニズムに関係します）。
- 宛先 URL：アラートの送信先となる HTTP または HTTPS URL。
- 転送方式：E メールまたは HTTP 転送によって、使用される宛先アドレスのタイプが決まります。
- メッセージ フォーマット：アラート送信に使用されるメッセージ フォーマット（フル テキスト、ショート テキスト、または XML）。
- メッセージ レベル：この宛先プロファイルの Smart Call Home メッセージの重大度。
- メッセージ サイズ：この宛先プロファイルの E メール アドレスに送信された Smart Call Home メッセージの長さ。

SUMMARY STEPS

1. **configure terminal**
2. **callhome**
3. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **email-addr** *address*
4. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **http** *address*
5. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **transport-method** {**email** | **http**}
6. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-level** *number*
7. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-size** *number*
8. **commit**
9. (Optional) **show callhome destination-profile** [**profile** *name*]
10. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

	Command or Action	Purpose

ステップ 2	**callhome**
	Example:
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **email-addr** *address* **Example:** `switch(config-callhome)# destination-profile full-txt-destination email-addr person@place.com`	ユーザ定義または定義済みの宛先プロファイルに E メール アドレスを設定します。宛先プロファイルには、最大 50 個の電子メール アドレスを設定できます。
ステップ 4	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **http** *address* **Example:** `switch(config-callhome)# destination-profile CiscoTAC-1 http https://tools.cisco.com/its/service/oddce/services/DDCEService`	ユーザ定義または定義済み宛先プロファイルの HTTP または HTTPS URL を設定します。URL の最大文字数は 255 文字です。
ステップ 5	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **transport-method** {**email** | **http**} **Example:** `switch(config-callhome)# destination-profile CiscoTAC-1 transport-method http`	ユーザ定義または定義済み宛先プロファイルに対応する電子メールまたは HTTP 転送方式を設定します。選択する転送方式のタイプによって、そのタイプに設定された宛先アドレスが決まります。
ステップ 6	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-level** *number* **Example:** `switch(config-callhome)# destination-profile full-txt-destination message-level 5`	この宛先プロファイルの Smart Call Home メッセージの重大度を設定します。Cisco NX-OS では、Smart Call Home 重大度が一致する、またはそれ以上であるアラートのみが、このプロファイルの宛先に送信されます。指定できる範囲は 0 ～ 9 です。9 は最大の重大度を示します。
ステップ 7	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-size** *number* **Example:** `switch(config-callhome)# destination-profile full-txt-destination message-size 100000`	この宛先プロファイルの最大メッセージ サイズを設定します。範囲は 0 ～ 5000000 です。デフォルト値は 2500000 です。
ステップ 8	**commit** **Example:** `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。

	Command or Action	Purpose
ステップ 9	(Optional) **show callhome destination-profile** [**profile** *name*] **Example:** `switch(config-callhome)# show callhome destination-profile profile full-text-destination`	1 つまたは複数の宛先プロファイルに関する情報を表示します。
ステップ 10	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

What to do next

1 つの宛先プロファイルに 1 つまたは複数のアラート グループを関連付けます。

アラート グループと宛先プロファイルのアソシエート

SUMMARY STEPS

1. **configure terminal**
2. **callhome**
3. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **alert-group** {**All** | **Cisco-TAC** | **Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**}
4. **commit**
5. (Optional) **show callhome destination-profile** [**profile** *name*]
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**callhome**
	Example:
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart Call Home コンフィギュレーション モードを開始します。

	Command or Action	Purpose
ステップ 3	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **alert-group** {**All** | **Cisco-TAC** | **Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**} **Example:** `switch(config-callhome)# destination-profile Noc101 alert-group All`	アラート グループをこの宛先プロファイルにアソシエートします。キーワード **All** を使用して、すべてのアラート グループをこの宛先プロファイルにアソシエートします。
ステップ 4	**commit** **Example:** `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	(Optional) **show callhome destination-profile** [**profile** *name*] **Example:** `switch(config-callhome)# show callhome destination-profile profile Noc101`	1 つまたは複数の宛先プロファイルに関する情報を表示します。
ステップ 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

What to do next

任意で **show** コマンドをアラート グループに追加し、SMTP 電子メール サーバを設定します。

アラート グループへの show コマンドの追加

1 つのアラート グループには、最大 5 個のユーザー定義 CLI **show** コマンドを割り当てることができます。

Note CiscoTAC-1 宛先プロファイルには、ユーザ定義の CLI **show** コマンドを追加できません。

SUMMARY STEPS

1. **configure terminal**
2. **callhome**
3. **alert-group** {**Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**} **user-def-cmd** *show-cmd*
4. **commit**
5. (Optional) **show callhome user-def-cmds**

6. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**callhome**
	Example:
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**alert-group** {**Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**} **user-def-cmd** *show-cmd* **Example:** `switch(config-callhome)# alert-group Configuration user-def-cmd show ip route`	**show** コマンド出力を、このアラート グループに送信された Smart Call Home メッセージに追加します。有効な **show** コマンドだけが受け入れられます。
ステップ 4	**commit** **Example:** `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	(Optional) **show callhome user-def-cmds** **Example:** `switch(config-callhome)# show callhome user-def-cmds`	アラート グループに追加されたすべてのユーザ定義 **show** コマンドに関する情報を表示します。
ステップ 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

What to do next

SMTP 電子メール サーバに接続するように Smart Call Home を設定します。

電子メール サーバの設定

Smart Call Home 機能が動作するよう SMTP サーバ アドレスを設定します。送信元および返信先 E メール アドレスも設定できます。

Smart Call Home には最大 5 個までの SMTP サーバを設定できます。サーバは、プライオリティに基づいて試行されます。最もプライオリティの高いサーバが最初に試行されます。メッセージが送信できない場合、制限に達するまでリスト内の次のサーバが試行されます。2 つのサーバのプライオリティが同じ場合は、先に設定された方が最初に試行されます。

SUMMARY STEPS

1. **configure terminal**
2. **callhome**
3. **transport email mail-server** *ip-address* [**port** *number*] [**priority** *number*] [**use-vrf** *vrf-name*]
4. (Optional) **transport email from** *email-address*
5. (Optional) **transport email reply-to** *email-address*
6. **commit**
7. (Optional) **show callhome transport**
8. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**callhome**
	Example:
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**transport email mail-server** *ip-address* [**port** *number*] [**priority** *number*] [**use-vrf** *vrf-name*] **Example:** `switch(config-callhome)# transport email mail-server 192.0.2.1 use-vrf Red`	ドメイン ネーム サーバ（DNS）名、IPv4 アドレス、または IPv6 アドレスのいずれかとして SMTP サーバを設定します。任意でポート番号を設定します。ポート範囲は 1 ～ 65535 です。デフォルト ポート番号は、25 です。 任意で、SMTP サーバのプライオリティを設定します。プライオリティの範囲は 1 ～ 100 で、1 が最高、100 が最低のプライオリティです。プライオリティを指定しない場合、デフォルト値の 50 が使用されます。 また、この SMTP サーバと通信する際に使用するよう任意で VRF を設定します。指定された VRF は、HTTP を使用したメッセージの送信には使用されません。

	Command or Action	Purpose
ステップ 4	(Optional) **transport email from** *email-address* **Example:** `switch(config-callhome)# transport email from person@company.com`	Smart Call Home メッセージの送信元電子メールフィールドを設定します。
ステップ 5	(Optional) **transport email reply-to** *email-address* **Example:** `switch(config-callhome)# transport email reply-to person@company.com`	Smart Call Home メッセージの返信先電子メールフィールドを設定します。
ステップ 6	**commit** **Example:** `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 7	(Optional) **show callhome transport** **Example:** `switch(config-callhome)# show callhome transport`	Smart Call Home に対する転送関係のコンフィギュレーションを表示します。
ステップ 8	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

What to do next

任意で、VRF を使用して HTTP で Smart Call Home メッセージを送信します。

HTTP を使用したメッセージ送信のための VRF 設定

VRF を使用すると、HTTP で Call Home メッセージを送信できます。HTTP VRF が設定されていない場合は、デフォルトの VRF を使用して HTTP でメッセージが転送されます。

手順の概要

1. **configure terminal**
2. **callhome**
3. **transport http use-vrf** *vrf-name*
4. **commit**
5. （任意） **show callhome**
6. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**callhome**
	例：
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**transport http use-vrf** *vrf-name* 例： `switch(config-callhome)# transport http use-vrf Blue`	HTTP で電子メールおよび他の Smart Call Home メッセージを送信するための VRF を設定します。
ステップ 4	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	（任意） **show callhome** 例： `switch(config-callhome)# show callhome`	Smart Call Home に関する情報を表示します。
ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

次のタスク

任意で、HTTP プロキシ サーバから HTTP メッセージを送信するように Smart Call Home を設定します。

HTTP プロキシ サーバの設定

手順の概要

1. **configure terminal**
2. **callhome**
3. **transport http proxy server** *ip-address* [**port** *number*]

4. **transport http proxy enable**
5. **commit**
6. （任意） **show callhome transport**
7. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**callhome**
	例：
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**transport http proxy server** *ip-address* [**port** *number*] 例： `switch(config-callhome)# transport http proxy server 192.0.2.1`	HTTP プロキシ サーバのドメイン ネーム サーバ（DNS）の名前、IPv4 アドレス、または IPv6 アドレスを設定します。任意でポート番号を設定します。ポート範囲は 1 ～ 65535 です。デフォルトのポート番号は 8080 です。
ステップ 4	**transport http proxy enable** 例： `switch(config-callhome)# transport http proxy enable`	Smart Call Home で、HTTP プロキシ サーバ経由ですべての HTTP メッセージを送信できるようにします。 （注） プロキシ サーバ アドレスが設定された後にだけ、このコマンドを実行できます。 （注） プロキシ サーバを経由してメッセージを転送するために使用する VRF は、**transport http use-vrf** コマンドを使用して設定したものと同じです。
ステップ 5	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 6	（任意） **show callhome transport** 例： `switch(config-callhome)# show callhome transport`	Smart Call Home に対する転送関係のコンフィギュレーションを表示します。

	コマンドまたはアクション	目的
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

次のタスク

任意で、定期的にインベントリ通知を送信するようにデバイスを設定します。

定期的なインベントリ通知の設定

デバイス上で現在有効にされて動作しているすべてのソフトウェア サービスのインベントリとともに、ハードウェア インベントリ情報を示すメッセージを定期的に送信するように、デバイスを設定できます。デバイスは 2 つの Smart Call Home 通知（定期的な設定メッセージと定期的なインベントリ メッセージ）を生成します。

SUMMARY STEPS

1. **configure terminal**
2. **callhome**
3. **periodic-inventory notification** [**interval** *days*] [**timeofday** *time*]
4. **commit**
5. (Optional) **show callhome**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**callhome**
	Example:
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**periodic-inventory notification** [**interval** *days*] [**timeofday** *time*] **Example:**	定期的なインベントリ メッセージを設定します。間隔の範囲は 1 ～ 30 日で、デフォルトは 7 です。*time* 引数は HH:MM の形式です。これは、*X* 日ごとに更

	Command or Action	Purpose
	`switch(config-callhome)# periodic-inventory notification interval 20`	新が送信される日の時間を定義します（ここで*X*は更新間隔です）。
ステップ 4	**commit** **Example:** `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	(Optional) **show callhome** **Example:** `switch(config-callhome)# show callhome`	Smart Call Home に関する情報を表示します。
ステップ 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

What to do next

任意で重複メッセージ スロットリングを無効にします。

重複メッセージ抑制のディセーブル化

同じイベントについて受信する重複メッセージの数を制限できます。デフォルトでは、デバイスは同じイベントについて受け取る重複メッセージの数を制限します。2 時間の時間枠内で送信された重複メッセージの数が 30 メッセージを超えると、デバイスは同じアラート タイプの以降のメッセージを廃棄します。

手順の概要

1. **configure terminal**
2. **callhome**
3. **no duplicate-message throttle**
4. **commit**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

	コマンドまたはアクション	目的

ステップ 2	**callhome**
	例：
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**no duplicate-message throttle** 例： `switch(config-callhome)# no duplicate-message throttle`	Smart Call Home の重複メッセージ抑制をディセーブルにします。 重複メッセージ抑制はデフォルトでイネーブルです。
ステップ 4	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

次のタスク

Smart Call Home をイネーブルにします。

Smart Call Home のイネーブル化またはディセーブル化

担当者情報を設定した場合、Smart Call Home 機能を有効にできます。

手順の概要

1. **configure terminal**
2. **callhome**
3. [**no**] **enable**
4. **commit**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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
	`switch# configure terminal` `switch(config)#`	

ステップ 2	**callhome**
	例：
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	[**no**] **enable** 例： `switch(config-callhome)# enable`	Smart Call Home をイネーブルまたはディセーブルにします。 Smart Call Home は、デフォルトでディセーブルです。
ステップ 4	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

次のタスク

任意でテスト メッセージを生成します。

Call Home メール転送用の SMTP-AUTH の設定

Call Home メール転送に SMTP-AUTH 機能を使用すると、ポート 25 経由のクリア テキストの代わりに、標準の SMTP-AUTH TCP ポート 587 または 465、またはその他のユーザー定義ポートを使用して、安全な方法でメールを共有できます。この機能は、Cisco NX-OS リリース 10.2(3)F からサポートされています。

始める前に

- SMTP-AUTH サーバー証明書がスイッチにインストールされている必要があります。

手順の概要

1. **configure terminal**
2. **callhome**
3. **email-contact** *email-address*
4. **destination-profile** *name*
5. **destination-profile** *name* **format** {**XML** | **full-txt** | **short-txt**}
6. **destination-profile** *name* **email-address** *email-address*

7. **destination-profile** *name* **alert-group all**
8. **transport email from callhome_***email-address*
9. **transport email smtp-server** *hostname/ip-address* **port 465 use-vrf** *vrf-name*
10. **transport email username** *username* **passwd** *password* {**cleartext** | **encrypted**}
11. （任意） **transport http use-vrf** *vrf-name*
12. [**no**] **enable**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**callhome**
	例：
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**email-contact** *email-address* 例： `switch(config-callhome)# email-contact admin@Mycompany.com`	デバイスの主要責任者の電子メール アドレスを設定します。 *email-address* には、電子メールアドレスの形式で、最大 255 の英数字を使用できます。 （注） 任意の有効な電子メール アドレスを使用できます。アドレスには、空白を含めることはできません。
ステップ 4	**destination-profile** *name* 例： `switch(config-callhome)# destination-profile testProfile-1`	新しい宛先プロファイルを作成します。名前は、最大 31 文字の英数字で指定できます。
ステップ 5	**destination-profile** *name* **format** {**XML**	**full-txt**	**short-txt**} 例： `switch(config-callhome)# destination-profile testProfile-1 format XML`	プロファイルのメッセージ フォーマットを設定します。名前は、最大 31 文字の英数字で指定できます。

	コマンドまたはアクション	目的

ステップ 6	**destination-profile** *name* **email-address** *email-address*
	例：
	`switch(config-callhome)# destination-profile testProfile-1`
	`index 1 email address person@company.com`
		安全なメールの配信先となる電子メール アドレスを構成します。宛先プロファイルには、最大 50 個の電子メール アドレスを設定できます。

ステップ 7	**destination-profile** *name* **alert-group all**
	例：
	`switch(config-callhome)# destination-profile testProfile-1`
	`alert-group all`
		接続先 プロファイルに全てのアラートグループを関連します。

ステップ 8	**transport email from callhome_***email-address* 例： `switch(config)# transport email from callhome_person@company.com`	Smart Call Home メッセージの **callhome** フィールドのメールを構成します。

ステップ 9	**transport email smtp-server** *hostname/ip-address* **port 465 use-vrf** *vrf-name*
	例：
	`switch(config)# transport email smtp-server 10.1.1.174 port 465 use-vrf management`
	`switch(config)# transport email smtp-server 10.1.1.174 port 587 use-vrf management`
		transport email smtp-server *hostname/ip-address* **port 587 use-vrf** *vrf-name*
		SMTP-AUTH メール転送方法を有効にします。これは、標準 TCP ポート、つまり 465 および 587 ポートを介した STARTTLS ベースの SMTP-AUTH です。

ステップ 10	**transport email username** *username* **passwd** *password* {**cleartext** | **encrypted**} 例： `switch(config)# transport email username user1 passwd Y2FsbGhvbWUK encrypted`	ユーザー名とパスワードを受け入れ、これらの詳細を SMTP-AUTH 認証に渡します。 ユーザー名は英数字で、256 バイト未満である必要があります。パスワード オプションは、クリア テキストまたは暗号化された形式で入力できます（ユーザーが既に暗号化されたパスワードを持っている場合）。パスワードの長さは、平文オプションの場合は 64 バイト未満、暗号化オプションの場合は 256 バイト未満にする必要があります。 （注） 次のシナリオでは SMTP-AUTH が正しく動作しません。 • 平文のパスワードの長さが 56 文字を超える場合。 • パスワードに次の特殊文字のいずれかが含まれている場合: • ドル記号 - **$**

	コマンドまたはアクション	目的
		• 丸カッコ - **(** と **)** • アンパサンド - **&** • 角カッコ - **[** と **]** • セミコロン - **;** • 疑問符 - **?** • 縦棒またはパイプ - **|** • アポストロフィ - **'** • 引用符- **'**、**"**、**‘**、**’**、**“**、および **”** • 小なり記号と大なり記号 - **>** および **<**
ステップ 11	（任意） **transport http use-vrf** *vrf-name* 例： `switch(config)# transport http use-vrf management`	HTTP で電子メールおよび他の Smart Call Home メッセージを送信するための VRF を設定します。
ステップ 12	[**no**] **enable** 例： `switch(config)# enable`	Smart Call Home をイネーブルにします。 このコマンドの no 形式は、Smart Call Home を無効にします。

送信元インターフェイスの構成

この手順を使用して、パケットが送信される VRF の送信元インターフェイスを識別します。

手順の概要

1. **configure terminal**
2. **callhome**
3. **source interface** *interface_name*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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
	`switch# configure terminal` `switch(config)#`	

ステップ 2	**callhome**
	例：
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**source interface** *interface_name* 例： `switch(config-callhome)# source-interface Ethernet1/1`	Call Home http(s) 転送用の送信元インターフェイスと、Call Home をスマート ライセンスのトランスポートとして使用する場合に定義します。

Smart Call Home 設定のテスト

テスト メッセージを生成して Smart Call Home 通信をテストできます。

手順の概要

1. **configure terminal**
2. **callhome**
3. **callhome send** [**configuration** | **diagnostic**]
4. **callhome test**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**callhome**
	例：
	`switch(config)# callhome`
	`switch(config-callhome)#`
		Smart 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 3	**callhome send** [**configuration**	**diagnostic**] 例： `switch(config-callhome)# callhome send diagnostic`	設定されたすべての宛先に指定の Smart Call Home テスト メッセージを送信します。

	コマンドまたはアクション	目的
ステップ 4	**callhome test** 例： `switch(config-callhome)# callhome test`	設定されたすべての宛先にテストメッセージを送信します。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

Smart Call Home 設定の確認

Smart Call Home 設定情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show callhome	Smart Call Home 設定を表示します。
show callhome destination-profile *name*	1 つまたは複数の Smart Call Home 宛先プロファイルを表示します。
show callhome transport	Smart Call Home に対する転送関係のコンフィギュレーションを表示します。
show callhome user-def-cmds	任意のアラート グループに追加された CLI コマンドを表示します。
show running-config callhome [**all**]	Smart Call Home の実行コンフィギュレーションを表示します。
show startup-config callhome	Smart Call Home のスタートアップ コンフィギュレーションを表示します。
show tech-support callhome	Smart Call Home のテクニカル サポート出力を表示します。

Smart Call Home の設定例

Noc101 という宛先プロファイルを作成し、コンフィギュレーションのアラート グループをこのプロファイルに関連付けて、コンタクト情報と電子メールの情報を設定した後で、HTTP を介して Smart Call Home メッセージを送信するための VRF を指定する例を示します。Noc101 という宛先プロファイルを作成し、コンフィギュレーションのアラート グループをこのプロ

ファイルに関連付けて、コンタクト情報と電子メールの情報を設定した後で、HTTP を介して Call Home メッセージを送信するための VRF を指定する例を示します。

```
configure terminal
snmp-server contact person@company.com
callhome
distribute
email-contact admin@Mycompany.com
phone-contact +1-800-123-4567
streetaddress 123 Anystreet st. Anytown,AnyWhere
destination-profile Noc101 format full-txt
destination-profile full-text-destination email-addr person@company.com
destination-profile full-text-destination message-level 5
destination-profile Noc101 alert-group Configuration
alert-group Configuration user-def-cmd show ip route
transport email mail-server 192.0.2.10 priority 1
transport http use-vrf Blue
enable
commit
```

次に、複数の SMTP サーバを Smart Call Home メッセージに設定する例を示します。

```
configure terminal
callhome
transport email mail-server 192.0.2.10 priority 4
transport email mail-server 172.21.34.193
transport email smtp-server 10.1.1.174
transport email mail-server 64.72.101.213 priority 60
transport email from person@company.com
transport email reply-to person@company.com
commit
```

（注） **callhome email mail-server** コマンドを使用した認証目的の複数の smtp サーバーの構成はサポートされていません。

上記のコンフィギュレーションに基づいて、SMTP サーバはこの順序で試行されます。

10.1.1.174（プライオリティ 0）

192.0.2.10（プライオリティ 4）

172.21.34.193（プライオリティ 50、デフォルト）

64.72.101.213（プライオリティ 60）

（注） **transport email smtp-server** コマンドのプライオリティは、最大の 0 です。このコマンドで指定されたサーバは最初に試行され、次に、**transport email mail-server** コマンドで指定されたサーバが、プライオリティの順に試行されます。

次に、HTTP プロキシ サーバからの HTTP メッセージを送信するように、Smart Call Home を設定する例を示します。

```
configure terminal
callhome
transport http proxy server 10.10.10.1 port 4
transport http proxy enable
commit
```

次に、call home mail transfer に対する SMTP-AUTH サーバーを構成する例を示します。

```
callhome
email-contact admin@Mycompany.com
destination-profile testProfile-1
destination-profile testProfile-1 format XML
destination-profile testProfile-1 index 1 email-addr person@company.com
destination-profile testProfile-1 alert-group all
destination-profile full_txt alert-group test
transport email from callhome_person@company.com
transport email smtp-server 10.1.1.174 port 587 use-vrf management
transport email username user1 passwd Y2FsbGhvbWUK encrypted
transport http use-vrf management
enable
```

その他の参考資料

イベント トリガ

次の表に、イベント トリガおよび Smart Call Home メッセージの重大度を示します。

アラート グループ	イベント名	説明	Smart Call Home 重大度
設定（Configuration）	PERIODIC_CONFIGURATION	定期的コンフィギュレーション アップデート メッセージ	2
診断	DIAGNOSTIC_MAJOR_ALERT	GOLD が生成したメジャー アラート	7
	DIAGNOSTIC_MINOR_ALERT	GOLD が生成したマイナー アラート	4
	DIAGNOSTIC_NORMAL_ALERT	Smart Call Home が生成した通常の診断アラート	2

アラート グループ	イベント名	説明	Smart Call Home 重大度
環境および CISCO_TAC	FAN_FAILURE	冷却ファンが障害になりました。	5
	POWER_SUPPLY_ALERT	電源モジュールに関する警告の発生	6
	POWER_SUPPLY_FAILURE	電源モジュールの故障	6
	POWER_SUPPLY_SHUTDOWN	電源モジュールのシャットダウン	6
	TEMPERATURE_ALARM	温度センサーの障害	6
	TEMPERATURE_MAJOR_ALARM	温度が動作メジャーしきい値を超えたことを示す温度センサーの表示	6
	TEMPERATURE_MINOR_ALARM	温度が動作マイナーしきい値を超えたことを示す温度センサーの表示	4
インベントリおよび CISCO_TAC	COLD_BOOT	スイッチの電源が投入され、コールドブート シーケンスにリセットされます。	2
	HARDWARE_INSERTION	シャーシへの新しいハードウェア コンポーネントの追加	2
	HARDWARE_REMOVAL	シャーシからのハードウェアの取り外し	2
	PERIODIC_INVENTORY	定期的インベントリ メッセージの作成	2
ライセンス	LICENSE_VIOLATION	使用中の機能にライセンスがなく、猶予期間を経てオフになった場合	6
Line module Hardware および CISCO_TAC	LINEmodule_FAILURE	モジュールの動作障害	7
スーパーバイザ ハードウェアおよび CISCO_TAC	SUP_FAILURE	スーパーバイザ モジュールの動作障害	7

アラート グループ	イベント名	説明	Smart Call Home 重大度
Syslog グループ ポート	PORT_FAILURE	ポート ファシリティに対応する syslog メッセージの生成	6
	SYSLOG_ALERT	syslog アラート メッセージの生成 （注） Link up/down syslog メッセージは、Smart Call Home メッセージまたはアラート通知をトリガーしません。	5
システムおよび CISCO_TAC	SW_CRASH	ステートレス リスタートによるソフトウェア プロセス障害、つまりサービスの停止スーパーバイザ モジュールでのプロセス クラッシュに対してメッセージが送信されます。	5
	SW_SYSTEM_INCONSISTENT	ソフトウェアまたはファイル システムにおける不整合の検出	5
テストおよび CISCO_TAC	TEST	ユーザが作成したテストの発生	2

メッセージ フォーマット

Smart Call Home では、次のメッセージ フォーマットがサポートされます。

ショート テキスト メッセージ フォーマット

次の表に、すべてのメッセージ タイプのショート テキスト書式設定オプションを示します。

データ項目	説明
デバイス ID	設定されたデバイス名
日時スタンプ	起動イベントのタイム スタンプ
エラー判別メッセージ	起動イベントの簡単な説明（英語）
アラームの緊急度	エラー レベル（システム メッセージに適用されるエラー レベルなど）

共通のイベント メッセージ フィールド

次の表では、フル テキストまたは XML メッセージに共通するイベント メッセージ フィールドの最初のセットについて説明します。

データ項目（プレーン テキストおよび **XML**）	説明（プレーン テキストおよび **XML**）	**XML** タグ（**XML** のみ）
Timestamp	ISO 時刻通知でのイベントの日付/タイム スタンプ YYYY-MM-DD HH:MM:SS GMT+HH:MM	/aml/header/time
メッセージ名	メッセージの名前。	/aml/header/name
メッセージ タイプ	リアクティブまたはプロアクティブなどのメッセージ タイプの名前。	/aml/header/type
メッセージ グループ	Syslog などのアラート グループの名前。	/aml/header/group
重大度	メッセージの重大度	/aml/header/level
送信元 ID	ルーティング製品タイプ（Cisco Nexus 9000 シリーズ スイッチなど）。	/aml/header/source
デバイス ID	メッセージを生成したエンド デバイスの固有デバイス識別情報（UDI）。メッセージがデバイスに対して固有でない場合は、このフィールドを空にする必要があります。形式は、*type@Sid@serial* です。 • *type* は、バックプレーン IDPROM からの製品の型番です。 • @ は区切り文字です。 • *Sid* は C で、シリアル ID をシャーシ シリアル番号として特定します。 • *serial* は、Sid フィールドによって識別される番号です。 例：N9K-C9508@C@12345678	/aml/ header/deviceId

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
カスタマー ID	サポート サービスによって契約情報やその他の ID に使用されるオプションのユーザ設定可能なフィールド	/aml/ header/customerID
連絡先 ID	サポート サービスによって契約情報やその他の ID に使用されるオプションのユーザ設定可能なフィールド	/aml/ header /contractId
サイト ID	シスコが提供したサイト ID または別のサポート サービスにとって意味のあるその他のデータに使用されるオプションのユーザ設定可能なフィールド	/aml/ header/siteId
Server ID	デバイスからメッセージが生成された場合、この ID はデバイスの Unique Device Identifier（UDI）フォーマットです。形式は、*type@Sid@serial* です。 • *type* は、バックプレーン IDPROM からの製品の型番です。 • @ は区切り文字です。 • *Sid* は C で、シリアル ID をシャーシ シリアル番号として特定します。 • *serial* は、Sid フィールドによって識別される番号です。 例：N9K-C9508@C@12345678	/aml/header/serverId
メッセージの説明	エラーを説明するショート テキスト。	/aml/body/msgDesc
デバイス名	イベントが発生したノード（デバイスのホスト名）。	/aml/body/sysName
担当者名	イベントが発生したノード関連の問題について問い合わせる担当者名。	/aml/body/sysContact

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
[連絡先電子メール（Contact email）]	この装置の担当者の電子メール アドレス。	/aml/body/sysContactEmail
連絡先電話番号	このユニットの連絡先である人物の電話番号	/aml/body/sysContactPhoneNumber
住所	この装置関連の返品許可（RMA）部品の送付先住所を保存するオプション フィールド。	/aml/body/sysStreetAddress
モデル名	デバイスのモデル名（製品ファミリ名に含まれる具体的なモデル）。	/aml/body/chassis/name
シリアル番号	ユニットのシャーシのシリアル番号	/aml/body/chassis/serialNo
シャーシの部品番号	シャーシの最上アセンブリ番号	/aml/body/chassis/partNo

アラート グループ メッセージ フィールド

次の表に、フル テキストおよび XML のアラート グループ メッセージに固有のフィールドについて説明します。1 つのアラート グループに対して複数の CLI コマンドが実行される場合は、これらのフィールドが繰り返されることがあります。

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
Command output name	実行された CLI コマンドの正確な名前。	/aml/attachments/attachment/name
添付ファイルの種類	特定のコマンド出力。	/aml/attachments/attachment/type
MIME タイプ	プレーン テキストまたは符号化タイプ。	/aml/attachments/attachment/mime
コマンド出力テキスト	自動的に実行されるコマンドの出力	/aml/attachments/attachment/atdata

リアクティブおよびプロアクティブ イベント メッセージのフィールド

次の表では、フルテキストまたはXMLメッセージのリアクティブおよびプロアクティブ イベント メッセージ形式について説明します。

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
シャーシのハードウェア バージョン	シャーシのハードウェア バージョン。	/aml/body/chassis/hwVersion
スーパーバイザ モジュールのソフトウェア バージョン	最上レベルのソフトウェア バージョン	/aml/body/chassis/swVersion
影響のある FRU 名	イベント メッセージを生成する関連 FRU の名前。	/aml/body/fru/name
影響のある FRU のシリアル番号	関連 FRU のシリアル番号。	/aml/body/fru/serialNo
影響のある FRU の製品番号	関連 FRU の部品番号。	/aml/body/fru/partNo
FRU スロット	イベント メッセージを生成する FRU のスロット番号。	/aml/body/fru/slot
FRU ハードウェア バージョン	関連 FRU のハードウェア バージョン。	/aml/body/fru/hwVersion
FRU ソフトウェアのバージョン	関連 FRU で稼働しているソフトウェア バージョン。	/aml/body/fru/swVersion

インベントリ イベント メッセージのフィールド

次の表に、フル テキストまたは XML メッセージのコンポーネント イベント メッセージ形式について説明します。

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
シャーシのハードウェア バージョン	シャーシのハードウェア バージョン。	/aml/body/chassis/hwVersion
スーパーバイザ モジュールのソフトウェア バージョン	最上レベルのソフトウェア バージョン	/aml/body/chassis/swVersion
FRU 名	イベント メッセージを生成する関連 FRU の名前。	/aml/body/fru/name
FRU s/n	FRU のシリアル番号。	/aml/body/fru/serialNo
FRU 製品番号	FRU の部品番号。	/aml/body/fru/partNo
FRU スロット	FRU のスロット番号。	/aml/body/fru/slot
FRU ハードウェア バージョン	FRU のハードウェア バージョン。	/aml/body/fru/hwVersion

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
FRU ソフトウェアのバージョン	FRU で稼働しているソフトウェア バージョン。	/aml/body/fru/swVersion

ユーザが作成したテスト メッセージのフィールド

次の表に、フルテキストまたはXMLのユーザが作成したテストメッセージ形式について説明します。

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
プロセス ID	固有のプロセス ID	/aml/body/process/id
プロセス状態	プロセスの状態（実行中、中止など）	/aml/body/process/processState
プロセス例外	原因コードの例外	/aml/body/process/exception

フル テキスト形式での syslog アラート通知の例

次の例では、Syslog ポート アラート グループ通知のフル テキスト形式を示します。

```
Severity Level:5
Series:Nexus9000
Switch Priority:0
Device Id:N9K-C9508@C@TXX12345678
Server Id:N9K-C9508C@TXX12345678
Time of Event:2013-05-17 16:31:33 GMT+0000 Message Name:
Message Type:syslog
System Name:dc3-test
Contact Name:Jay Tester
Contact Email:contact@example.com
Contact Phone:+91-80-1234-5678
Street Address:#1 Any Street
Event Description:SYSLOG_ALERT 2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR:
Error (0x20) while communicating with component MTS_SAP_ELTM
opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1)

syslog_facility:ETHPORT
start chassis information:
Affected Chassis:N9K-C9508
Affected Chassis Serial Number:TXX12345678 Affected Chassis Hardware Version:0.405
Affected Chassis Software Version:6.1(2) Affected Chassis Part No:11-11111-11 end chassis
 information:
start attachment
   name:show logging logfile | tail -n 200
   type:text
   data:
  2013 May 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages) cleared
 by user
   2013 May 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
 /dev/ttyS0 /dev/ttyS0_console
   2013 May 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
```

```
 /dev/ttyS0 /dev/ttyS0_console
   2013 May 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16:
Invalid argument: - sshd[14484]
   2013 May 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
 /dev/ttyS0 /dev/ttyS0_console
   2013 May 17 12:11:18 dc3-test %SYSMGR-STANDBY-5-SUBPROC_TERMINATED: "System Manager
(gsync controller)" (PID 12000) has finished with error code
SYSMGR_EXITCODE_GSYNCFAILED_NONFATAL (12).
   2013 May 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
 /dev/ttyS0 /dev/ttyS0_console
   2013 May 17 16:28:44 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
 Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 3504)
hasn't caught signal 9 (no core).
   2013 May 17 16:29:08 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
 Core not generated by system for eltm(0). WCOREDUMP(9) returned zero.
   2013 May 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23210)
hasn't caught signal 9 (no core).
   2013 May 17 16:29:17 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
 Core not generated by system for eltm(0). WCOREDUMP(9) returned zero.
   2013 May 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23294)
hasn't caught signal 9 (no core).
   2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is
becoming active (pre-start phase).
   2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is becoming
 active.
   2013 May 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send failed
 - device_test
   2013 May 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR: netstack [4336] Unrecognized
 message from MRIB. Major type 1807
   2013 May 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN
   2013 May 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]

   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10 - ntpd[19045]

   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined - ntpd[19045]

   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined -
ntpd[19045]
   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]

   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]

   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]

   2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
 recovery failed (0)
   2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
 recovery failed (0)
   2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
   2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19072]
   2013 May 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19073]
   2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
   2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19079]
   2013 May 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP
   2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
   2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
```

```
dcos-xinetd[19105]
   2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 2 present
but all AC inputs are not connected, ac-redundancy might be affected
   2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 3 present
but all AC inputs are not connected, ac-redundancy might be affected
   2013 May 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
   2013 May 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
   2013 May 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
   2013 May 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
   2013 May 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
   2013 May 17 16:30:47 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
 Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 4820)
hasn't caught signal 9 (no core).
   2013 May 17 16:31:02 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
 Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24239)
hasn't caught signal 9 (no core).
   2013 May 17 16:31:14 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
 Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24401)
hasn't caught signal 9 (no core).
   2013 May 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
   2013 May 17 16:31:23 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
 Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24407)
hasn't caught signal 9 (no core).
   2013 May 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
   2013 May 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
   2013 May 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
   2013 May 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR: netstack [4336] (null)
   2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
 with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT:
Ethernet3/1) end attachment start attachment
   type:text
   data:

   dc3-test interfaces:
        Ethernet3/1    Ethernet3/2    Ethernet3/3
        Ethernet3/4    Ethernet3/5    Ethernet3/6
        Ethernet3/7    Ethernet3/8    Ethernet3/9
        Ethernet3/10   Ethernet3/11   Ethernet3/12
        Ethernet3/13   Ethernet3/14   Ethernet3/15
        Ethernet3/16   Ethernet3/17   Ethernet3/18
        Ethernet3/19   Ethernet3/20   Ethernet3/21
        Ethernet3/22   Ethernet3/23   Ethernet3/24
        Ethernet3/25   Ethernet3/29   Ethernet3/30
        Ethernet3/31   Ethernet3/32   Ethernet3/33
        Ethernet3/34   Ethernet3/35   Ethernet3/36
        Ethernet3/37   Ethernet3/38   Ethernet3/39
        Ethernet3/40   Ethernet3/41   Ethernet3/42
        Ethernet3/43   Ethernet3/44   Ethernet3/45
        Ethernet3/46   Ethernet3/47   Ethernet3/48
end attachment
start attachment
   type:text
   data:
```

```
end attachment
start attachment
   name:show license usage
   type:text
   data:
   Feature Ins Lic Status Expiry Date Comments
             Count
   --------------------------------------------------------------------------------
   LAN_ENTERPRISE_SERVICES_PKG Yes - Unused Never -
   --------------------------------------------------------------------------------
end attachment
```

XML 形式での syslog アラート通知の例

次の例では、Syslog ポート アラート グループ通知の XML を示します。

```
<?xml version="1.0" encoding="UTF-8" ?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.cisco.com/2004/01/aml-session"
soap-env:mustUnderstand="true"
soap-env:role="http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.cisco.com/neddce/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.cisco.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.cisco.com/appliance/uri</aml-session:From>
<aml-session:MessageId>1004:TXX12345678:478F82E6</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.cisco.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.cisco.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2013-05-17 16:31:33 GMT+0000</aml-block:CreationDate>
<aml-block:Builder> <aml-block:Name>DC3</aml-block:Name>
<aml-block:Version>4.1</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>1005:TXX12345678:478F82E6</aml-block:GroupId>
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>5</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:CallHome xmlns:ch="http://www.cisco.com/2005/05/callhome" version="1.0">
<ch:EventTime>2013-05-17 16:31:33 GMT+0000</ch:EventTime>
<ch:MessageDescription>SYSLOG_ALERT 2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR:
 Error (0x20) while communicating with component MTS_SAP_ELTM
opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1) </ch:MessageDescription>
<ch:Event> <ch:Type>syslog</ch:Type> <ch:SubType></ch:SubType> <ch:Brand>Cisco</ch:Brand>
 <ch:Series>Nexus9000</ch:Series> </ch:Event> <ch:CustomerData> <ch:UserData>
<ch:Email>contact@example.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:DeviceId>N9K-C9508@C@TXX12345678</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
```

```
<ch:Name>dc3-test</ch:Name>
<ch:Contact>Jay Tester</ch:Contact> <ch:ContactEmail>contact@example.com</ch:ContactEmail>
<ch:ContactPhoneNumber>+91-80-1234-5678</ch:ContactPhoneNumber>
<ch:StreetAddress>#1, Any Street</ch:StreetAddress> </ch:SystemInfo> </ch:CustomerData>
 <ch:Device> <rme:Chassis xmlns:rme="http://www.cisco.com/rme/4.1">
<rme:Model>N9K-C9508</rme:Model>
<rme:HardwareVersion>0.405</rme:HardwareVersion>
<rme:SerialNumber>TXX12345678</rme:SerialNumber>
</rme:Chassis>
</ch:Device>
</ch:CallHome>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging logfile | tail -n 200</aml-block:Name> <aml-block:Data
encoding="plain">
<![CDATA[2013 May 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages)
 cleared by user
2013 May 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16: Invalid
 argument: - sshd[14484]
2013 May 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 12:11:18 dc3-test %SYSMGR-STANDBY-5-SUBPROC_TERMINATED: \"System Manager
(gsync controller)\" (PID 12000) has finished with error code
SYSMGR_EXITCODE_GSYNCFAILED_NONFATAL (12).
2013 May 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 16:28:44 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 3504)
hasn't caught signal 9 (no core).
2013 May 17 16:29:08 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23210)
hasn't caught signal 9 (no core).
2013 May 17 16:29:17 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23294)
hasn't caught signal 9 (no core).
2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is
becoming active (pre-start phase).
2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is becoming
 active.
2013 May 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send failed -
 device_test
2013 May 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR: netstack [4336] Unrecognized
 message from MRIB. Major type 1807
2013 May 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN
2013 May 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
```

```
2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19072]
2013 May 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19073]
2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19079]
2013 May 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP
2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19105]
2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 2 present but
all AC inputs are not connected, ac-redundancy might be affected
2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 3 present but
all AC inputs are not connected, ac-redundancy might be affected
2013 May 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
2013 May 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
2013 May 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
2013 May 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
2013 May 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
2013 May 17 16:30:47 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 4820)
hasn't caught signal 9 (no core).
2013 May 17 16:31:02 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24239)
hasn't caught signal 9 (no core).
2013 May 17 16:31:14 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24401)
hasn't caught signal 9 (no core).
2013 May 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
2013 May 17 16:31:23 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24407)
hasn't caught signal 9 (no core).
2013 May 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
2013 May 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
2013 May 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
2013 May 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR: netstack [4336] (null)
2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
 with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT:
Ethernet3/1) ]]> </aml-block:Data> </aml-block:Attachment> <aml-block:Attachment
type="inline"> <aml-block:Name> <aml-block:Data encoding="plain"> <![CDATA[
dc3-test interfaces:
   Ethernet3/1     Ethernet3/2     Ethernet3/3
   Ethernet3/4     Ethernet3/5     Ethernet3/6
   Ethernet3/7     Ethernet3/8     Ethernet3/9
   Ethernet3/10    Ethernet3/11    Ethernet3/12
   Ethernet3/13    Ethernet3/14    Ethernet3/15
   Ethernet3/16    Ethernet3/17    Ethernet3/18
   Ethernet3/19    Ethernet3/20    Ethernet3/21
```

```
   Ethernet3/22    Ethernet3/23    Ethernet3/24
   Ethernet3/25    Ethernet3/26    Ethernet3/27
   Ethernet3/28    Ethernet3/29    Ethernet3/30
   Ethernet3/31    Ethernet3/32    Ethernet3/33
   Ethernet3/34    Ethernet3/35    Ethernet3/36
   Ethernet3/37    Ethernet3/38    Ethernet3/39
   Ethernet3/40    Ethernet3/41    Ethernet3/42
   Ethernet3/43    Ethernet3/44    Ethernet3/45
   Ethernet3/46    Ethernet3/47    Ethernet3/48


]]>
</aml-block:Data>
</aml-block:Attachment>
<aml-block:Attachment type="inline">
<aml-block:Name> <aml-block:Data encoding="plain"> <!----> </aml-block:Data>
</aml-block:Attachment> <aml-block:Attachment type="inline"> <aml-block:Name>show license
 usage</aml-block:Name> <aml-block:Data encoding="plain">
<![CDATA[Feature Ins Lic Status Expiry Date Comments
                    Count
--------------------------------------------------------------------------------
LAN_ENTERPRISE_SERVICES_PKG Yes - Unused Never -
--------------------------------------------------------------------------------
]]>
</aml-block:Data>
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>
```

MIB

MIB	MIB のリンク
Smart Call Home に関連する MIB	サポートされている MIB を検索およびダウンロ 次の URL にアクセスしてください。 https://cisco.github.io/cisco-mibs/supportlists/nexus9 Nexus9000MIBSupportList.html

第 13 章

Session Manager の設定

この章では、Cisco NX-OS デバイスで Session Manager を設定する方法について説明します。

この章は、次の内容で構成されています。

- セッション マネージャについて, on page 251
- セッション マネージャの前提条件 （252 ページ）
- Session Manager の注意事項および制約事項 （252 ページ）
- Session Manager の設定 （252 ページ）
- Session Manager 設定の確認, on page 255
- Session Manager のコンフィギュレーション例, on page 256
- その他の参考資料 （256 ページ）

セッション マネージャについて

Session Manager を使用すると、設定変更をバッチ モードで実行できます。Session Manager は次のフェーズで機能します。

- コンフィギュレーション セッション：Session Manager モードで実行するコマンドのリストを作成します。
- 検証：設定の基本的なセマンティック チェックを行います。Cisco NX-OS は、設定の一部でセマンティクス検査が失敗した場合にエラーを返します。
- 検証：既存のハードウェア設定、ソフトウェア設定、およびリソースに基づいて、設定全体を確認します。Cisco NX-OS は、設定がこの確認フェーズで合格しなかった場合にエラーを返します。
- コミット：Cisco NX-OS はコンフィギュレーション全体を確認して、デバイスに対する変更を実行します。エラーが発生すると、Cisco NX-OS は元の設定に戻ります。
- 打ち切り：設定変更を実行しないで廃棄します。

任意で、変更をコミットしないでコンフィギュレーションセッションを終了できます。また、コンフィギュレーション セッションを保存することもできます。

高可用性

Session Manager セッションは、スーパーバイザのスイッチオーバー後も引き続き使用できます。セッションはソフトウェア リロード後までは維持されません。

セッション マネージャの前提条件

使用する予定の Session Manager コマンドをサポートする権限があることを確認してください。

Session Manager の注意事項および制約事項

Session Manager には、次の注意事項および制限事項があります。

- 1 つのセッションを使用して実行できるサービス アクセス ポイント（SAP）は 1 つだけです。
- 設定セッションは、リロード後に保持されません。
- Session Manager は、アクセス コントロール リスト（ACL）および Quality of Service（QoS）機能だけをサポートします。
- 作成できるコンフィギュレーション セッションの最大数は 32 です。
- すべてのセッションで設定できるコマンドの最大数は 20,000 です。
- 複数のコンフィギュレーションセッションまたはコンフィギュレーション ターミナル モードで、コンフィギュレーション コマンドを同時に実行することはできません。パラレル コンフィギュレーション（例えば 1 つのコンフィギュレーション セッションと 1 つのコンフィギュレーション ターミナル）は、コンフィギュレーション セッションで確認または検証が失敗する原因になります。
- コンフィギュレーション セッションで、あるインターフェイスを設定中にそのインターフェイスをリロードすると、そのときにインターフェイスがデバイス上になくても、セッション マネージャがコマンドを受け取ることになります。

Session Manager の設定

（注） Cisco NX-OS コマンドは Cisco IOS コマンドと異なる場合があるので注意してください。

セッションの作成

作成できるコンフィギュレーション セッションの最大数は 32 です。

SUMMARY STEPS

1. **configure session** *name*
2. (Optional) **show configuration session** [*name*]
3. (Optional) **save** *location*

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure session** *name*
	Example:
	`switch# configure session myACLs`
	`switch(config-s)#`
		コンフィギュレーションセッションを作成し、セッション コンフィギュレーション モードを開始します。名前は任意の英数字ストリングです。
		セッションの内容を表示します。

ステップ 2	(Optional) **show configuration session** [*name*] **Example:** `switch(config-s)# show configuration session myACLs`	セッションの内容を表示します。
ステップ 3	(Optional) **save** *location* **Example:** `switch(config-s)# save bootflash:sessions/myACLs`	セッションをファイルに保存します。保管場所には bootflash:、slot0:、または volatile: を指定できます。

セッションでの ACL の設定

コンフィギュレーション セッションで ACL を設定できます。

SUMMARY STEPS

1. **configure session** *name*
2. **ip access-list** *name*
3. (Optional) **permit** *protocol source destination*
4. **interface** *interface-type number*
5. **ip access-group** *name* {**in** | **out**}
6. (Optional) **show configuration session** [*name*]

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure session** *name*
	Example:
	`switch# configure session myacls`
	`switch(config-s)#`
		コンフィギュレーションセッションを作成し、セッション コンフィギュレーション モードを開始します。名前は任意の英数字ストリングです。

ステップ 2	**ip access-list** *name*
	Example:
	`switch(config-s)# ip access-list acl1`
	`switch(config-s-acl)#`
		ACL を作成し、その ACL の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	(Optional) **permit** *protocol source destination* **Example:** `switch(config-s-acl)# permit tcp any any`	ACL に許可文を追加します。

ステップ 4	**interface** *interface-type number*
	Example:
	`switch(config-s-acl)# interface ethernet 2/1`
	`switch(config-s-if)#`
		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 5	**ip access-group** *name* {**in** | **out**} **Example:** `switch(config-s-if)# ip access-group acl1 in`	アクセス グループを適用するトラフィックの方向を指定します。
ステップ 6	(Optional) **show configuration session** [*name*] **Example:** `switch(config-s-if)# show configuration session myacls`	セッションの内容を表示します。

セッションの確認

セッション モードで次のコマンドを使用して、セッションを確認します。

コマンド	目的
verify [**verbose**] **例:** `switch(config-s)# verify`	既存のハードウェアおよびソフトウェアのコンフィギュレーションおよびリソースに基づいて、コンフィギュレーション全体を確認します。Cisco NX-OS は、設定がこの確認で合格しなかった場合にエラーを返します。

セッションのコミット

セッション モードで次のコマンドを使用して、セッションをコミットします。

コマンド	目的
commit [**verbose**] **例:** `switch(config-s)# commit`	現在のセッションで行われたコンフィギュレーションの変更を検証し、有効な変更をデバイスに適用します。検証に失敗した場合、Cisco NX-OS は元の設定に戻ります。

セッションの保存

セッション モードで次のコマンドを使用して、セッションを保存します。

コマンド	目的
save *location* **例:** `switch(config-s)# save bootflash:sessions/myACLs`	（任意）セッションをファイルに保存します。保管場所には bootflash:、slot0:、または volatile: を指定できます。

セッションの廃棄

セッション モードで次のコマンドを使用して、セッションを廃棄します。

コマンド	目的

abort
例:
`switch(config-s)# abort`
`switch#`
	コマンドを適用しないで、コンフィギュレーションセッションを廃棄します。

Session Manager 設定の確認

Session Manager のコンフィギュレーション情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show configuration session [*name*]	コンフィギュレーション ファイルの内容を表示します。
show configuration session status [*name*]	コンフィギュレーション セッションのステータスを表示します。

コマンド	目的
show configuration session summary	すべてのコンフィギュレーション セッションのサマリーを表示します。

Session Manager のコンフィギュレーション例

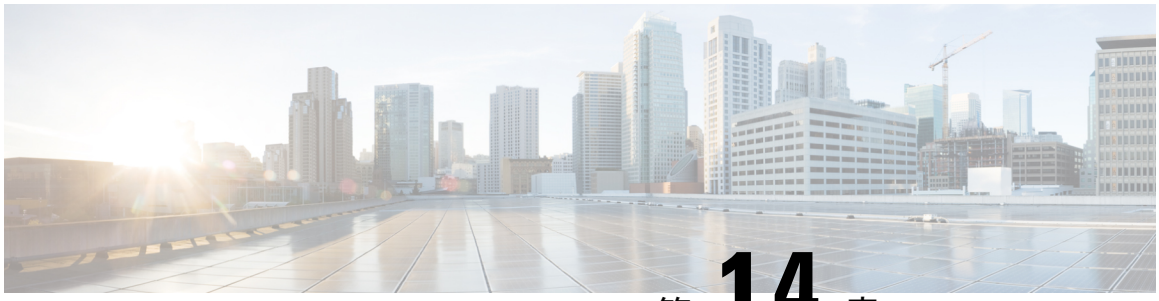
Session Manager を使用して ACL コンフィギュレーションを作成し、コミットする例を示します。

```
switch# configure session ACL_tcp_in
Config Session started, Session ID is 1
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-s)# ip access-list ACL1
switch(config-s-acl)# permit tcp any any
switch(config)# interface e 7/1
switch(config-if)# ip access-group ACL1 in
switch(config-if)# exit
switch(config)# exit
switch# config session ACL_tcp_in
Config Session started, Session ID is 1
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-s)# verify
Verification Successful
switch(config-s)# commit
Commit Successful
switch#
```

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
コンフィギュレーション ファイル	『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』



第 14 章

スケジューラの設定

この章では、Cisco NX-OS デバイス上でスケジューラを設定する方法について説明します。

この章は、次の項で構成されています。

- スケジューラについて （257 ページ）
- スケジューラの前提条件 （258 ページ）
- スケジューラの注意事項および制約事項 （259 ページ）
- スケジューラのデフォルト設定 （259 ページ）
- スケジューラの設定 （260 ページ）
- スケジューラの設定確認 （268 ページ）
- スケジューラの設定例 （268 ページ）

スケジューラについて

スケジューラを使用すると、次のようなメンテナンス作業のタイムテーブルを定義し、設定することができます。

- Quality of Service（QoS）ポリシーの変更
- データのバックアップ
- 設定の保存

ジョブは、定期的な作業を定義する単一または複数のコマンドで構成されています。ジョブは、1 回だけ、または定期的な間隔でスケジューリングすることができます。

スケジューラでは、ジョブと、そのタイムテーブルを次のように定義できます。

- ジョブ：コマンドリストとして定義され、特定のスケジュールに従って実行される定期的なタスク。
- スケジュール：ジョブを実行するタイムテーブル1 つのスケジュールに複数のジョブを割り当てることができます。1 つのスケジュールは、定期的、または 1 回だけ実行するように定義されます。

- 定期モード：ジョブを削除するまで、ジョブの実行が定期的な間隔で繰り返されます。次のタイプの定期的な間隔を設定できます。
  - Daily：ジョブは 1 日 1 回実行されます。
  - Weekly：ジョブは毎週 1 回実行されます。
  - Monthly：ジョブは毎月 1 回実行されます。
  - Delta：ジョブは、指定した時間に開始され、以後、指定した間隔（days:hours:minutes）で実行されます。
- 1 回限定モード：ジョブは、指定した時間に 1 回だけ実行されます。

リモート ユーザ認証

ジョブの開始前に、スケジューラはジョブを作成したユーザを認証します。リモート認証で得たユーザ クレデンシャルは短時間しか保有されないため、スケジューリングされたジョブをサポートできません。ジョブを作成するユーザの認証パスワードをローカルで設定する必要があります。これらのパスワードは、スケジューラのコンフィギュレーションに含まれ、ローカル設定のユーザとは見なされません。

ジョブを開始する前に、スケジューラはローカル パスワードとリモート認証サーバに保存されたパスワードを照合します。

ログ

スケジューラはジョブ出力を含むログファイルを管理します。ジョブ出力のサイズがログ ファイルのサイズより大きい場合、出力内容は切り捨てられます。

高可用性

スケジューリングされたジョブは、スーパバイザのスイッチオーバーまたはソフトウェアのリロード後も使用可能です。

スケジューラの前提条件

スケジューラの前提条件は次のとおりです。

- 条件付き機能をイネーブルにしてからでなければ、ジョブでそれらの機能を設定できません。
- ライセンスの必要な機能をジョブで設定するには、各機能の有効なライセンスをインストールしておく必要があります。
- スケジュールリングされたジョブを設定するには、network-admin のユーザ権限が必要です。

スケジューラの注意事項および制約事項

スケジューラに関する設定時の注意事項および制約事項は、次のとおりです。

- ジョブの実行中に次のいずれかの状況が発生した場合、スケジューラは失敗する可能性があります。
  - 時刻が設定されていることを確認します。スケジューラはデフォルトのタイムテーブルを適用しません。スケジュールを作成し、ジョブを割り当てても、時刻を設定しなければ、ジョブは開始しません。
  - ジョブは開始されると非インタラクティブ方式で実行されるため、ジョブの定義中、インタラクティブなコマンドや中断を伴うコマンド（例：**copy bootflash:** *file* **ftp:** *URI*、**write erase**、**reload** その他類似のコマンド）が指定されていないことを確認してください。特定の時間にリロードジョブがスケジュールされ、実行されると、スイッチはブート ループに入ります。したがって、スケジューラ構成では使用しないでください。
- スケジューラは、スケジュール モード設定で **time** コマンドの繰り返しオプションを使用して、任意のスケジュールの過去の `start_time` を承認します。次に、入力された開始時刻が過去であることを示す警告がスローされます。任意のスケジュールの `start_time` は、リブート後、および以前に保存された設定を再適用した後でも、最初と同じままです。
- Cisco NX-OS リリース 9.3(5) 以降では、スケジューラ ジョブ設定 CLI の出力に 2 番目のスペースが含まれています。

  以前は、出力にはジョブ設定 CLI の前に 1 つのスペースしかありませんでした。

```
scheduler job name show_fds.
 show clock >> bootflash:show_fds
^ (single space)
```

  ジョブ設定 CLI の前に 2 つのスペースがあります。

```
scheduler job name show_fds.
  show clock >> bootflash:show_fds
^^ (two spaces)
```

  設定の置換、ISSU、リロードなどの NX-OSソフトウェアのスケジューラ機能には影響しません。ただし、スケジューラ コンポーネント設定を読み取るための show run コマンドの出力を読み取るためにスクリプトを使用する場合は、スクリプト内のロジックを更新して、余分なスペースを確保する必要があります。

スケジューラのデフォルト設定

この表は、スケジューラのデフォルト設定を示します。

パラメータ	デフォルト
スケジューラの状態	ディセーブル
ログ ファイル サイズ	16 KB

スケジューラの設定

スケジューラの有効化または無効化

ジョブを設定してスケジュールできるようにスケジューラ機能を有効にすることができ、または、スケジューラを有効にした後にスケジューラ機能を無効にすることもできます。

手順の概要

1. **configure terminal**
2. [**no**] **feature scheduler**
3. （任意） **show scheduler config**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **feature scheduler** 例： `switch(config)# feature scheduler`	スケジューラを有効または無効にします。

ステップ 3	（任意） **show scheduler config**
	例：
	`switch(config)# show scheduler config`
	`config terminal`
	`feature scheduler`
	`scheduler logfile size 16`
	`end`
		スケジューラ設定を表示します。

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

	コマンドまたはアクション	目的
	`switch(config)# copy running-config startup-config`	

スケジューラ ログ ファイル サイズの定義

ジョブ、スケジュール、およびジョブ出力をキャプチャするログ ファイルのサイズを設定できます。

手順の概要

1. **configure terminal**
2. **scheduler logfile size** *value*
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**scheduler logfile size** *value* 例： `switch(config)# scheduler logfile size 1024`	スケジューラ ログ ファイル サイズをキロバイト（KB）で定義します。範囲は 16 ～ 1024 です。デフォルトは 16 です。 （注） ジョブ出力のサイズがログ ファイルのサイズより大きい場合、出力内容は切り捨てられます。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

リモート ユーザ認証の設定

ジョブの設定およびスケジューリングを行うユーザにリモート認証を使用するように、スケジューラを設定できます。

（注） リモート ユーザは、ジョブを作成および設定する前に、クリア テキスト パスワードを使用して認証する必要があります。

（注） **show running-config** コマンドの出力では、リモート ユーザ パスワードは常に暗号化された状態で表示されます。コマンドの暗号化オプション（**7**）は、ASCII デバイス設定をサポートします。

手順の概要

1. **configure terminal**
2. **scheduler aaa-authentication password** [**0** | **7**] *password*
3. **scheduler aaa-authentication username** *name* **password** [**0** | **7**] *password*
4. （任意） **show running-config** | **include "scheduler aaa-authentication"**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**scheduler aaa-authentication password** [**0** | **7**] *password* 例： `switch(config)# scheduler aaa-authentication password X12y34Z56a`	現在ログインしているユーザのクリアテキストパスワードを設定します。
ステップ 3	**scheduler aaa-authentication username** *name* **password** [**0** | **7**] *password* 例： `switch(config)# scheduler aaa-authentication username newuser password Z98y76X54b`	リモート ユーザのクリア テキスト パスワードを設定します。
ステップ 4	（任意） **show running-config** | **include "scheduler aaa-authentication"** 例： `switch(config)# show running-config | include "scheduler aaa-authentication"`	スケジューラのパスワード情報を表示します。

	コマンドまたはアクション	目的
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ジョブの定義

ジョブを定義して、ジョブ名とコマンド シーケンスを指定することができます。

⚠

注意 一旦ジョブを定義すると、コマンドの変更、削除はできません。ジョブを変更するには、そのジョブを削除して新しいジョブを作成する必要があります。

手順の概要

1. **configure terminal**
2. **scheduler job name** *string*
3. *command1* ;[*command2* ;*command3* ;...]
4. （任意） **show scheduler job** [**name** *name*]
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**scheduler job name** *string*
	例：
	`switch(config)# scheduler job name backup-cfg`
	`switch(config-job)`
		ジョブを作成し、ジョブ コンフィギュレーション モードを開始します。
		「backup-cfg」という名前のスケジューラ ジョブを作成する例を示します。

ステップ 3	*command1* ;[*command2* ;*command3* ;...]
	例：
	`switch(config-job)# copy running-config tftp://1.2.3.4/$(SWITCHNAME)-cfg.$(TIMESTAMP) vrf management`
	`switch(config-job)#`
		特定のジョブに対応するコマンド シーケンスを定義します。複数のコマンドは、スペースとセミコロンで（「;」のように）区切る必要があります。
		この例では、実行コンフィギュレーションをブートフラッシュ内のファイルに保存するスケジュール

	コマンドまたはアクション	目的
		ジョブを作成しています。その後ジョブはブートフラッシュから TFTP サーバにファイルをコピーし、現在のタイムスタンプとスイッチ名を使用してファイル名を作成します。
ステップ 4	（任意） **show scheduler job** [**name** *name*] 例： `switch(config-job)# show scheduler job`	ジョブ情報を表示します。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

ジョブの削除

スケジューラからジョブを削除できます。

手順の概要

1. **configure terminal**
2. **no scheduler job name** *string*
3. （任意） **show scheduler job** [**name** *name*]
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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 scheduler job name** *string*
	例：
	`switch(config)# no scheduler job name configsave`
	`switch(config-job)`
		特定のジョブおよびそこで定義されたすべてのコマンドを削除します。

ステップ 3	（任意） **show scheduler job** [**name** *name*] 例：	ジョブ情報を表示します。

	コマンドまたはアクション	目的
	`switch(config-job)# show scheduler job name configsave`	
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

タイムテーブルの定義

1 つまたは複数のジョブで使用するタイムテーブルをスケジューラで定義できます。

time コマンドで時刻を設定しない場合は、スケジューラは現在の時刻を使用します。たとえば、現在の時刻が 2013 年 3 月 24 日の 22 時 00 分である場合、ジョブは次のように開始されます。

- スケジューラは、**time start 23:00 repeat 4:00:00** コマンドの開始時刻が、2013 年 3 月 24 日 23 時 00 分であると見なします。
- スケジューラは、**time daily 55** コマンドの開始時刻が、毎日 22 時 55 分であると見なします。
- スケジューラは、**time weekly 23:00** コマンドの開始時刻が、毎週金曜日の 23 時 00 分であると見なします。
- スケジューラは、**time monthly 23:00** コマンドの開始時刻が、毎月 24 日の 23 時 00 分であると見なします。

（注） スケジューラは、1 つ前のジョブが完了しない限り、次のジョブを開始しません。たとえば、1 分間隔で実行するジョブを 22 時 00 分に開始するようジョブをスケジューリングしたが、ジョブを完了するには 2 分間必要である場合、ジョブは次のように実行されます。スケジューラは 22 時 00 分に最初のジョブを開始し、22 時 02 分に完了します。次に 1 分間待機し、22 時 03 分に次のジョブを開始します。

手順の概要

1. **configure terminal**
2. **scheduler schedule name** *string*
3. **job name** *string*
4. **time daily** *time*
5. **time weekly** [[*dow*:]*HH*:]*MM*
6. **time monthly** [[*dm*:]*HH*:]*MM*
7. **time start** {**now repeat** *repeat-interval* | *delta-time* [**repeat** *repeat-interval*]}
8. （任意） **show scheduler config**
9. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**scheduler schedule name** *string*
	例：
	`switch(config)# scheduler schedule name weekendbackupqos`
	`switch(config-schedule)#`
		スケジュールを作成し、スケジュール 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	**job name** *string* 例： `switch(config-schedule)# job name offpeakZoning`	このスケジュールにジョブを関連付けます。1 つのスケジュールに複数のジョブを追加できます。
ステップ 4	**time daily** *time* 例： `switch(config-schedule)# time daily 23:00`	ジョブが毎日 HH:MM の形式で指定された時刻に開始することを意味します。
ステップ 5	**time weekly** [[*dow*:]*HH*:]*MM* 例： `switch(config-schedule)# time weekly Sun:23:00`	ジョブが週の指定された曜日に開始することを意味します。 曜日（dow）は次のいずれかの方法で指定されます。 • 曜日を表す整数。たとえば 1 = 日曜日、2 = 月曜日。 • 曜日の省略形。たとえば Sun = Sunday。 引数全体の最大長は 10 です。
ステップ 6	**time monthly** [[*dm*:]*HH*:]*MM* 例： `switch(config-schedule)# time monthly 28:23:00`	ジョブが月の特定の日（dm）に開始することを意味します。29、30 または 31 のいずれかを指定した場合、そのジョブは各月の最終日に開始されます。
ステップ 7	**time start** {**now repeat** *repeat-interval* | *delta-time* [**repeat** *repeat-interval*]} 例： `switch(config-schedule)# time start now repeat 48:00`	ジョブが定期的に開始することを意味します。 start-time の形式は [[[[yyyy:]mmm:]dd:]HH]:MM です。 • *delta-time*：スケジュールの設定後、ジョブの開始までの待機時間を指定します。 • **now**：ジョブを今すぐ開始するよう指定します。

	コマンドまたはアクション	目的
		• **repeat** *repeat-interval*：ジョブを反復する回数を指定します。 この例では、ただちにジョブが開始され、48時間間隔で反復されます。
ステップ 8	（任意） **show scheduler config** 例： `switch(config)# show scheduler config`	スケジューラ設定を表示します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

スケジューラ ログ ファイルの消去

スケジューラ ログ ファイルを消去できます。

手順の概要

1. **configure terminal**
2. **clear scheduler logfile**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**clear scheduler logfile** 例： `switch(config)# clear scheduler logfile`	スケジューラ ログ ファイルを消去します。

スケジューラの設定確認

スケジューラの設定情報を表示するには、次のタスクのいずれかを行います。

コマンド	目的
show scheduler config	スケジューラ設定を表示します。
show scheduler job [**name** *string*]	設定されているジョブを表示します。
show scheduler logfile	スケジューラ ログ ファイルの内容を表示します。
show scheduler schedule [**name** *string*]	設定されているスケジュールを表示します。

スケジューラの設定例

スケジューラ ジョブの作成

この例では、実行コンフィギュレーションをブートフラッシュ内のファイルに保存するスケジュール ジョブを作成する方法を示します。このジョブは、その後で、ブートフラッシュからTFTP サーバにファイルをコピーします（現在のタイムスタンプとスイッチ名を使用してファイル名を作成します）。

```
switch# configure terminal
switch(config)# scheduler job name backup-cfg
switch(config-job)# copy running-config
tftp://1.2.3.4/$(SWITCHNAME)-cfg.$(TIMESTAMP) vrf management
switch(config-job)# end
switch(config)#
```

スケジューラ ジョブのスケジューリング

次に、backup-cfg という名前のスケジューラ ジョブを、毎日午前1時に実行するようスケジューリングする例を示します。

```
switch# configure terminal
switch(config)# scheduler schedule name daily
switch(config-if)# job name backup-cfg
switch(config-if)# time daily 1:00
switch(config-if)# end
switch(config)#
```

ジョブ スケジュールの表示

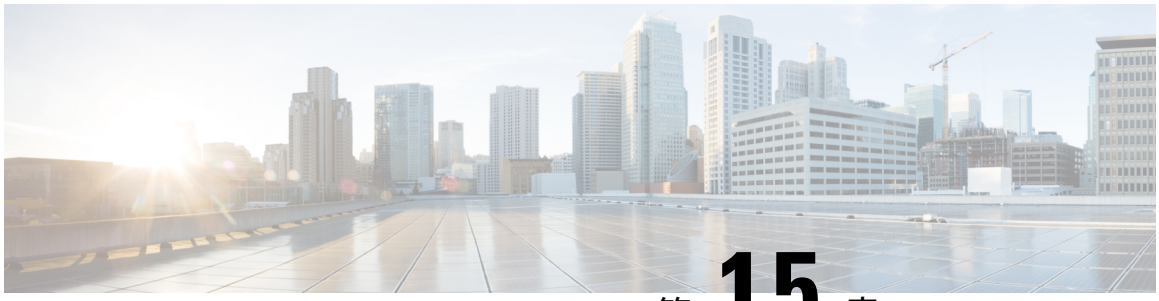
次に、ジョブ スケジュールを表示する例を示します。

```
switch# show scheduler schedule
Schedule Name : daily
---------------------------
User Name : admin
Schedule Type : Run every day at 1 Hrs 00 Mins
Last Execution Time : Fri Jan 2 1:00:00 2013
Last Completion Time: Fri Jan 2 1:00:01 2013
Execution count : 2
-----------------------------------------------
Job Name Last Execution Status
-----------------------------------------------
back-cfg Success (0)
switch#
```

スケジューラ ジョブの実行結果の表示

次に、スケジューラによって実行されたスケジューラ ジョブの結果を表示する例を示します。

```
switch# show scheduler logfile
Job Name : back-cfg Job Status: Failed (1)
Schedule Name : daily User Name : admin
Completion time: Fri Jan 1 1:00:01 2013
-------------------------------- Job Output ---------------------------------
`cli var name timestamp 2013-01-01-01.00.00`
`copy running-config bootflash:/$(HOSTNAME)-cfg.$(timestamp)`
`copy bootflash:/switch-cfg.2013-01-01-01.00.00 tftp://1.2.3.4/ vrf management `
copy: cannot access file '/bootflash/switch-cfg.2013-01-01-01.00.00'
==============================================================================
Job Name : back-cfg Job Status: Success (0)
Schedule Name : daily User Name : admin
Completion time: Fri Jan 2 1:00:01 2013
-------------------------------- Job Output ---------------------------------
`cli var name timestamp 2013-01-02-01.00.00`
`copy running-config bootflash:/switch-cfg.2013-01-02-01.00.00`
`copy bootflash:/switch-cfg.2013--01-02-01.00.00 tftp://1.2.3.4/ vrf management `
Connection to Server Established.
[ ] 0.50KBTrying to connect to tftp server......
[###### ] 24.50KB
TFTP put operation was successful
==============================================================================
switch#
```



第 15 章

SNMP の設定

この章では、Cisco NX-OS デバイス上で SNMP 機能を設定する方法について説明します。

この章は、次の内容で構成されています。

- SNMP について, on page 271
- SNMP の注意事項および制約事項 （280 ページ）
- SNMP のデフォルト設定 （283 ページ）
- SNMP の設定 （283 ページ）
- SNMP ローカル エンジン ID の設定, on page 312
- SNMP の設定の確認, on page 313
- SNMP エンティティ （315 ページ）
- SNMP の設定例 （315 ページ）
- その他の参考資料 （317 ページ）

SNMP について

簡易ネットワーク管理プロトコル（SNMP）は、SNMP マネージャとエージェント間の通信用メッセージフォーマットを提供する、アプリケーションレイヤプロトコルです。SNMP では、ネットワーク内のデバイスのモニタリングと管理に使用する標準フレームワークと共通言語が提供されます。

SNMP 機能の概要

SNMP フレームワークは 3 つの部分で構成されます。

- SNMP マネージャ：SNMP を使用してネットワークデバイスのアクティビティを制御し、モニタリングするシステム
- SNMP エージェント：デバイスのデータを維持し、必要に応じてこれらのデータを管理システムに報告する、管理対象デバイス内のソフトウェア コンポーネント。Cisco Nexus デバイスはエージェントおよび MIB をサポートします。SNMP エージェントをイネーブルにするには、マネージャとエージェントの関係を定義する必要があります。

- MIB（Management Information Base; 管理情報ベース）：SNMP エージェントの管理対象オブジェクトの集まり

SNMP は、RFC 3411 ～ 3418 で規定されています。

デバイスは、SNMPv1、SNMPv2c、およびSNMPv3をサポートします。SNMPv1およびSNMPv2cはどちらも、コミュニティベース形式のセキュリティを使用します。

Cisco NX-OS は IPv6 による SNMP をサポートしています。

SNMP 通知

SNMP の重要な機能の 1 つは、SNMP エージェントから通知を生成できることです。これらの通知では、要求をSNMPマネージャから送信する必要はありません。通知は、不正なユーザ認証、再起動、接続の切断、隣接ルータとの接続の切断、その他の重要なイベントを表示します。

Cisco NX-OS は、トラップまたはインフォームとして SNMP 通知を生成します。トラップは、エージェントからホスト レシーバ テーブルで指定された SNMP マネージャに送信される、非同期の非確認応答メッセージです。インフォームは、SNMP エージェントから SNMP マネージャに送信される非同期メッセージで、マネージャは受信したという確認応答が必要です。

トラップの信頼性はインフォームより低くなります。SNMPマネージャはトラップを受信しても確認応答（ACK）を送信しないからです。デバイスは、トラップが受信されたかどうかを判断できません。インフォーム要求を受信するSNMPマネージャは、SNMP応答プロトコルデータ ユニット（PDU）でメッセージの受信を確認応答します。デバイスが応答を受信しない場合、インフォーム要求を再び送信できます。

複数のホスト レシーバーに通知を送信するよう Cisco NX-OS を設定できます。

次の表は、デフォルトで有効になっている SNMP トラップを示します。

Trap Type	説明
全体	: coldStart
エンティティ	: entity_fan_status_change
エンティティ	: entity_mib_change
エンティティ	: entity_module_status_change
エンティティ	: entity_module_inserted
エンティティ	: entity_module_removed
エンティティ	: entity_power_out_change
エンティティ	: entity_power_status_change
エンティティ	: entity_unrecognised_module
リンク	: cErrDisableInterfaceEventRev1

Trap Type	説明
リンク	: cieLinkDown
リンク	: cieLinkUp
リンク	: cmn-mac-move-notification
リンク	: delayed-link-state-change
リンク	: extended-linkDown
リンク	: extended-linkUp
リンク	: linkDown
リンク	: linkUp
rf	: redundancy_framework
upgrade	: UpgradeOpNotifyOnCompletion
upgrade	: UpgradeJobStatusNotify
エンティティ	: entity_sensor
rmon	: fallingAlarm
rmon	: hcRisingAlarm
rmon	: hcFallingAlarm
rmon	: risingAlarm

SNMPv3

SNMPv3は、ネットワーク経由のフレームの認証と暗号化を組み合わせることによって、デバイスへのセキュアアクセスを実現します。SNMPv3が提供するセキュリティ機能は次のとおりです。

- メッセージの完全性：パケットが伝送中に改ざんされていないことを保証します。
- 認証：メッセージのソースが有効かどうかを判別します。
- 暗号化：許可されていないソースにより判読されないように、パケットの内容のスクランブルを行います。

SNMPv3では、セキュリティモデルとセキュリティレベルの両方が提供されています。セキュリティモデルは、ユーザおよびユーザが属するロールを設定する認証方式です。セキュリティレベルとは、セキュリティモデル内で許可されるセキュリティのレベルです。セキュリティモデルとセキュリティレベルの組み合わせにより、SNMPパケット処理中に採用されるセキュリティメカニズムが決まります。

SNMPv1、SNMPv2、SNMPv3 のセキュリティ モデルおよびセキュリティ レベル

セキュリティ レベルは、SNMP メッセージを開示から保護する必要があるかどうか、およびメッセージを認証するかどうか判断します。セキュリティ モデル内のさまざまなセキュリティ レベルは、次のとおりです。

- noAuthNoPriv：認証または暗号化を実行しないセキュリティ レベル。このレベルは、SNMPv3 ではサポートされていません。
- authNoPriv：認証は実行するが、暗号化を実行しないセキュリティ レベル。
- authPriv：認証と暗号化両方を実行するセキュリティ レベル。

SNMPv1、SNMPv2c、および SNMPv3 の 3 つのセキュリティ モデルを使用できます。セキュリティ モデルとセキュリティ レベルの組み合わせにより、SNMP メッセージの処理中に適用されるセキュリティ メカニズムが決まります。次の表に、セキュリティ モデルとレベルの組み合わせの意味を示します。

***Table 14: SNMP* セキュリティ モデルおよびセキュリティ レベル**

モデル	レベル	認証	暗号化	結果
v1	noAuthNoPriv	コミュニティ ストリング	なし	コミュニティ ストリングの照合を使用して認証します。
v2c	noAuthNoPriv	コミュニティ ストリング	なし	コミュニティ ストリングの照合を使用して認証します。
v3	authNoPriv	HMAC-MD5、HMAC-SHA、または SHA-256	未対応	Hash-Based Message Authentication Code（HMAC）メッセージ ダイジェスト 5（MD5）アルゴリズムまたは HMAC Secure Hash Algorithm（SHA）アルゴリズムに基づいて認証します。

モデル	レベル	認証	暗号化	結果
v3	authPriv	HMAC-MD5、HMAC-SHA、または SHA-256	DES	HMAC-MD5 アルゴリズムまたは HMAC-SHA アルゴリズムに基づいて認証します。データ暗号規格（DES）の 56 ビット暗号化、および暗号ブロック連鎖（CBC）DES（DES-56）標準に基づいた認証を提供します。

ユーザベースのセキュリティ モデル

SNMPv3 ユーザベース セキュリティ モデル（USM）は SNMP メッセージレベル セキュリティを参照し、次のサービスを提供します。

- メッセージの完全性：メッセージが不正な方法で変更または破壊されず、データ シーケンスが悪意なく起こり得る範囲を超えて変更されていないことを保証します。
- メッセージ発信元の認証：受信データを発信したユーザのアイデンティティが確認されたことを保証します。
- メッセージの機密性：情報が使用不可であること、または不正なユーザ、エンティティ、またはプロセスに開示されないことを保証します。

SNMPv3 は、設定済みユーザによる管理動作のみを許可し、SNMP メッセージを暗号化します。

Cisco NX-OS は、SNMPv3 に 3 つの認証プロトコルを使用します。

- HMAC-MD5-96 認証プロトコル
- HMAC-SHA-96 認証プロトコル
- SHA-256 認証プロトコル

Cisco NX-OS リリース 9.3(7) 以降では、SNMPv3 に HMAC-SHA-256 認証プロトコルが使用されます。

Note SHA-256 SNMP ユーザがスイッチで設定されている場合、ISSD は **install all** コマンドを使用することを推奨します。そうしないと、設定が失われます。

Cisco NX-OS は、SNMPv3 メッセージ暗号化用プライバシー プロトコルの 1 つとして、Advanced Encryption Standard（AES）を使用し、RFC 3826 に準拠します。

priv オプションで、SNMP セキュリティ暗号化方式として、DES または 128 ビット AES 暗号化を選択できます。**priv** オプションおよび **aes-128** トークンは、128 ビットの AES キーを生成するためのプライバシ パスワードであることを示します。AES のプライバシー パスワードは最小で 8 文字です。パスフレーズをクリアテキストで指定する場合は、大文字と小文字を区別して、最大 64 文字の英数字を指定できます。ローカライズドキーを使用する場合は、最大 130 文字を指定できます。

Note 外部の AAA サーバを使用して SNMPv3 を使う場合、外部 AAA サーバのユーザー設定でプライバシー プロトコルに AES を指定する必要があります。

CLI および SNMP ユーザの同期

SNMPv3 ユーザ管理は、Access Authentication and Accounting（AAA）サーバ レベルで集中化できます。この中央集中型ユーザ管理により、Cisco NX-OSの SNMP エージェントは AAA サーバのユーザ認証サービスを利用できます。ユーザ認証が検証されると、SNMP PDU の処理が進行します。AAA サーバはユーザ グループ名の格納にも使用されます。SNMP はグループ名を使用して、スイッチでローカルに使用できるアクセス ポリシーまたはロール ポリシーを適用します。

ユーザ グループ、ロール、またはパスワードの設定が変更されると、SNMP と AAA の両方のデータベースが同期化されます。

Cisco NX-OS は、次のようにユーザ設定を同期化します。

- **snmp-server user** コマンドで指定された認証パスフレーズは、CLI ユーザのパスワードになります。
- **username** コマンドで指定されたパスワードが SNMP ユーザの認証およびプライバシー パスフレーズになります。
- SNMP または CLI を使用してユーザを作成または削除すると、SNMP と CLI の両方でユーザが作成または削除されます。
- ユーザとロールの対応関係の変更は、SNMP と CLI で同期化されます。
- CLI から行ったロール変更（削除または変更）は、SNMP と同期します。

Note パスフレーズまたはパスワードをローカライズしたキーおよび暗号形式で設定した場合、Cisco NX-OS はユーザ情報（パスワードやロールなど）を同期させません。

Cisco NX-OS はデフォルトで、同期したユーザ設定を 60 分間維持します。

セキュリティおよび SNMP ユーザーの同期の無効化

Cisco NX-OS リリース 10.2(2)F 以降、SNMP とセキュリティ（AAA または CLI）コンポーネント間のユーザー同期を無効にするオプションを提供するために、次の同期解除コマンドが導入されました。

 # snmp-server disable snmp-aaa sync

このコマンドは、Nexus スイッチの構成端末から実行できます。デフォルトでは、desynchronization コマンドの **no** 形式がスイッチで使用できます。

デバイスで同期解除コマンドの no 形式が有効になっている場合、たとえば `switch (config)#` **no snmp-server disable snmp-aaa sync** の場合には、実行構成におけるそのユーザーの **username** 作成で、 **snmp-server user** CLI の結果を利用してユーザーを作成することができます。また、逆も可能です。したがって、ユーザーは、作成/更新時に **snmp-server user** CLI または **username** CLI に記載されている認証資格情報を使用してスイッチにログインできます。スイッチのネットワーク マネージャから SNMP 操作を実行することもできます。したがって、desynchronization コマンドの **no** 形式を使用すると、SNMP と AAA 間のユーザー同期は、10.2(2)F より前のリリースと同じように機能します。

デバイスで同期解除コマンドが有効になっている場合、たとえば `switch (config)#` **snmp-server disable snmp-aaa sync** の場合には、**snmp-server user** コマンドによって作成されたユーザーに対し、ユーザー名構成は作成されません。したがって、ユーザーはスイッチにログインできず、スイッチのネットワーク マネージャを介して SNMP 操作を実行することのみが許可されます。同様に、**username** CLI を使用してセキュリティ ユーザーを作成しても、そのユーザーに対応する **snmp-server user** CLI は作成されません。このユーザーはスイッチにログインできますが、スイッチで SNMP 操作を実行することはできません。これは、desynchronization コマンドによってリリース 10.2(2)F から導入された新機能です。

非同期コマンドのステータスは、次のいずれかの方法で確認できます：

- CLI **show snmp internal globals** の出力にある `SNMP-AAA sync disable` フィールドの値
- `sys/snmp/inst/globals` MO のフィールド `disableSnmpAaaSync` の値
- コマンドが有効か無効かに応じて、CLI は **show-running-config** 出力および **show-running-config-snmp** 出力または **show-running-all** 出力にそれぞれ出力します。

リモートユーザー

RADIUS や TACACS+ などのプロトコルを使用して外部サーバー経由でログイン認証されているリモートユーザーに関しては、スイッチで同期解除コマンドが有効になっている場合、SNMP でリモート ユーザーを作成できません。詳細については、*Cisco Nexus 9000 NX-OS Security Configuration Guide* の *AAA* を構成するの章を参照してください。

ただし、スイッチで desynchronization コマンドの **no** 形式が有効になっている場合、リモートユーザが AAA で作成されると、対応するユーザが SNMP でも作成されます。さらに、ユーザーは SNMP の running-config 出力には表示されませんが、管理対象デバイスで SNMP 操作を実行できます。これは、リリース 10.2(2)F より前からの既存の機能です。

DCNM セキュリティ ユーザー

desynchronization コマンドが有効になっている場合、DCNM（リリース 12.0.1a 以降は Nexus Dashboard Fabric Controller とも呼ばれます）を使用して作成されたセキュリティ ユーザーには、対応する SNMPv3 プロファイルがありません。同期が無効になっている場合、セキュリティ コンポーネントで作成されたユーザーはスイッチにログインできますが、コントローラはスイッチを検出しません。コントローラは、セキュリティ ユーザー用に作成された SNMP 構成を使用してスイッチを検出するためです。さらに、SNMP は、userDB の非同期状態のため、作成されたセキュリティ ユーザーを認識しないので、スイッチを検出できません。したがって、コントローラによってスイッチが検出されるようにするには、SNMP ユーザーを明示的に作成する必要があります。DCNM 機能とともに desynchronization コマンドを使用することはお勧めしません。詳細については、*Cisco Nexus 9000 NX-OS Security Configuration Guide* を参照してください。

ISSD と ISSU

一般に、SNMP ユーザーの同期が無効になっている場合は、非同期のユーザーをすべて削除しない限り、SNMP ユーザーの同期を有効にしないでください。このような組み合わせの実行コンフィギュレーションでは、設定の置換が失敗します。

古いリリースで、同期解除コマンドを使用せずに同期解除状態を実現する唯一の方法は、次のとおりです。

- 同期解除状態のリリースから、同期解除コマンドの存在しないリリースへ、中断を伴う/伴わない ISSD を実行します。同期解除されたデータベースは、ISSD により以前のリリースにそのまま取り込まれます。

（注） そのような ISSD の後にユーザー データベースに加えられた変更は、SNMP とセキュリティ コンポーネントの間で同期されます。

このような ISSD の後、同期解除コマンドの存在するリリースへの ISSU を実行すると、同期解除されたユーザー データベースがそのまま取り込まれます。一方、同期解除コマンドはデフォルトの **no** 形式で起動します。必要に応じて、同期解除コマンドを有効にしてください。

グループベースの SNMP アクセス

Note グループが業界全体で使用されている標準 SNMP 用語なので、この SNMP の項では、ロールのことをグループと言います。

SNMP アクセス権は、グループ別に編成されます。SNMP 内の各グループは、CLI を使用する場合のロールに似ています。各グループは読み取りアクセス権または読み取りと書き込みアクセス権を指定して定義します。

ユーザ名が作成され、ユーザのロールが管理者によって設定され、ユーザがそのロールに追加されていれば、そのユーザはエージェントとの通信を開始できます。

SNMP および Embedded Event Manager

Embedded Event Manager（EEM）機能は、SNMP MIB オブジェクトを含むイベントをモニタし、これらのイベントに基づいてアクションを開始します。SNMP 通知の送信もアクションの 1 つです。EEM は SNMP 通知として、CISCO-EMBEDDED-EVENT-MGR-MIB の cEventMgrPolicyEvent を送信します。

マルチ インスタンス サポート

デバイスは、プロトコルインスタンスや仮想ルーティングおよびフォワーディング（VRF）インスタンスなどの論理ネットワーク エンティティの複数のインスタンスをサポートできます。大部分の既存 MIB は、これら複数の論理ネットワーク エンティティを識別できません。たとえば、元々の OSPF-MIB ではデバイス上のプロトコル インスタンスが 1 つであることが前提になりますが、現在はデバイス上で複数の OSPF インスタンスを設定できます。

SNMPv3 ではコンテキストを使用して、複数のインスタンスを識別します。SNMP コンテキストは管理情報のコレクションであり、SNMP エージェントを通じてアクセスできます。デバイスは、さまざまな論理ネットワーク エンティティの複数のコンテキストをサポートできます。SNMP コンテキストによって、SNMP マネージャはさまざまな論理ネットワーク エンティティに対応するデバイス上でサポートされる、MIB モジュールの複数のインスタンスの 1 つにアクセスできます。

Cisco NX-OS は、SNMP コンテキストと論理ネットワーク エンティティ間のマッピングのために、CISCO-CONTEXT-MAPPING-MIB をサポートします。SNMP コンテキストは VRF、プロトコル インスタンス、またはトポロジに関連付けることができます。

SNMPv3 は、SNMPv3 PDU の contextName フィールドでコンテキストをサポートします。この contextName フィールドを特定のプロトコルインスタンスまたは VRF にマッピングできます。

SNMPv2c の場合は、SNMP-COMMUNITY-MIB の snmpCommunityContextName MIB オブジェクトを使用して、SNMP コミュニティをコンテキストにマッピングできます（RFC 3584）。さらに CISCO-CONTEXT-MAPPING-MIB または CLI を使用すると、この snmpCommunityContextName を特定のプロトコル インスタンスまたは VRF にマッピングできます。

SNMP のハイ アベイラビリティ

Cisco NX-OS は、SNMP のステートレス リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバーの後、Cisco NX-OS は実行コンフィギュレーションを適用します。

SNMP の仮想化サポート

Cisco NX-OS は、SNMP のインスタンスを 1 つサポートします。SNMP は複数の MIB モジュールインスタンスをサポートし、それらを論理ネットワーク エンティティにマッピングします。

SNMP も VRF を認識します。特定の VRF を使用して、SNMP 通知ホスト レシーバに接続するように SNMP を設定できます。通知が発生した VRF に基づいて、SNMP ホスト レシーバへの通知をフィルタリングするように SNMP を設定することもできます。

SNMP の注意事項および制約事項

SNMP には、次の注意事項および制限事項があります。

- SNMP SET を使用して構成されたコマンドは、SNMP SET のみを使用して削除する必要があります。コマンド ライン インターフェイス（CLI）または NX-API を使用して構成されたコマンドは、CLI または NX-API のみを使用して削除する必要があります。
- クリア テキスト パスワードを使用して AAA でユーザーを作成または編集すると、SNMP はデフォルトの認証（md5）および priv タイプを持つユーザーを作成または編集します。

  クリア テキスト パスワードを使用して SNMP でユーザーを作成または編集すると、AAA はデフォルトのパスワード タイプ（タイプ 5）を持つユーザーを作成または編集します。
- アクセス コントロール リスト（ACL）は、スイッチに設定されたローカル SNMPv3 ユーザのみに適用できます。ACL は、認証、許可、アカウンティング（AAA）サーバに保存されるリモート SNMPv3 ユーザに適用できません。
- 同期解除されたすべてのユーザが削除されない限り、SNMP ユーザ同期を無効にした後は有効にしないでください。このような組み合わせの実行コンフィギュレーションでは、設定の置換が失敗します。
- Cisco NX-OS は、一部の SNMP MIB への読み取り専用アクセスをサポートします。詳細については次の URL にアクセスして、Cisco NX-OS の MIB サポート リストを参照してください。https://cisco.github.io/cisco-mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html
- Cisco NX-OS は、SNMPv3 noAuthNoPriv セキュリティ レベルをサポートしていません。
- Cisco Nexus 9000 シリーズ スイッチと、Cisco Nexus 3164Q、31128PQ、3232C、3264Q スイッチは、SNMP ローカル エンジンID の設定をサポートしています。
- 以前のリリースへの無停止ダウングレード パスを行う場合、ローカル エンジン ID を設定していたなら、ローカル エンジン ID の設定を戻してから、SNMP ユーザとコミュニティ文字列を再設定する必要があります。
- 特殊文字 @ および % は、SNMP コミュニティ ストリングでは使用できません。
- デフォルトの SNMP PDU 値は 1500 バイトです。SNMP エージェントは、1500 バイトを超える応答 PDU をドロップするので、SNMP リクエストは失敗します。1500 バイトを超える MIB データ値を受信するには、**snmp-server packetsize** <*byte-count*>コマンドを使用し

て、パケット サイズを再設定します。有効なバイト数の範囲は 484 ー 17382 です。GETBULK応答がパケットサイズを超えると、データが切り捨てられることがあります。

- スイッチの機能を設定するには、CLIまたはSNMPを使用する必要があります。スイッチに、両方のインターフェイスを使用して機能を設定しないでください。
- シャーシにファンが装着されていない個々のファン OID ツリーでcefcFanTrayOperStatus snmpwalk を使用すると、ツリー内の次の OID エントリに対する応答が返されることがあります。この動作を防ぐには、*snmpwalk* で -CI オプションを使用します。

  この動作は、親OIDをポーリングする場合、またはgetmanyを使用する場合には見られません。
- Cisco Nexus 9000 シリーズ スイッチは、*snmpwalk* 要求に対して最大 10000 個のフラッシュ ファイルをサポートします。
- SNMP トラップが完全で適切な機能動作を実行するには、少なくとも 1 つの実行中の BGP インスタンスが必要です。snmp-server traps 関連のコマンドを設定する前に、BGP ルーティングインスタンスを設定します。
- リリース 10.1(1) 以降、AES-128 は強力な暗号化アルゴリズムであるため、推奨される暗号化アルゴリズムです。ただし、DES 暗号化もサポートされています。

  ダウングレード：DES プライバシー プロトコルを持つユーザが SNMP データベースに存在する場合、**install all** コマンドによる In-Service System Downgrade（ISSD）が中断されます。ユーザは（デフォルトの AES-128 を使用して）再設定または削除する必要があります。コールド リブートの場合、DES を持つ SNMP ユーザは削除されます。
- SNMPユーザーの構成後にエンジン識別子を構成する場合は、次のアクションを実行してください。
  - エンジン識別子を変更した後、SNMPユーザーと、グループ、ACLを含む関連構成をパスワードとともに再構成します。これにより、認証の失敗と、ユーザーに関連付けられた ACL およびグループへの影響が回避できます。
- Cisco NX-OS リリース 10.3(1)F 以降 、SNMP（MIB – 400G Optic MIB、スイッチ MIB、データパス MIB、インターフェイス MIB）が Cisco Nexus 9808 プラットフォーム スイッチでサポートされています。
- SVI 統計情報は、SNMP キャッシュに対して 120 秒ごとにのみポーリングされます。
- Cisco NX-OS リリース 10.3(3)F 以降では、SNMPv3 ユーザー パスワードのタイプ 6 暗号化が次の制限付きでサポートされています。
  - タイプ 6 暗号化は、次の点に注意した場合にのみ成功します。
    - **feature password encryption aes {tam}** がイネーブルになっていること。
    - プライマリ キーが構成されていること。
    - **pwd_type 6** オプションは、SNMPv3 ユーザーの構成時に指定されます。

- プライマリ キーの構成を変更すると、SNMP はデータベースに保存されているすべてのタイプ 6 ユーザーを再暗号化します。ただし、SNMP 機能は以前と同じように動作します。
- プライマリ キーの設定は、スイッチに対してローカルです。ユーザーが 1 つのスイッチからタイプ 6 で構成された実行データを取得し、別のプライマリ キーが構成されている別のスイッチに適用すると、同じユーザーの SNMP 機能が別のスイッチでは動作しない可能性があります。
- タイプ 6 が設定されている場合は、タイプ 6 がサポートされていないリリースにダウングレードする前に、構成を削除するか、タイプ 6 オプションを再構成してください。
- ISSU の場合、以前のイメージ（localizedkey、localizedV2key 構成が存在する）からタイプ 6 暗号化がサポートされている新しいイメージに移行すると、SNMP は既存のキーをタイプ 6 暗号化に変換しません。
- 既存の SALT 暗号化からタイプ 6 暗号化への変換は、**encryption re-encrypt obfuscated** コマンドを使用してサポートされます。
- 中断を伴うアップグレードや **reload-ascii** コマンドによる ASCII ベースのリロードを実行すると、プライマリ キーが失われ、タイプ 6 ユーザーの SNMP 機能に影響を与えます。
- ユーザーが **encryption re-encrypt obfuscated** コマンドを使用して再暗号化を強制すると、SNMP はタイプ 6 以外の SNMP ユーザーからのすべてのパスワードをタイプ 6 モードに暗号化します。

（注） SNMP は **encryption delete type6** コマンドをサポートしていません。同じことを示す syslog 警告メッセージも表示されます。

- Cisco NX-OS リリース 10.4(1)F 以降では、SNMP を使用して Electronic Programmable Logic Device（EPLD）ファームウェア バージョンを表示できます。エンティティ MIB 構造の一部として、ファームウェア バージョン、EPLD デバイスのタイプ（IO または MI FPGA）、および EPLD デバイスの親エンティティ（スーパーバイザ、ライン カード、ライン カード拡張モジュール（LEM）など）を表示できます。この機能は、Cisco Nexus 9300-FX/FX2/FX3/GX プラットフォーム スイッチ、N9K-C9332D-H2R スイッチ、および Nexus 9508 スイッチでサポートされています。
- Cisco NX-OS リリース 10.4(1)F 以降 、SNMP（MIB – 400G Optic MIB、スイッチ MIB、データパス MIB、インターフェイス MIB）は、次のライン カードとスイッチでサポートされています。
  - Cisco Nexus 9804 スイッチ
  - Cisco Nexus 9332D-H2R スイッチ

- Cisco Nexus X98900CD-A および X9836DM-A ラインカードと Cisco Nexus 9808 および 9804 スイッチ

SNMP のデフォルト設定

次の表に、SNMP パラメータのデフォルト設定を示します。

パラメータ	デフォルト
ライセンス通知	有効（Enabled）

SNMP の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

（注） Cisco NX-OS リリース 9.3(7) 以降、HMAC-SHA-256 認証プロトコルは SNMPv3 に使用されます。

SNMP ユーザーの構成

SNMP ユーザを設定できます。

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server user** *name* [**pwd_type 6**] [**auth** {**md5** | **sha** | **sha-224** | **sha-256** | **sha-384** | **sha-512**} *passphrase* [**auto**] [**priv [aes-128]** *passphrase*] [**engineID** *id*] [**localizedkey**] | [**localizedV2key**]]
3. (Optional) **show snmp user**
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
ステップ 1	**configure terminal** **Example:**	グローバル コンフィギュレーション モードを開始します

	Command or Action	Purpose
	`switch# configure terminal` `switch(config)#`	
ステップ 2	**snmp-server user** *name* [**pwd_type 6**] [**auth** {**md5** | **sha** | **sha-224** | **sha-256** | **sha-384** | **sha-512**} *passphrase* [**auto**] [**priv [aes-128]** *passphrase*] [**engineID** *id*] [**localizedkey**] | [**localizedV2key**]] **Example:** `switch(config)# snmp-server user Admin pwd_type 6 auth sha abcd1234 priv abcdefgh`	認証およびプライバシー パラメータのある SNMP ユーザを設定します。パスフレーズには最大 64 文字の英数字を使用できます。大文字と小文字が区別されます。 **localizedkey** キーワードを使用する場合は、パスフレーズに大文字と小文字を区別した英数字を 130 文字まで使用できます。 **sha** ：認証に HMAC SHA-1 アルゴリズムを使用します。 **sha-224** ：認証に HMAC SHA-224 アルゴリズムを使用します。 **sha-256** ：認証に HMAC SHA-256 アルゴリズムを使用します。 **sha-384** ：認証に HMAC SHA-384 アルゴリズムを使用します。 **sha-512** ：認証に HMAC SHA-512 アルゴリズムを使用します。 **localizedkey** - **localizedkey**キーワードを使用する場合は、パスフレーズに大文字と小文字を区別した英数字を 130 文字まで使用できます。[プレーン テキスト パスワードの代わりに、localizedkey キーワードを使用して**ハッシュされた** パスワード（**show running config**コマンドからコピーするか、snmpv3 ベースのオープン ソース ハッシュ ジェネレーター ツールを使用してオフラインで生成したもの、ハッシュ化されたパスワードをオフラインで生成する, on page 285を参照）を構成できます。 **Note** ローカライズされたキーを使用する場合は、ハッシュ値の前に `0x` を追加します（例: `0x84a716329158a97ac9f22780629bc26c`）。 **localizedV2key** - localizedV2key キーを使用する場合、パスフレーズは大文字と小文字を区別した、最大 130 文字の英数字文字列にすることができます。先頭に 0x を付ける必要はありません。これは暗号化されたデータであり、オフラインでは生成できないため、**show run** コマンドを使用して localizedv2key を収集します。

	Command or Action	Purpose
		engineID の形式は、12 桁のコロンで区切った 10 進数字です。 **Note** • Cisco NX-OS リリース 10.1(1) 以降、AES-128 は SNMPv3 のデフォルトのプライバシー プロトコルです。 • Cisco NX-OS リリース 10.3(3)F 以降では、SNMP ユーザー パスワードにタイプ 6 暗号化を提供するために **pwd_type 6** キーワードがサポートされています。
ステップ 3	(Optional) **show snmp user** **Example:** `switch(config)# show snmp user`	1 人または複数の SNMP ユーザに関する情報を表示します。
ステップ 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ハッシュ化されたパスワードをオフラインで生成する

snmpv3 ベースのオープン ソース ハッシュ ジェネレータ ツールを使用して、ハッシュ化されたパスワードをオフラインで生成する手順は、次のとおりです。

（注） 例としてい挙げられている ID はサンプルの ID で、手順を説明するためだけのものです。

1. スイッチから SNMP engineID を取得します。

   switch# **show snmp engineID**

   サンプル出力：

   ```
   Local SNMP engineID: [Hex] 8000000903D4C93CEA31CC
   [Dec] 128:000:000:009:003:212:201:060:234:049:204
   ```

2. SNMPv3 ベースのオープン ソース ハッシュ ジェネレータを使用して、ハッシュ化されたパスワードをオフラインで生成します。

   Linux$ **snmpv3-hashgen --auth** Hello123 --engine 8000000903D4C93CEA31CC --user1 **--mode priv --hash md5**

   サンプル出力：

   ```
   User: user1
   Auth: Hello123 / 84a716329158a97ac9f22780629bc26c
   ```

```
Priv: Hello123 / 84a716329158a97ac9f22780629bc26c
Engine: 8000000903D4C93CEA31CC
ESXi USM String:
u1/84a716329158a97ac9f22780629bc26c/84a716329158a97ac9f22780629bc26c/priv
```

3. auth および priv の値を使用して、スイッチのパスワードを構成します。

   snmp-server user user1 **auth md5** 0x84a716329158a97ac9f22780629bc26c **priv des 0x**84a716329158a97ac9f22780629bc26c **localizedkey**

SNMP メッセージ暗号化の適用

着信要求に認証または暗号化が必要となるよう SNMP を設定できます。デフォルトでは、SNMP エージェントは認証および暗号化を行わないでも SNMPv3 メッセージを受け付けます。プライバシーを適用する場合、Cisco NX-OS は、**noAuthNoPriv** または **authNoPriv** のいずれかのセキュリティ レベル パラメータを使用するすべての SNMPv3 PDU 要求に対して、許可エラーで応答します。

手順の概要

1. **configure terminal**
2. **snmp-server user** *name* **enforcePriv**
3. **snmp-server globalEnforcePriv**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**snmp-server user** *name* **enforcePriv** 例： `switch(config)# snmp-server user Admin enforcePriv`	このユーザに対して SNMP メッセージ暗号化を適用します。
ステップ 3	**snmp-server globalEnforcePriv** 例： `switch(config)# snmp-server globalEnforcePriv`	すべてのユーザに対して SNMP メッセージ暗号化を適用します。
ステップ 4	（任意） **copy running-config startup-config** 例：	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

	コマンドまたはアクション	目的
	`switch(config)# copy running-config startup-config`	

SNMPv3 ユーザに対する複数のロールの割り当て

SNMP ユーザを作成した後で、そのユーザに複数のロールを割り当てることができます。

（注） 他のユーザにロールを割り当てることができるのは、network-admin ロールに属するユーザだけです。

手順の概要

1. **configure terminal**
2. **snmp-server user** *name group*
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**snmp-server user** *name group* 例： `switch(config)# snmp-server user Admin superuser`	この SNMP ユーザと設定されたユーザ ロールをアソシエートします。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP コミュニティの作成

SNMPv1 または SNMPv2c の SNMP コミュニティを作成できます。

手順の概要

1. **configure terminal**
2. **snmp-server community** *name* {**group** *group* | **ro** | **rw**}
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**snmp-server community** *name* {**group** *group*	**ro**	**rw**} 例： `switch(config)# snmp-server community public ro`	SNMP コミュニティ ストリングを作成します。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP 要求のフィルタリング

アクセス コントロール リスト（ACL）を SNMPv2 コミュニティに割り当てて、SNMP 要求にフィルタを適用できます。割り当てたACLにより着信要求パケットが許可される場合、SNMPはその要求を処理します。ACLにより要求が拒否される場合、SNMPはその要求を廃棄して、システム メッセージを送信します。

ACL は次のパラメータで作成します。

- 送信元 IP アドレス
- 宛先 IP アドレス
- 送信元ポート
- 宛先ポート
- プロトコル（UDP または TCP）

手順の概要

1. **configure terminal**

2. **snmp-server community** *name* [**use-ipv4acl** *acl-name*]
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**snmp-server community** *name* [**use-ipv4acl** *acl-name*] 例： `switch(config)# snmp-server community public use-ipv4acl myacl`	SNMP コミュニティに IPv4 ACL ACL を割り当てて SNMPv2 要求をフィルタします。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP 通知レシーバの設定

複数のホスト レシーバーに対して SNMP 通知を生成するよう Cisco NX-OSを設定できます。

手順の概要

1. **configure terminal**
2. **snmp-server host** *ip-address* **traps version 1** *community* [**udp_port** *number*]
3. **snmp-server host** *ip-address* {**traps** | **informs**} **version 2c** *community* [**udp_port** *number*]
4. **snmp-server host** *ip-address* {**traps** | **informs**} **version 3** {**auth** | **noauth** | **priv**} *username* [**udp_port** *number*]
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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
	`switch# configure terminal` `switch(config)#`	
ステップ 2	**snmp-server host** *ip-address* **traps version 1** *community* [**udp_port** *number*] 例： `switch(config)# snmp-server host 192.0.2.1 traps version 1 public`	SNMPv1 トラップのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。コミュニティは、最大 255 文字の英数字で指定できます。UDP ポート番号の範囲は 0 ～ 65535 です。
ステップ 3	**snmp-server host** *ip-address* {**traps** | **informs**} **version 2c** *community* [**udp_port** *number*] 例： `switch(config)# snmp-server host 192.0.2.1 informs version 2c public`	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。コミュニティは、最大 255 文字の英数字で指定できます。UDP ポート番号の範囲は 0 ～ 65535 です。
ステップ 4	**snmp-server host** *ip-address* {**traps** | **informs**} **version 3** {**auth** | **noauth** | **priv**} *username* [**udp_port** *number*] 例： `switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS`	SNMPv3 トラップまたは応答要求のホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。ユーザ名は、最大 255 文字の英数字で指定できます。UDP ポート番号の範囲は 0 ～ 65535 です。 （注） SNMP マネージャは SNMPv3 メッセージを認証して解読するために、Cisco NX-OS デバイスの SNMP engineID に基づいてユーザ クレデンシャル（authKey/PrivKey）を調べる必要があります。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP 通知用の発信元 インターフェイスの設定

通知の送信元 IP アドレスとしてインターフェイスの IP アドレスを使用するよう、SNMP を設定できます。通知が生成される場合、送信元 IP アドレスは、この設定済みインターフェイスの IP アドレスに基づいています。

次のように発信元インターフェイスを設定できます。

- すべての通知が、すべての SNMP 通知レシーバへ送信される。
- すべての通知が、特定の SNMP 通知レシーバへ送信される。このコンフィギュレーションは、グローバル発信元インターフェイスのコンフィギュレーションよりも優先されます。

（注） 発信トラップパケットの送信元インターフェイスIPアドレスを設定すると、デバイスがトラップの送信に同じインターフェイスを使用することが保証されません。送信元インターフェイス IP アドレスは、SNMP トラップの内部で送信元アドレスを定義し、出力インターフェイス アドレスを送信元として接続が開きます。

手順の概要

1. **configure terminal**
2. **snmp-server host** *ip-address* **source-interface** *if-type if-number* **traps version 2c** *name*
3. **snmp-server host** *ip-address* **source-interface** *if-type if-number* **use-vrf** *vrf-name*
4. **snmp-server host** *ip-address* **source-interface** *if-type if-number* [**udp_port** *number*]
5. **snmp-server source-interface** {**traps** | **informs**} *if-type if-number*
6. **show snmp source-interface**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**snmp-server host** *ip-address* **source-interface** *if-type if-number* **traps version 2c** *name* 例： `snmp-server host 192.0.2.1 source-interface ethernet 2/1 traps version 2c public`	（任意）このホストにトラップ メッセージを送信します。 トラップのバージョンには、通知メッセージに使用する SNMP バージョンを指定します。2c は、SNMPv2c が使用されることを示します。
ステップ 3	**snmp-server host** *ip-address* **source-interface** *if-type if-number* **use-vrf** *vrf-name* 例： `snmp-server host 192.0.2.1 source-interface ethernet 2/1 use-vrf default`	特定の VRF を使用してホスト レシーバと通信するように SNMP を設定します。ip-address は IPv4 または IPv6 アドレスにできます。VRF 名は、最大 32 文字の英数字で指定できます。 （注） このコマンドによってホスト設定は削除されません。
ステップ 4	**snmp-server host** *ip-address* **source-interface** *if-type if-number* [**udp_port** *number*] 例：	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。サポートされているイン

	コマンドまたはアクション	目的
	`switch(config)# snmp-server host 192.0.2.1 source-interface ethernet 2/1`	ターフェイス タイプを特定するために「**?**」を使用します。UDP ポート番号の範囲は 0 ～ 65535 です。 このコンフィギュレーションは、グローバル発信元インターフェイスのコンフィギュレーションよりも優先されます。
ステップ 5	**snmp-server source-interface** {**traps** | **informs**} *if-type if-number* 例： `switch(config)# snmp-server source-interface traps ethernet 2/1`	SNMPv2c トラップまたは応答要求を送信するよう発信元インターフェイスを設定します。サポートされているインターフェイス タイプを特定するために「**?**」を使用します。
ステップ 6	**show snmp source-interface** 例： `switch(config)# show snmp source-interface`	設定した発信元インターフェイスの情報を表示します。

通知ターゲット ユーザの設定

SNMPv3 インフォーム通知を通知ホスト レシーバに送信するには、デバイスに通知ターゲット ユーザを設定する必要があります。

Cisco NX-OS は通知ターゲット ユーザのクレデンシャルを使用して、設定された通知ホスト レシーバへの SNMPv3 応答要求通知メッセージを暗号化します。

（注） 受信した INFORM PDU を認証して解読する場合、Cisco NX-OS で設定されているのと同じ、応答要求を認証して解読するユーザ クレデンシャルが通知ホスト レシーバに必要です。

手順の概要

1. **configure terminal**
2. **snmp-server user** *name* [**auth** {**md5** | **sha** | **sha-256**} *passphrase* [**auto**] [**priv** *passphrase*] [**engineID** *id*]
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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
	`switch# configure terminal` `switch(config)#`	
ステップ 2	**snmp-server user** *name* [**auth** {**md5** | **sha** | **sha-256**} *passphrase* [**auto**] [**priv** *passphrase*] [**engineID** *id*] 例： `switch(config)# snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID 00:00:00:63:00:01:00:10:20:15:10:03`	通知ホスト レシーバのエンジン ID を指定して、通知ターゲット ユーザを設定します。エンジン ID の形式は、12 桁のコロンで区切った 10 進数字です。 （注） リリース 10.1(1) 以降、AES-128 は SNMPv3 のデフォルトのプライバシー プロトコルです。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

VRF を使用する SNMP 通知レシーバの設定

SNMP 通知レシーバの VRF 到達可能性およびフィルタリング オプションを設定すると、SNMP によって CISCO-SNMP-TARGET-EXT-MIB の cExtSnmpTargetVrfTable にエントリが追加されます。

（注） VRF 到達可能性またはフィルタリング オプションを設定する前に、ホストを設定する必要があります。

ホスト レシーバに到達するように設定した VRF を使用したり、または通知が発生した VRF に基づいて通知をフィルタするように Cisco NX-OS を設定できます。

手順の概要

1. **configure terminal**
2. [**no**] **snmp-server host** *ip-address* **use-vrf** *vrf-name* [**udp_port** *number*]
3. [**no**] **snmp-server host** *ip-address* **filter-vrf** *vrf-name* [**udp_port** *number*]
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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
	`switch# configure terminal` `switch(config)#`	
ステップ 2	[**no**] **snmp-server host** *ip-address* **use-vrf** *vrf-name* [**udp_port** *number*] 例： `switch(config)# snmp-server host 192.0.2.1 use-vrf Blue`	特定の VRF を使用してホスト レシーバと通信するように SNMP を設定します。*ip-address* を IPv4 または IPv6 アドレスにできます。VRF 名には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。このコマンドによって、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable にエントリが追加されます。 このコマンドの **no** 形式は、設定されたホストの VRF 到達可能性情報を削除し、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable からエントリを削除します。 （注） このコマンドによってホスト設定は削除されません。
ステップ 3	[**no**] **snmp-server host** *ip-address* **filter-vrf** *vrf-name* [**udp_port** *number*] 例： `switch(config)# snmp-server host 192.0.2.1 filter-vrf Red`	設定された VRF に基づいて、通知ホスト レシーバへの通知をフィルタリングします。*ip-address* は IPv4 または IPv6 アドレスを使用できます。VRF 名には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。 このコマンドによって、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable にエントリが追加されます。 このコマンドの **no** 形式は、設定されたホストの VRF フィルタ情報を削除し、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable からエントリを削除します。 （注） このコマンドによってホスト設定は削除されません。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

帯域内ポートを使用してトラップを送信するための SNMP 設定

帯域内ポートを使用してトラップを送信するよう SNMP を設定できます。このようにするには、（グローバルまたはホスト レベルで）発信元インターフェイスを設定し、トラップを送信するための VRF を設定します。

手順の概要

1. **configure terminal**
2. **snmp-server source-interface traps** *if-type if-number*
3. （任意） **show snmp source-interface**
4. **snmp-server host** *ip-address* **use-vrf** *vrf-name* [**udp_port** *number*]
5. （任意） **show snmp host**
6. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**snmp-server source-interface traps** *if-type if-number* 例： `switch(config)# snmp-server source-interface traps ethernet 1/2`	SNMP トラップを送信するための発信元インターフェイスをグローバルに設定します。サポートされているインターフェイス タイプを特定するために「?」を使用します。 グローバル レベルまたはホスト レベルで発信元インターフェイスを設定できます。発信元インターフェイスをグローバルに設定すると、新しいホスト コンフィギュレーションはグローバルなコンフィギュレーションを使用してトラップを送信します。 （注） 発信元インターフェイスをホスト レベルで設定するには、**snmp-server host** *ip-address* **source-interface** *if-type if-number* コマンドを使用します。
ステップ 3	（任意） **show snmp source-interface** 例： `switch(config)# show snmp source-interface`	設定した発信元インターフェイスの情報を表示します。

	コマンドまたはアクション	目的
ステップ 4	**snmp-server host** *ip-address* **use-vrf** *vrf-name* [**udp_port** *number*] 例： `switch(config)# snmp-server host 171.71.48.164 use-vrf default`	特定の VRF を使用してホスト レシーバと通信するように SNMP を設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。VRF 名には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。このコマンドによって、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable にエントリが追加されます。 （注） デフォルトでは、SNMP は管理 VRF を使用してトラップを送信します。管理 VRF を使用しない場合は、このコマンドを使用して対象の VRF を指定する必要があります。
ステップ 5	（任意） **show snmp host** 例： `switch(config)# show snmp host`	設定した SNMP ホストの情報を表示します。
ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP 通知のイネーブル化

通知をイネーブルまたはディセーブルにできます。通知名を指定しなかった場合、Cisco NX-OS は、BGP、EIGRP、および OSPFの通知を除き、通知をすべてイネーブルにします。

Note **snmp-server enable traps** コマンドを使用すると、設定通知ホスト レシーバによっては、トラップとインフォームの両方をイネーブルにできます。

次の表に、Cisco NX-OS MIB の通知を有効にするコマンドを示します。

***Table 15: SNMP* 通知のイネーブル化**

MIB	関連コマンド
すべての通知（BGP、EIGRP、およびOSPFを除く）	**snmp-server enable traps**
CISCO-AAA-SERVER-MIB	**snmp-server enable traps aaa** **snmp-server enable traps aaa server-state-change**

MIB	関連コマンド
CISCO-BGP4-MIB	**snmp-server enable traps bgp**
CISCO-CALLHOME-MIB	**snmp-server enable traps callhome** **snmp-server enable traps callhome event-notify** **snmp-server enable traps callhome smtp-send-fail**
CISCO-CONFIG-MAN-MIB	**snmp-server enable traps config** **snmp-server enable traps config ccmCLIRunningConfigChanged**
CISCO-EIGRP-MIB	**snmp-server enable traps eigrp** [*tag*]
CISCO-ERR-DISABLE-MIB	**snmp-server enable traps link cerrDisableInterfaceEventRev1**
ENTITY-MIB、CISCO-ENTITY-SENSOR-MIB	**snmp-server enable traps entity** **snmp-server enable traps entity entity_fan_status_change** **snmp-server enable traps entity entity_mib_change** **snmp-server enable traps entity entity_module_inserted** **snmp-server enable traps entity entity_module_removed** **snmp-server enable traps entity entity_module_status_change** **snmp-server enable traps entity entity_power_out_change** **snmp-server enable traps entity entity_power_status_change** **snmp-server enable traps entity entity_unrecognised_module**
CISCO-FEATURE-CONTROL-MIB	**snmp-server enable traps feature-control** **snmp-server enable traps feature-control FeatureOpStatusChange**
CISCO-HSRP-MIB	**snmp-server enable traps hsrp** **snmp-server enable traps hsrp state-change**

MIB	関連コマンド
IF-MIB	**snmp-server enable traps link** **snmp-server enable traps link extended-linkDown** **snmp-server enable traps link extended-linkUp** **snmp-server enable traps link cieLinkDown** **snmp-server enable traps link cieLinkUp** **snmp-server enable traps link linkDown** **snmp-server enable traps link linkUp**
OSPF-MIB, OSPF-TRAP-MIB	**snmp-server enable traps ospf** [*tag*] **snmp-server enable traps ospf lsa** **snmp-server enable traps ospf rate-limit** *rate*
CISCO-RF-MIB	**snmp-server enable traps rf** **snmp-server enable traps rf redundancy_framework**
CISCO-RMON-MIB	**snmp-server enable traps rmon** **snmp-server enable traps rmon fallingAlarm** **snmp-server enable traps rmon hcFallingAlarm** **snmp-server enable traps rmon hcRisingAlarm** **snmp-server enable traps rmon risingAlarm**
SNMPv2-MIB	**snmp-server enable traps snmp** **snmp-server enable traps snmp authentication**
CISCO-MAC-NOTIFICATION-MIB	**snmp-server enable trap link cmn-mac-move-notification**
CISCO-PORT-STORM-CONTROL-MIB	**storm-control action trap**
CISCO-STP-EXTENSIONS-MIB	**snmp-server enable traps stpx stpxMstInconsistencyUpdate**
CISCO-STP-BRIDGE-MIB	**snmp-server enable traps bridge** **snmp-server enable traps bridge newroot** **snmp-server enable traps bridge topologychange**

MIB	関連コマンド
CISCO-STPX-MIB	**snmp-server enable traps stpx** **snmp-server enable traps stpx inconsistency** **snmp-server enable traps stpx loop-inconsistency** **snmp-server enable traps stpx root-inconsistency**
CISCO-SYSTEM-EXT-MIB	**snmp-server enable traps sysmgr** **snmp-server enable traps sysmgr cseFailSwCoreNotifyExtended**
UPGRADE-MIB	**snmp-server enable traps upgrade** **snmp-server enable traps upgrade UpgradeJobStatusNotify** **snmp-server enable traps upgrade UpgradeOpNotifyOnCompletion**
VTP-MIB	**snmp-server enable traps vtp** **snmp-server enable traps vtp notifs** **snmp-server enable traps vtp vlancreate** **snmp-server enable traps vtp vlandelete**

指定した通知を有効にするには、示しているようにコンフィギュレーションモードで次のコマンドを使用します。

コマンド	目的
snmp-server enable traps **例:** `switch(config)# snmp-server enable traps`	すべての SNMP 通知をイネーブルにします。
snmp-server enable traps aaa [**server-state-change**] **例:** `switch(config)# snmp-server enable traps aaa`	AAA SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **server-state-change**：AAA サーバの状態変化通知を有効にします。
snmp-server enable traps bgp **例:** `switch(config)# snmp-server enable traps bgp`	ボーダー ゲートウェイ プロトコル（BGP）SNMP 通知を有効にします。

コマンド	目的
snmp-server enable traps bridge [**newroot**] [**topologychange**] **例:** `switch(config)# snmp-server enable traps bridge`	STP ブリッジ SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **newroot**：STP の新しいルートブリッジ通知を有効にします。 • **topologychange**：STP ブリッジのトポロジ変更通知を有効にします。
snmp-server enable traps callhome [**event-notify**] [**smtp-send-fail**] **例:** `switch(config)# snmp-server enable traps callhome`	Call Home 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **event-notify**：Call Home の外部イベント通知を有効にします。 • **smtp-send-fail**：簡易メール転送プロトコル（SMTP）メッセージの送信失敗通知を有効にします。
snmp-server enable traps config [**ccmCLIRunningConfigChanged**] **例:** `switch(config)# snmp-server enable traps config`	コンフィギュレーションの変更に対して SNMP 通知をイネーブルにします。 • **ccmCLIRunningConfigChanged**：実行中または起動時のコンフィギュレーションで、コンフィギュレーションの変更に対して SNMP 通知を有効にします。
snmp-server enable traps eigrp [*tag*] **例:** `switch(config)# snmp-server enable traps eigrp`	CISCO-EIGRP-MIB SNMP 通知をイネーブルにします。

コマンド	目的
snmp-server enable traps entity [**entity_fan_status_change**] [**entity_mib_change**] [**entity_module_inserted**] [**entity_module_removed**] [**entity_module_status_change**] [**entity_power_out_change**] [**entity_power_status_change**] [**entity_unrecognised_module**] **例:** `switch(config)# snmp-server enable traps entity`	ENTITY-MIB SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **entity_fan_status_change**：エンティティ ファンの状態変化通知を有効にします。 • **entity_mib_change**：エンティティ MIB 変更通知を有効にします。 • **entity_module_inserted**：エンティティ モジュール挿入通知を有効にします。 • **entity_module_removed**：エンティティ モジュール削除通知を有効にします。 • **entity_module_status_change**：エンティティ モジュール ステータス変更通知を有効にします。 • **entity_power_out_change**：エンティティの出力パワー変更通知を有効にします。 • **entity_power_status_change**：エンティティのパワー ステータス変更通知を有効にします。 • **entity_unrecognised_module**：エンティティの未確認モジュール通知を有効にします。
snmp-server enable traps feature-control [**FeatureOpStatusChange**] **例:** `switch(config)# snmp-server enable traps feature-control`	機能制御 SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **FeatureOpStatusChange**：機能操作の状態変化通知を有効にします。
snmp-server enable traps hsrp state-change **例:** `switch(config)# snmp-server enable traps hsrp`	CISCO-HSRP-MIB SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **state-change**：HSRP の状態変化通知を有効にします。

コマンド	目的
snmp-server enable traps license [**notify-license-expiry**] [**notify-license-expiry-warning**] [**notify-licensefile-missing**] [**notify-no-license-for-feature**] **例:** `switch(config)# snmp-server enable traps license`	ENTITY-MIB SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **notify-license-expiry**：ライセンス失効通知を有効にします。 • **notify-license-expiry-warning**：ライセンス失効の警告通知を有効にします。 • **notify-licensefile-missing**：ライセンス ファイル不明通知を有効にします。 • **notify-no-license-for-feature**：no-license-installed-for-feature 通知を有効にします。
snmp-server enable traps link [**cieLinkDown**] [**cieLinkUp**] [**cmn-mac-move-notification**] [**IETF-extended-linkDown**] [**IETF-extended-linkUp**] [**cisco-extended-linkDown**] [**cisco-extended-linkUp**][**linkDown**] [**linkUp**] **例:** `switch(config)# snmp-server enable traps link`	IF-MIB リンク通知をイネーブルにします。任意で、以下の特定の通知をイネーブルにします。 • **IETF-extended-linkDown**：Cisco 拡張リンク ステート ダウン通知をイネーブルにします。 • **IETF-extended-linkUp**：Cisco 拡張リンク ステート アップ通知をイネーブルにします。 • **cmn-mac-move-notification**：MACアドレス移動通知をイネーブルにします。 • **cisco-extended-linkDown**—：Internet Engineering Task Force（インターネット エンジニアリング タスク フォース、IETF）の拡張リンク ステート ダウン通知をイネーブルにします。 • **cisco-extended-linkUP**：Internet Engineering Task Force（IETF）の拡張リンク ステート アップ通知をイネーブルにします。 • **linkDown**：IETF リンク ステート ダウン通知を有効にします。 • **linkUp**：IETF リンク ステート アップ通知を有効にします。

コマンド	目的
snmp-server enable traps ospf [*tag*] [**lsa**] **例:** `switch(config)# snmp-server enable traps ospf`	Open Shortest Path First（OSPF）通知を有効にします。任意で、次の特定の通知をイネーブルにします。 • **lsa**：OSPF リンク ステート アドバタイズメント（LSA）通知を有効にします。
snmp-server enable traps rf [**redundancy-framework**] **例:** `switch(config)# snmp-server enable traps rf`	冗長フレームワーク（RF）SNMP通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **redundancy-framework**：RF スーパーバイザ スイッチオーバー MIB 通知を有効にします。
snmp-server enable traps rmon [**fallingAlarm**] [**hcFallingAlarm**] [**hcRisingAlarm**] [**risingAlarm**] **例:** `switch(config)# snmp-server enable traps rmon`	リモート モニタリング（RMON）SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **fallingAlarm**：RMON 下限アラーム通知を有効にします。 • **hcFallingAlarm**：RMON high-capacity 下限アラーム通知を有効にします。 • **hcRisingAlarm**：RMON high-capacity 上限アラーム通知を有効にします。 • **risingAlarm**：RMON 上限アラーム通知を有効にします。
snmp-server enable traps snmp [**authentication**] **例:** `switch(config)# snmp-server enable traps snmp`	一般的な SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **authentication**：SNMP 認証通知を有効にします。

コマンド	目的
snmp-server enable traps stpx[**inconsistency**] [**loop-inconsistency**] [**root-inconsistency**] **例:** `switch(config)# snmp-server enable traps stpx`	SNMP STPX 通知を有効にします。任意で、次の特定の通知をイネーブルにします。 • **inconsistency**：SNMP STPX MIB 不一致アップデート通知を有効にします。 • **loop-inconsistency**：SNMP STPX MIB ループ不一致アップデート通知を有効にします。 • **root-inconsistency**：SNMP STPX MIB ルート不一致アップデート通知を有効にします。
snmp-server enable traps syslog [**message-generated**] **例:** `switch(config)# snmp-server enable traps syslog`	定義された SNMP ホストに syslog メッセージをトラップとして送信します。任意で、次の特定の通知をイネーブルにします。 • **message-generate**：ソフトウェア ログ メッセージ生成通知を有効にします。
snmp-server enable traps sysmgr [**cseFailSwCoreNotifyExtended**] **例:** `switch(config)# snmp-server enable traps sysmgr`	ソフトウェア変更通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **cseFailSwCoreNotifyExtended**：ソフトウェア コア通知を有効にします。
snmp-server enable traps upgrade [**UpgradeJobStatusNotify**] [**UpgradeOpNotifyOnCompletion**] **例:** `switch(config)# snmp-server enable traps upgrade`	アップグレード通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **UpgradeJobStatusNotify**：アップグレード ジョブ ステータス通知を有効にします。 • **UpgradeOpNotifyOnCompletion**：アップグレード グローバル ステータス通知を有効にします。

コマンド	目的
snmp-server enable traps vtp[**notifs**] [**vlancreate**] [**vlandelete**] **例:** `switch(config)# snmp-server enable traps vtp`	VTP 通知を有効にします。任意で、次の特定の通知をイネーブルにします。 • **notifs**：VTP 通知を有効にします。 • **vlancreate**：VLAN 作成の通知を有効にします。 • **vlandelete**：VLAN 削除の通知を有効にします。
storm-control action traps **例:** `switch(config-if)# storm-control action traps`	トラフィック ストーム制御の制限に達した場合のトラフィック ストーム制御通知を有効にします。

インターフェイスでのリンク通知のディセーブル化

個別のインターフェイスで linkUp および linkDown 通知をディセーブルにできます。フラッピング インターフェイス（Up と Down の間を頻繁に切り替わるインターフェイス）で、この制限通知を使用できます。

手順の概要

1. **configure terminal**
2. **interface** *type slot*/*port*
3. **no snmp trap link-status**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface** *type slot*/*port* 例： `switch(config)# interface ethernet 2/2`	インターフェイスの SNMP リンクステート トラップをディセーブルにします。このコマンドは、デフォルトでイネーブルになっています。

	コマンドまたはアクション	目的
ステップ 3	**no snmp trap link-status** 例： `switch(config-if)# no snmp trap link-status`	インターフェイスの SNMP リンクステート トラップをディセーブルにします。このコマンドは、デフォルトでイネーブルになっています。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

インターフェイスの SNMP ifIndex の表示

SNMP ifIndex は、関連するインターフェイス情報をリンクするために複数の SNMP MIB にわたって使用されます。

手順の概要

1. **show interface snmp-ifindex**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**show interface snmp-ifindex**
	例：
	`switch# show interface snmp-ifindex | grep -i Eth12/1`
	`Eth12/1 441974784 (0x1a580000)`
		すべてのインターフェイスについて、IF-MIB から永続的な SNMP ifIndex 値を表示します。任意で、**|** キーワードと **grep** キーワードを使用すると、出力で特定のインターフェイスを検索できます。

TCP による SNMP のワンタイム認証の有効化

TCP セッション上で SNMP に対するワンタイム認証をイネーブルにできます。

手順の概要

1. **configure terminal**
2. **snmp-server tcp-session** [**auth**]
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**snmp-server tcp-session** [**auth**] 例： `switch(config)# snmp-server tcp-session`	TCP セッション上で SNMP に対するワンタイム認証をイネーブルにします。デフォルトではディセーブルになっています。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

SNMP スイッチのコンタクト（連絡先）およびロケーション情報の指定

32 文字までの長さで（スペースを含まない）デバイスのコンタクト情報とデバイスのロケーションを指定できます。

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server contact** *name*
3. **snmp-server location** *name*
4. (Optional) **show snmp**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

	Command or Action	Purpose
ステップ 2	**snmp-server contact** *name* **Example:** `switch(config)# snmp-server contact Admin`	SNMP コンタクト名として sysContact を設定します。
ステップ 3	**snmp-server location** *name* **Example:** `switch(config)# snmp-server location Lab-7`	SNMP ロケーションとして sysLocation を設定します。
ステップ 4	(Optional) **show snmp** **Example:** `switch(config)# show snmp`	1 つまたは複数の宛先プロファイルに関する情報を表示します。
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

コンテキストとネットワーク エンティティ間のマッピング設定

プロトコル インスタンス、VRF などの論理ネットワーク エンティティに対する SNMP コンテキストのマッピングを設定できます。

Before you begin

論理ネットワーク エンティティのインスタンスを決定します。VRF およびプロトコル インスタンスの詳細については、『Cisco Nexus 9000 シリーズ NX-OS ユニキャスト ルーティング設定ガイド』または『Cisco Nexus 9000 シリーズ NX-OS マルチキャスト ルーティング設定ガイド』を参照してください。

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]
3. (Optional) **snmp-server mib community-map** *community-name* **context** *context-name*
4. (Optional) **show snmp context**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*] **Example:** `switch(config)# snmp-server context public1 vrf red`	SNMP コンテキストをプロトコル インスタンス、VRF、またはトポロジにマッピングします。名前には最大 32 の英数字を使用できます。 **no** オプションは、SNMP コンテキストとプロトコル インスタンス、VRF、またはトポロジ間のマッピングを削除します。 **Note** コンテキスト マッピングを削除する目的で、インスタンス、VRF、またはトポロジを入力しないでください。インスタンス、vrf、またはトポロジ キーワードを使用すると、コンテキストとゼロ長ストリング間のマッピングが設定されます。
ステップ 3	(Optional) **snmp-server mib community-map** *community-name* **context** *context-name* **Example:** `switch(config)# snmp-server mib community-map public context public1`	SNMPv2c コミュニティを SNMP コンテキストにマッピングします。名前には最大 32 の英数字を使用できます。
ステップ 4	(Optional) **show snmp context** **Example:** `switch(config)# show snmp context`	1 つまたは複数の SNMP コンテキストに関する情報を表示します。
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP のディセーブル化

デバイスの SNMP を無効にできます。

手順の概要

1. **configure terminal**

2. **no snmp-server protocol enable**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**no snmp-server protocol enable** 例： `switch(config)# no snmp-server protocol enable`	SNMP をディセーブルにします。SNMP はデフォルトでイネーブルになっています。 （注） SNMPv2 を無効にせずに SNMPv1 を無効にすることはできません。SNMPv1 を無効にする場合は、SNMPv3 のみを設定するか、SNMP を完全に無効にします。

SNMP サーバ カウンタ キャッシュ更新タイマーの管理

Cisco NX-OS がキャッシュ ポートの状態を保持する時間は、秒単位で変更できます。

手順の概要

1. **configure terminal**
2. **snmp-server counter cache timeout** *seconds*
3. （任意） **show running-config snmp all |i cac**
4. **no snmp-server counter cache enable**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**snmp-server counter cache timeout** *seconds* 例： `switch(config)# snmp-server counter cache timeout 1200`	ポートの状態がローカルキャッシュに保持される時間を秒単位で定義します。カウンタ キャッシュはデフォルトで有効になっており、デフォルトのキャッシュ タイムアウト値は 10 秒です。無効にすると、デフォルトのキャッシュ タイムアウト値は 50 秒になります。範囲は 1 ～ 3600 です。 （注） End of Row（EoR）スイッチングの場合、範囲は 10 ～ 3600 です。
ステップ 3	（任意） **show running-config snmp all |i cac** 例： `switch(config)# copy running-config snmp all | i cac`	設定された SNMP サーバ カウンタ キャッシュ更新タイムアウト値を表示します。
ステップ 4	**no snmp-server counter cache enable** 例： `switch(config)# no snmp-server counter cache enable`	カウンタ キャッシュの更新を無効にします。 （注） カウンタ キャッシュの更新が無効になっている場合、timeout パラメータに設定された値によって、ポートの状態がカウンタ キャッシュに保持される時間が決まります。

AAA 同期時間の変更

同期したユーザ設定を Cisco NX-OS に維持させる時間の長さを変更できます。

手順の概要

1. **configure terminal**
2. **snmp-server aaa-user cache-timeout** *seconds*
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**snmp-server aaa-user cache-timeout** *seconds* 例： `switch(config)# snmp-server aaa-user cache-timeout 1200`	ローカル キャッシュで AAA 同期ユーザ設定を維持する時間を設定します。値の範囲は 1 ～ 86400 秒です。デフォルトは 3600 です。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP ローカル エンジン ID の設定

Cisco NX-OS リリース 7.0(3)I6(1) 以降では、ローカルデバイスにエンジン ID を設定できます。

Note SNMP ローカル エンジン ID を設定すると、すべての SNMP ユーザ、V3 ユーザに設定されたホスト、およびコミュニティ ストリングを再設定する必要があります。Cisco NX-OS リリース 7.0(3)I7(1) 以降では、SNMP ユーザとコミュニティ ストリングのみを再設定する必要があります。

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server engineID local** *engineid-string*
3. **show snmp engineID**
4. [**no**] **snmp-server engineID local** *engineid-string*
5. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**snmp-server engineID local** *engineid-string* **Example:**	ローカルデバイスの SNMP エンジン ID を変更します。

	Command or Action	Purpose
	`switch(config)# snmp-server engineID local AA:BB:CC:1A:2C:10`	ローカル エンジン ID は、コロンで指定された 16 進数オクテットのリストとして設定する必要があります。ここでは 10 ～ 64 の範囲の偶数 16 進数文字が使用され、2 つの 16 進数文字ごとにコロンで区切られます。たとえば、80:00:02:b8:04:61:62:63 です。
ステップ 3	**show snmp engineID** **Example:** `switch(config)# show snmp engineID`	設定されている SNMP エンジンの ID を表示します。
ステップ 4	[**no**] **snmp-server engineID local** *engineid-string* **Example:** `switch(config)# no snmp-server engineID local AA:BB:CC:1A:2C:10`	ローカル エンジン ID を無効にし、自動生成されたデフォルトのエンジン ID を設定します。
ステップ 5	Required: **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP の設定の確認

SNMP 設定情報を表示するには、次の作業を行います。

コマンド	目的
show interface snmp-ifindex	すべてのインターフェイスについて（IF-MIB から）SNMP の ifIndex 値を表示します。

コマンド	目的
show running-config snmp [**all**]	SNMP の実行コンフィギュレーションを表示します。 10.1(1) より前のリリースから 10.1(1) に導入された SNMP ユーザは、設定されたプライバシー プロトコル AES-128 または DES で表示されます。新しいユーザ（リリース 10.1(1) 以降）は、デフォルトで AES-128 プロトコルで設定されます。 9.3(8) リリース以降、show run の SNMPv3 ユーザは、ハッシュではなく SALT 形式で表示されます。
show snmp	SNMP ステータスを表示します。
show snmp community	SNMP コミュニティ ストリングを表示します。 **Note** **snmp-server mib community-map** コマンドの SNMP コンテキストの名前が 11 文字を超える場合、**show snmp community** コマンドの出力は表形式ではなく垂直形式で表示されます。
show snmp context	SNMP コンテキスト マッピングを表示します。
show snmp engineID	SNMP engineID を表示します。
show snmp group	SNMP ロールを表示します。
show snmp host	設定した SNMP ホストの情報を表示します。
show snmp session	SNMP セッションを表示します。

コマンド	目的
show snmp source-interface	設定した発信元インターフェイスの情報を表示します。
show snmp trap	イネーブルまたはディセーブルである SNMP 通知を表示します。
show snmp user	SNMPv3 ユーザを表示します。

SNMP エンティティ

次の SNMP エンティティ MIB OID を使用して、SNMP EPLD 情報を表示できます。

1. entPhysicalName：1.3.6.1.2.1.47.1.1.1.1.7
2. entPhysicalDescr：1.3.6.1.2.1.47.1.1.1.1.2
3. entPhysicalContainedIn : 1.3.6.1.2.1.47.1.1.1.1.4
4. entPhysicalFirmwareRev : 1.3.6.1.2.1.47.1.1.1.1.9

次に、出力例を示します。

```
bgl-ads-4144:167> snmpget -v 2c -c Cisco_59485 10.197.137.54
.1.3.6.1.2.1.47.1.1.1.1.2.120408 << entPhysicalDescr
SNMPv2-SMI::mib-2.47.1.1.1.1.2.120408 = STRING: "Module-1 IO FPGA"
bgl-ads-4144:168> snmpget -v 2c -c Cisco_59485 10.197.137.54
.1.3.6.1.2.1.47.1.1.1.1.4.120408 << entPhysicalContainedIn
SNMPv2-SMI::mib-2.47.1.1.1.1.4.120408 = INTEGER: 22
bgl-ads-4144:169> snmpget -v 2c -c Cisco_59485 10.197.137.54
.1.3.6.1.2.1.47.1.1.1.1.7.120408 << entPhysicalName
SNMPv2-SMI::mib-2.47.1.1.1.1.7.120408 = STRING: "Module-1 IO FPGA"
bgl-ads-4144:170> snmpget -v 2c -c Cisco_59485 10.197.137.54
.1.3.6.1.2.1.47.1.1.1.1.9.120408 << entPhysicalFirmwareRev
SNMPv2-SMI::mib-2.47.1.1.1.1.9.120408 = STRING: "0x12"
```

SNMP の設定例

次に、Blue VRF を使用して、ある通知ホスト レシーバに Cisco linkUp または Down 通知を送信するよう Cisco NX-OS を設定し、Admin と NMS という 2 つの SNMP ユーザを定義する例を示します。

```
configure terminal
snmp-server contact Admin@company.com
snmp-server user Admin auth sha abcd1234 priv abcdefgh
snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID
00:00:00:63:00:01:00:22:32:15:10:03
snmp-server host 192.0.2.1 informs version 3 auth NMS
snmp-server host 192.0.2.1 use-vrf Blue
```

```
snmp-server enable traps link cisco
```

次に、ホスト レベルで設定された帯域内ポートを使用してトラップを送信するよう、SNMP を設定する例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server host 171.71.48.164 version 2c public
switch(config)# snmp-server host 171.71.48.164 source-interface ethernet 1/2
switch(config)# show snmp host
-------------------------------------------------------------------
Host Port Version Level Type SecName
-------------------------------------------------------------------
171.71.48.164 162 v2c noauth trap public
Source interface: Ethernet 1/2
-------------------------------------------------------------------
switch(config)# snmp-server host 171.71.48.164 use-vrf default
switch(config)# show snmp host
-------------------------------------------------------------------
Host Port Version Level Type SecName
-------------------------------------------------------------------
171.71.48.164 162 v2c noauth trap public
Use VRF: default
Source interface: Ethernet 1/2
-------------------------------------------------------------------
```

次に、グローバルに設定した帯域内ポートを使用してトラップを送信するよう SNMP を設定する例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server source-interface traps ethernet 1/2
switch(config)# show snmp source-interface
-------------------------------------------------------------------
Notification source-interface
-------------------------------------------------------------------
trap Ethernet1/2
inform -
-------------------------------------------------------------------
switch(config)# snmp-server host 171.71.48.164 use_vrf default
switch(config)# show snmp host
-------------------------------------------------------------------
Host Port Version Level Type SecName
-------------------------------------------------------------------
171.71.48.164 162 v2c noauth trap public
Use VRF: default
Source interface: Ethernet 1/2
-------------------------------------------------------------------
```

VRF red を SNMPv2c のパブリック コミュニティ ストリングにマッピングする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# vrf context red
switch(config-vrf)# exit
switch(config)# snmp-server context public1 vrf red
switch(config)# snmp-server mib community-map public context public1
```

OSPF インスタンス Enterprise を同じ SNMPv2c パブリック コミュニティ ストリングにマッピングする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# feature ospf
switch(config)# router ospf Enterprise
switch(config-router)# exit
switch(config)# snmp-server context public1 instance Enterprise
switch(config)# snmp-server mib community-map public context public1
```

その他の参考資料

関連資料

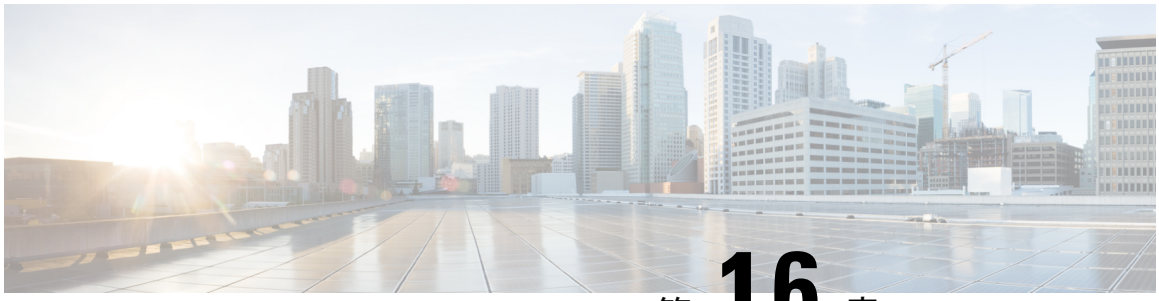
関連項目	マニュアル タイトル
IP ACL と AAA	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
MIB	『*Cisco Nexus 7000 Series and 9000 Series NX-OS MIB Quick Reference*』

RFC

RFC	タイトル
RFC 3414	シンプル ネットワーク管理プロトコル（*SNMPv3*）バージョン *3* 向けユーザベース セキュリティ モデル（*USM*）
RFC 3415	『*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*』

MIB

MIB	MIB のリンク
SNMP に関連する MIB	サポートされている MIB を検索およびダウンロ 次の URL にアクセスしてください。 https://cisco.github.io/cisco-mibs/supportlists/nexus9 Nexus9000MIBSupportList.html



第 16 章

RMON の設定

この章では、Cisco NX-OS デバイスでのリモート モニタリング（RMON）機能を設定する方法について説明します。

この章は、次の内容で構成されています。

- RMON について, on page 319
- RMON の注意事項と制約事項 （321 ページ）
- RMON のデフォルト設定 （321 ページ）
- RMON の設定 （322 ページ）
- RMON 設定の確認, on page 324
- RMON の設定例 （325 ページ）
- その他の参考資料 （325 ページ）

RMON について

RMON は、各種ネットワーク エージェントおよびコンソール システムがネットワーク モニタリング データを交換できるようにする、簡易ネットワーク管理プロトコル（SNMP）インターネット技術特別調査委員会（IETF）の標準モニタリング仕様です。Cisco NX-OS では、Cisco NX-OS デバイスをモニタするための、RMON アラーム、イベント、およびログをサポートします。

RMON アラームは、指定された期間、特定の管理情報ベース（MIB）オブジェクトをモニタリングし、指定されたしきい値でアラームを発生させ、別のしきい値でアラームをリセットします。アラームと RMON イベントを組み合わせて使用し、RMON アラームが発生したときにログ エントリまたは SNMP 通知を生成できます。

Cisco NX-OS では、RMON はデフォルトで有効ですが、アラームは設定されていません。RMON アラームを設定するには、CLI または SNMP 互換ネットワーク管理ステーションを使用します。

RMON アラーム

SNMP INTEGER タイプの解決を行う任意の MIB オブジェクトにアラームを設定できます。指定されたオブジェクトは、標準のドット付き表記（たとえば、1.3.6.1.2.1.2.2.1.14 は ifInOctets.14 を表します）の既存の SNMP MIB オブジェクトでなければなりません。

アラームを作成する場合、次のパラメータを指定します。

- モニタする MIB オブジェクト。
- サンプリング間隔：MIB オブジェクトのサンプル値を収集するのにデバイスが使用する間隔
- サンプル タイプ：絶対サンプルでは、MIB オブジェクト値の現在のスナップショットを使用します。デルタ サンプルは連続した 2 つのサンプルを使用し、これらの差を計算します。
- 上限しきい値：デバイスが上限アラームを発生させる、または下限アラームをリセットするときの値
- 下限しきい値：デバイスが下限アラームを発生させる、または上限アラームをリセットするときの値
- イベント：アラーム（上限または下限）の発生時にデバイスが実行するアクション

Note hcalarms オプションを使用して、アラームを 64 ビットの整数の MIB オブジェクトに設定します。

たとえば、エラー カウンタ MIB オブジェクトにデルタ タイプ上限アラームを設定できます。エラー カウンタ デルタがこの値を超えた場合、SNMP 通知を送信し、上限アラーム イベントを記録するイベントを発生させることができます。この上限アラームは、エラーカウンタのデルタ サンプルが下限しきい値を下回るまで再度発生しません。

Note 下限しきい値には、上限しきい値よりも小さな値を指定してください。

RMON イベント

特定のイベントを各 RMON アラームにアソシエートさせることができます。RMON は次のイベント タイプをサポートします。

- SNMP 通知：関連したアラームが発生したときに、SNMP risingAlarm または fallingAlarm 通知を送信します。
- ログ：関連したアラームが発生した場合、RMON ログテーブルにエントリを追加します。

- 両方：関連したアラームが発生した場合、SNMP 通知を送信し、RMON ログ テーブルにエントリを追加します。

下限アラームおよび上限アラームに異なるイベントを指定できます。

Note デフォルトの RMON イベント テンプレート設定の使用を選択することも、これらのエントリを削除して新しい RMON イベントを作成することもできます。RMON アラーム設定を作成するまで、これらの設定によってトリガーされるアラームはありません。

RMON のハイ アベイラビリティ

Cisco NX-OS は、RMON のステートレス リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバーの後、Cisco NX-OS は実行コンフィギュレーションを適用します。

RMON の仮想化サポート

Cisco NX-OS は、RMON のインスタンスを 1 つサポートします。

RMON は Virtual Routing and Forwarding（VRF）を認識します。特定の VRF を使用して RMON SMTP サーバに接続するように RMON を設定できます。

RMON の注意事項と制約事項

RMON には、次の注意事項および制限事項があります。

- SNMP通知イベントタイプを使用するには、SNMPユーザおよび通知レシーバを設定する必要があります。
- 整数になる MIB オブジェクトに、RMON アラームのみを設定できます。
- RMON アラームを設定する場合は、オブジェクト ID がインデックスで 1 オブジェクトだけを示すようになっている必要があります。たとえば、1.3.6.1.2.1.2.2.1.14 は cpmCPUTotal5minRev に対応し、.1 は cpmCPUTotalIndex インデックスに対応し、オブジェクト ID の 1.3.6.1 .2.1.2.2.1.14.1 を作成します。

RMON のデフォルト設定

次の表に、RMON パラメータのデフォルト設定を示します。

パラメータ	デフォルト
RMON	有効
アラーム	未設定
イベント	設定済み（ただし、トリガーされたイベントは何も引き起こしません）

RMON の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

RMON アラームの設定

任意の整数の SNMP MIB オブジェクトに RMON アラームを設定できます。

次のパラメータを任意で指定することもできます。

- 上限および下限しきい値が指定値を超えた場合に発生させるイベント番号
- アラームのオーナー

SNMP ユーザが設定され、SNMP 通知がイネーブルであることを確認します。

Before you begin

SNMP ユーザーが設定され、SNMP 通知が有効であることを確認します。

SUMMARY STEPS

1. **configure terminal**
2. **rmon alarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-index*] **falling-threshold** *value* [*event-index*] [**owner** *name*]
3. **rmon hcalarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold-high** *value* **rising-threshold-low** *value* [*event-index*] **falling-threshold-high** *value* **falling-threshold-low** *value* [*event-index*] [**owner** *name*] [**storagetype** *type*]
4. (Optional) **show rmon** {**alarms** | **hcalarms**}
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**rmon alarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-index*] **falling-threshold** *value* [*event-index*] [**owner** *name*]
	Example:
	`switch(config)# rmon alarm 20 1.3.6.1.2.1.2.2.1.14.1 2900`
	`delta rising-threshold 1500 1 falling-threshold 0 owner test`
		RMON アラームを作成します。値の範囲は -2147483647 ～ 2147483647 です。オーナー名は任意の英数字ストリングです。

ステップ 3	**rmon hcalarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold-high** *value* **rising-threshold-low** *value* [*event-index*] **falling-threshold-high** *value* **falling-threshold-low** *value* [*event-index*] [**owner** *name*] [**storagetype** *type*]
	Example:
	`switch(config)# rmon alarm 20 1.3.6.1.2.1.2.2.1.14.16777216 2900`
	`delta rising-threshold-high 15 rising-threshold-low 151`
	`falling-threshold-high 0 falling-threshold-low 0 owner test`
		RMON 高容量アラームを作成します。値の範囲は -2147483647 ～ 2147483647 です。オーナー名は任意の英数字ストリングです。
		ストレージ タイプの範囲は 1 ～ 5 です。

ステップ 4	(Optional) **show rmon** {**alarms** | **hcalarms**} **Example:** `switch(config)# show rmon alarms`	RMONアラームまたは高容量アラームに関する情報を表示します。
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

RMON イベントの設定

RMON アラームとアソシエートするよう RMON イベントを設定できます。複数の RMON アラームで同じイベントを再利用できます。

Before you begin

SNMP ユーザが設定され、SNMP 通知が有効であることを確認します。

SUMMARY STEPS

1. **configure terminal**
2. **rmon event** *index* [**description** *string*] [**log**] [**trap** *string*] [**owner** *name*]
3. (Optional) **show rmon events**
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**rmon event** *index* [**description** *string*] [**log**] [**trap** *string*] [**owner** *name*] **Example:** `switch(config)# rmon event 1 trap trap1`	RMON イベントを設定します。説明の文字列、トラップの文字列、およびオーナー名は、任意の英数字文字列です。
ステップ 3	(Optional) **show rmon events** **Example:** `switch(config)# show rmon events`	RMON イベントに関する情報を表示します。
ステップ 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

RMON 設定の確認

RMON 設定情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show rmon alarms	RMON アラームに関する情報を表示します。
show rmon events	RMON イベントに関する情報を表示します。
show rmon hcalarms	RMON 高容量アラームに関する情報を表示します。

コマンド	目的
show rmon logs	RMON ログに関する情報を表示します。

RMON の設定例

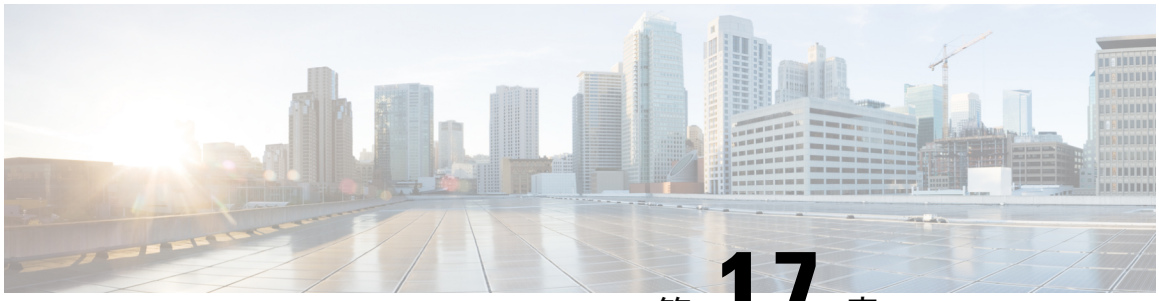
ifInOctets.14 にデルタ上限アラームを作成し、このアラームに通知イベントを関連付ける方法の例を示します。

```
configure terminal
rmon alarm 20 1.3.6.1.2.1.2.2.1.14.1 2900 delta rising-threshold 1500 1 falling-threshold
 0 owner test
rmon event 1 trap trap1
```

その他の参考資料

MIB

MIB	MIB のリンク
RMON に関連する MIB	サポートされている MIB を検索およびダウンロ 次の URL にアクセスしてください。 https://cisco.github.io/cisco-mibs/supportlists/nexus9 Nexus9000MIBSupportList.html



第 17 章

オンライン診断の設定

この章では、デバイス上で汎用オンライン診断（GOLD）機能を設定する方法について説明します。

- オンライン診断について, on page 327
- オンライン診断の注意事項と制約事項 （339 ページ）
- オンライン診断のデフォルト設定 （340 ページ）
- オンライン診断の設定 （341 ページ）
- オンライン診断設定の確認, on page 345
- オンライン診断のコンフィギュレーション例 （346 ページ）

オンライン診断について

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。

オンライン診断機能には、さまざまなハードウェア コンポーネントを検査し、データ パスと制御信号を確認するテストが組み込まれています。中断を伴うオンライン診断テスト（破壊モードのループバック テストなど）、および中断を伴わないオンライン診断テスト（ASIC レジスタ検査など）は、起動時、ライン モジュールの活性挿抜（OIR）時、およびシステム リセット時に実行されます。中断を伴わないオンライン診断テストは、バックグラウンドヘルスモニタリングの一部として実行され、これらのテストはオンデマンドで実行できます。

オンライン診断は、起動、ランタイムまたはヘルスモニタリング診断、およびオンデマンド診断に分類されます。起動診断は起動時に、ヘルスモニタリングテストはバックグラウンドで、オンデマンド診断はアクティブ ネットワークにデバイスが接続されたときに1回だけ、またはユーザが指定した間隔で実行されます。

ブートアップ診断

起動診断は起動中に実行され、Cisco NX-OS がモジュールをオンラインにする前に、障害ハードウェアが検出されます。たとえば、デバイスに障害モジュールを搭載した場合、起動診断で

モジュールがテストされ、デバイスがそのモジュールをトラフィックの転送に使用しないうちに、モジュールがオフラインにされます。

起動診断では、スーパーバイザとモジュール ハードウェア間、およびすべての ASIC のデータパスと制御パス間の接続も検査されます。次の表では、モジュールおよびスーパーバイザの起動診断テストについて説明します。

Table 16: ブートアップ診断

診断	説明
OBFL	オンボード障害ロギング フラッシュ（Cisco NX-OS）の整合性を確認します。
MacSecPortLoopback（Cisco Nexus 9736C-FX および 9736Q-FX ライン カードのみ）	スーパーバイザから ASIC の各物理前面パネルポートへのパケット パス、各ポートの MACSEC 機能、および Cisco Nexus 9736C-FX および 9736Q-FX ライン カードの暗号化機能と復号化機能をテストします。 **diagnostic bootup level** が **complete** に設定されている場合、ブート時に MacSecPortLoopback テストが実行されます。 MacSecPortLoopback テストは、Cisco Nexus 9736C-FX および 9736Q-FX ライン カードの 36 個の前面ポートのすべてのポートで実行されます。MAC sec ハードウェアは、使用可能な 4 つの暗号スイート アルゴリズム（GCM-AES-128、GCM-AES-256、GCM-AES-XPN-128、および GCM-AES-XPN-256）でテストされます。 **Note** MacSecPortLoopback テストが失敗すると、テストは SYSLOG または OBFL の形式でレポートします。テスト障害が発生すると、ポートがダウンし、 **show interface** CLI 出力に `MACsec 障害`が表示されます。MACsecテストをスキップするには、**diagnostic bootup level** を **minimal** または **bypass** に設定します。
USB	中断を伴わないテスト。モジュールにおける USB コントローラの初期化を検査
ManagementPortLoopback	中断を伴うテスト、非オンデマンド型テスト。モジュールの管理ポートでループバックをテスト
EOBCPortLoopback	中断を伴うテスト、非オンデマンド型テスト。イーサネット帯域外。

起動診断テストはエラーを Onboard Failure Logging（OBFL）および syslog に記録し、診断の LED 表示（オン、オフ、合格、失敗）を開始します。

起動診断テストをバイパスするようにデバイスを設定することも、またはすべての起動診断テストを実行するように設定することもできます。

ランタイムまたはヘルス モニタリング診断

ランタイム診断はヘルス モニタリング（HM）診断ともいいます。これらの診断テストによって、アクティブデバイスの状態に関する情報が得られます。ランタイム ハードウェア エラー、メモリ エラー、ハードウェア モジュールの経時的劣化、ソフトウェア障害、およびリソース不足が検出されます。

アクティブ ネットワーク トラフィックを処理するデバイスの状態を確認するヘルス モニタリング診断テストは、中断を伴わず、バックグラウンドで実行されます。ヘルス モニタリング テストはイネーブルまたはディセーブルにできます。また、ランタイム インターバルの変更が可能です。

次の表に、モジュールおよびスーパーバイザのヘルス モニタリング診断とテスト ID を示します。

（注） モジュールの機能に応じて、テストが存在する場合と存在しない場合があります。モジュールで使用可能なテストのリストは、CLI コマンド、 **show diagnostic content module** <*module*> を使用して確認できます。

表 *17*：ヘルス モニタリングの無停止での診断

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
モジュール				
ACT2	30 分	アクティブ	モジュール上のセキュリティ デバイスの整合性を確認します。	GOLD "ACT2" テストに 20 回連続で失敗した場合は、CallHome を実行し、エラーを記録し、その後 HM テストをディセーブルにします。

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
ASICRegisterCheck	モジュラスイッチ：1分 非モジュラスイッチ：20 秒、最小設定のデフォルトシミュレーション間隔は 10 秒	アクティブ	モジュール上の ASIC への読み取り/書き込みアクセスを検証します。	CallHome を実行し、エラーを記録し、GOLD "ASICRegisterCheck" テストに 20 回連続で失敗した場合は、その後その ASIC デバイスおよびインスタンスの HM テストをディセーブルにします。
PrimaryBootROM	24 時間 [1]	アクティブ	モジュール上のプライマリ ブート デバイスの完全性を確認します。	CallHome を実行し、エラーを記録し、GOLD "PrimaryBootROM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
SecondaryBootROM	24 時間 [1]	アクティブ	モジュール上のセカンダリ ブート デバイスの完全性を確認します。	CallHome を実行し、エラーを記録し、GOLD "SecondaryBootROM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
BootupPortLoopback	起動時のみ	起動時のみ：アクティブ	スーパーバイザから前面パネルのポート（および背面）パスが動作しているかどうかを確認します。すべてのフロントポートについて、テストはアクティブ スーパーバイザでパケットを生成し、ターゲットポートにパケットを送信し、フロント ポート内の内部ループバックを使用して、パケットをアクティブ スーパーバイザにリダイレクトします。	GOLD "BootupPortLoopback" テストに1回連続で失敗した場合は、CallHome を実行し、影響があるポートのエラーを無効にして、影響を受けたポートでのエラーテストを記録します。
PortLoopback	30 分	アクティブ	すべての管理ダウンポートでポート単位で診断をチェックします。	CallHome を実行し、Syslog、OBFL、または例外ログにエラーを記録し、GOLD "PortLoopback" テストに 10 回連続で失敗した場合は、その後影響を受けたポートでの HM テストをディセーブルにします。
RewriteEngineLoopback	1分	アクティブ	1 エンジン ASIC デバイスまでのすべてのポートの無停止ループバックの整合性を確認します。	CallHome を実行し、Syslog、OBFL、または例外ログにエラーを記録し、GOLD "RewriteEngine" テストに 10 回連続で失敗した場合は、その後影響を受けたポートでの HM テストをディセーブルにします。

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
AsicMemory	起動時のみ	起動時のみ：非アクティブ	ASIC の Mbist ビットを使用して AsicMemory の整合性をチェックします。	GOLD "AsicMemory" テストに失敗した場合には、CallHome を実行し、エラーを記録します。テストの失敗の原因となる問題は一時的なものである可能性があるため、カーネルパニックによるリカバリ リロードを試行します。 （注） テストが失敗したときにカーネルパニックを回避するには、EEM システムポリシーを上書します。
FpgaRegTest	30 秒	ヘルス モニタリング テスト：30 秒ごと：アクティブ	FPGA への読み取り/書き込みによって FPGA のステータスをテストします。	GOLD "FpgaRegTest" テストに 20 回連続で失敗した場合は、CallHome を実行し、エラーを記録し、その後 HM テストをディセーブルにします。テストの失敗の原因となる問題は一時的なものである可能性があるため、カーネルパニックによるリカバリ リロードを試行します。 （注） テストが失敗したときにカーネルパニックを回避するには、EEM システムポリシーを上書します。

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
L2ACLRedirect	1分	ヘルスモニタリング テスト：30分：アクティブ	アクティブ ノードが動作しているかどうかを確認します。テストでは、アクティブ ファブリックモジュールを介してアクティブスーパーバイザでパケットを生成します。次に、パケットを前面パネル ポート（ラインカード上の物理インターフェイス）に送信し、ACL エントリを使用して、パケットをアクティブ スーパーバイザにリダイレクトします。	L2ACLRedirectテストを10回連続で失敗した場合は、CallHome を実行し、エラーを記録し、その後HMテストをディセーブルにします。テストの失敗の原因となる問題は一時的なものである可能性があるため、カーネルパニックによるリカバリ リロードを試行します。 （注） テストが失敗したときにカーネルパニックを回避するには、EEMシステムポリシーを上書します。
OBFL	30 分	アクティブ	オンボード障害ロギング（OBFL）フラッシュの整合性を確認し、デバイスの利用可能なストレージをモニタリングします。	

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
FabricConnectivityTest	1分	アクティブ	ファブリック/ラインカードのリンク ステータスを確認します。 ファブリック リンクが機能していることを検証します。 （注） Cisco Nexus 9500-R シリーズ ライン カードおよび Cisco N9K-X9836DM-A ライン カードでのみ使用できます。これは、Cisco Nexus 9808 および 9804 スイッチを搭載した X98366DM-A および X98900CD-A ラインカードでもサポートされています。	

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
FabricReachabilityTest	1分	アクティブ	ファブリック/ライン カードの到達可能性ステータスを確認します。 各ファブリック コンポーネントに、システム内の他のすべてのファブリック コンポーネントへの有効なパスがあることを検証します。 （注） Cisco Nexus 9500-R シリーズ ライン カードおよび Cisco N9K-X9836DM-A ライン カードでのみ使用できます。これは、Cisco Nexus 9808 および 9804 スイッチを搭載した X98366DM-A および X98900CD-A ラインカードでもサポートされています。	
スーパーバイザ（Supervisor）				
バックプレーン	30 分	アクティブ	バックプレーン SPROM デバイスの整合性を確認します。	

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
NVRAM	5 分	アクティブ	スーパーバイザのNVRAMブロックの健全性を確認します。	CallHome を実行し、エラーを記録し、GOLD "NVRAM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
RealTimeClock	5 分	アクティブ	スーパーバイザ上のリアルタイムクロックが時を刻んでいるかどうかを確認します。	CallHome を実行し、エラーを記録し、GOLD "RealTimeClock" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
PrimaryBootROM	30 分	アクティブ	スーパーバイザ上のプライマリブート デバイスの完全性を確認します。	CallHome を実行し、エラーを記録し、GOLD "PrimaryBootROM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
SecondaryBootROM	30 分	アクティブ	スーパーバイザ上のセカンダリブート デバイスの完全性を確認します。	CallHome を実行し、エラーを記録し、GOLD "SecondaryBootROM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
ブートフラッシュ	30 分	アクティブ	ブートフラッシュデバイスへのアクセスを確認します。	GOLD "CryptoDevice" テストに失敗したら、CallHome を実行し、エラーを記録します。
USB	30 分	アクティブ	USBデバイスへのアクセスを確認します。	Call Home を実行し、GOLD "USB" テストに失敗するとエラーを記録します。

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
SystemMgmtBus	30 秒	アクティブ	システム管理バスの使用可能性を確認します。	Call Home を実行し、エラーを記録し、GOLD "SystemMgmtBus" テストに 20 回連続で失敗した場合は、そのファンまたは電源の HM テストを無効にします。
MCE	30 分	ヘルスモニタリング テスト：30 分：アクティブ	このテストは mcd_dameon を使用し、カーネルによって報告されたマシン チェック エラーを報告します。	GOLD "ACT2" テストに 20 回連続で失敗した場合は、CallHome を実行し、エラーを記録し、その後 HM テストをディセーブルにします。
Pcie	起動時のみ	起動時のみ：非アクティブ	PCIe ステータス レジスタを読み取り、PCIe デバイスのエラーをチェックします。	GOLD "Pcie" テストに失敗したら、CallHome を実行し、エラーを記録します。
コンソール	起動時のみ	起動時のみ：非アクティブ	これにより、起動時に管理ポートでポート ループバック テストが実行され、整合性が確認されます。	GOLD "Cosole" テストに 20 回連続で失敗した場合は、CallHome を実行し、エラーを記録し、その後 HM テストをディセーブルにします。

診断	デフォルトのインターバル	デフォルト設定	説明	改善処置
FpgaRegTest	30 秒	ヘルス モニタリング テスト：30秒ごと：アクティブ	FPGA への読み取り/書き込みによってFPGA のステータスをテストします。 （注） ファブリック モジュール（19 ～ 26）用の Cisco Nexus 9808 および 9804 スイッチの FpgaRegTest は、Active-SUP の FpgaRegTest 結果の下に表示されます。	GOLD "FpgaRegTest" テストに 20 回連続で失敗した場合は、CallHome を実行し、エラーを記録し、その後 HM テストをディセーブルにします。テストの失敗の原因となる問題は一時的なものである可能性があるため、カーネルパニックによるリカバリ リロードを試行します。 （注） テストが失敗したときにカーネルパニックを回避するには、EEM システム ポリシーを上書きします。

[1] 設定可能な最小テスト間隔は 6 時間です。

オンデマンド診断

オンデマンドテストは、障害の場所を特定するのに役立ちます。通常は、次のような状況で必要です。

- 障害の分離など、発生したイベントに対処する場合。
- リソース使用限度の超過などのイベントの発生が予測される場合。

すべてのヘルス モニタリング テストをオンデマンドで実行できます。即時実行するオンデマンド診断テストをスケジューリングできます。

ヘルス モニタリング テストのデフォルト インターバルも変更可能です。

高可用性

ハイアベイラビリティの重要な機能は、アクティブなネットワークでデバイスが稼働している状態のままハードウェア障害を検出して、対処することです。ハイアベイラビリティのオンライン診断では、ハードウェア障害を検出して、スイッチオーバーを判断するためにハイアベイラビリティ ソフトウェアにフィードバックします。

Cisco NX-OS は、オンライン診断のステートレス リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバーの後、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化のサポート

オンライン診断機能は Virtual Routing and Forwarding（VRF）を認識します。特定の VRF を使用してオンライン診断 SMTP サーバに接続するようにオンライン診断機能を設定できます。

オンライン診断の注意事項と制約事項

オンライン診断には、次の注意事項と制限事項があります。

- 次の Cisco Nexus プラットフォーム スイッチおよびライン カードは、ランタイム PortLoopback テストをサポートしていませんが、BootupPortLoopback テストをサポートしています。

  スイッチ

  - Cisco Nexus 92160YC-X
  - Cisco Nexus 92304QC
  - Cisco Nexus 9264PQ
  - Cisco Nexus 9272Q
  - Cisco Nexus 9232C
  - Cisco Nexus 9236C
  - Cisco Nexus 9256PV
  - Cisco Nexus 92300YC
  - Cisco Nexus 93108TC-EX
  - Cisco Nexus 93108TC-EX-24
  - Cisco Nexus 93180LC-EX
  - Cisco Nexus 93180YC-EX
  - Cisco Nexus 93180YC-EXU
  - Cisco Nexus 93180YC-EX-24
  - Cisco Nexus 9232E-B1
  - Cisco Nexus 93180YC-FX3S
  - Cisco Nexus 9348GC-FX3
  - Cisco Nexus 9348GC-FX3PH
  - Cisco Nexus 9332D-H2R

ラインカード

- Cisco Nexus 9736C-EX
- Cisco Nexus 97160YC-EX
- Cisco Nexus 9732C-EX
- Cisco Nexus 9732C-EXM

- 中断を伴うオンライン診断テストをオンデマンド方式で実行することはできません。
- インターフェイス Rx および Tx パケット カウンタは、シャットダウン状態のポートで増えます（およそ 15 分ごとに 4 パケット）。
- PortLoopback テストは定期的に行われるため、パケット カウンタは管理ダウン ポートで 30 分ごとに追加されます。テストは管理ダウン ポートでのみ実行されます。ポートが閉じられている場合は、カウンタは影響を受けません。
- ポートごとの BootupPortLoopback テストでポートが失敗すると、ポートは errdisable ステートになります。（この状態を削除するには、ポートで **shutdown** および **no shutdown** およびコマンドを入力します）。
- Cisco NX-OS リリース 10.3(1)F 以降、Generic Online Diagnostics（GOLD；汎用オンライン診断）は Cisco Nexus 9800 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、Generic Online Diagnostics（GOLD、汎用オンライン診断）は次のライン カードおよびスイッチでサポートされます。
  - Cisco Nexus 9804 スイッチ
  - Cisco Nexus 9332D-H2R スイッチ
  - Cisco Nexus X98900CD-A ライン カード
  - Cisco Nexus X98900CD-A および X9836DM-A ラインカードと Cisco Nexus 9808 および 9804 スイッチ

オンライン診断のデフォルト設定

次の表に、オンライン診断パラメータのデフォルト設定を示します。

パラメータ	デフォルト
起動時診断レベル	complete
中断を伴わないテスト	アクティブ

オンライン診断の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

起動診断レベルの設定

一連のすべてのテストを実行するように起動時診断を設定することも、またはモジュールが短時間で起動するように、すべての起動時診断テストをバイパスするように設定することもできます。

（注） 起動時オンライン診断レベルを complete に設定することを推奨します。起動時オンライン診断をバイパスすることは推奨しません。

手順の概要

1. **configure terminal**
2. **diagnostic bootup level** {**complete** | | **bypass**}
3. （任意） **show diagnostic bootup level**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**diagnostic bootup level** {**complete**		**bypass**} 例： `switch(config)# diagnostic bootup level complete`	デバイスの起動に続いて診断テストが開始されるように、起動診断レベルを設定します。 • **complete**：すべての起動診断テストを実行します。complete がデフォルトです。 • **bypass**：起動診断テストをまったく実行しません。

	コマンドまたはアクション	目的
ステップ 3	（任意） **show diagnostic bootup level** 例： `switch(config)# show diagnostic bootup level`	デバイスに現在設定されている起動診断レベル（bypass または complete）を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

診断テストのアクティブ化

診断テストをアクティブに設定し、任意でテストの実行間隔（時間、分、秒単位）を変更できます。

手順の概要

1. **configure terminal**
2. **diagnostic monitor interval module** *slot* **test** [*test-id* | *name* | **all**] **hour** *hour* **min** *minute* **second** *second*
3. [**no**] **diagnostic monitor module** *slot* **test** [*test-id* | *name* | **all**]
4. （任意） **show diagnostic content module** {*slot* | **all**}

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**diagnostic monitor interval module** *slot* **test** [*test-id* | *name* | **all**] **hour** *hour* **min** *minute* **second** *second* 例： `switch(config)# diagnostic monitor interval module 6 test 3 hour 1 min 0 second 0`	指定されたテストを実行する間隔を設定します。インターバルを設定しなかった場合は、過去に設定されたインターバルまたはデフォルトのインターバルでテストが実行されます。 引数の範囲は次のとおりです。 • *slot*：範囲は 1 ～ 10 です。 • *test-id*：範囲は 1 ～ 14 です。 • *name*：32 文字以内の英数字のストリング（大文字と小文字を区別）で指定します。

	コマンドまたはアクション	目的
		• *hour*：範囲は 0 ～ 23 時間です。 • *minute*：範囲は 0 ～ 59 分です。 • *second*：範囲は 0 ～ 59 秒
ステップ 3	[**no**] **diagnostic monitor module** *slot* **test** [*test-id* | *name* | **all**] 例： `switch(config)# diagnostic monitor interval module 6 test 3`	指定されたテストをアクティブにします。 引数の範囲は次のとおりです。 • *slot*：範囲は 1 ～ 10 です。 • *test-id*：範囲は 1 ～ 14 です。 • *name*：32 文字以内の英数字のストリング（大文字と小文字を区別）で指定します。 このコマンドの[**no**]形式は、指定されたテストを非アクティブにします。非アクティブにしたテストでは、現在の設定が維持されますが、スケジュール上の間隔ではテストは実行されません。
ステップ 4	（任意） **show diagnostic content module** {*slot* | **all**} 例： `switch(config)# show diagnostic content module 6`	診断テストおよび対応する属性の情報を表示します。

オンデマンド診断テストの開始または中止

オンデマンド診断テストを開始または中止できます。任意で、このテストを繰り返す回数の変更や、テストが失敗した場合のアクションの変更を行えます。

スケジューリングされたネットワーク メンテナンス期間内に、破壊モードの診断テストを開始する場合は、手動での開始に限定することを推奨します。

手順の概要

1. （任意） **diagnostic ondemand iteration** *number*
2. （任意） **diagnostic ondemand action-on-failure** {**continue failure-count** *num-fails* | **stop**}
3. **diagnostic start module** *slot* **test** [*test-id* | *name* | **all** | **non-disruptive**] [**port** *port-number* | **all**]
4. **diagnostic stop module** *slot* **test** [*test-id* | *name* | **all**]
5. （任意） **show diagnostic status module** *slot*

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	（任意） **diagnostic ondemand iteration** *number* 例： `switch# diagnostic ondemand iteration 5`	オンデマンドテストの実行回数を設定します。範囲は 1 ～ 999 です。デフォルトは 1 です。
ステップ 2	（任意） **diagnostic ondemand action-on-failure** {**continue failure-count** *num-fails* | **stop**} 例： `switch# diagnostic ondemand action-on-failure stop`	オンデマンドテストが失敗した場合のアクションを設定します。*num-fails* の範囲は 1 ～ 999 です。デフォルトは 1 です。
ステップ 3	必須: **diagnostic start module** *slot* **test** [*test-id* | *name* | **all** | **non-disruptive**] [**port** *port-number* | **all**] 例： `switch# diagnostic start module 6 test all`	モジュール上で 1 つまたは複数の診断テストを開始します。モジュール スロットの範囲は 1 ～ 10 です。*test-id* の範囲は 1 ～ 14 です。テスト名は大文字と小文字を区別し、最大 32 の英数字を使用できます。ポート範囲は 1 ～ 48 です。
ステップ 4	必須: **diagnostic stop module** *slot* **test** [*test-id* | *name* | **all**] 例： `switch# diagnostic stop module 6 test all`	モジュール上で 1 つまたは複数の診断テストを中止します。モジュール スロットの範囲は 1 ～ 10 です。*test-id* の範囲は 1 ～ 14 です。テスト名は大文字と小文字を区別し、最大 32 の英数字を使用できます。
ステップ 5	（任意） **show diagnostic status module** *slot* 例： `switch# show diagnostic status module 6`	診断テストがスケジューリングされていることを確認します。

診断結果のシミュレーション

診断テスト結果のシミュレーションが可能です。

手順の概要

1. **diagnostic test simulation module** *slot* **test** *test-id* {**fail** | **random-fail** | **success**} [**port** *number* | **all**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**diagnostic test simulation module** *slot* **test** *test-id* {**fail** | **random-fail** | **success**} [**port** *number* | **all**] 例： `switch# diagnostic test simulation module 2 test 2 fail`	テスト結果のシミュレーションを行います。*test-id* の範囲は 1 ～ 14 です。ポート範囲は 1 ～ 48 です。

診断結果の消去

診断テスト結果を消去できます。

手順の概要

1. **diagnostic clear result module** [*slot* | **all**] **test** {*test-id* | **all**}
2. **diagnostic test simulation module** *slot* **test** *test-id* clear

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**diagnostic clear result module** [*slot* | **all**] **test** {*test-id* | **all**} 例： `switch# diagnostic clear result module 2 test all`	指定されたテストのテスト結果を消去します。 引数の範囲は次のとおりです。 • *slot*：範囲は 1 ～ 10 です。 • *test-id*：範囲は 1 ～ 14 です。
ステップ 2	**diagnostic test simulation module** *slot* **test** *test-id* clear 例： `switch# diagnostic test simulation module 2 test 2 clear`	シミュレーションしたテスト結果を消去します。*test-id* の範囲は 1 ～ 14 です。

オンライン診断設定の確認

オンライン診断設定情報を表示するには、次の作業を行います。

コマンド	目的
show diagnostic bootup level	起動診断に関する情報を表示します。

コマンド	目的
show diagnostic content module {*slot* | **all**}	モジュールの診断テスト内容に関する情報を表示します。
show diagnostic description module *slot* **test** [*test-name* | **all**]	診断テストの説明を表示します。
show diagnostic events [**error** | **info**]	診断イベントをエラーおよび情報イベント タイプ別に表示します。
show diagnostic ondemand setting	オンデマンド診断に関する情報を表示します。
show diagnostic result module *slot* [**test** [*test-name* | **all**]] [**detail**]	診断結果に関する情報を表示します。
show diagnostic simulation module *slot*	シミュレーションした診断テストに関する情報を表示します。
show diagnostic status module *slot*	モジュールのすべてのテストについて、テスト状況を表示します。
show hardware capacity[**eobc** | **forwarding** | **interface** | **module** | **power**]	ハードウェアの機能、およびシステムによる現在のハードウェア使用率の情報を表示します。
show module	オンライン診断テストの状況を含むモジュール情報を表示します。

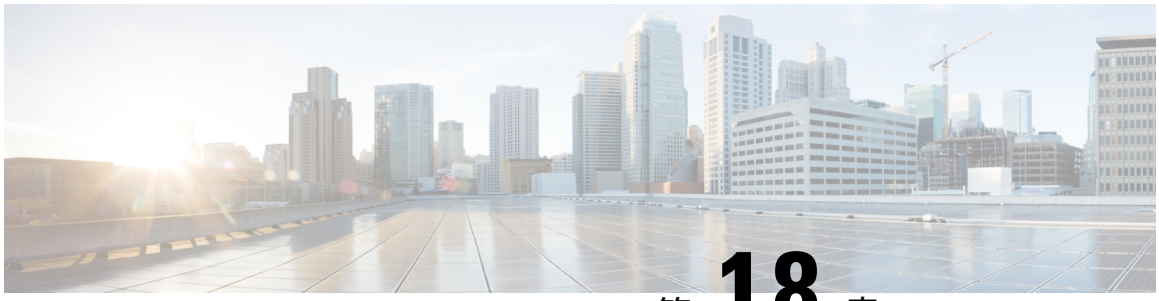
オンライン診断のコンフィギュレーション例

この例は、モジュール 6 ですべてのオンデマンド テストを開始する方法を示しています。

```
diagnostic start module 6 test all
```

この例は、モジュール 6 でテストテスト 2 をアクティブにして、テスト インターバルを設定する方法を示しています。

```
configure terminal
diagnostic monitor module 6 test 2
diagnostic monitor interval module 6 test 2 hour 3 min 30 sec 0
```



第 18 章

Embedded Event Manager の設定

この章では、Embedded Event Manager（EEM）を設定して Cisco NX-OS デバイス上のクリティカル イベントを検出し、対処する方法について説明します。

- EEM について （347 ページ）
- EEM の前提条件 （352 ページ）
- EEM の注意事項と制約事項 （352 ページ）
- EEM のデフォルト設定 （354 ページ）
- EEM の設定 （354 ページ）
- EEM の設定確認 （370 ページ）
- EEM の設定例 （371 ページ）
- イベント ログの自動収集とバックアップ （372 ページ）

EEM について

EEM はデバイス上で発生するイベントをモニタし、設定に基づいて各イベントの回復またはトラブルシューティングのためのアクションを実行します。

EEM は次の 3 種類の主要コンポーネントからなります。

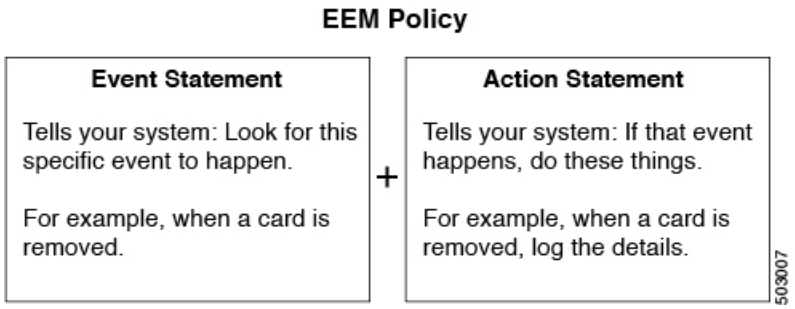
- イベント文：別の Cisco NX-OS コンポーネントからモニタし、アクション、回避策、または通知が必要になる可能性のあるイベント。
- アクション文：CLI コマンドの実行、Smart Call Home 機能を使用した電子メールの送信、インターフェイスの無効化など、イベントから回復するために EEM が実行できるアクション。
- ポリシー：イベントのトラブルシューティングまたはイベントからの回復を目的とした 1 つまたは複数のアクションとペアになったイベント。

ポリシー

EEM ポリシーは、イベント文および 1 つまたは複数のアクション文からなります。イベント文では、探すイベントとともに、イベントのフィルタリング特性を定義します。アクション文では、イベントの発生時に EEM が実行するアクションを定義します。

この図は、EEM ポリシーの基本的な 2 種類の文を示します。

図 *5: EEM* ポリシー文



コマンドライン インターフェイス（CLI）または VSH スクリプトを使用して EEM ポリシーを設定できます。

EEM からデバイス全体のポリシー管理ビューが得られます。スーパーバイザ上で EEM ポリシーを設定すると、EEM がイベント タイプに基づいて、正しいモジュールにポリシーをプッシュします。EEM はモジュール上でローカルに、またはスーパーバイザ上で（デフォルトのオプション）、発生したイベントに対応するアクションを実行します。

EEM はスーパーバイザ上でイベント ログを維持します。

Cisco NX-OS には、設定済みのさまざまなシステム ポリシーがあります。これらのシステム ポリシーでは、デバイスに関連する多数の一般的なイベントおよびアクションが定義されています。システム ポリシー名は、2 個の下線記号（__）から始まります。

使用するネットワークに合わせてユーザ ポリシーを作成できます。ユーザ ポリシーを作成すると、そのポリシーと同じイベントに関連するシステム ポリシー アクションが EEM によって発生したあと、ユーザ ポリシーで指定したアクションが行われます。

一部のシステム ポリシーは上書きすることもできます。設定した上書き変更がシステム ポリシーの代わりになります。イベントまたはアクションの上書きが可能です。

設定済みのシステム ポリシーを表示して、上書き可能なポリシーを判断するには、**show event manager system-policy** コマンドを使用します。

（注） **show running-config eem** コマンドを使用して、各ポリシーのコンフィギュレーションを確認してください。イベント文が指定されていて、アクション文が指定されていない上書きポリシーを設定した場合、アクションは開始されません。また、障害も通知されません。

（注） 上書きポリシーには、必ずイベント文を指定します。上書きポリシーにイベント文が含まれていないと、システム ポリシーで可能性のあるイベントがすべて上書きされます。

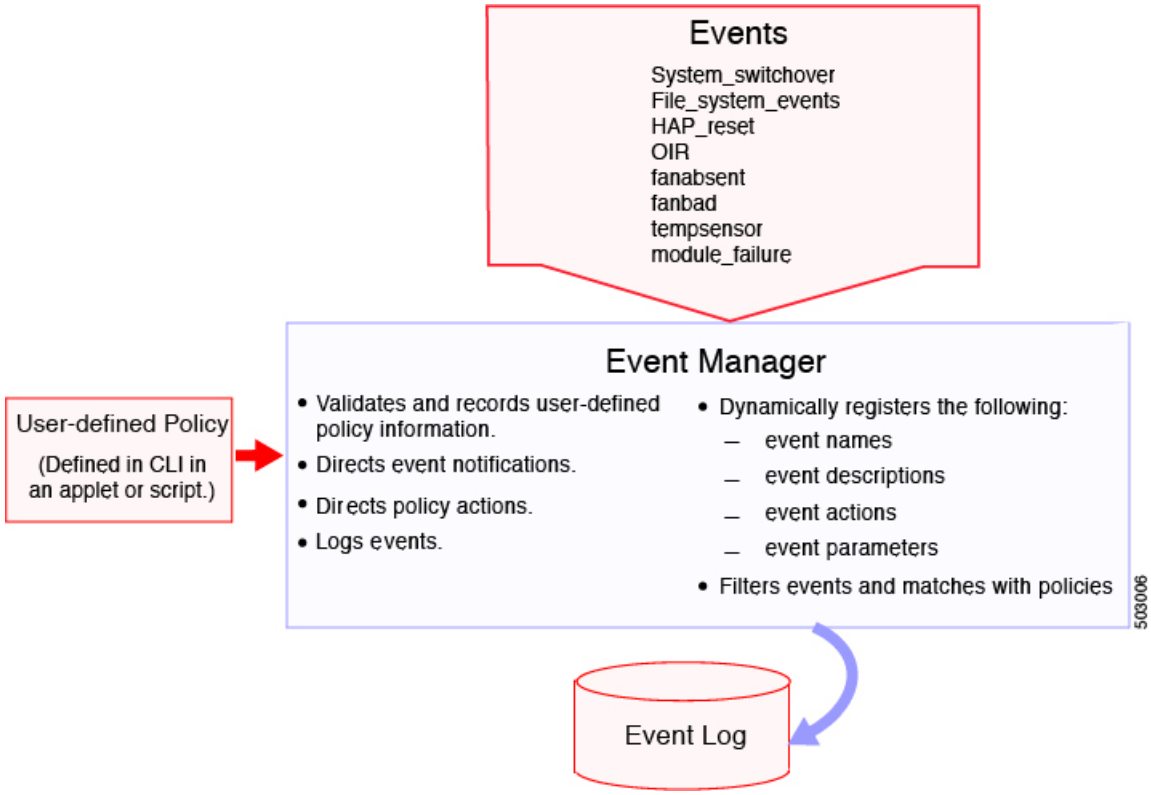
イベント文

イベントは、回避、通知など、何らかのアクションが必要なデバイスアクティビティです。これらのイベントは通常、インターフェイスやファンの誤動作といったデバイスの障害に関連します。

EEM ではイベント フィルタを定義して、クリティカル イベントまたは指定された時間内で繰り返し発生したイベントだけが関連付けられたアクションのトリガーになるようにします。

この図は、EEM によって処理されたイベントを示します。

図 *6 : EEM* の概要



イベント文では、ポリシー実行のトリガーになるイベントを指定します。複数イベント トリガーを設定できます。

EEM はイベント文に基づいてポリシーをスケジューリングし、実行します。EEM はイベントおよびアクション コマンドを検証し、定義に従ってコマンドを実行します。

（注） 発生したイベントでデフォルトのアクションを処理できるようにする場合は、event-default アクション文を許可して EEM ポリシーを設定する必要があります。

アクション文

アクション文では、ポリシーによって実行されるアクションを記述します。各ポリシーに複数のアクション文を設定できます。ポリシーにアクションを関連付けなかった場合、EEM はイベント観察を続けますが、アクションは実行されません。

EEM がアクション文でサポートするアクションは、次のとおりです。

- CLI コマンドの実行。
- カウンタのアップデート。
- 例外の記録。
- モジュールの強制的シャットダウン
- デバイスをリロードします。
- 電力のバジェット超過による特定モジュールのシャットダウン。
- Syslog メッセージの生成。
- Call Home イベントの生成。
- SNMP 通知の生成。
- システム ポリシー用デフォルト アクションの使用。

（注） EEM は、合計 1024 文字までの、完全なアクション CLI リストのみを処理できます。さらにアクションが必要な場合は、同じトリガーを持つ新しい冗長アプレットとして定義する必要があります。

（注） 発生したイベントでデフォルトのアクションを処理できるようにする場合は、デフォルトのアクションを許可する EEM ポリシーを設定する必要があります。たとえば、match 文で CLI コマンドを照合する場合、EEM ポリシーに event-default アクション文を追加する必要があります。この文がないと、EEM では CLI コマンドを実行できません。

（注） ユーザ ポリシーまたは上書きポリシーの中に、相互に否定したり、関連付けられたシステム ポリシーに悪影響を与えたりするようなアクション文がないかどうかを確認してください。

VSH スクリプト ポリシー

テキスト エディタを使用し、VSH スクリプトでポリシーを作成することもできます。このようなポリシーにも、他のポリシーと同様、イベント文およびアクション文（複数可）を使用します。また、これらのポリシーでシステム ポリシーを補うことも上書きすることもできます。VSH スクリプト ポリシーの作成後、そのポリシーをデバイスにコピーしてアクティブにします。

環境変数

すべてのポリシーに使用できる、EEM の環境変数を定義できます。環境変数は、複数のポリシーで使用できる共通の値を設定する場合に便利です。たとえば、外部電子メール サーバの IP アドレスに対応する環境変数を作成できます。

パラメータ置換フォーマットを使用することによって、アクション文で環境変数を使用できます。

この例では、「EEM action」というリセット理由を指定し、モジュール 1 を強制的にシャットダウンするアクション文の例を示します。

```
switch (config-eem-policy)# action 1.0 forceshut module 1 reset-reason "EEM action."
```

シャットダウンの理由に default-reason という環境変数を定義すると、次の例のように、リセット理由を環境変数に置き換えることができます。

```
switch (config-eem-policy)# action 1.0 foreshut module 1 reset-reason $default-reason
```

この環境変数は、任意のポリシーで再利用できます。

EEM イベント相関

イベントの組み合わせに基づいて EEM ポリシーをトリガーできます。まず、**tag** キーワードを使用して EEM ポリシーに複数のイベントを作成し区別します。次に、一連のブール演算子（**AND**、**OR**、**ANDNOT**）を使用して、回数および時間をもとに、カスタム処理をトリガーするこれらのイベントの組み合わせを定義できます。

高可用性

Cisco NX-OS は、EEM のステートレス リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバーの後、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化のサポート

アクションまたはイベントがすべて表示されるわけではありません。ポリシーを設定するには、network-admin の権限が必要です。

EEM の前提条件

EEM の前提条件は、次のとおりです。

- EEM を設定するには、network-admin のユーザ権限が必要です。

EEM の注意事項と制約事項

EEM 設定時の注意事項と制約事項は次のとおりです。

- 設定可能な EEM ポリシーの最大数は 500 です。
- ユーザポリシーまたは上書きポリシー内のアクション文が、相互に否定したり、関連付けられたシステムポリシーに悪影響を与えたりするようなことがないようにする必要があります。
- 発生したイベントでデフォルトのアクションを処理できるようにするには、デフォルトのアクションを許可する EEM ポリシーを設定する必要があります。たとえば、match 文で CLI コマンドを照合する場合、EEM ポリシーに event-default アクション文を追加する必要があります。この文がないと、EEM では CLI コマンドを実行できません。
- 同じクライアントからは 10 個のトリガーのみ（たとえば、vshd は「イベント cli」のクライアント、snmp は「イベント snmp」のクライアントなど）が 1 秒以内に発行されます。
- オプション **collect** のアクションは、常に event applet action 文の最初のアクションである必要があります。
- イベント ログの自動収集とバックアップには、次の注意事項があります。
  - デフォルトでは、スイッチのログ収集を有効にすると、サイズ、規模、コンポーネントのアクティビティに応じて、15 分から数時間のイベントログが利用できるようになります。
  - 長期間にわたる関連ログを収集できるようにするには、必要な特定のサービス/機能に対してのみイベントログの保持を有効にします。「単一サービスの拡張ログ ファイル保持の有効化」を参照してください。内部イベントログをエクスポートすることもできます。「外部ログ ファイル ストレージ」を参照してください。
  - トラブルシューティングを行うときは、内部イベントログのスナップショットを手動によりリアルタイムで収集することをお勧めします。「最近のログ ファイルのローカル コピーの生成」を参照してください。

- **show tech** コマンドを収集するように EEM ポリシーアクションを設定する場合は、同じアクションが再度呼び出される前に、**show tech** コマンドが完了するのに十分な時間を割り当ててください。
- オーバーライド ポリシーについては、次の点に注意してください。
  - イベント文が指定されていても、アクション文が指定されていない上書きポリシーを設定した場合、アクションは開始されません。また、障害も通知されません。
  - 上書きポリシーにイベント文が含まれていないと、システム ポリシーで可能性のあるイベントがすべて上書きされます。
- 正規コマンド式には、次のルールが適用されます。
  - すべての正規表現は、Portable Operating System Interface for uniX（POSIX）拡張標準に準拠している必要があります。
  - すべてのキーワードを展開する必要があります。
  - 引数の置換には * 記号のみを使用できます。
- EEM イベント相関については、次の点に注意してください。
  - EEM イベント相関はスーパーバイザ モジュールだけでサポートされます。
  - EEMイベント相関は、単一ポリシー内の別のモジュール間ではサポートされません。
  - EEM イベント相関は 1 つのポリシーに最大 4 つのイベント文をサポートします。イベントタイプは同じでも別でもかまいませんが、サポートされるイベントタイプは、cli、カウンタ、モジュール、モジュール障害、oir、snmp、syslog だけです。
  - EEM イベント相関はシステムのデフォルト ポリシーを上書きしません。
- 複数のイベント文が EEM ポリシーに存在する場合は、各イベント文に **tag** キーワードと一意な tag 引数が必要です。
- デフォルトアクション実行は、タグ付きのイベントで設定されているポリシーではサポートされません。
- Python から EEM を呼び出すことができます。Python の詳細については、『Cisco Nexus 9000 シリーズ NX-OS プログラマビリティ ガイド』を参照してください。
- Cisco NX-OS リリース 10.3(1)F 以降、デフォルトの自動収集はシステム スイッチオーバーでサポートされていません。システムの切り替え時に、新しいアクティブ スーパーバイザで bloggerd auto-collect コマンドを再実行して、それぞれのコンポーネントの自動収集を有効にしてください。
- Cisco NX-OS リリース 10.3(3)F 以降、デフォルトのブロガー自動収集は、adjmgr、cts、l2fm、および vmtracker でサポートされます。
- Cisco NX-OS リリース 10.4(1)F 以降、デフォルトのブロガー自動収集は、ipqosmgr、aclqos、cfs、ethport、feature-mgr、icam、interface manager、lacp、m2rib、mfdm、nbm、ngoam、

nve、port-channel、qos、sla_responder、sla_sender、sla_twamp、smm、spm、sysmgr、および vpc でサポートされます。

- 構成可能な最小消去時間が 0 時間から 48 時間に増加しました。
- 14 日を超えたファイルは、予約済みのブートフラッシュ領域（最大 5%）に関係なく、自動的に消去されます。
- 自動収集用に予約されたスペースが最大まで使用されている場合、ファイルの消去または手動によるファイルの削除によって予約済みスペースにまた空きができるまで、新しい自動収集は拒否されます。

EEM のデフォルト設定

この表では、EEM のデフォルト設定を一覧にしています。

パラメータ	デフォルト
システム ポリシー	アクティブ

EEM の設定

システム ポリシーに基づいて実行されるアクションを含むポリシーを作成できます。システム ポリシーに関する情報を表示するには、**show event manager system-policy** コマンドを使用します。

環境変数の定義

EEM ポリシーでパラメータとして機能する変数を定義できます。

手順の概要

1. **configure terminal**
2. **event manager environment** *variable-name variable-value*
3. （任意） **show event manager environment** {*variable-name* | **all**}
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**event manager environment** *variable-name variable-value* 例： `switch(config)# event manager environment emailto “admin@anyplace.com”`	EEM 用の環境変数を作成します。*variable-name* は大文字と小文字を区別し、最大 29 文字の英数字を使用できます。*variable-value* には最大 39 文字の英数字を引用符で囲んで使用できます。
ステップ 3	（任意） **show event manager environment** {*variable-name* | **all**} 例： `switch(config)# show event manager environment all`	設定した環境変数に関する情報を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

CLI によるユーザ ポリシーの定義

CLI を使用して、デバイスにユーザ ポリシーを定義できます。

手順の概要

1. **configure terminal**
2. **event manager applet** *applet-name*
3. （任意） **description** *policy-description*
4. **event** *event-statement*
5. （任意） **tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | **or** {*tag*}] {**happens** *occurs in seconds*}
6. **action** *number*[*.number2*] *action-statement*
7. （任意） **show event manager policy-state** *name* [**module** *module-id*]
8. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**event manager applet** *applet-name*
	例：
	`switch(config)# event manager applet monitorShutdown`
	`switch(config-applet)#`
		EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。*applet-name* は大文字と小文字を区別し、最大 29 文字の英数字を使用できます。

ステップ 3	（任意） **description** *policy-description* 例： `switch(config-applet)# description "Monitors interface shutdown."`	ポリシーの説明になるストリングを設定します。string には最大 80 文字の英数字を使用できます。ストリングは引用符で囲みます。
ステップ 4	**event** *event-statement* 例： `switch(config-applet)# event cli match "conf t ; interface * ; shutdown"`	ポリシーのイベント文を設定します。イベント文が複数ある場合、このステップを繰り返します。「イベント文の設定 （357 ページ）」を参照してください。
ステップ 5	（任意） **tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | **or** {*tag*}] {**happens** *occurs in seconds*} 例： `switch(config-applet)# tag one or two happens 1 in 10000`	ポリシー内の複数のイベントを相互に関連付けます。 *occurs* 引数の範囲は 1 ～ 4294967295 です。*seconds* 引数の範囲は 0 ～ 4294967295 秒です。
ステップ 6	**action** *number*[.*number2*] *action-statement* 例： `switch(config-applet)#action 1.0 cli show interface Ethernet 3/1`	ポリシーのアクション文を設定します。アクション文が複数ある場合、このステップを繰り返します。「アクション文の設定 （363 ページ）」を参照してください。
ステップ 7	（任意） **show event manager policy-state** *name* [**module** *module-id*] 例： `switch(config-applet)# show event manager policy-state monitorShutdown`	設定したポリシーの状態に関する情報を表示します。
ステップ 8	（任意） **copy running-config startup-config** 例：	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

	コマンドまたはアクション	目的
	`switch(config)# copy running-config startup-config`	

イベント文の設定

イベント文を設定するには、アプレット コンフィギュレーション モードで次のいずれかのコマンドを使用します。

コマンド	目的
event application [**tag** *tag*] **sub-system** *sub-system-id* **type** *event-type* **例:** `switch(config-applet)# event application sub-system 798 type 1`	イベントの指定がサブシステムIDおよびアプリケーションイベントタイプに一致する場合に、イベントを発生させます。 *sub-system-id* と *event-type* の範囲は 1 ～ 4294967295 です。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 （注） このコマンドを使用するには、まず **feature evmed** コマンドを有効にして一般的なイベントディテクタを有効にする必要があります。
event cli [**tag** *tag*] **match** *expression* [**count** *repeats* | **time** *seconds*] **例:** `switch(config-applet)# event cli match "conf t ; interface * ; shutdown"`	正規表現と一致するコマンドが入力された場合に、イベントを発生させます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 *repeats* の範囲は 1 ～ 65000 です。time の範囲は 0 ～ 4294967295 秒です。0 は無制限を示します。

コマンド	目的
event counter [**tag** *tag*] **name** *counter* **entry-val** *entry* **entry-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} [**exit-val** *exit* **exit-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**}] **例:** `switch(config-applet)# event counter name mycounter entry-val 20 gt`	カウンタが、開始演算子に基づいて開始のしきい値を超えた場合にイベントを発生させます。イベントはただちにリセットされます。任意で、カウンタが終了のしきい値を超えたあとでリセットされるように、イベントを設定できます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 *counter* name は大文字と小文字を区別し、最大 28 の英数字を使用できます。*entry* および *exit* の値の範囲は 0 ～ 2147483647 です。
event fanabsent [**fan** *number*] **time** *seconds* **例:** `switch(config-applet)# event fanabsent time 300`	秒数で設定された時間を超えて、ファンがデバイスから取り外されている場合に、イベントを発生させます。*number* の範囲はモジュールに依存します。*seconds* の範囲は 10 ～ 64000 です。
event fanbad [**fan** *number*] **time** *seconds* **例:** `switch(config-applet)# event fanbad time 3000`	秒数で設定された時間を超えて、ファンが故障状態の場合に、イベントを発生させます。*number* の範囲はモジュールに依存します。*seconds* の範囲は 10 ～ 64000 です。
event fib {**adjacency extra** | **resource tcam usage** | **route** {**extra** | **inconsistent** | **missing**}} **例:** `switch(config-applet)# event fib adjacency extra`	次のいずれかに対するイベントを発生させます。 • **adjacency extra**：ユニキャスト FIB に追加のルートがある場合。 • **resource tcam usage**：TCAM 使用率がいずれかの方向で 5 の倍数になるごとに。 • **route** {**extra** | **inconsistent** | **missing**}：ユニキャスト FIB でルートが追加、変更、または削除される場合。
event gold module {*slot* | **all**} **test** *test-name* [**severity** {**major** | **minor** | **moderate**}] **testing-type** {**bootup** | **monitoring** | **ondemand** | **scheduled**} **consecutive-failure** *count* **例:** `switch(config-applet)# event gold module 2 test ASICRegisterCheck testing-type ondemand consecutive-failure 2`	名前で指定されたオンライン診断テストが、設定された回数だけ連続して、設定された重大度で失敗した場合に、イベントを発生させます。*slot* の範囲は 1 ～ 10 です。*test-name* は設定されたオンライン診断テストの名前です。*count* の範囲は 1 ～ 1000 です。

コマンド	目的
event memory {**critical** | **minor** | **severe**} **例:** `switch(config-applet)# event memory critical`	メモリのしきい値を超えた場合にイベントを発生させます。メモリのしきい値の設定（367 ページ）も参照してください。
event module [**tag** *tag*] **status** {**online** | **offline** | **any**} **module** {**all** | *module-num*} **例:** `switch(config-applet)# event module status offline module all`	指定したモジュールが選択された状態になったときにイベントを発生させます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。
event module-failure [**tag** *tag*] **type** *failure-type* **module** {*slot* | **all**} **count** *repeats* [**time** *seconds*] **例:** `switch(config-applet)# event module-failure type lc-failed module 3 count 1`	モジュールが設定された障害タイプになった場合に、イベントを発生させます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 *repeats* 範囲は 0 ～ 4294967295 です。*seconds* の範囲は 0 ～ 4294967295 秒です。0 は無制限を示します。
event none **例:** `switch(config-applet)# event none`	手動で指定されたイベントがないポリシー イベントを実行します。 （注） このコマンドを使用するには、まず **feature evmed** コマンドを有効にして一般的なイベントディテクタを有効にする必要があります。

コマンド	目的
event oir [**tag** *tag*] {**fan** | **module** | **powersupply**} {**anyoir** | **insert** | **remove**} [*number*] **例:** `switch(config-applet)# event oir fan remove 4`	設定されたデバイス構成要素（ファン、モジュール、または電源モジュール）がデバイスに取り付けられた場合、またはデバイスから取り外された場合に、イベントを発生させます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 任意で、ファン、モジュール、または電源モジュールの具体的な番号を設定できます。*number* の範囲は次のとおりです。 • ファン番号：モジュール依存 • モジュール番号：デバイス依存 • 電源モジュール番号：範囲は 1 ～ 3
event policy-default count *repeats* [**time** *seconds*] **例:** `switch(config-applet)# event policy-default count 3`	システム ポリシーで設定されているイベントを使用します。このオプションは、ポリシーを上書きする場合に使用します。 *repeats* の範囲は 1 ～ 65000 です。*seconds* の範囲は 0 ～ 4294967295 秒です。0 は無制限を示します。
event poweroverbudget **例:** `switch(config-applet)# event poweroverbudget`	電力バジェットが設定された電源モジュールの容量を超えた場合に、イベントを発生させます。

コマンド	目的
event snmp [**tag** *tag*] **oid** *oid* **get-type** {**exact** | **next**} **entry-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} **entry-val** *entry* [**exit-comb** {**and** | **or**}] **exit-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} **exit-val** *exit* **exit-time** *time* **polling-interval** *interval* **例:** `switch(config-applet)# event snmp oid 1.3.6.1.2.1.31.1.1.1.6 get-type next entry-op lt 300 entry-val 0 exit-op eq 400 exit-time 30 polling-interval 300`	SNMP OID が、開始演算子に基づいて開始のしきい値を超えた場合にイベントを発生させます。イベントはただちにリセットされます。または任意で、カウンタが終了のしきい値を超えたあとでリセットされるように、イベントを設定できます。OID はドット付き10 進表記です。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 *entry* および *exit* の値の範囲は 0 ～ 18446744073709551615 です。time の範囲は 0 ～ 2147483647 秒です。interval の範囲は 1 ～ 2147483647 秒です。
event storm-control **例:** `switch(config-applet)# event storm-control`	ポート上のトラフィックが設定されたストーム制御しきい値を超えた場合に、イベントを発生させます。
event syslog [**occurs** *count*] {**pattern** *string* | **period** *time* | **priority** *level* | **tag** *tag*} **例:** `switch(config-applet)# event syslog period 500`	指定した syslog のしきい値を超えた場合にイベントを発生させます。カウントの範囲は 1 ～ 65000 で、時間の範囲は 1 ～ 4294967295 です。プライオリティの範囲は 0 ～ 7 です。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。
event sysmgr memory [**module** *module-num*] **major** *major-percent* **minor** *minor-percent* **clear** *clear-percent* **例:** `switch(config-applet)# event sysmgr memory minor 80`	指定したシステム マネージャのメモリのしきい値を超えた場合にイベントを発生させます。パーセンテージの範囲は 1 ～ 99 です。
event sysmgr switchover count *count* **time** *interval* **例:** `switch(config-applet)# event sysmgr switchover count 10 time 1000`	指定した switchover count が、指定した time interval を超えた場合にイベントを発生させます。switchover count の範囲は 1 ～ 65000 です。time interval の範囲は 0 ～ 2147483647 です。

コマンド	目的
event temperature [**module** *slot*] [*sensor-number*] **threshold** {**any** | **major** | **minor**} **例:** `switch(config-applet)# event temperature module 2 threshold any`	温度センサーが設定されたしきい値を超えた場合に、イベントを発生させます。sensor の範囲は 1 ～ 18 です。
event timer {**absolute time** *time* **name** *name* | **countdown time** *time* **name** *name* | **cron cronentry** *string* | **tag** *tag* | **watchdog time** *time* **name** *name*} **例:** `switch(config-applet)# event timer absolute time 100 name abtimer`	指定した時間に到達した場合に、イベントを発生させます。時間の範囲は 1 ～ 4294967295 です。 • **absolute time**：指定された絶対時刻が発生した場合に、イベントを発生させます。 • **countdown time**：指定された時間がゼロにカウントダウンされたときに、イベントを発生させます。タイマーはリセットされません。 • **cron cronentry**：CRON 文字列の指定が現在時刻に一致する場合に、イベントを発生させます。 • **watchdog time**：指定された時間がゼロにカウントダウンされたときに、イベントを発生させます。タイマーは、初期値に自動的にリセットされ、カウント ダウンが続行されます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 （注） このコマンドを使用するには、まず **feature evmed** コマンドを有効にして一般的なイベントディテクタを有効にする必要があります。
event track [**tag** *tag*] *object-number* **state** {**any** | **down** | **up**} **例:** `switch(config-applet)# event track 1 state down`	トラッキング対象オブジェクトが設定された状態になった場合に、イベントを発生させます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 指定できる *object-number* の範囲は 1 ～ 500 です。

アクション文の設定

アクション文を設定するには、EEM コンフィギュレーション モードで次のいずれかのコマンドを使用します。

コマンド	目的
action *number*[*.number2*] **cli** *command1* [*command2...*] [**local**] **例:** `switch(config-applet)# action 1.0 cli show interface Ethernet 3/1`	設定された CLI コマンドを実行します。任意で、イベントが発生したモジュール上でコマンドを実行できます。アクション ラベルのフォーマットは *number1.number2* です。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
action *number*[*.number2*] **counter name** *counter* **value** *val* **op** {**dec** | **inc** | **nop** | **set**} **例:** `switch(config-applet)# action 2.0 counter name mycounter value 20 op inc`	設定された値および操作でカウンタを変更します。アクション ラベルのフォーマットは *number1.number2* です。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。 counter name は大文字と小文字を区別し、最大 28 の英数字を使用できます。*val* には 0 ～ 2147483647 の整数または置換パラメータを指定できます。
action *number*[*.number2*] **event-default** **例:** `switch(config-applet)# action 1.0 event-default`	関連付けられたイベントのデフォルトアクションを実行します。アクション ラベルのフォーマットは *number1.number2* です。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
action *number*[*.number2*] **forceshut** [**module** *slot* | **xbar** *xbar-number*] **reset-reason** *seconds* **例:** `switch(config-applet)# action 1.0 forceshut module 2 reset-reason "flapping links"`	モジュール、クロスバー、またはシステム全体を強制的にシャットダウンします。アクション ラベルのフォーマットは *number1.number2* です。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。 リセット理由は、引用符で囲んだ最大 80 文字の英数字ストリングです。
action *number*[*.number2*] **overbudgetshut** [**module** *slot*[*-slot*]] **例:** `switch(config-applet)# action 1.0 overbudgetshut module 3-5`	電力バジェット超過の問題により、1 つまたは複数のモジュールまたはシステム全体を強制的にシャットダウンします。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。

コマンド	目的
action *number*[*.number2*] **policy-default** **例:** `switch(config-applet)# action 1.0 policy-default`	上書きしているポリシーのデフォルトアクションを実行します。アクション ラベルのフォーマットは *number1.number2* です。 *number* は16桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
action *number*[*.number2*] **publish-event** **例:** `switch(config-applet)# action 1.0 publish-event`	アプリケーション固有のイベントの発行を強制します。アクション ラベルのフォーマットは *number1.number2* です。 *number* は16桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
action *number*[*.number2*] **reload** [**module** *slot*[*-slot*]] **例:** `switch(config-applet)# action 1.0 reload module 3-5`	1つまたは複数のモジュールまたはシステム全体を強制的にリロードします。 *number* は16桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
action *number*[*.number2*] **snmp-trap** {[**intdata1** *data* [**intdata2** *data*]] [**strdata** *string*]} **例:** `switch(config-applet)# action 1.0 snmp-trap strdata "temperature problem"`	設定されたデータを使用して SNMP トラップを送信します。*number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。 *data* 引数には、最大80桁の任意の数を指定できます。*string* には最大 80 文字の英数字を使用できます。
action *number*[*.number2*] **syslog** [**priority** *prio-val*] **msg** *error-message* **例:** `switch(config-applet)# action 1.0 syslog priority notifications msg "cpu high"`	設定されたプライオリティで、カスタマイズした syslog メッセージを送信します。*number* は16桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。 *error-message* には最大 80 文字の英数字を引用符で囲んで使用できます。

（注） 発生したイベントでデフォルトのアクションを処理できるようにする場合は、デフォルトのアクションを許可する EEM ポリシーを設定する必要があります。たとえば、match 文で CLI コマンドを照合する場合、EEM ポリシーに event-default アクション文を追加する必要があります。この文がないと、EEM では CLI コマンドを実行できません。**terminal event-manager bypass** manager bypass コマンドを使用して、CLI での すべての EEM ポリシーを、CLI コマンドの実行と一致させることができます。

VSH スクリプトによるポリシーの定義

VSH スクリプトを使用してポリシーを定義できます。

始める前に

管理者の権限でログインしていることを確認します。

スクリプト名がスクリプト ファイル名と同じ名前であることを確認します。

手順

ステップ 1 テキスト エディタで、ポリシーを定義するコマンド リストを指定します。

ステップ 2 テキスト ファイルに名前をつけて保存します。

ステップ 3 次のシステム ディレクトリにファイルをコピーします。bootflash://eem/user_script_policies

VSH スクリプト ポリシーの登録およびアクティブ化

VSH スクリプトで定義したポリシーを登録してアクティブにできます。

手順の概要

1. **configure terminal**
2. **event manager policy** *policy-script*
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**event manager policy** *policy-script* 例： `switch(config)# event manager policy moduleScript`	EEM スクリプト ポリシーを登録してアクティブにします。*policy-script* は大文字と小文字を区別し、最大 29 の英数字を使用できます。

	コマンドまたはアクション	目的
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ポリシーの上書き

システム ポリシーは上書き可能です。

手順の概要

1. **configure terminal**
2. （任意） **show event manager policy-state** *system-policy*
3. **event manager applet** *applet-name* **override** *system-policy*
4. （任意） **description** *policy-description*
5. **[no] event** *event-statement*
6. **action** *number action-statement*
7. （任意） **show event manager policy-state** *name*
8. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	（任意） **show event manager policy-state** *system-policy*
	例：
	`switch(config-applet)# show event manager policy-state __ethpm_link_flap`
	`Policy __ethpm_link_flap`
	`Cfg count : 5`
	`Cfg time interval : 10.000000 (seconds)`
	`Hash default, Count 0`
		上書きするシステム ポリシーの情報をしきい値を含めて表示します。システム ポリシー名を突き止めるには、**show event manager system-policy** コマンドを使用します。システム ポリシーについては、Embedded Event Manager システム イベントおよび設定例 （707 ページ） を参照してください。

ステップ 3	**event manager applet** *applet-name* **override** *system-policy*
	例：
	`switch(config)# event manager applet ethport override __ethpm_link_flap`
	`switch(config-applet)#`
		システム ポリシーを上書きし、アプレット コンフィギュレーション モードを開始します。*applet-name* は大文字と小文字を区別し、最大 29 文字の英数字

	コマンドまたはアクション	目的
		を使用できます。*system-policy* は、システム ポリシーの 1 つにする必要があります。
ステップ 4	（任意） **description** *policy-description* 例： `description "Overrides link flap policy."`	ポリシーの説明になるストリングを設定します。string には最大 80 文字の英数字を使用できます。ストリングは引用符で囲みます。
ステップ 5	必須: **[no] event** *event-statement* 例： `switch(config-applet)# event policy-default count 2 time 1000`	ポリシーのイベント文を構成します。 構成を削除するには、このコマンドの **no** 形式を使用します。
ステップ 6	必須: **action** *number action-statement* 例： `switch(config-applet)# action 1.0 syslog priority warnings msg "Link is flapping."`	ポリシーのアクション文を設定します。 アクション文が複数ある場合、このステップを繰り返します。
ステップ 7	（任意） **show event manager policy-state** *name* 例： `switch(config-applet)# show event manager policy-state ethport`	設定したポリシーに関する情報を表示します。
ステップ 8	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

メモリのしきい値の設定

イベントを発生させるメモリしきい値を設定し、オペレーティング システムがメモリを割り当てられない場合にプロセスを終了させるかどうかを設定できます。

始める前に

管理者の権限でログインしていることを確認します。

手順の概要

1. **configure terminal**
2. **system memory-thresholds minor** *minor* **severe** *severe* **critical** *critical*
3. （任意） **system memory-thresholds threshold critical no-process-kill**
4. （任意） **show running-config | include "system memory"**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**system memory-thresholds minor** *minor* **severe** *severe* **critical** *critical* 例： `switch(config)# system memory-thresholds minor 60 severe 70 critical 80`	EEM メモリ イベントを生成するシステム メモリしきい値を設定します。デフォルト値は次のとおりです。 • マイナー - 85 • 深刻 - 90 • 重大 - 95 これらのメモリのしきい値を超えた場合、システムは次の syslog を生成します。 • 2013 May 7 17:06:30 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR • 2013 May 7 17:06:30 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE • 2013 May 7 17:06:30 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL • 2013 May 7 17:06:35 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR ALERT RECOVERED • 2013 May 7 17:06:35 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE ALERT RECOVERED • 2013 May 7 17:06:35 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL ALERT RECOVERED
ステップ 3	（任意） **system memory-thresholds threshold critical no-process-kill** 例：	メモリを割り当てられない場合もプロセスを終了しないようにシステムを設定します。デフォルト値では、最もメモリを消費するプロセスから終了できます。

	コマンドまたはアクション	目的
	`switch(config)# system memory-thresholds threshold critical no-process-kill`	
ステップ 4	（任意） **show running-config | include "system memory"** 例： `switch(config-applet)# show running-config | include "system memory"`	システム メモリ設定に関する情報を表示します。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

EEM パブリッシャとしての syslog の設定

スイッチからの syslog メッセージをモニタできます。

（注） syslog メッセージをモニタする検索文字列の最大数は 10 です。

始める前に

EEM は、Syslog による登録に使用可能である必要があります。

Syslog デーモンが設定され、実行される必要があります。

手順の概要

1. **configure terminal**
2. **event manager applet** *applet-name*
3. **event syslog** [**tag** *tag*] {**occurs** *number* | **period** *seconds* | **pattern** *msg-text* | **priority** *priority*}
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的

ステップ 2	**event manager applet** *applet-name*
	例：
	`switch(config)# event manager applet abc`
	`switch(config-applet)#`
		EEM にアプレットを登録し、アプレット 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	**event syslog** [**tag** *tag*] {**occurs** *number* | **period** *seconds* | **pattern** *msg-text* | **priority** *priority*} 例： `switch(config-applet)# event syslog occurs 10`	syslog メッセージを監視し、ポリシーの検索文字列に基づいてポリシーを呼び出します。 • **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 • **occurs** *number* のキーワードと引数のペアは、発生回数を指定します。指定できる範囲は 1 ～ 65000 です。 • **period** *seconds* のキーワードと引数のペアは、発生回数を指定します。値の範囲は 1 ～ 4294967295 です。 • **pattern** *msg-text* のキーワードと引数のペアは、マッチさせる正規表現を指定します。パターンには、文字テキスト、環境変数、またはこの 2 つの組み合わせを含めることができます。文字列に空白が含まれる場合は引用符で囲みます。 • **priority** *priority* のキーワードと引数のペアは、syslog メッセージのプライオリティを指定します。このキーワードを指定しないと、すべての Syslog メッセージのプライオリティ レベルが「情報レベル」に設定されます。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

EEM の設定確認

EEM 設定情報を表示するには、次のいずれかの作業を実行します。

コマンド	目的
show event manager environment [*variable-name* | **all**]	イベント マネージャの環境変数に関する情報を表示します。

コマンド	目的
show event manager event-types [*event* | **all** | **module** *slot*]	イベントマネージャのイベントタイプに関する情報を表示します。
show event manager history events [**detail**] [**maximum** *num-events*] [**severity** {**catastrophic** | **minor** | **moderate** | **severe**}]	すべてのポリシーについて、イベント履歴を表示します。
show event manager policy-state *policy-name*	しきい値を含め、ポリシーの状態に関する情報を表示します。
show event manager script system [*policy-name* | **all**]	スクリプト ポリシーに関する情報を表示します。
show event manager system-policy [**all**]	定義済みシステム ポリシーに関する情報を表示します。
show running-config eem	EEM の実行コンフィギュレーションに関する情報を表示します。
show startup-config eem	EEM のスタートアップ コンフィギュレーションに関する情報を表示します。

EEM の設定例

モジュール 3 の中断のないアップグレード エラーのしきい値だけを変更することによって、__lcm_module_failure システム ポリシーを上書きする方法の例を示します。この例では、syslog メッセージも送信されます。その他のすべての場合、システム ポリシー __lcm_module_failure の設定値が適用されます。

```
event manager applet example2 override __lcm_module_failure
event module-failure type hitless-upgrade-failure module 3 count 2
action 1 syslog priority errors msg module 3 “upgrade is not a hitless upgrade!”
action 2 policy-default
```

__ethpm_link_flap システム ポリシーを上書きし、インターフェイスをシャットダウンする方法の例を示します。

```
event manager applet ethport override __ethpm_link_flap
event policy-default count 2 time 1000
action 1 cli conf t
action 2 cli int et1/1
action 3 cli no shut
```

CLI コマンドの実行を許可し、ユーザがデバイスでコンフィギュレーション モードを開始すると SNMP 通知を送る EEM ポリシーを作成する例を示します。

```
event manager applet TEST
event cli match "conf t"
action 1.0 snmp-trap strdata "Configuration change"
```

```
action 2.0 event-default
```

（注） EEM ポリシーに **event-default** アクション文を追加する必要があります。この文がないと、EEM では CLI コマンドを実行できません。

次に、EEM ポリシーの複数イベントを関連付け、イベント トリガーの組み合わせに基づいてポリシーを実行する例を示します。この例では、EEM ポリシーは、指定された syslog パターンのいずれかが 120 秒以内に発生したときにトリガーされます。

```
event manager applet eem-correlate
event syslog tag one pattern "copy bootflash:.* running-config.*"
event syslog tag two pattern "copy run start"
event syslog tag three pattern "hello"
tag one or two or three happens 1 in 120
action 1.0 reload module 1
```

最大障害しきい値に達すると、AsicMemory、FpgaRegTest、および L2ACLRedirect システム ポリシーによってスイッチのリロードが強制されます。次に、これらのポリシーのいずれかのデフォルト アクションを上書きし、代わりに syslog を発行する例を示します。

```
event manager applet gold override __fpgareg
action 1 syslog priority emergencies msg FpgaRegTest_override
```

次に、デフォルト ポリシーを上書きし、デフォルト アクションを実行する例を示します。

```
event manager applet gold_fpga_ovrd override __fpgareg
  action 1 policy-default
  action 2 syslog priority emergencies msg FpgaRegTest_override
```

（注） その他の設定例については、「Embedded Event Manager システム イベントおよび設定例 （707 ページ）」を参照してください。

イベント ログの自動収集とバックアップ

自動的に収集されたイベント ログは、スイッチのメモリにローカルに保存されます。イベント ログ ファイル ストレージは、一定期間ファイルを保存する一時バッファです。時間が経過すると、バッファのロールオーバーによって次のファイルのためのスペースが確保されます。ロールオーバーでは、先入れ先出し方式が使用されます。

Cisco NX-OS リリース 9.3(3) 以降、EEM は以下の収集およびバックアップ方法を使用します。

- 拡張ログ ファイルの保持
- トリガーベースのイベント ログの自動収集

拡張ログ ファイルの保持

Cisco NX-OS リリース 9.3 (3) 以降、すべての Cisco Nexus プラットフォーム スイッチは、少なくとも 8 GB のシステムメモリを備え、イベント ロギング ファイルの拡張保持をサポートします。ログ ファイルをスイッチにローカルに保存するか、外部コンテナを介してリモートに保存すると、ロールオーバーによるイベント ログの損失を削減できます。

すべてのサービスの拡張ログ ファイル保持のイネーブル化

拡張ログ ファイル保持は、スイッチで実行されているすべてのサービスに対してデフォルトで有効になっています。スイッチでログファイル保持機能がイネーブルになっていない場合（**no bloggerd log-dump** が設定されている場合）、次の手順を使用してイネーブルにします。

手順の概要

1. **configure terminal**
2. **bloggerd log-dump all**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**bloggerd log-dump all**
	例：
	`switch(config)# bloggerd log-dump all`
	`switch(config)#`
		すべてのサービスのログ ファイル保持機能をイネーブルにします。

例

```
switch# configure terminal
switch(config)# bloggerd log-dump all
Sending Enable Request to Bloggerd
Bloggerd Log Dump Successfully enabled
switch(config)#
```

すべてのサービスの拡張ログ ファイル保持の無効化

拡張ログ ファイル保持は、スイッチで実行されているすべてのサービスに対してデフォルトで有効になっています。

スイッチのログファイル保持機能がすべてのサービスに対して有効になっている場合は、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **no bloggerd log-dump all**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**no bloggerd log-dump all**
	例：
	`switch(config)# no bloggerd log-dump all`
	`switch(config)#`
		スイッチ上のすべてのサービスのログ ファイル保持機能を無効にします。

例

```
switch# configure terminal
switch(config)# no bloggerd log-dump all
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)#
```

単一サービスの拡張ログファイル保持の有効化

この手順を使用して、単一のサービスのログ ファイルの保持を有効にします。

手順の概要

1. **show system internal sysmgr service name** *service-type*
2. **configure terminal**
3. **bloggerd log-dump sap** *number*
4. **show system internal bloggerd info log-dump-info**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**show system internal sysmgr service name** *service-type* 例： `switch# show system internal sysmgr service name aclmgr`	サービス SA P番号を含む ACL Manager に関する情報を表示します。

ステップ 2	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 3	**bloggerd log-dump sap** *number* 例： `switch(config)# bloggerd log-dump sap 351`	ACL Manager サービスのログ ファイル保持機能をイネーブルにします。
ステップ 4	**show system internal bloggerd info log-dump-info** 例： `switch(config)# show system internal bloggerd info log-dump-info`	スイッチ上のログ ファイル保持機能に関する情報を表示します。

例

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
UUID = 0x182, PID = 653, SAP = 351
State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
Restart count: 1
Time of last restart: Mon Nov  4 11:10:39 2019.
The service never crashed since the last reboot.
Tag = N/A
Plugin ID: 0
switch(config)# configure terminal
switch(config)# bloggerd log-dump sap 351
Sending Enable Request to Bloggerd
Bloggerd Log Dump Successfully enabled
switch(config)# show system internal bloggerd info log-dump-info
-------------------------------------------------------------
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
-------------------------------------------------------------
Module     | VDC        | SAP                        | Enabled?
-------------------------------------------------------------
1          | 1          | 351 (MTS_SAP_ACLMGR     ) | Enabled
-------------------------------------------------------------
-------------------------------------------------------------
Log Dump Throttle Switch-Wide Config:
```

```
-------------------------------------
Log Dump Throttle                                    : ENABLED
Minimum buffer rollover count (before throttling)    : 5
Maximum allowed rollover count per minute            : 1
-------------------------------------------------------------

switch(config)#
```

拡張ログ ファイルの表示

スイッチに現在保存されているイベント ログ ファイルを表示するには、次の作業を実行します。

手順の概要

1. **dir debug:log-dump/**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**dir debug:log-dump/** 例： `switch# dir debug:log-dump/`	スイッチに現在保存されているイベント ログ ファイルを表示します。

例

```
switch# dir debug:log-dump/

3676160 Dec 05 02:43:01 2019 20191205023755_evtlog_archive.tar
3553280 Dec 05 06:05:06 2019 20191205060005_evtlog_archive.tar

Usage for debug://sup-local
913408 bytes used
4329472 bytes free
5242880 bytes total
```

ログ統計ごとのグローバル ディクショナリの表示

この CLI は、各コンポーネントによって記録されたログ メッセージの統計をカウンタとともに表示し、システムの稼働時間中に記録されたログの繰り返し回数を保存します。

手順の概要

1. **show system internal sdwrap buffers sap <sap-num> dict-stats detailed**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**show system internal sdwrap buffers sap <sap-num> dict-stats detailed** 例： `switch# show system internal sdwrap buffers sap <sap-num> dict-stats detailed`	各コンポーネントのログごとの統計を表示します。

例

```
switch# show system internal sdwrap buffers sap 221 dict-stats detailed

Sap received is: 221

SDWrap Format Strings Dictionary stats for sap MTS_SAP_L2FM (221)

UUID: SRVUUID_LIBSDWRAP, Inst Type: 0

MsgId Frequency Message                                                    ---------
--------- ------------------------------
 4    1 System is not undergoing ISSU
78    1 Vlan %d is part of reserved vlan bmp from sdb                       179   1 Vlan
 %d is not found in L2FM database. Skipping the delete request 306   1 Vlan %d is removed
 from L2FM database and MTM database
416   1 mts_drap_get_my_local_swid_only_msg failed with rc %#x
496   1 Lookup for backplane mac failed for vdc %d with st = %s
598   1 L2FM - Slot %d SwCardId %d Port %d - %d Fp %d Cli %d
```

単一サービスに対する拡張ログファイル保持の無効化

拡張ログファイル保持は、リリース9.3（5）からのスイッチ上のすべてのサービスに対してデフォルトで有効になっています。スイッチで単一またはすべてのサービスに対してログファイル保持機能が有効になっている場合に、特定のサービスを無効にするには、次の手順を実行します。

手順の概要

1. **show system internal sysmgr service name** *service-type*
2. **configure terminal**
3. **no bloggerd log-dump sap** *number*
4. **show system internal bloggerd info log-dump-info**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**show system internal sysmgr service name** *service-type* 例： `switch# show system internal sysmgr service name aclmgr`	サービス SA P番号を含む ACL Manager に関する情報を表示します。

ステップ 2	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 3	**no bloggerd log-dump sap** *number* 例： `switch(config)# no bloggerd log-dump sap 351`	ACL Managerサービスのログ ファイル保持機能を無効にします。
ステップ 4	**show system internal bloggerd info log-dump-info** 例： `switch(config)# show system internal bloggerd info log-dump-info`	スイッチ上のログ ファイル保持機能に関する情報を表示します。

例

次に、「aclmgr」という名前のサービスの拡張ログ ファイル保持を無効にする例を示します。

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
        UUID = 0x182, PID = 653, SAP = 351
        State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
        Restart count: 1
        Time of last restart: Mon Nov  4 11:10:39 2019.
        The service never crashed since the last reboot.
        Tag = N/A
        Plugin ID: 0
switch(config)# configure terminal
switch(config)# no bloggerd log-dump sap 351
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)# show system internal bloggerd info log-dump-info
------------------------------------------------------------
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
------------------------------------------------------------
Module     | VDC        | SAP                        | Enabled?
------------------------------------------------------------
1          | 1          | 351 (MTS_SAP_ACLMGR     ) | Disabled
```

```
------------------------------------------------------------
------------------------------------------------------------
Log Dump Throttle Switch-Wide Config:
-------------------------------------
Log Dump Throttle                                    : ENABLED
Minimum buffer rollover count (before throttling)    : 5
Maximum allowed rollover count per minute            : 1
------------------------------------------------------------

switch(config)#
```

トリガーベースのイベント ログの自動収集

トリガーベースのログ収集機能：

- 問題発生時に関連データを自動的に収集します。
- コントロール プレーンへの影響なし
- カスタマイズ可能な設定ですか：
  - シスコが入力するデフォルト
  - 収集対象は、ネットワーク管理者または Cisco TACによって、選択的に上書きされます。
  - イメージのアップグレード時は新しいトリガーを自動的に更新します。
- ログをスイッチにローカルに保存するか、外部サーバにリモートで保存します。
- 重大度 0、1、および 2 の syslog をサポートします。
- BGP、BFD、OSPF、ISIS などの予期しないプロトコル イベントをサポートします。
- アドホック イベントのカスタム syslog（syslog と接続する自動収集コマンド）

トリガーベースのログ ファイルの自動収集の有効化

ログ ファイルのトリガーベースの自動作成を有効にするには、__syslog_trigger_default システム ポリシーのオーバーライド ポリシーをカスタム YAML ファイルで作成し、情報を収集する特定のログを定義する必要があります。

ログ ファイルの自動収集を有効にするカスタム YAML ファイルの作成の詳細については、自動収集 YAML ファイルの設定 （384 ページ） を参照してください。

ログプロファイル YAML ファイル

Log-Profile YAML ファイルは、コンポーネントのスロットル制限を定義するために使用されます。**log_profile.yaml** ファイルは、スイッチの /bootflash ディレクトリにあります。

Bloggerd は、コンポーネント名とロールオーバー情報を保持し、特定のコンポーネントのグローバル YAML ファイルで定義されている制限に基づいてログ ファイルを保存/保持します。

デフォルトでは、スイッチのスロットル値は5です。**log_profile.yaml** ファイルにエントリを追加すれば、スロットル カウントをオーバーライドできます。

/bootflash/log_profile.yaml ファイルに加えられた変更を反映するには、bloggerd の実行時に次の CLI を実行します。

- `switch# bloggerd reparse log-profile`

ログプロファイル YAML ファイルの例

以下は、イメージの一部としてパッケージ化されたデフォルトの log_profile.yaml ファイルの例です。ファイル内のキー/値の定義を次の表に示します。

```
273:
    entry_1:
          srv_uuid: 273
          instance: 0
          rollovers_allowed: 250
          rotations_allowed: 5
          mod: sup

274:
    entry_1:
          srv_uuid: 274
          instance: 0
          rollovers_allowed: 250
          rotations_allowed: 5
          mod: sup
```

キー：値	説明
273	sdwrap バッファー スロットリングをオーバーライドする必要があるコンポーネントの UUID。
entry_1：	コンポーネントごとにサポートされるエントリは 1 つだけです コンポーネントごとに最大20のエントリを作成できます。各エントリは、**entry_1** ～ **entry_20** として識別されます。
srv_uuid：	各 sdwrap ログ バッファーは、(uuid, instance ID) のタプルで識別されます。
instance:	上記の srv_uuid フィールドに関連する sdwrap ログ バッファ インスタンス ID。「**-1**」は、すべてのインスタンスを意味します。
rollovers_allowed：	1 分あたりに許可されるロールオーバーの数。許容値は **0** ～ **500** です。
回転の許可：	スロットルごとに許容される回転数。
mod：	syslog コンポーネントの名前（`platform` は syslog のファシリティ名）。

自動収集 YAML ファイル

EEM 機能の **action** コマンドで指定される自動収集 YAML ファイルは、さまざまなシステムまたは機能コンポーネントのアクションを定義します。このファイルは、スイッチ ディレクト

リ: /bootflash/scripts にあります。デフォルトの YAML ファイルに加えて、コンポーネント固有の YAML ファイルを作成し、同じディレクトリに配置できます。コンポーネント固有の YAML ファイルの命名規則は **component-name.yaml** です。コンポーネント固有のファイルが同じディレクトリに存在する場合は、**action** コマンドで指定されたファイルよりも優先されます。たとえば、アクションファイル **bootflash/scripts/platform.yaml** がデフォルトのアクションファイル **/bootflash/scripts** とともに **bootflash/scripts/test.yaml** ディレクトリにある場合、**platform.yaml** ファイルで定義された命令がデフォルトの **test.yaml** ファイルに存在するプラットフォーム コンポーネントの手順よりも優先します。

コンポーネントの例としては、ARP、BGP、IS-IS などがあります。すべてのコンポーネント名に精通していない場合は、シスコ カスタマー サポートに連絡して、コンポーネント固有のアクション（およびデフォルトの **test.yaml** ファイル）の YAML ファイルを定義してください。

例：

```
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml $_syslog_msg
```

コンポーネントごとの自動収集の作成または削除

Cisco NX-OS リリース 10.2(2)F 以降、自動収集の採用改善機能により、要件に基づいて単一または一連のコンポーネントの自動収集を制御できます。自動収集 YAML ファイルの作成または削除には、次のコマンドを使用できます。

（注） Cisco NX-OS リリース 10.3(1)F 以降、複数のコンポーネントがデフォルトで有効になり、コンポーネントの YAML ファイルがデフォルトの自動収集フォルダにコピーされます。ただし、このコマンドを使用して、ブロガーの自動収集コンポーネントを無効または有効にすることはできます。

（注） YAML ファイルは編集可能であるため、慎重に処理してください。ファイルが何らかの構文のため破損している場合、tar は生成されません。。

```
switch# bloggerd auto-collect component <component_name> {enable | disable}
```

enable コマンドを使用すると、コンポーネントの YAML ファイルがバックアップ フォルダからデフォルトの自動収集フォルダにコピーされます。バックアップ ステージング フォルダは読み取り専用フォルダであるため、そこにはコンテンツをコピーできないことに注意してください。一方、必要に応じて、デフォルトの自動収集フォルダ（bootflash:scripts フォルダ）にはコンテンツをコピーできます。

disable コマンドを使用すると、コンポーネントの YAML ファイルが、bootflash:scripts フォルダの下のデフォルトの自動収集フォルダから削除されます。

出力例は次のようになります。

```
switch# bloggerd auto-collect component arp enable
Component arp auto-collect successfully enabled.
arp.yaml file copied from /bootflash/scripts/backup-staging to
```

```
/bootflash/scripts/default-autocollect
switch# dir bootflash:scripts/default-autocollect
435 Nov 10 08:43:21 2021 arp.yaml
438 Oct 25 05:55:11 2021 fex.yaml
579 Oct 25 05:55:11 2021 kern.yaml
Usage for bootflash://sup-local
11078049792 bytes used
10653151232 bytes free
21731201024 bytes total
switch# dir bootflash:scripts/backup-staging/
switch# bloggerd auto-collect component ?
  CrdCfg         Auto-collect for CRDCFG
  aclmgr         Auto-collect for ACLMgr
  aclqos         Auto-collect for ACLQOS
  adjmgr         Auto-collect for Adjacency Manager
  arp            Auto-collect for ARP
  bcm_usd        Auto-collect for BCM USD
  bgp            Auto-collect for BGP
  cardclient     Auto-collect for CARD CLIENT
  cdp            Auto-collect for CPD
  cfs            Auto-collect for CFS
  clis           Auto-collect for CLIS
  cts            Auto-collect for CTS
  dhcp_snoop     Auto-collect for DHCP Snoop
  eigrp          Auto-collect for EIGRP
  eltm           Auto-collect for ELTM
  ethport        Auto-collect for Eth Port Manager
  feature-mgr    Auto-collect for Feature Manager
  fex            Auto-collect for Fex (Satellite Manager)
  hmm            Auto-collect for HMM
  hsrp_engine    Auto-collect for HSRP
  icam           Auto-collect for ICAM
  icmpv6         Auto-collect for ICMPv6
  iftmc          Auto-collect for IFTMC
  im             Auto-collect for IM
  ip             Auto-collect for IP
  ipfib          Auto-collect for IPFIB Manager
  ipqosmgr       Auto-collect for QOS Manager
  isis           Auto-collect for ISIS
  jer_usd        Auto-collect for JER USD
  kafka          Auto-collect for KAFKA Manager
  kern           Auto-collect for Kernel
  l2fm           Auto-collect for L2FM
  l2rib          Auto-collect for L2RIB
  l3vm           Auto-collect for L3VM
  lacp           Auto-collect for LACP
  lldp           Auto-collect for LLDP
  m2rib          Auto-collect for M2RIB
  mfdm           Auto-collect for MFDM
  mrib           Auto-collect for MRIB
  nbm            Auto-collect for NBM Daemon
  netstack       Auto-collect for Netstack
  ngoam          Auto-collect for NGOAM
  nve            Auto-collect for NVE
  ospf           Auto-collect for Open Shortest Path First Unicast Routing Protocol
(OSPF)
  ospfv3         Auto-collect for Open Shortest Path First Version 3 Unicast Routing
Protocol
  pfma           Auto-collect for PFM
  pim            Auto-collect for PIM
  pktmgr         Auto-collect for Packet Manager
  pltfm_config   Auto-collect for PLTFM CONFIG
  port-channel   Auto-collect for Port Channel Manager
  qos            Auto-collect for QOS Manager
```

```
  rip             Auto-collect for RIP
  sdaa            Auto-collect for SDAA
  sla_responder   Auto-collect for SLA Responder
  sla_sender      Auto-collect for SLA Sender
  sla_twamp       Auto-collect for SLA Twamp
  smm             Auto-collect for SMM
  snmpmib_proc    Auto-collect for Snmpmib_proc
  spm             Auto-collect for SPM
  statsclient     Auto-collect for Statistics Client
  sysmgr          Auto-collect for SYSMGR
  tahusd          Auto-collect for TAHUSD
  tctrl_usd       Auto-collect for TCTRL USD
  tun_enc_mgr     Auto-collect for TEM
  udld            Auto-collect for UDLD
  ufdm            Auto-collect for UFDM
  vmtracker       Auto-collect for VMTRACKER
  vntag_mgr       Auto-collect for VNTAG Mgr
  vpc             Auto-collect for VPC
  vrrp-cfg        Auto-collect for VRRP Configuration
  vrrp-eng        Auto-collect for VRRP Engine
  vrrpv3          Auto-collect for VRRPV3
Usage for bootflash://sup-local 11078049792 bytes used
10653151232 bytes free
21731201024 bytes total
switch# dir bootflash:scripts/default-autocollect^C n9k-A# dir
bootflash:scripts/default-autocollect
435 Nov 10 08:43:21 2021 arp.yaml
438 Oct 25 05:55:11 2021 fex.yaml
579 Oct 25 05:55:11 2021 kern.yaml Usage for bootflash://sup-local 11078049792 bytes
used
10653151232 bytes free
21731201024 bytes total
```

以下は、UDLD コンポーネント用に事前入力された YAML ファイルを作成する例です。

```
n9k-A# bloggerd auto-collect component udld enable
Component udld auto-collect successfully enabled.
udld.yaml file copied from /bootflash/scripts/backup-staging to
/bootflash/scripts/default-autocollect
n9k-A# dir bootflash:scripts/default-autocollect
435 Nov 10 08:43:21 2021 arp.yaml
438 Oct 25 05:55:11 2021 fex.yaml
579 Oct 25 05:55:11 2021 kern.yaml
431 Nov 10 08:44:45 2021 udld.yaml
Usage for bootflash://sup-local
11078053888 bytes used
10653147136 bytes free
21731201024 bytes total
n9k-A# sh running-config all | include bloggerd
bloggerd log-dump all
bloggerd log-throttle
no bloggerd log-transfer
```

以下は、UDLD コンポーネント用に事前入力された YAML ファイルを削除する例です。

```
n9k-A# bloggerd auto-collect component udld disable
Component udld auto-collect successfully disabled.
udld.yaml file deleted from /bootflash/scripts/default-autocollect
n9k-A# dir bootflash:scripts/default-autocollect
435 Nov 10 08:43:21 2021 arp.yaml
438 Oct 25 05:55:11 2021 fex.yaml
579 Oct 25 05:55:11 2021 kern.yaml
Usage for bootflash://sup-local
11078049792 bytes used
10653151232 bytes free
```

```
21731201024 bytes total
n9k-A#
```

プロトコル フラップの自動収集の作成

Cisco NX-OS リリース 10.4(1)F 以降では、予期しないプロトコル フラップ イベントが発生した場合、BGP、OSPF、ISIS、および BFD プロトコルの自動データ収集がデフォルトで有効になります。

既存の自動収集は、重大度 0、1、および 2 の syslog メッセージに対して機能します。予期しないプロトコルフラップ自動収集は、ルーティングプロトコルの予期しないイベントが特定のプロトコル プロセスによって検出されるとトリガされます。この自動収集は、制御システム EEM ポリシー `trigger_generic_evt_default` です。

次に、BGP コンポーネントでデフォルトの自動収集を有効にする例を示します。

```
switch# configure terminal
switch(config)# bloggerd auto-collect component bgp enable
component bgp auto-collect successfully enabled
bgp.yaml file copied from /bootflash/scripts/backup-staging to
/bootflash/scripts/default-autocollect
switch(config)# run bash
bash-4.4$ cd /bootflash/scripts/default-autocollect/
bash-4.4$ ls bgp.yaml
```

次に、スイッチでの自動収集履歴の例を示します。

```
switch(config)# show system internal event-logs auto-collect history
DateTime    SnapshotID  Syslog      Status/Secs/Logsize(Bytes)
2023-Mar-21 11:04:03 1395380375    NVE-0- TEST_SYSLOGPROCESSED:23:5756541
```

自動収集 YAML ファイルの設定

YAMLファイルの内容によって、トリガーベースの自動収集時に収集されるデータが決まります。スイッチには YAML ファイルが 1 つだけ存在しますが、任意の数のスイッチ コンポーネントとメッセージの自動収集メタデータを含めることができます。

スイッチの次のディレクトリで YAML ファイルを見つけます。

```
/bootflash/scripts
```

次の例を使用して、トリガーベース収集のYAMLファイルを呼び出します。この例は、ユーザ定義の YAML ファイルを使用してトリガーベース収集を実行するために最低限必要な設定を示しています。

```
switch# show running-config eem
!Command: show running-config eem
!Running configuration last done at: Mon Sep 30 19:34:54 2019
!Time: Mon Sep 30 22:24:55 2019
version 9.3(3) Bios:version 07.59
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml $_syslog_msg
```

上記の例では、「`test_1`」がアプレットの名前で、「`test.yaml`」が /bootflash/scripts ディレクトリにあるユーザ設定の YAML ファイルの名前です。

YAML ファイルの例

次に、トリガーベースのイベント ログ自動収集機能をサポートする基本的な YAML ファイルの例を示します。ファイル内のキー/値の定義を次の表に示します。

（注） YAML ファイルに適切なインデントがあることを確認します。ベスト プラクティスとして、スイッチで使用する前に任意の「オンライン YAML 検証」を実行します。

```
bash-4.3$ cat /bootflash/scripts/test.yaml
version: 1
components:
   securityd:
        default:
            tech-sup: port
            commands: show module
   platform:
        default:
            tech-sup: port
            commands: show module
```

キー：値	説明
バージョン：1	1 に設定します。他の番号を使用すると、自動収集スクリプトに互換性がなくなります。
コンポーネント：	以下がスイッチ コンポーネントであることを指定するキーワード。
securityd：	syslog コンポーネントの名前（`securityd` は syslog のファシリティ名）。
デフォルト：	コンポーネントに属するすべてのメッセージを識別します。
tech-sup：port	`securityd` syslog コンポーネントのポート モジュールのテクニカル サポートを収集します。
コマンド：show module	`securityd` syslog コンポーネントの show module コマンド出力を収集します。
プラットフォーム：	syslog コンポーネントの名前（`platform` は syslog のファシリティ名）。
tech-sup：port	`platform` syslog コンポーネントのポート モジュールのテクニカル サポートを収集します。
コマンド：show module	`platform` syslog コンポーネントの show module コマンド出力を収集します。

特定のログにのみ自動収集メタデータを関連付けるには、次の例を使用します。たとえば、SECURITYD-2-FEATURE_ENABLE_DISABLE

```
securityd:
                feature_enable_disable:
                        tech-sup: security
```

```
commands: show module
```

キー：値	説明
securityd：	syslog コンポーネントの名前（securityd は syslog のファシリティ名）。
feature_enable_disable：	syslog メッセージのメッセージ ID。
tech-sup：security	securityd syslog コンポーネントのセキュリティモ ジュールのテクニカル サポートを収集します。
コマンド：show module	セキュリティ syslog コンポーネントの show module コマンド出力を収集します。

上記の YAML エントリの syslog 出力の例：

```
2019 Dec 4 12:41:01 n9k-c93108tc-fx %SECURITYD-2-FEATURE_ENABLE_DISABLE: User
has enabled the feature bash-shell
```

複数の値を指定するには、次の例を使用します。

```
version: 1
components:
   securityd:
        default:
            commands: show module;show version;show module
            tech-sup: port;lldp
```

（注） 複数の show コマンドとテクニカル サポート キーの値を区切るには、セミコロンを使用します（前の例を参照）。

リリース 10.1(1) 以降では、test.yaml は複数の YAML ファイルが存在するフォルダに置き換えることができます。フォルダ内のすべての YAML ファイルは、ComponentName.yaml 命名規則に従う必要があります。

次の例では、test.yaml が test_folder に置き換えられます。

```
test.yaml:
event manager applet logging2 override __syslog_trigger_default
   action 1.0 collect test.yaml rate-limt 30 $_syslog_msg

test_folder:
event manager applet logging2 override __syslog_trigger_default
   action 1.0 collect test_folder rate-limt 30 $_syslog_msg
```

次の例は、test_folder のパスとコンポーネントを示しています。

```
ls /bootflash/scripts/test_folder
bgp.yaml ppm.yaml
```

コンポーネントあたりの自動収集の量の制限

自動収集の場合、コンポーネントイベントあたりのバンドル数の制限は、Cisco NX-OS リリース 10.2(2)F 以降では、デフォルトで 1 に設定されています。それより前は、この制限はデフォルトで 3 です。1 つのコンポーネントでデフォルトよりも多くのイベントが発生すると、イベントはドロップされ、ステータス メッセージ **EVENTLOGLIMITREACHED** が表示されます。イベント ログがロールオーバーすると、コンポーネント イベントの自動収集が再開されます。

例：

```
switch# show system internal event-logs auto-collect history
DateTime              Snapshot ID  Syslog                  Status/Secs/Logsize(Bytes)
2020-Jun-27 07:20:03  1140276903   ACLMGR-0-TEST_SYSLOG    EVENTLOGLIMITREACHED
2020-Jun-27 07:15:14  1026359228   ACLMGR-0-TEST_SYSLOG    RATELIMITED
2020-Jun-27 07:15:09  384952880    ACLMGR-0-TEST_SYSLOG    RATELIMITED
2020-Jun-27 07:13:55  1679333688   ACLMGR-0-TEST_SYSLOG    PROCESSED:2:9332278
2020-Jun-27 07:13:52  1679333688   ACLMGR-0-TEST_SYSLOG    PROCESSING
2020-Jun-27 07:12:55  502545693    ACLMGR-0-TEST_SYSLOG    RATELIMITED
2020-Jun-27 07:12:25  1718497217   ACLMGR-0-TEST_SYSLOG    RATELIMITED
2020-Jun-27 07:08:25  1432687513   ACLMGR-0-TEST_SYSLOG    PROCESSED:2:10453823
2020-Jun-27 07:08:22  1432687513   ACLMGR-0-TEST_SYSLOG    PROCESSING
2020-Jun-27 07:06:16  90042807     ACLMGR-0-TEST_SYSLOG    RATELIMITED
2020-Jun-27 07:03:26  1737578642   ACLMGR-0-TEST_SYSLOG    RATELIMITED
2020-Jun-27 07:02:56  40101277     ACLMGR-0-TEST_SYSLOG    PROCESSED:3:10542045
2020-Jun-27 07:02:52  40101277     ACLMGR-0-TEST_SYSLOG    PROCESSING
```

自動収集ログ ファイル

自動収集ログ ファイルについて

YAML ファイルの設定によって、自動収集ログ ファイルの内容が決まります。収集ログ ファイルで使用されるメモリの量は設定できません。保存後のファイルが消去される頻度は設定できます。

自動収集ログ ファイルは、次のディレクトリに保存されます。

```
switch# dir bootflash:eem_snapshots
   44205843    Sep 25 11:08:04 2019
1480625546_SECURITYD_2_FEATURE_ENABLE_DISABLE_eem_snapshot.tar.gz
  Usage for bootflash://sup-local
 6940545024 bytes used
44829761536 bytes free
51770306560 bytes total
```

ログ ファイルへのアクセス

コマンド キーワード「debug」を使用してログを検索します。

```
switch# dir debug:///
...
       26    Oct 22 10:46:31 2019  log-dump
       24    Oct 22 10:46:31 2019  log-snapshot-auto
       26    Oct 22 10:46:31 2019  log-snapshot-user
```

次の表に、ログの場所と保存されるログの種類を示します。

場所	説明
log-dump	このフォルダには、ログ ロールオーバー時にイベント ログが保存されます。
log-snapshot-auto	このフォルダには、syslogイベント0、1、2の自動収集ログが含まれます。
log-snapshot-user	このフォルダには、`bloggerd log-snapshot` の実行時に収集されたログが保存されます。

ログ ロールオーバーで生成されたログ ファイルを表示するには、次の例を参考にしてください。

```
switch# dir debug:log-dump/
debug:log-dump/20191022104656_evtlog_archive.tar
debug:log-dump/20191022111241_evtlog_archive.tar
debug:log-dump/20191022111841_evtlog_archive.tar
debug:log-dump/20191022112431_evtlog_archive.tar
debug:log-dump/20191022113042_evtlog_archive.tar
debug:log-dump/20191022113603_evtlog_archive.tar
```

ログ tar ファイルの解析

tar ファイル内のログを解析するには、次の例を参考にしてください。

```
switch# show system internal event-logs parse
debug:log-dump/20191022104656_evtlog_archive.tar
--------LOGS:/tmp/BLOGGERD0.991453012199/tmp/1-191022104658-191022110741-device_test-M27-V1-I1:0-P884.gz--------
2019 Oct 22 11:07:41.597864 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Data Space
Limits(bytes): Soft: -1  Ha rd: -1
2019 Oct 22 11:07:41.597857 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Stack Space
Limits(bytes): Soft: 500000  Hard: 500000
2019 Oct 22 11:07:41.597850 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):AS: 1005952076
 -1
2019 Oct 22 11:07:41.597406 E_DEBUG Oct 22 11:07:41 2019(device_test_process_events):Sdwrap
 msg unknown
2019 Oct 22 11:07:41.597398 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Going back to
 select
2019 Oct 22 11:07:41.597395 E_DEBUG Oct 22 11:07:41 2019(nvram_test):TestNvram examine
27 blocks
2019 Oct 22 11:07:41.597371 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Parent: Thread
 created test index:4 thread_id:-707265728
2019 Oct 22 11:07:41.597333 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Node inserted
2019 Oct 22 11:07:41.597328 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):The test index
 in diag is 4
2019 Oct 22 11:07:41.597322 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):result severity
 level
2019 Oct 22 11:07:41.597316 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):callhome alert
 level
```

次の表に、特定の tar ファイルの解析に使用できる追加のキーワードを示します。

キーワード	説明
component	プロセス名で識別されるコンポーネントに属するログをデコードします。
from-datetime	yy [mm [dd [HH [MM [SS]]]]] 形式で指定した、特定の日時のログをデコードします。

キーワード	説明
instance	デコードする SDWRAP バッファ インスタンスのリスト（カンマ区切り）。
module	SUP や LC などのモジュールからのログをデコードします（モジュール ID を使用）。
to-datetime	yy [mm [dd [HH [MM [SS]]]]] 形式で指定した、特定の日時までのログをデコードします。

別の場所へログをコピーする

リモート サーバなどの別の場所にログをコピーするには、次の例を参考にしてください。

```
switch# copy debug:log-dump/20191022104656_evtlog_archive.tar
scp://<ip-adress>/nobackup/<user> vrf management  use-kstack
Enter username: user@<ip-address>'s password:
20191022104656_evtlog_archive.tar                                          100%  130KB
 130.0KB/s   00:00
Copy complete, now saving to disk (please wait)...
Copy complete.
```

自動収集ログファイルの消去

生成されるトリガー ベースの自動収集ログには、EventHistory と EventBundle の 2 種類があります。

EventHistory ログの消去ロジック

イベント履歴の場合は、/var/sysmgr/srv_logs/xport フォルダで消去が行われます。250 MB のパーティション RAM が、/var/sysmgr/srv_logs ディレクトリにマウントされます。

/var/sysmgr/srv_logs のメモリ使用率が、割り当てられた 250 MB の 65% 未満の場合、ファイルは消去されません。メモリ使用率が 65% の制限レベルに達すると、新しいログの保存を続行するのに十分なメモリが使用可能になるまで、最も古いファイルから消去されます。

EventBundle ログの消去ロジック

イベント バンドルの場合、消去ロジックは /bootflash/eem_snapshots フォルダでの状態に基づいて実行されます。自動収集されたスナップショットを保存するために、EEM 自動収集スクリプトは、ブートフラッシュ ストレージの 5% を割り当てます。ブート フラッシュ容量の 5% が使用されると、ログは消去されます。

新しい自動収集ログが利用可能になっているものの、ブート フラッシュに保存するスペースがない場合（すでに 5% の容量に達している）、システムは次のことを確認します。

1. 12 時間以上経過した既存の自動収集ファイルがある場合、システムはファイルを削除し、新しいログをコピーします。
2. 既存の自動収集ファイルが 12 時間未満の場合、新しく収集されたログは保存されずに廃棄されます。

デフォルトパージ時間である 12 時間は、次のコマンドを使用して変更できます。コマンドで指定する時間は分単位です。

```
switch(config)# event manager applet test override __syslog_trigger_default
switch(config-applet)# action 1.0 collect test.yaml purge-time 300 $_syslog_msg
```

event manager command: *test* は、ポリシー例の名前です。**__syslog_trigger_default** は、オーバーライドする必要のあるシステムポリシーの名前です。この名前は、二重アンダースコア（__）で始まる必要があります。

action command: **1.0** は、アクションの実行順番を示している例となっています。**collect** は、データが YAMU ファイルを使用して収集されることを示しています。*test.yaml* は、YAML ファイルの名前の例です。**$_syslog_msg** は、コンポーネントの名前です。

（注） どの時点でも、進行中のトリガーベースの自動収集イベントは1つだけです。自動収集がすでに発生しているときに別の新しいログ イベントを保存しようとすると、新しいログ イベントは破棄されます。

デフォルトでは、トリガーベースのバンドルは 5 分（300 秒）ごとに 1 つだけ収集されます。このレート制限は、次のコマンドでも設定できます。コマンドで指定する時間は秒単位です。

```
switch(config)# event manager applet test override __syslog_trigger_default
switch(config-applet)# action 1.0 collect test.yaml rate-limit 600 $_syslog_msg
```

event manager command: *test* はポリシーの名前の例です。**__syslog_trigger_default** は、オーバーライドするシステムポリシーの名前の例です。この名前は、二重アンダースコア（__）で始まる必要があります。

action command: **1.0** は、アクションの実行順番を示している例となっています。**collect** は、データが YAMU ファイルを使用して収集されることを示しています。*test.yaml* は、YAML ファイルの名前の例です。**$_syslog_msg** は、コンポーネントの名前です。

リリース 10.1(1) 以降では、トリガーの最大数オプションを使用して収集レートを調整することもできます。これは、この数のトリガーだけを保つものです。 **max-triggers** の値に達すると、syslog が発生しても、これ以上バンドルは収集されなくなります。

```
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml rate-limt 30 max-triggers 5 $_syslog_msg
```

（注） 自動収集されたバンドルを `debug:log-snapshot-auto/` により手動で削除すれば、次のイベントが発生したとき、 **max-triggers** の設定数に基づいて収集が再開されます。

自動収集の統計情報と履歴

トリガーベースの収集統計情報の例を次に示します。

```
switch# show system internal event-logs auto-collect statistics
---------------------EEM Auto Collection Statistics--------------------
Syslog Parse Successful :88 Syslog Parse Failure :0
```

```
Syslog Ratelimited :0 Rate Limit Check Failed :0
Syslog Dropped(Last Action In Prog) :53 Storage Limit Reached :0
User Yaml Action File Unavailable :0 User Yaml Parse Successful :35
User Yaml Parse Error :0 Sys Yaml Action File Unavailable :11
Sys Yaml Parse Successful :3 Sys Yaml Parse Error :0
Yaml Action Not Defined :0 Syslog Processing Initiated :24
Log Collection Failed :0 Tar Creation Error :0
Signal Interrupt :0 Script Exception :0
Syslog Processed Successfully :24 Logfiles Purged :0
```

次の例は、CLI コマンドを使用して取得されたトリガーベースの収集履歴（処理された syslog 数、処理時間、収集されたデータのサイズ）を示しています。

```
switch# show system internal event-logs auto-collect history
DateTime Snapshot ID Syslog Status/Secs/Logsize(Bytes)
2019-Dec-04 05:30:32 1310232084 VPC-0-TEST_SYSLOG PROCESSED:9:22312929
2019-Dec-04 05:30:22 1310232084 VPC-0-TEST_SYSLOG PROCESSING
2019-Dec-04 04:30:13 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSED:173:33194665
2019-Dec-04 04:28:47 897805674 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:28:47 947981421 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:27:19 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSING
2019-Dec-04 02:17:16 1957148102 CARDCLIENT-2-FPGA_BOOT_GOLDEN NOYAMLFILEFOUND
```

サポートされているログ収集の例

いくつかのコンポーネントでサポートされるログのサンプル コレクションを次に示します。

コンポーネント：**IPQoSMgr**

サポートされるログ：

```
QOSMGR_MTS_FAILURE
QOSMGR_NETWORK_QOS_POLICY_CHANGE
QOSMGR_LLFC_APPLY_FAILURE
QOSMGR_FCOE_POLICY_NOT_REMOVED
```

コンポーネント：**ACLQOS**

サポートされるログ：

```
ACLQOS_UNEXPECTED_MCAST_FRAMES
ACLQOS_UNEXPECTED_PFC_FRAMES
PPF_SUBSCRIPTION_FAILED
ACLQOS_QOS_NO_DROP_CLASSIFICATION_UNSUPPORTED
ACLQOS_QUEUE_LIMIT_IGNORED_ON_FEX
ACLQOS_BUFFER_DRAIN_FAILURE
ACLQOS_BURST_DETECT_FPGA_INCOMPATIBLE
ACLQOS_BURST_DETECT_OVER_THRESHOLD
ACLQOS_FAILED
PPF_FAILED
```

トリガーベースのログ収集の確認

次の例のように **show event manager system-policy | i trigger** コマンドを入力して、トリガーベースのログ収集機能が有効になっていることを確認します。

```
switch# show event manager system-policy | i trigger n 2
          Name : __syslog_trigger_default
    Description : Default policy for trigger based logging
    Overridable : Yes
     Event type : 0x2101
```

トリガーベースのログ ファイル生成の確認

トリガーベースの自動収集機能によってイベント ログ ファイルが生成されたかどうかを確認できます。次の例のいずれかのコマンドを入力します。

```
switch# dir bootflash:eem_snapshots
9162547 Nov 12 22:33:15 2019
1006309316_SECURITYD_2_FEATURE_ENABLE_DISABLE_eem_snapshot.tar.gz

Usage for bootflash://sup-local
8911929344 bytes used
3555950592 bytes free
12467879936 bytes total

switch# dir debug:log-snapshot-auto/
63435992 Dec 03 06:28:52 2019
20191203062841_1394408030_PLATFORM_2_MOD_PWRDN_eem_snapshot.tar.gz

Usage for debug://sup-local
544768 bytes used
4698112 bytes free
5242880 bytes total
```

ローカル ログ ファイルのストレージ

ローカル ログ ファイルのストレージ機能：

- ローカルデータ ストレージ時間の量は、導入の規模とタイプによって異なります。モジュラ スイッチと非モジュラ スイッチの両方で、ストレージ時間は 15 分から数時間のデータです。長期間にわたる関連ログを収集するには、次の手順を実行します。
  - 必要な特定のサービス/機能に対してのみイベント ログの保持を有効にします。「単一サービスの拡張ログファイル保持の有効化 （374 ページ）」を参照してください。
  - スイッチから内部イベント ログをエクスポートします。「外部ログ ファイルのストレージ （395 ページ）」を参照してください。
- 圧縮されたログは RAM に保存されます。
- 250MB のメモリは、ログ ファイル ストレージ用に予約されています。
- ログ ファイルは tar 形式で最適化されます（5 分ごとに 1 ファイルまたは 10 MB のいずれか早い方）。
- スナップ ショット収集を許可します。

最近のログ ファイルのローカル コピーの生成

拡張ログ ファイル保持は、スイッチで実行されているすべてのサービスに対してデフォルトで有効になっています。ログ ファイルは、フラッシュ メモリにローカルに保存されます。次の手順を使用して、最新のイベント ログ ファイルを最大 10 個生成します。

手順の概要

1. **bloggerd log-snapshot** [*file-name*] [**bootflash:** *file-path* | **logflash:** *file-path* | **usb1:**] [**size** *file-size*] [**time** *minutes*]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**bloggerd log-snapshot** [*file-name*] [**bootflash:** *file-path* | **logflash:** *file-path* | **usb1:**] [**size** *file-size*] [**time** *minutes*] 例： `switch# bloggerd log-snapshot snapshot1`	スイッチに保存されている最新の 10 個のイベント ログのスナップショット バンドル ファイルを作成します。この操作のデフォルトのストレージは **logflash** です。 *file-name*：生成されたスナップショット ログ ファイル バンドルのファイル名。*file-name* には最大 64 文字を使用します。 （注） この変数はオプションです。設定されていない場合、システムはタイムスタンプと「_snapshot_bundle.tar」をファイル名として適用します。例： `20200605161704_snapshot_bundle.tar` **bootflash:** *file-path*：スナップショット ログ ファイル バンドルがブート フラッシュに保存されているファイル パス。次の初期パスのいずれかを選択します。 • bootflash:/// • bootflash://module-1/ • bootflash://sup-1/ • bootflash://sup-active/ • bootflash://sup-local/ **logflash:** *file-path*：スナップショット ログ ファイル バンドルがログ フラッシュに保存されるファイル パス。次の初期パスのいずれかを選択します。 • logflash:/// • logflash://module-1/ • logflash://sup-1/ • logflash://sup-active/

	コマンドまたはアクション	目的
		• logflash://sup-local/ **usb1:**：USB デバイス上のスナップショット ログ ファイル バンドルが保存されているファイル パス。 **size** *file-size*：メガバイト（MB）単位のサイズに基づくスナップショット ログ ファイル バンドル。範囲は 5MB ー 250MB です。 **time** *minutes*：最後の x 時間（分）に基づくスナップショット ログ ファイル バンドル。範囲は 1 ～ 30 分です。

例

```
switch# bloggerd log-snapshot snapshot1
Snapshot generated at logflash:evt_log_snapshot/snapshot1_snapshot_bundle.tar Please
cleanup once done.
switch#
switch# dir logflash:evt_log_snapshot
159098880 Dec 05 06:40:24 2019 snapshot1_snapshot_bundle.tar
159354880 Dec 05 06:40:40 2019 snapshot2_snapshot_bundle.tar

Usage for logflash://sup-local
759865344 bytes used
5697142784 bytes free
6457008128 bytes total
```

次の例のコマンドを使用して、同じファイルを表示します。

```
switch# dir debug:log-snapshot-user/
159098880 Dec 05 06:40:24 2019 snapshot1_snapshot_bundle.tar
159354880 Dec 05 06:40:40 2019 snapshot2_snapshot_bundle.tar

Usage for debug://sup-local
929792 bytes used
4313088 bytes free
5242880 bytes total
```

（注） 例の最後のファイル名に注意してください。個々のログ ファイルは、生成された日時によっても識別されます。

リリース 10.1(1) 以降、LC コアファイルには `log-snapshot` バンドルが含まれています。`log-snapshot` バンドル ファイル名は、`tac_snapshot_bundle.tar.gz` です。次に例を示します。

```
bash-4.2$ tar -tvf 1610003655_0x102_aclqos_log.17194.tar.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 pss/
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_runtime_info_lc.gz
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_runtime_cfg_lc.gz
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_debug.gz
```

```
-rw-rw-rw- root/root 129583 2021-01-07 12:44 pss/clqosdb_ver1_0_user.gz
-rw-rw-rw- root/root 20291 2021-01-07 12:44 pss/clqosdb_ver1_0_node.gz
-rw-rw-rw- root/root 444 2021-01-07 12:44 pss/clqosdb_ver1_0_ctrl.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 proc/
-rw-rw-rw- root/root 15159 2021-01-07 12:44 0x102_aclqos_compress.17194.log.25162
-rw-rw-rw- root/root 9172392 2021-01-07 12:43 0x102_aclqos_core.17194.gz
-rw-rw-rw- root/root 43878 2021-01-07 12:44 0x102_aclqos_df_dmesg.17194.log.gz
-rw-rw-rw- root/root 93 2021-01-07 12:44 0x102_aclqos_log.17194
-rw-rw-rw- root/root 158 2021-01-07 12:44 0x102_aclqos_mcore.17194.log.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 usd17194/
-rw-rw-rw- root/root 11374171 2021-01-07 12:44 tac_snapshot_bundle.tar.gz
```

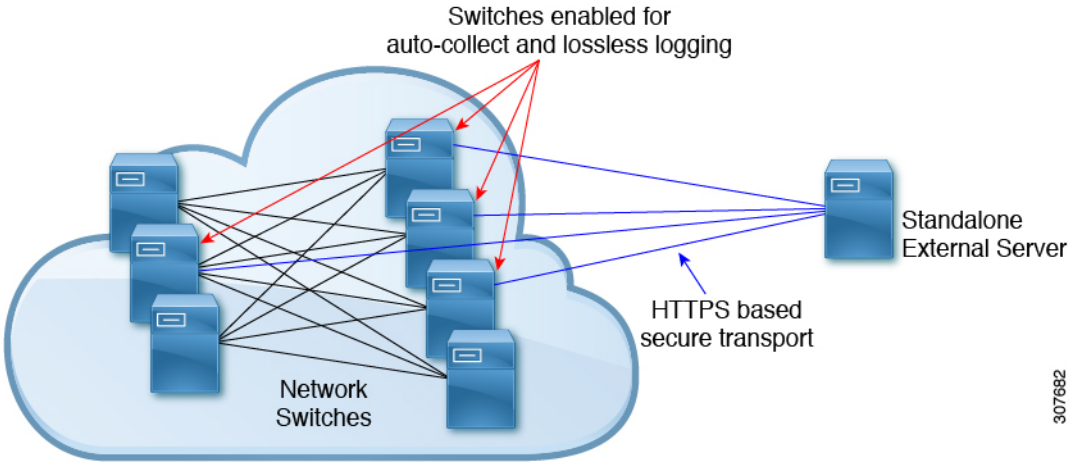
外部ログ ファイルのストレージ

外部サーバ ソリューションは、ログを安全な方法でオフスイッチに保存する機能を提供します。

（注） 外部ストレージ機能を作成するため、Cisco Technical Assistance Center（TAC）に連絡して、外部サーバ ソリューションの展開をサポートを求めてください。

次に、外部ログ ファイルの保存機能を示します。

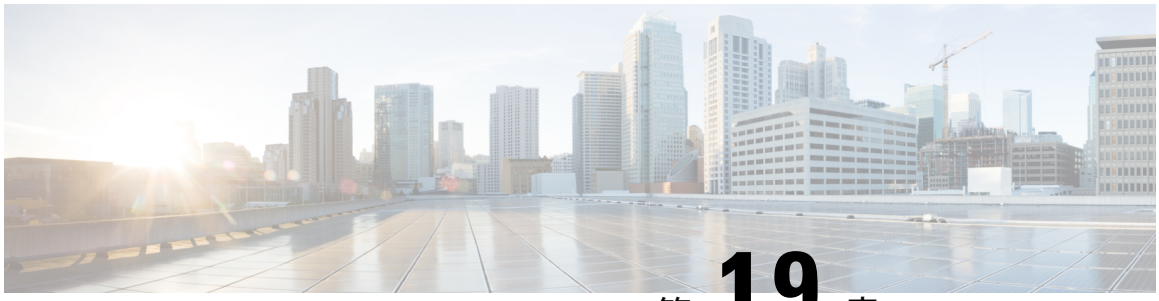
- オンデマンドで有効
- HTTPS ベースの転送
- ストレージ要件：
  - 非モジュラ スイッチ：300 MB
  - モジュラ スイッチ：12 GB（1 日あたり、スイッチあたり）
- 通常、外部サーバには 10 台のスイッチのログが保存されます。ただし、外部サーバでサポートされるスイッチの数に厳密な制限はありません。



外部サーバ ソリューションには、次の特性があります。

- コントローラレス環境
- セキュリティ証明書の手動管理
- サポートされている 3 つの使用例：
  - 選択したスイッチからのログの継続的な収集
  - TAC のサポートによる、シスコ サーバへのログの展開とアップロード。
  - 限定的なオンプレミス処理

（注） 外部サーバでのログ ファイルの設定と収集については、Cisco TAC にお問い合わせください。



第 19 章

MAC 移動ポリシーの構成

この章では、Cisco NX-OS デバイス上でMAC 移動ポリシーを構成する方法について説明します。

この章は、次の項で構成されています。

- MAC 移動ポリシーについて （397 ページ）
- MAC 移動ポリシーの注意事項と制約事項 （397 ページ）
- MAC 移動ポリシーの構成 （398 ページ）
- MAC 移動ポリシーの構成の確認 （399 ページ）

MAC 移動ポリシーについて

MAC 移動ポリシー機能は、ファブリック全体の MAC アドレスの移動をモニタするために使用されます。この機能は、ネットワークの安定性を強化し、レイヤ2フォワーダマニピュレータ（L2FM）でのメッセージングおよびトランザクションサービス（MTS）の蓄積を防ぐために、MAC 移動パラメータを変更するオプションを提供します。

デフォルトでは、MAC移動ポリシーは無効になっています。MAC移動ポリシーの構成とMTSビルドアップチェックの有効化の詳細については、MAC 移動ポリシーの構成 （398 ページ）セクションを参照してください。

MAC 移動ポリシーの利点は次のとおりです。

- ループを回避するために、MAC レベルでの MAC アドレス移動の追跡を有効にします。
- L2FM でのクラッシュを防ぐために、MTS の蓄積を定期的に確認します。
- MAC 移動カウントが指定されたしきい値を超えた場合、VLAN ラーニングを無効にします。

MAC 移動ポリシーの注意事項と制約事項

MAC 移動ポリシーの 構成時の注意事項および制約事項は、次のとおりです：

- Cisco NX-OS リリース 10.3(1)F 以降、MAC 移動ポリシーが Cisco Nexus 9300-X Cloud スケール スイッチでサポートされます。
- **mac-move policy** コマンドを使用して MAC 移動ポリシーを有効にすると、デフォルトのパラメータ（120 秒のホールド インターバルで 30 秒間に 6 回の MAC 移動）が設定されます。デフォルトの設定パラメータは変更できます。詳細については、MAC 移動ポリシーの構成 （398 ページ）を参照してください。
- MAC 移動ポリシーと L2RIB 検出は、デフォルト値では共存できません。これらのメカニズムは両方とも重複検出を処理するためのものですが、異なるアプローチを取っています。
- VXLAN 環境で MAC 移動ポリシーが必要ない場合は、有効にしないでください。必要に応じて、L2RIB ポリシーまたは MAC 移動ポリシーのデフォルト値を変更してください。
- L2RIB 検出は、**l2rib dup-host-mac-detection <mac moves threshold> <detect-interval>** コマンドを使用して変更できます。
- MAC 移動ポリシーと L2RIB 検出の両方が構成されている場合、次の動作が観察されるようになります。
  - L2RIB 検出が L2 ポリシーより小さい場合、L2RIB 検出のみがトリガーされます。ただし、L2 ポリシーはトリガされません。
  - L2RIB 検出が L2 ポリシーと等しい場合、L2 ポリシーまたは L2RIB 検出のいずれかがトリガされます。ただし、これらのポリシーがトリガされる順序は保証されません。
  - L2RIB 検出が L2 ポリシーより大きい場合、L2 ポリシーのみがトリガされます。ただし、L2RIB 検出はトリガされません。

MAC 移動ポリシーの構成

この手順では、スイッチの MAC 移動ポリシーを有効または無効にします：

手順の概要

1. **configure terminal**
2. **[no] mac-move policy {[move-threshold**<*thresh*>]} {[**detect-intvl**<*d_intvl*>]} {[**hold-intvl** <*h_intvl*>]}
3. **[no] mts-buildup check** {[**mts-percent**<*percent*>]} {[**detect-intvl**<*d_intvl*>]}

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**[no] mac-move policy {[move-threshold**<*thresh*>]} {[**detect-intvl**<*d_intvl*>]} {[**hold-intvl** <*h_intvl*>]} 例： `switch(config)# mac-move policy move-threshold 6 detect-intvl 30 hold-intvl 120`	MAC 移動ポリシーを有効にします。 オプション **no** は、MAC 移動ポリシーを無効にします。 • *thresh* : 最大許容移動。範囲は 2 ～ 4096 で、デフォルト値は 6 です。 • *d_intvl* : 動きを追跡する間隔。範囲は 30 ～ 3600 で、デフォルト値は 30 です。 *h_intvl* : 学習を無効にする間隔。範囲は 120 ～ 360 で、デフォルト値は 120 です。
ステップ 3	**[no] mts-buildup check** {[**mts-percent**<*percent*>]} {[**detect-intvl**<*d_intvl*>]} 例： `switch(config)# mts-buildup check mts-percent 40 detect-intvl 60`	mts-buildup チェックを有効にします。 オプション **no** は、mts-buildup チェックを無効にします。 • *percent* : MTS ビルドアップのパーセント値。範囲は 10 ～ 50 で、デフォルト値は 40 です。 • *d_intvl* : MTS ビルドアップをチェックする間隔。範囲は 60 ～ 600 で、デフォルト値は 60 です。

MAC 移動ポリシーの構成の確認

MAC 移動ポリシー構成情報を表示するには、次のいずれかの作業を実行します。

コマンド	目的
show mac-move policy	MAC 移動ポリシーに関する情報を表示します。
show mts-buildup check	mts-buildup チェックに関する情報を表示します。

次の例は、**show mac-move policy** コマンドのサンプル出力を示しています。

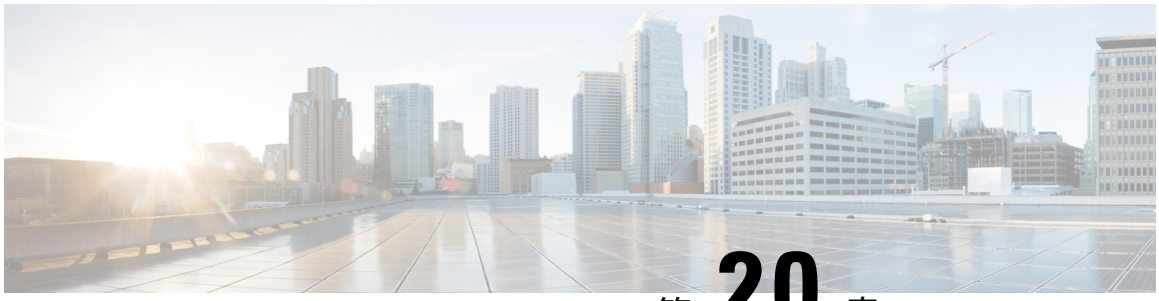
```
switch(config)# show mac-move policy
MAC move policy enabled = TRUE
MAC move policy threshold = 6
MAC move policy detect interval = 30
MAC move policy hold interval =120
switch(config)#
```

次の例は、チェックの有効時の **show mts-buildup check** コマンドのサンプル出力を示しています：

```
switch(config)# show mts-buildup check
MTS buildup check enabled =TRUE
MTS check percent =40
MTS check interval =60
switch(config)#
```

次の例は、チェックの無効時の **show mts-buildup check** コマンドのサンプル出力を示しています：

```
switch(config)# show mts-buildup check
MTS buildup check enabled =FALSE
ppwrks5(config)# show mac-move policy
MAC move policy enabled = FALSE
switch(config)#
```



第 20 章

VSH セッションの端末ロック

• VSH セッションの端末ロック （401 ページ）

VSH セッションの端末ロック

概要

現在 NX-OS では、多くのユーザがスイッチにログインしており、CLI のセッションで設定を変更しています。目標は、このシナリオを制限し、1 人のユーザだけがスイッチを設定できるようにすることです。これは、端末をロックして1人のユーザだけが configure terminal コマンドにアクセスできるようにする端末ロック CLI によって実現されます。その結果、他のユーザが NX-OS の実行コンフィギュレーションを変更できないようにする「コンフィギュレーション ロック」の効果が得られます。

端末ロック機能は、ユーザがNX-OS実行コンフィギュレーションを変更するための排他的コンフィギュレーションアクセスを可能にするロックメカニズムを提供します。

動作のシーケンスは、次のとおりです。

1. `端末ロック`：この CLI はユーザに設定ロックを提供します。
2. `terminal unlock`：この CLI は、任意のセッションで取得された端末ロックを解除します。
3. `show terminal lock`：現在の端末ロックのステータスと詳細を表示します。

端末ロック

端末ロックの使用に関するガイドラインは次のとおりです。

- `端末ロック`では、ロックが保持されている現在のセッションでのみconfigコマンドを実行できます。
- `端末ロック`は、他のセッションの config コマンドのみをブロックします。つまり、SHOW または EXEC CLI は引き続き許可されます。
- 端末ロックのデフォルトのタイムアウトは 1800 秒（30 分）です。

VSH セッションの端末ロック

- ロック タイマーが期限切れになると、端末ロックは自動的に解除されます。
- 端末ロック CLI は、network-admin 権限を持つ任意のユーザが実行できます。
- 「デュアル ステージの構成」セッションが進行中の場合、端末ロックは拒否されます。

次に、端末ロックの CLI の例を示します。

```
switch# terminal lock?
lock Locks the CLI Config mode
switch# terminal lock ?
<CR>
<60-43200> Enter terminal lock timeout in seconds
*Default value is 1800
“terminal lock” locks the parser configuration mode and prints a syslog message as shown
 in below example.
switch# terminal lock
switch# 2021 Jun 19 17:53:37 switch %VSHD-5-VSHD_CLI_TERM_LOCK: terminal lock is taken
by admin on console0
```

（注） ユーザが別のセッションで設定済みの端末を入力しようとすると、次のエラーメッセージが表示されます。端末ロックは他の VSH セッションによって取得されます。」

Cisco NX-OS リリース 10.2(2)F 以降、RESTCONF、NETCONF、gRPC、gNMI などのモデル駆動型プログラマビリティインターフェイスをロックするための新しいCLIオプション、**"terminal lock mdp"**が導入されました。

"terminal lock mdp" CLI は、DME セッションを含むすべての設定セッションにターミナルロックを適用できるようにします。

以下は、**"terminal lock mdp"** CLI のサンプル出力です：

```
switch# terminal lock?
  lock  Locks the CLI Config mode

switch# terminal lock ?
  <CR>
  <mdp>  Locks Model Driven Programmability sessions
  <60-43200>  Enter terminal lock timeout in seconds
              *Default value is 1800

switch# terminal lock mdp
2021 Oct 26 06:33:19 switch %VSHD-5-VSHD_CLI_TERM_LOCK: terminal lock is taken by admin
 on console0
switch#
switch# show terminal lock
PID: 10018
User: admin
Session: console0
State: LOCKED
MDP lock: True
Lock acquired time: Mon Mar  8 09:24:03 2021
Lock Expiration timer (in Sec): 1800
switch#
```

VSH セッションの端末ロック

端末ロック解除

次に、端末ロック解除の CLI の例を示します。

```
switch# terminal unlock?
unlock Force unlocking of the CLI config mode
switch# terminal unlock ?
<CR>
switch# terminal unlock
switch# 2021 Jun 19 17:53:21 switch %VSHD-5-VSHD_CLI_TERM_LOCK: terminal lock is released
 by admin on console0
```

（注） 「端末ロック」は1人の管理者ユーザだけが取得できますが、「端末ロック解除」を使用して管理者ユーザがロックを解除できます。

端末ロックの表示

このコマンドは、所有者、ユーザ、セッション、ロック状態、ロック タイマーなど、現在の設定ロックのステータスと詳細を表示します。

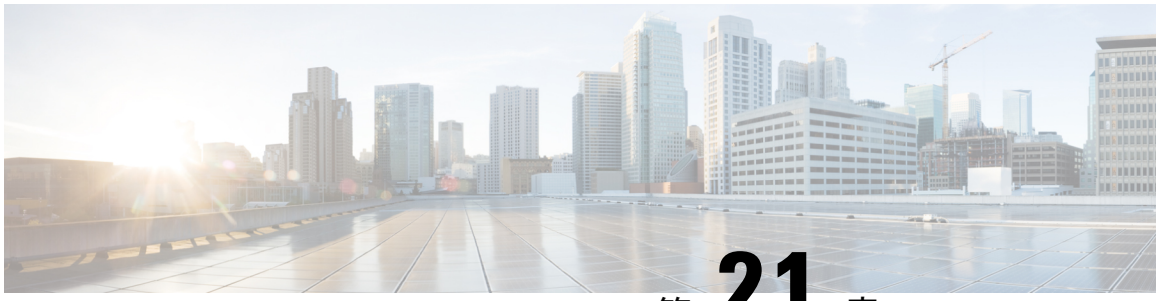
次に、ロックがアクティブな場合の端末ロックの表示の CLI の例を示します。

```
switch# terminal lock
switch#
switch# show terminal lock
PID: 10018
User: admin
Session: console0
State: LOCKED
Lock acquired time: Mon Mar 8 09:24:03 2021
```

次に、ロックが解放されている場合の端末ロックの表示の CLI の例を示します。

```
switch# terminal unlock
switch#
switch#
switch# show terminal lock
PID: -1
User: unknown
Session: NA
State: FREE
Lock acquired time:
Lock Expiration timer (in Sec): 0
switch#
```

VSH セッションの端末ロック



第 21 章

オンボード障害ロギングの設定

この章では、Cisco NX-OS デバイスで Onboard Failure Logging（OBFL）機能を設定する方法について説明します。

この章は、次の項で構成されています。

- OBFL の概要 （405 ページ）
- OBFL の前提条件 （406 ページ）
- OBFL の注意事項と制約事項 （406 ページ）
- OBFL のデフォルト設定 （406 ページ）
- OBFL の設定 （406 ページ）
- OBFL 設定の確認 （409 ページ）
- OBFL のコンフィギュレーション例 （410 ページ）
- その他の参考資料 （411 ページ）

OBFL の概要

Cisco NX-OS には永続ストレージに障害データを記録する機能があるので、あとから記録されたデータを取得して表示し、分析できます。このオンボード障害ロギング（OBFL）機能は、障害および環境情報をモジュールの不揮発性メモリに保管します。この情報は、障害モジュールの分析に役立ちます。

OBFL は次のタイプのデータを保存します。

- 最初の電源投入時刻
- モジュールのシャーシ スロット番号
- モジュールの初期温度
- ファームウェア、BIOS、FPGA、および ASIC のバージョン
- モジュールのシリアル番号
- クラッシュのスタック トレース
- CPU hog 情報

- メモリ リーク情報
- ソフトウェア エラー メッセージ
- ハードウェア例外ログ
- 環境履歴
- OBFL 固有の履歴情報
- ASIC 割り込みおよびエラー統計の履歴
- ASIC レジスタ ダンプ

OBFL の前提条件

network-admin ユーザ権限が必要です。

OBFL の注意事項と制約事項

OBFL に関する注意事項および制約事項は、次のとおりです。

- OBFL はデフォルトでイネーブルになっています。
- OBFL フラッシュがサポートする書き込みおよび消去の回数には制限があります。イネーブルにするロギング数が多いほど、この書き込みおよび消去回数に早く達してしまいます。

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

OBFL のデフォルト設定

次の表に、VACL パラメータのデフォルト設定を示します。

パラメータ	デフォルト
OBFL	すべての機能がイネーブル

OBFL の設定

Cisco NX-OS デバイス上で OBFL 機能を設定できます。

始める前に

グローバル コンフィギュレーション モードになっていることを確認します。

手順の概要

1. **configure terminal**
2. **hw-module logging onboard**
3. **hw-module logging onboard counter-stats**
4. **hw-module logging onboard cpuhog**
5. **hw-module logging onboard environmental-history**
6. **hw-module logging onboard error-stats**
7. **hw-module logging onboard interrupt-stats**
8. **hw-module logging onboard module** *slot*
9. **hw-module logging onboard obfl-logs**
10. （任意） **show logging onboard**
11. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**hw-module logging onboard**
	例：
	`switch(config)# hw-module logging onboard`
	`Module: 7 Enabling ... was successful.`
	`Module: 10 Enabling ... was successful.`
	`Module: 12 Enabling ... was successful.`
		すべての OBFL 機能をイネーブルにします。

ステップ 3	**hw-module logging onboard counter-stats**
	例：
	`switch(config)# hw-module logging onboard counter-stats`
	`Module: 7 Enabling counter-stats ... was successful.`
	`Module: 10 Enabling counter-stats ... was successful.`
	`Module: 12 Enabling counter-stats ... was successful.`
		OBFL カウンタ統計情報を有効にします。

ステップ 4	**hw-module logging onboard cpuhog** 例：	OBFL CPU hog イベントを有効にします。

	コマンドまたはアクション	目的

	`switch(config)# hw-module logging onboard cpuhog`
	`Module: 7 Enabling cpu-hog ... was successful.`
	`Module: 10 Enabling cpu-hog ... was successful.`
	`Module: 12 Enabling cpu-hog ... was successful.`	

ステップ 5	**hw-module logging onboard environmental-history**
	例：
	`switch(config)# hw-module logging onboard environmental-history`
	`Module: 7 Enabling environmental-history ... was successful.`
	`Module: 10 Enabling environmental-history ... was successful.`
	`Module: 12 Enabling environmental-history ... was successful.`
		OBFL 環境履歴をイネーブルにします。

ステップ 6	**hw-module logging onboard error-stats**
	例：
	`switch(config)# hw-module logging onboard error-stats`
	`Module: 7 Enabling error-stats ... was successful.`
	`Module: 10 Enabling error-stats ... was successful.`
	`Module: 12 Enabling error-stats ... was successful.`
		OBFL エラー統計をイネーブルにします。

ステップ 7	**hw-module logging onboard interrupt-stats**
	例：
	`switch(config)# hw-module logging onboard interrupt-stats`
	`Module: 7 Enabling interrupt-stats ... was successful.`
	`Module: 10 Enabling interrupt-stats ... was successful.`
	`Module: 12 Enabling interrupt-stats ... was successful.`
		OBFL 割り込み統計をイネーブルにします。

ステップ 8	**hw-module logging onboard module** *slot*
	例：
	`switch(config)# hw-module logging onboard module 7`
	`Module: 7 Enabling ... was successful.`
		モジュールの OBFL 情報をイネーブルにします。

ステップ 9	**hw-module logging onboard obfl-logs**
	例：
	`switch(config)# hw-module logging onboard obfl-logs`
	`Module: 7 Enabling obfl-log ... was successful.`
	`Module: 10 Enabling obfl-log ... was successful.`
	`Module: 12 Enabling obfl-log ... was successful.`
		ブート動作時間、デバイス バージョン、および OBFL 履歴をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 10	（任意） **show logging onboard** 例： `switch(config)# show logging onboard`	OBFL に関する情報を表示します。 （注） モジュールのフラッシュに保存されている OBFL 情報を表示するには、OBFL 設定の確認 （409 ページ）を参照してください。
ステップ 11	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

OBFL 設定の確認

モジュールのフラッシュに保存されている OBFL 情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show logging onboard boot-uptime	ブートおよび動作時間の情報を表示します。
show logging onboard counter-stats	すべての ASIC カウンタについて、統計情報を表示します。
show logging onboard credit-loss	OBFL クレジット損失のログを表示します。
show logging onboard device-version	デバイス バージョン情報を表示します。
show logging onboard endtime	指定した終了時刻までの OBFL ログを表示します。
show logging onboard environmental-history	環境履歴を表示します。
show logging onboard error-stats	エラー統計情報を表示します。
show logging onboard exception-log	例外ログ情報を表示します。
show logging onboard interrupt-stats	割り込み統計情報を表示します。
show logging onboard module スロット	指定したモジュールの OBFL 情報を表示します。
show logging onboard obfl-history	履歴情報を表示します。
show logging onboard obfl-logs	ログ情報を表示します。
show logging onboard stack-trace	カーネル スタック トレース情報を表示します。

コマンド	目的
show logging onboard starttime	指定した開始時刻からの OBFL ログを表示します。
show logging onboard status	OBFL ステータス情報を表示します。

OBFL の設定ステータスを表示するには、**show logging onboard status** コマンドを使用します。

```
switch# show logging onboard status
----------------------------
OBFL Status
----------------------------
Switch OBFL Log: Enabled

Module: 4 OBFL Log: Enabled
cpu-hog Enabled
credit-loss Enabled
environmental-history Enabled
error-stats Enabled
exception-log Enabled
interrupt-stats Enabled
mem-leak Enabled
miscellaneous-error Enabled
obfl-log (boot-uptime/device-version/obfl-history) Enabled
register-log Enabled
request-timeout Enabled
stack-trace Enabled
system-health Enabled
timeout-drops Enabled
stack-trace Enabled

Module: 22 OBFL Log: Enabled
cpu-hog Enabled
credit-loss Enabled
environmental-history Enabled
error-stats Enabled
exception-log Enabled
interrupt-stats Enabled
mem-leak Enabled
miscellaneous-error Enabled
obfl-log (boot-uptime/device-version/obfl-history) Enabled
register-log Enabled
request-timeout Enabled
stack-trace Enabled
system-health Enabled
timeout-drops Enabled
stack-trace Enabled
```

上記の各 **show** コマンド オプションの OBFL 情報を消去するには、**clear logging onboard** コマンドを使用します。

OBFL のコンフィギュレーション例

モジュール 2 で環境情報について OBFL を有効にする例を示します。

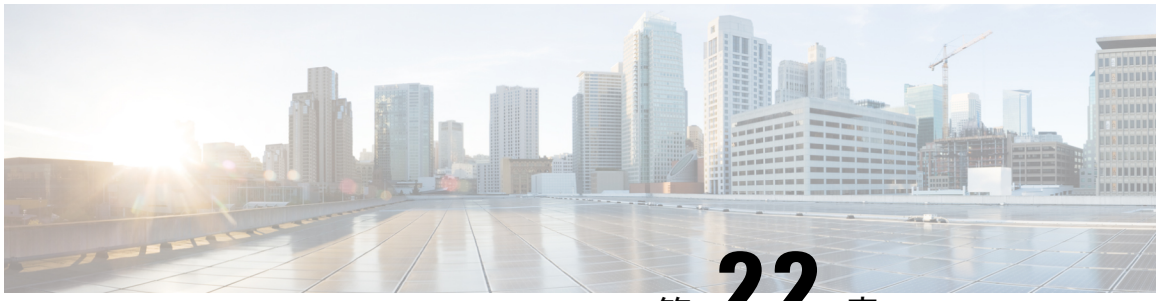
```
switch# configure terminal
switch(config)# hw-module logging onboard module 2 environmental-history
```

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
コンフィギュレーション ファイル	『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』

関連資料



第 22 章

SPAN の設定

この章では、Cisco NX-OS デバイス上のポート間のトラフィックを分析するようにイーサネット スイッチド ポート アナライザ（SPAN）を設定する方法について説明します。

- SPAN の概要, on page 413
- SPAN の前提条件 （417 ページ）
- SPAN の注意事項および制約事項 （417 ページ）
- SPAN のデフォルト設定 （429 ページ）
- SPAN の設定 （430 ページ）
- SPAN 設定の確認 （444 ページ）
- SPAN のコンフィギュレーション例 （444 ページ）
- その他の参考資料 （449 ページ）

SPAN の概要

SPAN は、外付けアナライザが接続された宛先ポートに SPAN セッション トラフィックを送ることで、送信元ポート間のすべてのトラフィックを分析します。

ローカル デバイス上で、SPAN セッションでモニタする送信元と宛先を定義できます。

SPAN ソース

トラフィックを監視できる監視元インターフェイスのことを SPAN 送信元と呼びます。送信元では、監視するトラフィックを指定し、さらに入力（Rx）、出力（Tx）、または両方向のトラフィックをコピーするかどうかを指定します。SPAN 送信元には次のものが含まれます。

- イーサネット ポート（ただしサブインターフェイスではない）
- コントロール プレーン CPU への帯域内インターフェイス。

Note SPAN 送信元としてスーパーバイザ インバンド インターフェイスを指定すると、デバイスはスーパーバイザ CPU により送信されたすべてのパケットをモニタします。

- VLAN
  - VLAN を SPAN 送信元として指定する場合は、VLAN 内でサポートされているすべてのインターフェイスが SPAN ソースになります。
  - VLAN は、入力方向にのみ SPAN 送信元となることができます。

Note これは、Cisco Nexus 9300-EX/-FX/-FX2/-FX3/-GX プラットフォームスイッチ、および -EX/-FX ライン カードを搭載する Cisco Nexus 9500 シリーズ プラットフォームスイッチを除くすべてのスイッチに適用されます。

- Cisco Nexus 2000 シリーズ ファブリック エクステンダ（FEX）のサテライト ポートおよびホスト インターフェイス ポート チャネル
  - これらのインターフェイスは、レイヤ 2 アクセス モードおよびレイヤ 2 トランク モードでサポートされます。レイヤ 3 モードではサポートされず、レイヤ 3 サブインターフェイスはサポートされません。
  - Cisco Nexus 9300 および 9500 プラットフォーム スイッチは、FEX ポートを SPAN 送信元としてサポートします。この場合、入力方向については、すべてのトラフィックを対象としますが、出力方向については、スイッチと FEX を通る既知のレイヤ 2 ユニキャスト トラフィック フローに限られます。ルーティングされたトラフィックは FEX HIF 出力 SPAN で表示されないことがあります。

Note 1 つの SPAN セッションに、上述の送信元を組み合わせて使用できます。

送信元ポートの特性

SPAN 送信元ポートには、次の特性があります。

- 送信元ポートとして設定されたポートを宛先ポートとしても設定することはできません。
- スーパーバイザ インバンド インターフェイスを SPAN 送信元として使用する場合、スーパーバイザ ハードウェア（出力）によって生成されたすべてのパケットがモニタされます。

Note Rx は ASIC の観点から見たものです（トラフィックはインバンドを介してスーパーバイザから出力され、ASIC / SPAN で受信されます）。

SPAN 宛先

SPAN 宛先とは、送信元ポートを監視するインターフェイスを指します。宛先ポートは SPAN 送信元からコピーされたトラフィックを受信します。SPAN 宛先には、次のものが含まれます。

- アクセス モードまたはトランク モードのイーサネット ポート
- アクセス モードまたはトランク モードのポート チャネル
- 宛先ポートとしての CPU
- Cisco Nexus 9300 シリーズ スイッチのアップリンク ポート

Note FEX ポートは SPAN 宛先ポートとしてサポートされません。

宛先ポートの特性

SPAN 宛先元ポートには、次の特性があります。

- 宛先ポートとして設定されたポートは、送信元ポートとして設定できません。
- 同じ宛先インターフェイスを、複数の SPAN セッションに使用することはできません。ただし、インターフェイスは SPAN および ERSPAN セッションの宛先として機能できます。
- 宛先ポートはスパニングツリー インスタンスに関与しません。SPAN 出力には、ブリッジ プロトコル データ ユニット（BPDU）スパニング ツリー プロトコル hello パケットを含みます。

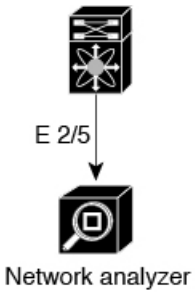
SPAN セッション

SPAN セッションを作成し、送信元と宛先をモニタに指定できます。

サポートされる SPAN セッション数に関する情報については、『*Cisco Nexus 9000* シリーズ *NX-OS* 検証済みスケーラビリティ ガイド』を参照してください。

この図では、SPAN 設定を示します。3 つのイーサネット ポート上のパケットが宛先ポートのイーサネット 2/5 にコピーされます。コピーされるのは、指定した方向のトラフィックだけです。

図 *7: SPAN* の設定



Source Port	Direction	Destination Ports
E 2/1	Rx	E 2/5
E 2/2	Rx, Tx	
E 2/3	Tx	

503005

ローカライズされた SPAN セッション

すべての送信元インターフェイスが同じライン カード上にある場合、SPANセッションはローカライズされます。セッション宛先インターフェイスは、任意のライン カードに配置できます。

（注） VLAN 送信元との SPAN セッションはローカライズされません。

SPAN 切り捨て

Cisco NX-OS Release 7.0(3)I7(1) 以降では、MTU のサイズに基づいて各 SPAN セッションの送信元パケットの切り捨てを設定できます。切り捨てにより、モニタするパケットのサイズを減らすことで、SPAN の帯域幅を効果的に軽減できます。設定された MTU サイズよりも大きい SPAN パケットはすべて、設定されたサイズに切り捨てられます。たとえば、MTU を 300 バイトに設定すると、300 バイトを超えるパケットは 300 バイトに切り捨てられます。

SPAN切り捨てはデフォルトでディセーブルです。切り捨てを使用するには、個々のSPANセッションで有効にしておく必要があります。

ACL TCAM リージョン

ハードウェアの ACL Ternary Content Addressable Memory（TCAM）リージョンのサイズを変更できます。SPANセッションで使用されるTCAMリージョンの詳細については、『Cisco Nexus 9000シリーズNX-OSセキュリティ設定ガイド』の「IP ACLの設定」のセクションを参照してください。

高可用性

SPAN機能はステートレスおよびステートフル リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバー後に、実行コンフィギュレーションを適用します。ハイ アベイラビリティの詳細については、『Cisco Nexus 9000 シリーズ NX-OS ハイ アベイラビリティおよび冗長性ガイド』を参照してください。

SPAN の前提条件

SPAN の前提条件は、次のとおりです。

- 各デバイス上で、まず所定の SPAN 設定をサポートするポートを設定する必要があります。詳細については、『Cisco Nexus 9000 シリーズ NX-OS インターフェイス設定ガイド』を参照してください。

SPAN の注意事項および制約事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

SPAN に関する設定時の注意事項および制約事項は、次のとおりです。

- SPAN セッション（Rx および Tx、Rx、または Tx）ごとに最大 48 の送信元インターフェイスがサポートされます。
- ACL によって拒否されたトラフィックは、SPAN 宛先ポートに到達する可能性があります。これは、SPAN 複製が ACL の適用（ACL ドロップ トラフィック）の前に入力側で実行されるためです。
- SPAN セッションの制限については、『*Cisco Nexus 9000* シリーズ *NX-OS* 検証スケーラビリティ ガイド』を参照してください。
- SPAN セッションの構成時に、最大 32 の送信元 VLAN を構成できます。
- すべての SPAN のレプリケーションはハードウェアで行われます。スーパーバイザ CPU は関与しません。
- SPAN セッションを設定できるのはローカル デバイス上だけです。
- 同じ送信元インターフェイスで 2 つの SPAN または ERSPAN セッションを 1 つのフィルタだけで設定することはできません。同じ送信元が複数の SPAN または ERSPAN セッションで使用されている場合は、すべてのセッションに異なるフィルタを設定するか、セッションにフィルタを設定しないでください。
- FCS エラーがあるパケットは、SPAN セッションでミラーリングされません。
- アクセス ポート dot1q ヘッダーの SPAN コピーには、次のガイドラインが適用されます。
  - トラフィックがトランクポート もしくはルーテッド ポート から入力され、アクセス ポートに出力された場合、スイッチ インターフェイス上のアクセス ポートの出力 SPAN コピーには常に dot1q ヘッダーが含まれます。

- トラフィックがアクセス ポートから入り、トランクポート もしくはルーテッド ポート に出た場合、スイッチ インターフェイスのアクセス ポートの入力 SPAN コピーには dot1q ヘッダーが含まれません。
- トラフィックがアクセス ポートから入力され、アクセス ポートに出力される場合、スイッチ インターフェイス上のアクセス ポートの入力/出力 SPAN コピーには dot1q ヘッダーがありません。
- この動作は、9700-EX、9700-FX、9700-GX ラインカードを備えた Cisco Nexus 9300-EX、9300-FX、9300-FX2、9300-FX3、9300-GX、9300-GX2、9500 プラットフォーム スイッチに適用されます。

- SAPN セッションで 1 つの宛先ポートはのみ設定できます。
- SPAN ミラーリングは、PBR トラフィックではサポートされません。
- ポートを送信元ポートと宛先ポートの両方として設定することはできません。
- SPAN 送信元ポートと宛先ポートでの単方向リンク検出（UDLD）の同時イネーブル化はサポートされていません。UDLD フレームがこのような SPAN セッションの送信元ポートでキャプチャされることが予想される場合は、SPAN セッションの宛先ポートで UDLD をディセーブルにします。
- SPAN は、管理ポートではサポートされません。
- フィルタ アクセス グループの統計情報はサポートされていません。
- 単一のトラフィック フローがCPU（Rx SPAN）とイーサネット ポート（Tx SPAN）にスパンされる場合、両方の SPAN コピーがポリシングされます。**hardware rate-limiter span** コマンドによって設定されたポリサー値は、CPU に向かう SPAN コピーとイーサネットインターフェイスに向かう SPAN コピーの両方に適用されます。この制限は、次のスイッチに適用されます。
  - Cisco Nexus 92348GC-X、Cisco Nexus 9332C、および Cisco Nexus 9364C スイッチ
  - Cisco Nexus 9300 EX、 FX、FX2、FX3、GX プラットフォーム スイッチ
  - EX および FX ライン カードを備えた Cisco Nexus 9504、9508 および 9516 プラットフォーム スイッチ
- SPAN はレイヤ 3 モードでサポートされます。ただし、SPAN はレイヤ 3 サブインターフェイスまたはレイヤ 3 ポートチャネル サブインターフェイスではサポートされません。
- SPAN セッションに、送信方向または送信および受信方向でモニタされている送信元ポートが含まれている場合、パケットが実際にはその送信元ポートで送信されなくても、これらのポートを受け取るパケットが SPAN の宛先ポートに複製される可能性があります。送信元 ポート上でのこの動作の例を、次に示します。
  - フラッディングから発生するトラフィック
  - ブロードキャストおよびマルチキャスト トラフィック

- SPAN セッションは、セッションの送信元がスーパーバイザのイーサネット インバンド インターフェイスの場合、ARP 要求および Open Shortest Path First（OSPF）プロトコル helloパケットのようなスーパーバイザに到達するブロードキャストまたはマルチキャスト MAC アドレスを持つパケットをキャプチャできません。これらのパケットをキャプチャするには、SPAN セッションの送信元として物理インターフェイスを使用する必要があります。
- VLAN SPAN がモニタするのは、VLAN のレイヤ 2 ポートを出入りするトラフィックだけです。
- VLAN は、SPAN 送信元またはフィルタとして使用される場合、属することができるのは 1 つのセッションだけです。
- SPAN 宛先ポートへの VLAN ACL リダイレクトはサポートされません。
- VLAN ACL を使用して SPAN をフィルタリングする場合、**action forward** のみがサポートされます。**action drop** および **action redirect** はサポートされていません。
- VLAN送信元セッションおよびポート送信元セッションの組み合わせはサポートされていません。トラフィック ストリームが VLAN 送信元セッションとポート送信元セッションと一致する場合、2 つの宛先ポートで 2 つのコピーが必要です。ハードウェアの制限により、VLAN 送信元 SPAN と特定の宛先ポートのみが SPAN パケットを受信します。この制限は、次のシスコ デバイスにのみ適用されます。

表 *18 : Cisco Nexus 9000* シリーズ スイッチ

Cisco Nexus 93120TX	Cisco Nexus 93128TX	Cisco Nexus 9332PQ
Cisco Nexus 9372PX	Cisco Nexus 9372PX-E	Cisco Nexus 9372TX
Cisco Nexus 9396PX	Cisco Nexus 9372TX-E	Cisco Nexus 9396TX

表 *19 : Cisco Nexus 9000* シリーズ ラインカード、ファブリック モジュールおよび *GEM* モジュール

N9K-X9408PC-CFP2	N9K-X9536PQ	N9K-C9504-FM
N9K-X9432PQ	N9K-X9464TX	—

- モニターセッションをフィルタリングする場合は、指定されたアクセス グループが、フィルタリング目的の通常の ACL ではなく、VACL または VLAN アクセス マップでなければならないことを確認してください。このガイドラインは、9636C-R および9636Q-R ラインカードを搭載した Cisco Nexus 9508 スイッチには適用されません。
- SPAN セッションのアクセス グループ フィルタは、vlan-accessmap として設定する必要があります。このガイドラインは、9636C-R および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 スイッチには適用されません。
- スーパーバイザ生成の Stream Of Bytes Module Header（SOBMH）パケットには、インターフェイスから出力されるための情報がすべて含まれており、SPAN および ERSPAN を含めた、ハードウェア内部でのフォワーディング ルックアップはすべてバイパス可能です。レ

イヤ 3 インターフェイスの CPU 生成フレームおよびパケットのブリッジ プロトコル データ ユニット（BPDU）クラスは、SOBMH を使用して送信されます。このガイドラインは、9636C-R および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 スイッチには適用されません。Cisco Nexus 9636C-R と 9636Q-R は両方とも、インバンド SPAN とローカル SPAN をサポートします。

- Cisco NX-OS は、送信元インターフェイスがホスト インターフェイス ポート チャネルでないときは、リンク層検出プロトコル（LLDP）またはリンク集約制御プロトコル（LACP）パケットをスパンしません。
- マルチキャスト パケットの SPAN コピーは、書き換え前に作成されます。したがって、TTL、VLAN ID、出力ポリシーによる再マーキングなどは、SPAN コピーにキャプチャされません。
- SPAN が ASIC インスタンスのインターフェイスに入力され、別の ASIC インスタンスのレイヤ 3 インターフェイス（SPAN 送信元）に出力されるトラフィックをミラーリングしている場合、Tx ミラーリング パケットは Cisco Nexus 9300 プラットフォーム スイッチ（EX、FX、または -FX2 を除く）および Cisco Nexus 9500 プラットフォーム モジュラー スイッチで 4095 の VLAN 識別子 をもちます。
- スイッチ インターフェイスのアクセス ポートの出力 SPAN コピーには、常に dot1q ヘッダーがあります。このガイドラインは、9636C-R および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 プラットフォーム スイッチには適用されません。
- 不明ユニキャストでフラッディングされたパケットのルーティング後のフローは SPAN セッションに置かれますが、これはフローが転送されるポートをモニタしないよう SPAN セッションが設定されている場合であっても同様です。この制限は、ネットワーク フォワーディング エンジン（NFE）と NFE2 対応 EOR スイッチおよび SPAN セッションで Tx ポートの送信元を持つものに適用されます。
- VLAN 送信元は、Rx 方向にのみスパンされます。この制限は、両方向の VLAN スパニングをサポートする次のスイッチ プラットフォームには適用されません。
  - Cisco Nexus 9300-EX プラットフォーム スイッチ
  - Cisco Nexus 9300-FX プラットフォーム スイッチ
  - Cisco Nexus 9300-FX2 プラットフォーム スイッチ
  - Cisco Nexus 9300-FX3 プラットフォーム スイッチ
  - Cisco Nexus 9300-GX プラットフォーム スイッチ
  - 97160YC-EX ライン カードを搭載した Cisco Nexus 9504、9508 および 9516 スイッチ。
  - 9636C-R および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 スイッチ。
- VLAN 送信元が 1 つのセッションで両方向として設定され、物理インターフェイス送信元が他の 2 つのセッションで設定されている場合、物理インターフェイス送信元セッションでは Rx SPAN はサポートされません。この制限は、Cisco Nexus 97160YC-EX ライン カードに適用されます。

- セッション フィルタリング機能に関しては、ACL フィルタは Rx ソースでのみサポートされ、VLAN フィルタは Tx および Rx ソースの両方でサポートされます。このガイドラインは、9636C-R および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 スイッチには適用されません。
- VLAN フィルタが構成されている場合、複数のスパン セッションで同じソースを構成することはできません。
- FEX NIF インターフェイスまたはポート チャネルは、SPAN 送信元または SPAN 宛先として使用できません。FEX NIF インターフェイスまたはポート チャネルが SPAN 送信元または SPAN 宛先として指定されている場合、ソフトウェアではサポートされていないエラーが表示されます。
- SPAN / ERSPAN を使用して FEX HIF ポートで Rx トラフィックをキャプチャすると、キャプチャされたトラフィックに追加の VNTAG および 802.1Q タグが存在します。
- VLAN および ACL フィルタは FEX ポートではサポートされません。
- 双方向 SPAN セッションで使用される送信元が同じ FEX からのものである場合、ハードウェア リソースは 2 つの SPAN セッションに制限されます。
- 切り捨てはローカルおよびERSPAN送信元セッションでのみサポートされます。それは、ERSPAN 宛先セッションではサポートされません。
- sFlow が N9K-X9716D-GX ライン カードを使用して N9K-C9508-FM-G で設定されている場合は、SPAN セッションを設定する前に sFlow を無効にします。
- SPAN セッションで MTU を設定すると、（そのセッションの）SPAN 宛先で出力されるすべてのパケットが、指定した MTU 値に切り捨てられます。
  - 切り捨てられたパケットの巡回冗長検査（CRC）が再計算されます。
  - 指定されたバイトは、パケットのヘッダーから保持されます。パケットが MTU より長い場合、残りは切り捨てられます。
- Cisco NX-OS リリース 10.1(2) 以降、SPAN は Cisco Nexus N9K-X9624D-R2 ライン カードでサポートされます。
- Cisco NX-OS リリース 10.2(1q)F 以降、SPAN は N9K-C9332D-GX2B プラットフォーム スイッチでサポートされます。
- MTU トランケーションは、Cisco Nexus 9504/9508 モジュラ シャーシ（N9K-X9636C-R、N9K-X9636Q-R、N9K-X9636C-RX、および N9K-X96136YC-R ライン カードを搭載）ではサポートされません。
- Cisco NX-OS リリース 10.2(2)F 以降では、マルチキャスト SPAN Tx が Cisco Nexus 9300-GX、9300-GX2、および 9300-GX3 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.3(1)F 以降、Cisco Nexus 9808 プラットフォーム スイッチで SPAN のサポートが提供されます。

- Cisco NX-OS リリース 10.4(1)F 以降、SPAN は次のスイッチおよびライン カードでサポートされます。
  - Cisco Nexus 9804 スイッチ
  - Cisco Nexus 9332D-H2R スイッチ
  - Cisco Nexus X98900CD-A および X9836DM-A ラインカードと Cisco Nexus 9808 および 9804 スイッチ

Cisco Nexus 3000 プラットフォーム スイッチの SPAN の制限

次の注意事項と制約事項は、Cisco Nexus 9000 コードを実行する Nexus 3000 シリーズ スイッチにのみ適用されます。

- Cisco Nexus 3232C および 3264Q スイッチは、宛先として CPU で SPAN をサポートしていません。

Cisco Nexus 9200 プラットフォーム スイッチの SPAN の制限事項（9232E-B1 を除く）

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

次の注意事項と制約事項は、Cisco Nexus 9200 プラットフォーム スイッチにのみ適用されます。

- Cisco Nexus 9200 プラットフォーム スイッチの場合、Rx SPAN は、SPAN 宛先ポートと同じスライス上に転送インターフェイスがないマルチキャストではサポートされません。
- Cisco Nexus 9200 プラットフォーム スイッチでは、マルチキャスト、未知のマルチキャスト、およびブロードキャスト トラフィックに対する Tx SPAN はサポートされません。
- CPU 生成パケットの Tx SPAN は、Cisco Nexus 9200 プラットフォーム スイッチではサポートされません。
- UDF ベースの SPAN は、Cisco Nexus 9200 プラットフォーム スイッチでサポートされます。
- Cisco Nexus 9200 プラットフォーム スイッチは、同じ送信元での複数の ACL フィルタをサポートしていません。
- VLAN Tx SPAN は、Cisco Nexus 9200 プラットフォーム スイッチでサポートされます。

- 同じスライスにある複数の出力ポートで、出力 SPAN トラフィックのために輻輳が発生すると、Cisco Nexus 9200 プラットフォーム スイッチ上のこれらの出力ポートでは、ラインレートを取得できません。
- ACL フィルタを使用した、親インターフェイスでのサブインターフェイス トラフィックのスパンは、Cisco Nexus 9200 プラットフォーム スイッチではサポートされません。
- Cisco Nexus 9200 プラットフォーム スイッチでは、CPU SPAN ソースはRx 方向（CPU からの SPAN パケット）でのみ追加できます。
- Cisco Nexus 9200 プラットフォーム スイッチでは、CPU への SPAN パケットはレート制限され、インバンド パスでドロップされます。レート制限の変更は、 **hardware rate-limiter span** コマンドで行えます。スーパーバイザの SPAN コピーの分析は、 **ethanalyzer local interface inband mirror detail** コマンドで行えます。

Cisco Nexus 9300 プラットフォーム スイッチの SPAN の制限事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

次の注意事項と制約事項は、Cisco Nexus 9300 プラットフォーム スイッチにのみ適用されます。

- SPAN は、Cisco Nexus 9300-GX プラットフォーム スイッチの送信元での ECMP ハッシュ/ロード バランシングをサポートしません。
- 次のフィルタリング制限は、すべての Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォーム スイッチの出力（Tx）SPANに適用されます。
  - ACL フィルタリングはサポートされていません（ユニキャストおよびブロードキャスト、不明なユニキャストおよびマルチキャスト（BUM）トラフィックの両方に適用されます）
  - VLAN フィルタリングはサポートされますが、ユニキャスト トラフィックのみ
  - VLAN フィルタリングは BUM トラフィックではサポートされません。
- Cisco Nexus 9300-EX/FX プラットフォーム スイッチでは、SPAN とsFlow の両方を同時に有効にすることはできません。一方がアクティブな場合、もう一方は有効にできません。ただし、Cisco Nexus 9300-EX/FX/FX2 プラットフォーム スイッチでは、NetFlow と SPAN を同時に有効にすることができるので、sFlow と SPAN を併用する代わりに使用できます。

（注） Cisco Nexus 9300-FX2 スイッチは、sFlow と SPAN の共存をサポートします。

- VLAN Tx SPAN は、Cisco Nexus 9300-EX および FX プラットフォーム スイッチでサポートされます。
- Cisco Nexus 9300 プラットフォーム スイッチは、同じソースに対する複数の ACL フィルタをサポートします。
- 1 つのフォワーディング エンジン インスタンスで 4 つの SPAN セッションがサポートされます。Cisco Nexus 9300 シリーズ スイッチの場合は、最初の 3 つのセッションに双方向のソースが含まれていると、4 番目のセッションのハードウェア リソースは Rx ソース専用になります。
- Cisco Nexus 9300-EX/FX/FX2/FX3/FXP プラットフォーム スイッチは、入力方向の SPAN ソースとしてのみ FEX ポートをサポートします。
- Cisco Nexus 9300 プラットフォーム スイッチ（Cisco Nexus 9300-EX/FX/FX2/FX3/FXP スイッチを除く）は、FEX ポートを SPAN ソースとしてサポートします。この場合、入力方向については、すべてのトラフィックを対象としますが、出力方向については、スイッチと FEX を通る既知のレイヤ 2 ユニキャスト トラフィック フローに限られます。ルーティングされたトラフィックは FEX HIF 出力 SPAN で表示されないことがあります。
- Cisco Nexus 9300 シリーズ スイッチは、Tx SPAN を 40G アップリンク ポートでサポートしません

（注） この制限は、100G インターフェイスを持つ Nexus 9300-EX/FX/FX2 スイッチには適用されません。

- CPU 生成パケットの Tx SPAN は、Cisco Nexus 9200、9300-EX/FX/FXP/FX2/FX3/GX/GX2、9300C、C9516-FM-E2 および C9508-FM-E2 スイッチではサポートされません。
- 異なるスライス間でマルチキャスト Tx トラフィックの SPAN をサポートするのは、Cisco Nexus 9300-EX プラットフォーム スイッチだけです。スライスは同じリーフ スパイン エンジン（LSE）上にある必要があります。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォーム スイッチのレイヤ 2 スイッチ ポートおよびポートチャネル ソースを使用する Tx インターフェイス SPAN の場合、同じ VLAN でストリームを受信しているレイヤ 2 メンバーの数に関係なく、レシーバ ユニットごとに 1 つのコピーのみが作成されます。たとえば、e1/1 ~ 8 がすべて Tx 方向の SPAN ソースであり、すべてが同じグループに参加している場合、SPAN ディスティネーション ポートは、8 つのコピーではなく、書き換え前のストリームの 1 つのコピーを認識します。さらに、何らかの理由で、これらのポートの 1 つ以上が出力でパケットをドロップした場合でも（輻輳など）、パケットは SPAN ディスティネーション ポートに到達できます。Cisco Nexus 9732C-EX ライン カードの場合、メンバーを持つユニットごとに 1 つのコピーが作成されます。ポートチャネル ソースの場合、SPAN を実行するレイヤ 2 メンバーが最初のポートチャネル メンバーになります。
- SPAN Tx ブロードキャストおよび SPAN Tx マルチキャストは、Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォーム スイッチおよびCisco Nexus 9732C-EX ライン

カードのスライス全体のレイヤ 2 ポートおよびポートチャネル ソースでサポートされます。ただし IGMP スヌーピングがディセーブルの場合に限られます。（それ以外の場合は、スライスの制限が適用されます）。これらの機能は、レイヤ 3 ポート ソース、FEX ポート（ユニキャストまたはマルチキャスト トラフィック）、および VLAN ソースではサポートされません。

- レイヤ 2 の SPAN Tx マルチキャストの場合、マルチキャスト レプリケーションとは無関係に SPAN コピーが作成されます。このため、マルチキャストと SPAN パケットでは、VLAN タグ（入力インターフェイス VLAN ID）の値が異なります。
- Cisco Nexus 9300 シリーズ スイッチ 40G アップリンク インターフェイスの SPAN コピーは、Rx 方向にスパンする際に、dot1q 情報を取り逃がします。

（注） この制限は、100Gインターフェイスを持つNexus 9300-EX/FX/FX2プラットフォーム スイッチには適用されません。

- UDF ベースの SPAN は、Cisco Nexus 9300-EX/-FX/-FX2/FX3/GX プラットフォーム スイッチでサポートされます。
- UDF-SPAN の ACL フィルタリングはソース インターフェイス rx のみをサポートします。この制限は、次のスイッチに適用されます。
  - Cisco Nexus 9332PQ
  - Cisco Nexus 9372PX
  - Cisco Nexus 9372PX-E
  - Cisco Nexus 9372TX
  - Cisco Nexus 9372TX-E
  - Cisco Nexus 93120TX
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォーム スイッチは、同じソースの複数の ACL フィルタをサポートしていません。
- 同じスライスにある複数の出力ポートで、出力 SPAN トラフィックのために輻輳が発生すると、Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォーム スイッチ上のこれらの出力ポートでは、ライン レートを取得できません。
- ACL フィルタを使用した、親インターフェイスでのサブインターフェイス トラフィックのスパンは、Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォーム スイッチではサポートされません。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォーム スイッチでは、CPU SPAN ソースは Rx 方向（CPU からの SPAN パケット）でのみ追加できます。
- Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォームスイッチでは、CPU への SPAN パケットはレート制限され、インバンド パスでドロップされます。レート制限の変更は、

hardware rate-limiter span コマンドで行えます。スーパーバイザの SPAN コピーの分析は、 **ethanalyzer local interface inband mirror detail** コマンドで行えます。

- 次の Cisco Nexus スイッチは、sFlow と SPAN を同時にサポートします。
  - Cisco Nexus 9336C-FX2
  - Cisco Nexus 93240YC-FX2
  - Cisco Nexus 93360YC-FX2
- Cisco NX-OS リリース 9.3(3) 以降、Cisco Nexus 9300-GX プラットフォーム スイッチは、sFlow と SPAN の両方をサポートしています。
- Cisco NX-OS リリース 9.3(5) 以降、 Cisco Nexus 9300-GX プラットフォーム スイッチは SPAN 切り捨てをサポートしています。
- Cisco NX-OS リリース 10.1(1) 以降、sFlow および SPAN は Cisco N9K-C93180YC-FX3 プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.2(3)F 以降、FC スパン 機能は、Cisco Nexus C93180YC-FX、C9336C-FX2-E、および C93360YC-FX2 プラットフォームスイッチの NPV および SAN スイッチングモードの両方で、FC ポート、SAN ポートチャネル、および VSAN のパケットキャプチャサポートを提供します。
- FC ポート、SAN ポート チャネル、およびソースとしての VSAN は、ERSPAN ではサポートされていません。
- FC ポート、SAN ポート チャネル、および VSAN は、複数のスパン セッションでソースとして追加できません。
- ガイドライン — 単一の転送エンジン インスタンスは 4 つのアクティブな SPAN セッションをサポートします — は、FC スパン機能にも適用できます。
- FC スパン機能の SNMP サポートは、Cisco NX-OS リリース 10.2(3)F では使用できません。

Cisco Nexus 9500 プラットフォーム スイッチの SPAN の制限事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

次の注意事項と制約事項は、Cisco Nexus 9500 プラットフォーム スイッチにのみ適用されます。

- 次のフィルタリング制限は、EX または FX ライン カードを搭載した 9500 プラットフォーム スイッチの出力（Tx）SPANに適用されます。

- ACL フィルタリングはサポートされていません（ユニキャストおよびブロードキャスト、不明なユニキャストおよびマルチキャスト（BUM）トラフィックの両方に適用されます）
- VLAN フィルタリングはサポートされますが、ユニキャスト トラフィックのみ
- VLAN フィルタリングは BUM トラフィックではサポートされません。

- FEX および SPAN ポート チャネルの宛先は、EX または FX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチではサポートされません。
- EX/FX モジュールを搭載した Cisco Nexus 9500 プラットフォーム スイッチでは、SPAN と sFlow の両方を同時に有効にすることはできません。一方がアクティブな場合、もう一方は有効にできません。ただし、EX または FX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチでは、NetFlow と SPAN の両方を同時に有効にすることができ、sFlow と SPAN を使用する代わりに実行可能です。
- Cisco Nexus 9500 プラットフォーム スイッチは、次のライン カードを備えた VLAN Tx SPAN をサポートします。
  - Cisco Nexus 97160YC-EX
  - Cisco Nexus 9732C-EX
  - Cisco Nexus 9732C-FX
  - Cisco Nexus 9736C-EX
  - Cisco Nexus 9736C-FX
  - Cisco Nexus 9736Q-FX
  - Cisco Nexus 9788TC-FX
- Cisco Nexus 9500 プラットフォーム スイッチは、同じソースに対する複数の ACL フィルタをサポートします。
- CPU で生成されたパケットの Tx SPAN は、EX ベースのライン カードを搭載した Cisco Nexus 9500 プラットフォーム スイッチではサポートされません。
- TCAM カービングは、次のライン カードの SPAN/ERSPAN には必要ありません。
  - Cisco Nexus 9636C-R
  - Cisco Nexus 9636Q-R
  - Cisco Nexus 9636C-RX
  - Cisco Nexus 96136YC-R
  - Cisco Nexus 9624D-R2

（注） SPAN/ERSPAN をサポートする他のすべてのスイッチは、TCAM カービングを使用する必要があります。

- Cisco Nexus 9500 プラットフォーム スイッチでは、SPAN 送信元の転送エンジン インスタンス マッピングに応じて、単一の転送エンジンインスタンスが 4 つの SPAN セッションをサポートする場合があります。このガイドラインは、9636C-R および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 スイッチには適用されません。
- N9K-X96136YC-R ライン カードの複数の SPAN セッションで同じ送信元インターフェイスを構成することはできません。
- 複数の ACL フィルタは、同じ送信元ではサポートされません。
- Cisco Nexus 9500 プラットフォーム スイッチは、スイッチと FEX を通過する既知のレイヤ 2 ユニキャスト トラフィック フローに対してのみ、すべてのトラフィックの入力方向と出力方向の SPAN 送信元として FEX ポートをサポートします。ルーティングされたトラフィックが FEX HIF 出力 SPAN で表示されないことがあります。
- SPAN は、Cisco Nexus 9408PC-CFP2 ライン カード ポートの宛先をサポートしません。
- 切り捨ては、9700-EX または 9700-FX ライン カードを搭載した Cisco Nexus 9500 プラットフォーム スイッチでサポートされます。
- VLAN は、9636C-R および 9636Q-R ライン カードを備えた Cisco Nexus 9508 スイッチの入力および出力方向の SPAN 送信元にできます。
- UDF-SPAN acl-filtering は送信元インターフェイス rx のみをサポートします。この制限は、次のライン カードに適用されます。
  - Cisco Nexus 9564PX
  - Cisco Nexus 9464TX2
  - Cisco Nexus 9464TX
  - Cisco Nexus 9464TX2
  - Cisco Nexus 9564TX
  - Cisco Nexus 9464PX
  - Cisco Nexus 9536PQ
  - Cisco Nexus 9636PQ
  - Cisco Nexus 9432PQ

Cisco Nexus 9800 および 9232E-B1 プラットフォーム スイッチの SPAN の注意事項と制限事項

（注） スケールの情報については、Cisco.comにあるリリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

次の注意事項と制約事項は、Cisco Nexus 9800 および 9232-B1 プラットフォーム スイッチにのみ適用されます。

- RX のみが CPU への SPAN でサポートされます。
- セッション間での同じ送信元ポートまたはインターフェイスの共有はサポートされていません。
- 一度に最大 10 台のモニタ セッションがサポートされます。
- 一度に 10 個のアクティブ SPAN セッションがサポートされます。
- モニタの統計は、SPAN から CPU については表示されません。
- SPAN は、L2 ポート、ポート チャネル、およびトンネル ポートではサポートされていません。
- VLAN 送信元での SPAN はサポートされていません。
- MTU 切り捨ては、FCS を除き、
- MTU の切り捨ては RX でのみサポートされ、TX ではサポートされません。
- UDF はサポートされていません。
- SPAN は、サブインターフェイスではサポートされていません。
- SDK の制限により、ミラーリングされたマルチキャスト トラフィックは、マルチキャスト トラフィックではなくユニキャスト トラフィックとして処理されます。

SPAN のデフォルト設定

次の表に、SPAN パラメータのデフォルト設定を示します。

パラメータ	デフォルト
SPAN セッション	シャット ステートで作成されます

SPAN の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドと異なる場合があります。

SPAN セッションの設定

SPAN セッションを設定できるのはローカル デバイス上だけです。デフォルトでは、SPAN セッションはシャット ステートで作成されます。

Note 双方向性の従来のセッションでは、トラフィックの方向を指定せずにセッションを設定できます。

Before you begin

アクセス モードまたはトランク モードで宛先ポートを設定する必要があります。詳細については、『Cisco Nexus 9000 シリーズ NX-OS インターフェイス設定ガイド』を参照してください。

SUMMARY STEPS

1. **configure terminal**
2. **interface** *ethernet slot*/*port*
3. **switchport**
4. **switchport monitor**
5. (Optional) ステップ 2 ～ 4 を繰り返して、追加の SPAN 宛先でモニタリングを設定します。
6. **no monitor session** *session-number*
7. **monitor session** *session-number*[**rx | tx**] [**shut**]
8. **description** *description*
9. **source** {**interface** *type* [**rx** | **tx** | **both**] | [**vlan** {*number* | *range*}[**rx**]} | [**vsan** {*number* | *range*}[**rx**]}
10. (Optional) ステップ 9 を繰り返して、すべての SPAN 送信元を設定します。
11. **filter vlan** {*number* | *range*}
12. (Optional) ステップ 11 を繰り返して、すべての送信元 VLAN のフィルタリングを設定します。
13. (Optional) **filter access-group** *acl-filter*
14. **destination interface** *type slot*/*port*
15. **no shut**
16. (Optional) **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**]
17. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	**interface** *ethernet slot/port*
	Example:
	`switch(config)# interface ethernet 2/5`
	`switch(config-if)#`
		選択したスロットおよびポート上でインターフェイス 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	**switchport** **Example:** `switch(config-if)# switchport`	選択したスロットおよびポートまたはポート範囲でスイッチポート パラメータを設定します。
ステップ 4	**switchport monitor** **Example:** `switch(config-if)# switchport monitor`	SPAN 宛先としてスイッチポート インターフェイスを設定します。
ステップ 5	(Optional) ステップ 2 ～ 4 を繰り返して、追加の SPAN 宛先でモニタリングを設定します。	—
ステップ 6	**no monitor session** *session-number* **Example:** `switch(config)# no monitor session 3`	指定した SPAN セッションのコンフィギュレーションを消去します。新しいセッション コンフィギュレーションは、既存のセッション コンフィギュレーションに追加されます。

ステップ 7	**monitor session** *session-number*[**rx** | **tx**] [**shut**]
	Example:
	`switch(config)# monitor session 3 rx`
	`switch(config-monitor)#`
	Example:
	`switch(config)# monitor session 3 tx`
	`switch(config-monitor)#`
	Example:
	`switch(config)# monitor session 3 shut`
	`switch(config-monitor)#`
		モニタ コンフィギュレーション モードを開始します。新しいセッション コンフィギュレーションは、既存のセッション コンフィギュレーションに追加されます。デフォルトでは、セッションが shut ステートで作成されます。このセッションは、ローカル SPAN セッションです。オプションの shut キーワードは、選択したセッションに対して shut ステートを指定します。

ステップ 8	**description** *description* **Example:** `switch(config-monitor)# description my_span_session_3`	セッションの説明を設定します。デフォルトでは、説明は定義されません。説明には最大 32 の英数字を使用できます。

	Command or Action	Purpose
ステップ 9	**source** {**interface** *type* [**rx** | **tx** | **both**] | [**vlan** {*number* | *range*}[**rx**]} | [**vsan** {*number* | *range*}[**rx**]}	送信元およびパケットをコピーするトラフィックの方向を設定します。一定範囲のイーサネット ポート、FC ポート、ポート チャネル、SAN ポート チャネル、インバンド インターフェイス、一定範囲の VLAN、一定範囲の VSAN または Cisco Nexus 2000 シリーズ ファブリック エクステンダ（FEX）上のサテライト ポートまたはホスト インターフェイス ポート チャネルを入力できます。 送信元は 1 つ設定することも、またはカンマで区切った一連のエントリとして、または番号の範囲として、複数設定することもできます。 コピーするトラフィックの方向は、受信（rx）、送信（tx）、または両方（both）を設定できます。 **Note** 送信元 VLAN は、入力方向でのみサポートされます。送信元 FEX ポートは、すべてのトラフィックに対して入力方向でサポートされ、既知のレイヤ 2 ユニキャスト トラフィックには出力方向のみがサポートされます。 この注意事項は、Cisco Nexus EX/-FX/-FX2/-FX3/-GX シリーズ プラットフォーム スイッチ、および -EX/-FX ライン カードを備えた Cisco Nexus 9500 シリーズ プラットフォーム スイッチには適用されません。 送信元としてのスーパーバイザは、Rx 方向でのみサポートされます。 単一方向のセッションには、送信元の方向はセッションで指定された方向に一致する必要があります。 **Note** 送信元 VSAN もまた、入力方向でのみサポートされます。

	Example:
	`switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	Example:
	`switch(config-monitor)# source interface fc1/1 both`
	Example:
	`switch(config-monitor)# source interface port-channel 2`
	Example:
	`switch(config-monitor)# source interface san-port-channel201 both`
	Example:
	`switch(config-monitor)# source interface sup-eth 0 rx`
	Example:
	`switch(config-monitor)# source vlan 3, 6-8 rx`
	Example:
	`switch(config-monitor)# source vsan 500 rx`
	Example:
	`switch(config-monitor)# source interface ethernet 101/1/1-3`	

ステップ 10	(Optional) ステップ 9 を繰り返して、すべての SPAN 送信元を設定します。	
ステップ 11	**filter vlan** {*number* | *range*} **Example:** `switch(config-monitor)# filter vlan 3-5, 7`	設定された送信元から選択する VLAN を設定します。VLAN は 1 つ設定することも、またはカンマで区切った一連のエントリとして、または番号の範囲として、複数設定することもできます。

	Command or Action	Purpose
		Note SPAN送信元として設定されたFEXポートはVLANフィルタをサポートしません。 **Note** 送信元が FC インターフェイスまたは VSAN の場合、フィルタはサポートされません。
ステップ 12	(Optional) ステップ 11 を繰り返して、すべての送信元 VLAN のフィルタリングを設定します。	
ステップ 13	(Optional) **filter access-group** *acl-filter* **Example:** `switch(config-monitor)# filter access-group ACL1`	ACL を SPAN セッションにアソシエートします。 **Note** 送信元が FC インターフェイスまたは VSAN の場合、フィルタはサポートされません。
ステップ 14	Required: **destination interface** *type slot/port* **Example:** `switch(config-monitor)# destination interface ethernet 2/5`	コピーする送信元パケットの宛先を設定します。 **Note** FCポートは接続先インターフェイスとしてサポートされていません。 **Note** SPAN 宛先ポートは、アクセス ポートまたはトランク ポートのどちらかにする必要があります。 **Note** 宛先ポートでモニタ モードを有効にする必要があります。 次のプラットフォーム スイッチの SPAN 宛先として CPU を設定できます。 • Cisco Nexus 9200 シリーズ スイッチ（Cisco NX-OS リリース 7.0(3)I4(1) 以降） • Cisco Nexus 9300-EX シリーズ スイッチ（Cisco NX-OS リリース 7.0(3)I4(2) 以降） • Cisco Nexus 9300-FX シリーズ スイッチ（Cisco NX-OSリリース 7.0(3)I7(1) 以降） • Cisco Nexus 9300-FX2 シリーズ スイッチ（Cisco NX-OSリリース 7.0(3)I7(3) 以降） • Cisco Nexus 9300-FX3 シリーズ スイッチ（Cisco NX-OSリリース 9.3(5) 以降）

	Command or Action	Purpose
		• Cisco Nexus 9300-GXシリーズスイッチ（Cisco NX-OSリリース 9.3(3) 以降） • -EX/FX ライン カード搭載の Cisco Nexus 9500-EX シリーズ スイッチ これを行うには、インターフェイス タイプに **sup-eth 0** を入力します。
ステップ 15	Required: **no shut** **Example:** `switch(config-monitor)# no shut`	SPANセッションをイネーブルにします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 16	(Optional) **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**] **Example:** `switch(config-monitor)# show monitor session 3`	SPAN 設定を表示します。
ステップ 17	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

UDF ベース SPAN の設定

外部または内部パケット フィールド（ヘッダまたはペイロード）のユーザ定義フィールド（UDF）で照合し、一致するパケットを SPAN 宛先に送信するようにデバイスを設定できます。そのように設定することで、ネットワークのパケット ドロップを分析して、分離することができます。

始める前に

UDF ベース SPAN をイネーブルにするのに十分な空き領域を確保するために、**hardware access-list tcam region** コマンドを使用して適切なTCAMリージョン（racl、ifacl、またはvacl）が設定されていることを確認します。詳細については『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』の「Configuring ACL TCAM Region Sizes」の項を参照してください。

手順の概要

1. **configure terminal**
2. **udf** *udf-name offset-base offset length*
3. **hardware access-list tcam region** {**racl | ifacl | vacl** } **qualify** *qualifier-name*
4. **copy running-config startup-config**
5. **reload**
6. **ip access-list** *span-acl*

7. 次のいずれかのコマンドを入力します。
   - **permit udf** *udf-name value mask*
   - **permit ip** *source destination* **udf** *udf-name value mask*
8. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**udf** *udf-name offset-base offset length*
	例：
	`switch(config)# udf udf-x packet-start 12 1`
	`switch(config)# udf udf-y header outer l3 20 2`
		次のように UDF を定義します。
		• *udf-name*：UDF の名前を指定します。名前には最大 16 文字の英数字を入力できます。
		• *offset-base*：UDF オフセット ベースを以下のように指定します。ここで**header**は、オフセットのために考慮に入れるべきパケット ヘッダーです： **packet-start** | **header** {**outer** | **inner** {**l3** | **l4**}}.
		• オフセット：オフセットベースからのオフセット バイト数を指定します。オフセット ベース（レイヤ 3 /レイヤ 4 ヘッダー）の最初のバイトを照合するには、オフセットを 0 に設定します。
		• 長さ：オフセット からバイトの数を指定します。1 または 2 バイトのみがサポートされています。追加のバイトに一致させるためには、複数の UDF を定義する必要があります。
		複数の UDF を定義できますが、シスコは必要な UDF のみ定義することを推奨します。

ステップ 3	**hardware access-list tcam region** {**racl** | **ifacl** | **vacl** } **qualify** *qualifier-name* 例： `switch(config)# hardware access-list tcam region racl qualify ing-l3-span-filter`	次のいずれかの TCAM リージョンに UDF を付加します。 • racl：レイヤ 3 ポートに適用されます。 • ifacl：レイヤ 2 ポートに適用します。 • vacl：送信元 VLAN に適用します。

	コマンドまたはアクション	目的
		UDF は TCAM リージョンに最大 8 個まで付加できます。 （注） UDF 修飾子が追加されると、TCAM リージョンはシングル幅から倍幅に拡大します。十分な空きスペースがあることを確認してください。 それ以外の場合このコマンドは拒否されます。必要な場合、未使用のリージョンから TCAM スペースが減りますので、このコマンドを再入力します。詳細については『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』の「Configuring ACL TCAM Region Sizes」の項を参照してください。 （注） このコマンドの **no** 形式は、UDF を TCAM リージョンから切り離し、リージョンをシングル幅に戻します。
ステップ 4	必須: **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。
ステップ 5	必須: **reload** 例： `switch(config)# reload`	デバイスがリロードされます。 （注） UDF 設定は **copy running-config startup-config** + **reload** を入力した後のみ有効になります。

ステップ 6	**ip access-list** *span-acl*
	例：
	`switch(config)# ip access-list span-acl-udf-only`
	`switch(config-acl)#`
		IPv4 アクセス コントロール リスト（ACL）を作成して、IP アクセス リスト コンフィギュレーション モードを開始します。

ステップ 7	次のいずれかのコマンドを入力します。
	• **permit udf** *udf-name value mask*
	• **permit ip** *source destination* **udf** *udf-name value mask*
	例：
	`switch(config-acl)# permit udf udf-x 0x40 0xF0 udf-y 0x1001 0xF00F`
	例：
	`switch(config-acl)# permit ip 10.0.0./24 any udf udf-x 0x02 0x0F udf-y 0x1001 0xF00F`
		ACL を設定し、UDF（例 1）でのみ、または外部パケット フィールドについて現在のアクセス コントロール エントリ（ACE）と併せて UDF で一致させるように設定します（例 2）
		シングル ACL は、UDFがある場合とない場合の両方とも、ACE を有することができます。各 ACE には一致する異なる UDF フィールドがあるか、すべての ACE を UDF の同じリストに一致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 8	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SPAN 切り捨ての設定

切り捨ては、ローカルおよび SPAN 送信元セッションに対してのみ設定できます。

手順の概要

1. **configure terminal**
2. **monitor session** *session number*
3. **source interface** *type slot*/*port* [**rx** | **tx** | **both**]
4. **mtu** *size*
5. **destination interface** *type slot*/*port*
6. **no shut**
7. （任意） **show monitor session** *session*
8. **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**monitor session** *session number*
	例：
	`switch(config)# monitor session 5`
	`switch(config-monitor)#`
		指定した SPAN セッションのモニタ コンフィギュレーション モードを開始します。

ステップ 3	**source interface** *type slot*/*port* [**rx**	**tx**	**both**] 例： `switch(config-monitor)# source interface ethernet 1/5 both`	送信元インターフェイスを設定します。
ステップ 4	**mtu** *size* 例： `switch(config-monitor)# mtu 320`	MTU の切り捨てサイズを設定します。設定された MTU サイズよりも大きい SPAN パケットはすべて、設定されたサイズに切り捨てられます。SPAN パケット切り捨ての MTU 範囲は次のとおりです。

	コマンドまたはアクション	目的

	例：
	`switch(config-monitor)# mtu ?`
	`<320-1518> Enter the value of MTU truncation size for SPAN packets`
		• Cisco Nexus 9300-EX プラットフォーム スイッチの MTU サイズの範囲は、320ー1518 バイトです。
		• Cisco Nexus 9300-FX プラットフォーム スイッチの MTU サイズの範囲は 64ー1518 バイトです。
		• 9700-EX および 9700-FX ライン カードを搭載した Cisco Nexus 9500 プラットフォーム スイッチの場合、MTU サイズの範囲は 320ー1518 バイトです。
		• Cisco Nexus 9808 および 9804 プラットフォーム スイッチの MTU サイズは 343 バイトです（FCS を除く）。

ステップ 5	**destination interface** *type slot/port* 例： `switch(config-monitor)# destination interface Ethernet 1/39`	イーサネット SPAN 宛先ポートを設定します。
ステップ 6	**no shut** 例： `switch(config-monitor)# no shut`	SPAN セッションをイネーブルにします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 7	（任意） **show monitor session** *session* 例： `switch(config-monitor)# show monitor session 5`	SPAN 設定を表示します。
ステップ 8	**copy running-config startup-config** 例： `switch(config-monitor)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

異なる LSE スライス間のマルチキャスト Tx トラフィックの SPAN の設定

Cisco NX-OS Release 7.0(3)I7(1) 以降では、Cisco Nexus 9300-EX プラットフォーム スイッチ上の異なるリーフ スパイン エンジン（LSE）スライス間で、マルチキャスト Tx トラフィックの SPAN を設定できます。

手順の概要

1. **configure terminal**

2. [no] **hardware multicast global-tx-span**
3. **copy running-config startup-config**
4. **reload**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[no] **hardware multicast global-tx-span** 例： `switch(config)# hardware multicast global-tx-span`	異なるリーフ スパイン エンジン（LSE）スライス間のマルチキャスト Tx トラフィックの SPAN を設定します。 （注） Cisco NX-OS リリース 10.2(2)F 以降、送信元と接続先が異なるスライス上にある場合は、マルチキャスト SPAN Tx にこのコマンドを使用します。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。
ステップ 4	**reload** 例： `switch(config)# reload`	デバイスがリロードされます。

SPAN から CPU への構成

はじめに

SPAN-to-CPU は、Cisco Nexus 9000 シリーズ スイッチを通過するパケット フローのトラブルシューティングを行うためのものです。通常の SPAN または Encapsulated Remote SPAN（ERSPAN）セッションと同様に、SPAN-to-CPU モニタ セッションには、1 つ以上の送信元インターフェイスとトラフィック方向の定義が含まれます。ソースインターフェイスで定義された方向（TX、RX、またはその両方）に一致するトラフィックはすべて、スーパーバイザCPUに複製されます。このトラフィックはフィルタリングされ、ethanalyzer を使用して分析されるか、結果を確認するためにローカル ストレージ デバイスに保存されます。

Cisco Nexus 9000 シリーズ スイッチの CPU によって生成されたパケットが特定のインターフェイスから送信されているかどうかを確認するには、インターフェイスに接続されているリモート デバイスでパケット キャプチャ ユーティリティを使用することをお勧めします。

1. **CPU 接続先として SPAN を構成する**

   モニタ セッションの接続先として CPU を構成できることが必要であり、ハードウェアで同じように構成する必要があります。Tahoe プラットフォームでは、顧客が ERSPAN 終端セッションでサポートする必要がないため、この設定はローカル スパンに対してのみサポートされます。N9K-C9508-FM-R2 でも同様にサポートされます。

2. **SPAN トラフィックの分析**

   SPAN トラフィックが前述のスーパーバイザ CPU に到達したとき：モジュールは SPAN パケットとして識別し、必要なアクションを実行し、ethanalyzer がこれらのパケットを表示します。Ethanalyzer コントロールプレーン パケット キャプチャ ユーティリティを使用して、CPU に複製されたトラフィックを表示できます。Ethanalyzer コマンドの mirror キーワードは、SPAN-to-CPU モニタ セッションによって複製されたトラフィックのみが表示されるようにトラフィックをフィルタリングします。Ethanalyzer のキャプチャおよび表示フィルタを使用して、表示されるトラフィックをさらに制限できます。

3. **SPAN トラフィック レートの制限**

   コントロールプレーンの中断を避けるために、CPU のスパンド トラフィックをレート制限する必要があります。Ethanalyzer は、パケット ヘッダーの処理、ストリッピング、およびデコードに libpcap モジュールを使用します。Ethanalyzer はミラー オプションを使用して、スーパーバイザ CPU に到達するスパン トラフィックを表示します。SPAN と CPU のマッチングのため、別のスパン クラスが作成されます。すべてのトラフィックは SPAN クラスとして作成され、このクラスにはコントロールプレーン ポリシング（COPP）として個別のレートが作成されます。COPP のトラフィック レートは 50 kbps に制限されます。

4. **ACL フィルタ処理**

   これにより、顧客は監視するトラフィックを選択できます。この機能は、あらゆる種類のモニタ セッションでサポートされます。トラフィックのレートは制限されるため、スパンから CPU の場合、これは特に重要です。スパンされることを意図してトラフィックを分類することが重要になります。

注意事項と制約事項

SPAN-to-CPU に関する設定時の注意事項および制約事項は、次のとおりです。

- インバンド送信元では ACL フィルタ処理はサポートされていません。
- 物理インターフェイス（L2 および L3）、ポート チャネル、L3 サブインターフェイスなどの送信元は、ACL フィルタでサポートされます。
- ACL フィルタは、Rx 送信元のみに対してサポートされます。
- VLAN 送信元では ACL フィルタ処理はサポートされていません。

- 同じソースに対して複数のスパンセッションを構成することはサポートされていません。
- MTU切り捨ては、N9K-X9636C-R、N9K-X9636Q-R、N9K-X9636C-RX、N9K-X96136YC-R、N9K-X9624D-R2、N9K-C9508-FM-R、N9K-C9504-FM-R、N9K-C9508-FM-R2、N9K-C9504-FM-R2、N3K-C36180YC-R、N3K-C3636C-R、およびN3K-C36480LD-R2 ではサポートされていません。
- ACL フィルタは、Cisco NX-OS リリース 10.2(2)F までは、N9K-X9624D-R2 ラインカードではサポートされていません。
- Cisco NX-OSリリース10.2（3）以降では、N9K-X9624D-R2 ライン カードで ACL フィルタがサポートされます。

SPAN から CPU への構成

CPU への SPAN を構成できます。

手順

	コマンドまたはアクション	目的

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

ステップ 2	**configure CPU as SPAN** 例： `switch(config-monitor)# destination interface sup-eth0`	CPU を SPAN 接続先として構成します。
ステップ 3	**configure ACL Filter** 例： `switch(config-monitor)# filter access-group <acl_filter_name>`	フィルタ処理に使用されるアクセス リストを構成します。
ステップ 4	**configure ethanalyzer** 例： `switch# ethanalyzer local interface inband mirror`	スパンされるパケットを表示します。

例

この例は、モニタ セッションの出力を示しています。

```
show monitor session 1 session 1
type : local
state : up
acl-name : acl-name not specified
```

```
source intf :
rx : Eth3/44
tx : Eth3/44
both : Eth3/44
source VLANs :
rx :
tx :
both :
filter VLANs : filter not specified
source fwd drops :
destination ports : sup-eth0
PFC On Interfaces :
source VSANs :
rx :
```

この例は、copp の出力を示しています。

```
# show policy-map interface control-plane | begin span
class-map copp-system-p-class-span (match-any)
match exception span
set cos 0
police cir 50 pps , bc 256 packets
module 1 : <Designated Module>
conformed 910228778 bytes;
7217965 packets;
violated 7217965 bytes;
0 packets;
module 3 :
conformed 0 bytes;
0 packets;
violated 0 bytes;
0 packets;
0 packets;
```

SPAN セッションのシャットダウンまたは再開

SPAN セッションをシャットダウンすると、送信元から宛先へのパケットのコピーを切断できます。1セッションをシャットダウンしてハードウェア リソースを解放し、別のセッションを有効にできます。デフォルトでは、SPAN セッションはシャット ステートで作成されます。

SPAN セッションを再開（イネーブルに）すると、送信元から宛先へのパケットのコピーを再開できます。すでにイネーブルになっていて、動作状況がダウンの SPAN セッションをイネーブルにするには、そのセッションをいったんシャットダウンしてから、改めてイネーブルにする必要があります。

SPAN セッションのシャット ステートおよびイネーブル ステートは、グローバルまたはモニタ コンフィギュレーション モードのどちらのコマンドでも設定できます。

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **monitor session** {*session-range* | **all**} **shut**
3. **monitor session** *session-number*
4. [**no**] **shut**
5. (Optional) **show monitor**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **monitor session** {*session-range* | **all**} **shut** **Example:** `switch(config)# monitor session 3 shut`	指定の SPAN セッションをシャットダウンします。デフォルトでは、セッションはシャット ステートで作成されます。 コマンドの **no** 形式は、指定された SPAN セッションを再開（イネーブルに）します。デフォルトでは、セッションはシャット ステートで作成されます。 **Note** モニタ セッションが有効で動作状況がダウンの場合、セッションを有効にするには、最初に **monitor session shut** コマンドを指定してから、**no monitor session shut** コマンドを続ける必要があります。

ステップ 3	**monitor session** *session-number*
	Example:
	`switch(config)# monitor session 3`
	`switch(config-monitor)#`
		モニタ コンフィギュレーション モードを開始します。新しいセッション コンフィギュレーションは、既存のセッション コンフィギュレーションに追加されます。

ステップ 4	[**no**] **shut** **Example:** `switch(config-monitor)# shut`	SPAN セッションをシャットダウンします。デフォルトでは、セッションはシャット ステートで作成されます。 コマンドの **no** 形式は SPAN セッションを有効にします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 5	(Optional) **show monitor** **Example:** `switch(config-monitor)# show monitor`	SPAN セッションのステータスを表示します。
ステップ 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SPAN 設定の確認

SPAN 設定を表示するには、次のいずれかの作業を行います。

コマンド	目的
show monitor session {**all** | *session-number* | **range** *session-range*} [**brief**]	SPAN セッションの設定を表示します。
show monitor session [*session-id* | **all**] **stats**	Cisco Nexus 9808 および 9804 プラットフォーム スイッチの SPAN セッション統計を表示します。
clear monitor session [*session-id* | **all**] **stats** [**both** | **rx** | **tx**]	Cisco Nexus 9808 および 9804 プラットフォーム スイッチの SPAN セッション統計をクリアします。

SPAN のコンフィギュレーション例

SPAN セッションのコンフィギュレーション例

SPAN セッションを設定する手順は、次のとおりです。

手順の概要

1. アクセス モードで宛先ポートを設定し、SPAN モニタリングをイネーブルにします。
2. SPAN セッションを設定します。

手順の詳細

手順

ステップ 1 アクセス モードで宛先ポートを設定し、SPAN モニタリングをイネーブルにします。

例：

```
switch# configure terminal
switch(config)# interface ethernet 2/5
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# no shut
switch(config-if)# exit
switch(config)#
```

ステップ 2 SPAN セッションを設定します。

例：

```
switch(config)# no monitor session 3
switch(config)# monitor session 3
switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx
switch(config-monitor)# source interface port-channel 2
switch(config-monitor)# source interface sup-eth 0 both
switch(config-monitor)# source vlan 3, 6-8 rx
switch(config-monitor)# source interface ethernet 101/1/1-3
switch(config-monitor)# filter vlan 3-5, 7
switch(config-monitor)# destination interface ethernet 2/5
switch(config-monitor)# no shut
switch(config-monitor)# exit
switch(config)# show monitor session 3
switch(config)# copy running-config startup-config
```

例：

```
switch(config)# monitor session 1
switch(config-monitor)# source interface fc 1/9/1
switch(config-monitor)# source interface san-port-channel 171
switch(config-monitor)# source vsan 3701
switch(config-monitor)# destination interface ethernet 1/8
switch(config-monitor)# no shutdown
switch(config-monitor)# exit
switch(config)# show monitor session 1
switch(config)# copy running-config startup-config
```

単一方向 SPAN セッションの設定例

単一方向 SPAN セッションを設定するには、次の手順を実行します。

手順の概要

1. アクセス モードで宛先ポートを設定し、SPAN モニタリングをイネーブルにします。
2. SPAN セッションを設定します。

手順の詳細

手順

ステップ 1 アクセス モードで宛先ポートを設定し、SPAN モニタリングをイネーブルにします。

例：

```
switch# configure terminal
switch(config)# interface ethernet 2/5
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# no shut
switch(config-if)# exit
switch(config)#
```

ステップ 2 SPAN セッションを設定します。

例：

```
switch(config)# no monitor session 3
switch(config)# monitor session 3 rx
switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx
switch(config-monitor)# filter vlan 3-5, 7
switch(config-monitor)# destination interface ethernet 2/5
switch(config-monitor)# no shut
switch(config-monitor)# exit
switch(config)# show monitor session 3
switch(config)# copy running-config startup-config
```

SPAN ACL の設定例

次に、SPAN ACL を構成する例を示します。

```
switch# configure terminal
switch(config)# ip access-list match_11_pkts
switch(config-acl)# permit ip 11.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# ip access-list match_12_pkts
switch(config-acl)# permit ip 12.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# vlan access-map span_filter 5
switch(config-access-map)# match ip address match_11_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# vlan access-map span_filter 10
switch(config-access-map)# match ip address match_12_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# monitor session 1
switch(config-erspan-src)# filter access_group span_filter
```

UDF ベース SPAN の設定例

次に、以下の一致基準を使用して、カプセル化された IP-in-IP パケットの内部 TCP フラグで照合する UDF ベース SPAN を設定する例を示します。

- 外部送信元 IP アドレス：10.0.0.2
- 内部 TCP フラグ：緊急 TCP フラグを設定
- バイト：Eth Hdr（14）＋外部 IP（20）＋内部 IP（20）＋内部 TCP（20、ただし、13 番目のバイトの TCP フラグ）
- パケットの先頭からのオフセット：14 + 20 + 20 + 13 = 67
- UDF の照合値：0x20
- UDF マスク：0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify ing-l3-span-filter
copy running-config startup-config
reload
ip access-list acl-udf
permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1
source interface Ethernet 1/1
filter access-group acl-udf
```

次に、以下の一致基準を使用して、レイヤ 4 ヘッダーの先頭から 6 バイト目のパケット署名（DEADBEEF）と通常の IP パケットを照合する UDF ベース SPAN を設定する例を示します。

- 外部送信元 IP アドレス：10.0.0.2
- 内部 TCP フラグ：緊急 TCP フラグを設定
- バイト：Eth Hdr（14）＋IP（20）＋TCP（20）＋ペイロード：112233445566DEADBEEF7788
- レイヤ 4 ヘッダーの先頭からのオフセット：20 + 6 = 26
- UDF の照合値：0xDEADBEEF（2 バイトのチャンクおよび 2 つの UDF に分割）
- UDF マスク：0xFFFFFFFF

```
udf udf_pktsig_msb header outer l4 26 2
udf udf_pktsig_lsb header outer l4 28 2
hardware access-list tcam region racl qualify ing-l3-span-filter
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1
source interface Ethernet 1/1
filter access-group acl-udf-pktsig
```

SPAN 切り捨ての設定例

この例では、MPLS ストリッピングで使用する SPAN 切り捨てを設定する方法を示します。

```
mpls strip
ip access-list mpls
statistics per-entry
20 permit ip any any redirect Ethernet1/5
interface Ethernet1/5
switchport
switchport mode trunk
mtu 9216
no shutdown
monitor session 1
source interface Ethernet1/5 tx
mtu 64
destination interface Ethernet1/6
  no shut
```

LSE スライス間のマルチキャスト Tx SPAN の設定例

次に、Cisco Nexus 9300-EX プラットフォーム スイッチの LSE スライス間でマルチキャスト Tx SPAN を設定する例を示します。また、マルチキャスト Tx SPAN の設定前後の出力例を示します。

マルチキャスト Tx SPAN の設定前

```
switch# show interface eth1/15-16, ethernet 1/27 counters

--------------------------------------
Port           InOctets    InUcastPkts
--------------------------------------
Eth1/15          580928              0
Eth1/16             239              0
Eth1/27               0              0

--------------------------------------
Port        InMcastPkts    InBcastPkts
--------------------------------------
Eth1/15            9077              0
Eth1/16               1              0
Eth1/27               0              0

--------------------------------------
Port          OutOctets   OutUcastPkts
--------------------------------------
Eth1/15             453              0
Eth1/16          581317              0
Eth1/27               0              0

--------------------------------------
Port       OutMcastPkts   OutBcastPkts
--------------------------------------
Eth1/15               4              0
Eth1/16            9080              0
Eth1/27               0              0
```

マルチキャスト Tx SPAN の設定

```
switch(config)# hardware multicast global-tx-span
Warning: Global Tx SPAN setting changed, please save config and reload
switch(config)# copy running-config start-up config
[########################################] 100%
Copy complete.
switch(config)# reload
This command will reboot the system. (y/n)?  [n] y
```

マルチキャスト Tx SPAN の設定後

```
switch# show interface eth1/15-16, eth1/27 counters

--------------------------------------
Port           InOctets    InUcastPkts
--------------------------------------
Eth1/15          392576              0
Eth1/16               0              0
Eth1/27               0              0
--------------------------------------
Port        InMcastPkts    InBcastPkts
--------------------------------------
```

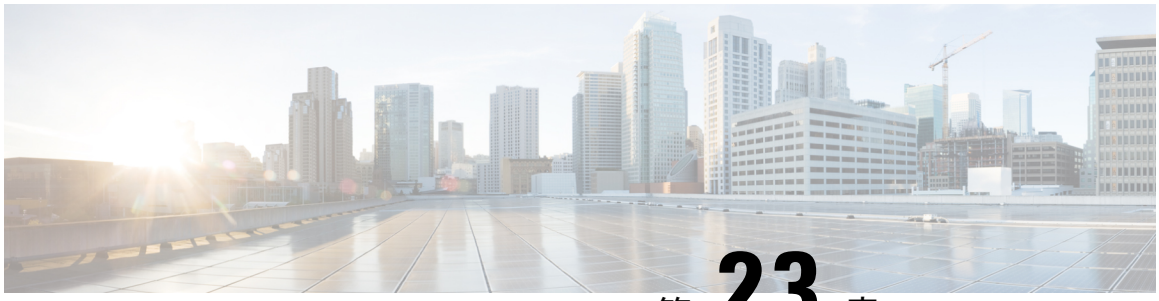
```
Eth1/15            6134              0
Eth1/16               0              0
Eth1/27               0              0

----------------------------------------
Port          OutOctets   OutUcastPkts
----------------------------------------
Eth1/15               0              0
Eth1/16          392644              0
Eth1/27          417112              0

----------------------------------------
Port       OutMcastPkts   OutBcastPkts
----------------------------------------
Eth1/15               0              0
Eth1/16            6135              0
Eth1/27            6134              0
```

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
FEX	『*Cisco Nexus 2000 Series NX-OS Fabric Extender Software Configuration Guide for Cisco Nexus 9000 Series Switches*』

関連資料



第 23 章

ERSPAN の設定

この章は、カプセル化リモート スイッチド ポート アナライザ（ERSPAN）を Cisco NX-OS デバイスの IP ネットワークでミラーリングされたトラフィックを転送するように設定する方法について説明します。

- ERSPAN について （451 ページ）
- ERSPAN の前提条件 （453 ページ）
- ERSPAN の注意事項および制約事項 （453 ページ）
- デフォルト設定 （459 ページ）
- ERSPAN の設定 （459 ページ）
- ERSPAN 設定の確認 （476 ページ）
- ERSPAN の設定例 （477 ページ）

ERSPAN について

ERSPAN は、IPv4 または IPv6 ネットワークでミラーリングされたトラフィックを転送して、ネットワーク内で複数のスイッチのリモートモニタリングを提供します。トラフィックは、送信元ルータでカプセル化され、ネットワーク間を転送されます。パケットは宛先ルータでカプセル化解除され、宛先インターフェイスに送信されます。もう 1 つの方法は、パケットを解析して内部（SPAN コピー）フレームにアクセスするために、ERSPAN カプセル化形式を理解する必要があるアナライザ自体を宛先とする方法です。

ERSPAN 送信元

トラフィックをモニタできるモニタ元インターフェイスのことを ERSPAN 送信元と呼びます。送信元では、監視するトラフィックを指定し、さらに入力、出力、または両方向のトラフィックをコピーするかどうかを指定します。ERSPAN 送信元には次のものが含まれます。

- イーサネット ポート（ただしサブインターフェイスではない）
- ポート チャネル
- コントロール プレーン CPU への帯域内インターフェイス。

（注） SPAN 送信元としてスーパーバイザ インバンド インターフェイスを指定すると、デバイスはスーパーバイザ CPU により送信されたすべてのパケットをモニタします。

（注） スーパーバイザ インバンド インターフェイスを SPAN 送信元として使用する場合、スーパーバイザ ハードウェア（出力）によって生成されたすべてのパケットがモニタされます。

Rx は ASIC の観点から見たものです（トラフィックはインバンドを介してスーパーバイザから出力され、ASIC / SPAN で受信されます）。

- VLAN
  - VLAN が ERSPAN 送信元として指定されている場合は、VLAN 内でサポートされているすべてのインターフェイスが ERSPAN 送信元になります。
  - VLAN は、Cisco Nexus 9300-EX/-FX/-FX2/-FX3/-GX シリーズ プラットフォーム スイッチおよび -EX/-FX ライン カードを備えた Cisco Nexus 9500 シリーズ プラットフォーム スイッチを除き、入力方向でのみ ERSPAN 送信元にすることができます。

（注） 1 つの ERSPAN セッションに、上述の送信元を組み合わせて使用できます。

ERSPAN の宛先

宛先ポートは ERSPAN 送信元からコピーされたトラフィックを受信します。宛先ポートは、リモート モニタリング（RMON）プローブなどのデバイス、あるいはコピーされたパケットを 1 つまたは複数の送信元ポートから受信したり、解析することができるセキュリティ デバイスに接続されたポートです。宛先ポートはスパニング ツリー インスタンスまたはレイヤ 3 プロトコルに参加しません。

Cisco Nexus 9200、9300-EX、9300-FX、および 9300-FX2 プラットフォーム スイッチは、GRE ヘッダー トラフィック フローを使用して、スイッチポート モードの物理インターフェイスまたはポートチャネル インターフェイスで設定された ERSPAN 宛先セッションをサポートします。送信元 IP アドレスは、デフォルト VRF で設定する必要があります。複数の ERSPAN 宛先セッションを同じ送信元 IP アドレスで設定する必要があります。

ERSPAN セッション

モニタする送信元を指定する ERSPAN セッションを作成できます。

ローカライズされた ERSPAN セッション

すべての送信元インターフェイスが同じライン カード上にある場合、ERSPAN セッションはローカライズされます。

（注） VLAN 送信元の ERSPAN セッションはローカライズされません

ERSPAN の切り捨て

Cisco NX-OS Release 7.0(3)I7(1) 以降では、MTU のサイズに基づいて各 ERSPAN セッションの送信元パケットの切り捨てを設定できます。切り捨てにより、モニタするパケットのサイズを減らすことで、ERSPAN の帯域幅を効果的に軽減できます。設定された MTU サイズよりも大きい ERSPAN パケットはすべて、設定されたサイズに切り捨てられます。ERSPAN では、ERSPAN ヘッダータイプに応じて、切り捨てられたパケットに 54 ー 166 バイトの ERSPAN ヘッダーが追加されます。たとえば、MTU を 300 バイトに設定すると、ERSPAN ヘッダー タイプの設定に応じて、パケットは 354 ー 466 バイトの ERSPAN ヘッダーサイズで複製されます。

ERSPAN切り捨てはデフォルトでは無効です。切り捨てを使用するには、個々のERSPANセッションで有効にしておく必要があります。

ERSPAN の前提条件

ERSPAN の前提条件は、次のとおりです。

- 各デバイス上で、まず所定の ERSPAN 設定をサポートするポートを設定する必要があります。詳細については、『Cisco Nexus 9000 シリーズ NX-OS インターフェイス設定ガイド』を参照してください。

ERSPAN の注意事項および制約事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

ERSPAN 設定時の注意事項と制限事項は次のとおりです。

- ERSPAN セッション（Rx および Tx、Rx、または Tx）ごとに最大 48 の送信元インターフェイスがサポートされます。

- ERSPAN 宛先は、プラットフォームに基づいて MTU のジャンボ フレームを異なる方法で処理します。次の Cisco Nexus 9300 プラットフォーム スイッチおよびサポート ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチの場合、ERSPAN 宛先はジャンボ フレームをドロップします。
  - Cisco Nexus 9332PQ
  - Cisco Nexus 9372PX
  - Cisco Nexus 9372PX-E
  - Cisco Nexus 9372TX
  - Cisco Nexus 9372TX-E
  - Cisco Nexus 93120TX
  - 次のライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチ
    - Cisco Nexus 9564PX
    - Cisco Nexus 9464TX
    - Cisco Nexus 9464TX2
    - Cisco Nexus 9564TX
    - Cisco Nexus 9464PX
    - Cisco Nexus 9536PQ
    - Cisco Nexus 9636PQ
    - Cisco Nexus 9432PQ

  次の Cisco Nexus 9200 プラットフォーム スイッチおよびサポート ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチの場合、ERSPAN はポート MTU でパケットを切り捨て、TX 出力エラーを発行します。
  - Cisco Nexus 92160YC-X
  - Cisco Nexus 92304QC
  - Cisco Nexus 9272Q
  - Cisco Nexus 9232C
  - Cisco Nexus 9236C
  - Cisco Nexus 92300YC
  - Cisco Nexus 93108TC-EX
  - Cisco Nexus 93180LC-EX
  - Cisco Nexus 93180YC-EX
  - 次のライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチ
    - Cisco Nexus 9736C-EX

- Cisco Nexus 97160YC-EX
- Cisco Nexus 9732C-EX
- Cisco Nexus 9732C-EXM

- ACL フィルタを使用した、親インターフェイスでの ERSPAN サブインターフェイストラフィックは、Cisco Nexus 9200 プラットフォームスイッチではサポートされません。
- ACL フィルタを使用した、親インターフェイスでの ERSPAN サブインターフェイストラフィックは、Cisco Nexus 9300-EX/FX/FX2/FX3/GX プラットフォームスイッチではサポートされません。
- ERSPAN ミラーリングは、PBR トラフィックではサポートされません。
- タイプ 3 ヘッダをもつ ERSPAN は、Cisco NX-OS リリース 9.3(3) ではサポートされません。
- ERSPAN セッションの制限については、『*Cisco Nexus 9000* シリーズ *NX-OS* 検証スケーラビリティ ガイド』を参照してください。
- ラインカードごとの ERSPAN セッションの数は、同じインターフェイスが複数セッションの双方向送信元として設定されている場合は、2 に減少します。
- 同じ送信元インターフェイスで 2 つの SPAN または ERSPAN セッションを 1 つのフィルタだけで設定することはできません。同じ送信元が複数の SPAN または ERSPAN セッションで使用されている場合は、すべてのセッションに異なるフィルタを設定するか、セッションにフィルタを設定しないでください。
- Cisco NX-OS リリース 9.3(5) 以降、次の ERSPAN 機能は Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされています。
  - ERSPAN タイプ III ヘッダー
  - ERSPAN 宛先サポート
- FCS エラーがあるパケットは、ERSPAN セッションでミラーリングされません。
- TCAM カービングは、次のライン カードの SPAN/ERSPAN には必要ありません。
  - Cisco Nexus 9636C-R
  - Cisco Nexus 9636Q-R
  - Cisco Nexus 9636C-RX
  - Cisco Nexus 96136YC-R
  - Cisco Nexus 9624D-R2

（注） SPAN/ERSPAN をサポートする他のすべてのスイッチは、TCAM カービングを使用する必要があります。

- フィルタ アクセス グループの統計情報はサポートされていません。
- ERSPAN セッションのアクセス グループ フィルタは、vlan-accessmap として設定する必要があります。
- スーパーバイザによって生成されたコントロール プレーン パケットは、ERSPAN カプセル化または ERSPAN アクセス コントロール リスト（ACL）によるフィルタ処理をすることはできません。
- ERSPAN は、管理ポートではサポートされません。
- ERSPAN は、レイヤ 3 ポート チャネル サブインターフェイスの宛先をサポートしません。
- 送信元としての VLAN は、R シリーズ ライン カードおよび N3K-C36180YC-R、N3KC36480LD-R2、および N3K-C3636C-R プラットフォーム スイッチの ERSPAN 設定ではサポートされません。
- VLAN は、ERSPAN 送信元またはフィルタとして使用される場合、属することができるのは 1 つのセッションだけです。
- VLAN ERSPAN がモニタするのは、VLAN のレイヤ 2 ポートを出入りするトラフィックだけです。
- vPC で ERSPAN をイネーブルにし、ERSPAN パケットが vPC を介して宛先にルーティングされなければならない場合は、vPC ピア リンクを通過するパケットはキャプチャできません。
- ERSPAN は、VXLAN オーバーレイではサポートされません。
- マルチキャスト パケットの ERSPAN コピーは、書き換え前に作成されます。したがって、TTL、VLAN ID、出力ポリシーによる再マーキングなどは ERSPAN コピーにキャプチャされません。
- ERSPAN タイプ III セッションのタイムスタンプの粒度は、CLI では設定できません。100 ピコ秒で、PTP を介して駆動されます。
- ERSPAN はデフォルトおよびデフォルト以外の VRF で動作しますが、ERSPAN マーカー パケットはデフォルト VRF でのみ動作します。
- 同じ送信元は、複数のセッションの一部にすることができます。

次の注意事項と制約事項が （Tx）ERSPAN に適用されます。

- 不明ユニキャストでフラッディングされたパケットのルーティング後のフローは ERSPAN セッションに置かれますが、これはフローが転送されるポートをモニタしないよう ERSPAN セッションが設定されている場合であっても同様です。この制限は、ネットワーク フォ

ワーディング エンジン（NFE）と NFE2 対応 EOR スイッチおよび ERSPAN セッションで Tx ポートの送信元を持つものに適用されます。

- レイヤ 2 の ERSPAN Tx マルチキャストの場合、ERSPAN コピーはマルチキャスト レプリケーションとは無関係に作成されます。このため、マルチキャストと SPAN パケットでは、VLAN タグ（入力インターフェイス VLAN ID）の値が異なります。
- 次の注意事項と制約事項が （Rx）ERSPAN に適用されます。
  - VLAN 送信元は Rx 方向のみがサポートされます。
  - セッション フィルタリング機能（VLAN または ACL フィルタ）は、Rx 送信元でのみサポートされます。
  - VLAN は、ERSPAN 送信元として入力方向でのみサポートされます。
- プライオリティ フロー制御（PFC）ERSPAN には、次の制約事項と制約事項があります。
  - フィルタとは共存できません。
  - 物理または port-channel インターフェイスの Rx 方向でのみサポートされています。VLAN インターフェイスの Rx 方向、または Tx 方向ではサポートされていません。
- 次の注意事項および制約事項が FEX ポートに適用されます。
  - 双方向 ERSPAN セッションで使用される送信元が同じ FEX からのものである場合、ハードウェア リソースは 2 つの ERSPAN セッションに制限されます。
  - FEX ポートは、ERSPAN としてすべてのトラフィックに対して入力方向でサポートされ、既知のレイヤ 2 ユニキャスト トラフィックには出力方向のみがサポートされます。
  - Cisco Nexus 9300 プラットフォーム スイッチは、FEX インターフェイスに接続されている ERSPAN 宛先をサポートしていません。ERSPAN 宛先は、前面パネル ポートに接続する必要があります。
  - VLAN および ACL フィルタは FEX ポートではサポートされません。フィルタとは共存できません。
- ERSPAN 宛先には、次の注意事項と制約事項が適用されます。
  - Cisco Nexus 9200、9300-EX、9300-FX、および 9300-FX2 プラットフォーム スイッチは、GRE ヘッダー トラフィック フローを使用して、スイッチポート モードの物理インターフェイスまたはポートチャネルインターフェイスで設定された ERSPAN 宛先セッションをサポートします。
  - ERSPAN 宛先は、Cisco Nexus 9200、9300、9300-EX、9300-FX、および 9300-FX2 プラットフォーム スイッチの MPLS や VXLAN などの他のトンネル機能と共存できません。
  - Cisco Nexus 9300-GX スイッチでは、ERSPAN 宛先セッションがアクティブであるデバイスを通過する dot1q タグ付きブロードキャストまたはマルチキャスト パケット

は、ハードウェアの制限により、正しい VLAN ではなくネイティブ VLAN でタグ付けされます。

- ERSPAN 宛先セッションは、デフォルトの VRF のみをサポートします。
- Cisco Nexus 9300-EX/FX スイッチは、Cisco Nexus 3000 および非 EX/FX Cisco Nexus 9000 スイッチの ERSPAN 宛先として機能できません。

- Cisco NX-OS リリース 10.1 (2) 以降、ERSPAN は Cisco Nexus N9K-X9624D-R2 ライン カードでサポートされます。
- IPv6 経由の ERSPAN 宛先には、次の注意事項と制約事項が適用されます。
  - Cisco NX-OS リリース 10.2(1)F 以降、IPv6 機能経由の ERSPAN は Cisco Nexus 9300-GX2、9300-GX、9300-FXP、9300-FX2、9300-EX、9300-FX3、9300-FX3S、および 9300-FX3P プラットフォーム スイッチ、 N9K-X9716D-GX、N9K-X9736C-EX、N9K-X9732C-EX（X86_64 Atom）、N9K-X9732C-EXM、N9K-X97160YC-EX、および N9K-X9736C-FX ライン カードでサポートされています。
  - この機能は、出力ポート チャネル メンバーと出力 ECMP パス間のロード バランシングではサポートされません。
  - この機能は、ヘッダータイプ 3、フィルタ ACL の udf、およびマーカー パケットではサポートされません。
  - この機能は、IPv6 の ERSPAN 送信元としての FEX ホスト インターフェイスではサポートされません。

- Cisco NX-OS リリース 10.2(3)F 以降、IPv6 機能経由の ERSPAN 接続先/終端先は Cisco Nexus 9300-GX2、9300-GX、9300-FXP、9300-FX2、9300-EX、9300-FX3、9300-FX3S、および 9300-FX3P プラットフォーム スイッチ、 N9K-X9716D-GX、N9K-X9736C-EX、N9K-X9732C-EX（X86_64 Atom）、N9K-X9732C-EXM、N9K-X97160YC-EX、および N9K-X9736C-FX ライン カードでサポートされています。
- 次の注意事項と制限事項が適用されます。
  - VRF デフォルトのみがサポートされています。
  - スイッチごとに設定できる IPv6 アドレスは 1 つだけです。
  - この機能は、ほかのトンネル機能ではサポートされていません。
  - 一度に 4 つの ERSPAN 宛先セッションを起動できます。
  - ERSPAN ID はセッションごとに一意で、範囲は 1～32 です。

- Cisco NX-OS リリース 10.3(1)F 以降、Cisco Nexus 9808 プラットフォーム スイッチで ERSPAN のサポートが提供されます。
  - ERSPAN では RX のみがサポートされています。
  - タイプ 3 ヘッダーはサポートされていません。

- ERSPAN 接続先/終端 はサポートされていません。

- Cisco NX-OS リリース 10.4(1)F 以降、ERSPAN は次のスイッチおよびライン カードでサポートされます。
  - Cisco Nexus 9332D-H2R スイッチ
  - Cisco Nexus 9804 スイッチ
    - タイプ 3 ヘッダーはサポートされていません
    - ERSPAN 接続先/終端 はサポートされていません
  - Cisco Nexus X98900CD-A および X9836DM-A ラインカードと Cisco Nexus 9808 および 9804 スイッチ

デフォルト設定

次の表に、ERSPAN パラメータのデフォルト設定を示します。

表 *20* : デフォルトの *ERSPAN* パラメータ

パラメータ	デフォルト
ERSPAN セッション	シャット ステートで作成されます

ERSPAN の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

ERSPAN 送信元セッションの設定

ERSPANセッションを設定できるのはローカルデバイス上だけです。デフォルトでは、ERSPAN セッションはシャット ステートで作成されます。

（注） ERSPAN は送信元に関係なく、スーパーバイザによって生成されるパケットをモニタしません。

手順の概要

1. **configure terminal**
2. **monitor erspan origin ip-address** *ip-address* **global** or **monitor erspan origin ipv6-address** *ipv6-address* **global**
3. **no monitor session** {*session-number* | **all**}
4. **monitor session** {*session-number* | **all**} **type erspan-source** [**shut**]
5. **description** *description*
6. **source** {**interface** *type* [**tx** | **rx** |**both**] **vlan** {**number** | **range**} [**rx**]}
7. （任意） ステップ 7 を繰り返して、すべての ERSPAN 送信元を設定します。
8. **filter vlan** {*number* | *range*}
9. （任意） ステップ 9 を繰り返して、すべての送信元 VLAN のフィルタリングを設定します。
10. （任意） **filter access-group** *acl-filter*
11. **destination ip** *ip-address*
12. **erspan-id** *erspan-id*
13. **vrf** *vrf-name*
14. （任意） **ip ttl** *ttl-number*
15. （任意） **ip dscp** *dscp-number*
16. **no shut**
17. **exit**
18. （任意） **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**]
19. （任意） **show running-config monitor**
20. （任意） **show startup-config monitor**
21. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**monitor erspan origin ip-address** *ip-address* **global** or **monitor erspan origin ipv6-address** *ipv6-address* **global**
	例：
	`switch(config)# monitor erspan origin ip-address 10.0.0.1 global`
	`switch(config)# monitor erspan origin ipv6-address 2001:DB8:1::1 global`
		ERSPAN のグローバルな送信元 IPv4 または IPv6 アドレスを設定します。

	コマンドまたはアクション	目的
ステップ 3	**no monitor session** {*session-number* | **all**} 例： `switch(config)# no monitor session 3`	指定したERSPANセッションの設定を消去します。新しいセッション コンフィギュレーションは、既存のセッション コンフィギュレーションに追加されます。

ステップ 4	**monitor session** {*session-number* | **all**} **type erspan-source** [**shut**]
	例：
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		ERSPAN タイプ II 送信元セッションを設定します。デフォルトでは、セッションは双方向です。オプションの shut キーワードは、選択したセッションに対して shut ステートを指定します。

ステップ 5	**description** *description* 例： `switch(config-erspan-src)# description erspan_src_session_3`	セッションの説明を設定します。デフォルトでは、説明は定義されません。説明には最大 32 の英数字を使用できます。

ステップ 6	**source** {**interface** *type* [**tx** | **rx** |**both**] **vlan** {**number** | **range**} [**rx**]}
	例：
	`switch(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	例：
	`switch(config-erspan-src)# source interface port-channel 2`
	例：
	`switch(config-erspan-src)# source interface sup-eth 0 rx`
	例：
	`switch(config-erspan-src)# source vlan 3, 6-8 rx`
	例：
	`switch(config-erspan-src)# source interface ethernet 101/1/1-3`
		送信元およびパケットをコピーするトラフィックの方向を設定します。一定範囲のイーサネット ポート、ポート チャネル、インバンド インターフェイス、または一定範囲の VLAN、または Cisco Nexus 2000 シリーズ ファブリック エクステンダ（FEX）上のサテライト ポートまたはホスト インターフェイス ポート チャネルを入力できます。
		送信元は 1 つ設定することも、またはカンマで区切った一連のエントリとして、または番号の範囲として、複数設定することもできます。コピーするトラフィックの方向には、入力、出力、または両方を指定できます。
		単一方向のセッションには、送信元の方向はセッションで指定された方向に一致する必要があります。
		（注）
		送信元 VLAN は、入力方向でのみサポートされます。送信元FEXポートは、すべてのトラフィックに対して入力方向でサポートされ、既知のレイヤ 2 ユニキャスト トラフィックには出力方向のみがサポートされます。
		送信元としてのスーパーバイザは、Rx方向でのみサポートされます。

ステップ 7	（任意） ステップ 7 を繰り返して、すべての ERSPAN 送信元を設定します。	—

	コマンドまたはアクション	目的
ステップ 8	**filter vlan** {*number* | *range*} 例： `switch(config-erspan-src)# filter vlan 3-5, 7`	設定された送信元から選択する VLAN を設定します。VLAN は 1 つ設定することも、またはカンマで区切った一連のエントリとして、または番号の範囲として、複数設定することもできます。VLAN の範囲については、『Cisco Nexus 9000 シリーズ NX-OS レイヤ 2 スイッチング設定ガイド』を参照してください。 （注） ERSPAN 送信元として設定された FEX ポートは VLAN フィルタをサポートしません。
ステップ 9	（任意） ステップ 9 を繰り返して、すべての送信元 VLAN のフィルタリングを設定します。	—
ステップ 10	（任意） **filter access-group** *acl-filter* 例： `switch(config-erspan-src)# filter access-group ACL1`	ACL を ERSPAN セッションにアソシエートします。（標準の ACL 設定プロセスを使用して ACL を作成できます。詳細については、*Cisco Nexus 9000* シリーズ *NX-OS* セキュリティ コンフィギュレーション ガイドを参照してください。） （注） このコマンドを実行する前に、ip アクセス リストおよび関連する vlan アクセスマップ を構成します。ERSPAN ACL の構成を参照してください。

ステップ 11	**destination ip** *ip-address*
	例：
	`switch(config-erspan-src)# destination ip 10.1.1.1`
	`switch(config-erspan-src)# destination ipv6 2001:DB8:1::1`
		destination ipv6 *ipv6-address*
		ERSPAN セッションの宛先 IPv4 または IPv6 アドレスを設定します。
		（注）
		ERSPAN 送信元セッションごとに 1 つの宛先 IPv4 または IPv6 アドレスのみがサポートされます。

ステップ 12	**erspan-id** *erspan-id* 例： `switch(config-erspan-src)# erspan-id 5`	ERSPAN 送信元セッションの ERSPAN ID を設定します。ERSPAN の範囲は 1 ～ 1023 です。
ステップ 13	**vrf** *vrf-name* 例： `switch(config-erspan-src)# vrf default`	ERSPAN 送信元セッションがトラフィックの転送に使用する仮想ルーティングおよびフォワーディング（VRF）インスタンスを設定します。VRF 名は、32 文字以内の英数字のストリング（大文字と小文字を区別）で指定します。

	コマンドまたはアクション	目的
ステップ 14	（任意） **ip ttl** *ttl-number* 例： `switch(config-erspan-src)# ip ttl 25`	ERSPAN トラフィックの IP 存続可能時間（TTL）値を設定します。範囲は 1 ～ 255 です。
ステップ 15	（任意） **ip dscp** *dscp-number* 例： `switch(config-erspan-src)# ip dscp 42`	ERSPAN トラフィックのパケットの DiffServ コードポイント（DSCP）値を設定します。範囲は 0 ～ 63 です。
ステップ 16	**no shut** 例： `switch(config-erspan-src)# no shut`	ERSPAN送信元セッションをイネーブルにします。デフォルトでは、セッションはシャット ステートで作成されます。

ステップ 17	**exit**
	例：
	`switch(config-erspan-src)# exit`
	`switch(config)#`
		モニタ設定モードを閉じます。

ステップ 18	（任意） **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**] 例： `switch(config)# show monitor session 3`	ERSPAN セッション設定を表示します。
ステップ 19	（任意） **show running-config monitor** 例： `switch(config)# show running-config monitor`	ERSPAN の実行コンフィギュレーションを表示します。
ステップ 20	（任意） **show startup-config monitor** 例： `switch(config)# show startup-config monitor`	ERSPAN のスタートアップ コンフィギュレーションを表示します。
ステップ 21	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ERSPAN セッションのシャットダウンまたはアクティブ化

ERSPANセッションをシャットダウンすると、送信元から宛先へのパケットのコピーを切断できます。1セッションをシャットダウンしてハードウェア リソースを解放し、別のセッションを有効にできます。デフォルトでは、ERSPAN セッションはシャット ステートで作成されます。

ERSPANセッションをイネーブルにすると、送信元から宛先へのパケットのコピーをアクティブ化できます。すでにイネーブルになっていて、動作状況がダウンの ERSPAN セッションをイネーブルにするには、そのセッションをいったんシャットダウンしてから、改めてイネーブルにする必要があります。ERSPAN セッション ステートをシャットダウンおよびイネーブルにするには、グローバルまたはモニタ コンフィギュレーション モードのいずれかのコマンドを使用できます。

手順の概要

1. **configure terminal**
2. **monitor session** {*session-range* | **all**} **shut**
3. **no monitor session** {*session-range* | **all**} **shut**
4. **monitor session** *session-number* **type erspan-source**
5. **shut**
6. **no shut**
7. **exit**
8. （任意） **show monitor session all**
9. （任意） **show running-config monitor**
10. （任意） **show startup-config monitor**
11. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**monitor session** {*session-range* | **all**} **shut** 例： `switch(config)# monitor session 3 shut`	指定の ERSPAN セッションをシャットダウンします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 3	**no monitor session** {*session-range* | **all**} **shut** 例： `switch(config)# no monitor session 3 shut`	指定のERSPANセッションを再開（イネーブルに）します。デフォルトでは、セッションはシャット ステートで作成されます。 モニタ セッションがイネーブルで動作状況がダウンの場合、セッションをイネーブルにするには、最初に **monitor session shut** コマンドを指定してから、**no monitor session shut** コマンドを続ける必要があります。

	コマンドまたはアクション	目的

ステップ 4	**monitor session** *session-number* **type erspan-source**
	例：
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		ERSPAN 送信元タイプのモニタ コンフィギュレーション モードを開始します。新しいセッション コンフィギュレーションは、既存のセッション コンフィギュレーションに追加されます。

ステップ 5	**shut** 例： `switch(config-erspan-src)# shut`	ERSPAN セッションをシャットダウンします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 6	**no shut** 例： `switch(config-erspan-src)# no shut`	ERSPANセッションをイネーブルにします。デフォルトでは、セッションはシャット ステートで作成されます。

ステップ 7	**exit**
	例：
	`switch(config-erspan-src)# exit`
	`switch(config)#`
		モニタ設定モードを閉じます。

ステップ 8	（任意） **show monitor session all** 例： `switch(config)# show monitor session all`	ERSPAN セッションのステータスを表示します。
ステップ 9	（任意） **show running-config monitor** 例： `switch(config)# show running-config monitor`	ERSPAN の実行コンフィギュレーションを表示します。
ステップ 10	（任意） **show startup-config monitor** 例： `switch(config)# show startup-config monitor`	ERSPAN のスタートアップ コンフィギュレーションを表示します。
ステップ 11	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ERSPAN ACL の設定

デバイスに IPv4 ERSPAN ACL を作成して、ルールを追加できます。

始める前に

DSCP 値または GRE プロトコルを変更するには、新しい宛先モニタ セッションを割り当てる必要があります。最大 4 つの宛先モニタ セッションがサポートされます。

手順の概要

1. **configure terminal**
2. **ip access-list** *acl-name*
3. [*sequence-number*] {**permit** | **deny**} *protocol source destination* [**set-erspan-dscp** *dscp-value*] [**set-erspan-gre-proto** *protocol-value*]
4. **vlan access-map erpsan-acl** *map name* [*sequence-number*]
5. **match ip address** *acl-name*
6. **action forward**
7. **exit**
8. **monitor session** [*session-number*| **all**] **type erspan-source** [**shut**]
9. **filter access_group** *name*
10. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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 access-list** *acl-name*
	例：
	`switch(config)# ip access-list erspan-acl`
	`switch(config-acl)#`
		ERSPAN ACL を作成して、IP ACL コンフィギュレーション モードを開始します。*acl-name* 引数は 64 文字以内で指定します。

ステップ 3	[*sequence-number*] {**permit** | **deny**} *protocol source destination* [**set-erspan-dscp** *dscp-value*] [**set-erspan-gre-proto** *protocol-value*]
	例：
	`switch(config-acl)# permit ip 192.168.2.0/24 any`
	
	`set-erspan-dscp 40 set-erspan-gre-proto 5555`
	例：
	`switch(config)# ip access-list match_11_pkts`
	`switch(config-acl)# permit ip 10.0.0.0/24 any`
	`switch(config-acl)# exit`
		ERSPAN ACL 内にルールを作成します。多数のルールを作成できます。
		sequence-number 引数には、1 ～ 4294967295 の整数を指定します。
		permit コマンドと **deny** コマンドには、トラフィックを識別するための多くの方法が用意されています。
		set-erspan-dscpオプションは、ERSPAN 外部 IP ヘッダーに DSCP 値を設定します。DSCP 値の範囲は 0 ～ 63 です。ERSPAN ACL に設定された DSCP 値で、モニタ セッションに設定されている値が上書

	コマンドまたはアクション	目的
		きされます。ERSPAN ACL にこのオプションを含めない場合、0 またはモニタ セッションで設定されている DSCP 値が設定されます。 **set-erspan-gre-proto** オプションは、ERSPAN GRE ヘッダーにプロトコル値を設定します。プロトコル値の範囲は 0 ～ 65535 です。ERSPAN ACL にこのオプションを含めない場合、ERSPAN カプセル化パケットの GRE ヘッダーのプロトコルとしてデフォルト値の 0x88be が設定されます。 **set-erspan-gre-proto** または **set-erspan-dscp** アクションが設定されている各アクセス コントロール エントリ（ACE）は、1 つの宛先モニタ セッションを消費します。ERSPAN ACL ごとに、これらのアクションのいずれかが設定されている最大 3 つの ACE がサポートされます。たとえば、次のいずれかを設定できます。 • **set-erspan-gre-proto** または **set-erspan-dscp** アクションが設定された最大 3 つの ACE を持つ ACL が設定されている、1 つの ERSPAN セッション • **set-erspan-gre-proto** または **set-erspan-dscp** アクションが設定され、1 つの追加のローカルまたは ERSPAN セッションが設定された 2 つの ACE を持つ ACL が設定されている、1 つの ERSPAN セッション • **set-erspan-gre-proto** または **set-erspan-dscp** アクションが設定された 1 つの ACE を持つ ACL が設定されている、2 つの ERSPAN セッションのうち大きなもの
ステップ 4	**vlan access-map erpsan-acl** *map name* [*sequence-number*] 例： `switch(config)# vlan access-map erspan_filter`	指定した VLAN アクセス マップの VLAN アクセス マップ コンフィギュレーション モードを開始します。VLAN アクセス マップが存在しない場合は、デバイスによって作成されます。 シーケンス番号を指定しなかった場合、デバイスによって新しいエントリが作成され、このシーケンス番号はアクセス マップの最後のシーケンス番号よりも 10 大きい番号となります。

	コマンドまたはアクション	目的
ステップ 5	**match ip address** *acl-name* 例： `switch(config-access-map)# match ip address erspan-acl`	アクセス マップ エントリに ACL を指定します。
ステップ 6	**action forward** 例： `switch(config-access-map)# action forward`	ACL に一致したトラフィックにデバイスが適用する処理を指定します。
ステップ 7	**exit** 例： `switch(config-access-map)# exit`	VLAN アクセスマップ コンフィギュレーション モードを終了します。
ステップ 8	**monitor session** [*session-number*| **all**] **type erspan-source** [**shut**] 例： `switch(config)# monitor session 1 type erspan-source`	ERSPAN タイプ II 送信元セッションを設定します。デフォルトでは、セッションは双方向です。オプションの **shut** キーワードは、選択したセッションに対して shut ステートを指定します。
ステップ 9	**filter access_group** *name* 例： `switch(config-erspan-src)# filter access_group erspan_filter`	ACL を ERSPAN セッションにアソシエートします。（標準の ACL 設定プロセスを使用して ACL を作成できます。詳細については、*Cisco Nexus 9000* シリーズ *NX-OS* セキュリティ構成ガイドを参照してください。）
ステップ 10	（任意） **copy running-config startup-config** 例： `switch(config-acl)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ERSPAN ACL 構成の確認

ERSPAN ACL 構成を表示するには、次の表に示す適切な show コマンドを実行します。

コマンド	目的
show ip access-lists *name* 例： `switch(config-acl)# show ip access-lists erpsan-acl`	ERSPAN ACL の設定を表示します。
show vlan access-map *name* 例： `switch(config-acl)# show vlan access-map erspan_filter`	VLAN アクセス マップに関する情報を表示します。

コマンド	目的
show monitor session {**all** | *session-number* | **range** *session-range*} [**brief**] 例： `switch(config-acl)# show monitor session 1`	ERSPAN セッション設定を表示します。

UDF ベース ERSPAN の設定

外部または内部パケット フィールド（ヘッダまたはペイロード）のユーザ定義フィールド（UDF）で照合し、一致するパケットを ERSPAN 宛先に送信するようにデバイスを設定できます。そのように設定することで、ネットワークのパケットドロップを分析して、分離することができます。

始める前に

UDF ベース ERSPAN をイネーブルにするのに十分な空き領域を確保するために、**hardware access-list tcam region** コマンドを使用して適切な TCAM リージョン（racl、ifacl、または vacl）が設定されていることを確認します。詳細については、Cisco Nexus 9000 シリーズ NX-OS セキュリティ設定ガイドの『ACL TCAM リージョン サイズの設定』セクションを参照してください。

手順の概要

1. **configure terminal**
2. **udf** *udf-name offset-base offset length*
3. **hardware access-list tcam region** {**racl | ifacl | vacl** }**qualify udf** *udf-names*
4. **copy running-config startup-config**
5. **reload**
6. **ip access-list** *erspan-acl*
7. 次のいずれかのコマンドを入力します。
   - **permit udf** *udf-name value mask*
   - **permit ip** *source destination* **udf** *udf-name value mask*
8. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例：	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
	`switch# configure terminal` `switch(config)#`	

ステップ 2	**udf** *udf-name offset-base offset length*
	例：
	`switch(config)# udf udf-x packet-start 12 1`
	`switch(config)# udf udf-y header outer l3 20 2`
		次のように UDF を定義します。
		• *udf-name*：UDF の名前を指定します。名前には最大 16 文字の英数字を入力できます。
		• *offset-base*：UDF オフセット ベースを以下のように指定します。ここで**header**は、オフセットのために考慮に入れるべきパケットヘッダーです： **packet-start** | **header** {**outer** | **inner** {**l3** | **l4**}}.
		• オフセット：オフセットベースからのオフセット バイト数を指定します。オフセット ベース（レイヤ 3/レイヤ 4 ヘッダー）の最初のバイトを照合するには、オフセットを 0 に設定します。
		• 長さ：オフセット からバイトの数を指定します。1 または 2 バイトのみがサポートされています。追加のバイトに一致させるためには、複数の UDF を定義する必要があります。
		複数の UDF を定義できますが、シスコは必要な UDF のみ定義することを推奨します。

ステップ 3	**hardware access-list tcam region** {**racl | ifacl | vacl** }**qualify udf** *udf-names* 例： `switch(config)# hardware access-list tcam region racl qualify udf udf-x udf-y`	次のいずれかの TCAM リージョンに UDF を付加します。 • racl：レイヤ 3 ポートに適用します：レイヤ 2 およびレイヤ 3 ポートに適用します。 • ifacl：レイヤ 2 ポートに適用します。 • vacl：送信元 VLAN に適用します。 UDF は TCAM リージョンに最大 8 個まで付加できます。 （注） UDF 修飾子が追加されると、TCAM リージョンはシングル幅から倍幅に拡大します。十分な空きスペースがあることを確認してください。 それ以外の場合このコマンドは拒否されます。必要な場合、未使用のリージョンから TCAM スペースが減りますので、このコマンドを再入力します。詳細については、Cisco Nexus 9000 シリーズ NX-OS セキュリ

	コマンドまたはアクション	目的
		ティ設定ガイドの『ACL TCAM リージョン サイズの設定』セクションを参照してください。 （注） このコマンドの **no** 形式は、UDF を TCAM リージョンから切り離し、リージョンをシングル幅に戻します。
ステップ 4	必須: **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。
ステップ 5	必須: **reload** 例： `switch(config)# reload`	デバイスがリロードされます。 （注） UDF 設定は **copy running-config startup-config** + **reload** を入力した後のみ有効になります。

ステップ 6	**ip access-list** *erspan-acl*
	例：
	`switch(config)# ip access-list erspan-acl-udf-only`
	`switch(config-acl)#`
		IPv4 アクセス コントロール リスト（ACL）を作成して、IP アクセス リスト コンフィギュレーション モードを開始します。

ステップ 7	次のいずれかのコマンドを入力します。
	• **permit udf** *udf-name value mask*
	• **permit ip** *source destination* **udf** *udf-name value mask*
	例：
	`switch(config-acl)# permit udf udf-x 0x40 0xF0 udf-y 0x1001 0xF00F`
	例：
	`switch(config-acl)# permit ip 10.0.0.0/24 any udf udf-x 0x02 0x0F udf-y 0x1001 0xF00F`
		ACL を設定し、UDF（例 1）でのみ、または外部パケット フィールドについて現在のアクセス コントロール エントリ（ACE）と併せて UDF で一致させるように設定します（例 2）
		
		シングル ACL は、UDFがある場合とない場合の両方とも、ACE を有することができます。各 ACE には一致する異なる UDF フィールドがあるか、すべての ACE を UDF の同じリストに一致させることができます。

ステップ 8	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ERSPAN 切り捨ての設定

切り捨ては、ローカルおよび ERSPAN 送信元セッションに対してのみ設定できます。

手順の概要

1. **configure terminal**
2. **monitor session** *session-number* **type erspan-source**
3. **source interface** *type slot*/*port* [**rx** | **tx** | **both**]
4. **mtu** *size*
5. **destination interface** *type slot*/*port*
6. **no shut**
7. （任意） **show monitor session** *session*
8. **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**monitor session** *session-number* **type erspan-source**
	例：
	`switch(config)# monitor session 10 type erspan-source`
	`switch(config-erspan-src)#`
		指定されたERSPANセッションのモニタ設定モードに入ります。

ステップ 3	**source interface** *type slot*/*port* [**rx**	**tx**	**both**] 例： `switch(config-erspan-src)# source interface ethernet 1/5 both`	送信元インターフェイスを設定します。

ステップ 4	**mtu** *size*
	例：
	`switch(config-erspan-src)# mtu 512`
	例：
	`switch(config-erspan-src)# mtu ?`
	`<512-1518> Enter the value of MTU truncation size for ERSPAN packets (erspan header + truncated original packet)`
		MTU の切り捨てサイズを設定します。設定された MTU サイズよりも大きい ERSPAN パケットはすべて、設定されたサイズに切り捨てられます。ERSPAN パケットの切り捨てのMTU範囲は次のとおりです。
		• Cisco Nexus 9300-EX シリーズ スイッチの MTU サイズの範囲は 512ー1518 バイトです。
		• Cisco Nexus 9300-FX シリーズ スイッチの MTU サイズの範囲は 64ー1518 バイトです。
		• 9700-EX および 9700-FX ライン カードを搭載した Cisco Nexus 9500 プラットフォーム スイッチの場合、MTU サイズの範囲は 512ー1518 バイトです。

	コマンドまたはアクション	目的
		• Cisco Nexus 9808 および 9804 プラットフォームスイッチの MTU サイズは 343 バイトです（FCS を除く）。
ステップ 5	**destination interface** *type slot/port* 例： `switch(config-erspan-src)# destination interface Ethernet 1/39`	イーサネット ERSPAN 宛先ポートを設定します。
ステップ 6	**no shut** 例： `switch(config-erspan-src)# no shut`	ERSPAN セッションをイネーブルにします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 7	（任意） **show monitor session** *session* 例： `switch(config-erspan-src)# show monitor session 5`	ERSPAN の設定を表示します。
ステップ 8	**copy running-config startup-config** 例： `switch(config-erspan-src)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ERSPAN 宛先セッションの設定

送信元 IP アドレスからローカル デバイス上の宛先ポートにパケットをコピーするように ERSPAN宛先セッションを設定できます。デフォルトでは、ERSPAN宛先セッションはシャット ステートで作成されます。

始める前に

スイッチポート モニタ モードで宛先ポートが設定されていることを確認します。

手順の概要

1. **configure terminal**
2. **interface ethernet** *slot/port*[*-port*]
3. **switchport**
4. **switchport mode** [**access** | **trunk**]
5. **switchport monitor**
6. ステップ 2 ～ 5 を繰り返して、追加の ERSPAN 宛先でモニタリングを設定します。
7. **no monitor session** {*session-number* | **all**}
8. **monitor session** {*session-number* | **all**} **type erspan-destination**
9. **description** *description*

10. **source ip** *ip-address*
11. **destination** {[**interface** [*type slot/port*[*-port*]]] [**port-channel** *channel-number*]]}
12. （任意） ステップ 11 を繰り返して、すべての ERSPAN 宛先を設定します。
13. **erspan-id** *erspan-id*
14. **no shut**
15. **exit**
16. **exit**
17. （任意） **show monitor session** {**all** | *session-number* | **range** *session-range*}
18. （任意） **show running-config monitor**
19. （任意） **show startup-config monitor**
20. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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 ethernet** *slot/port*[*-port*]
	例：
	`switch(config)# interface ethernet 2/5`
	`switch(config-if)#`
		選択したスロットおよびポートまたはポート範囲で、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	**switchport** 例： `switch(config-if)# switchport`	選択したスロットおよびポートまたはポート範囲でスイッチポート パラメータを設定します。
ステップ 4	**switchport mode** [**access** | **trunk**] 例： `switch(config-if)# switchport mode trunk`	選択したスロットおよびポートまたはポート範囲で次のスイッチポート モードを設定します。 • アクセス • トランク
ステップ 5	**switchport monitor** 例： `switch(config-if)# switchport monitor`	ERSPAN 宛先としてスイッチポート インターフェイスを設定します。
ステップ 6	ステップ 2 ～ 5 を繰り返して、追加の ERSPAN 宛先でモニタリングを設定します。	—

	コマンドまたはアクション	目的
ステップ 7	**no monitor session** {*session-number* | **all**} 例： `switch(config-if)# no monitor session 3`	指定したERSPANセッションの設定を消去します。新しいセッション コンフィギュレーションは、既存のセッション コンフィギュレーションに追加されます。

ステップ 8	**monitor session** {*session-number* | **all**} **type erspan-destination**
	例：
	`switch(config-if)# monitor session 3 type erspan-destination`
	`switch(config-erspan-dst)#`
		ERSPAN 宛先セッションを設定します。

ステップ 9	**description** *description* 例： `switch(config-erspan-dst)# description erspan_dst_session_3`	セッションの説明を設定します。デフォルトでは、説明は定義されません。説明には最大 32 の英数字を使用できます。

ステップ 10	**source ip** *ip-address*
	例：
	`switch(config-erspan-dst)# source ip 10.1.1.1`
	`switch(config-erspan-dst)# source ipv6 2001:DB8:1::1`
		source ipv6 *ipv6-address*
		ERSPAN セッションの宛先 IPv4 または IPv6 アドレスを構成します。送信元 IPv4 または IPv6 アドレスは、ローカルに構成された IPv4 または IPv6 アドレスです。ERSPAN 宛先セッションの送信元 IPv4 または IPv6 アドレスは、カプセル化されたデータの受信元である ERSPAN 送信元セッションで構成された宛先 IPv4 または IPv6 アドレスと一致する必要があります。ERSPAN 送信元セッションごとに 1 つの宛先 IPv4 または IPv6 アドレスのみがサポートされます。
		（注）
		IPv6 は、Cisco NX-OS リリース 10.2(3)F からサポートされています。

ステップ 11	**destination** {[**interface** [*type slot/port*[*-port*]]] [**port-channel** *channel-number*]]} 例： `switch(config-erspan-dst)# destination interface ethernet 2/5`	コピーする送信元パケットの宛先を設定します。宛先インターフェイスを設定できます。 （注） 宛先ポートをトランク ポートとして設定できます。
ステップ 12	（任意） ステップ 11 を繰り返して、すべての ERSPAN 宛先を設定します。	—
ステップ 13	**erspan-id** *erspan-id* 例： `switch(config-erspan-dst)# erspan-id 5`	ERSPAN セッションの ERSPAN ID を設定します。指定できる範囲は 1 ～ 1023 です。

	コマンドまたはアクション	目的
ステップ 14	**no shut** 例： `switch(config-erspan-dst)# no shut`	ERSPAN 宛先セッションを有効にします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 15	**exit** 例： `switch(config-erspan-dst)# exit`	モニタ設定モードを閉じます。
ステップ 16	**exit** 例： `switch(config)# exit`	グローバル コンフィギュレーション モードを終了します。
ステップ 17	（任意） **show monitor session** {**all** | *session-number* | **range** *session-range*} 例： `switch(config)# show monitor session 3`	ERSPAN セッション設定を表示します。
ステップ 18	（任意） **show running-config monitor** 例： `switch(config-erspan-src)# show running-config monitor`	ERSPAN の実行コンフィギュレーションを表示します。
ステップ 19	（任意） **show startup-config monitor** 例： `switch(config-erspan-src)# show startup-config monitor`	ERSPAN のスタートアップ コンフィギュレーションを表示します。
ステップ 20	（任意） **copy running-config startup-config** 例： `switch(config-erspan-src)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ERSPAN 設定の確認

ERSPAN 設定を表示するには、次のいずれかの作業を行います。

コマンド	目的
show monitor session {**all** | *session-number* | **range** *session-range*} [**brief**]	ERSPAN セッション設定を表示します。
show running-config monitor	ERSPAN の実行コンフィギュレーションを表示します。

コマンド	目的
show startup-config monitor	ERSPAN のスタートアップ コンフィギュレーションを表示します。
show monitor session [*session-id* | **all**] **stats**	Cisco Nexus 9808 および 9804 プラットフォーム スイッチの ERSPAN セッション統計を表示します。
clear monitor session [*session-id* | **all**] **stats** [**both** | **rx** | **tx**]	Cisco Nexus 9808 および 9804 プラットフォーム スイッチの ERSPAN セッション統計をクリアします。

ERSPAN の設定例

IPv6 経由の ERSPAN 送信元セッションの設定例

次に、IPv6 経由の ERSPAN 送信元セッションを設定する例を示します。

```
switch# configure terminal
switch(config)# monitor erspan origin ipv6-address 2001::10:0:0:9 global
switch(config)# moni session 10 type erspan-source
switch(config-erspan-src)# erspan-id 10
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# source interface ethernet 1/64
switch(config-erspan-src)# destination ip 10.1.1.2
```

単一方向 ERSPAN セッションの設定例

次に、単一方向 ERSPAN セッションを設定する例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 14/30
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# no monitor session 3
switch(config)# monitor session 3 rxswitch(config-erspan-src)# source interface ethernet
 2/1-3 rx
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 10.1.1.2
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
switch(config)# show monitor session 1
```

ERSPAN ACL の設定例

次に、ERSPAN ACL を設定する例を示します。

```
switch# configure terminal
switch(config)# ip access-list match_10_pkts
switch(config-acl)# permit ip 10.0.0.0/24 any
switch(config-acl)# exit
switch(config)# ip access-list match_172_pkts
switch(config-acl)# permit ip 172.16.0.0/24 any
switch(config-acl)# exit
```

定義済みの ACL フィルタに基づいて対象トラフィックが選択されるさまざまな ERSPAN 接続先の場合、最後に設定されたセッションが常に高い優先順位を持ちます。

たとえば、モニター セッション 1 が構成されているとします。次に、モニター セッション 2 が構成されます。この場合、ERSPAN トラフィック フィルタは意図したとおりに機能します。ただし、ユーザーがモニター セッション 1 に戻り、既存の構成行の 1 つを再適用した場合 (構成に新しい変更はありません)。その後、スパンされたトラフィックはモニター セッション 1 に戻ります。

マーカー パケットの設定例

次に、2 秒間隔で ERSPAN マーカー パケットを有効にする例を示します。

```
switch# configure terminal
switch(config)# monitor erspan origin ip-address 172.28.15.250 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# header-type 3
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 10.1.1.2
switch(config-erspan-src)# source interface ethernet 1/15 both
switch(config-erspan-src)# marker-packet 100
switch(config-erspan-src)# no shut
switch(config-erspan-src)# show monitor session 1
session 1
---------------
type              : erspan-source
state             : up
granularity       : nanoseconds
erspan-id         : 1
vrf-name          : default
destination-ip    : 10.1.1.2
ip-ttl            : 16
ip-dscp           : 5
header-type       : 3
origin-ip         : 172.28.15.250 (global)
source intf       :
    rx            : Eth1/15
    tx            : Eth1/15
    both          : Eth1/15
    rx            :
marker-packet     : enabled
packet interval   : 100
packet sent       : 25
packet failed     : 0
egress-intf       :
```

UDF ベース ERSPAN の設定例

次に、以下の一致基準を使用して、カプセル化された IP-in-IP パケットの内部 TCP フラグで照合する UDF ベース ERSPAN を設定する例を示します。

- 外部送信元 IP アドレス：10.0.0.2
- 内部 TCP フラグ：緊急 TCP フラグを設定
- バイト：Eth Hdr（14）＋外部 IP（20）＋内部 IP（20）＋内部 TCP（20、ただし、13 番目のバイトの TCP フラグ）
- パケットの先頭からのオフセット：14 + 20 + 20 + 13 = 67
- UDF の照合値：0x20
- UDF マスク：0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify udf udf_tcpflags
copy running-config startup-config
reload
ip access-list acl-udf
permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1 type erspan-source
source interface Ethernet 1/1
filter access-group acl-udf
```

次に、以下の一致基準を使用して、レイヤ 4 ヘッダーの先頭から 6 バイト目のパケット署名（DEADBEEF）と通常の IP パケットを照合する UDF ベース ERSPAN を設定する例を示します。

- 外部送信元 IP アドレス：10.0.0.2
- 内部 TCP フラグ：緊急 TCP フラグを設定
- バイト：Eth Hdr（14）＋IP（20）＋TCP（20）＋ペイロード：112233445566DEADBEEF7788
- レイヤ 4 ヘッダーの先頭からのオフセット：20 + 6 = 26
- UDF の照合値：0xDEADBEEF（2 バイトのチャンクおよび 2 つの UDF に分割）
- UDF マスク：0xFFFFFFFF

```
udf udf_pktsig_msb header outer l3 26 2
udf udf_pktsig_lsb header outer l3 28 2
hardware access-list tcam region racl qualify udf udf_pktsig_msb udf_pktsig_lsb
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1 type erspan-source
source interface Ethernet 1/1
filter access-group acl-udf-pktsig
```

ERSPAN 切り捨ての設定例

次に、MPLS ストリッピングで使用する ERSPAN 切り捨てを設定する例を示します。

```
mpls strip
ip access-list mpls
  statistics per-entry
  20 permit ip any any redirect Ethernet1/5

interface Ethernet1/5
  switchport
  switchport mode trunk
  mtu 9216
  no shutdown

monitor session 1
  source interface Ethernet1/5 tx
  mtu 64
  destination interface Ethernet1/6
  no shut
monitor session 21 type erspan-source
  description "ERSPAN Session 21"
  header-type 3
  erspan-id 21
  vrf default
  destination ip 10.1.1.2
  source interface Ethernet1/5 tx
  mtu 64
  no shut
monitor session 22 type erspan-source
  description "ERSPAN Session 22"
  erspan-id 22
  vrf default
  destination ip 10.2.1.2
  source interface Ethernet1/5 tx
  mtu 750
  no shut
monitor session 23 type erspan-source
  description "ERSPAN Session 23"
  header-type 3
  marker-packet 1000
  erspan-id 23
  vrf default
  destination ip 10.3.1.2
  source interface Ethernet1/5 tx
  mtu 1000
  no shut
```

IPv4 上の ERSPAN 接続先セッションの構成例

次に、IPv4 上でERSPAN 接続先セッションを構成する例を示します。

destination interface eth1/1 はスイッチポート モニタ モードです。このインターフェイスは、mpls strip、tunnel、nv Overlay、vn-segment-vlan-based、mpls segment-routing、mpls evpn、mpls static、mpls oam、mpls l3vpn、mpls ldp、および nv overlay evpn 機能と共存できません。

```
switch# monitor session 1 type erspan-destination
switch(config)# erspan-id 1
switch(config-erspan-dst)# source ip 10.1.1.1
```

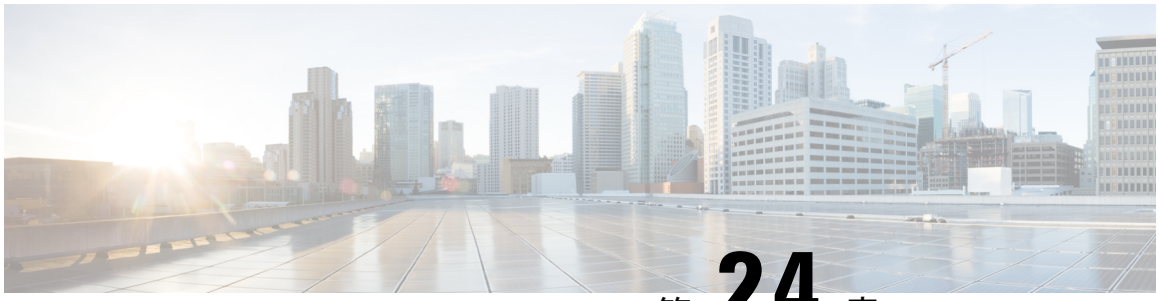
```
switch(config-erspan-dst)# destination interface eth1/1
switch(config-erspan-dst)# no shut
switch(config-erspan-dst)# exit
```

IPv6 上の ERSPAN 接続先セッションの構成例

次に、IPv6 上でERSPAN 接続先セッションを構成する例を示します。

destination interface eth1/1 はスイッチポート モニタ モードです。このインターフェイスは、mpls strip、tunnel、nv Overlay、vn-segment-vlan-based、mpls segment-routing、mpls evpn、mpls static、mpls oam、mpls l3vpn、mpls ldp、および nv overlay evpn 機能と共存できません。

```
switch# monitor session 1 type erspan-destination
switch(config)# erspan-id 1
switch(config-erspan-dst)# source ipv6 2001:DB8:1::1
switch(config-erspan-dst)# destination interface eth1/1
switch(config-erspan-dst)# no shut
switch(config-erspan-dst)# exit
```



第 24 章

LLDP の構成

この章では、ローカル ネットワーク上の他のデバイスを検出するために、Link Layer Discovery Protocol（LLDP）を設定する方法について説明します。

この章は、次の内容で構成されています。

- LLDP について （483 ページ）
- LLDP に関する注意事項および制約事項 （486 ページ）
- LLDP のデフォルト設定 （487 ページ）
- LLDP の構成 （488 ページ）
- LLDP 設定の確認 （500 ページ）
- LLDP の設定例 （501 ページ）

LLDP について

Cisco Discovery Protocol（CDP）は、ネットワークに接続された他のシスコ デバイスを自動的に検出し学習することをネットワーク管理アプリケーションによって可能にするデバイス検出プロトコルです。Cisco Discovery Protocol（CDP）は、ネットワークに接続された他のシスコ デバイスを自動的に検出し学習することをネットワーク管理アプリケーションによって可能にするデバイス検出プロトコルです。

他社製デバイスのディスカバリを許可するために、スイッチは、IEEE 802.1ab 規格で定義されているベンダー ニュートラルなデバイス ディスカバリ プロトコルである Link Layer Discovery Protocol（LLDP）もサポートしています。LLDP を使用すると、ネットワーク デバイスはそれ自体のデバイスに関する情報を、ネットワーク上の他のデバイスにアドバタイズできます。このプロトコルはデータリンク層で動作するため、異なるネットワーク層プロトコルが稼働する 2 つのシステムで互いの情報を学習できます。

LLDP は、デバイスおよびそのインターフェイスの機能と現在のステータスに関する情報を送信する単一方向のプロトコルです。LLDP デバイスはこのプロトコルを使用して、他の LLDP デバイスからだけ情報を要求します。

LLDP は一連の属性をサポートしており、これを使用して他のデバイスを検出します。これらの属性には、タイプ、長さ、および値（TLV）の説明が含まれています。LLDP デバイスは

TLV を使用して、ネットワーク上の他のデバイスと情報を送受信できます。設定情報、デバイスの機能、デバイス ID などの詳細情報は、このプロトコルを使用してアドバタイズできます。

LLDP は、デフォルトで次の TLV をアドバタイズします。

- DCBXP
- 管理用アドレス
- ポートの説明
- ポートVLAN
- システム機能
- システムの説明
- システム名

DCBXP について

Data Center Bridging Exchange Protocol（DCBXP）は、LLDP を拡張したプロトコルです。このプロトコルは、ピア間のノードパラメータのアナウンス、交換、およびネゴシエートに使用されます。DCBXP パラメータは、LLDP パケットの DCBXP TLV としてパッケージ化されます。CEE を使用する場合、DCBXP は LLDP 経由の確認応答メカニズムを使用します。ポートが起動すると、DCBX TLV が送信され、受信した DCBX TLV が処理されます。デフォルトでは、DCBX プロトコルは自動検出に設定され、両方のピアでサポートされている最新のプロトコルバージョンが使用されます。

DCBXP を使用してパラメータとピア ノードの交換およびネゴシエーションが必要な機能は次のとおりです。

- 優先度ベース フロー制御（PFC）：PFC は、イーサネットの既存のポーズ メカニズムを拡張するものです。これは、ユーザプライオリティまたはサービス クラスに基づいてポーズを有効にします。PFC を使用して 8 つの仮想リンクに分割された物理リンクは、他の仮想リンクのトラフィックに影響を与えることなく、単一の仮想リンクでポーズを使用できる機能を提供します。ユーザごとのプライオリティ単位でポーズを有効にすることで、IP トラフィック用のパケット ドロップの輻輳管理を維持しながら、ドロップの無いサービスが必要なトラフィックに対し管理者がロスレス リンクを作成できます。
- 強化された転送選択（ETS）：ETS は、仮想リンクの最適帯域幅管理を可能にします。ETS（Enhanced Transmission Selection）は、優先度グルーピングとも呼ばれます。PFC の同じ優先度クラス内の処理の区別を有効にします。帯域幅割り当て、低遅延、またはベストエフォートに基づいて処理の優先順位が付けられるため、結果としてグループごとのトラフィック クラス割り当てが可能になります。たとえば、イーサネット トラフィック クラスに高優先度を指定し、その同じクラスの中でベスト エフォートを指定する場合です。ETS によって、同じ優先度クラスの中でトラフィックを差別化する、つまり優先度グループを作成することが可能になります

- アプリケーション プライオリティ構成：特定のプロトコルに割り当てられたプライオリティに関する情報を伝送します。
- DSCP マッピングへのプライオリティ：QoS ポリシーで構成された DSCP 値と COS 値のマッピングは、アプリケーション プライオリティTLV で送信されます。

（注） Quality of Service（QoS）機能の詳細については、『*Cisco Nexus 9000* シリーズ *NX-OS Quality of Service* 設定ガイド』を参照してください。

DCBXP はデフォルトでイネーブルであり、提供された LLDP はイネーブルです。LLDP が有効な場合、DCBXP は **[no] lldp tlv-select dcbxp** コマンドお使用して有効または無効にできます。LLDP の送信または受信がディセーブルになっているポートでは、DCBXP はディセーブルです。

Cisco NX-OS リリース 10.2(3)F 以降、追加のコマンドが導入されました: **[no] lldp tlv-select dcbxp egress-queuing**。**[no] lldp tlv-select dcbxp** コマンドはピアと交換される ETS 情報で入力キューイング パラメータを送信しますが、**[no] lldp tlv-select dcbxp egress-queuing** コマンドは ETS 情報で出力キューイング パラメータを送信します。したがって、帯域幅とプライオリティ情報は出力キューイング ポリシーから抽出され、ピアと交換されます。

一度に、出力キューイングまたは入力キューイングのいずれかを設定するには、それらが互いに上書きするときに **lldp tlv-select dcbxp egress-queuing** または **lldp tlv-select dcbxp** コマンドのいずれかを実行します。

両方のコマンドの no 形式は、すべてのインターフェイスで DCBXP 交換を停止します。

上記の 2 つのコマンドのどちらが有効になっているかを表示するには、**show lldp tlv-select** コマンドを実行します。

システム レベルでのデフォルトの入力キューイング ポリシーが切り離されると、すべてのインターフェイスの DCBXP 交換は、ETS 設定および推奨 TLV の送信を停止します。ただし、システム レベルのデフォルトの出力キューイング ポリシーは切り離すことができません。

高可用性

LLDP 機能はステートレス リスタートおよびステートフル リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバー後に、実行コンフィギュレーションを適用します。

ハイ アベイラビリティの詳細については、『*Cisco Nexus 9000* シリーズ *NX-OS* ハイ アベイラビリティおよび冗長性ガイド』を参照してください。

仮想化のサポート

サポートされる LLDP のインスタンスは 1 個です。

LLDP に関する注意事項および制約事項

LLDP の設定のガイドラインおよび制限事項は、次のとおりです。

- インターフェイス上でLLDP機能をイネーブルまたはディセーブルにするには、事前にデバイス上で LLDP プロトコルをイネーブルにしておく必要があります。
- LLDP は物理インターフェイスだけでサポートされています。
- LLDP は 1 つのポートにつき 1 つのデバイスを検出できます。リリース 10.1(1) 以降では、物理インターフェイスごとに複数の LLDP ネイバーが次のプラットフォームでサポートされます。
  - N9K-C93180YC-FX3
  - N9K-C93108TC-FX3P
  - N9K-C93180YC-FX3
- LLDP は 1 つのポートにつき 1 つのデバイスを検出できます。
- DCBXP は次のプラットフォームでサポートされます。
  - Cisco Nexus 9200、9300、9300-EX、9300-FX、9300-FX2 および 9300-FX3 シリーズ スイッチ
  - Cisco Nexus 9332C、9332PQ、9364C、9372PX、9372PX-E、および 9396PX スイッチ
  - Cisco Nexus 9504 および 9508 スイッチで、X9432PQ、X9564PX、X9636PQ、X9732C-EX、および X9736C-FX ライン カードを搭載したもの
- Cisco Nexus 3232C および 3264Q スイッチは、DCBXP をサポートしていません。
- DCBXP の非互換性のメッセージは、物理ケーブル ループバック接続がデバイスに存在する場合にスイッチ上の network QoS ポリシーを変更するときに表示されることがあります。非互換性があるのは短時間であり、自力に解決されます。
- PFC TLV は、ネットワーク QoS ポリシーで少なくとも 1 つの COS 値に対して一時停止が有効になっており、priority-flow-control モードは、インターフェイス レベルで auto に設定されます。
- Cisco NX-OS リリース 10.2(3)F 以降、[**no**] **lldp tlv-select dcbxp egress-queuing** コマンドが導入され、スイッチの出力キューイング設定をアドバタイズするオプションが提供されます。この機能は、Cisco Nexus 9200、9300-EX と 9300-FX プラットフォーム スイッチでサポートされます。
- **lldp tlv-select dcbxp** コマンドが使用されている場合は入力キューイングが適用され、**lldp tlv-select dcbxp egress-queuing** が使用されている場合は出力キューイングが適用されているときに、DCBX TLV が送信されます。

- Cisco NX-OS リリース 10.2(3)F 以降、LLDP シャーシ ID を正しくアドバタイズする機能には、新しいグローバル構成コマンド、**lldp chassis-id switch** が導入されています。これは、ポートの MAC アドレスの代わりに、スイッチ シャーシの MAC アドレスをアドバタイズします。つまり、すべてのポートはスイッチ シャーシの MAC アドレスのみを公開するということです。この機能は、すべての Cisco Nexus 9000 シリーズ プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.3(1)F 以降、LLDP（マルチ ネイバーとポート チャネル）は Cisco Nexus 9808 プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.3（1）F 以降、PFC、および ECN は Cisco Nexus 9300-FX3 プラットフォーム スイッチでサポートされます。FEX モードで動作している 9300-FX3 スイッチでは、DCBX は入力キューイングでサポートされます。
- Cisco NX-OS リリース 10.3（2）F 以降、802.1 リンク集約、802.1 VLAN 名、802.3 最大フレーム サイズなどの新しいオプション パラメータは、**lldp tlv-select** コマンドに対してデフォルトで有効になっています。

  ユーザーが VLAN の作成時に 32 文字以上を構成した場合でも、TLV 名には標準規格で指定されている最初の 32 文字のみが含まれます。25 の現用系 VLAN の名前のみが表示されます。
- Cisco NX-OS リリース 10.4(1)F 以降、LLDP は次のスイッチおよびライン カードでサポートされます。
  - Cisco Nexus 9804 プラットフォーム スイッチ
  - Cisco Nexus X98900CD-A ライン カード
  - Cisco Nexus X98900CD-A ライン カードと Cisco Nexus 9808 および 9804 スイッチ
- Cisco NX-OS リリース 10.4(2)F 以降、LLDP は Cisco Nexus 9232E-B1 プラットフォーム スイッチでサポートされます。
- 構成済みの場合、LLDP のシステム名でドメイン名を表示することもできます。管理 VRF にドメイン名が構成されている場合は、その名がデフォルトの VRF ドメイン名よりも高い優先度に与えられます。管理 VRF のドメイン名が存在しない場合、デフォルトの VRF ドメイン名を調べ、構成されている場合はそれを使用します。

LLDP のデフォルト設定

この表は、LLDP のデフォルト設定を示します。

パラメータ	デフォルト
グローバル LLDP	無効
インターフェイス上の LLDP	イネーブル（LLDP がグローバルにイネーブルになった後）

パラメータ	デフォルト
LLDP 保持時間（ディセーブルになる前）	120 秒
LLDP 再初期化遅延	2 秒
LLDP タイマー（パケット更新頻度）	30 秒
LLDP TLV	有効
LLDP 受信	イネーブル（LLDPがグローバルにイネーブルになった後）
LLDP 転送	イネーブル（LLDPがグローバルにイネーブルになった後）
DCBXP	有効（提供された LLDP が有効になります）
DCBXP のバージョン	自動検出

LLDP の構成

（注） この機能の Cisco NX-OS コマンドは、類似した機能の Cisco IOS コマンドと異なる場合があります。

LLDP をグローバルに有効化または無効化する

デバイスでLLDPをグローバルにイネーブルまたはディセーブルにできます。デバイスでLLDPパケットの送信および受信を可能にするには、LLDP をグローバルにイネーブルにする必要があります。

手順の概要

1. **configure terminal**
2. [**no**] **feature lldp**
3. （任意） [**no**] **lldp chassis-id switch**
4. （任意） **show running-config lldp**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **feature lldp** 例： `switch(config)# feature lldp`	デバイス上で LLDP をイネーブルまたはディセーブルにします。LLDP はデフォルトでディセーブルです。 （注） このコマンドを有効にすると、デフォルトで、スイッチはポートごとにポートの MAC アドレスをアドバタイズします。
ステップ 3	（任意） [**no**] **lldp chassis-id switch** 例： `switch(config)# lldp chassis-id switch`	このコマンドを有効にして、スイッチ シャーシの MAC アドレスをすべてのポートにアドバタイズする必要があることを示します。 ポートごとのポート MAC アドレスのアドバタイズに戻すには、このコマンドの no 形式を使用します。 （注） スイッチ シャーシの MAC アドレスを表示するには、**show vdc detail** コマンドを使用します。
ステップ 4	（任意） **show running-config lldp** 例： `switch(config)# show running-config lldp`	LLDP のグローバル コンフィギュレーションを表示します。LLDP が有効の場合、「feature lldp」と表示されます。LLDP が無効の場合、「Invalid command」エラーが表示されます。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

インターフェイス上での LLDP の有効化または無効化

LLDP をグローバルに有効にすると、LLDP は、デフォルトで、サポートされているすべてのインターフェイスで有効になります。ただし、LLDP パケットの送信だけ、または受信だけを実行するために、個々のインターフェイスでの LLDP のイネーブルまたはディセーブル、あるいはインターフェイスの選択的な設定を実行できます。

始める前に

デバイスで LLDP をグローバルにイネーブルにしていることを確認します。

手順の概要

1. **configure terminal**
2. **interface** *interface slot/port*
3. [**no**] **lldp transmit**
4. [**no**] **lldp receive**
5. （任意） **show lldp interface** *interface slot/port*
6. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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 slot/port*
	例：
	`switch(config)# interface ethernet 7/1`
	`switch(config-if)#`
		LLDP 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	[**no**] **lldp transmit** 例： `switch(config-if)# lldp transmit`	インターフェイス上で LLDP パケットの送信をイネーブルまたはディセーブルにします。LLDP をグローバルに有効にすると、LLDPは、デフォルトで、サポートされているすべてのインターフェイスで有効になります。
ステップ 4	[**no**] **lldp receive** 例： `switch(config-if)# lldp receive`	インターフェイス上で LLDP パケットの受信をイネーブルまたはディセーブルにします。LLDP をグローバルに有効にすると、LLDPは、デフォルトで、サポートされているすべてのインターフェイスで有効になります。
ステップ 5	（任意） **show lldp interface** *interface slot/port* 例： `switch(config-if)# show lldp interface ethernet 7/1`	インターフェイス上で LLDP の設定を表示します。

	コマンドまたはアクション	目的
ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

DCBXP 出力キューイングの構成

DCBXP 出力キューイングを構成するには、次の手順を使用します。

始める前に

- デバイスでLLDPをグローバルで有効にされていることを確認します（グローバル構成コマンド **feature lldp**）。

（注） LLDP をグローバルに有効にすると、LLDP は、デフォルトで、サポートされているすべてのインターフェイスで有効になります。

手順の概要

1. **configure terminal**
2. **lldp tlv-select dcbxp**
3. （任意） **lldp tlv-select dcbxp egress-queuing**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	必須: **lldp tlv-select dcbxp**
	例：
	`switch(config)# lldp tlv-select dcbxp`
	`switch(config)#`
		DCBXP TLV をグローバルに有効にし、ピアと交換される ETS 情報の入力キューイング パラメータの送信を開始します。

	コマンドまたはアクション	目的

ステップ 3	（任意） **lldp tlv-select dcbxp egress-queuing**
	例：
	`switch(config)# lldp tlv-select dcbxp egress-queuing`
	`switch(config)#`
		DCBXP TLV をグローバルに有効にし、ETS 情報の出力キューイング パラメータの送信を開始します。

DCBXP プロトコル バージョンの設定

DCBX TLVが送信されるプロトコルバージョンを指定できます。

（注）
- Nexus 9000 スイッチは、CIN バージョンを使用した自動ネゴシエーションまたはハードコーディングの DCBXP TLV をサポートしていません。
- ピアが同じバージョンを実行していない場合、リンクの DCBX パラメータが収束しない可能性があります。新しいプロトコルバージョンを有効にするには、リンクをリセットします。

始める前に

デバイスで LLDP をグローバルにイネーブルにしていることを確認します。

手順の概要

1. **configure terminal**
2. **interface** *interface slot*/*port*
3. **lldp dcbx version** *cee*/*ieee*/*auto*

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**interface** *interface slot*/*port*
	例：
	`switch(config)# interface ethernet 1/25`
	`switch(config-if)#`
		インターフェイス設定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	**lldp dcbx version** *cee/ieee/auto* 例： `switch(config-if)#lldp dcbx version cee`	Specifies the protocol version mode sent.送信されるプロトコル バージョン モードを指定します。 • *cee* 変数は、Converged Enhanced Ethernet（CEE）プロトコルバージョンの TLV のみを送信するようにポートを設定します。 • *ieee* 変数は、IEEE 802.1Qaz プロトコルバージョンの TLV のみを送信するようにポートを設定します。 • *auto* 変数は、両方のピアでサポートされている最新のプロトコルバージョンで TLV を送信するようにポートを設定します。 デフォルトは *auto* に設定されています。 （注） IEEE 802.1Qaz をサポートしていないデバイスは、自動ネゴシエーションの試行に適切に応答せず、lldp dcbx version cee 用にインターフェイスを手動で設定する必要があります。

物理インターフェイスごとの複数の LLDP ネイバー

多くの場合、ネットワーク デバイスは複数の LLDP パケットを送信しますが、そのうちの 1 つは実際のホストからのものです。Cisco Nexus スイッチがデバイスと通信しているが、インターフェイスごとに 1 つの LLDP ネイバーしか管理できない場合は、実際に必要なホストとのネイバーになることが失敗する可能性があります。これを最小限に抑えるために、Cisco Nexus スイッチ インターフェイスは複数の LLDP ネイバーをサポートできるため、正しいデバイスで LLDP ネイバーになる可能性が高くなります。

同じインターフェイスで複数の LLDP ネイバーをサポートするには、LLDP マルチネイバー サポートをグローバルに設定する必要があります。

（注） LLDP マルチネイバー サポートを設定する前に、DCBX をグローバルに無効にする必要があります。これを行わないと、エラー メッセージが表示されます。

LLDP マルチネイバー サポートのイネーブル化またはディセーブル化

始める前に

インターフェイスでLLDPマルチネイバーサポートを有効にする前に、次の点を考慮してください。

- デバイスでLLDPをグローバルにイネーブルにしていることを確認します（グローバル設定コマンド **feature lldp**）。

（注） LLDP をグローバルに有効にすると、LLDP は、デフォルトで、サポートされているすべてのインターフェイスで有効になります。

- 1 つのインターフェイスで最大 3 つのネイバーがサポートされます。
- LLDP マルチネイバーは、FEX インターフェイスではサポートされません。

手順の概要

1. **configure terminal**
2. **no lldp tlv-select dcbxp**
3. **no lldp tlv-select dcbxp egress-queuing**
4. [**no**] **lldp multi-neighbor**
5. **interface** *port* / *slot*
6. （任意） [**no**] **lldp transmit**
7. （任意） [**no**] **lldp receive**
8. （任意） **show lldp interface** *port* / *slot*
9. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します
		（注）
		show lldp tlv-select コマンドの出力で dcbxp と表示された場合、 ステップ 2 を実行し、ステップ 3 をスキップします。出力が dcbxp egress-queuing と表示した場合、ステップ 2 をスキップし、ステップ 3 を実行します。

	コマンドまたはアクション	目的
		これは、LLDPマルチネイバー サポートが設定されたときに、エラーメッセージが呼び出されないようにします。

ステップ 2	必須: **no lldp tlv-select dcbxp**
	例：
	`switch(config)# no lldp tlv-select dcbxp`
	`switch(config)#`
		DCBXP TLV をグローバルに無効にします。

ステップ 3	必須: **no lldp tlv-select dcbxp egress-queuing**
	例：
	`switch(config)# no lldp tlv-select dcbxp egress-queuing`
	`switch(config)#`
		DCBXP TLV をグローバルに無効にします。

ステップ 4	必須: [**no**] **lldp multi-neighbor**
	例：
	`switch(config)# lldp multi-neighbor`
	`switch(config)#`
		すべてのインターフェイスのLLDPマルチネイバー サポートをグローバルに有効または無効にします。

ステップ 5	**interface** *port* / *slot*
	例：
	`switch(config)# interface 1/1`
	`switch(config-if)#`
		LLDP 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 6	（任意） [**no**] **lldp transmit** 例： `switch(config-if)# lldp transmit`	インターフェイスでの LLDP パケットの送信をディセーブル（またはイネーブル）にします。 （注） このインターフェイスでの LLDP パケットの送信は、グローバル **feature lldp** コマンドを使用してイネーブルにされました。このオプションは、この特定のインターフェイスの機能を無効にします。
ステップ 7	（任意） [**no**] **lldp receive** 例： `switch(config-if)# lldp receive`	インターフェイスでの LLDP パケットの受信をディセーブル（またはイネーブル）にします。 （注） このインターフェイスでの LLDP パケットの受信は、グローバル **feature lldp** コマンドを使用してイネーブルになりました。このオプションは、この特定のインターフェイスの機能を無効にします。
ステップ 8	（任意） **show lldp interface** *port* / *slot* 例：	インターフェイス上で LLDP の設定を表示します。

	コマンドまたはアクション	目的
	`switch(config-if)# show lldp interface 1/1`	
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ポート チャネル インターフェイスでの LLDP サポートの有効化または無効化

始める前に

ポート チャネルで LLDP サポートを有効にする前に、次の点を考慮してください。

- デバイスでLLDPをグローバルにイネーブルにしていることを確認します（グローバル設定コマンド **feature lldp**）。

（注） LLDP をグローバルに有効にすると、LLDP は、デフォルトで、サポートされているすべてのインターフェイスで有効になります。

- ポート チャネルに **lldp transmit** および **lldp receive** コンフィギュレーション コマンドを適用しても、ポート チャネルのメンバーの設定には影響しません。
- LLDP ネイバーは、LLDP 送受信がポート チャネルの両側で設定されている場合にのみ、ポート チャネル間で形成されます。
- LLDP の送受信コマンドは、MCT、VPC、FEX ファブリック、FEX ポート チャネル、およびポート チャネル サブ インターフェイスでは機能しません。

（注） LLDP ポート チャネル機能をグローバルに有効にすると、LLDP 設定はこれらのポートタイプのいずれにも適用されません。ポート チャネルから設定が削除された場合、またはポート タイプ機能がグローバルに無効になった場合は、**lldp port-channel** コマンドを使用して新しくサポートされたポートチャネルで有効にすることはできません。コマンドはすでに発行されています。問題のポート チャネルで LLDP ポート チャネルを有効にするには、**lldp transmit** および **lldp receive** を各ポート チャネルに対して設定します（次の手順のステップ 4、5、および 6 を参照）。

手順の概要

1. **configure terminal**
2. **no lldp tlv-select dcbxp**
3. **no lldp tlv-select dcbxp egress-queuing**
4. [**no**] **lldp port-channel**
5. **interface port-channel** [*port-channel-number* | *port-channel-range*]
6. （任意） [**no**] **lldp transmit**
7. （任意） [**no**] **lldp receive**
8. （任意） **show lldp interface port-channel**l *port-channel-number*
9. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します
		（注）
		show lldp tlv-select コマンドの出力で dcbxp と表示された場合、 ステップ 2 を実行し、ステップ 3 をスキップします。出力が dcbxp egress-queuing と表示した場合、ステップ 2 をスキップし、ステップ 3 を実行します。
		これは、ポート チャネルで LLDP を設定する前に必要です。

ステップ 2	必須: **no lldp tlv-select dcbxp**
	例：
	`switch(config)# no lldp tlv-select dcbxp`
	`switch(config)#`
		DCBXP TLV をグローバルに無効にします。

ステップ 3	必須: **no lldp tlv-select dcbxp egress-queuing**
	例：
	`switch(config)# no lldp tlv-select dcbxp egress-queuing`
	`switch(config)#`
		DCBXP TLV をグローバルに無効にします。

ステップ 4	必須: [**no**] **lldp port-channel**
	例：
	`switch(config)# lldp port-channel`
	`switch(config)#`
		すべてのポート チャネルの LLDP 送受信をグローバルに有効または無効にします。

	コマンドまたはアクション	目的

ステップ 5	**interface port-channel** [*port-channel-number* | *port-channel-range*]
	例：
	`switch(config)# interface port-channel 3`
	`switch(config-if)#`
	例：
	複数のポートチャネルでLLDPを設定する場合は、ポート チャネル番号の範囲を入力します。
	`switch(config)# interface port-channel 1-3`
	`switch(config-if-range)#`
		LLDP を有効にするインターフェイス ポート チャネルを指定し、インターフェイス設定モードを開始します。
		LLDP を有効にするインターフェイス ポートチャネル範囲を指定し、インターフェイス範囲設定モードを開始します。

ステップ 6	（任意） [**no**] **lldp transmit** 例： `switch(config-if)# lldp transmit`	ポート チャネルまたはポート チャネルの範囲で LLDP パケットの送信を無効（または有効）にします。 （注） このポートチャネルでのLLDPパケットの送信は、ステップ 3 の **lldp port-channel** コマンドを使用して有効になりました。このオプションは、この特定のポート チャネルの機能を無効にします。
ステップ 7	（任意） [**no**] **lldp receive** 例： `switch(config-if)# lldp receive`	ポート チャネルまたはポート チャネルの範囲での LLDP パケットの受信を無効（または有効）にします。 （注） このポートチャネルでのLLDPパケットの受信は、ステップ 3 の **lldp port-channel** コマンドを使用して有効になりました。このオプションは、この特定のポート チャネルの機能を無効にします。
ステップ 8	（任意） **show lldp interface port-channel**l *port-channel-number* 例： `switch(config-if)# show lldp interface port-channel 3`	ポートチャネル上の LLDP 設定を表示します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

LLDP オプション パラメータの設定

LLDP の更新頻度、受信デバイスが情報を破棄するまでに保持している時間、および初期化の遅延時間を設定できます。TLV を選択して、LLDP パケットに含まれるようにすることもできます。

手順の概要

1. **configure terminal**
2. （任意）[**no**] **lldp holdtime** *seconds*
3. （任意）[**no**] **lldp reinit** *seconds*
4. （任意）[**no**] **lldp timer** *seconds*
5. （任意）**show lldp timers**
6. （任意）[**no**] **lldp tlv-select** *tlv*
7. （任意）**show lldp tlv-select**
8. （任意）**copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

ステップ 2	（任意）[**no**] **lldp holdtime** *seconds* 例： `switch(config)# lldp holdtime 200`	ユーザのデバイスから送信された情報が、受信側デバイスで廃棄されるまでに保持される時間を秒単位で指定します。 値の範囲は 10 ～ 255 秒で、デフォルト値は 120 秒です。
ステップ 3	（任意）[**no**] **lldp reinit** *seconds* 例： `switch(config)# lldp reinit 5`	任意のインターフェイス上で LLDP を初期化する際の遅延時間を秒単位で指定します。 指定できる範囲は 1 ～ 10 秒です。デフォルトは 2 秒です。
ステップ 4	（任意）[**no**] **lldp timer** *seconds* 例： `switch(config)# lldp timer 50`	LLDP アップデートの送信頻度を秒単位で設定します。 値の範囲は 5 ～ 254 秒で、デフォルト値は 30 秒です。

	コマンドまたはアクション	目的
ステップ 5	（任意） **show lldp timers** 例： `switch(config)# show lldp timers`	LLDP の保持時間、遅延時間、更新頻度の設定を表示します。

ステップ 6	（任意）[**no**] **lldp tlv-select** *tlv*
	例：
	`switch(config)# lldp tlv-select system-name`
	`switch(config)# lldp tlv-select port link-aggregation`
		LLDP パケットで送受信する TLV を指定します。使用できる TLV は、管理アドレス、ポート詳細、ポート vlan、システム機能、システム詳細、[ポート vlan 名、ポートリンク集約、ポート最大フレームサイズ（port vlan-name, port link-aggregation, port max-framesize,）]およびシステム名です。使用できるすべての TLV はデフォルトでイネーブルになっています。
		TLV を無効化にするには、コマンドの no 形式を使用します。

ステップ 7	（任意） **show lldp tlv-select** 例： `switch(config)# show lldp tlv-select`	LLDP TVL コンフィギュレーションを表示します。
ステップ 8	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

LLDP 設定の確認

LLDP 設定を表示するには、次のタスクのうちのいずれかを実行します。

コマンド	目的
show running-config lldp	LLDP のグローバル コンフィギュレーションを表示します。
show lldp interface *interface slot/port*	LLDP のインターフェイス コンフィギュレーションを表示します。
show lldp timers	LLDP の保持時間、遅延時間、更新頻度の設定を表示します。
show lldp tlv-select	LLDP TVL コンフィギュレーションを表示します。

コマンド	目的
show lldp neighbors {**detail** | **interface** *interface slot*/*port*}	LLDP ネイバーのデバイス ステータスを表示します。 （注） 隣接スイッチがスイッチ MAC をアドバタイズする場合、この show コマンドはスイッチ MAC を表示し、ポート MAC をアドバタイズする場合、show コマンドはポート MAC を表示します。
show lldp traffic	LLDP カウンタ（デバイスによって送信および受信された LLDP パケットの数、破棄されたパケットの数、未確認 TLV の数など）を表示します。
show lldp traffic interface *interface slot*/*port*	インターフェイス上で送信および受信された LLDP パケットの数を表示します。
show qos dcbxp *interface slot*/*port*	特定のインターフェイスの DCBXP 情報を表示します。

LLDP の統計を消去するには、**clear lldp counters** コマンドを使用します。

LLDP の設定例

次に、1 つのデバイス上での LLDP のイネーブル化、一部のインターフェイス上での LLDP のディセーブル化、オプション パラメータ（保持時間、遅延時間、更新頻度など）の設定、およびいくつかの LLDP TLV のディセーブル化の例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# feature lldp
switch(config)# interface ethernet 7/9
switch(config-if)# no lldp transmit
switch(config-if)# no lldp receive
switch(config-if)# exit
switch(config)# interface ethernet 7/10
switch(config-if)# no lldp transmit
switch(config-if)# no lldp receive
switch(config-if)# exit
switch(config)# lldp holdtime 200
switch(config)# lldp reinit 5
switch(config)# lldp timer 50
switch(config)# no lldp tlv-select port-vlan
switch(config)# no lldp tlv-select system-name
```

第 25 章

NetFlow の設定

この章では、Cisco NX-OS デバイス上で NetFlow 機能を設定する方法について説明します。

この章は、次の内容で構成されています。

• NetFlow の構成：入力および出力 （503 ページ）

NetFlow の構成：入力および出力

Cisco NX-OS リリース 10.4(1)F 以降、NetFlow は次の手順で構成されます。

入力 NetFlow の構成

入力 NetFlow について

入力 NetFlow は入力 IP パケットについてパケット フローを識別し、各パケット フローに基づいて統計情報を提供します。NetFlow のためにパケットやネットワーキング デバイスを変更する必要はありません。

入力 NetFlow ではフローを使用して、アカウンティング、ネットワーク モニタリング、およびネットワーク プランニングに関連する統計情報を提供します。フローは送信元インターフェイス（VLAN 向け）に届く単方向のパケット ストリームで、キーの値は同じです。キーは、パケット内のフィールドを識別する値です。フローを作成するには、フロー レコードを使用して、フロー固有のキーを定義します。

Cisco NX-OS は、ネットワーク異常とセキュリティ問題の高度な検出を有効にする Flexible NetFlow 機能をサポートします。フレクシブル NetFlow 機能を使用すると、大量の定義済みフィールドの集合からキーを選択することで、そのアプリケーションに最適なフロー レコードを定義できます。

1 つのフローと見なされるパケットでは、すべてのキー値が一致している必要があります。フローは、設定したエクスポート レコード バージョンに基づいて、関係のある他のフィールドを集めることもあります。フローは NetFlow キャッシュに格納されます。

フロー用に NetFlow が収集したデータをエクスポートするには、フロー エクスポータを使用し、このデータを Cisco Stealthwatch などのリモート NetFlow コレクタにエクスポートします。

Cisco NX-OS は次の状況で、NetFlow エクスポート用のユーザ データグラム プロトコル（UDP）データグラムの一部としてフローをエクスポートします。

- フローはフロー タイムアウト値に従って定期的にエクスポートされます。設定されていない場合、デフォルトは 10 秒です。
- ユーザがフローの強制的エクスポートを行った。

フロー レコードによってフロー用に収集するデータのサイズが決まります。フローモニタで、フロー レコードおよびフロー エクスポータを NetFlow キャッシュ情報と結合します。

Cisco NX-OS は NetFlow 統計を集計し、インターフェイスまたはサブインターフェイス上のすべてのパケットを分析します。

デュアルレイヤ NetFlow の実装

他の Cisco Nexus プラットフォームとは異なり、Cisco Nexus 9000 シリーズ スイッチは、NetFlow 処理を次の 2 つのレイヤに分離します。

- 第 1 レイヤは、ラインレート トラフィックのパケット単位の可視性をサポートします。パケットをサンプリングして統計的に分析する必要はありません。代わりに、パケットをライン レートで処理および集約できます。
- 2 番目のレイヤは、大規模なフローの収集を可能にします。フローを失うことなく何十万ものフローを維持でき、定期的に外部コレクタにエクスポートします。

フロー レコード

フロー レコードでは、パケットを識別するために NetFlow で使用するキーとともに、NetFlow がフローについて収集する関連フィールドを定義します。キーと関連フィールドを任意の組み合わせで指定して、フローレコードを定義できます。Cisco NX-OS は、様々なキーセットをサポートしています。フローレコードでは、フロー単位で収集するカウンタのタイプも定義します。32 ビットまたは 64 ビットのパケット カウンタまたはバイト カウンタを設定できます。

キー フィールドは、**match** キーワードで指定されます。対象フィールドとカウンタは **collect** キーワードで指定されます。

Cisco NX-OS では、フロー レコードの作成時に次の match フィールドをデフォルトとして使用できます。

- match interface input
- match flow direction

フロー エクスポータ

フローエクスポータでは、NetFlow エクスポート パケットに関して、ネットワーク層およびトランスポート層の詳細を指定します。フロー エクスポータで設定できる情報は次のとおりです。

- エクスポート宛先 IP アドレス

- 送信元インターフェイス
- UDP ポート番号（NetFlow コレクタが NetFlow パケットをリスニングするところ）：デフォルト値は 9995 です。

（注） NetFlow エクスポート パケットでは、送信元インターフェイスに割り当てられた IP アドレスを使用します。送信元インターフェイスを設定しない場合、フローエクスポータはエクスポートする予定のフローをドロップします。[Netflowエクスポータの送信元インターフェイスと接続先 IP は、同じ VRF を使用する必要があります。（The Netflow Exporter source interface and destination IP must use the same VRF.）]

Cisco NX-OS は、タイムアウトが発生するたびにデータを NetFlow コレクタへエクスポートします。キャッシュをフラッシュし、フローを強制的にエクスポートするには、フラッシュ キャッシュ タイムアウトを設定できます（**flow timeout** コマンドを使用）。

エクスポート形式

Cisco NX-OS は、バージョン 9 のエクスポート形式をサポートします。この形式は、古いバージョン 5 のエクスポート形式よりも効率的なネットワーク使用率をサポートし、IPv6 およびレイヤ 2 フィールドをサポートします。さらに、バージョン 9 エクスポート形式は、NetFlow コレクタで完全な 32 ビット SNMP ifIndex 値をサポートします。

レイヤ 2 NetFlow キー

フレクシブル NetFlow レコード内でレイヤ 2 キーを定義できます。このレコードを使用して、レイヤ 2 インターフェイスのフローをキャプチャできます。レイヤ 2 のキーは次のとおりです。

- 送信元および宛先 MAC アドレス
- 送信元 VLAN ID
- イーサネット フレームのイーサネット タイプ

受信方向については、次のインターフェイスに対してレイヤ 2 NetFlow を適用できます。

- アクセス モードのスイッチ ポート
- トランク モードのスイッチ ポート
- レイヤ 2 のポート チャネル

（注） Layer 2 NetFlow を VLAN、送信インターフェイス、またはレイヤ 3 インターフェイス（VLAN インターフェイスなど）に適用できます。

フロー モニタ

フロー モニタは、フロー レコードおよびフロー エクスポータを参照します。フロー モニタはインターフェイスに適用します。

NetFlow 出力インターフェイス

スイッチの NetFlow 出力インターフェイスには、次の機能があります。

- **show flow cache** コマンドの NetFlow は `output_if_id` を表示し、出力インターフェイスをコレクタにエクスポートします。
- Cisco Nexus プラットフォーム スイッチの NetFlow 出力インターフェイスは、IPv4 と IPv6 の両方のトラフィック フローをサポートします。ただし、Cisco Nexus 9500 プラットフォーム スイッチの NetFlow 出力インターフェイスは、IPv4 トラフィック フローでのみサポートされ、IPv6 トラフィック フローではサポートされません。
- **show flow cache** コマンドは、`output_if_id` を `0x0` として表示します。ただし、Cisco NX-OS リリース 10.3(3)F 以降では、宛先 IP アドレスが解決できない、またはルーティング テーブルに登録されていない、または受信したパケットが制御パケットであった場合に限り、**show flow cache** コマンドは `output_if_id` を `0x0` として表示します。
- NetFlow は、宛先インターフェイスとしてネクストホップを持つ IPv4/IPv6 着信トラフィック フローのコレクタへの出力インターフェイスのエクスポートをサポートします。`InputInt` および `OutputInt` の NetFlow エクスポート形式は、NetFlow コレクタで完全な 32 ビット SNMP `ifIndex` 値をサポートします。
- NetFlow 出力インターフェイスは、MPLS、VXLAN、GRE などのトンネル トラフィック フローではサポートされません。
- NetFlow 出力インターフェイスの例の詳細については、入力 NetFlow の表示例 （524 ページ） を参照してください。

高可用性

Cisco NX-OS は NetFlow のステートフル リスタートをサポートします。リブート後、Cisco NX-OS は実行コンフィギュレーションを適用します。

フローキャッシュは再起動で保持されず、再起動中にソフトウェアに送信されるパケットは処理されません。

入力 NetFlow の前提条件

入力 NetFlow の前提条件は、次のとおりです。

- 使用しているデバイスで必要とされるリソースを正しく理解していること。NetFlow はメモリと CPU リソースを消費するからです。

入力 NetFlow に関する注意事項および制約事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

入力 NetFlow に関する構成の注意事項および制約事項は、次のとおりです。

- Cisco Nexus 9300-FX プラットフォーム スイッチに対して、レイヤ 2 NetFlow に対してすでに設定されているポート チャネルにメンバを追加すると、NetFlow の設定が削除され、ポート チャネルのレイヤ 2 設定が追加されます。
- NetFlow はトンネル インターフェイスではサポートされていません。
- NetFlowは、CPU で送信されるパケットではサポートされません。
- 入力 NetFlow のみがサポートされます。出力 NetFlow はサポートされていません。
- フローキャッシュは、レイヤ2、IPv4、IPv6などのフロータイプごとにクリアできます。フロー モニタごとにクリアすることはできません。
- Nexus 9000 スイッチでは、NetFlow は ICMP フロー情報を収集してコレクタに送信します。ICMP タイプとコードはパケットに本質的に含まれます。NetFlow またはフローエクスポート レコード内の ICMP パケットは、TCP や UDP などの従来の送信元および宛先ポートを使用しません。代わりに、エクスポータは ICMP タイプとコードを SPORT（送信元ポート）や DPORT（宛先ポート）などのフィールドにエンコードすることがよくあります。次に例を示します。
  - ICMP Echo Request: SPORT=2048, DPORT=0
  - ICMP Echo Reply: SPORT=0, DPORT=0
  - ICMP Time Exceeded: SPORT=2816, DPORT=0
- フロー収集は ARP トラフィックに対して実行されません。
- NetFlowデータエクスポート（NDE）では、送信元インターフェイスを設定する必要があります。送信元インターフェイスを設定しない場合、フローエクスポータはエクスポートする予定のフローをドロップします。
- レイヤ 2 スイッチド フロー モニタは、レイヤ 2 インターフェイスにのみ適用されます。IP および IPv6 フロー モニタは、VLAN、SVI、レイヤ 3 ルーテッド インターフェイス、またはサブインターフェイスに適用できます。
- レイヤ 2 インターフェイスをレイヤ 3 インターフェイスへ変更するか、レイヤ 3 インターフェイスをレイヤ2インターフェイスへ変更すると、ソフトウェアで、インターフェイスからレイヤ 2 の NetFlow 設定が削除されます。
- 同じフローモニタをVLANおよびレイヤ3インターフェイス（物理レイヤ3インターフェイス、SVI インターフェイス、またはレイヤ 3 サブインターフェイスなど）と共有することはできません。ACLは異なるため共有できないため、VLANとレイヤ3インターフェイ

スを区別する必要があります。これらは2つの異なるプロファイルとして扱う必要があります。

- ロールバック中、ハードウェアでプログラムされているレコードを変更しようとすると、ロールバックは失敗します。
- 入力 NetFlow 機能の制限は次のとおりです。
  - MPLS データパスの NetFlow はサポートされていません
  - OIF（出力インターフェイス）を含む VXLAN トラフィックの NetFlow がサポートされています。内部フローの詳細をキャプチャするには、VXLAN トランジット ノードで feature nv overlay コマンドを有効にする必要があります。そうでない場合、NetFlow は外部ヘッダー情報のみをキャプチャします。
  - VXLAN カプセル化ノードの OIF は 0x0 と表示されます。
  - NetFlow は、ループバックおよびスイッチ管理インターフェイスではサポートされません。
- VXLAN 環境の NetFlow には、次のガイドラインと制限事項が適用されます。
  - NetFlow は、VXLAN VTEP の SVI および非アップリンク L3 インターフェイスでサポートされます。これには L3VNI SVI は含まれません。
  - NetFlow は、VXLAN VTEP のアップリンク インターフェイスではサポートされません。
  - マルチサイト境界ゲートウェイでの NetFlow はサポートされていません。
  - VXLAN ファブリックを介して到達可能な NetFlow コレクタがサポートされています。
- Cisco NX-OS リリース 9.2(1) 以降：
  - FEX レイヤ 3 ポートの NetFlow は Cisco Nexus 9300 EX と 9300 FX プラットフォーム スイッチでサポートされています。
  - Cisco Nexus 9300-EX プラットフォーム スイッチで NetFlow CE がサポートされています。

（注） すべての EX タイプのプラットフォーム スイッチ（Cisco Nexus 9700-EX ライン カードを含む）では、CE NetFlow は非 IPv4 および IPv6 トラフィック フローの CE フローレコードのみをキャプチャします。FX および FX2 タイプのプラットフォーム スイッチとライン カードでは、**mac packet-classify** がインターフェイスに適用されている限り、IP フローの CE フロー データをキャプチャできます。

- Cisco NX-OS リリース 9.2(2) 以降、Cisco Nexus 9300-FX スイッチは NetFlow データ エクスポート（NDE）の OUTPUT_SNMP フィールドの収集をサポートしています。他の Cisco Nexus 9000 プラットフォーム スイッチまたは Cisco Nexus ライン カードは、OUTPUT_SNMP フィールドの収集をサポートしていません。
- Cisco NX-OS リリース9.2(2) 以降では、NetFlow はCisco Nexus 9700-EX ライン カードと FM-E モジュールを搭載した Cisco Nexus 9500 プラットフォーム スイッチでサポートされます。
- 入力 NetFlow は、Cisco Nexus 92348GC-X プラットフォーム スイッチではサポートされていません。
- Cisco Nexus 9300-EX プラットフォーム スイッチの場合、VLAN または SVI に適用されたフロー モニタは、スイッチド トラフィックとルーテッド トラフィックの両方のフローを収集できます。Cisco Nexus 9300-FX プラットフォーム スイッチの場合、NetFlow VLAN はスイッチド トラフィックに対してのみサポートされ、NetFlow SVI はルーテッド トラフィックに対してのみサポートされます。
- Cisco Nexus 9300-EX プラットフォーム スイッチは、同じインターフェイスで NetFlow と SPAN を同時にサポートします。この機能は、SPAN および sFlow の代わりに使用できます。
- Cisco Nexus 9300-EX/FX プラットフォーム スイッチ、および EX/FX モジュールを搭載した Cisco Nexus 9500 プラットフォーム スイッチでは、SPAN と sFlow の両方を同時に有効にすることはできません。一方がアクティブな場合、もう一方は有効にできません。ただし、Cisco Nexus 9300-EX/FX/FX2 および EX モジュールを搭載した Cisco Nexus 9500 プラットフォーム スイッチでは、NetFlow と SPAN の両方を同時に有効にすることができ、sFlow と SPAN を使用する代わりに実行可能です。

（注） Cisco Nexus 9300-FX2 プラットフォーム スイッチは、sFlow と SPAN の共存をサポートします。

- Cisco Nexus 9300-EX プラットフォーム スイッチでは、同じフロー モニタを VLAN と SVI に同時に接続することはできません。
- Cisco Nexus 9300-EX プラットフォーム スイッチには専用の TCAM があり、カービングは必要ありません。
- ing-netflow リージョンの TCAM カービング設定は、FX ライン カードでは実行できます。EX ライン カードでは、デフォルトの ing-netflow リージョン TCAM カービングが 1024 であり、それ以外の場合は設定できません。EX および FX ライン カードのポートの場合、ing-netflow リージョンの推奨最大値は 1024 です。
- ToS フィールドは、Cisco Nexus 9300-EX プラットフォーム スイッチではエクスポートされません。
- IP ToS に基づくレコード一致は、IPv6 フロー モニタではサポートされません。ToS 値は、トラフィックが保持する値に関係なく、コレクタで 0x0 として収集されます。

この制限は、次のプラットフォーム スイッチ ファミリに適用されます。

- Cisco Nexus 9300-EX
- Cisco Nexus 9300-FX
- Cisco Nexus 9300-FX2
- Cisco Nexus 9300-FX3
- Cisco Nexus 9300-GX
- EX または FX ライン カード搭載の Cisco Nexus 9500

- 次の注意事項は、EX および FX ライン カード搭載のすべての Cisco Nexus 9500 プラットフォーム スイッチに適用されます。

  FX ポートがすでに適用されている NetFlow 設定のトランクである場合、EX ポートをトランクとして設定しても、サポートされていない EX NetFlow 設定は FX ポート トランクから削除されません。たとえば、3 つ以上の異なる IPv4 フロー モニタを FX ポート トランクに適用し、EX ポートが同じトランクに追加された場合、EX ポートの制限のみであるため、2 つのモニタを超えるトランクの設定は自動的に削除されません。この設定では、EX トランク ポートの 2 つのモニタを超えるフローはレポートされないため、EX ポートと FX ポートの両方が同じトランクに存在する可能性があるモジュラ スイッチでは、プロトコルごとに 2 つのモニタ（v4/v6/CE）のみを使用することを推奨します。

-
- **record netflow ipv4 original-input**、**record netflow ipv4 original-output**、および **record netflow layer2-switched input** コマンドは、Cisco NX-OS リリース 9.3(1) ではサポートされていません。
- Cisco NX-OS リリース 9.3(3) 以降、NetFlow に関する次の無停止インサービス ソフトウェア アップグレード（ND ISSU）の制限がすべての Cisco Nexus 9000 シリーズ スイッチに適用されます。
  - ND ISSU の実行中、2 分間のエクスポート損失が予想されます。
  - ND ISSU 中は、管理インターフェイスの送信元ポートを持つエクスポータはサポートされません。エクスポート損失は、管理インターフェイスが起動するまで予想されます。
- Cisco NX-OS リリース 9.3(3) 以降、入力 NetFlow は Cisco Nexus 9300-GX プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 9.3（4）以降では、次の RTP / NetFlow モニタリング制限が存在します。

  RTP モニタリング機能は、スイッチのすべてのインターフェイスで RTP フローのモニタをイネーブルにし、**show flow rtp detail** コマンド出力で報告します。RTP フローは、16384 ー 32767 の範囲内の送信元ポートを持つ UDP フローです。RTP モニタリングがイネーブルになっているスイッチ インターフェイスに NetFlow モニタが接続されている場合、そのインターフェイス上のすべてのトラフィック/フロー（RTP フローを含む）が **show flow**

cache コマンドの出力で報告されます。RTP フローは、**show flow rtp detail** コマンドの出力に表示されなくなります。接続されたモニタが削除されると、RTP フローが **show flow rtp detail** コマンド出力で再度報告されます。

この制限は、次のスイッチに影響します。

- Cisco Nexus 9336C-FX2
- Cisco Nexus 93240YC-FX2
- Cisco Nexus 9348GC-FXP
- Cisco Nexus 93180YC-FX
- Cisco Nexus 93108TC-FX
- Cisco Nexus 9316D-GX
- Cisco Nexus 93600CD-GX
- Cisco Nexus 9364C-GX
- 9636C-RX ライン カードを搭載した Cisco Nexus 9504、9508 および 9516 スイッチ

- FM-E、FM-E2、および FM-E3 モジュールを搭載した Cisco Nexus 9500 プラットフォーム スイッチおよび Cisco Nexus 9300-FX/FX3 スイッチは、NetFlow 出力インターフェイスには機能をサポートします。ただし、9300-EX および 9500-EX プラットフォーム スイッチの出力インターフェイスはサポートされません。
- 入力 NetFlow は、EX、FX、および GX 混合シャーシの Cisco Nexus 9500 プラットフォーム スイッチでサポートされます。EX、FX、および GX 混合シャーシの Cisco Nexus 9500 プラットフォーム スイッチでは、SPAN を NetFlow と同時に使用できます。Cisco Nexus 9500-GX プラットフォーム スイッチは、sFlow 機能を組み合わせた SPAN をサポートしていません。
- Cisco Nexus 3232C および 3264Q スイッチは、NetFlow をサポートしていません。
- Cisco NX-OS リリース 10.1(2) 以降、Netflow は N9K-X9716D-GX ライン カードでサポートされます。
- この機能をサポートするプラットフォームでのみ NetFlow を有効にします。
- **match ip tos** コマンドはフロー レコード設定オプションにありますが、機能はサポートされていません。
- Cisco NX-OS リリース 10.2(1)F 以降、レイヤ 2 インターフェイス上のレイヤ 3 NetFlow は、Cisco Nexus 9300-EX、9300-FX、9300-FX2、9300-FX3、9300-GX、および 9300-GX2 プラットフォーム、9500-EX LC および 9500-FX LC でサポートされます。注意事項と制約事項は次のとおりです。
  - レイヤ 3 フロー モニタまたはレイヤ 2 フロー モニタのいずれかをレイヤ 2 インターフェイスに接続できます（両方は接続できません）。

- フローモニタがすでにレイヤ3インターフェイスに接続されている場合、同じフローモニタをレイヤ 2 インターフェイスに接続することはできません。
- レイヤ 3 フロー モニタがレイヤ 2 インターフェイスに適用されている場合、**mac-packet-classify** コマンドはサポートされません。

- Cisco NX-OS Release 10.3(3)F 以降では、NetFlow Output_if_id は Cisco 9300-FX2、9300-GX、および 9300-GX2 スイッチおよび 9500-GX ライン カードでサポートされます。
  - ユニキャスト フローには、output_if_id が 0x0 と表示されている場合は、宛先 IP が解決されていないか、ルーティング テーブルまたは ARP テーブルで使用できないことを意味します。
  - コントロールプレーン宛ての受信パケットの output_if_id は 0x0 として表示されます。このようなトラフィックの例としては、ルーティングプロトコルまたは ICMP パケットがあります。
  - Cisco Nexus 9500 ライン カードの IPv6 フローの場合、output_if_id はハードウェアの制限によりサポート対象外と表示されます。
  - データプレーン マルチキャスト トラフィックの output_if_id は、0x0 として表示されます。

- Cisco NX-OS リリース 10.4(1)F 以降、入力 NetFlow は N9K-C9332D-H2R スイッチでサポートされます。次の制限が適用されます。
  - ドロップフローは、デフォルトではサポートされていません。ドロップフローを有効にするには、**hardware flow-table collect-drop-reason** コマンドを構成します。

（注） 確認済みの NetFlow のスケール数については、『Cisco Nexus 9000 Series NX-OS Verified Scalability Guide』を参照してください。

入力 NetFlow の構成

NetFlow を設定する手順は、次のとおりです。

手順

ステップ 1 入力 NetFlow 機能を有効化します。

ステップ 2 フローにキーおよびフィールドを指定することによって、フロー レコードを定義します。

ステップ 3 エクスポート フォーマット、プロトコル、宛先、およびその他のパラメータを指定することによって、任意でフロー エクスポータを定義します。

ステップ 4 フロー レコードおよびフロー エクスポータに基づいて、フロー モニタを定義します。

ステップ 5 送信元インターフェイス、サブインターフェイス、またはVLANインターフェイスにフローモニタを適用します。

入力 NetFlow 機能の有効化

フローを設定するには、先に NetFlow をグローバルで有効しておく必要があります。

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **feature netflow** 例： `switch(config)# feature netflow`	NetFlow 機能を有効にします。デフォルトではディセーブルになっています。 （注） N9K-T2 EoR を搭載した Cisco Nexus 9500 プラットフォーム スイッチは、NetFlow をサポートしていません。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

フロー レコードの作成

フロー レコードを作成し、照合するためのキー、および収集するための非キー フィールドをフロー内に追加します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的

ステップ 2	**flow record** *name*
	例：
	`switch(config)# flow record Test`
	`switch(config-flow-record)#`
		フロー レコードを作成し、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。フロー レコード名には最大 63 文字の英数字を入力できます。

ステップ 3	（任意） **description** *string* 例： `switch(config-flow-record)# description IPv4Flow`	最大 63 文字で、フロー レコードの説明を示します。
ステップ 4	（任意） **match** *type* 例： `switch(config-flow-record)# match transport destination-port`	一致キーを指定します。詳細については、match パラメータの指定 （514ページ）を参照してください。 （注） レイヤ4ポートデータをエクスポートするには、**match transport destination-port** および **match ip protocol** コマンドが必要です。
ステップ 5	（任意） **collect** *type* 例： `switch(config-flow-record)# collect counter packets`	コレクション フィールドを指定します。詳細については、collect パラメータの指定 （515 ページ）を参照してください。
ステップ 6	（任意） **show flow record** [*name*] [*record-name*] {**netflow-original** | **netflow protocol-port** | **netflow** {**ipv4** | **ipv6**} {**original-input** | **original-output**}} 例： `switch(config-flow-record)# show flow record netflow protocol-port`	NetFlow のフロー レコード情報を表示します。フロー レコード名には最大 63 文字の英数字を入力できます。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config-flow-record)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

match パラメータの指定

フロー レコードごとに、次の match パラメータを 1 つ以上設定する必要があります。

コマンド	目的
match datalink {**mac source-address** | **mac destination-address** | **ethertype** | **vlan**} **例:** `switch(config-flow-record)# match datalink ethertype`	レイヤ 2 属性をキーとして指定します。

コマンド	目的
match ip {**protocol** | **tos**} **例:** `switch(config-flow-record)# match ip protocol`	IP プロトコルまたは ToS フィールドをキーとして指定します。 （注） レイヤ 4 ポートデータをエクスポートするには、**match transport destination-port** および **match ip protocol** コマンドが必要です。 データは **show hardware flow ip** コマンドの出力に収集されて表示されますが、両方のコマンドを設定するまで収集とエクスポートは行われません。
match ipv4 {**destination address** | **source address**} **例:** `switch(config-flow-record)# match ipv4 destination address`	IPv4 送信元または宛先アドレスをキーとして指定します。
match ipv6 {**destination address** | **source address** | **flow-label** | **options**} **例:** `switch(config-flow-record)# match ipv6 flow-label`	IPv6 キーを指定します。
match transport {**destination-port** | **source-port**} **例:** `switch(config-flow-record)# match transport destination-port`	トランスポート送信元または宛先ポートをキーとして指定します。 （注） レイヤ 4 ポートデータをエクスポートするには、**match transport destination-port** および **match ip protocol** コマンドが必要です。 データは **show hardware flow ip** コマンドの出力に収集されて表示されますが、両方のコマンドを設定するまで収集とエクスポートは行われません。

collect パラメータの指定

フロー レコードごとに、次の collect パラメータを 1 つ以上設定する必要があります。

コマンド	目的
collect counter {**bytes** | **packets**} [**long**] **例:** `switch(config-flow-record)# collect counter packets`	フローからパケットベースまたはバイト カウンタを収集します。任意で、64 ビット カウンタを使用することを指定できます。
collect ip version **例:** `switch(config-flow-record)# collect ip version`	フローの IP バージョンを収集します。
collect timestamp sys-uptime {**first** | **last**} **例:** `switch(config-flow-record)# collect timestamp sys-uptime last`	フローの先頭または最終パケットに関するシステム稼働時間を収集します。
collect transport tcp flags **例:** `switch(config-flow-record)# collect transport tcp flags`	フローのパケットに対応する TCP トランスポート層フラグを収集します。

フロー エクスポータの作成

フロー エクスポータの設定では、フローに対するエクスポート パラメータを定義し、リモート NetFlow Collector への到達可能性情報を指定します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow exporter** *name*
	例：
	`switch(config)# flow exporter flow-exporter-one`
	`switch(config-flow-exporter)#`
		フロー エクスポータを作成し、フロー エクスポータ コンフィギュレーション モードを開始します。フロー エクスポータ名を最大 63 文字の英数字で入力できます。

ステップ 3	**destination** {*ipv4-address* | *ipv6-address*} [**use-vrf** *name*] 例： `switch(config-flow-exporter)# destination 192.0.2.1`	このフロー エクスポータの宛先 IPv4 または IPv6 アドレスを設定します。任意で、NetFlow Collector に到達するために使用する VRF を設定できます。VRF 名には最大 32 文字の英数字を入力できます。

	コマンドまたはアクション	目的
ステップ 4	**source** *interface-type name/port* 例： `switch(config-flow-exporter)# source ethernet 2/1`	設定された宛先で NetFlow Collector に到達するために使用するインターフェイスを指定します。
ステップ 5	（任意） **description** *string* 例： `switch(config-flow-exporter)# description exportversion9`	このフロー エクスポータについて説明します。説明には最大 63 文字の英数字を入力できます。
ステップ 6	（任意） **dscp** *value* 例： `switch(config-flow-exporter)# dscp 0`	DSCP（DiffServ コードポイント）値を指定します。範囲は 0 ～ 63 です。
ステップ 7	（任意） **transport udp** *port* 例： `switch(config-flow-exporter)# transport udp 200`	NetFlow Collector に到達するために使用する UDP ポートを指定します。範囲は 0 ～ 65535 です。 （注） UDP ポートを指定しない場合は、9995 がデフォルトとして選択されます。

ステップ 8	**version 9**
	例：
	`switch(config-flow-exporter)# version 9`
	`switch(config-flow-exporter-version-9)#`
		NetFlow エクスポート バージョンを指定します。フローエクスポータのバージョン 9 コンフィギュレーション サブモードを開始するには、バージョン 9 を選択します。

ステップ 9	（任意） **option** {**exporter-stats** | **interface-table**} **timeout** *seconds* 例： `switch(config-flow-exporter-version-9)# option exporter-stats timeout 1200`	フロー エクスポータの統計情報再送信タイマーを設定します。値の範囲は 1 ～ 86400 秒です。
ステップ 10	（任意） **template data timeout** *seconds* 例： `switch(config-flow-exporter-version-9)# template data timeout 1200`	テンプレートデータ再送信タイマーを設定します。値の範囲は 1 ～ 86400 秒です。
ステップ 11	（任意） **copy running-config startup-config** 例： `switch(config-flow-exporter-version-9)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

フロー モニタの作成

フロー モニタを作成して、フロー レコードおよびフロー エクスポータと関連付けることができます。1 つのモニタに属しているすべてのフローは、様々なフィールド上で照合するために関連するフロー レコードを使用します。データは指定されたフローエクスポータにエクスポートされます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow monitor** *name*
	例：
	`switch(config)# flow monitor flow-monitor-one`
	`switch(config-flow-monitor)#`
		フローモニタを作成し、フローモニタ コンフィギュレーション モードを開始します。フロー モニタ名を最大 63 文字の英数字で入力できます。

ステップ 3	（任意） **description** *string* 例： `switch(config-flow-monitor)# description IPv4Monitor`	このフローモニタについて説明します。説明には最大 63 文字の英数字を入力できます。
ステップ 4	（任意） **exporter** *name* 例： `switch(config-flow-monitor)# export v9`	フロー エクスポータとこのフロー モニタを関連付けます。エクスポータ名には最大 63 文字の英数字を入力できます。
ステップ 5	**record** *name* [**netflow-original** | **netflow protocol-port** | **netflow** {**ipv4** | **ipv6**} {**original-input** | **original-output**}] 例： `switch(config-flow-monitor)# record IPv4Flow`	フロー レコードを指定したフロー モニタと関連付けます。レコード名には最大 63 文字の英数字を入力できます。 （注） **record netflow ipv4 original-input**、**record netflow ipv4 original-output**、**record netflow layer2-switched input** は、Cisco NX-OS リリース 9.3(1) ではサポートされていません。
ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config-flow-monitor)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

インターフェイスへのフロー モニタの適用

フロー モニタは入力インターフェイスに適用できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface vlan** *vlan-id*
	例：
	`switch(config)# interface vlan 10`
	`switch(config-if)#`
		VLAN 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 3	**ip flow monitor** {**ipv4** | **ipv6** | **layer-2-switched**} **input** 例： `switch(config-if)# ip flow monitor ipv4 input`	入力パケットのインターフェイスに、IPv4、IPv&、またはレイヤ 2 スイッチ フロー モニタを関連付けます。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

VLAN 上でのブリッジ型 NetFlow の設定

VLAN のレイヤ 2 スイッチド パケットでレイヤ 3 データを収集するために、VLAN にフロー モニタを適用できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**vlan configuration** *vlan-id*
	例：
	`switch(config)# vlan configuration 30`
	`switch(config-vlan-config)#`
		VLAN コンフィギュレーション モードを開始します。VLAN ID の範囲は 1 ～ 3967 または 4048 ～ 4093 です。
		（注）

	コマンドまたはアクション	目的
		VLAN コンフィギュレーション モードでは、作成とは無関係に VLAN を設定できます。これは、VTP クライアントのサポートに必要です。
ステップ 3	{**ip** | **ipv6**} **flow monitor** *name* 例： `switch(config-vlan-config)# ip flow monitor testmonitor`	入力パケットのフロー モニタを VLAN に関連付けます。フロー モニタ名を最大 63 文字の英数字で入力できます。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config-vlan-config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

レイヤ 2 NetFlow キーの設定

フレクシブル NetFlow レコード内でレイヤ 2 キーを定義できます。このレコードを使用して、レイヤ 2 インターフェイスのフローをキャプチャできます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow record** *name*
	例：
	`switch(config)# flow record L2_record`
	`switch(config-flow-record)#`
		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。フロー レコードの設定の詳細については、フロー レコードの作成 （513 ページ） を参照してください。

ステップ 3	**match datalink** {**mac source-address** | **mac destination-address** | **ethertype** | **vlan**} 例： `switch(config-flow-record)# match datalink ethertype`	レイヤ 2 属性をキーとして指定します。

ステップ 4	**exit**
	例：
	`switch(config-flow-record)# exit`
	`switch(config)#`
		フロー レコード コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的

ステップ 5	**interface** {**ethernet** *slot/port* | **port-channel** *number*}
	例：
	`switch(config)# interface Ethernet 6/3`
	`switch(config-if#)`
		インターフェイス設定モードを開始します。インターフェイス タイプは、物理的なイーサネット ポートまたはポート チャネルを指定できます。

ステップ 6	**switchport** 例： `switch(config-if)# switchport`	インターフェイスをレイヤ 2 の物理インターフェイスに変更します。スイッチ ポートの設定に関する詳細については、「Cisco Nexus 9000 シリーズ NX-OS レイヤ 2 スイッチング設定ガイド」を参照してください。
ステップ 7	**mac packet-classify** 例： `switch(config-if)# mac packet-classify`	パケットの MAC 分類を強制します。 このコマンドの使用に関する詳細については、「Cisco Nexus 9000 シリーズ NX-OS セキュリティ設定ガイド」を参照してください）。 （注） フローを検出するためにこのコマンドを使用する必要があります。
ステップ 8	**layer2-switched flow monitor** *flow-name* **input** 例： `switch(config-if)# layer2-switched flow monitor L2_monitor input`	フロー モニタをスイッチ ポートの入力パケットに関連付けます。フロー モニタ名を最大 63 文字の英数字で入力できます。
ステップ 9	（任意） **show flow record netflow layer2-switched input** 例： `switch(config-if)# show flow record netflow layer2-switched input`	レイヤ 2 NetFlow のデフォルト レコードの情報を表示します。
ステップ 10	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

レイヤ 2 インターフェイスでのレイヤ 3 NetFlow の設定

レイヤ 2 インターフェイスでレイヤ 3 フロー情報をキャプチャするために、レイヤ 2 インターフェイスでレイヤ 3 フロー モニタを定義できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow record** *name*
	例：
	`switch(config)# flow record L3_record`
	`switch(config-flow-record)#`
		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。フロー レコードの設定の詳細については、フロー レコードの作成（513 ページ）を参照してください。

ステップ 3	**interface** {**ethernet** *slot/port* | **port-channel** *number*}
	例：
	`switch(config)# interface Ethernet 6/3`
	`switch(config-if#)`
		インターフェイス設定モードを開始します。インターフェイス タイプは、物理的なイーサネット ポートまたはポート チャネルを指定できます。

ステップ 4	**switchport** 例： `switch(config-if)# switchport`	インターフェイスをレイヤ 2 モードに変更します。スイッチ ポートの設定に関する詳細については、「Cisco Nexus 9000 シリーズ NX-OS レイヤ 2 スイッチング設定ガイド」を参照してください。
ステップ 5	**ip flow monitor** *flow-name* **input** 例： `switch(config-if)# ip flow monitor v41 input`	フロー モニタをスイッチ ポートの入力パケットに関連付けます。フロー モニタ名を最大 63 文字の英数字で入力できます。
ステップ 6	**ipv6 flow monitor** *flow-name* **input** 例： `switch(config-if)# ipv6 flow monitor v61 input`	IPv6 フロー モニタをスイッチ ポートの入力パケットに関連付けます。フロー モニタ名を最大 63 文字の英数字で入力できます。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

NetFlow タイムアウトの設定

任意で、システム内のすべてのフローに適用されるグローバルなNetFlowタイムアウトを設定できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow timeout** *seconds* 例： `switch(config)# flow timeout 30`	フラッシュ タイムアウト値を秒単位で設定します。範囲は 5 ～ 60 秒です。デフォルト値は 10 秒です。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

入力 NetFlow 構成の確認

入力 NetFlow 構成を表示するには、次のタスクのうちのいずれかを実行します。

コマンド	目的
show flow cache [**ipv4** | **ipv6** | **ce**]	入力 NetFlow IP フローに関する情報を表示します。 （注） このコマンドは、EOR スイッチでは有効ではないように見え、フローは表示されません。EOR スイッチでこのコマンドを表示するには、 **attach mod** *x* コマンドを使用してモジュールにアタッチします。または、**slot** *x***quoted "show flow cache"** コマンドを使用してこのコマンドをチェックします。ここで、*x* は入力 NetFlow のモジュール番号です。
show flow exporter [*name*]	入力 NetFlow のフロー エクスポータ情報と統計情報を表示します。フロー エクスポータ名を最大 63 文字の英数字で入力できます。
show flow interface [*interface-type slot/port*]	入力 NetFlow インターフェイスに関する情報を表示します。

コマンド	目的
show flow record [*name*]	入力 NetFlow のフロー レコード情報を表示します。フロー レコード名には最大 63 文字の英数字を入力できます。
show flow record netflow layer2-switched input	レイヤ 2 入力 NetFlow 構成の情報を表示します。
show running-config netflow	現在デバイスにある NetFlow 設定を表示します。

入力 NetFlow のモニタリング

NetFlow の統計情報を表示するには、**show flow exporter** コマンドを使用します。NetFlow エクスポータの統計情報を消去するには、**clear flow exporter** コマンドを使用します。

入力 NetFlow の表示例

IPv4 の **show flow cache** コマンドの出力には、次のように表示されます。

```
show flow cache
IPV4 Entries
SIP           DIP          BD ID  S-Port  D-Port Protocol Byte Count  Packet Count  TCP
FLAGS  TOS  if_id      output_if_id  flowStart flowEnd
10.10.30.4    30.33.1.2    1480   30000   17998  17       683751850   471553        0x0
      0x0  0x90105c8  0x1a005000    14096494  14153835
30.33.1.2     10.10.39.4  4145   30000   18998  17       43858456    30164         0x0
      0x0  0x1a005000 0x1a006600    14096477  14099491
10.10.29.4    30.33.1.2    1479   30000   17998  17       683751850   471553        0x0
      0x0  0x90105c7  0x1a005000    14096476  14153817
10.10.7.4     30.33.1.2    1457   30000   17998  17       683753300   471554        0x0
      0x0  0x90105b1  0x1a005000    14096481  14153822
30.33.1.2     10.10.42.4  4145   30000   18998  17       95289344    65536         0x0
      0x0  0x1a005000 0x1a006600    14112551  14119151
10.10.49.4    30.33.1.2    1499   30000   17998  17       683753300   471554        0x0
      0x0  0x90105db  0x1a005000    14096486  14153827
```

入力 NetFlow の構成例

この例では、IPv4 に対して出力 NetFlow エクスポータを構成する方法を示します。

```
feature netflow
flow exporter ee
 destination 171.70.242.48 use-vrf management
 source mgmt0
 version 9
  template data timeout 20
flow record rr
 match ipv4 source address
 match ipv4 destination address
 collect counter bytes
 collect counter packets
flow monitor foo
 record rr
```

```
 exporter ee
interface Ethernet2/45
 ip flow monitor foo input
 ip address 10.20.1.1/24
 no shutdown
```

出力 NetFlow の構成

出力 NetFlow について

出力 NetFlow は発信 IP パケットとレイヤ 2 フローについてパケット フローを識別し、これらのパケット フローに基づいて統計情報を提供します。Cisco Nexus Cloudscale 9000 スイッチでは、出力 NetFlow は入力パイプラインと入力 TCAM カービング egr-netflow を使用して、スイッチから出力されるフロー情報を記録します。フロー レコード、フロー エクスポータ、フロー モニターなど、通常の NetFlow（入力 NetFlow）のすべてのパラメータは、出力 NetFlow に適用できます。同様に、入力 NetFlow の制限のほとんどは、出力 NetFlow にも適用されます。詳細については、入力 NetFlow について （503 ページ）を参照してください。

出力 NetFlow の前提条件

出力 NetFlow の前提条件は、次のとおりです。

1. 出力 NetFlow TCAM リージョンで Nexus 9000 スイッチを構成します。
2. 出力 NetFlow TCAM カービング後にスイッチをリロードします。
3. スイッチがオンラインになったら、**ip flow monitor <> output or layer2-switched flow monitor <> output** コマンドを使用して、インターフェイスの任意の L2 または L3 インターフェイスで出力 NetFlow を有効にします。

出力 NetFlow に関する注意事項および制約事項

（注） 確認済みの NetFlow のスケール数については、Cisco.com に掲載されている、各リリースの『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

出力 NetFlow に関する設定時の注意事項および制約事項は、次のとおりです。

- インターフェイスで入力 NetFlow と出力 NetFlow の両方が有効になっている場合、入力 NetFlow は出力 NetFlow よりも優先され、フロー情報は入力方向に生成されます。
- 入力 NetFlow の場合、フローは CFLOW パケットの Ingress(0) フィールドで識別されます。
- 出力 NetFlow の場合、フローは CFLOW パケットの Egress(1) フィールドで識別されます。
- 出力 NetFlow と機能分析は共存できますが、分析フィルタにヒットしたトラフィックフローは、分析で設定されたコレクタにのみエクスポートされます。

- モジュラ型 Cisco Nexus 9500 シャーシでは、トラフィックは異なるライン カードで送受信されます。出力 NetFlow がトラフィックの発信インターフェイスで有効で、トラフィックの着信インターフェイスで有効になっていない場合、出力フロー情報は入力インターフェイスのラインカードでキャプチャされ、出力インターフェイスではキャプチャされません。
- L2 インターフェイスに複数の SVI および sub_interface がある場合、出力 NetFlow に対する入力インターフェイス（if_id）の導出が正しくなくてもかまいません。
- 出力 NetFlow は、Cisco Nexus 9300-EX、EX ライン カードを搭載した Nexus 9500 モジュラ スイッチ、N9K-C9364C、および N9K-C9332C スイッチではサポートされません。
- 出力 NetFlow は、IPv4 およびレイヤ 2 トラフィックでのみサポートされます。出力 NetFlow は、IPv6 およびマルチキャスト トラフィックをサポートしません。
- 出力 NetFlow は、次をサポートしていません。
  - vrf-id
  - 発信制御プレーントラフィックの記録
  - 出力インターフェイス（NetFlow OIF）

出力 NetFlow の構成

出力 NetFlow を構成する手順は、次のとおりです。

手順

ステップ 1 NetFlow 機能を有効にします。

ステップ 2 egr-netflow TCAM カービングを実行し、スイッチをリロードします。

ステップ 3 フローにキーおよびフィールドを指定することによって、フロー レコードを定義します。

ステップ 4 エクスポート フォーマット、プロトコル、宛先、およびその他のパラメータを指定することによって、フロー エクスポータを定義します。

ステップ 5 フロー レコードおよびフロー エクスポータに基づいて、フロー モニタを定義します。

ステップ 6 送信元インターフェイス、サブインターフェイス、または VLAN インターフェイスにフロー モニタを適用します。

出力 NetFlow 機能の有効化

フローを構成するには、先に出力 NetFlow をグローバルで有効しておく必要があります。

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **feature netflow** 例： `switch(config)# feature netflow`	NetFlow 機能を有効にします。デフォルトではディセーブルになっています。 （注） N9K-T2 EoR を搭載した Cisco Nexus 9500 プラットフォーム スイッチは、NetFlow をサポートしていません。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

フロー レコードの作成

フロー レコードを作成し、照合するためのキー、および収集するための非キー フィールドをフロー内に追加します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow record** *name*
	例：
	`switch(config)# flow record Test`
	`switch(config-flow-record)#`
		フロー レコードを作成し、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。フロー レコード名には最大 63 文字の英数字を入力できます。

ステップ 3	（任意） **description** *string* 例： `switch(config-flow-record)# description IPv4Flow`	最大 63 文字で、フロー レコードの説明を示します。

	コマンドまたはアクション	目的
ステップ 4	（任意） **match** *type* 例： `switch(config-flow-record)# match transport destination-port`	一致キーを指定します。詳細については、match パラメータの指定 （530ページ）を参照してください。 （注） レイヤ4ポートデータをエクスポートするには、**match transport destination-port** および **match ip protocol** コマンドが必要です。
ステップ 5	（任意） **collect** *type* 例： `switch(config-flow-record)# collect counter packets`	コレクションフィールドを指定します。詳細については、collect パラメータの指定 （531 ページ）を参照してください。
ステップ 6	（任意） **show flow record** [*name*] [*record-name*] {**netflow-original** | **netflow protocol-port** | **netflow** {**ipv4** | **ipv6**} {**original-input** | **original-output**}} 例： `switch(config-flow-record)# show flow record netflow protocol-port`	NetFlow のフロー レコード情報を表示します。フロー レコード名には最大 63 文字の英数字を入力できます。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config-flow-record)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

出力 NetFlow TCAM カービングを実行する

出力NetFlowTCAMカービングの実行後には、構成を保存してスイッチをリロード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	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	[**no**] **hardware access-list tcam region egr-netflow** {**0** | **512**} 例： `switch(config)# hardware access-list tcam region egr-netflow 512`	出力 NetFlow TCAM カービングを有効にします。デフォルトは無効ですegr-netflow TCAM カービングの場合、サポートされている値は 0 と 512 です。

	コマンドまたはアクション	目的
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ステップ 4	**reload**
	例：
	`switch(config)# reload`
	`This command will reboot the system. (y/n)? [n]`
	y
		スイッチをリロードします。

フロー レコードの作成

フロー レコードを作成し、照合するためのキー、および収集するための非キー フィールドをフロー内に追加します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow record** *name*
	例：
	`switch(config)# flow record Test`
	`switch(config-flow-record)#`
		フロー レコードを作成し、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。フロー レコード名には最大 63 文字の英数字を入力できます。

ステップ 3	（任意） **description** *string* 例： `switch(config-flow-record)# description IPv4Flow`	最大63文字で、フロー レコードの説明を示します。
ステップ 4	（任意） **match** *type* 例： `switch(config-flow-record)# match transport destination-port`	一致キーを指定します。詳細については、match パラメータの指定 （530ページ）を参照してください。 （注） レイヤ4ポートデータをエクスポートするには、**match transport destination-port** および **match ip protocol** コマンドが必要です。
ステップ 5	（任意） **collect** *type* 例： `switch(config-flow-record)# collect counter packets`	コレクション フィールドを指定します。詳細については、collect パラメータの指定 （531 ページ）を参照してください。

	コマンドまたはアクション	目的
ステップ 6	（任意） **show flow record** [*name*] [*record-name*] {**netflow-original** | **netflow protocol-port** | **netflow** {**ipv4** | **ipv6**} {**original-input** | **original-output**}} 例： `switch(config-flow-record)# show flow record netflow protocol-port`	NetFlow のフロー レコード情報を表示します。フロー レコード名には最大 63 文字の英数字を入力できます。
ステップ 7	（任意） **copy running-config startup-config** 例： `switch(config-flow-record)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

match パラメータの指定

フロー レコードごとに、次の match パラメータを 1 つ以上設定する必要があります。

コマンド	目的
match datalink {**mac source-address** | **mac destination-address** | **ethertype** | **vlan**} **例:** `switch(config-flow-record)# match datalink ethertype`	レイヤ 2 属性をキーとして指定します。
match ip {**protocol** | **tos**} **例:** `switch(config-flow-record)# match ip protocol`	IP プロトコルまたは ToS フィールドをキーとして指定します。 （注） レイヤ 4 ポートデータをエクスポートするには、**match transport destination-port** および **match ip protocol** コマンドが必要です。 データは **show hardware flow ip** コマンドの出力に収集されて表示されますが、両方のコマンドを設定するまで収集とエクスポートは行われません。
match ipv4 {**destination address** | **source address**} **例:** `switch(config-flow-record)# match ipv4 destination address`	IPv4 送信元または宛先アドレスをキーとして指定します。

コマンド	目的
match ipv6 {**destination address** | **source address** | **flow-label** | **options**} **例:** `switch(config-flow-record)# match ipv6 flow-label`	IPv6 キーを指定します。
match transport {**destination-port** | **source-port**} **例:** `switch(config-flow-record)# match transport destination-port`	トランスポート送信元または宛先ポートをキーとして指定します。 （注） レイヤ 4 ポートデータをエクスポートするには、**match transport destination-port** および **match ip protocol** コマンドが必要です。 データは **show hardware flow ip** コマンドの出力に収集されて表示されますが、両方のコマンドを設定するまで収集とエクスポートは行われません。

collect パラメータの指定

フロー レコードごとに、次の collect パラメータを 1 つ以上設定する必要があります。

コマンド	目的
collect counter {**bytes** | **packets**} [**long**] **例:** `switch(config-flow-record)# collect counter packets`	フローからパケットベースまたはバイト カウンタを収集します。任意で、64 ビット カウンタを使用することを指定できます。
collect ip version **例:** `switch(config-flow-record)# collect ip version`	フローの IP バージョンを収集します。
collect timestamp sys-uptime {**first** | **last**} **例:** `switch(config-flow-record)# collect timestamp sys-uptime last`	フローの先頭または最終パケットに関するシステム稼働時間を収集します。
collect transport tcp flags **例:** `switch(config-flow-record)# collect transport tcp flags`	フローのパケットに対応する TCP トランスポート層フラグを収集します。

フロー エクスポータの作成

フロー エクスポータの構成では、フローに対するエクスポート パラメータを定義し、リモートの出力 NetFlow Collector への到達可能性情報を指定します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow exporter** *name*
	例：
	`switch(config)# flow exporter flow-exporter-one`
	`switch(config-flow-exporter)#`
		フロー エクスポータを作成し、フロー エクスポータ コンフィギュレーション モードを開始します。フローエクスポータ名を最大 63 文字の英数字で入力できます。

ステップ 3	**destination** {*ipv4-address* | *ipv6-address*} [**use-vrf** *name*] 例： `switch(config-flow-exporter)# destination 192.0.2.1`	このフロー エクスポータの宛先 IPv4 または IPv6 アドレスを設定します。任意で、出力 NetFlow Collector に到達するために使用する VRF を構成できます。VRF 名には最大 32 文字の英数字を入力できます。
ステップ 4	**source** *interface-type name/port* 例： `switch(config-flow-exporter)# source ethernet 2/1`	構成された宛先で出力 NetFlow Collector に到達するために使用するインターフェイスを指定します。
ステップ 5	（任意） **description** *string* 例： `switch(config-flow-exporter)# description exportversion9`	このフロー エクスポータについて説明します。説明には最大 63 文字の英数字を入力できます。
ステップ 6	（任意） **dscp** *value* 例： `switch(config-flow-exporter)# dscp 0`	DSCP（DiffServ コードポイント）値を指定します。範囲は 0 ～ 63 です。
ステップ 7	（任意） **transport udp** *port* 例： `switch(config-flow-exporter)# transport udp 200`	NetFlow Collector に到達するために使用する UDP ポートを指定します。範囲は 0 ～ 65535 です。 （注） UDP ポートを指定しない場合は、9995 がデフォルトとして選択されます。

	コマンドまたはアクション	目的

ステップ 8	**version 9**
	例：
	`switch(config-flow-exporter)# version 9`
	`switch(config-flow-exporter-version-9)#`
		NetFlowエクスポートバージョンを指定します。フローエクスポータのバージョン9コンフィギュレーション サブモードを開始するには、バージョン 9 を選択します。

ステップ 9	（任意） **option** {**exporter-stats** | **interface-table**} **timeout** *seconds* 例： `switch(config-flow-exporter-version-9)# option exporter-stats timeout 1200`	フロー エクスポータの統計情報再送信タイマーを設定します。値の範囲は 1 ～ 86400 秒です。
ステップ 10	（任意） **template data timeout** *seconds* 例： `switch(config-flow-exporter-version-9)# template data timeout 1200`	テンプレートデータ再送信タイマーを設定します。値の範囲は 1 ～ 86400 秒です。
ステップ 11	（任意） **copy running-config startup-config** 例： `switch(config-flow-exporter-version-9)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

フロー モニタの作成

フロー モニタを作成して、フロー レコードおよびフロー エクスポータと関連付けることができます。1 つのモニタに属しているすべてのフローは、様々なフィールド上で照合するために関連するフロー レコードを使用します。データは指定されたフローエクスポータにエクスポートされます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow monitor** *name*
	例：
	`switch(config)# flow monitor flow-monitor-one`
	`switch(config-flow-monitor)#`
		フローモニタを作成し、フローモニタ コンフィギュレーション モードを開始します。フロー モニタ名を最大 63 文字の英数字で入力できます。

ステップ 3	（任意） **description** *string* 例：	このフローモニタについて説明します。説明には最大 63 文字の英数字を入力できます。

	コマンドまたはアクション	目的
	`switch(config-flow-monitor)# description IPv4Monitor`	
ステップ 4	（任意） **exporter** *name* 例： `switch(config-flow-monitor)# export v9`	フロー エクスポータとこのフロー モニタを関連付けます。エクスポータ名には最大 63 文字の英数字を入力できます。
ステップ 5	**record** *name* [**netflow-original** | **netflow protocol-port** | **netflow** {**ipv4** | **ipv6**} {**original-input** | **original-output**}] 例： `switch(config-flow-monitor)# record IPv4Flow`	フロー レコードを指定したフロー モニタと関連付けます。レコード名には最大 63 文字の英数字を入力できます。
ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config-flow-monitor)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

インターフェイスへのフロー モニタの適用

output キーワードを使用して、出力方向にフロー モニタを適用できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface vlan** *vlan-id*
	例：
	`switch(config)# interface vlan 10`
	`switch(config-if)#`
		VLAN 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 3	**ip flow monitor** {**ipv4** | **ipv6** | **layer-2-switched**} **input** 例： `switch(config-if)# ip flow monitor ipv4 input`	入力パケットのインターフェイスに、IPv4、IPv&、またはレイヤ 2 スイッチ フロー モニタを関連付けます。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

レイヤ 2 インターフェイスでのレイヤ 3 出力 NetFlow の構成

レイヤ 2 インターフェイスでレイヤ 3 フロー情報をキャプチャするために、レイヤ 2 インターフェイスでレイヤ 3 フロー モニタを定義できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**flow record** *name*
	例：
	`switch(config)# flow record L3_record`
	`switch(config-flow-record)#`
		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。フロー レコードの設定の詳細については、フロー レコードの作成（527ページ） を参照してください。

ステップ 3	**interface** {**ethernet** *slot/port* | **port-channel** *number*}
	例：
	`switch(config)# interface Ethernet 6/3`
	`switch(config-if#)`
		インターフェイス設定モードを開始します。インターフェイス タイプは、物理的なイーサネット ポートまたはポート チャネルを指定できます。

ステップ 4	**switchport** 例： `switch(config-if)# switchport`	インターフェイスをレイヤ 2 モードに変更します。スイッチ ポートの構成に関する詳細については、『Cisco Nexus 9000 シリーズ NX-OS レイヤ 2 スイッチング構成ガイド』を参照してください。

ステップ 5	**ip flow monitor** *flow-name* **input** | **output**
	例：
	`switch(config-if)# ip flow monitor v41 input`
	`switch(config-if)# layer2-switched flow monitor v41 output`
		または
		layer2-switched flow monitor *flow-name* **input** | **output**
		IPv4 フロー モニタをスイッチ ポートの入力または出力パケットに関連付けます。フローモニタ名を最大 63 文字の英数字で入力できます。

ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

出力 NetFlow 構成の確認

出力 NetFlow 構成を表示するには、次のタスクのうちのいずれかを実行します。

コマンド	目的
show flow cache [**ipv4** | **ce**] [**output**]	出力 NetFlow IP フローに関する情報を表示します。
show running-config netflow	現在デバイスにある出力 NetFlow 構成を表示します。

出力 NetFlow の表示例

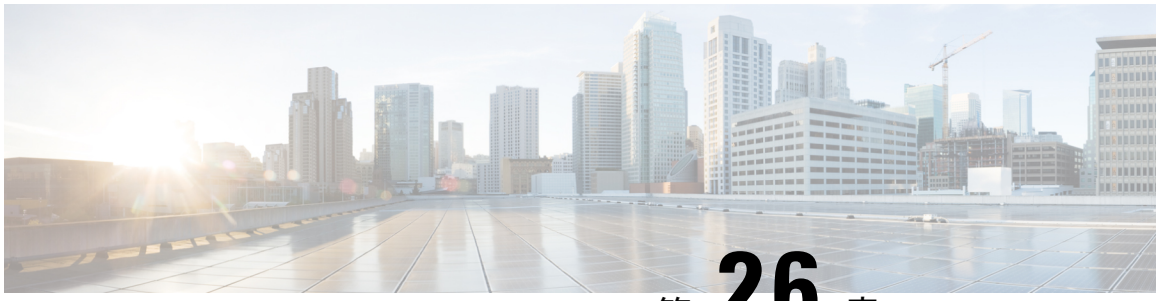
IPv4 の **show flow cache** コマンドの出力には、次のように表示されます。

```
switch(config-if-range)# show flow cache
NOTE: Only 10k flows are displayed in XML output
Egress IPV4 Entries
SIP              DIP              BD ID    S-Port   D-Port   Protocol  Byte Count
  Packet Count       TCP FLAGS    TOS     if_id        output_if_id       flowStart
flowEnd     Profile   Ing-VRF
130.1.1.2        162.1.1.1        4119     60       893      6         161082
  171              0x0          0x0     0x1a002600   0x1a002800         716994
732532        4  : NF   21
130.1.1.2        162.1.1.1        4151     60       11013    6         61230
  65               0x0          0x0     0x1a006600   0x1a006800         715951
728074        4  : NF   38
130.1.1.2        162.1.1.1        4145     60       3441     6         162966
  173              0x0          0x0     0x1a005a00   0x1a005c00         713085
727941        4  : NF   35
```

NetFlow のコンフィギュレーション例

この例では、IPv4 に対して出力 NetFlow エクスポータを構成する方法を示します。

```
feature netflow
flow exporter ee
 destination 171.70.242.48 use-vrf management
 source mgmt0
 version 9
  template data timeout 20
flow record rr
 match ipv4 source address
 match ipv4 destination address
 collect counter bytes
 collect counter packets
flow monitor foo
 record rr
 exporter ee
interface Ethernet2/45
 ip flow monitor foo output
 ip address 10.20.1.1/24
 no shutdown
```



第 26 章

混合モードで構成する

この章では、Cisco NX-OS デバイス上で混合モード（分析とNetFlow）機能を構成する方法について説明します。

この章は、次の項で構成されています。

- 混合モードについて （537 ページ）
- 混合モードに関する注意事項と制限事項 （537 ページ）
- 混合モード：ユース ケース （539 ページ）
- 混合モード構成の検証 （542 ページ）
- 混合モードの表示例 （543 ページ）

混合モードについて

スイッチで NetFlow 機能と分析機能を構成して、両方の機能を共存させ、CPU からの標準の V9 エクスポートを利用することができます。両方の機能が共存するこのモードは、混合モードと呼ばれます。

（注） Cisco NX-OS リリース 10.2(3)F までは、標準 V9 エクスポートは NetFlow フロー レコードに対してのみサポートされていました。Cisco NX-OS リリース 10.2(3)F 以降、標準の V9 エクスポートは分析でもサポートされています。ただし、NetFlow機能と分析機能は相互に排他的でした。

混合モードに関する注意事項と制限事項

次の注意事項と制限事項は、混合モードに適用されます。

- Cisco NX-OS リリース 10.3(1)F 以降、NetFlow と分析の両方は共存でき、CPU からの標準の V9 エクスポートを使用できるため、コレクタの処理負荷が減少します。ただし、この混合モードは9300-EX モジュールではサポートされていません。また、分析モードと混合モードの間で、相互に移行することはできません。適用される注意事項と制限事項は次のとおりです。

- L2 フロー モニタはサポートされていません。
- VRF フィルタはサポートされていません。
- ND ISSU はサポートされていません。
- IPv4 および IPv6 プロファイルは次のとおりです。
  - IP フロー モニタ：28
  - IPv6 フロー モニタ：26
- 分析レコード構成は、すべてのレコードパラメータのスーパーセットである必要があります。
- システム フィルタ/インターフェイス フィルタ構成を構成する場合には、まずシステム モニタを構成します。
- システムモニタを構成解除する場合には、まずシステムフィルタ/インターフェースフィルタ構成を構成解除します。
- 混合モードでは、EOR の AN フローに対して 2 つの NetFlow レコードがエクスポートされます。

- インターフェイス ベースの FT は、MPLS、VXLAN、GRE などのトンネル トラフィック フローではサポートされません。
- Cisco NX-OS リリース 10.3(3)F 以降では、すべての Cisco Nexus 9000 スイッチの NetFlow および分析機能で Ingress_VRF_ID がサポートされています。

  入力 vrf-id がキャプチャされ、**show flow cache** に表示され、NetFlow コレクタに送信されます。

  レイヤ 3 NetFlow がレイヤ 2 インターフェイスで設定され、トラフィックが送信されると、**show flow cache** コマンド出力に Ingress_VRF_ID の値がゼロとして表示されます。
- Cisco NX-OS リリース 10.3(3)F 以降、NetFlow 混合モードはデフォルトで有効になっています。これにより、分析機能に割り当てられる TCAM スペースが最大 512 エントリから最大 256 エントリに削減されます。
- Cisco NX-OS リリース 10.3(3)F 以降では、以前のリリースとは異なり、システム フィルタで定義されているが、インターフェイス フィルタで定義されていないフロー レコードが表示されます。以前のリリースでは、インターフェイスフィルタが構成されている場合、フロー レコードはインターフェイス フィルタで定義されている場合にのみ表示されました。

混合モード：ユース ケース

混合モードは、NetFlowモードからのみ設定できます。スイッチですでに分析機能が有効になっているシナリオでは、最初に分析を構成解除し、NetFlow 機能を構成してから、混合モードに移行します。

混合モードで考えられるユース ケースは次のとおりです。

- 機能分析がすでに展開されているスイッチ
- 機能 NetFlow がすでに展開されているスイッチ
- どちらの機能も構成されていないスイッチ

混合モードを構成したら、標準の V9 フォーマットを使用して、CPU からそれぞれのコレクタに NetFlow と分析の両方のフロー レコードをエクスポートします。

（注） 分析データは、NetFlow データのスーパーセットです。フロー遅延、トラフィック バースト データ、ペイロード長、TCP フラグ、IP フラグ、パケット処理フラグなどの追加の分析フロー データは、ベンダー固有フィールド（VSF）を介して通信されます。

ユースケース：機能分析がすでに展開されたスイッチ

機能分析構成を構成解除または保存し、「ユースケース：どちらの機能も構成されていないスイッチ」に示されている手順を実行します。分析モードと混合モードの間では移行できないことに注意してください。

ユースケース：すでに機能 NetFlow が展開されたスイッチ

機能 netflow がすでに展開されているスイッチに対して、次の手順を実行します。

1. 次のコマンドを使用して、混合モードの tcam カービングを実行します：

   hardware flow-table analytics-netflow

（注） このコマンドは、フロー モニタリングを中断し、短い期間、エクスポートを記録します。

2. 次のように機能分析を構成します。

```
feature analytics
analytics
  flow filter telemetryFP
    ipv4 telemetryIpv4Acl
    ipv6 telemetryIpv6Acl
  flow exporter e11
```

```
  destination 10.10.20.21 v9
  transport udp 1100
  events transport udp 55
  source Ethernet1/42
flow exporter e12
  destination 10.10.20.21 v9
  transport udp 9200
  events transport udp 555
  source Ethernet1/42
flow record fte-record
  match ip source address
  match ip destination address
  match ip protocol
  match transport source-port
  match transport destination-port
  collect counter packets
  collect timestamp sys-uptime first
  collect timestamp sys-uptime last
flow monitor m1
  record fte-record
  exporter-bucket-id 1 0 4095
    exporter e11
flow monitor m2
  record fte-record
  exporter-bucket-id 1 0 2000
    exporter e11
  exporter-bucket-id 2 2001 4095
    exporter e12
flow profile telemetryProf
  collect interval 1000
  source port 1001
flow event fte-event1
  group drop-events
    capture buffer-drops
    capture acl-drops
    capture fwd-drops
  group packet-events
    capture tos 50
    capture ttl 50
flow system config
  exporter-id 4
  monitor m1 input
  profile telemetryProf
  event fte-event1
  filter telemetryFP
```

ユースケース：どちらの機能も構成されていないスイッチ

機能 netflow を構成してから、「ユース ケース：機能 NetFlow ですでに導入されているスイッチ」に記載されている手順、または次の手順を実行します。

```
feature netflow
hardware flow-table analytics-netflow
feature analytics
flow exporter e1
  destination 10.10.20.21
  transport udp 100
  source Ethernet1/42
  version 9
flow record r4
  match ipv4 source address
```

```
  match ipv4 destination address
  match ip protocol
  match transport source-port
  match transport destination-port
  collect counter bytes
  collect counter packets
  collect timestamp sys-uptime first
  collect timestamp sys-uptime last
flow record r6
  match ip protocol
  match transport source-port
  match transport destination-port
  match ipv6 source address
  match ipv6 destination address
  collect counter bytes
  collect counter packets
  collect timestamp sys-uptime first
  collect timestamp sys-uptime last
flow monitor m41
  record r4
  exporter e1
flow monitor m6
  record r6
  exporter e1
analytics
  flow filter telemetryFP
    ipv4 telemetryIpv4Acl
    ipv6 telemetryIpv6Acl
  flow exporter e11
    destination 10.10.20.21 v9
    transport udp 1100
    events transport udp 55
    source Ethernet1/42
  flow exporter e12
    destination 10.10.20.21 v9
    transport udp 9200
    events transport udp 555
    source Ethernet1/42
  flow record fte-record
    match ip source address
    match ip destination address
    match ip protocol
    match transport source-port
    match transport destination-port
    collect counter packets
    collect timestamp sys-uptime first
    collect timestamp sys-uptime last
  flow monitor m1
    record fte-record
    exporter-bucket-id 1 0 4095
      exporter e11
  flow monitor m2
    record fte-record
    exporter-bucket-id 1 0 2000
      exporter e11
    exporter-bucket-id 2 2001 4095
      exporter e12
  flow profile telemetryProf
    collect interval 1000
    source port 1001
  flow event fte-event1
    group drop-events
      capture buffer-drops
      capture acl-drops
```

```
      capture fwd-drops
    group packet-events
      capture tos 50
      capture ttl 50
  flow system config
    exporter-id 4
    monitor m1 input
    profile telemetryProf
    event fte-event1
    filter telemetryFP

interface Ethernet1/42
  ip flow monitor m41 input
  ipv6 flow monitor m6 input
```

混合モード構成の検証

混合モードの構成を表示するには、次のいずれかの作業を行います。

コマンド	目的
show flow cache [**ipv4** | **ipv6**]	NetFlow IP フローに関する情報を表示します。 （注） Cisco NX-OS リリース 10.3(3)F 以降では、このコマンド出力には入力 VRF ID も表示されます。入力 vrf-id がキャプチャされ、show flow cache に表示され、NetFlow コレクタに送信されます。
show flow exporter [*name*]	NetFlow/分析のフロー エクスポータ情報と統計情報を表示します。フロー エクスポータ名を最大 63 文字の英数字で入力できます。
show flow interface [*interface-type slot/port*]	NetFlow/分析インターフェイスに関する情報を表示します。
show flow record [*name*]	NetFlow/分析のフロー レコード情報を表示します。フロー レコード名には最大 63 文字の英数字を入力できます。
show running-config [**netflow** | **analytics**]	現在デバイスにある、共存している NetFlow と分析の構成を表示します。
show flow monitor	NetFlow/分析モニタの構成を表示します。
show flow system	分析システム構成に関する情報を表示します。
show flow filter	分析フィルタに関する情報を表示します。
show flow profile	分析プロファイルに関する情報を表示します。

コマンド	目的
show flow event	分析イベントに関する情報を表示します。

混合モードの表示例

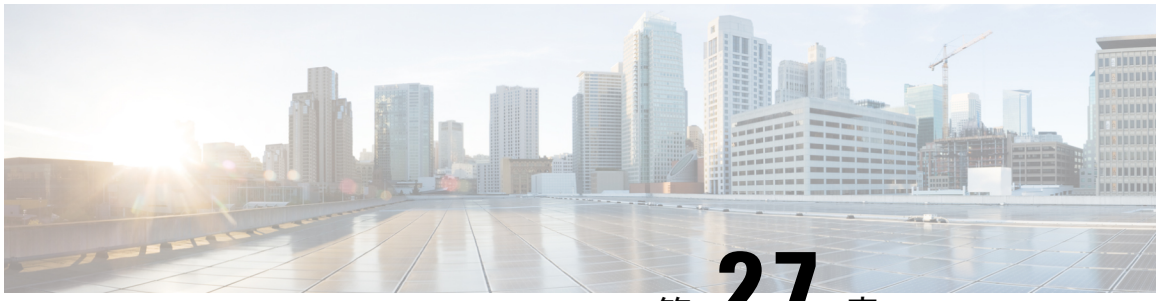
show flow cache コマンドの出力は次のように表示されます。

（注） XML 出力には 10k のフローのみが表示されます。

（注） レイヤ 3 NetFlow がレイヤ 2 インターフェイスで構成され、トラフィックが送信されてから **show flow cache** コマンドが実行されると、出力には Ing-VRF の値がゼロとして表示されます。

show flow cache

```
Ingress IPV4 Entries
SIP              DIP          BD ID    S-Port   D-Port   Protocol  Byte Count   Packet
Count  TCP FLAGS    TOS     if_id       flowStart     flowEnd      Profile    Ing-VRF
17.1.1.2         17.1.1.1     1671     0        0        89        480          8
       0x0          0xc0    0x1a004400  2938966       2976728      5  : NF    0
17.1.1.2         224.0.0.13   1672     0        0        103       144          2
       0x0          0xc0    0x1a004400  2941719       2969951      5  : NF    0
17.1.1.2         224.0.0.13   1675     0        0        103       72           1
       0x0          0xc0    0x1a004400  2961417       2961667      5  : NF    0
17.1.1.2         224.0.0.5    1675     0        0        89        340          5
       0x0          0xc0    0x1a004400  2943341       2979400      5  : NF    0
17.1.1.2         17.1.1.1     1671     2048     0        1         3612         43
       0x0          0x0     0x1a004400  2938188       2980184      5  : NF    0

Ingress IPV6 Entries
SIP                         DIP       BD ID    S-Port  D-Port  Protocol  Byte Count
Packet Count  TCP FLAGS Flow Label  if_id       flowStart  flowEnd    Ing-VRF
fe80::822d:bfff:fe81:e415   ff02::5   4147     0       0       89        490        5
           0x0       0x0         0x1a003400  11217548   11254367   1
```



第 27 章

sFlow の設定

この章では、Cisco NX-OS デバイスで sFlow を設定する方法について説明します。

この章は、次の項で構成されています。

- sFlow （545 ページ）
- sFlow の前提条件 （546 ページ）
- sFlow の注意事項および制約事項 （546 ページ）
- sFlow のデフォルト設定 （549 ページ）
- sFlow の設定 （550 ページ）
- sFlow 設定の確認 （560 ページ）
- sFlow 統計情報のモニタリングとクリア （561 ページ）
- sFlow の設定例 （561 ページ）
- その他の参考資料 （562 ページ）

sFlow

サンプリングされたフロー（sFlow）は、次のサンプリングされたフローモニタリングテクノロジーです。

- スイッチとルータと一緒のデータ ネットワークのリアルタイム トラフィックの監視を有効にし、
- トラフィックを監視するために sFlow エージェント ソフトウェアのサンプリング メカニズムを導入し、
- サンプル データを中央のデータ コレクタに転送します。

sFlow の詳細については、RFC 3176 を参照してください。

sFlow エージェント

Cisco NX-OS ソフトウェアに組み込まれている sFlow エージェントは、サンプリングされるパケットのデータ ソースに関連付けられたインターフェイス カウンタを定期的にサンプリング

またはポーリングします。このデータ ソースは、イーサネットインターフェイス、EtherChannel インターフェイス、ある範囲に属するイーサネット インターフェイスのいずれかです。sFlow エージェントは、イーサネット ポート マネージャにクエリーを送信して対応する EtherChannel メンバーシップ情報を確認するほか、イーサネット ポート マネージャからもメンバーシップの変更の通知を受信します。

sFlow サンプリングをイネーブルにすると、サンプリング レートとハードウェア内部の乱数に基づいて、入力パケットと出力パケットが sFlow でサンプリングされたパケットとして CPU に送信されます。sFlow エージェントはサンプリングされたパケットを処理し、sFlow アナライザに sFlow データグラムを送信します。sFlow データグラムには、元のサンプリングされたパケットに加えて、入力ポート、出力ポート、および元のパケット長に関する情報が含まれます。sFlow データグラムには、複数の sFlow サンプルを含めることができます。

sFlow の前提条件

sFlow には、次の前提条件があります。

- Cisco Nexus 9332PQ、9372PX、9372TX、93120TX スイッチ、および N9K-M6PQ 汎用拡張モジュール（GEM）搭載の Cisco Nexus 9396PX、9396TX、93128TX スイッチについては、sFlow データ ソースとして設定するすべてのアップリンク ポート用の sFlow および SPAN ACL TCAM リージョン サイズを設定する必要があります。これを行うには、**hardware access-list tcam region sflow** および **hardware access-list tcam region span** コマンドを使用します。詳細については、『ACL TCAM リージョン サイズの設定』を参照してください。

（注） デフォルトでは、sflow リージョンサイズはゼロで、span リージョン サイズはゼロ以外です。ポートを sFlow データ ソースとして設定するには、sflow リージョンを 256 に設定し、十分なエントリを span リージョンに割り当てる必要があります。

- マルチキャストトラフィックの出力sFlowには、**ハードウェアマルチキャストグローバル TXスパン設定**が必要です

sFlow の注意事項および制限事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

sFlow には、次の注意事項と制限事項があります。

- 少なくとも 1 つの sFlow データソースが設定されている場合、SPAN セッションは起動できません。
  - 少なくとも 1 つの SPAN セッションが **no shut** として設定されている場合、sFlow データソースは追加できません。
  - sFlow に使用されるサンプリング モードは、LFSR と呼ばれるアルゴリズムに基づいています。LFSR を使用するため、数個のパケットごとに 1 個がサンプリング レート n でサンプリングされることは保証されません。ただし、サンプリングされるパケットの数は、一定期間の合計パケット数と同じです。
- sFlow を使用して FEX HIF ポートで Rx トラフィックをサンプル化すると、サンプル化されたトラフィックに追加の VNTAG および 802.1Q タグが存在します。
- Cisco Nexus 9300-EX および 9300-FX プラットフォーム スイッチでは、FEX、HIF、および NIF ポートを sFlow データ ソース インターフェイスとして設定できません。
- sFlow と SPAN が同じインターフェイスに設定されており、ハードウェア レート リミッタが sFlow 用に設定されている場合、**show hardware rate-limiter** コマンドの出力の Rate-Limiter Drops カウンタは予想よりも多くのドロップを表示します。
- sFlow はソフトウェア駆動型の機能で、ハードウェアは sFlow 送信元インターフェイスから CPU にトラフィックのコピーを送信するだけです。高い CPU 使用率が予想されます。ハードウェアによって CPU に送信される sFlow トラフィックは、CPU を保護するためにレート制限されます。
- インターフェイスの sFlow をイネーブルにすると、入力と出力の両方に対してイネーブルになります。入力だけまたは出力だけの sFlow をイネーブルにできません。

  Cisco Nexus 9636C-R および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 スイッチの場合、sFlow は入力方向のインターフェイスに対してのみ有効にできます。
- sFlow も有効になっているインターフェイスでストーム制御を有効にした場合、ストーム制御機能は動作しません。
- sFlow は SVI ではサポートされません。
- サブインターフェイスは sFlow ではサポートされていません。
- システムの sFlow の設定およびトラフィックに基づいてサンプリング レートを設定することをお勧めします。
- スイッチは 1 つのみの sFlow コレクタをサポートします。
- sFlow とネットワーク アドレス変換（NAT）は、同じポートではサポートされません。
- sFlow は、IPv6 トラフィックのサンプリングをサポートしていますが、に限られます。
- sFlow は、マルチキャスト、ブロードキャスト、または未知のユニキャスト パケットの出力のサンプリングはサポートしません。

- sFlow カウンタは、sFlow データ送信元インターフェイスに入力される制御パケットに対しても増加します。これらのパケットはサンプリングされ、sFlow データグラムとして送信されます（データ プレーン トラフィックと同様）。
- 次の Cisco Nexus スイッチは、sFlow と SPAN を同時にサポートします。
  - N9336C-FX2
  - N93240YC-FX2
  - N93360YC-FX2
- Cisco NX-OS リリース 9.3(3) 以降、Cisco Nexus 9300-GX プラットフォーム スイッチは、sFlow と SPAN の両方をサポートしています。
- Nexus 9000-EX、FX、および GX ファミリのスイッチは、4096、8192、16384、32768、65536 の値でのみサンプリングをサポートします。これら以外の値を構成すると、値はサポートされている次の順番の値に丸められます。
- sFlow が N9K-C9508-FM-G で N9K-X9716D-GX ライン カードを搭載した状態で設定されている場合、SPAN セッションを設定する前に sFlow を無効にします。
- Cisco NX-OS リリース 10.1(1) 以降、sFlow および SPAN は Cisco N9K-C93180YC-FX3 プラットフォーム スイッチでサポートされています。
- Cisco NX-OS リリース 10.1(2) 以降、sFlow は Cisco Nexus N9K-X9624D-R2 ライン カードでサポートされます。
- Cisco NX-OS リリース 10.1(2) 以降、sFlow は N9K-X9716D-GX ライン カードを搭載した Cisco Nexus N9K-C9508-FM-G クラウドスケール ファブリック モジュールで VXLAN トラフィックをサポートします。
- Cisco NX-OS リリース 10.2（1）以降、sFlow 拡張 BGP（ゲートウェイ）は Cisco Nexus N9K-C93600CD-GX、N9K-C93240YC-FX2、N9K-C93180YC-EX、N9K-C93180YC-FX、N9K-C93180YC-FX3S、N9K-93600CD-GX プラットフォーム スイッチ、および Nexus 9500 と一緒の N9K-X9716D-GX と N9K-X9736C-FX ライン カードでサポートされます。
- NX-OS は、顧客のニーズに応じてハードウェア リソースを利用するための柔軟な転送テンプレートを提供します。sFlow 入力 IPv6 サンプリングで sFlow レコードに BGP 情報を正しく入力するには、ライン カード上のすべての IPv6 ルートを持つテンプレートを選択する必要があります。たとえば、顧客は **system routing template-mpls-heavy** を設定できます。詳細については、『Cisco Nexus 9000 シリーズ NX-OS コマンド参照（設定コマンド）、リリース 9.3(x)』を参照してください。コマンドを有効にするには、システムを再起動する必要があります。これは、GX モジュラ シャーシに適用されます。
- ECMP が BGP で設定され、ECMP 宛先ルートの場合、エクスポートされた sFlow レコードの拡張ゲートウェイ レコードのネクストホップ情報は 0 になります。自律システムなどの他の BGP 情報は、最初のパスから取得されます。sFlow レコードの出力インターフェイスは 0（不明）に設定され、フローがいずれかのパスを通過する可能性があることを示します。

- Cisco NX-OS リリース 10.2(1q)F 以降、sFlow は Cisco N9K-C9332D-GX2B プラットフォーム スイッチでサポートされます
- Cisco NX-OS リリース 10.2(1) 以降、拡張 BGP データを収集できるようになりました。sFlow がこのデータを収集するには、物理インターフェイスやポート チャネルなどの非 SVI レイヤ 3 インターフェイスを sFlow ソースとして構成する必要があります。
- Cisco NX-OS リリース 10.2(3)F 以降、sFlow フロー キャッシュ サイズは、以前のリリースの 3k ルート エントリから 30k v4 および 30k v6 ルート エントリに増加します。Cisco Nexus C93600CD-GX、C93240YC-FX2、C93180YC-EX、C93180YC-FX、C93180YC-FX3S、93600CD-GX と X9716D-GX プラットフォーム スイッチでこの機能はサポートされています。
- Cisco NX-OS リリース 10.3(1)F 以降、sFlow は Cisco Nexus 9808 プラットフォーム スイッチでサポートされます。
  - 出力サンプルパケットの場合、書き換えられた情報はsFlow レコードで利用できません。
  - 出力 sFlow は、直接接続されたホストではサポートされていません。
  - sFlow は、サブインターフェイス トラフィックではサポートされていません。
- Cisco NX-OS リリース 10.3(1)F 以降、sFlow は IPv6 コレクタをサポートします。ただし、一度に設定できるコレクタは、IPv4 または IPv6 のいずれか 1 つだけです。また、送信元 IP アドレスとコレクタ IP アドレスは、同じアドレス ファミリ、つまり IPv4 または IPv6 アドレス ファミリに属している必要があります。
- Cisco NX-OS リリース 10.4(1)F 以降、sFlow は次のライン カードおよびスイッチでサポートされます。
  - Cisco Nexus 9804 スイッチおよび次の制限が適用されます。
    - 出力サンプルパケットの場合、書き換えられた情報はsFlow レコードで利用できません。
    - 出力 sFlow は、直接接続されたホストではサポートされません。
    - sFlow は、サブインターフェイス トラフィックではサポートされません。
  - Cisco Nexus 9332D-H2R スイッチ
  - Cisco Nexus X98900CD-A および X9836DM-A ラインカードと Cisco Nexus 9808 および 9804 スイッチ

sFlow のデフォルト設定

次の表に、sFlow パラメータのデフォルト設定を示します。

sFlow の設定

表 *21 :* デフォルトの *sFlow* パラメータ

パラメータ	デフォルト
sFlow のサンプリング レート	4096
sFlow のサンプリング サイズ	128
sFlow カウンタのポーリング間隔	20
sFlow の最大データグラム サイズ	1400
sFlow コレクタの IP アドレス	0.0.0.0
sFlow のコレクタ ポート	6343
sFlow エージェントの IP アドレス	0.0.0.0

sFlow の設定

sFlow の有効化

スイッチの sFlow を設定する前に sFlow 機能を有効にする必要があります。

手順の概要

1. **configure terminal**
2. [**no**] **feature sflow**
3. （任意） **show feature**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **feature sflow** 例： `switch(config)# feature sflow`	sFlow を有効または無効にします。

	コマンドまたはアクション	目的
ステップ 3	（任意） **show feature** 例： `switch(config)# show feature`	有効および無効にされた機能を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

サンプリング レートの設定

sFlow のサンプリング レートを設定できます。

始める前に

sFlow が有効になっていることを確認します。

Nexus 9000-EX、FX、および GX ファミリのスイッチは、4096、8192、16384、32768、65536 の値でのみサンプリングをサポートします。これら以外の値を構成すると、値はサポートされている次の順番の値に丸められます。

手順の概要

1. **configure terminal**
2. [**no**] **sflow sampling-rate** *sampling-rate*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **sflow sampling-rate** *sampling-rate* 例： `switch(config)# sflow sampling-rate 50000`	パケットの sFlow のサンプリング レートを設定します。 *sampling-rate* には 4096 ～ 1000000000 の整数を指定できます。

	コマンドまたはアクション	目的
ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

最大サンプリング サイズの設定

サンプリングされたパケットからコピーする最大バイト数を設定できます。

始める前に

sFlow が有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow max-sampled-size** *sampling-size*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **sflow max-sampled-size** *sampling-size* 例： `switch(config)# sflow max-sampled-size 200`	sFlow の最大サンプリング サイズを設定します。 *sampling-size* の範囲は 64～256 バイトです。
ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。

	コマンドまたはアクション	目的
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

カウンタのポーリング間隔の設定

データ ソースに関連するカウンタの継続的なサンプル間の最大秒数を設定できます。サンプリング間隔 0 は、カウンタのサンプリングをディセーブルにします。

始める前に

sFlow が有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow counter-poll-interval** *poll-interval*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **sflow counter-poll-interval** *poll-interval* 例： `switch(config)# sflow counter-poll-interval 100`	インターフェイスの sFlow のポーリング間隔を設定します。 *poll-interval* の範囲は 0～2147483647 秒です。
ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例：	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

	コマンドまたはアクション	目的
	`switch(config)# copy running-config startup-config`	

最大データグラム サイズの設定

1 つのサンプル データグラムで送信できるデータの最大バイト数を設定できます。

始める前に

sFlow が有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow max-datagram-size** *datagram-size*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **sflow max-datagram-size** *datagram-size* 例： `switch(config)# sflow max-datagram-size 2000`	sFlow の最大データグラム サイズを設定します。 *datagram-size* の範囲は 200～9000 バイトです。
ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

sFlow コレクタ アドレスの設定

管理ポートに接続されている sFlow データ コレクタの IPv4 [または、 IPv6（ or IPv6）] アドレスを構成できます。

始める前に

sFlow が有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow collector-ip** *ip-address* **vrf** *vrf* [**source** *ip-address*]
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **sflow collector-ip** *ip-address* **vrf** *vrf* [**source** *ip-address*]
	例：
	`switch(config)# sflow collector-ip 192.0.2.5 vrf management`
	`switch(config)# sflow collector-ip 2001::1 vrf management`
		sFlow コレクタの IPv4 または IPv6 アドレスを構成します。IP アドレスを 0.0.0.0 に設定すると、すべてのサンプリングがドロップされます。
		vrf は次のいずれかになります。
		• ユーザ定義の VRF 名：最大 32 文字の英数字を指定できます。
		• **vrf 管理**：sFlow データ コレクタが管理ポートに接続されたネットワークに存在する場合は、このオプションを使用する必要があります。
		• **vrf デフォルト**：sFlow データ コレクタが前面パネルのポートに接続されたネットワークに存在する場合は、このオプションを使用する必要があります。
		source *ip-address* オプションを指定すると、送信される sFlow データグラムで送信元 IP アドレスが IP パケットの送信元アドレスとして使用されるように

	コマンドまたはアクション	目的
		なります。送信元IPアドレスは、スイッチのローカルインターフェイスの1つですでに設定されている必要があります。それ以外の場合は、エラー メッセージが表示されます。このオプションの設定後に送信元IPアドレスを持つインターフェイスが変更または削除されると、sFlow データグラムは送信されなくなり、イベント履歴エラーと syslog エラーがログに記録されます。**source** *ip-address* オプションが未設定の場合、Cisco NX-OS は送信される sFlow データグラムに対して、IP パケットの送信元アドレスを自動的に選択します。
ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

sFlow コレクタ ポートの設定

sFlow データグラムの宛先ポートを設定できます。

始める前に

sFlow が有効になっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow collector-port** *collector-port*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **sflow collector-port** *collector-port* 例： `switch(config)# sflow collector-port 7000`	sFlow コレクタの UDP ポートを設定します。 *collector-port* の範囲は 1～65535 です。
ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

sFlow エージェント アドレスの設定

sFlow エージェントの IPv4 または IPv6 アドレスを構成します。

始める前に

sFlow を有効にしていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow agent-ip** *ip-address*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **sflow agent-ip** *ip-address*
	例：
	`switch(config)# sflow agent-ip 192.0.2.3`
	`switch(config)# sflow agent-ip 2001::10`
		sFlow エージェントの IPv4 または IPv6 アドレスを構成します。
		デフォルトの IP アドレスは 0.0.0.0 です。つまり、すべてのサンプルはドロップされます。sFlow 機能をイネーブルにするには、有効な IP アドレスを指定する必要があります。
		（注）
		この IP アドレスは、コレクタに sFlow データグラムを送信するための送信元 IP アドレスとは限りません。
		エージェントの IP アドレスとコレクタの IP アドレスは、同じアドレス ファミリ、つまり IPv4 または IPv6 アドレス ファミリに属している必要があります。

ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

sFlow サンプリング データ ソースの設定

sFlow のサンプリング データ ソースには、イーサネット ポート、イーサネット ポートの範囲、またはポート チャネルとして設定できます。

始める前に

sFlow を有効にしていることを確認します。

データ ソースとしてポート チャネルを使用する場合は、すでにポート チャネルを設定して、ポート チャネル番号がわかっていることを確認してください。

Cisco Nexus 9332PQ、9372PX、9372TX、93120TX スイッチ、および N9K-M6PQ または 汎用拡張モジュール（GEM）搭載の Cisco Nexus 9396PX、9396TX、93128TX スイッチについて、これらのデバイスで sFlow データ ソースとして設定されているすべてのアップリンク ポート用の sFlow および SPAN ACL TCAM リージョン サイズが設定されていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow data-source interface** [**ethernet** *slot*/*port*[-*port*] | **port-channel** *channel-number*]
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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**] **sflow data-source interface** [**ethernet** *slot*/*port*[-*port*]	**port-channel** *channel-number*] 例： `switch(config)# sflow data-source interface ethernet 1/5-12`	sFlow のサンプリング データ ソースを設定します。 イーサネットのデータ ソースの場合、*slot* はスロット番号、*port* は 1 つのポート番号または *port-port* で指定されたポートの範囲です。
ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

sFlow 拡張 BGP（Gateway）の設定

スイッチで sFlow 拡張 BGP を設定できます。

始める前に

sFlow が有効になっていることを確認します。

送信元ポートが、物理インターフェイスやポートチャネルなどの非SVI レイヤ3インターフェイスであることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow extended bgp**
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **sflow extended bgp** 例： `switch(config)# sflow extended bgp`	スイッチで拡張 bgp を設定します。 BGP がインストールされたルートへの宛先 IP アドレスを持つサンプリングされたsFlowパケットには、エクスポートされたsFlow レコードに拡張ゲートウェイ（bgp）データが含まれます。
ステップ 3	（任意） **show sflow** 例： `switch(config)# show sflow`	sFlow 設定を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

sFlow 設定の確認

sFlow 設定を表示するには、次のコマンドを使用します。

表 *22 : sFlow Show* コマンド

コマンド	目的
show sflow	sFlowサンプラーおよびsFlowエージェント設定のすべてのデータ ソースを表示します。
show process	sFlowプロセスが実行されているかどうかを確認します。
show running-config sflow [**all**]	現在実行中の sFlow コンフィギュレーションを表示します。

sFlow 統計情報のモニタリングとクリア

sFlow 統計情報を表示するには、**show sflow statistics** コマンドを使用します。

sFlow 統計情報をクリアするには、次のコマンドを使用します。

コマンド	説明
clear sflow statistics	**show sflow statistics** コマンドから sFlow 統計情報のほとんどをクリアします。
clear counters interface all	**show sflow statistics** コマンドの [トータル パケット（Total Packets）] フィールドをクリアします。
clear hardware rate-limiter sflow	**show sflow statistics** コマンドの [トータル サンプル（Total Samples）] フィールドをクリアします。

sFlow の設定例

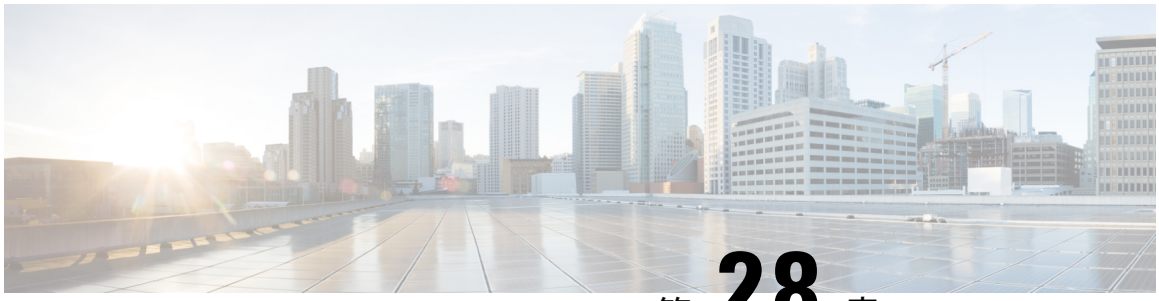
次に sFlow を設定する例を示します。

```
feature sflow
sflow sampling-rate 4096
sflow max-sampled-size 200
sflow counter-poll-interval 100
sflow max-datagram-size 2000
sflow collector-port 7000
sflow agent-ip 192.0.2.3
sflow collector-ip 192.0.2.5 vrf management
sflow data-source interface ethernet 1/5
```

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
ACL TCAM リージョン	IP ACL の設定



第 28 章

TAP アグリゲーション

この章では、Cisco NX-OS デバイスで TAP アグリゲーションおよび MPLS ストリッピングを設定する方法について説明します。

この章は、次の項で構成されています。

- TAP アグリゲーションについて （563 ページ）
- MPLS ストリッピングについて （567 ページ）
- TAP アグリゲーションの設定 （569 ページ）
- TAP アグリゲーションの設定の確認 （575 ページ）
- TAP アグリゲーションの設定例 （575 ページ）
- MPLS ストリッピングの設定 （576 ページ）
- MPLS ストリッピング設定の確認 （581 ページ）
- MPLS ストリッピング カウンタおよびラベル エントリのクリア （583 ページ）
- MPLS ストリッピングの設定例 （583 ページ）
- その他の参考資料 （584 ページ）

TAP アグリゲーションについて

ネットワーク TAP

さまざまなメソッドを使用して、パケットをモニタできます。1 つのメソッドでは、物理ハードウェア テスト アクセス ポイント（TAP）が使用されます。

ネットワーク タップは、ネットワークを通過するデータへの直接インライン アクセスが可能なので、トラフィックのモニターリングに非常に役立ちます。多くの場合、サードパーティがネットワーク内の 2 ポイント間のトラフィックをモニタします。ポイント A と B の間のネットワークが物理ケーブルで構成されている場合、ネットワーク TAP がこのモニタリングを実現する最良の方法になります。ネットワーク TAP には、少なくとも 3 つのポート（A ポート、B ポート、およびモニタ ポート）があります。A ポートと Bポートの間に挿入される TAP は、すべてのトラフィックをスムーズに通過させますが、同じデータをそのモニタ ポートにもコピーするため、サード パーティがリッスンできるようになります。

TAP には次の利点があります。

- 全二重データ伝送を処理可能。
- 目立たず、ネットワークによって検出されることがなく、物理または論理アドレッシングが不要
- 一部の TAP は、分散 TAP を構築する機能のあるフル インライン パワーをサポート

ネットワークのエッジまたは仮想エッジにおけるサーバー間データ通信に対する可視性を確保しようとする場合、またはネットワークのインターネット エッジで侵入防御システム（IPS）アプライアンスにトラフィックのコピーを提供する場合でも、ネットワーク TAP は、環境内のほぼすべての場所で使用できます。ただし、大規模環境にネットワーク タップを導入する場合、多くのコストがかかり、運用の複雑さが増し、ケーブル配線の問題が生じます。

TAP アグリゲーション

TAP アグリゲーションは、データ センターのタスクのモニタリングとトラブルシューティングに役立つ代替ソリューションです。複数のテスト アクセス ポイント（TAP）の集約を許可し、複数のモニタリング システムに接続するようにデバイスを指定することで機能します。タップ アグリゲーション スイッチは、監視する必要があるパケットを処理するネットワーク ファブリック内の特定のポイントにすべてのモニターリング デバイスをリンクします。

タップ アグリゲーション スイッチ ソリューションでは、Cisco Nexus 9000 シリーズ スイッチは、パケットのモニターリングに都合の良い、ネットワーク内のさまざまなポイントに接続されます。各ネットワーク要素から、スイッチド ポート アナライザ（SPAN）または光 TAP を使用して、この TAP]アグリゲーション スイッチにトラフィック フローを直接送信できます。TAP アグリゲーション スイッチ自体は、ネットワーク ファブリック内のイベントをモニタするために使用されるすべての分析ツールに直接接続されます。これらのモニタリング デバイスには、リモート モニタリング（RMON）プローブ、アプリケーション ファイアウォール、IPS デバイス、およびパケット スニファ ツールが含まれます。

特定のトラフィックをフィルタリングして 1 つ以上のツールにリダイレクトするように TAPアグリゲーション スイッチを設定できます。トラフィックを複数のインターフェイスにリダイレクトするために、マルチキャスト グループがスイッチの内部で作成され、リダイレクト リストの一部であるインターフェイスがメンバー ポートとして追加されます。リダイレクト アクションを持つアクセス コントロール リスト（ACL）ポリシーがインターフェイスに適用されると、作成された内部マルチキャスト グループに ACL ルールに一致するトラフィックがリダイレクトされます。

TAP 集約の注意事項と制約事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

TAP アグリゲーションに関する注意事項と制約事項は次のとおりです。

- TAP アグリゲーション：
  - すべての Cisco Nexus 9300 シリーズ スイッチおよび 3164Q、31128PQ、3232C と 3264Q スイッチでサポートされます。
  - 100G ポートでサポートされます。
  - スイッチ ポートおよび入力方向でのみサポートされます。
  - Cisco Nexus 9200、9300、および 9300-EX シリーズ スイッチの UDF ベースの一致で IPv4 ACL をサポートします。
  - Cisco Nexus 9300-FX、9300-FX2、9300-FX3、9300-GX、9300-GX2、9500-EX、および 9500-FX プラットフォーム スイッチでサポートされます。
  - サポートされるリダイレクト ポートの最大数は 32 インターフェイスです。
- Cisco NX-OS リリース 9.2(1) 以降、MPLS タグに基づく TAP アグリゲーション フィルタは、次の Cisco Nexus プラットフォーム スイッチでサポートされています。
  - 9700-EX および 9700-FX ライン カードを搭載した Cisco Nexus 9000 プラットフォーム スイッチ。
  - Cisco Nexus 9200 プラットフォーム スイッチ。
  - Cisco Nexus 9300 プラットフォーム スイッチ。
  - Cisco Nexus 9500 スイッチ。
- 次の Cisco Nexus シリーズ スイッチ、ライン カードおよびファブリック モジュールでは、MPLS タグでの TAP アグリゲーション フィルタはサポートされていません。

表 *23 : Cisco Nexus 9000* シリーズ スイッチ

Cisco Nexus 3164Q-40GE	Cisco Nexus 9372PX	Cisco Nexus 9372PX-E
Cisco Nexus 9372TX	Cisco Nexus 9372TX-E	Cisco Nexus 9332PQ
Cisco Nexus 3232C	Cisco Nexus 93120TX	Cisco Nexus 31128PQ
Cisco Nexus 3264Q-S	—	—

表 *24 : Cisco Nexus 9500* シリーズ ラインカードおよびファブリック モジュール

N9K-M6PQ	N9K-X9632PC-QSFP100	N9K-X9536PQ
N9K-S X9432C	N9K-C93128TX	N9K-C9396PX
N9K-X9432PQ	N9K-X9464TX	—

- Cisco Nexus 9700-EX および 9700-FX ライン カードは、IPv4、IPv6、および MAC ACL による TAP アグリゲーションをサポートします。
- レイヤ 2 インターフェイスのみが TAP アグリゲーション ポリシーをサポートします。レイヤ 3 インターフェイスにポリシーを設定できますが、そのポリシーは機能しなくなります。
- リダイレクト ポートは、送信元（TAP）ポートと同じ VLAN の一部である必要があります。
- 各ルールは、1 つの固有の一致基準とのみ関連付ける必要があります。
- TAP アグリゲーション ポリシー用インターフェイスのリストを入力する場合は、スペースではなくカンマでエントリを区切る必要があります。たとえば、port-channel50、ethernet1/12、port-channel20 などです。
- ポリシーにターゲット インターフェイスを指定する場合、簡略版ではなく、完全なインターフェイス タイプを入力する必要があります。たとえば、**eth1/1** の代わりに **ethernet1/1** を入力し、**po50** の代わりに **port-channel50** を入力します。
- *tcp-option-length* と *VLAN ID* フィルタを同時に使用する HTTP 要求はサポートされていません。両方のフィルタを同時に設定すると、ACE に対するトラフィック照合が機能しない場合があります。
- Cisco NX-OS リリース 10.2(1)F 以降では、TAP アグリゲーション機能はライセンスによるもので、関連する CLI を構成する前に、機能の TAP アグリゲーションを構成する必要があります。ただし、TAP アグリゲーションに依存する CLI の使用が以前の設定で見つかった場合、この機能は sysmgr の ISSU インフラ変換フェーズ中に自動生成されます。この機能は、すべての Cisco Nexus 9000 シリーズ スイッチでサポートされています。ライセンスの詳細については、『ポリシー ガイドを使用する *Cisco Nexus 9000 NX-OS* スマート ライセンシング』を参照してください。
- Cisco NX-OS リリース 10.2(2)F 以降では、L2 インターフェイスに TapAgg ACL をアタッチする前に、**mode tap-aggregation** コマンドを設定するようにしてください。
- まだ設定されていないポート チャネルへのリダイレクトを使用して ACL エントリを設定する場合、ユーザーは指定されたポート チャネルを後で設定するように注意する必要があります。
- Cisco NX-OS リリース 10.3(1)F 以降、選択的な Q-in-Q トランク モードのインターフェイスでは、プロバイダー VLAN タギングが Cisco Nexus 9300-GX、N9K-C9504-FM-G、および N9K- C9508-FM-G スイッチおよび N9K-X9716D-GX ライン カードでサポートされていますが、以下の制限があります。
  - VXLAN が有効になっている場合、この機能はサポートされません。
  - システム レベル全体で最大 7000 の外部 VXLAN レイト エントリ、およびポートごとに 4000 のエントリを持つことができます。
- 入力インターフェイスで二重 VLAN タグを許可するには、次のように **switchport trunk allow-multi-tag** コマンドを正しく構成する必要があります。

- Cisco Nexus 9300-FX2 スイッチでは、NDB が構成されている場合に限りこのコマンドを使用する必要があります。
- Cisco Nexus 9300-GX/GX2 スイッチでは、NDB が構成されている場合でもこのコマンドは必要ありません。

MPLS ストリッピングについて

Cisco Nexus 9000 シリーズ スイッチの入力ポートは、さまざまなマルチプロトコル ラベル スイッチング（MPLS）パケット タイプを受信します。MPLS ネットワークの各データ パケットには、1 つ以上のラベル ヘッダーがあります。これらのパケットはリダイレクト アクセス コントロール リスト（ACL）に基づいてリダイレクトされます。

ラベルは、Forwarding Equivalence Class（FEC）を特定するために使用される短い 4 バイトの固定長のローカルで有効な識別子です。特定のパケットに設定されているラベルは、そのパケットが割り当てられている FEC を表します。次のコンポーネントがあります。

- Label：ラベルの値（非構造化）、20 ビット
- Exp：試験的使用、3 ビット、現在、サービス クラス（CoS）フィールドとして使用
- S：スタックの一番下、1 ビット
- TTL：存続可能時間、8 ビット

標準のネットワーク モニタリング ツールでは、MPLS トラフィックのモニタリングと分析はできません。標準のネットワーク監視ツールでMPLSトラフィックを監視できるようにするには、MPLS ストリップ機能を有効にする必要があります。この機能は、トラフィックの MPLS ラベル ヘッダーを取り除き、トラフィックをモニタリング デバイスにリダイレクトします。

MPLS ストリッピングに関する注意事項と制限事項

（注） スケールの情報については、リリース特定の『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

MPLS ストリッピングに関する注意事項と制約事項は次のとおりです。

- Cisco Nexus 9700-EX および 9700-FX ライン カードは、MPLS ストリッピングをサポートしていません。
- Cisco NX-OS リリース 10.2(1)F 以降、すべてのタップ アグリゲーションおよびストリッピング機能に対して**機能タップ アグリゲーション**を有効にする必要があります。
- MPLS ストリッピングを有効にする前に、すべてのレイヤ 3 および vPC 機能を無効にします。

- スタティック MPLS、MPLS セグメント ルーティング、および MPLS ストリッピングを同時に有効にすることはできません。
- MPLS ストリッピングに関係する入力インターフェイスで、TAP 集約が有効になっている必要があります。
- 目的の宛先にパケットを転送するためには、入力インターフェイスのリダイレクト アクションを使用してタップ アグリゲーション ACL を設定する必要があります。
- MPLS ストリップ後、SMAC はスイッチ mac (**show vdc**) に変更され、DMAC は **00:00:00:ab:cd:ef** に設定されます。
- 削除されたパケットが出力される出力インターフェイスは、許可 VLAN としての VLAN 1 が存在するインターフェイスである必要があります。出力インターフェイスは、デフォルトですべての VLAN が許可されるトランクとして設定することを推奨します。
- ストリッピングは IP PACL に基づいており、ストリッピングに MAC-ACL を使用することはできません。
- MPLS ストリッピングは、IPv4 トラフィックに対してのみサポートされます。
- MPLS ストリッピング パケットの場合、ポートチャネル ロード バランシングがサポートされます。
- レイヤ 3 ヘッダー ベースのハッシュおよびレイヤ 4 ヘッダー ベースのハッシュはサポートされていますが、レイヤ 2 ヘッダー ベースのハッシュはサポートされていません。
- MPLS ストリッピング中、着信 VLAN は維持されません。
- Cisco Nexus 9200、9300-EX、および9300-FX プラットフォーム スイッチは、リダイレクト ポートから送信されるパケットへの VLAN のタギングをサポートします。入力/出力ポートは、イーサネットまたはポート チャネルのいずれかです。VLAN タグは、着信ポート設定から取得されます。入力インターフェイスの新しい ACL を、インターフェイス VLAN 値とは異なる VLAN 値に関連付けないでください。
- 一意のリダイレクト ポート リストを持つすべての ACE（特定の VLAN に関連付けられた ACL の下で）に対して、ハードウェア エントリを割り当てます。現在の ACE 数のハードウェア制限は 50 で、50 を超える ACE を設定することはできません。
- MPLS ストリップは、MPLS ラベル スタックのレイヤ 3 パケットでのみサポートされます。
- Cisco NX-OS Release 10.2(2)F 以降では、IPv6 は Cisco Nexus 9300-EX プラットフォーム スイッチでのみサポートされます。ただし、VPLS ストリップおよび制御ワードパケット ストリップはサポートされていません。
- Cisco NX-OS リリース 10.2(3)F 以降、OFM ベースの MPLS ストリッピングが追加されています。新しい OFM ベースの MPLS ストリッピングと従来の実装は共存できません。詳細については、Nexus Data Broker のヘッダ ストリッピング機能の構成 （591 ページ） の OFM ベースの MPLS ヘッダー ストリップのセクションを参照してください。

- 新しい OFM ベースの MPLS ストリッピング機能は、展開で MPLS ストリッピングと、VXLAN、iVXLAN、GRE、ERSPAN ヘッダーなどの他のタイプのヘッダー ストリッピングとの共存が必要な場合にのみ使用します。

  他のストリッピング機能との共存が必要ない場合、既存の MPLS ストリッピング機能は、MPLS ストリッピングを引き続きサポートします。

- Cisco NX-OS Release 10.3（2）F 以降では、EoMPLS ラベル ストリッピングは Cisco Nexus 9300-FX ToR スイッチでもサポートされます。

TAP アグリゲーションの設定

ライン カードの TAP 集約のイネーブル化

Cisco NX-OS リリース 7.0(3)I7(2) 以降では、9700-EX および 9700-FX ライン カードを備えた Cisco Nexus 9500 プラットフォーム スイッチの TAP 集約を有効にできます。

手順の概要

1. **configure terminal**
2. [**no**] **hardware acl tap-agg**
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **hardware acl tap-agg** 例： `switch(config)# hardware acl tap-agg`	Cisco Nexus 9700-EX および 9700-FX ライン カードの TAP 集約を有効にします。 このコマンドは、Cisco Nexus 9300-GX および 9300-GX2 プラットフォーム スイッチでも必要であり、リロードが必要になる場合があります。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

TAP 集約ポリシーの設定

IP アクセス コントロール リスト（ACL）または MAC ACL で、TAP アグリゲーション ポリシーを設定できます。

始める前に

IPv4 ポート ACL または MAC ポート ACL 用の ACL TCAM のリージョン サイズは、**hardware access-list tcam region** {*ifacl* | *mac-ifacl*} コマンドを使用して設定する必要があります。**hardware access-list team region ipv6-ifcal** コマンドを使用して、IPv6 ポート ACL の ACL TCAM リージョン サイズを設定します。

詳細については、『Cisco Nexus 9000 シリーズ NX-OS セキュリティの設定ガイド』の「ACL TCAM リージョン サイズの設定」を参照してください。

（注） デフォルトでは、ifacl と mac-ifacl の両方の領域サイズはゼロです。TAP 集約をサポートするには、ifacl または mac-ifacl リージョンに十分なエントリを割り当てる必要があります。

手順の概要

1. **configure terminal**
2. **feature tap-aggregation**
3. 次のいずれかのコマンドを入力します。
   - **ip access-list** *access-list-name*
   - **mac access-list** *access-list-name*
4. （任意） **statistics per-entry**
5. [**no**] **permit** *protocol source destination* **redirect** *interfaces*
6. （任意） 次のいずれかのコマンドを入力します。
   - **show ip access-lists** [*access-list-name*]
   - **show mac access-lists** [*access-list-name*]
7. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します

	コマンドまたはアクション	目的

ステップ 2	**feature tap-aggregation**
	例：
	`switch(config)# feature tap-aggregation`
	`switch(config)#`
		タップ集約に関連する CLI を設定できます。
		（注）
		Cisco NX-OSリリース10.2(1)F 以降、以前のリリースからこの機能を備えた新しい NX-OS リリースへのソフトウェア アップグレードでは、サポートされているマトリックスで ISSU が完了した場合、機能タップアグリゲーション設定が自動的に生成されます。

ステップ 3	次のいずれかのコマンドを入力します。
	• **ip access-list** *access-list-name*
	• **mac access-list** *access-list-name*
	例：
	`switch(config)# ip access-list test`
	`switch(config-acl)#`
	`switch(config)# mac access-list mactap1`
	`switch(config-mac-acl)#`
		IP ACL を作成して IP アクセス リスト コンフィギュレーション モードを開始するか、あるいは MAC ACL を作成して MAC アクセス リスト コンフィギュレーション モードを開始します。

ステップ 4	（任意） **statistics per-entry** 例： `switch(config-acl)# statistics per-entry`	各エントリで許可または拒否されるパケット数の統計情報の記録を開始します。
ステップ 5	[**no**] **permit** *protocol source destination* **redirect** *interfaces* 例： `switch(config-acl)# permit ip any any redirect ethernet1/8`	条件ごとにトラフィックのリダイレクトを許可する IP または MAC AC Lルールを作成します。このコマンドの**いずれの**バージョンも、ポリシーからのパーミッションを削除することはありません。 （注） TAP集約ポリシーのインターフェイスを入力するときは、それを省略しないでください。インターフェイスのリストを入力するときは、コンマで区切り、スペースを入れないでください。

ステップ 6	（任意） 次のいずれかのコマンドを入力します。
	• **show ip access-lists** [*access-list-name*]
	• **show mac access-lists** [*access-list-name*]
	例：
	`switch(config-acl)# show ip access-lists test`
	`switch(config-mac-acl)# show mac access-lists mactap1`
		すべての IPv4 または MAC ACL、あるいは特定の IPv4 または MAC ACL 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	（任意） **copy running-config startup-config** 例： `switch(config-acl)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

TAP アグリゲーション ポリシーのインターフェイスへのアタッチ

TAP アグリゲーションで設定された ACL をレイヤ 2 インターフェイスに適用できます。

手順の概要

1. **configure terminal**
2. **interface** *type slot/port*
3. **switchport**
4. 次のいずれかのコマンドを入力します。
   - [**no**] **ip port access-group** *access-list-name* **in**
   - [**no**] **mac port access-group** *access-list-name* **in**
5. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 2/2`
	`switch(config-if)#`
		指定したインターフェイスに対してインターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**switchport** 例： `switch(config-if)# switchport`	レイヤ 3 インターフェイスをレイヤ 2 インターフェイスに変更します。 （注） インターフェイスがレイヤ2インターフェイスであることを確認します。

	コマンドまたはアクション	目的

ステップ 4	次のいずれかのコマンドを入力します。
	• [**no**] **ip port access-group** *access-list-name* **in**
	• [**no**] **mac port access-group** *access-list-name* **in**
	例：
	`switch(config-if)# ip port access-group test in`
	`switch(config-if)# mac port access-group test in`
		TAP 集約で設定された IPv4 または MAC ACL をインターフェイスに適用します。このコマンドの **no** 形式を使用すると、インターフェイスから ACL を削除します。

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

プロバイダー VLAN で選択的 Q-in-Q を構成する

始める前に

プロバイダー VLAN を設定する必要があります。

手順の概要

1. **configure terminal**
2. **interface** *[インターフェイス id（interface-id）]*
3. **switchport**
4. **switchport mode trunk**
5. 次のいずれかのコマンドを入力します。
   - **switchport vlan mapping** *[vlan-id 範囲（vlan-id-range）]***dot1q-tunnel** *[外部 vlan-id（outer vlan-id）]*
   - **switchport vlan mapping all dot1q-tunnel** *[外部 vlan-id（outer vlan-id）]*
6. **switchport trunk allowed vlan** *vlan_list*
7. 次のいずれかのコマンドを入力します。
   - [**no**] **ip port access-group** *access-list-name* **in**
   - [**no**] **mac port access-group** *access-list-name* **in**
8. （任意） **mode tap-aggregation**
9. （任意） **copy running-config startup-config**
10. switch(config-if)# **exit**
11. （任意） switch(config-if)# **show interfaces** *interface-id* **vlan mapping**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *[インターフェイス id（interface-id）]* 例： `switch(config)# interface Ethernet1/1`	サービス プロバイダ ネットワークに接続されるインターフェイスのインターフェイス コンフィギュレーションモードを開始します。物理インターフェイスまたは EtherChannel ポート チャネルを入力できます。
ステップ 3	**switchport** 例： `switch(config if)# switchport`	インターフェイスをレイヤ 2 スイッチング ポートとして設定します。
ステップ 4	**switchport mode trunk** 例： `switch(config-if)# switchport mode trunk`	インターフェイスをレイヤ 2 トランク ポートとして設定します。
ステップ 5	次のいずれかのコマンドを入力します。 • **switchport vlan mapping** *[vlan-id 範囲（vlan-id-range）]***dot1q-tunnel** *[外部 vlan-id（outer vlan-id）]* • **switchport vlan mapping all dot1q-tunnel** *[外部 vlan-id（outer vlan-id）]* 例： `switch(config-if)# switchport vlan mapping all dot1q-tunnel 300`	マッピングする VLAN ID を入力します。 • vlan-id-range1：カスタマー ネットワークからスイッチに入るカスタマー VLAN ID（C-VLAN）の範囲。指定できる範囲は 1 ～ 4094 です。VLAN-ID のストリングを入力できます。 • outer vlan-id：サービス プロバイダー ネットワークの外部 VLAN ID（S-VLAN）を入力します。指定できる範囲は 1 ～ 4094 です。
ステップ 6	**switchport trunk allowed vlan** *vlan_list* 例： `switch(config-if)# switchport trunk allowed vlan 300`	トランク インターフェイスの許可 VLAN を設定します。
ステップ 7	次のいずれかのコマンドを入力します。 • [**no**] **ip port access-group** *access-list-name* **in** • [**no**] **mac port access-group** *access-list-name* **in** 例：	TAP 集約で設定された IPv4 または MAC ACL をインターフェイスに適用します。このコマンドの **no** 形式を使用すると、インターフェイスから ACL を削除します。

	コマンドまたはアクション	目的

	`switch(config-if)# ip port access-group test in`
	
	`switch(config-if)# mac port access-group test in`	

ステップ 8	（任意） **mode tap-aggregation**
	例：
	`switch(config-if)# mode tap-aggregation`
	`switch(config-if)# no shutdown`
		TAP アグリゲーション ポリシーを設定した ACL のインターフェイスへのアタッチメントを禁止します。

ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。
ステップ 10	switch(config-if)# **exit**	コンフィギュレーション モードを終了します。
ステップ 11	（任意） switch(config-if)# **show interfaces** *interface-id* **vlan mapping**	マッピングの設定の確認

TAP アグリゲーションの設定の確認

TAP アグリゲーションの設定情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show ip access-lists [*access-list-name*]	すべての IPv4 ACL または特定の IPv4 ACL を表示します。
show ip access-lists [*access-list-name*]	すべての MAC ACL または特定の MAC ACL を表示します。

TAP アグリゲーションの設定例

次に、IPv4 ACL で TAP アグリゲーション ポリシーを設定する例を示します。

```
switch# configure terminal
switch(config)# feature tap-aggregation
switch(config)# ip access-list test
switch(config-acl)# 10 deny ip 100.1.1/24 any
switch(config-acl)# 20 permit tcp any eq www any redirect port-channel4
switch(config-acl)# 30 permit ip any any redirect
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
switch(config-acl)# show ip access-lists test
IP access list test
        10 deny ip 100.1.1/24 any
        20 permit tcp any eq www any redirect port-channel4
        30 permit ip any any redirect
```

```
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
```

次に、MAC ACL で TAP アグリゲーション ポリシーを設定する例を示します。

```
switch# configure terminal
switch(config)# feature tap-aggregation
switch(config)# mac access-list mactap1
switch(config-mac-acl)# 10 permit any any 0x86dd redirect port-channel1
switch(config-mac-acl)# show mac access-lists mactap1
MAC access list mactap1
        10 permit any any 0x86dd redirect port-channel1
```

次に、TAP アグリゲーション ポリシーをレイヤ 2 インターフェイスにアタッチする例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# ip port access-group test in
switch(config-if)#
```

MPLS ストリッピングの設定

MPLS ストリッピングの有効化

MPLS ストリッピングをグローバルに有効にできます。

始める前に

MPLS ストリッピングを有効にする前に、すべてのレイヤ 3 および vPC 機能を無効にします。

mode tap-aggregation コマンドを使用して、TAP アグリゲーション ポリシーを含む ACL をレイヤ 2 インターフェイスまたはポート チャネルにアタッチします。詳細については、TAP アグリゲーション ポリシーのインターフェイスへのアタッチ （572 ページ）を参照してください。

手順の概要

1. **configure terminal**
2. [**no**] **mpls strip**
3. [**no**] **mpls strip mode dot1q**
4. **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	[**no**] **mpls strip** 例： `switch(config)# mpls strip`	MPLS ストリッピングをグローバルに有効にします。このコマンドの **no** 形式を使用すると、MPLS ストリッピングが無効化されます。
ステップ 3	[**no**] **mpls strip mode dot1q** 例： `switch(config)# mpls strip mode dot1q`	リダイレクト ポートからのパケットの VLAN タギングを有効にします。タグ付けする必要がある VLAN は、入力ポートで指定する必要があります。
ステップ 4	必須: **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

VLAN タグの着信ポートの設定

VLAN タグは、着信ポート設定から取得されます。入力/出力ポートは、イーサネットまたはポート チャネルのいずれかです。

手順の概要

1. **configure terminal**
2. **interface** *type slot/port*
3. **switchport**
4. 次のいずれかのコマンドを入力します。
   - [**no**] **ip port access-group** *access-list-name* **in**
   - [**no**] **mac port access-group** *access-list-name* **in**
5. 次のいずれかのコマンドを入力します。
   - [**no**] **ip port access-group** *access-list-name* **in**
   - [**no**] **mac port access-group** *access-list-name* **in**
6. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 1/26`
	`switch(config-if)#`
		指定したインターフェイスに対してインターフェイス コンフィギュレーション モードを開始します。

ステップ 3	**switchport** 例： `switch(config-if)# switchport`	レイヤ 3 インターフェイスをレイヤ 2 インターフェイスに変更します。 （注） インターフェイスがレイヤ2インターフェイスであることを確認します。

ステップ 4	次のいずれかのコマンドを入力します。
	• [**no**] **ip port access-group** *access-list-name* **in**
	• [**no**] **mac port access-group** *access-list-name* **in**
	例：
	`switch(config-if)# ip port access-group test in`
	`switch(config-if)# mac port access-group test in`
		TAP 集約で設定された IPv4 または MAC ACL をインターフェイスに適用します。このコマンドの **no** 形式を使用すると、インターフェイスから ACL を削除します。

ステップ 5	次のいずれかのコマンドを入力します。
	• [**no**] **ip port access-group** *access-list-name* **in**
	• [**no**] **mac port access-group** *access-list-name* **in**
	例：
	`switch(config-if)# ip port access-group test in`
	`switch(config-if)# mac port access-group test in`
		TAP 集約で設定された IPv4 または MAC ACL をインターフェイスに適用します。このコマンドの **no** 形式を使用すると、インターフェイスから ACL を削除します。

ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

MPLS ラベルの追加と削除

デバイスは、フレームが TAP インターフェイスで不明なラベルを受信するたびにラベルを動的に学習できます。また、スタティック MPLS ラベルを追加または削除できます。

始める前に

TAPアグリゲーションポリシーを設定してインターフェイスへアタッチする詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

目的の宛先にパケットを転送するためには、入力インターフェイスのリダイレクトアクションを使用してタップ アグリゲーション ACL を設定する必要があります。

手順の概要

1. **configure terminal**
2. **mpls strip label** ラベル
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**mpls strip label** ラベル 例： `switch(config)# mpls strip label 100`	指定したスタティック MPLS ラベルを追加します。ラベルの 20 ビット値の範囲は 1 ～ 1048575 です。 （注） このCLIは、次のクラウドスケールプラットフォーム スイッチを除き、「注意事項と制限事項」の項でMPLSストリッピング機能に指定されたすべてのプラットフォーム スイッチで使用できます。 • N9K-C93180YC-EX • N9K-C93180YC-FX • N9K-C93240YC-FX2 • N9K-C93180YC-FX3 • N9K-C93600CD-GX

	コマンドまたはアクション	目的
		[**no**] **mpls strip label** {*label* | **all**} コマンドは、指定したスタティック MPLS ラベルを削除します。**all** オプションは、すべてのスタティック MPLS ラベルを削除します。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

宛先 MAC アドレスの設定

削除された出力フレームの宛先 MAC アドレスを設定できます。

手順の概要

1. **configure terminal**
2. **mpls strip dest-mac** *mac-address*
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**mpls strip dest-mac** *mac-address* 例： `switch(config)# mpls strip dest-mac 1.1.1`	ヘッダーが削除された出力フレームの宛先 MAC アドレスを指定します。 MAC アドレスは、次の 4 つのいずれかの形式で指定できます。 • E.E.E • EE-EE-EE-EE-EE-EE • EE:EE:EE:EE:EE:EE • EEEE.EEEE.EEEE
ステップ 3	（任意） **copy running-config startup-config** 例：	実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

	コマンドまたはアクション	目的
	`switch(config)# copy running-config startup-config`	

MPLS ラベル エージングの設定

使用されていないダイナミック MPLS ラベルがエージ アウトする時間を定義できます。

手順の概要

1. **configure terminal**
2. **mpls strip label-age** 経過期間
3. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**mpls strip label-age** 経過期間 例： `switch(config)# mpls strip label-age 300`	ダイナミック MPLS ラベルがエージアウトする時間を指定します（秒）。範囲は 61ー 31622400 です。
ステップ 3	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

MPLS ストリッピング設定の確認

MPLS ストリッピングの設定を表示するには、次のいずれかの作業を行います。

コマンド	目的
show mpls strip labels [*label* | **all** | **dynamic** | **static**]	MPLS ラベルに関する情報を表示します。次のオプションを指定できます。 • *label*：表示するラベル • **all**：すべてのラベルを表示することを指定します。これがデフォルトのオプションです。 • **dynamic**：ダイナミック ラベルのみ表示することを指定します。 • **static**：スタティック ラベルのみ表示することを指定します。

次に、すべての MPLS ラベルを表示する例を示します。

```
switch# show mpls strip labels
MPLS Strip Labels:
    Total      : 3005
    Static     : 5
Legend:     * - Static Label
    Interface - where label was first learned
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
--------------------------------------------------------------------------------
     Label    Interface      Idle-Age   SW-Counter       HW-Counter
--------------------------------------------------------------------------------
     4096    Eth1/53/1            15            1              210
     4097    Eth1/53/1            15            1              210
     4098    Eth1/53/1            15            1              210
     4099    Eth1/53/1             7            2              219
     4100    Eth1/53/1             7            2              219
     4101    Eth1/53/1             7            2              219
     4102    Eth1/53/1            39            1              206
     4103    Eth1/53/1            39            1              206
     4104    Eth1/53/1            39            1              206
     4105    Eth1/53/1             1            1              217
     4106    Eth1/53/1             1            1              217
     4107    Eth1/53/1             1            1              217
     4108    Eth1/53/1            15            1              210
*   25000    None <User>          39            1              206
*   20000    None <User>          39            1              206
*   21000    None <User>           1            1              217
```

次に、スタティック MPLS ラベルのみ表示する例を示します。

```
switch(config)# show mpls strip labels static
MPLS Strip Labels:
    Total      : 3005
    Static     : 5
Legend:     * - Static Label
    Interface - where label was first learned
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
```

```
--------------------------------------------------------------------------------
     Label    Interface      Idle-Age   SW-Counter       HW-Counter
--------------------------------------------------------------------------------
*     300    None <User>         403            0                0
*     100    None <User>         416            0                0
*   25000    None <User>         869            0                0
*   20000    None <User>         869            0                0
*   21000    None <User>         869            0                0
```

MPLS ストリッピング カウンタおよびラベル エントリのクリア

MPLS ストリッピング カウンタとラベル エントリをクリアするには、次の作業を行います。

コマンド	目的
clear mpls strip label dynamic	MPLS ラベル テーブルからダイナミック ラベル エントリをクリアします。
clear counters mpls strip	すべての MPLS ストリッピング カウンタをクリアします。

次に、すべての MPLS ストリッピング カウンタをクリアする例を示します。

```
switch# clear counters mpls strip
switch# show mpls strip labels
MPLS Strip Labels:
    Total      : 15000
    Static     : 2
Legend:     * - Static Label
    Interface - where label was first learned
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
--------------------------------------------------------------------------------
     Label    Interface      Idle-Age   SW-Counter       HW-Counter
--------------------------------------------------------------------------------
     4096    Eth1/44              15            0                0
     8192    Eth1/44              17            0                0
    12288    Eth1/44              15            0                0
    16384    Eth1/44              39            0                0
    20480    Eth1/44              47            0                0
    24576    Eth1/44               7            0                0
    28672    Eth1/44               5            0                0
    36864    Eth1/44               7            0                0
    40960    Eth1/44              19            0                0
    45056    Eth1/44               9            0                0
    49152    Eth1/44              45            0                0
    53248    Eth1/44               9            0                0
```

MPLS ストリッピングの設定例

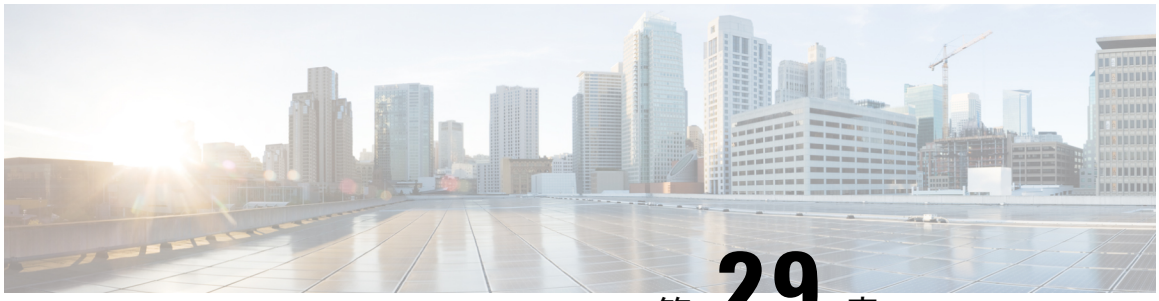
次に、スタティック MPLS ラベルを追加する例を示します。

```
switch# configure terminal
switch(config)# mpls strip label 100
switch(config)# mpls strip label 200
switch(config)# mpls strip label 300
```

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
IP ACL	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
MAC ACL	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
ポートチャネル対称ハッシュ	『*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*』
リモート モニタリング（RMON）	RMON の設定 （322 ページ）
スイッチド ポート アナライザ（SPAN）	SPAN の設定 （413 ページ）
トラブルシューティング	『*Cisco Nexus 9000 Series NX-OS Troubleshooting Guide*』



第 29 章

重複排除の構成

この章では、Cisco NX-OS デバイスで重複排除を構成する方法について説明します。

この章は、次の項で構成されています。

- 重複排除について （585 ページ）
- 注意事項と制約事項 （586 ページ）
- 重複排除の構成 （587 ページ）

重複排除について

Cisco NX-OS リリース 10.4(1)F 以降では、重複排除機能を使用して、Nexus データ ブローカー（NDB）のスイッチを通過するデータ フローの重複コピーを削除できます。

データフローが継続的に増加すると、ネットワーク内を移動し、異なるアプリケーションで処理されたデータが、ソースとターゲットの両方でデータの重複につながります。効率的なデータ管理、セキュリティ、およびストレージのために、重複排除を使用して重複データを削除できます。

重複排除機能は、トラフィック アナライザまたはデータ ストレージ ツールから受信した重複トラフィックを排除します。この機能は、NDB スイッチで転送される重複フローを識別します。重複排除は、次の 2 つのモデル タイプをサポートします。

- インライン重複排除モデル（インターフェイス モード）

  既存の NDB スイッチでインライン モデルを構成できます。これにより、インターフェイスを通過する重複パケットが排除されます。最初のインターフェイス パケットは転送されますが、その後で他のインターフェイスから来た重複インターフェイス パケットはドロップされます。

- スティック モデル重複排除（VLAN モード）

  このモデルでは、パケット フローは特定の VLAN にバインドされます。パケット フローは最初の VLAN では許可されますが、他の VLAN を通る重複フローは制限されます。フローが NDB スイッチに入ると、パケットは特定の VLAN にタグ付けされます。各パケットには一意の VLAN がタグ付けされます。VLAN タグが付いたパケットは重複排除スイッチに転送され、重複したフローはそのスイッチで削除されます。

注意事項と制約事項

次に、重複排除の注意事項と制限事項を示します。

- 重複排除は、内部パケット フローに対してのみ構成できます。
- TCP および UDP フローがフロー重複排除でサポートされます。
- 重複排除は、ローカル SPAN および光 TAP フロー セッションでサポートされます。
- 重複排除は、単一カプセル化された VXLAN および GRE トンネル パケットに対して実行できます。
- 重複排除は、Cisco Nexus 9300-FX2/FX3 および 9300-GX/GX2 拡張スイッチでサポートされています。
- 次のデバイスでは、重複排除はサポートされていません。
  - Cisco Nexus 9500 プラットフォーム スイッチ
  - Cisco Nexus 9300-C、9300-EX、9300-FX および EOR スイッチ
  - Cisco N9K-C9408 スイッチ
- ERSPAN などの同一インターフェイスで重複パケットの重複排除を設定することはできません。インターフェイスまたは SVI で複数の ERSPAN セッションを終了させる場合があります。これは、同じフローのコピーが同じインターフェイスまたは VLAN に到達するかどうかに影響します。スイッチは、重複排除の際に異なる ERSPAN セッションを区別できません。
- 存続期間が短いフローの複製フローは削除できません。
- 類似した 5 つのタプルを持つ異なる VRF からのフロー パケットは削除できません。重複排除は、パケットの 5 タプルを使用して実行されるフロー用だからです。VRF パケットは識別できません。
- 重複排除スティック モデル（VLAN モデル）は、トンネル終端および Q-in-Q ポートではサポートされません。
- 重複排除は、IPv6 およびマルチキャスト フローではサポートされません。
- ダイナミック エージングは 128k フローでのみサポートされています。
- ISSU 中、重複排除はデフォルトで無効になります。ISSU 後すべてのフローがクリアされ、更新されます。
- トンネル トラフィックで重複排除を適切に機能させるには、フロー終了を有効にする必要があります。

重複排除の構成

Cisco NX-OS リリース 10.4(1)F 以降では、重複排除機能を使用して、Nexus Data Broker（NDB）のスイッチを通過するデータ フローの重複コピーを削除できます。

構成の変更を有効にするには、重複排除を構成した後にスイッチをリロードしてください。

手順の概要

1. **configure terminal**
2. **tap-aggregation flow-deduplication**
3. （任意） **absolute-timer***time in minutes*
4. （任意） **dynamic-timer***time in milli seconds*
5. **mode** {**vlan**|**interface**}
6. **clear hardware deduplication statistics**{**slot**|**module in number**}

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**tap-aggregation flow-deduplication** 例： `switch(config-dedup)#`	スイッチでフロー重複排除機能を有効にします。

ステップ 3	（任意） **absolute-timer***time in minutes*
	例：
	`switch(config)# absolute-timer 10`
	`switch(config-dedup)#`
		パケット フローの重複排除に絶対タイマーを構成できます。間隔の範囲は 0 ～ 1440 分です。

ステップ 4	（任意） **dynamic-timer***time in milli seconds*
	例：
	`switch(config)# dynamic-timer 2000`
	`switch(config-dedup) mode interface#`
		パケット フローの重複排除に動的タイマーを構成できます。間隔の範囲は 0 ～ 300000 ミリ秒です。

ステップ 5	**mode** {**vlan**|**interface**}
	例：
	`switch(config)# interface`
	`switch(config-dedup)#`
		必要なモードで重複排除を構成できます。
		（注）
		スイッチで重複の削除を構成するには、構成を保存してスイッチをリロードしてください。

	コマンドまたはアクション	目的
ステップ 6	**clear hardware deduplication statistics**{**slot**|**module in number**}	必要なスロットまたはモジュールの重複排除をクリアします。モジュール番号の範囲は 1 ～ 30 です。

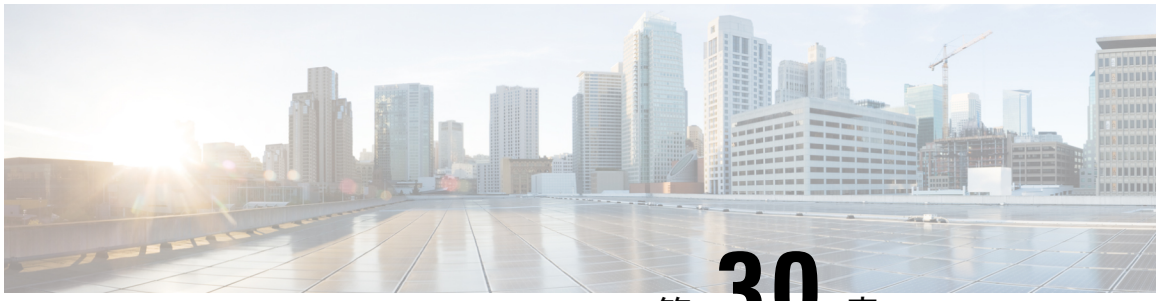
例

次に、参考までに出力例を示します。

```
switch# show hardware deduplication summary
slot 1
=======
Deduplication      : Enabled
Dedup Mode         : Interface
Dynamic timer                 : 200000 milliseconds
Absolute timer                : 5 minutes
Max Supported Flows           : 240K
Total number of learned flows : 240000
Total number of dropped bytes : 65698869600
```

```
switch# show hardware deduplication detail
slot1
================================================================================
   Dedup Flows
================================================================================
SourceIP  Destination IP  Ports(Src:Dst) Protocol    Interface     Learn-time
================================================================================
33.1.1.2  12.1.1.2   3000 :3001  6  Eth1/1  07/28/2023 11:47:09.532376
55.1.1.2  12.1.1.2   15000:15001  17     Eth1/1  07/28/2023 11:47:09.532229
11.1.1.2  12.1.1.2   1841 :1842  6  Eth1/1  07/28/2023 11:47:09.532340
1.22.1.2  1.12.1.2   2000 :2001  6  Eth1/1  07/28/2023 11:47:09.532428
1.44.1.2  1.12.1.2   4000 :4001  6  Eth1/23  07/28/2023 11:47:09.532133
```

```
switch#show hardware deduplication age-history
slot 1
================================================================================
    Dedup Flows
================================================================================
Source  Destination Ports  Protocol Interface Timer    Learn-Time    Aged-Time
IP  IP   (Src:Dst)
================================================================================
1.44.1.2 1.12.1.2   4000:4001 6  Eth1/17 Dynamic   08/05/2023 2:24:49.26020     08/05/2023
 12:33:29.21904
33.1.1.2  12.1.1.2  3000:3001       6  Eth1/27 Dynamic  08/05/2023 12:24:49.126246
08/05/2023 12:33:29.21945
55.1.1.2  12.1.1.2  15000:15001    17  Eth1/5 Dynamic    08/05/2023 12:24:49.26070
08/05/2023 12:33:29.21957
1.22.1.2  1.12.1.2   2000 :2001     6  Eth1/5 Dynamic    08/05/2023 12:24:49.26115
08/05/2023 12:33:29.21969
11.1.1.2  12.1.1.2   1841 :1842     6  Eth1/17 Dynamic   08/05/2023 12:24:49.25949
08/05/2023 12:33:29.21979
```



第 30 章

MPLS アクセス リストの構成

- MPLS アクセス リストの構成 （589 ページ）
- MPLS アクセス リスト構成の検証 （590 ページ）
- MPLS アクセス リストの構成例 （590 ページ）

MPLS アクセス リストの構成

MPLS アクセス リストを使用すると、MPLS ラベルに基づいて MPLS パケットをフィルタリングし、フィルタリングされたパケットを構成済みのリダイレクト インターフェイスに送信できます。

手順の概要

1. **configure terminal**
2. **[no]install feature-set mpls**
3. **mpls access list mpls-acl**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**[no]install feature-set mpls**
	例：
	`switch(config)# install feature-set mpls`
	`switch(config)# feature-set mpls`
	`switch(config)# feature mpls segment-routing`
		MPLS パケットの解析を有効にします。これは、MPLS ラベルに基づいて MPLS パケットをフィルタリングするために必須です。

	コマンドまたはアクション	目的

ステップ 3	**mpls access list mpls-acl**
	例：
	`switch(config)# mpls access list mpls-acl`
	`switch(config-mpls-acl)# 10 permit mpls 1600 any redirect Ethernet1/15`
		着信外部 MPLS ラベルに基づくフィルタリングを使用して、mpls-access リストを構成します。
		この例では、着信ラベル 1600 と MPLS パケットが一致し、Ethernet1/15 にリダイレクトされます。

ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意）実行構成をスタートアップ構成にコピーします。

MPLS アクセス リスト構成の検証

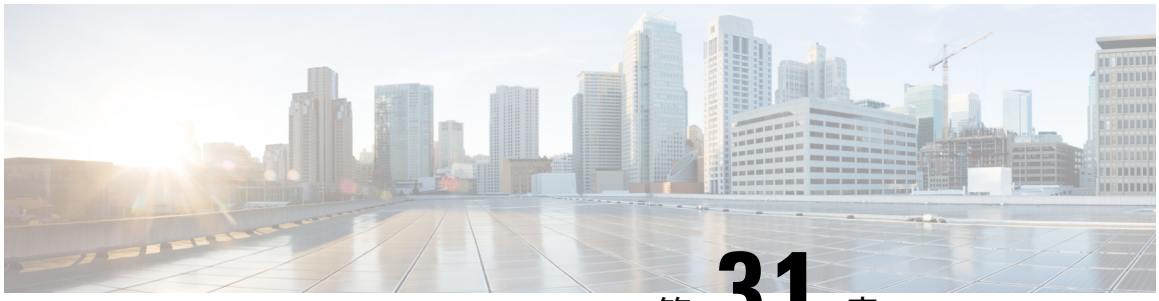
MPLS アクセス リスト構成を表示するには、 の作業を実行します。

コマンド	目的
show mpls access lists	MPLS アクセス リストの情報を表示します。

MPLS アクセス リストの構成例

次の例は、MPLS アクセス リストを構成する方法を示しています。

```
switch# configure terminal
switch(config)# install feature-set mpls
switch(config)# feature-set mpls
switch(config)# feature mpls segment-routing
switch(config)# mpls access list mpls-acl
switch(config-mpls-acl)# 10 permit mpls 1600 any redirect Ethernet1/15
switch(config)# copy running-config startup-config
```



第 31 章

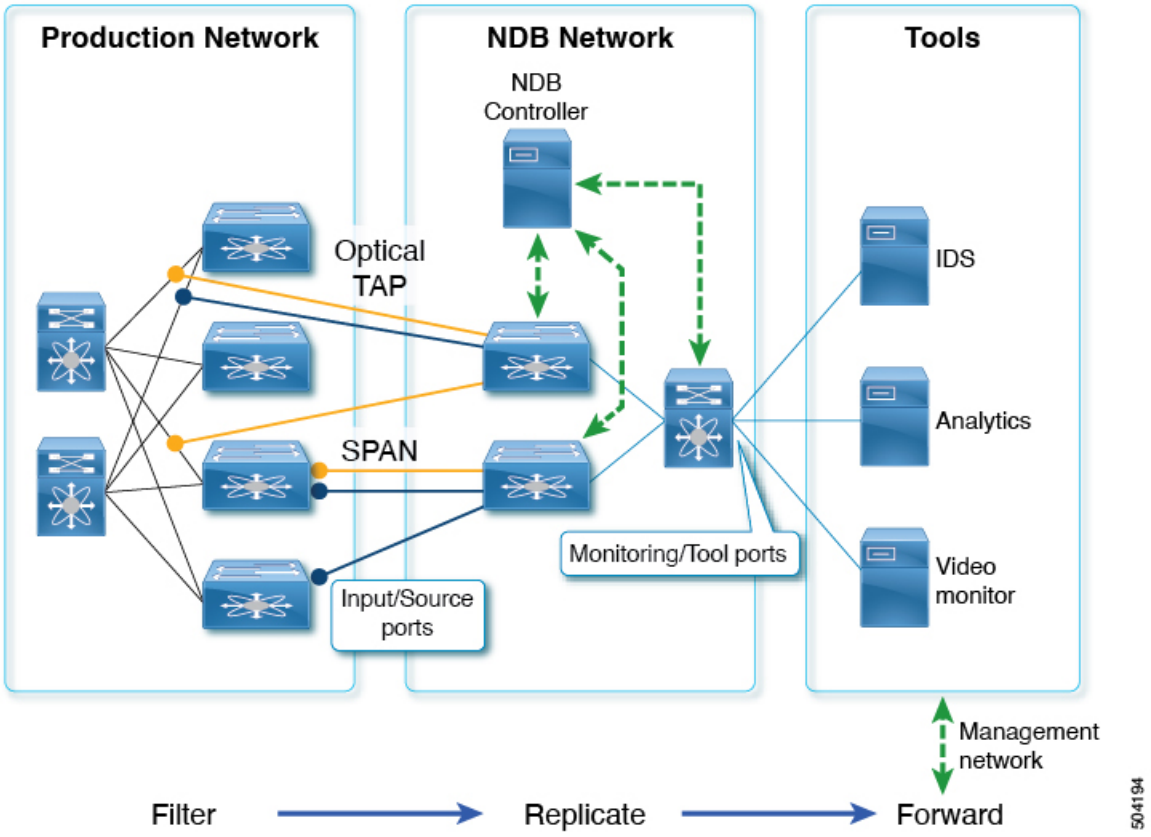
Nexus Data Broker のヘッダ ストリッピング機能の構成

- Nexus Data Broker の ヘッダー ストリッピングの紹介 （591 ページ）
- ヘッダー ストリッピングに関する注意事項と制限事項 （593 ページ）
- Nexus Data Broker の VXLAN および iVXLAN ヘッダー ストリッピング （594 ページ）
- Nexus Data Broker の ERSPAN ヘッダー ストリッピング （600 ページ）
- Nexus Data Broker の GRE ヘッダ ストリッピング （604 ページ）
- Nexus Data Broker の MPLS ヘッダ ストリッピング （606 ページ）

Nexus Data Broker の ヘッダー ストリッピングの紹介

Cisco Nexus Data Broker（NDB）は、操作が簡単なスケーラブルなパケット ブローカー ネットワーク ソリューションを構築します。Cisco Nexus Dashboard Data Broker コントローラ ソフトウェアと Cisco Nexus スイッチは、アウトオブバンドとインライン ネットワーク トラフィックの両方をモニタするための新たなソフトウェア定義アプローチを可能にします。

図 *8 : NBD* 集中型展開モデル



NDB スイッチは、パケットの監視に使用されます。パフォーマンス監視、侵入検知、コンプライアンスチェックなどには、パケット監視が必要です。

ヘッダーストリップの場合、アウトオブバンド監視が実行されます。非侵入型であり、パケットのコピーが TAP または SPAN を使用して監視されます。したがって、トラフィックに対しフィルタ処理、本番ネットワークからの複製、NDB スイッチのヘッダーの除去が行われて、監視のためにツールに転送されます。ここで言及されている入力/送信元ポートは、ヘッダーストリッピングが行われるポートです。モニタリング/ツール ポートは、ツールに直接接続するポートです。

ヘッダーを削除する理由は次のとおりです。

- 一部の監視ツールは、カプセル化されたパケットを認識しません。
- 追加のヘッダーが存在すると、分析データに間違いが生じます。
- ヘッダーを追加すると、パケット サイズが増加するため、ツールに送信されて処理されるデータ量が最適化されません。

Cisco Nexus Data Broker スイッチのパケット ヘッダーまたはラベル ストリッピング機能の利点は次のとおりです。

- マルチプロトコル ラベル スイッチング（MPLS）ラベル ストリッピング

- コピー トラフィックからの VXLAN ヘッダー ストリッピングのネイティブ サポート
- Generic Route Encapsulation（GRE）ヘッダー ストリッピングのサポート
- 出力での Q-in-Q VLAN ヘッダー ストリッピング

これらにより、NDB は、従来の VXLAN、IVXLAN、ERSPAN、GRE、および MPLS ストリッピング機能をオーバーレイ フォワーディング マネージャー（OFM）ベースのモデルに整合させることができます。OFM は、ヘッダー ストリッピング機能のためのコマンド ライン インターフェイス（CLI）をホストします。

この章は、次の内容で構成されています。

- [Nexus Data Broker の VXLAN および IVXLAN ヘッダー ストリッピング（VXLAN and IVXLAN Header Stripping for Nexus Data Broker）]
- Nexus Data Broker の ERSPAN ヘッダー ストリッピング
- Nexus Data Broker の GRE ヘッダー ストリッピング
- Nexus Data Broker の MPLS ヘッダー ストリッピング

ヘッダー ストリッピングに関する注意事項と制限事項

すべてのヘッダー ストリッピング機能に適用される注意事項と制限事項は次のとおりです。

- VxLAN、iVxLAN、GRE、MPLS などのさまざまなカプセル化タイプを持つすべてのトンネル プロファイルで、最大 500 のフロー終端インターフェイスがサポートされます。ERSPAN の場合、サポートされるフロー終端インターフェイスの最大数は 31 です。
- Cisco NX-OS リリース 10.2(3)F 以降、OFM モデルを使用した MPLS ストリッピングが、他のストリッピング機能と共存するようになります。しかし、他の種類のストリッピング機能との共存が必要ない場合、既存の MPLS ストリッピング機能が、MPLS ストリッピングを引き続きサポートします。
- 同じインターフェイスまたは異なるインターフェイス上で共存させることができます。

（注） Cisco NX-OS リリース 10.2(3)F 以降、同じインターフェイスでの ERSPAN の共存がサポートされています。ただし、これは 9300-FX2 以降のプラットフォームでのみサポートされます。

- 従来の MPLS ストリッピング機能と OFM ストリッピング機能は相互に排他的です。
- Cisco NX-OS リリース 10.2(3)F 以降、IPv6 内部パケットのトラフィックは、すべてのストリッピング機能でサポートされます。
- 以前のリリースから Cisco NX-OS リリース 10.2(3)F への中断のない ISSU を実行し、ヘッダー ストリッピング機能を実行した後、dot1q トンネル VLAN_tag が見つからないか、

vlan_id=1 に設定されている場合は、その特定のストリッピング対応インターフェイスの L2 インターフェイスからポート ACL を削除して追加します。

- インターフェイスに VLAN が設定されていないものの、switchport mode dot1q-tunnel コマンドがそのインターフェイスに設定されている場合、ストリップされたパケットはデフォルトで VLAN=1 になります。
- 互換性のない OFM コマンドが show running コマンドの出力に存在し、Cisco NX-OS リリース 10.2(3)F から以前のリリースへの中断を伴う ISSU が実行されるシナリオで、その以前の NX-OS バージョンで OFM コマンドがサポートされていなかった場合、適切なエラーが表示されます。ただし、show incompatibility コマンドは、OFM 関連の非互換性コマンドのそのようなエラーにフラグを立てません。
- OFM ベースの GRE、ERSPAN、および MPLS ストリッピング機能は、ライン カードではなく TOR でのみサポートされます。
- カプセル化（iVXLAN、VXLAN、GRE、MPLS、ERSPAN）の一部として、次の制限が一般的です。
  - 2 つ以上のトンネル プロファイルが同じカプセル化タイプを持つことはできません。
  - 機能トンネルが有効になっている場合、OFM ベースのヘッダー ストリッピング機能はサポートされません。

Nexus Data Broker の VXLAN および iVXLAN ヘッダー ストリッピング

このsubchapterでは、Nexus Data Broker（NDB）の VXLAN および IVXLAN ヘッダー ストリッピング手順について説明します。

この章は、次の項で構成されています。

Nexus Data Broker – VXLAN および iVXLAN ヘッダ ストリッピングについて

Nexus Data Broker（NDB）VXLAN および iVXLAN 終端により、スイッチは VXLAN および iVXLAN パケットの受信時にヘッダーを削除できます。

NDB スイッチは、以下のシナリオでパケットを受信します。

- スパインとリーフ間のテスト アクセス ポイント（TAP）ポートは、ACI ファブリックのファブリック リンクに配置されます。
- スイッチド ポート アナライザ（SPAN）セッションが設定されるか、TAP が VXLAN オーバーレイ ネットワークに配置されます。

ストリップ VXLAN および iVXLAN をサポートされている PID

Cisco NX-OS リリース 10.2(2)F 以降、VXLAN ストリッピング機能は Cisco Nexus 9364C および 9300-EX、9300-FX、9300-FX2、9300-FX3、9300-GX、9300-GX2、9500-EX、および 9500-FX ラインカードでサポートされています。

Cisco NX-OS リリース 10.2(2)F 以降、iVXLAN ストリッピング機能は Cisco Nexus 9364C および 9300-EX、9300-FX、9300-FX2、9300-GX、9300-GX2、9500- EX および 9500-FX ラインカードでサポートされています。

VXLAN および IVXLAN ヘッダー ストリップに関する注意事項と制限事項

- VXLAN アンダーレイが V4 の場合、VXLAN ヘッダ ストリップがサポートされます。
- PTEP / VTEP を使用せずに VXLAN および iVXLAN ヘッダを削除できる必要があります。
- VXLAN ヘッダ ストリップはポートごとに有効になります。
- VXLAN および iVXLAN ストリッピングは、次の機能が有効になっている場合はサポートされません。
  - NV オーバーレイ
  - VN-segment-vlan
  - レガシー MPLS ストリップおよび tap-aggregation
- VXLAN ストリッピングは、デフォルトの UDP 値が使用されている場合にサポートされます。
- ポートは、トンネリングされたパケットとトンネリングされていないパケットの両方を管理できる必要があります。
- レイヤ 2 スイッチポートモードトランクまたはレイヤ 2 PO インターフェイスは、VXLAN ヘッダを削除できる必要があります。
- リダイレクトインターフェイスが出力ポートまたはアナライザポートを指している場合、Tap-ACL に redirect キーワードを含む適切な ACE が含まれていることを確認します。そうでない場合、パケットは同じ入力ポートにフラッディングされます。
- OFM は、標準 ISSU および LXC-ISSUの VXLAN ストリッピング機能を有効にします。
- Cisco NX-OS リリース 10.2(1)F 以降、VXLAN および iVXLAN ストリッピング機能は、Cisco Nexus 9364C および 9300-EX、9300-FX、9300-FX2、9500-EX、および 9500-FX ラインカードでサポートされています。
- Cisco NX-OS リリース 10.2（2）F 以降、VXLAN と iVXLAN ストリッピング機能は Cisco Nexus 9300-GX と 9300-GX2 プラットフォーム スイッチでサポートされます。

- カプセル化のタイプごとに 1 つずつ、最大 4 つのトンネル プロファイルをスイッチ上に作成できます。 ただし、Cisco NX-OS リリース 10.2(3)F 以降では、最大 5 つのトンネル プロファイルがサポートされます。
- 最大 12 のリダイレクト インターフェイス (リリース 10.2(1) より前) および 32 のリダイレクト インターフェイス (リリース 10.2(1) 以降) は、TAP アグリゲーション ポリシーの単一の ACE でのみ構成できます。
- Cisco Nexus 9300-GX プラットフォーム スイッチの場合、VXLAN ストリップ後、L2 ヘッダー アドレスの送信元 MAC は VDC MAC アドレス、宛先 MAC は 000000abcdef に書き換えられます。
- Cisco NX-OS リリース 10.2(3)F 以降、VXLAN ストリップは Cisco N9K-C93180YC-FX3 と N9K-C93108TC-FX3P プラットフォーム スイッチでサポートされます。
- Cisco NX-OS リリース 10.2(4)M 以降、iVXLAN ストリッピング機能は Cisco N9K-C93180YC-FX3 と N9K-C93108TC-FX3P プラットフォーム スイッチでサポートされます。
- 次のスイッチは、上記のリリースからの VXLAN および iVXLAN ヘッダー ストリッピング機能をサポートしています。
  - N9K-C9348GC-FX3 – 10.4(1)F
  - N9K-C9332D-H2R – 10.4(1)F

VXLAN および iVXLAN ヘッダ ストリップでは、以下のステートメントが当てはまります。

- インターフェイスは、内部パケットで Q-in-Q VLAN のスラップを許可します。
- パケット CRC が正しく実行されます。
- 内部パケットは、入力ポート ACL を使用してフィルタリングできます。

Nexus Data Broker 終了の構成

次の手順は、NDB for VXLAN の終了の概要を示しています。iVXLAN ヘッダ ストリップについても同じ手順に従います。

（注） カプセル化トンネル タイプを VXLAN から iVXLAN に、またはその逆に変更するには、構成されたトンネルを no encapsulation CLI を使用して削除する必要があります。

（注） 次の CLI が、インターフェイスで VXLAN または iVXLAN のストリッピングを有効にするように構成されていることを確認します。

- 宛先
- encapsulation vxlan
- flow terminate interface add Ethernet 1/1

上記の CLI のいずれかが存在しない場合、CLI で指定されたポートでVXLAN または iVXLAN の除去は行われません。

手順の概要

1. **configure terminal**
2. **feature ofm**
3. **tunnel-profile profile-name**
4. **encapsulation vxlan**
5. **destination any**
6. **flow terminate interface ethernet 1/1**
7. **flow terminate interface remove ethernet 1/1**
8. **flow terminate interface add ethernet 1/2-5**
9. **flow terminate interface add port-channel 100-110**
10. **no flow terminate interface**
11. **feature tap-aggregation**
12. **ip access-list** <access-list name>
13. **[no] permit** protocol source destination redirect interfaces
14. **ip port access-group** <access-group name> **in**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します
ステップ 2	**feature ofm** 例： `switch (config)# feature ofm`	機能 ofm を有効にします。
ステップ 3	**tunnel-profile profile-name** 例：	スタティック VXLANトンネルを有効にします。

	コマンドまたはアクション	目的
	`switch(config)# tunnel-profile vtep_vxlan_term` `switch(config-tnl-profile)#`	

ステップ 4	**encapsulation vxlan**
	例：
	`switch(config-tnl-profile)# encapsulation vxlan`
	`switch(config-tnl-profile)#`
		トンネル プロファイルの適切なカプセル化タイプを設定します。

ステップ 5	**destination any** 例： `switch(config-tnl-profile)# destination any`	トンネルプロファイルに必要な宛先を設定します。
ステップ 6	**flow terminate interface ethernet 1/1** 例： `switch(config-tnl-profile)# flow terminate interface ethernet 1/1`	フロー条件リストに ethernet1/1 を追加します（**no flow terminate interface**コマンドは、構成されていた場合）。
ステップ 7	**flow terminate interface remove ethernet 1/1** 例： `switch(config-tnl-profile)# flow terminate interface remove ethernet 1/1`	イーサネット 1/1 ポートのみを削除します。
ステップ 8	**flow terminate interface add ethernet 1/2-5** 例： `switch(config-tnl-profile)# flow terminate interface add ethernet 1/2-5`	e1/2、e1/3、e1/4、e1/5 をフロー終端インターフェイスの既存のリストに追加します。 （注） フロー終了インターフェイスを追加する際、CLI は L2 ポート インターフェイスが存在するか、または有効になっているかを確認しません。たとえば、e1/10 は非ブレークアウト モードです。CLI では、インターフェイス e1/10/1-4 でフロー終了リストを追加できます。e1/10 がブレークアウトの場合、VXLAN ヘッダー ストリップ機能が機能します。
ステップ 9	**flow terminate interface add port-channel 100-110** 例： `switch(config-tnl-profile)# flow terminate interface add po100-110`	ポート チャネル 100-110 を古いリストに追加します。新しいリストは e1/10-11 と po100-110 です。
ステップ 10	**no flow terminate interface** 例： `switch(config-tnl-profile)# no flow terminate interface`	プロファイルからすべてのフローを削除し、インターフェイスを終了するには。

	コマンドまたはアクション	目的
ステップ 11	**feature tap-aggregation** 例： `switch(config)# feature tap-aggregation`	機能のタップ集約を有効にします。

ステップ 12	**ip access-list** <access-list name>
	例：
	`switch(config)# ip access-list test`
	`switch(config-acl)#`
		IPACL を作成し、IP アクセス リスト コンフィギュレーション モードを開始します。

ステップ 13	**[no] permit** protocol source destination redirect interfaces 例： permit ip any any redirect interface ethernet 1/1, ethernet 1/19	条件ごとにトラフィックのリダイレクトを許可する IP AC Lルールを作成します。 このコマンドの no バージョンは、ポリシーから許可ルール フォームを削除します。 （注） TAP アグリゲーション ポリシーのインターフェイスを入力するときは、それを省略しないでください。インターフェイスのリストを入力するときは、コンマで区切り、スペースを入れないでください。
ステップ 14	**ip port access-group** <access-group name> **in** 例： configure terminal interface Ethernet 1/32 ip port access-group test in	ERSPAN ストリップ/終端ポートにポート アクセス リストを適用します。

VXLAN および iVXLAN ヘッダー ストリップの構成例

次に、VXLAN および iVXLAN ヘッダー ストリッピングの例を示します。手順は iVXLAN でも同じです：

```
switch(config-tnl-profile)# show run ofm
show running-config ofm
feature ofm
tunnel-profile vxlan1
encapsulation vxlan
destination any
flow terminate interface add port-channel101
flow terminate interface add Ethernet1/1

tunnel-profile vxlan2
encapsulation ivxlan
destination any
flow terminate interface add port-channel101
flow terminate interface add Ethernet1/1
switch(config-tnl-profile)#
switch(config-tnl-profile)# show tunnel-profile
```

```
Profile : vxlan1
Encapsulation : Vxlan
State : UP
Destination : Any
Terminate Interfaces : 2
Terminate List : port-channel101 Ethernet1/1
Profile : vxlan2
Encapsulation : iVxlan
State : UP
Destination : Any
Terminate Interfaces : 2
Terminate List : port-channel101 Ethernet1/1
switch(config-tnl-profile)#
```

Nexus Data Broker の ERSPAN ヘッダー ストリッピング

この節では、Cisco Nexus プラットフォーム スイッチの ERSPAN ヘッダ ストリッピング手順について説明します。これの主な使用例は、Nexus Data Broker（NDB）スイッチです。

この章は、次の項で構成されています。

ERSPAN ヘッダ ストリッピングについて

この機能は、NX-OS スイッチまたは Nexus Data Broker（NDB）スイッチの着信 ERSPAN パケットからのインライン ERSPAN ヘッダ ストリッピングを実装します。

ERSPAN パケットが着信すると、この機能によって ERSPAN ヘッダが削除され、インラインで外部ボックスに転送されます。つまり、パケットは終端ポートに着信し、ACL 設定に基づいて、外部サーバに接続されているポートにリダイレクトされます。

この機能は、単一パスの ERSPAN ヘッダ ストリッピングと PACL リダイレクトを実行します。

ERSPAN ヘッダをストリッピングするためにサポートされる PID

Cisco NX-OS リリース 10.2(1)F 以降では、Cisco Nexus 9300-FX2、9300-FX3、9300-GX、および 9300-GX2 プラットフォーム スイッチで ERSPAN ヘッダー ストリッピングがサポートされています。ただし、この機能は TOR スイッチでのみサポートされます。

ERSPAN ヘッダ ストリッピングに関する注意事項と制限事項

- 着信ポートはレイヤ 2 ポートである必要がありますが、レイヤ 3 への接続は SVI 経由である必要があります。
- ERSPAN 接続先セッションと ERSPAN ストリッピングは共存できません。
- ポート チャネル メンバーを含む終端ポートの総数は、31 を超えることはできません。
- この機能にはモード タップアグを設定しないでください。

- すべての ERSPAN ID のトンネル プロファイルがサポートされます。特定の ERSPAN セッション ID の終了はサポートされていません。ERSPAN セッション ID を持つトラフィックは、終端ノードで終端されます。
- ノードごとに 1 つのトンネル プロファイルのみがサポートされます。
- 最大 31 のフロー終端インターフェイスが、encap タイプ：ERSPAN のトンネル プロファイルでサポートされます。
- Cisco Nexus 9300-FX2、9300-FX3、9300-GX、および 9300-GX2 プラットフォーム スイッチで ERSPAN ヘッダ ストリッピング機能がサポートされます。この機能は TOR スイッチでのみサポートされます。
- ERSPAN 削除/リダイレクトが正しく動作するように、ポートで ERSPAN 削除を有効にする必要があります。他のストリップが有効になっているポートでは、ERSPAN トラフィックを送信しないでください。
- 終端ポートのすべての着信 ERSPAN ヘッダを削除します。
- この機能は、OFM トンネル プロファイル および ACL リダイレクトが構成されている場合にのみ機能します。
- この機能は、ポート ACL がレイヤ 2 終端ポートに適用されている場合にのみ機能します。
- スイッチ上の ERSPAN カプセル化のトンネル プロファイルは 1 つだけです。
- ポート ACL を使用するには、適切な tcam をカービングする必要があります。たとえば、カービングに **tcam region ing-ifacl** を使用します。

ERSPAN ヘッダ ストリッピングの設定

次の手順では、ERSPAN ヘッダ ストリッピングの設定の概要を示します。

（注） 次の CLI がインターフェイスで ERSPAN のストリッピングを有効にするように設定されていることを確認します。

- encapsulation erspan
- erspan セッション id すべて
- flow terminate interface add e1 / 16

上記の CLI のいずれかが欠落している場合、ERSPAN の除去は、CLI で指定されたポートでは発生しません。

手順の概要

1. **configure terminal**
2. **feature ofm**

3. **tunnel-profile** <profile-name>
4. **encapsulation erspan**
5. **erspan session-id all**
6. **flow terminate interface add ethernet1/16**
7. **ip access-list** <access-list-name>
8. [**no**] **permit** protocol source destination **redirect** interfaces
9. **ip port access-group** <access-group name>**_redir in**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します
ステップ 2	**feature ofm** 例： `switch (config)# feature ofm`	機能 ofm を有効にします。

ステップ 3	**tunnel-profile** <profile-name>
	例：
	`switch(config)# tunnel-profile foo`
	`switch(config-tnl-profile)#`
		スタティック ERSPANトンネルを有効にします。

ステップ 4	**encapsulation erspan**
	例：
	`switch(config-tnl-profile)# encapsulation erspan`
	`switch(config-tnl-profile)#`
		トンネルプロファイルの適切なカプセル化タイプを設定します。

ステップ 5	**erspan session-id all** 例： `switch(config-tnl-profile)# erspan session-id all`	ERSPAN セッション ID は、関連する ERSPAN パケットが送信元スイッチで関連付けられているモニタ対象セッションを示します。
ステップ 6	**flow terminate interface add ethernet1/16** 例： `switch(config-tnl-profile)# flow terminate interface add ethernet1/16`	フロー条件リストに ethernet1/16 を追加します（フロー CLI が設定されていない場合）。

ステップ 7	**ip access-list** <access-list-name>
	例：
	`switch(config)# ip access-list test`
	`switch(config-acl)#`
		IPACL を作成し、IP アクセス リスト 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 8	[**no**] **permit** protocol source destination **redirect** interfaces 例： `permit ip any any redirect ethernet1/1,ethernet1/19`	条件ごとにトラフィックのリダイレクトを許可する IP AC Lルールを作成します。 このコマンドの いずれの バージョンも、ポリシーからのパーミッションを削除することはありません。 （注） TAP アグリゲーション ポリシーのインターフェイスを入力するときは、それを省略しないでください。インターフェイスのリストを入力するときは、コンマで区切り、スペースを入れないでください。
ステップ 9	**ip port access-group** <access-group name>**_redir in** 例： `interface e1/16 (config-if)# ip port access-group test in`	ERSPAN ストリップ/終端ポートにポート アクセス リストを適用します。

ERSPAN ヘッダ ストリッピングの設定例

次に、ERSPAN ヘッダ ストリッピングの例を示します。

```
switch(config)# feature ofm
switch(config)# tunnel-profile foo
switch(config-tnl-profile)# encapsulation erspan
switch(config-tnl-profile)# erspan session-id all
switch(config-tnl-profile)# flowterminate interface add ethernet1/16
switch(config)# ip access-list test
permit ip any any redirect ethernet1/1,ethernet1/19
interfacee1/16 (config-if)# ip port access-group test in
```

ERSPAN ヘッダ ストリッピングの設定の確認

ERSPAN ヘッダ ストリッピング設定を表示するには、次のいずれかの作業を行います。

コマンド	目的
show run ofm	トンネル プロファイルを表示します。
show run aclmgr	インターフェイス上のすべての ACL とそれらの ACL のアプリケーションを表示します。
show ip access-list acl_nam	ACL のヒット数とリダイレクトされたパケット数を表示します。
show tunnel-profile	全てのトンネル プロファイルの状態を表示します。

Nexus Data Broker の GRE ヘッダ ストリッピング

この節では、Cisco Nexus プラットフォーム スイッチの GRE ヘッダ ストリッピング手順について説明します。これの主な使用例は、Nexus Data Broker（NDB）スイッチです。

この章は、次の項で構成されています。

NDB GRE ヘッダー ストリッピングについて

この機能を使用すると、GRE カプセル化されて着信するパケットから GRE ヘッダーを取り除くことができます。GRE カプセル化パケットの内部パケットには、イーサネット ヘッダーが含まれていません。したがって、GRE ストリップの後、イーサネット ヘッダーが次のカスタム フィールドとともに内部パケットに追加されます：

1. 802.1q ヘッダーには、着信ポートで構成された VLAN が設定されます。
2. 接続先 MAC アドレスは に設定されます。 00:00:00:ab:cd:ef または 000.000.abc.def。
3. 送信元 MAC アドレスは、スイッチの VDC MAC アドレスに設定されます。

NDB GRE ヘッダー ストリッピングに関する注意事項と制限事項

- トンネルプロファイルからフローインターフェイスを削除するには、**no** の代わりに **remove** を使用します。**no** コマンドを使用すると、フロー終了リストからすべてのインターフェースが削除されます。

  次に例を示します。

  ```
  switch(config)# tunnel-profile gre_strip
  switch(config-tnl-profile)# flow terminate interface remove Ethernet 1/48
  ```

- フロー終了インターフェイスは、ESPRAN および GRE/VXLAN/IVXLAN プロファイルを共有できません。
- GRE ストリップ対応インターフェイスが ERSPAN トラフィックを受信した場合、ストリップは成功しますが、トラフィックはリダイレクト ポートに転送されません。
- 機能 OFM と機能トンネルは、同じスイッチ上に共存できません。
- Cisco Nexus 9300-EX、9300-FX、9300-FX2、9300-FX3、9300-GX、および N9K-C9332D-GX2B TOR上でサポートされている NBD GRE ヘッダー ストリッピング機能。ただし、この機能はラインカードではサポートされていません。
- **mode tap-aggregation** の構成は、GRE ヘッダー ストリッピング機能が有効になっているインターフェイスに存在しないようにする必要があります。
- トンネル カプセル化タイプの変更は許可されていません。

  ```
  QP-CF-1(config-tnl-profile)# encapsulation gre
  Error: encap-type modify not allowed, delete and add again
  ```

- 最大 500 のフロー終端インターフェイスが、encap タイプ iVXLAN/VXLAN/GRE のトンネル プロファイルでサポートされます。
- 最大 31 のフロー終端インターフェイスが、encap タイプ ERSPAN のトンネル プロファイルでサポートされます。
- フロー終了インターフェイス CLI が **add** キーワードなしで設定されている場合、それは **replace** として機能します。つまり、以前に追加されたフロー終了インターフェイスが削除され、新しいインターフェイスだけがフロー終了インターフェイスとして機能します。
- 以前の NX-OS バージョンから 10.2(3)F への中断のないアップグレード後、特定のインターフェイスの GRE ヘッダー ストリップ機能を有効にする前に、ポート ACL をすべてのインターフェイスから削除して追加する必要があります。
- dot1q トンネル伝搬を許可するには、9300-GX で **hardware acl tap-agg redirect disable-dot1q-sharing** コマンドが必要です。このコマンドを有効にした後、スイッチをリロードする必要があります。

GRE ヘッダー ストリップ機能の CLI

インターフェイスで GRE ヘッダーを有効にするために構成する CLI は次のとおりです：

```
feature ofm
tunnel-profile gre_strip
    encapsulation gre
    destination any
    flow terminate interface add Ethernet1/1-10
```

次に、トンネル プロファイルの show コマンドを示します：

```
switch# show tunnel-profile gre_strip
  Profile               : gre_strip
  Encapsulation         : GRE
  State                 : UP
  Destination           : Any
  Terminate Interfaces  : 10
  Terminate List        : Ethernet1/1 Ethernet1/2 Ethernet1/3 Ethernet1/4 Ethernet1/5
Ethernet1/6 Ethernet1/7 Ethernet1/8 Ethernet1/9 Ethernet1/10
```

出力ポートと入力ポートの構成

入力ポートの構成は次のとおりです。

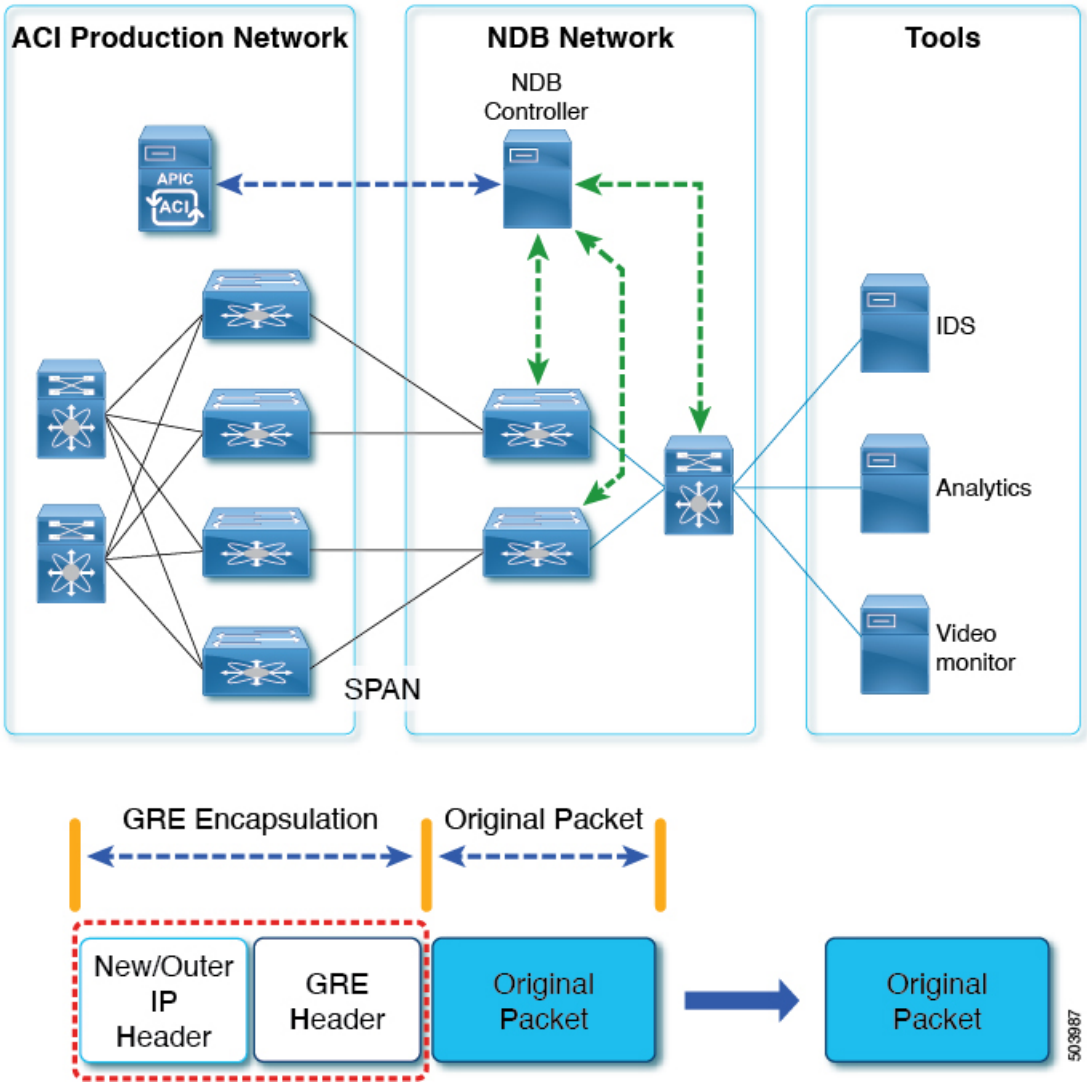
```
interface eth1/1
    switchport access vlan 101
    switchport mode dot1q-tunnel
    ip port access-group ndb_acl in <<<
    no shutdown
```

出力ポートの構成は次のとおりです。

```
interface Ethernet1/7
    switchport mode trunk
    no shutdown

IP access list ndb_acl
```

```
statistics per-entry
10 permit udp any any eq 4789 redirect Ethernet1/7
15 permit ip any any redirect Ethernet1/7
```

図 *9 : NDB GRE* ヘッダー ストリップ ソリューション



Nexus Data Broker の MPLS ヘッダ ストリッピング

このの節では、Cisco Nexus プラットフォーム スイッチの MPLS ヘッダ ストリッピング手順について説明します。これの主な使用例は、Nexus Data Broker（NDB）スイッチです。

この章は、次の項で構成されています。

NDB MPLS ヘッダー ストリッピングについて

この機能を使用すると、MPLS カプセル化されて着信したパケットから MPLS ヘッダーを取り除くことができます。MPLS ラベル ストリッピングは、IPoMPLS および EoMPLS パケット フォーマットの両方でサポートされています。MPLS ラベル ストリップの後、イーサネット ヘッダーが次のカスタム フィールドを使用して内部パケットに追加されます。

1. 着信ポートに 802.1q ヘッダー と vlan が構成されます。
2. 接続先 MAC アドレスは 00:00:00:ab:cd:ef または 000.000.abc.def に設定されます。

（注） EoMPLS ヘッダー ストリッピングの場合、これは Cisco Nexus 9300-EX、9300-FX、および 9300-GX プラットフォームにのみ適用されます。

3. 送信元 MAC アドレスは、スイッチの VDC MAC アドレスに設定されます。

（注） EoMPLS ヘッダー ストリッピングの場合、これは Cisco Nexus 9300-EX、9300-FX、および 9300-GX プラットフォームにのみ適用されます。

NDB MPLS ヘッダー ストリッピングに関する注意事項と制限事項

レガシー MPLS ヘッダー ストリッピングから OFM ベースの構成に移行する場合は、次の注意事項と制限事項が適用されます。

- レガシー MPLS ストリッピング導入は OFM ベースのストリッピングと共存できません。
- 機能 OFM と機能トンネルは、同じスイッチ上に共存できません。
- レガシー MPLS ストリッピング機能から移行するには、OFM ベースの MPLS ストリッピングを有効にする前に、次のクリーンアップが必要です。
  - インターフェース レベルでの **mode tap-aggregation** の削除
  - グローバル レベルでの**mpls strip; mpls strip dot1q** の除去
  - 構成を保存して、上記の構成でスイッチをリロードします。
- Cisco NX-OS リリース 10.2（3）F 以降、NDM MPLS ヘッダー ストリッピング機能がサポートされています。
  - IPoMPLS（パケット フォーマット）ヘッダー ストリッピングは、Cisco Nexus 9300-EX、9300-FX、9300-FX2、9300-FX3、9300-GX、および C9332D-GX2B プラットフォームでサポートされています。

- EoMPLS（パケット形式）ヘッダー ストリッピングは、Cisco Nexus 9300-EX プラットフォーム スイッチでのみサポートされています。ただし、VPLS ストリップおよび制御ワード パケット ストリップはサポートされていません。

（注） OFM MPLS ストリッピング機能は、TOR でのみサポートされます。ラインカードではサポートされていません。

- 以前の NX-OS バージョンから 10.2(3)F への中断のないアップグレード後、特定のインターフェイスの MPLS ヘッダー ストリッピング機能を有効にする前に、ポート ACL をすべてのインターフェイスから削除して追加する必要があります。
- dot1q トンネル伝搬を許可するには、Cisco Nexus 9300-GX プラットフォーム スイッチで **hardware acl tap-agg redirect disable-dot1q-sharing** コマンドが必要です。このコマンドを有効にした後、スイッチをリロードする必要があります。
- トンネル カプセル化タイプの変更は許可されていません。

```
QP-CF-1(config-tnl-profile)# encapsulation mpls
Error: encap-type modify not allowed, delete and add again
```

- ERSPAN ACL リダイレクト トンネル プロファイルが構成されておらず、インターフェイスが ERSPAN パケットを受信している場合、ERSPAN パケットは TapAgg ポリシーの ERSPAN ACL リダイレクト エントリにヒットし、削除されません。
- MPLS ヘッド ストリップが有効になっているインターフェイスでは、モード タップ アグリゲーションが存在しないようにする必要があります。
- MPLS ストリッピングは IP PACL に基づいており、ストリッピングに MAC-ACL を使用しないでください。
- MPLS ストリッピング中、オリジナル パケットの着信 VLAN は維持されません。
- ERSPAN トンネル プロファイルでは、入力インターフェイスが dot1q-tunnel からトランクモードに変換されると、出力パケットに VLAN=1 の dot1q タグが付けられます。このタグ付けは、ストリップされたパケットとリダイレクトされる通常の IP パケットの両方に対して行われます。
- MPLS ストリップ対応インターフェイスが ERSPAN トラフィックを受信すると、ストリップは成功しますが、トラフィックはリダイレクト ポートに転送されません。
- トンネルプロファイルからフローインターフェイスを削除するには、**no** の代わりに **remove** を使用します。**no** コマンドを使用すると、フロー終了リストからすべてのインターフェースが削除されます。

  次に例を示します。

```
switch(config)# tunnel-profile mpls_strip
switch(config-tnl-profile)# flow terminate interface remove Ethernet 1/48
```

- **add** キーワードなしでフロー 終端 インターフェイス コマンドを構成すると、**replace** として動作します。このことは、以前追加したフロー終了インターフェイスは削除され、新しいものだけがフロー 終端 インターフェイスとして動作することを意味します。
- 入力インターフェイスは、トランク モードまたはアクセス モードのいずれかです。どちらのモードでも、タグ付きパケットとタグなしパケットのリダイレクトが可能です。access-mode が dot1q-tunnel モードで使用される場合、ヘッダー ストリッピングの後に、access-mode で指定された方法で VLAN_tag が追加されます。
- Cisco NX-OS リリース 10.3（1）F までは、EoMPLS ヘッダー ストリッピングは Cisco Nexus 9300-EX プラットフォーム スイッチでのみサポートされていました（VPLS ストリップおよび制御ワード パケット ストリップはサポートされていませんでした）。Cisco NX-OS リリース 10.3（2）F 以降、EoMPLS ヘッダー ストリッピング機能は、Cisco Nexus 9300-FX、9300-FX2、9300-FX3、9300-GX、および 9300-GX2 ToR でもサポートされています。ラインカードではサポートされていません。次の注意事項と制限事項が適用されます。
  - EoMPLS ストリッピングは、同じまたは異なるインターフェイス上で他のすべてのヘッダー ストリッピング機能と共存できます。
  - Cisco Nexus 9300-EX、9300-FX、および 9300-GX プラットフォーム スイッチの場合、EoMPLS ヘッダー ストリップの後、L2 ヘッダー アドレスは次のように書き換えられます。送信元 MAC は VDC MAC アドレス、接続先 MAC は 000000abcdef です。
  - 疑似ワイヤー コントロール ワードはサポートされていません。
  - Cisco Nexus 9300-GX プラットフォーム スイッチでは、dot1q vlan 設定が同じでない限り、2 つの入力ポートは ACL を共有できません。そうでない場合、タグ付けは機能しません。

MPLS ヘッダー ストリップ機能のコマンド

インターフェイスで MPLS ヘッダーを有効にするには、次のコマンドを構成する必要があります：

```
feature ofm
tunnel-profile
mpls_strip encapsulation mpls destination any
flow terminate interface add Ethernet1/1-10
```

トンネル プロファイルの show コマンドは次のとおりです。

```
switch# show tunnel-profile mpls_strip
  Profile               : mpls_strip
  Encapsulation         : MPLS
  State                 : UP
  Destination           : Any
  Terminate Interfaces  : 10
  Terminate List        : Ethernet1/1 Ethernet1/2 Ethernet1/3 Ethernet1/4 Ethernet1/5
Ethernet1/6 Ethernet1/7 Ethernet1/8 Ethernet1/9 Ethernet1/10
```

出力ポートと入力ポートの構成

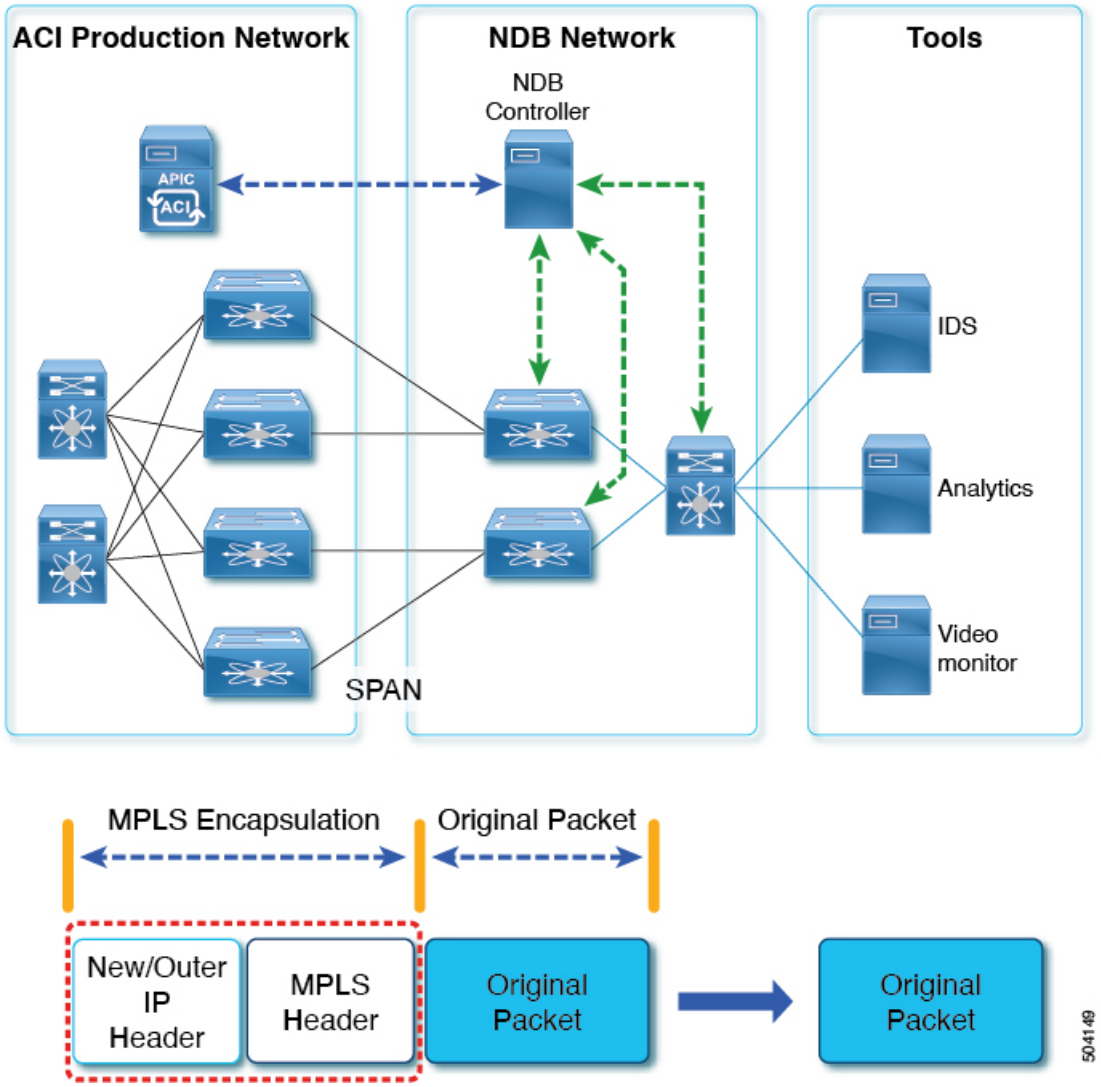
入力ポートの構成は次のとおりです。

```
interface eth1/1
    switchport access vlan 101
    switchport mode dot1q-tunnel
    ip port access-group ndb_acl in
    no shutdown
```

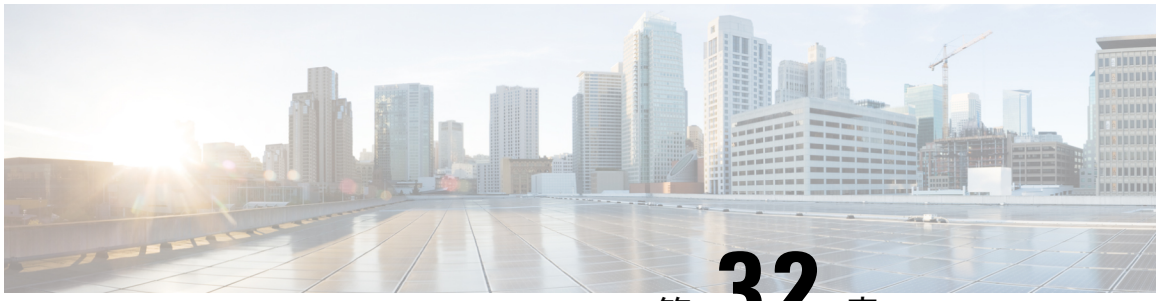
出力ポートの構成は次のとおりです。

```
interface Ethernet1/7
    switchport mode trunk
    no shutdown

IP access list ndb_acl
    statistics per-entry
    10 permit udp any any eq 4789 redirect Ethernet1/7
    15 permit ip any any redirect Ethernet1/7
```

図 *10 : NDB MPLS* ヘッダー ストリップ ソリューション



（注） MPLS などのカプセル化解除されたパケットの場合、NDB スイッチはイーサネット/VLAN ヘッダーを**オリジナルのパケット**に追加するため、出力パケットはイーサネット/VLAN を持つオリジナルのパケットになります。



第 32 章

グレースフル挿入と削除の設定

この章では、Cisco Nexus 9000 シリーズ スイッチでグレースフル挿入と削除（GIR）を設定する方法について説明します。

この章は、次の内容で構成されています。

- グレースフル挿入と削除について （613 ページ）
- GIR の注意事項と制限事項 （616 ページ）
- GIR ワークフロー （617 ページ）
- メンテナンス モード プロファイルの設定 （617 ページ）
- 通常モード プロファイルの設定 （619 ページ）
- スナップショットの作成 （621 ページ）
- スナップショットへの show コマンドの追加 （623 ページ）
- グレースフル削除のトリガー （625 ページ）
- グレースフル挿入のトリガー （628 ページ）
- メンテナンス モードの強化 （629 ページ）
- GIR 設定の確認 （631 ページ）
- GIR の設定例 （632 ページ）

グレースフル挿入と削除について

グレースフル挿入と削除を使用してスイッチを正常に取り出し、そのスイッチをネットワークから分離して、デバッグ操作やアップグレード操作を実行することができます。スイッチは、最小限のトラフィックの中断だけで、通常の転送パスから取り外されます。デバッグ操作やアップグレード操作の実行が終了したら、グレースフル挿入を使用して、そのスイッチを完全な運用（通常）モードに戻すことができます。

スイッチをメンテナンス モードにすると、すべての設定済みのレイヤ 3 コントロール プレーンがネットワークから分離されます。この状態では、直接接続されたルートは取り消されたり変更されたりしません。通常モードが復元されると、すべてのルートのアドバタイズメントが復元されます。

グレースフル削除では、すべてのプロトコルとvPCドメインが正常に停止し、スイッチはネットワークから分離されます。グレースフル挿入では、すべてのプロトコルとvPCドメインが復元されます。

次のプロトコルは、IPv4 と IPv6 両方のアドレス ファミリでサポートされます。

- Border Gateway Protocol（BGP）
- Enhanced Interior Gateway Routing Protocol（EIGRP）
- Intermediate System-to-Intermediate System（ISIS）
- Open Shortest Path First（OSPF）
- Protocol Independent Multicast（PIM）
- Routing Information Protocol（RIP）

（注） グレースフル挿入と削除の場合、PIMプロトコルはvPC環境にのみ適用できます。グレースフル削除の間、vPC転送ロールがマルチキャスト トラフィックのすべてのノースバウンド送信元に対する vPC ピアに転送されます。

プロファイル

デフォルトでは、すべての有効なプロトコルは、グレースフル削除中に分離され、グレースフル挿入時に復元されます。プロトコルは、定義済みの順序で分離および復元されます。

プロトコルを個別に分離、シャットダウン、または復元する（あるいは追加の設定を実施する）場合は、グレースフル削除またはグレースフル挿入時に適用できる設定コマンドを使用して、プロファイルを作成できます。ただし、プロトコルの順序が正しいことを確認し、すべての依存関係を考慮する必要があります。

スイッチは、次のプロファイルをサポートしています。

- メンテナンス モード プロファイル：スイッチがメンテナンス モードになったときに、グレースフル削除中に実行されるすべてのコマンドが含まれます。
- 通常モードプロファイル：スイッチが通常モードに戻ったときに、グレースフル挿入中に実行されるすべてのコマンドが含まれます。

プロファイルでは、次のコマンド（および任意の設定コマンド）がサポートされています。

（注） ルーティング プロトコル インスタンスまたはメンテナンスモード プロファイルで **shutdown** と **isolate** の両方が設定されている場合、**shutdown** コマンドが優先されます。

コマンド	説明
isolate	プロトコルをスイッチから分離し、プロトコルをメンテナンスモードにします。
no isolate	プロトコルを復元し、プロトコルを通常モードにします。
shutdown	プロトコルまたは vPC ドメインをシャットダウンします。
no shutdown	プロトコルまたは vPC ドメインを起動します。
system interface shutdown [**exclude fex-fabric**]	システム インターフェイスをシャットダウンします（管理インターフェイスを除く）。
no system interface shutdown [**exclude fex-fabric**]	システム インターフェイスを起動します。
sleep instance *instance-number seconds*	指定の秒数だけコマンドの実行を遅延させます。コマンドの複数のインスタンスを遅延できます。 *instance-number* および *seconds* 引数の範囲は、0 ～ 2177483647 です。
python instance *instance-number uri* [*python-arguments*] 例：**python instance 1 bootflash://script1.py**	Python スクリプトの呼び出しをプロファイルに設定します。コマンドの複数の呼び出しをプロファイルに追加できます。 Python 引数には最大 32 文字の英数字を入力できます。

（注） Cisco NX-OS リリース 9.3(5) 以降、**isolate** コマンドは **include-local** オプションとともに提供されます。これは、**router bgp** にのみ適用されます。

このオプションを使用すると、BGP はピアからすべてのルートを取り消します。このオプションを使用しない場合、BGP はリモートで学習したルートのみを撤回し、集約、注入、ネットワーク、再頒布などのローカルで生成されたルートは、eBGP ピアへの最大の Multi-Exit Discriminator (MED) と iBGP ピアへの最小のローカル プリファレンスで引き続きアドバタイズされます。

スナップショット

Cisco NX-OS では、スナップショットは選択した機能の実行状態をキャプチャし、永続ストレージ メディアに保存するプロセスです。

スナップショットは、グレースフル削除前とグレースフル挿入後のスイッチの状態を比較する場合に役立ちます。スナップショット プロセスは、次の 3 つの部分で構成されます。

- 事前に選択したスイッチの一部機能の状態のスナップショットを作成し、永続ストレージ メディアに保存する
- さまざまな時間間隔で取得したスナップショットを一覧にして、管理する
- スナップショットを比較して、機能間の相違を表示する

GIR の注意事項と制限事項

グレースフル挿入と置換（GIR）には、設定に関し、次の注意事項と制約事項があります。

- Cisco NX-OS リリース 9.2(1) 以降では、L2 グレースフル挿入および置換がサポートされています。通常モードからメンテナンス モードに移行すると、MCT がダウンし、垂直型トラフィックが収束します。ゼロ パケット損失はサポートされていません。次の表に、各 VPC ポートに 2 ポート メンバー、60k MACスケールを持つ 10 の vPC でのトラフィック コンバージェンスの例を示します。

表 *25:*

トリガー	ロール	垂直型トラフィック	逆垂直型トラフィック
通常からメンテナンス モードへ	プライマリ	760 ms	1320 ms
メンテナンス モードから通常モードへ	プライマリ	13155 ms	27980 ms
通常からメンテナンス モードへ	セカンダリ	300 ミリ秒	1375 ms
メンテナンス モードから通常モードへ	セカンダリ	15905 ms	23350 ms

- Cisco NX-OS リリース 9.2(1) 以降では、OSPF の分離オプションを設定すると、直接ルートとスタブルートが最大メトリック ルートとしてアドバタイズされます。その結果、1 つの vPC スイッチだけが分離されている場合、SVI ホストへの垂直型トラフィックは vPC ピアを通過します。

- 通常モードとメンテナンス モードの新しいカスタム プロファイルを作成する前に、すべての既存のカスタムプロファイルを削除してください。
- Cisco NX-OS リリース 9.3(5) 以降、**include-local** オプションが既存の **isolate** コマンドに追加されています。ただし、**include-local** オプションは **router bgp** のみに適用されます。
- Cisco NX-OS リリース 10.3(1)F 以降では、バイナリ リロード シナリオ中にシステムがメンテナンス モードに切り替わると、システムが完全に起動してシステムの準備ができたことを宣言するまで、インターフェイスはシャットダウンされます。

GIR ワークフロー

グレースフル挿入と削除（GIR）のワークフローを完了する手順は、次のとおりです。

1. （任意）メンテナンス モード プロファイルを作成します（メンテナンス モード プロファイルの設定 （617 ページ） を参照）。
2. （任意）通常モード プロファイルを作成します（通常モード プロファイルの設定 （619 ページ） を参照）。
3. グレースフル削除をトリガーする前のスナップショットを取得します（スナップショットの作成 （621 ページ） を参照）。
4. グレースフル削除をトリガーして、スイッチをメンテナンス モードにします（グレースフル削除のトリガー （625 ページ） を参照）。
5. グレースフル挿入をトリガーして、スイッチを通常モードに戻します（グレースフル挿入のトリガー （628 ページ） を参照）。
6. グレースフル挿入をトリガーした後のスナップショットを取得します（スナップショットの作成 （621 ページ） を参照）。
7. show snapshots compare コマンドを使用して、グレースフル削除と挿入の前後のスイッチの運用データを比較して、すべてが想定どおりに動作していることを確認します（GIR 設定の確認 （631 ページ） を参照）。

メンテナンス モード プロファイルの設定

グレースフル削除またはグレースフル挿入時に適用できる設定コマンドを使用して、メンテナンス モード プロファイルを作成できます。

（注） メンテナンス モードでは、リロード後に SVI が UP 状態になります。このシナリオでは、ルータ BGP で **isolate include-local** コマンドを使用するか、メンテナンス モードでインターフェイスをシャットダウン状態に維持して、接続/静的ルートのアドバタイズの影響を回避します。

手順の概要

1. **[no] configure maintenance profile maintenance-mode**
2. **end**
3. **show maintenance profile maintenance-mode**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**[no] configure maintenance profile maintenance-mode**
	例：
	`switch# configure maintenance profile maintenance-mode`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch(config-mm-profile)#`
		メンテナンス モード プロファイルのコンフィギュレーション セッションを開始します。**no** オプションは、メンテナンス プロファイルのメンテナンス モードを削除します。
		設定しているプロトコルに応じて、プロトコルを停止する適切なコマンドを入力する必要があります。サポートされるコマンドの一覧については、プロファイル （614 ページ）を参照してください。

ステップ 2	**end**
	例：
	`switch(config-mm-profile)# end`
	`switch#`
		メンテナンス モード プロファイルを終了します。

ステップ 3	**show maintenance profile maintenance-mode** 例： `switch# show maintenance profile maintenance-mode`	メンテナンス モード プロファイルの詳細を表示します。

例

次に、メンテナンス モード プロファイルを作成する例を示します。

```
switch# configure maintenance profile maintenance-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# ip pim isolate
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# shutdown
switch(config-mm-profile)# router eigrp 10
switch(config-mm-profile-router)# shutdown
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# shutdown
switch(config-mm-profile)# vpc domain 10
switch(config-mm-profile-config-vpc-domain)# shutdown
switch(config-mm-profile)# system interface shutdown
switch(config-mm-profile)# end
Exit maintenance profile mode.
switch# show maintenance profile maintenance-mode
[Maintenance Mode]
```

```
ip pim isolate
router bgp 100
  shutdown
router eigrp 10
  shutdown
  address-family ipv6 unicast
    shutdown
vpc domain 10
  shutdown
system interface shutdown
```

次に、カスタムプロファイルでスリープインスタンスを設定して、次のプロトコル変更までの遅延を追加する例を示します。

```
switch# configure maintenance profile maintenance-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# router bgp 65001
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# sleep instance 1 10
switch(config-mm-profile)# router eigrp 200
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# sleep instance 2 15
switch(config-mm-profile)# router ospf 100
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# sleep instance 3 20
switch(config-mm-profile)# router ospfv3 300
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# sleep instance 4 5
switch(config-mm-profile)# router isis 400
switch(config-mm-profile-router)# isolate
switch(config-mm-profile)#end
Exit maintenance profile mode.
switch#
```

（注） メンテナンス モード プロファイルの適用中に exec コマンドを実行するか、動的遅延を追加する必要がある場合は、**python instance** *instance-number uri* [*python-arguments*] スクリプトを使用します。

通常モード プロファイルの設定

グレースフル削除またはグレースフル挿入時に適用できる設定コマンドを使用して、通常モード プロファイルを作成できます。

手順の概要

1. ［**no**］ **configure maintenance profile normal-mode**
2. **end**
3. **show maintenance profile normal-mode**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	[**no**] **configure maintenance profile normal-mode**
	例：
	`switch# configure maintenance profile normal-mode`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch(config-mm-profile)#`
		通常モードプロファイルのコンフィギュレーションセッションを開始します。**no** バージョンは、メンテナンス プロファイルの normal-mode を削除します。
		設定しているプロトコルに応じて、プロトコルを起動する適切なコマンドを入力する必要があります。サポートされるコマンドの一覧については、プロファイル （614 ページ）を参照してください。

ステップ 2	**end**
	例：
	`switch(config-mm-profile)# end`
	`switch#`
		通常モード プロファイルを終了します。

ステップ 3	**show maintenance profile normal-mode** 例： `switch# show maintenance profile normal-mode`	通常モード プロファイルの詳細を表示します。

例

次に、メンテナンス プロファイルの通常モードを作成する例を示します。

```
switch# configure maintenance profile normal-mode
switch(config-mm-profile)# no system interface shutdown
switch(config-mm-profile)# vpc domain 10
switch(config-mm-profile-config-vpc-domain)# no shutdown
switch(config-mm-profile)# router eigrp 10
switch(config-mm-profile-router)# no shutdown
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# no shutdown
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# no shutdown
switch(config-mm-profile)# no ip pim isolate
switch(config-mm-profile)# end
Exit maintenance profile mode.
switch# show maintenance profile normal-mode
[Normal Mode]
no system interface shutdown
vpc domain 10
  no shutdown
  router eigrp 10
    no shutdown
address-family ipv6 unicast
  no shutdown
router bgp 100
  no shutdown
```

```
no ip pim isolate
```

スナップショットの作成

選択した機能の実行状態のスナップショットを作成できます。スナップショットを作成すると、事前定義された一連の **show** コマンドが実行され、出力が保存されます。

手順の概要

1. **snapshot create** *snapshot-name description*
2. **show snapshots**
3. **show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**]

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**snapshot create** *snapshot-name description*
	例：
	`switch# snapshot create snap_before_maintenance`
	`Taken before maintenance`
	`Executing 'show interface'... Done`
	`Executing 'show ip route summary vrf all'... Done`
	`Executing 'show ipv6 route summary vrf all'...`
	`Done`
	`Executing 'show bgp sessions vrf all'... Done`
	`Executing 'show ip eigrp topology summary'... Done`
	`Executing 'show ipv6 eigrp topology summary'...`
	`Done`
	`Feature 'vpc' not enabled, skipping...`
	`Executing 'show ip ospf vrf all'... Done`
	`Feature 'ospfv3' not enabled, skipping...`
	`Feature 'isis' not enabled, skipping...`
	`Feature 'rip' not enabled, skipping...`
	`Snapshot 'snap_before_maintenance' created`
		選択した機能の実行状態または運用データをキャプチャし、データを永続ストレージメディアに保存します。
		最大 64 文字の英数字のスナップショット名と最大 254 文字の英数字の説明を入力できます。
		すべてのスナップショットまたは特定のスナップショットを削除するには、**snapshot delete** {**all** | *snapshot-name*} コマンドを使用します。

ステップ 2	**show snapshots**
	例：
	`switch# show snapshots`
	`Snapshot Name Time`
	`Description`
	`---`
	`snap_before_maintenance Wed Aug 19 13:53:28 2015`
	`Taken before maintenance`
		スイッチ上に存在するスナップショットを表示します。

ステップ 3	**show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**]	2 つのスナップショットの比較を表示します。

	コマンドまたはアクション	目的
	例： `switch# show snapshots compare snap_before_maintenance snap_after_maintenance`	**summary** オプションは、2 つのスナップショット間の全体的な変更を確認するのに十分な情報のみ表示します。 **ipv4routes** および **ipv6routes** オプションは、2 つのスナップショット間の IPv4 および IPv6 ルートの変更を表示します。

例

次に、2 つのスナップショット間の変更の概要の例を示します。

```
switch# show snapshots compare snapshot1 snapshot2 summary
feature                             snapshot1       snapshot2       changed
basic summary
    # of interfaces                 16              12              *
    # of vlans                      10              4               *
    # of ipv4 routes                33              3               *
    ……

interfaces
    # of eth interfaces             3               0               *
    # of eth interfaces up          2               0               *
    # of eth interfaces down        1               0               *
    # of eth interfaces other       0               0

    # of vlan interfaces            3               1               *
    # of vlan interfaces up         3               1               *
    # of vlan interfaces down       0               0
    # of vlan interfaces other      0               1               *
    ……
```

次に、2 つのスナップショット間の IPv4 ルートの変更の例を示します。

```
switch# show snapshots compare snapshot1 snapshot2 ipv4routes
metric                          snapshot1       snapshot2       changed
# of routes                     33              3               *
# of adjacencies                10              4               *

Prefix              Changed Attribute
------              -----------------
23.0.0.0/8          not in snapshot2
10.10.10.1/32       not in snapshot2
21.1.2.3/8          adjacency index has changed from 29 (snapshot1) to 38 (snapshot2)
……

There were 28 attribute changes detected
```

スナップショットへの show コマンドの追加

スナップショットでキャプチャされる追加の **show** コマンドを指定できます。それらの **show** コマンドは、ユーザ指定のスナップショット セクションで定義されます。

手順の概要

1. **snapshot section add** *section "show-command" row-id element-key1* [*element-key2*]
2. **show snapshots sections**
3. **show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**snapshot section add** *section "show-command" row-id element-key1* [*element-key2*] 例： `switch# snapshot section add myshow "show ip interface brief" ROW_intf intf-name`	ユーザ指定のセクションをスナップショットに追加します。*section* は、**show** コマンドの出力に名前を付けるために使用されます。任意の単語を使用して、セクションに名前を付けることができます。 **show** コマンドは、引用符で囲む必要があります。**show** 以外のコマンドは拒否されます。 *row-id* 引数では、**show** コマンドの XML 出力の各行エントリのタグを指定します。*element-key1* および *element-key2* 引数では、行エントリ間を区別するために使用されるタグを指定します。ほとんどの場合、行エントリ間を区別するために指定する必要があるのは *element-key1* 引数だけです。 （注） スナップショットからユーザ指定のセクションを削除するには、**snapshot section delete** *section* コマンドを使用します。
ステップ 2	**show snapshots sections** 例： `switch# show snapshots sections`	ユーザ指定のスナップショット セクションを表示します。
ステップ 3	**show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary**	**ipv4routes**	**ipv6routes**] 例： `switch# show snapshots compare snap1 snap2`	2 つのスナップショットの比較を表示します。 **summary** オプションは、2 つのスナップショット間の全体的な変更を確認するのに十分な情報のみ表示します。

	コマンドまたはアクション	目的
		ipv4routes および **ipv6routes** オプションは、2 つのスナップショット間の IPv4 および IPv6 ルートの変更を表示します。

例

次に、**show ip interface brief** コマンドを myshow スナップショット セクションに追加する例を示します。この例では、2 つのスナップショット（snap1 および snap2）が比較され、両方のスナップショットにユーザ指定のセクションが表示されます。

```
switch# snapshot section add myshow "show ip interface brief" ROW_intf intf-name
switch# show snapshots sections
user-specified snapshot sections
--------------------------------
[myshow]
  cmd:  show ip interface brief
  row:  ROW_intf
  key1: intf-name
  key2: -

[sect2]
  cmd:  show ip ospf vrf all
  row:  ROW_ctx
  key1: instance_number
  key2: cname

switch# show snapshots compare snap1 snap2
================================================================================
Feature              Tag                  snap1                snap2
================================================================================

[bgp]
--------------------------------------------------------------------------------
.............

[interface]
--------------------------------------------------------------------------------

       [interface:mgmt0]
                     vdc_lvl_in_pkts      692310             **692317**
                     vdc_lvl_in_mcast     575281             **575287**
                     vdc_lvl_in_bcast     77209              **77210**
                     vdc_lvl_in_bytes     63293252           **63293714**
                     vdc_lvl_out_pkts     41197              **41198**
                     vdc_lvl_out_ucast    33966              **33967**
                     vdc_lvl_out_bytes    6419714            **6419788**
.............

[ospf]
--------------------------------------------------------------------------------
.............

[myshow]
--------------------------------------------------------------------------------

       [interface:Ethernet1/1]
                     state                up                 **down**
```

```
                    admin_state          up                  **down**
.............
```

（注） リロード中にシステムがメンテナンス モードに移行すると、スナップショットの差分に、state_rsn_desc が関連する値とともに表示される場合があります。対処の必要はありません。

グレースフル削除のトリガー

デバッグ操作やアップグレード操作を実行するために、スイッチのグレースフル削除をトリガーして、スイッチを取り出し、ネットワークからそのスイッチを分離できます。

始める前に

作成したメンテナンスモード プロファイルを使用するシステムの場合は、メンテナンス モード プロファイルの設定 （617 ページ） を参照してください。

手順の概要

1. **configure terminal**
2. **system mode maintenance** [**dont-generate-profile** | **timeout** *value* | **shutdown** | **on-reload reset-reason** *reason*]
3. （任意） **show system mode**
4. （任意） **copy running-config startup-config**

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル 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	**system mode maintenance** [**dont-generate-profile** | **timeout** *value* | **shutdown** | **on-reload reset-reason** *reason*]
	例：
	`switch(config)# system mode maintenance`
	`Following configuration will be applied:`
	`ip pim isolate`
		すべての有効なプロトコルをメンテナンス モードにします（**isolate** コマンドを使用）。
		次のオプションを使用できます。
		• **dont-generate-profile**：有効なプロトコルの動的な検索が回避され、メンテナンス モード プロファイルに設定されているコマンドが実行され

	コマンドまたはアクション	目的

	<pre> router bgp 65502
	     isolate
	   router ospf p1
	     isolate
	   router ospfv3 p1
	     isolate
	
	Do you want to continue (y/n)? [no] y
	
	Generating a snapshot before going into
	maintenance mode
	
	Starting to apply commands...
	
	Applying : ip pim isolate
	Applying : router bgp 65502
	Applying : isolate
	Applying : router ospf p1
	Applying : isolate
	Applying : router ospfv3 p1
	Applying : isolate
	
	Maintenance mode operation successful.</pre>
		ます。作成したメンテナンス モード プロファイルをシステムに使用させる場合は、このオプションを使用します。
		• **timeout** *value*：指定した分数の間、スイッチをメンテナンス モードのままにします。範囲は 5 ～ 65535 です。設定した時間が経過すると、スイッチは自動的に通常モードに戻ります。**no system mode maintenance timeout** コマンドは、タイマーを無効にします。
		• **shutdown**：すべてのプロトコル、vPC ドメインおよび管理インターフェイスを除くインターフェイスをシャットダウンします（**shutdown** コマンドを使用）。このオプションを指定すると中断が発生しますが、デフォルト（**isolate** コマンドを使用）の場合、中断は発生しません。
		• **on-reload reset-reason** *reason*：指定されているシステム クラッシュが発生した場合、スイッチは自動的にメンテナンス モードで起動します。**no system mode maintenance on-reload reset-reason** コマンドを使用すると、システム クラッシュ時にスイッチがメンテナンス モードで起動するのを回避できます。
		メンテナンス モードのリセット理由は次のとおりです。
		• HW_ERROR：ハードウェア エラー
		• SVC_FAILURE：重大なサービス障害
		• KERN_FAILURE：カーネル パニック
		• WDOG_TIMEOUT：ウォッチドッグ タイムアウト
		• FATAL_ERROR：致命的なエラー
		• LC_FAILURE：ライン カード障害
		• MATCH_ANY：上記のいずれかの理由
		（注）
		• リロード中、システムを復元するためにバイナリ形式の構成ファイルが使用されます。ただし、これは、すべてのシナリオで可能というわけではありません。リロードされたイメージが

	コマンドまたはアクション	目的
		新しい場合などがそうです。その場合、システムは ASCII リロードに切り替わります。 • 予期しないリロード中に、システムがメンテナンス モードに切り替わると、システムが通常モードからメンテナンス モードに完全に移行するまでは、インターフェイスはシャットダウン状態になります（理由：**mmodeBootIntfShut**）。 • リロード中に、システムが予期よりも長く完全に起動しない場合は、次のコマンドを使用してデバッグ情報を収集し、Cisco サポートに連絡してください。 • **show tech support mmode** • **show tech support system manager** • **show tech support interface manager** • **show accounting** 続行を促すプロンプトが表示されます。続行する場合は**y**、プロセスを終了する場合は**n**を入力します。

ステップ 3	（任意） **show system mode**
	例：
	`switch(config)# show system mode`
	`System Mode: Maintenance`
		現在のシステム モードを表示します。
		スイッチはメンテナンス モードになっています。スイッチに対する目的のデバッグ操作やアップグレード操作を実行できます。

ステップ 4	（任意） **copy running-config startup-config** **例**： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。このコマンドは、再起動後にメンテナンス モードを維持する場合に必要です。

例

次に、スイッチのすべてのプロトコル、vPC ドメイン、およびインターフェイスをシャットダウンする例を示します。

```
switch(config)# system mode maintenance shutdown

Following configuration will be applied:

   vpc domain 10
     shutdown
   router bgp 65502
```

```
    shutdown
  router ospf p1
    shutdown
  router ospfv3 p1
    shutdown
  system interface shutdown

Do you want to continue (y/n)? [no] y

Generating a snapshot before going into maintenance mode

Starting to apply commands...

Applying : vpc domain 10
Applying :   shutdown
Applying : router bgp 65502
Applying :   shutdown
Applying : router ospf p1
Applying :   shutdown
Applying : router ospfv3 p1
Applying :   shutdown

Maintenance mode operation successful.
```

次に、致命的なエラーが発生した場合に、スイッチを自動的にメンテナンス モードで起動する例を示します。

```
switch(config)# system mode maintenance on-reload reset-reason fatal_error
```

グレースフル挿入のトリガー

デバッグ操作やアップグレード操作の実行が終了したら、グレースフル挿入をトリガーして、すべてのプロトコルを復元できます。

始める前に

作成する通常モード プロファイルをシステムに使用させる場合は、メンテナンス モード プロファイルの設定 （617 ページ）を参照してください。

手順の概要

1. **configure terminal**
2. **no system mode maintenance** [**dont-generate-profile**]
3. （任意） **show system mode**

手順の詳細

手順

	コマンドまたはアクション	目的

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

ステップ 2	**no system mode maintenance** [**dont-generate-profile**]
	例：
	`switch(config)# no system mode maintenance dont-generate-profile`
	`Following configuration will be applied:`
	
	`no ip pim isolate`
	`router bgp 65502`
	`no isolate`
	`router ospf p1`
	`no isolate`
	`router ospfv3 p1`
	`no isolate`
	
	`Do you want to continue (y/n)? [no] y`
	
	`Starting to apply commands...`
	
	`Applying : no ip pim isolate`
	`Applying : router bgp 65502`
	`Applying : no isolate`
	`Applying : router ospf p1`
	`Applying : no isolate`
	`Applying : router ospfv3 p1`
	`Applying : no isolate`
	
	`Maintenance mode operation successful.`
	
	`Generating Current Snapshot`
		すべての有効なプロトコルを通常モードにします（**no isolate** コマンドを使用）。
		dont-generate-profile オプションを指定すると、有効なプロトコルの動的な検索が回避され、通常モードプロファイルに設定されているコマンドが実行されます。作成した通常モードプロファイルをシステムに使用させる場合は、このオプションを使用します。
		続行を促すプロンプトが表示されます。続行する場合は**y**、プロセスを終了する場合は**n**を入力します。

ステップ 3	（任意） **show system mode**
	例：
	`switch(config)# show system mode`
	`System Mode: Normal`
		現在のシステム モードを表示します。スイッチは通常モードになっていて、完全に機能しています。

メンテナンス モードの強化

リリース 7.0(3)I5(1) 以降、メンテナンス モードの次の機能拡張が Cisco Nexus 9000 シリーズ スイッチに追加されました。

- システム メンテナンス シャットダウン モードで次のメッセージが追加されます。

```
NOTE: The command system interface shutdown will shutdown all interfaces excluding
mgmt 0.
```

- CLI コマンドを入力すると、**system mode maintenance** によって孤立ポートがチェックされ、アラートが送信されます。
- 隔離モードで vPC が設定されると、次のメッセージが追加されます。

```
NOTE: If you have vPC orphan interfaces, please ensure vpc orphan-port suspend is
configured under them, before proceeding further.
```

- カスタム プロファイル設定：新しい CLI コマンド、**system mode maintenance always-use-custom-profile** がカスタム プロファイル設定に追加されます。新しい CLI コマンド、**system mode maintenance non-interactive** は Cisco Nexus 9000 シリーズ スイッチのみに追加されます。これにより、確認を行わずに、またはCLIセッションに各ステップを出力することなく、メンテナンス モードまたは通常モードへの移行を容易に行うことができます。

  ループバック インターフェイスがデバイス上の IP アドレスで設定され、このデバイスがピアデバイスにアドバタイズされると、デバイス（ループバック インターフェイスを含む）はメンテナンス モードに移行します。このような場合、 **system interface shutdown** がデバイスで設定されている場合は、カスタム メンテナンス プロファイルを使用します。

  （メンテナンスまたは通常モードで）カスタムプロファイルを作成すると、次のメッセージが表示されます。

  ```
  Please use the command system mode maintenance always-use-custom-profile
   if you want to always use the custom profile.
  ```

- after_maintenance スナップショットが取得される前に遅延が追加されました。**no system mode maintenance** コマンドは、通常モードのすべての設定が適用され、モードが通常モードに変更され、after_maintenance スナップショットを取得するためのタイマーが開始されると終了します。タイマーの期限が切れると、after_maintenance スナップショットがバックグラウンドで取得され、スナップショットが完了すると新しい警告 Syslog、MODE_SNAPSHOT_DONE が送信されます。

  CLI コマンド **no system mode maintenance** の最終出力は、after_maintenance スナップショットが生成されるタイミングを示します。

  ```
  The after_maintenance snapshot will be generated in <delay> seconds.
  After that time, please use show snapshots compare before_maintenance
  after_maintenance to check the health of the system. The timer delay
  for the after_maintenance snapshot is defaulted to 120 seconds but
  it can be changed by a new configuration command.
  ```

  after_maintenance snapshot のタイマー遅延を変更する新しい設定コマンドは、**system mode maintenance snapshot-delay <seconds>** です。この設定は、デフォルト設定の 120 秒を 0 ～ 65535 の任意の値に上書きします。これは ASCII 設定で表示されます。

  現在のスナップショット遅延の値を表示する新しい show コマンド、**show maintenance snapshot-delay** も追加されています。この新しい show コマンドでは、XML 出力がサポートされています。

- システムがメンテナンス モードであるときに表示される CLI インジケータが追加されました（例：`switch(m-mode)#`）。
- CLI リロードまたはシステム リセットによってデバイスがメンテナンス モードから通常モードおよびその逆に移行するときの SNMP トラップのサポートが追加されました。**snmp-server enable traps mmode cseMaintModeChangeNotify** トラップは、メンテナンス モードのトラップ通知の変更を有効にするために追加されました。**snmp-server enable traps mmode cseNormalModeChangeNotify** は、通常モードへのトラップ通知の変更を有効にするために追加されました。デフォルトでは両方のトラップが無効になっています。
- メンテナンス モードでは、BGP がアクティブな転送トラフィックからルータを削除できます。グレースフル シャットダウンがない場合は、ローカル設定を使用して、ローカルにインストールされたルートを iBGP ピアにアドバタイズできます。Cisco NX-OS リリース 10.4(1)F 以降では、ルートマップでカスタム分離を構成できます。このモードでは、ルートマップを構成して、アドバタイズする BGP のルートを選択できます。

GIR 設定の確認

GIR の設定を表示するには、次のいずれかの作業を行います。

コマンド	目的
show interface brief	インターフェイスの要約情報を表示します。
show maintenance on-reload reset-reasons	スイッチがメンテナンス モードで起動されることになる、リセット理由を表示します。メンテナンス モードのリセット理由の説明については、グレースフル削除のトリガー （625 ページ）を参照してください。
show maintenance profile [**maintenance-mode** | **normal-mode**]	メンテナンス モードまたは通常モードのプロファイルの詳細を表示します。
show maintenance timeout	メンテナンス モードのタイムアウト期間を表示します。この期間後、スイッチは自動的に通常モードに戻ります。
show {**running-config** | **startup-config**} **mmode** [**all**]	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のメンテナンス モードのセクションを表示します。**all** オプションには、デフォルト値が含まれます。
show snapshots	スイッチ上に存在するスナップショットを表示します。

コマンド	目的
show snapshots compare *snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**]	2 つのスナップショットの比較を表示します。 **summary** オプションは、2 つのスナップショット間の全体的な変更を確認するのに十分な情報のみ表示します。 **ipv4routes** および **ipv6routes** オプションは、2 つのスナップショット間の IPv4 および IPv6 ルートの変更を表示します。
show snapshots dump *snapshot-name*	スナップショットの取得時に生成された各ファイルの内容を表示します。
show snapshots sections	ユーザ指定のスナップショットセクションを表示します。
show system mode	現在のシステム モードを表示します。

GIR の設定例

ボーダー ゲートウェイ プロトコル（BGP）の isolate モードではダイレクト ルートが撤回されないため、BGP での **redistribute direct** の設定でトラフィックが収集されます。次に、**route-map** コマンドを使用して BGP をイネーブルにし、isolate モードでダイレクト ルートを撤回する例を示します。

ポリシー設定

Cisco NX-OS リリース 10.4(1)F 以降では、ローカルで発信され、再配布された直接ルートを照合できます。次のコマンドを使用してルートマップを構成します。

route-map の再配布の構成

```
switch(config)# route-map <route map name>
switch(config-routemap)# match route-type redistributed-direct
```

メンテナンス モードで **route-map my-rmap-deny** コマンドを使用して、タグ 200 が設定された SVI を除外します。

```
switch(config)# route-map my-rmap-deny deny 10
switch(config-route-map)# match tag 200
switch(config-route-map)# exit
switch(config)# route-map my-rmap-deny permit 20
```

メンテナンス モードで **route-map my-rmap-permit** コマンドを使用して、タグ 200 が設定された SVI を含めます。

```
switch(config)# route-map my-rmap-permit permit 10
switch(config-route-map)# match tag 200
switch(config-route-map)# exit
```

```
switch(config)# route-map my-rmap-permit permit 20
```

仮想 IP（vIP）/スイッチ仮想インターフェイス（SVI）の設定

```
switch(config)# interface loopback 200
switch(config-if)# ip address 192.0.2.100/8 tag 200
switch(config)# interface vlan 2
switch(config-if)# ip address 192.0.2.108/8 tag 200
....
switch(config)# interface vlan 3
switch(config-if)# ip address 192.0.2.102/8 tag 200
```

BGP の設定

```
switch(config)# feature bgp
switch(config)# router bgp 100
switch(config-router)# neighbor 192.0.2.100
....
```

メンテナンス モード プロファイル

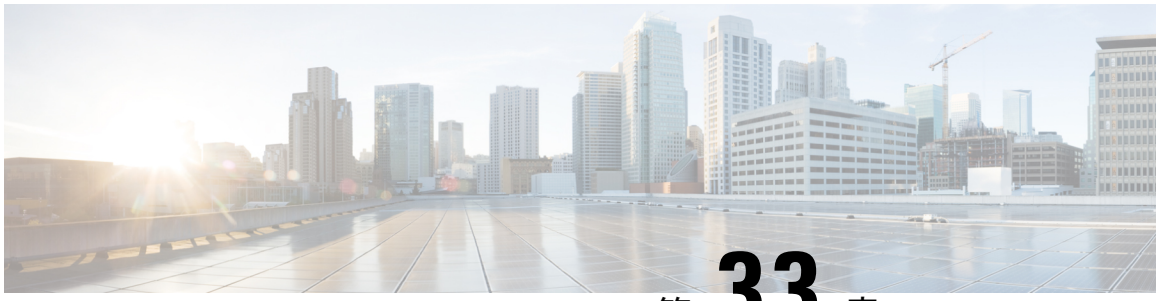
```
switch# configure maintenance profile maintenance-mode
switch(config-mm-profile)# router bgp 200
switch(config-mm-profile-router)# address-family ipv4 unicast
switch(config-mm-profile-router-af)# redistribute direct route-map my-rmap-deny
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# sleep instance 1 10
```

通常モード プロファイル

```
switch# configure maintenance profile normal-mode
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# address-family ipv4 unicast
switch(config-mm-profile-router-af)# redistribute direct route-map my-rmap-permit
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# sleep instance 1 20
```

カスタム分離モード プロファイル

```
switch# configure maintenance profile maintenance-mode
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# isolate route-map <route map name>
```



第 33 章

ソフトウェア メンテナンス アップグレードの実行

この章では、Cisco NX-OS デバイスでソフトウェア メンテナンス アップグレード（SMU）を実行する方法について説明します。

この章は、次の項で構成されています。

- SMU について （635 ページ）
- SMU の前提条件 （637 ページ）
- SMU の注意事項と制約事項 （638 ページ）
- Cisco NX-OS のソフトウェア メンテナンス アップグレードの実行 （639 ページ）
- Guest Shell Bash のソフトウェア メンテナンス アップグレードの実行 （659 ページ）
- その他の参考資料 （661 ページ）

SMU について

ソフトウェア メンテナンス アップグレード（SMU）は、特定の障害の修正を含むパッケージ ファイルです。SMU は、直近の問題に対処するために作成され、新しい機能は含まれていません。通常、SMU がデバイスの動作に大きな影響を及ぼすことはありません。SMU のバージョンは、アップグレードするパッケージのメジャー、マイナー、およびメンテナンス バージョンに同期されます。

SMU の影響は次のタイプによって異なります。

- プロセスの再起動 SMU：アクティベーション時にプロセスまたはプロセスのグループの再起動を引き起こします。
- リロード SMU：スーパーバイザおよびライン カードのパラレル リロードを引き起こします。

SMU は、メンテナンス リリースの代わりになるものではありません。重要な問題に対する迅速な解決策を提供します。SMU で修正されたすべての不具合は、今後のソフトウェア トレーンの次回のメンテナンス リリースに統合されます。SMU には、次の考慮事項もあります。

- SMU は次の目的で作成されます。
  - 回避策または修正のない重大な SIR PSIRT
  - 回避策または修正なしの重大度 1 および重大度 2 の問題
- 同じソフトウェア トレインのメンテナンス リリースで修正プログラムがすでに使用可能な場合、またはそれ以降の長期リリースですでにリリースされている場合、SMU は提供されません。メンテナンス リリースから修正を取得することをお勧めします。

（注） 修正によっては、SMUを提供できない場合があります。このような場合、唯一の選択肢は、次のメンテナンス リリースにアップグレードすることです。

デバイスを新しい機能やメンテナンス リリースにアップグレードする詳細については、『Cisco Nexus 9000 シリーズ NX-OS ソフトウェア アップグレードおよびダウングレード ガイド』を参照してください。

詳細については、『Cisco Nexus 9000 Series NX-OS Software Upgrade and Downgrade Guide』を参照してください。

（注） SMU をアクティブにすると、以前の SMU、または SMU が適用されるパッケージが自動的に非アクティブ化されることはありません。

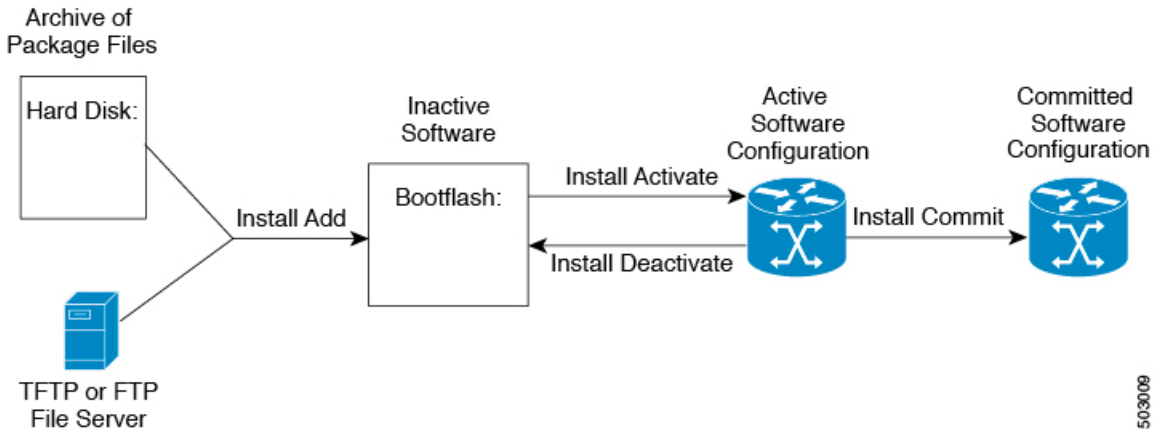
パッケージ管理

デバイスでの SMU パッケージの追加およびアクティブ化の一般的な手順は次のとおりです。

1. パッケージ ファイルをローカル ストレージ デバイスまたはファイル サーバにコピーします。
2. **install add** コマンドを使用してデバイス上でパッケージを追加します。
3. **install activate** コマンドを使用して、デバイス上でパッケージをアクティブ化します。
4. **install commit** コマンドを使用して、現在のパッケージのセットをコミットします。
5. （オプション）パッケージをアクティブでなくし、除去します。

次の図は、パッケージの管理プロセスの主要な手順について説明します。

図 *11 : SMU* パッケージを追加、アクティブ化およびコミットするプロセス



パッケージのアクティブ化と非アクティブ化の影響

SMU パッケージのアクティブ化または非アクティブ化は、システムにすぐさま影響を与える可能性があります。システムは次のように影響を受ける場合があります。

- 新しいプロセスが開始する場合があります。
- 実行しているプロセスが停止または再起動する場合があります。
- ライン カードのすべてのプロセスが再起動する場合があります。ライン カードのプロセスの再起動は、ソフト リセットと同等です。
- ライン カードがリロードする場合があります。
- ライン カードのプロセスは影響を受けない場合があります。

（注） 必要に応じて、改訂されたコンフィギュレーションおよびコンフィギュレーションの再適用によって起こる問題に対処する必要があります。

ヒント アクティブ化または非アクティブ化のプロセスが完了した後で、**show install log** コマンドを入力してプロセスの結果を表示します。

SMU の前提条件

アクティブ化または非アクティブ化するパッケージでは、これらの前提条件が満たされている必要があります。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- すべてのライン カードが取り付けられ、正常に動作していることを確認します。たとえば、ライン カードのブート中、ライン カードのアップグレード中または交換中、または自動スイッチオーバーアクティビティが予想される場合は、パッケージのアクティブ化や非アクティブ化はできません。

SMU の注意事項と制約事項

SMU に関する注意事項および制約事項は次のとおりです。

- Cisco NX-OS リリース 9.3(9) 以降、スイッチの中断を伴うリロードなしで、リロード SMU を（ND）ISSU とともに同じイメージバージョン（スイッチで現在実行中のイメージ）に適用できます。リロード SMU を適用するには、**install all nxos** <*same image*> **package** <*smu*> **non-disruptive** コマンドを使用して、ND-ISSU とリロード SMU により、同じイメージバージョンにアップグレードします。
- Cisco NX-OS リリース 10.3(4a)M 以降では、 **install all nxos** コマンドに新しいオプション、**keep-committed-smu** が追加されています。SMU がターゲット イメージと互換性がある場合には、以前にインストールした SMU を、同じイメージ バージョンにアップグレードした後も保持します。コマンドは次のように実行します。

  install all nxos <*image*> **package** <*smu*> **keep-committed-smu**

  例：

  ```
  switch# install all nxos nxos64-cs.10.4.3.M.bin package
  nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm keep-committed-smu non-disruptive
  ```

- SMU インストール用の Cisco NX-OS リリース 9.3(9) では、リロードなしのオプションがサポートされています。**no-immediate-reload** オプションは、SMU 機能をアクティブ化または非アクティブ化するために使用されます。
- パッケージによっては、他のパッケージのアクティブ化または非アクティブ化が必要です。SMU に相互に依存関係がある場合は、前の SMU をまずアクティブにしないとそれらをアクティブ化できません。
- アクティブ化する SMU パッケージは、スイッチで実行されているイメージ バージョンと互換性がある必要があります。
- パッケージの互換性が確認できた場合に限り、アクティブ化が実行されます。競合がある場合は、エラー メッセージが表示されます。
- tar バンドルを作成することで、複数の SMU パッケージをインストールできます。詳細については、高度な SMU インストール方法（655 ページ）セクションを参照してください。

- ソフトウェアパッケージをアクティブ化する間、その他の要求はすべての影響のあるノードで実行できません。これと同様のメッセージが表示されると、パッケージのアクティブ化は完了します。

```
Install operation 1 completed successfully at Thu Jan 9 01:19:24 2014
```

- 各 CLI インストール要求には要求 ID が割り当てられます。これは後でイベントを確認するのに使用できます。

Cisco NX-OS のソフトウェア メンテナンス アップグレードの実行

パッケージ インストールの準備

SMUパッケージのインストールの準備に関する情報を収集するには、複数の**show**コマンドを使用する必要があります。

始める前に

ソフトウェアの変更が必要かどうかを確認します。

使用中のシステムで新しいパッケージがサポートされていることを確認する。ソフトウェアパッケージによっては、他のパッケージまたはパッケージバージョンをアクティブにする必要があり、特定のライン カードのみをサポートするパッケージもあります。

そのリリースに関連する重要な情報についてリリース ノートを確認し、そのパッケージとデバイス設定の互換性の有無を判断する。

システムの動作が安定していて、ソフトウェアの変更に対応できることを確認します。

手順の概要

1. **show logging logfile | grep -i "System ready"**
2. **show install active**
3. **show module**
4. **show clock**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**show logging logfile | grep -i "System ready"** 例：	システムが稼働しているかどうかを表示します。このコマンドを使用して、システムで SMU パッケージをインストールする準備ができていることを確認

	コマンドまたはアクション	目的
	`switch# show logging logfile | grep -i "System ready"`	します。システムの準備が整う前にインストール コマンドを設定すると、「Install operation 11 failed because cannot lock config」エラー メッセージが表示されることがあります。
ステップ 2	**show install active** 例： `switch# show install active`	デバイス上のアクティブなソフトウェアを表示します。デバイスに追加する必要があるソフトウェアを決定するため、またインストール操作完了後にアクティブなソフトウェアのレポートと比較するために、このコマンドを使用します。
ステップ 3	**show module** 例： `switch# show module`	すべてのモジュールが安定状態であることを確認します。
ステップ 4	**show clock** 例： `switch# show clock`	システム クロックが正しいことを確認します。ソフトウェア操作は、デバイス クロックの時刻に基づいて証明書を使用します。

例

次に、システムが稼働していることを確認する例を示します。「System ready」応答は、システムがSMUパッケージのインストールの準備ができていることを示します。

```
switch# show logging logfile | grep -i "System ready"
2018 Feb 19 11:13:04 switch %ASCII-CFG-2-CONF_CONTROL: System ready
```

次に、システム全体のアクティブなパッケージを表示する例を示します。この情報を使用して、ソフトウェアの変更が必要かどうかを判断します。

```
switch# show install active
Boot Image:
        NXOS Image: bootflash:///nxos.7.0.3.I7.3.1.bin

Active Packages:

switch#
```

次に、現在のシステム クロックの設定を表示する例を示します。

```
switch# show clock
02:14:51.474 PST Wed Jan 04 2014
```

Cisco.com からの SMU パッケージ ファイルのダウンロード

SMU パッケージ ファイルをダウンロードするには、次の手順に従ってください。

手順の概要

1. Cisco.com にログインします。
2. 次の URL から Download Software ページに移動します。http://software.cisco.com/download/navigator.html
3. [製品の選択（Select a Product）] リストから、**[スイッチ（Switches）]** > **[データ センター スイッチ（Data Center Switches）]** > **[Cisco Nexus 9000 シリーズ スイッチ（Cisco Nexus 9000 Series Switches）]** > *[モデル（model）]* を選択します。
4. デバイスに適した SMU ファイルを選択し、**[ダウンロード（Download）]** をクリックします。

手順の詳細

手順

ステップ 1 Cisco.com にログインします。

ステップ 2 次の URL から Download Software ページに移動します。http://software.cisco.com/download/navigator.html

ステップ 3 [製品の選択（Select a Product）] リストから、**[スイッチ（Switches）]** > **[データ センター スイッチ（Data Center Switches）]** > **[Cisco Nexus 9000 シリーズ スイッチ（Cisco Nexus 9000 Series Switches）]** > *[モデル（model）]* を選択します。

ステップ 4 デバイスに適した SMU ファイルを選択し、**[ダウンロード（Download）]** をクリックします。

ローカルストレージデバイスまたはネットワークサーバへのパッケージ ファイルのコピー

デバイスがアクセスできるローカル ストレージ デバイスまたはネットワーク ファイル サーバに SMU パッケージ ファイルをコピーする必要があります。この作業が完了したら、パッケージをデバイスに追加しアクティブにできます。

デバイスにパッケージ ファイルを保存する必要がある場合は、ハード ディスクにファイルを保存することを推奨します。ブートデバイスは、パッケージを追加しアクティブするローカル ディスクです。デフォルトのブート デバイスは bootflash: です。

ヒント ローカル ストレージ デバイスにパッケージ ファイルをコピーする前に、**dir** コマンドを使用して、必要なパッケージ ファイルがデバイスに存在するかどうかを確認します。

SMU パッケージ ファイルがリモート TFTP、FTP、または SFTP サーバにある場合、ローカル ストレージ デバイスにファイルをコピーできます。ファイルがローカル ストレージ デバイスに置かれた後、パッケージをそのストレージデバイスからデバイスに追加しアクティブにできます。次のサーバ プロトコルがサポートされます。

- TFTP：ネットワークを介して、あるコンピュータから別のコンピュータへファイルを転送できるようにします。通常は、クライアント認証（たとえば、ユーザ名およびパスワード）を使用しません。これは FTP の簡易版です。

（注） パッケージ ファイルによっては、大きさが 32 MB を超える場合もありますが、一部のベンダーにより提供される TFTP サービスではこの大きさのファイルがサポートされていない場合があります。32 MB を超えるファイルをサポートする TFTP サーバにアクセスできない場合は、FTP を使用してファイルをダウンロードします。

- ファイル転送プロトコル：FTP は TCP/IP プロトコル スタックの一部であり、ユーザ名とパスワードが必要です。
- SSH ファイル転送プロトコル：SFTP は、セキュリティ パッケージの SSHv2 機能の一部で、セキュアなファイル転送を提供します。詳細については、『Cisco Nexus 9000 シリーズ NX-OS セキュリティ設定ガイド』を参照してください）。

（注） お使いのネットワーク サーバの場所と可用性については、システム管理者に問い合わせてください。

ファイル転送プロトコルを使用してサーバからデバイスに SMU パッケージ ファイルをコピーするには、次の表のコマンドを使用します。

表 *26 : SMU* パッケージ ファイルをデバイスにコピーするためのコマンド

コマンド	目的
copy tftp://*hostname-or-ipaddress/directory-path/filename* **bootflash:**	TFTP サーバから bootflash: にパッケージファイルをコピーします。 • *hostname-or-ipaddress*：ネットワーク ファイル サーバのホスト名または IP アドレス。 • *directory-path*：追加されるパッケージ ファイルに導くネットワーク ファイルのサーバ パス。 • *filename*：追加するパッケージ ファイルの名前。

コマンド	目的
copy ftp://*username***:***password***@***hostname-or-ipaddress***/***directory-path***/***filename*** bootflash:**	FTP サーバから bootflash: にパッケージファイルをコピーします。 • *username*：パッケージ ファイルを保存するディレクトリへのアクセス権を持つユーザのユーザ名。 • *password*：パッケージ ファイルを保存するディレクトリへのアクセス権を持つユーザのユーザ名に関連付けられたパスワード。パスワードを指定しないと、ネットワーキング デバイスは、anonymous FTP を受け入れます。 • *hostname-or-ipaddress*：ネットワーク ファイル サーバのホスト名または IP アドレス。 • *directory-path*：追加されるパッケージ ファイルに導くネットワーク ファイルのサーバ パス。指定されるディレクトリは、ユーザのホーム ディレクトリの下のディレクトリである必要があります。この例では、ダウンロードされたファイルはユーザ「john」のホーム ディレクトリ内の「images」というサブディレクトリにあります。 （注） FTP および rcp サービスの場合、*directory-path* は *username* ホーム ディレクトリの相対パスです。ディレクトリの絶対パスを指定するには、サーバ アドレスの後ろに「/」を追加する必要があります。 • *filename*：追加するパッケージ ファイルの名前。

コマンド	目的
copy sftp://*hostname-or-ipaddress*/*directory-path*/*filename* **bootflash:**	SFTP サーバから bootflash: にパッケージファイルをコピーします。 • *hostname-or-ipaddress*：ネットワーク ファイル サーバのホスト名または IP アドレス。 • *directory-path*：追加されるパッケージ ファイルに導くネットワーク ファイルのサーバ パス。 • *filename*：追加するパッケージ ファイルの名前。

SMU パッケージ ファイルをネットワーク ファイル サーバまたはローカル ストレージ デバイスに転送した後に、ファイルを追加しアクティブ化することができます。

パッケージの追加とアクティブ化

ローカル ストレージ デバイスまたはリモート TFTP、FTP、SFTP サーバに保存されている SMU パッケージ ファイルをデバイスに追加できます。

（注） アクティブ化する SMU パッケージは、現在アクティブで動作可能なソフトウェアと互換性がなければなりません。アクティブ化が試行されると、システムは自動互換性チェックを実行し、パッケージがデバイス上でアクティブなその他のソフトウェアと互換性があることを確認します。競合がある場合は、エラー メッセージが表示されます。アクティブ化が実行されるのは、すべての互換性が確認できた場合だけです。

（注） SMU をアクティブにすると、以前の SMU、または SMU が適用されるパッケージが自動的に非アクティブ化されることはありません。

始める前に

追加するすべてのパッケージがローカル ストレージ デバイスまたはネットワーク ファイル サーバにあることを確認します。

パッケージのアクティブ化の前提条件をすべて満たしていることを確認します。

ローカル ストレージ デバイスまたはネットワーク サーバへのパッケージ ファイルのコピー（641 ページ） に記載されている手順を完了します。

手順の概要

1. コンソール ポートに接続して、ログインします。
2. （任意） **dir bootflash:**
3. **install add** *filename* [**activate**]
4. （任意） **show install inactive**
5. **install activate** *filename*
6. すべてのパッケージがアクティブ化されるまで手順 5 を繰り返します。
7. （任意） **show install active**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	コンソール ポートに接続して、ログインします。	コンソール ポートにCLI管理セッションを確立します。
ステップ 2	（任意） **dir bootflash:**	追加可能なパッケージ ファイルを表示します。 （注） このプロシージャを使用して追加およびアクティブ化できるのはSMUパッケージ ファイルだけです。
ステップ 3	**install add** *filename* [**activate**] 例：	ローカル ストレージ デバイスまたはネットワーク サーバからパッケージ ソフトウェア ファイルを解凍してブートフラッシュおよびデバイスにインストールされているすべてのアクティブ スーパーバイザおよびスタンバイ スーパーバイザに追加します。 *filename* 引数は、次の形式をとることができます。 • **bootflash:***filename* • **tftp:**//*hostname-or-ipaddress*/*directory-path*/*filename* • **ftp:**//*username*:*password*@*hostname-or-ipaddress*/*directory-path*/*filename* • **usb1:***filename* • **usb2:***filename* CSCur02700 SMU パッケージを除くすべての SMU パッケージで、正常に追加された後に自動的にパッケージをアクティブにするには、オプションの **activate** キーワードを使用します。 （注） CSCur02700 SMU パッケージの場合は、ステップ 5 の **install activate** コマンドを使用してパッケージをアクティブ化します。パッケージが失敗し、リブー

	コマンドまたはアクション	目的
		トが必要になる可能性があるため、**install add** コマンドでオプションの **activate** キーワードを使用しないでください。 SMUパッケージの複数バージョンが、実行コンフィギュレーションに影響を与えずにストレージデバイスに追加できます。しかし、ライン カードに対してアクティブ化できるのは、1 つのバージョンのパッケージだけです。 （注） パッケージ名を部分的に入力してから **?** を押すと、アクティブ化に使用できるすべての候補が表示されます。候補が 1 つしかない場合に **Tab** キーを押すと、パッケージ名の残りの部分が自動入力されます。
ステップ 4	（任意） **show install inactive** 例： `switch# show install inactive`	デバイス上の非アクティブなパッケージを表示します。前述の手順で追加されたパッケージが表示に出ることを確認します。
ステップ 5	必須: **install activate** *filename* 例： 例：	デバイスに追加されたパッケージをアクティブにします。SMUパッケージは、アクティブにされるまで無効のままです。（**install add activate** コマンドを使用して、パッケージが前にアクティブにされた場合は、この手順を省略します。） ヒント アクティブ化プロセスが終了したら、**show install log** コマンドを入力してプロセスの結果を表示します。
ステップ 6	すべてのパッケージがアクティブ化されるまで手順 5 を繰り返します。	必要に応じて他のパッケージもアクティブ化します。
ステップ 7	（任意） **show install active** 例： `switch# show install active`	すべてのアクティブなパッケージを表示します。このコマンドを使用して、正しいパッケージがアクティブであるかどうかを判断します。

アクティブなパッケージ セットのコミット

SMUパッケージがデバイス上でアクティブになると、それは現在の実行コンフィギュレーションの一部になります。パッケージのアクティブ化をシステム全体のリロード間で持続させるには、デバイス上でパッケージをコミットする必要があります。

（注） 起動時に、デバイスはコミットされたパッケージセットをロードします。現在のアクティブなパッケージがコミットされる前にシステムがリロードされると、以前にコミットされたパッケージ セットが使用されます。

始める前に

パッケージセットをコミットする前に、デバイスが正常に動作し、想定どおりにパットを転送していることを検証します。

パッケージの追加とアクティブ化 （644 ページ） に記載されている手順を完了します。

手順の概要

1. **install commit** *filename*
2. （任意） **show install committed**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**install commit** *filename* 例： 	現在のパッケージのセットをコミットして、デバイスが再起動したときにこれらのパッケージが使用されるようにします。
ステップ 2	（任意） **show install committed** 例： `switch# show install committed`	コミットされたパッケージを表示します。

例

次に、デバイス上でアクティブな SMU パッケージをコミットして、次にコミットされたパッケージを確認する例を示します。

パッケージの非アクティブ化と削除

パッケージを非アクティブ化すると、そのデバイスではアクティブではなくなりますが、パッケージ ファイルはブート ディスクに残ります。パッケージ ファイルは、後で再アクティブ化できます。また、ディスクから削除もできます。

Cisco NX-OS ソフトウェアでは、選択されたパッケージ セットを前に保存されたパッケージ セットにロール バックする柔軟性も提供されます。以前のパッケージ セットの方が現在アク

ティブなパッケージ セットよりも適切であることがわかった場合は、**install deactivate** および **install commit** コマンドを使用して、以前アクティブだったパッケージ セットを再びアクティブにできます。

始める前に

別のアクティブなパッケージに必要なパッケージを非アクティブ化することはできません。パッケージを非アクティブ化しようとすると、システムがそのパッケージが他のアクティブなパッケージによって必要とされていないかを自動的にチェックします。非アクティブ化が実行されるのは、すべての互換性が確認できた場合だけです。

デバイスの実行中のソフトウェアまたはコミットされたソフトウェアの一部であるパッケージは削除できません。

手順の概要

1. コンソール ポートに接続して、ログインします。
2. **install deactivate** *filename*
3. （任意） **show install inactive**
4. （任意） **install commit**
5. （任意） **install remove** {*filename* | **inactive**}

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	コンソール ポートに接続して、ログインします。	コンソールポートにCLI管理セッションを確立します。
ステップ 2	**install deactivate** *filename* 例： 	デバイスに追加されたパッケージを非アクティブ化し、ライン カードのパッケージ機能をオフにします。 （注） パッケージを完全に非アクティブ化するには、**install deactivate** の後に **install commit** を実行する必要があります。そうしないと、パッケージはリロード後に再度アクティブ化されます。SMU をリロードするには、デバイスのリロード後に **install commit** を実行します。
ステップ 3	（任意） **show install inactive** 例： `switch# show install inactive`	デバイス上の非アクティブなパッケージを表示します。

	コマンドまたはアクション	目的
ステップ 4	（任意） **install commit** 例： `switch# install commit`	現在のパッケージのセットをコミットして、デバイスが再起動したときにこれらのパッケージが使用されるようにします。 （注） パッケージを削除できるのは、非アクティブ化操作がコミットされた場合だけです。

ステップ 5	（任意） **install remove** {*filename* | **inactive**}
	例：
	例：
	`switch# install remove inactive`
	`Proceed with removing? (y/n)? [n] y`
		非アクティブなパッケージを削除します。
		• 削除できるのは非アクティブなパッケージだけです。
		• パッケージは、デバイスのすべてのライン カードから非アクティブにされた場合にのみ削除できます。
		• パッケージの非アクティブ化はコミットする必要があります。
		• ストレージデバイスから特定の非アクティブなパッケージを削除するには、**install remove** コマンドに *filename* 引数を指定して使用します。
		• システムのすべてのノードから非アクティブなパッケージをすべて削除するには、**install remove** コマンドと **inactive** キーワードを使用します。

SMU インストールのリロードなしオプション

SMU をインストールするための no-reload オプションは次のとおりです。

方法 1：**CLI Install Add / Activate**

```
switch# show version internal build-identifier
nxos image file: bootflash:///nxos64.10.2.0.184.bin :  S184
switch# show install inactive
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Inactive Packages:

Inactive Base Packages:
        tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
        tor-2.0.0.0-10.2.0.184.lib32_n9000
        tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
switch#
switch# install add nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm
[####################] 100%
Install operation 3 completed successfully at Mon Jul 12 11:32:28 2021

switch# show install inactive
Boot Image:
```

```
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Inactive Packages:
        nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 available

Inactive Base Packages:
        tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
        tor-2.0.0.0-10.2.0.184.lib32_n9000
        tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
switch#
switch# show install pkg-info nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000
Request timedout:: Success
Name          :  nxos64.CSCaa12345-n9k_ALL
Version       :  1.0.0
Release       :  10.2.1
License       :  Cisco proprietary
Patch Type    :  reload
Requires      :  core
Provides      :  nxos64.CSCaa12345-n9k_ALL
Conflicts     :
Description   :  This is a patch for CSCaa12345-n9k_ALL
switch#
```

CLI Install Activate PATCH with no-immediate-reload option

```
switch# install activate nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 ?
  <CR>
  WORD                Package Name
  forced              Non-interactive
  no-immediate-reload  Skip immediate reload for reload type patches.
switch# install activate nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000
no-immediate-reload
[####################] 100%
Install operation 4 !!WARNING!! This patch will get activated only after
a reload of the switch. at Mon Jul 12 11:33:50 2021

switch#
switch# show install inactive
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Inactive Packages:
        nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 activate_pending_reload

Inactive Base Packages:
        tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
        tor-2.0.0.0-10.2.0.184.lib32_n9000
        tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
switch#
switch# show install patch
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

-----------------------------------------------------
nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 Inactive Committed
(activate_pending_reload)
-----------------------------------------------------
switch##

switch# reload
This command will reboot the system. (y/n)?  [n] y

CISCO SWITCH Ver7.69
Switch G2
Device detected on 0:1:2 after 0 msecs
```

```
Device detected on 0:1:1 after 0 msecs
Device detected on 0:1:0 after 0 msecs
…..
```

スイッチのリロード後、システムが準備完了状態になるのを待ちます。

```
:///nxos64.10.2.0.184.bin :  S184
switch#

switch# show logging logfile | include ready
2021 Jul 12 11:40:34 N93180-1 %ASCII-CFG-2-CONF_CONTROL: System ready

switch#

switch# show install patch
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

----------------------------------------------------
nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 Active
----------------------------------------------------

switch#

switch# show install active
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Active Packages:
        nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 active

Active Base Packages:
…..
```

CLI Install Activate PATCH with no-immediate-reload option

```
switch# install deactivate nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 ?
  <CR>
  WORD                Package Name[Note: startup configuration may get affected]
  forced              Non-interactive
  no-immediate-reload  Skip immediate reload for reload type patches.


switch# install deactivate nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000
no-immediate-reload
[####################] 100%
Install operation 5 !!WARNING!! This patch will get deactivated only after
a reload of the switch. at Mon Jul 12 11:42:24 2021

switch#

switch# show install patch
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

----------------------------------------------------
nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 Active (deactivate_pending_reload)
----------------------------------------------------
switch#
switch# show install active
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Active Packages:
        nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 active
```

```
Active Base Packages:
…..
switch# reload
WARNING: Uncommitted patches present
This command will reboot the system. (y/n)?  [n] y

CISCO SWITCH Ver7.69
Switch G2
Device detected on 0:1:2 after 0 msecs
Device detected on 0:1:1 after 0 msecs
Device detected on 0:1:0 after 0 msecs
…..
```

スイッチのリロード後、システムが準備完了状態になるのを待ちます。

```
switch# show logging logfile | include ready
2021 Jul 12 11:52:28 N93180-1 %ASCII-CFG-2-CONF_CONTROL: System ready
switch#

switch# show install patch
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

----------------------------------------------------
nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 Inactive Committed
----------------------------------------------------
switch# show install inactive
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Inactive Packages:
        nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 available

Inactive Base Packages:
        tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
        tor-2.0.0.0-10.2.0.184.lib32_n9000
        tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
switch#



switch# install remove nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000
Proceed with removing nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000? (y/n)?  [n]
 y
[####################] 100%
Install operation 6 completed successfully at Mon Jul 12 11:57:06 2021
switch# show install patch
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

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

----------------------------------------------------
switch# show install inactive
Boot Image:
        NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Inactive Packages:

Inactive Base Packages:
        tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
        tor-2.0.0.0-10.2.0.184.lib32_n9000
        tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
switch#
```

CLI install ADD ACTIVATE via bootflash：with no-immediate-reload

```
switch# install add nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm activate
?
  <CR>
  downgrade            Downgrade package
  forced               Non-interactive
  no-immediate-reload  Skip immediate reload for reload type patches.
  upgrade              Upgrade package

switch# install add nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm activate
no-immediate-reload
Adding the patch (/nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm)
[####################] 100%
Install operation 7 completed successfully at Mon Jul 12 12:03:02 2021

Activating the patch (/nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm)
[####################] 100%
Install operation 8 !!WARNING!! This patch will get activated only after
a reload of the switch. at Mon Jul 12 12:03:10 2021

switch#
```

CLI Install ADD ACTIVATE via tftp with no-immediate-reload

```
switch# install add
tftp://172.27.250.42/auto/tftp-sjc-users1/shuojiun/nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm
 vrf management activate ?
  <CR>
  downgrade            Downgrade package
  forced               Non-interactive
  no-immediate-reload  Skip immediate reload for reload type patches.
  upgrade              Upgrade package

switch# install add
tftp://172.27.250.42/auto/tftp-sjc-user1/tester/nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm
 vrf management activate no-immediate-reload
[####################] 100%
Install operation 11 !!WARNING!! This patch will get activated only after
a reload of the switch. at Mon Jul 12 12:06:49 2021

switch#
```

方法 2：VIA DME RESTアクション/実行ペイロード

（注） 次のペイロード「reloadFlag」：「noreload」では、「reloadFlag」を「noreload」として設定する必要があります。「reloadFlag」は、Action / Exec 項目では新規ではありません。

```
POST URL:
http://172.27.250.239//api/mo/sys/action.json

{
    "actionLCont": {
        "children": [
            {
                "actionLSubj": {
                    "attributes": {
                        "dn": "sys/action/lsubj-[sys]"
                    },
                    "children" : [
                        {
```

```
                                "topSystemSwpkgsInstallLTask": {
                                    "attributes": {
                                        "dn":
"sys/action/lsubj-[sys]/topSystemSwpkgsInstallLTask",
                                        "pkgAction": "add-activate",
                                        "reloadFlag": "noreload",
                                        "adminSt": "start",
                                        "url":
"nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm"
                                    }
                                }
                            }
                        ]
                    }
                }
            ]
        }
}


{
    "actionLCont": {
        "children": [
            {
                "actionLSubj": {
                    "attributes": {
                        "dn": "sys/action/lsubj-[sys]"
                    },
                    "children" : [
                        {
                            "topSystemSwpkgsInstallLTask": {
                                "attributes": {
                                    "dn":
"sys/action/lsubj-[sys]/topSystemSwpkgsInstallLTask",
                                    "pkgAction": "activate",
                                    "reloadFlag": "noreload",
                                    "adminSt": "start",
                                    "url":
"nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000"
                                }
                            }
                        }
                    ]
                }
            }
        ]
    }
}


{
    "actionLCont": {
        "children": [
            {
                "actionLSubj": {
                    "attributes": {
                        "dn": "sys/action/lsubj-[sys]"
                    },
                    "children" : [
                        {
                            "topSystemSwpkgsInstallLTask": {
                                "attributes": {
                                    "dn":
"sys/action/lsubj-[sys]/topSystemSwpkgsInstallLTask",
```

```
                                        "pkgAction": "deactivate",
                                        "reloadFlag": "noreload",
                                        "adminSt": "start",
                                        "url":
"nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000"
                                    }
                                }
                            }
                        ]
                    }
                }
            ]
        }
}
```

高度な SMU インストール方法

単一の TAR ファイルを使用した複数の SMU パッケージのインストール

複数の SMU パッケージをインストールする場合は、単一の TAR バンドルファイルを作成します。これをデータセンター内のスイッチ間で使用できます。

ソフトウェア ダウンロードセンターからダウンロードした SMU パッケージの特定のリストから TAR ファイルを生成するには、次の手順を実行します。

（注） 次の例に記載されているファイル名は説明のみを目的としており、実際のファイル名は対応するリリースによって異なります。

手順

ステップ 1 ユーザーのコンピュータまたは仮想マシンに新しいフォルダを作成します。

```
bash# mkdir nx1043
```

ステップ 2 シスコ ソフトウェア ダウンロードセンター ポータルから必要な SMU パッケージをダウンロードし、SMU パッケージを新しいフォルダにコピーします。

```
bash# cp nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm  nx1043/
bash# cp nxos64-cs.CSCxy22222-1.0.0-10.4.3.lib32_64_n9000.rpm  nx1043/
```

ステップ 3 tar バンドル ファイルを作成します。

```
bash# cd nx1043
bash# tar cf nxos64-cs.10.4.3.smu.bundle.tar *.rpm
```

ステップ 4 既存の **install add** *filename* **activate** コマンドを使用して、TAR バンドルから SMU パッケージをインストールします。

```
switch# install add nxos64-cs.10.4.3.smu.bundle.tar activate
```

新しい NX-OS ソフトウェア イメージのインストールの一部としての SMU パッケージのインストール

Cisco Nexus スイッチでは、 **install all** コマンドを使用して NX-OS ソフトウェア イメージを新しいバージョンにアップグレードできます。このコマンドは、NX-OS スイッチ ソフトウェア イメージとは別に SMU パッケージを含めるように拡張されました。これにより、ソフトウェア イメージと SMU パッケージの両方のインストール プロセス中に必要なリロードの回数が削減されるため、ソフトウェア メンテナンス操作にメリットがあります。

install all コマンドは、次のいずれかを含む単一の .tar バンドルファイルで開始できます。

- 1 つの NX-OS ソフトウェア イメージと 1 つの SMU .rpm ファイル
- 1 つの NX-OS ソフトウェア イメージと複数の SMU .rpm ファイルの tar バンドル

（注） 子 tar バンドルには、SMU .rpm ファイルと SMU .rpm ファイルの別の tar バンドルを混在させることはできません。

install all コマンドが 1 つ以上の SMU .rpm ファイルで開始されると、スイッチはアップグレード後に SMU ファイルを自動的にコミットします。

ブートアップ中にスイッチがリロードされると、SMU は適用されず、非アクティブ状態のままになります。SMU は、 **install all** または **install activate** コマンドを使用してインストールできます。

次のセクションでは、SMU パッケージが **install all** コマンドに含まれている場合にサポートされるすべてのシナリオについて説明します。

（注） 次の例に記載されているファイル名は説明のみを目的としており、実際のファイル名は対応するリリースによって異なります。

例 1：このシナリオでは、新しいソフトウェアイメージと単一の SMU パッケージが使用されます。

```
switch# install all nxos nxos64-cs.10.4.3.M.bin package
nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm
```

例 2：このシナリオでは、上記の TAR ファイル方式に従って一連の SMU パッケージが TAR バンドルとして作成され、NX-OS ソフトウェアイメージとともにインストールされます。

```
switch# install all nxos nxos64-cs.10.4.3.M.bin package nxos64-cs.10.4.3.smu.bundle.tar
```

例 3：このシナリオでは、1 つの SMU パッケージと NX-OS ソフトウェア イメージを 1 つの tar ファイルにバンドルし、 **install all** コマンドを使用してインストールできます。

```
switch# install all nxos nxos64-cs.10.4.3.M.SMU.plus.IMAGE.tar
```

1. Cisco ダウンロード センターから SMU パッケージをダウンロードします。例：nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm

2. **nxos64-cs.10.4.3.M.bin** をダウンロードし、同じフォルダに配置します。
3. NX-OS イメージと SMU パッケージで構成される tar バンドル **nxos64-cs.10.4.3.M.SMU.plus.IMAGE.tar** を作成します。

```
bash# tar cf nxos64-cs.10.4.3.M.SMU.plus.IMAGE.tar  nxos64-cs.10.4.3.M.bin
nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm
```

例 4：NX-OSイメージとともに複数のSMUパッケージをインストールする必要がある場合は、単一の TAR ファイルを使用した複数の SMU パッケージのインストール （655 ページ） セクションで説明されているように、SMU パッケージを最初に SMU tar バンドル ファイルに組み込む必要があります。その後、この SMU tar バンドルをさらに NX-OS イメージと一緒にバンドルし、単一の tar ファイルを **install all** コマンドで使用できます。

```
Switch# install all nxos nxos64-cs.10.4.3.M.SMU.BUNDLE.plus.IMAGE.tar
```

1. 単一の TAR ファイルを使用した複数の SMU パッケージのインストール （655 ページ） セクションの説明に従って、SMU パッケージのリストを使用して SMU tar バンドル イメージを作成します。

```
bash# mkdir nx1043
bash# cp nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm  nx1043/
bash# cp nxos64-cs.CSCxy22222-1.0.0-10.4.3.lib32_64_n9000.rpm  nx1043/
bash# cd nx1043
bash# tar cf nxos64-cs.10.4.3.smu.bundle.tar *.rpm
```

2. **nxos64-cs.10.4.3.M.bin** をダウンロードし、同じフォルダに配置します。
3. 別の tar バンドル **nxos64-cs.10.4.3.M.SMU.BUNDLE.plus.IMAGE.tar** を作成します。

```
bash# tar cf nxos64-cs.10.4.3.M.SMU.BUNDLE.plus.IMAGE.tar  nxos64-cs.10.4.3.M.bin
nxos64-cs.10.4.3.smu.bundle.tar
```

機能 RPM のダウングレード

インストールされている機能RPMを基本機能RPMにダウングレードするには、この手順を実行します。

手順の概要

1. （任意） **show install packages**
2. **run bash**
3. **cd /rpms**
4. **ls ****feature****
5. **cp** *filename* **/bootflash**
6. **exit**
7. **install add bootflash:***filename* **activate downgrade**
8. （任意） **show install packages | i** *feature*

手順の詳細

手順

	コマンドまたはアクション	目的

ステップ 1	（任意） **show install packages**
	例：
	`switch# show install packages`
	`ntp.lib32_n9000 1.0.1-7.0.3.I2.2e`
	`installed`
		デバイス上の機能 RPM パッケージを表示します。

ステップ 2	必須: **run bash**
	例：
	`switch# run bash`
	`bash-4.2$`
		Bash をロードします。

ステップ 3	必須: **cd /rpms** 例： `bash-4.2$ cd /rpms`	Bash の RPM フォルダへの変更。

ステップ 4	必須: **ls ****feature****
	例：
	`bash-4.2$ ls *ntp*`
	`ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm`
		指定された機能の RPM を一覧表示します。

ステップ 5	必須: **cp** *filename* **/bootflash** 例： `bash-4.2$ cp ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm /bootflash`	基本機能RPMをブートフラッシュにコピーします。
ステップ 6	必須: **exit** 例： `bash-4.2$ exit`	Bash を終了します。

ステップ 7	必須: **install add bootflash:***filename* **activate downgrade**
	例：
	`switch# install add bootflash:ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm activate downgrade`
	`Adding the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) [############## ] 60%`
	`Adding the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) [####################] 100%`
	`Install operation 11 completed successfully at Thu Sep 8 15:35:35 2015`
		機能 RPM をダウングレードします。
		（注）
		デバイスのリロードを要求されたら、**Y** を入力します。リロードは、NTP および SNMP 機能 RPM をダウングレードする場合にのみ必要です。

	コマンドまたはアクション	目的
	Activating the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) This install operation requires system reload. Do you wish to continue (y/n)?: [n] y [217.975959] [1473348971] writing reset reason 132, System reset due to reload patch(es) activation [217.991166] [1473348971]\ufffd\ufffd CISCO SWITCH Ver7.51 Device detected on 0:6:0 after 0 msecs Device detected on 0:1:1 after 0 msecs Device detected on 0:1:0 after 0 msecs MCFrequency 1333Mhz Relocated to memory	
ステップ 8	（任意） **show install packages | i** *feature* 例： switch# show install packages | i ntp ntp.lib32_n9000 1.0.0-7.0.3.I2.2e installed	デバイス上の基本機能 RPM を表示します。

インストール ログ情報の表示

インストールログは、インストール動作の履歴についての情報を提供します。インストール動作が実行されるたびに、その動作に対して番号が割り当てられます。

- **show install log** コマンドを使用して、インストール動作の成功および失敗の両方について情報を表示します。
- 引数を指定しない **show install log** コマンドを使用して、すべてのインストール動作のサマリーを表示します。ある動作に固有の情報を表示するには、*request-id* 引数を指定します。ファイルの変更、リロードできなかったノード、その他プロセスに影響する操作など、特定の操作の詳細を表示するには、**detail** キーワードを使用します。

次に、すべてのインストール要求の情報を表示する例を示します。

次に、ノードやプロセスへの影響を含む追加情報を表示する例を示します。

次に、SMU パッケージが起動した後、スイッチがリロードされる前の出力の例を示します。

Guest Shell Bash のソフトウェア メンテナンス アップグレードの実行

Guest Shell の Bash のソフトウェア メンテナンス アップグレードを実行します。

手順の概要

1. Cisco.com から Guest Shell Bash の SMU パッケージ ファイルをダウンロードします。

2. SMU パッケージファイルをスイッチの bootflash: にコピーします。
3. **guestshell**
4. **sudo rpm -Uvh** /bootflash/*filename*
5. **rpm -qa | grep bash**
6. **guestshell sync**

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	Cisco.com から Guest Shell Bash の SMU パッケージファイルをダウンロードします。	Cisco.com からパッケージ ファイルを取得します。この説明については、Cisco.com からの SMU パッケージファイルのダウンロード（640ページ） を参照してください。
ステップ 2	SMU パッケージファイルをスイッチの bootflash: にコピーします。	パッケージファイルをデバイスにコピーします。この説明については、ローカル ストレージ デバイスまたはネットワーク サーバへのパッケージ ファイルのコピー（641 ページ） を参照してください。

ステップ 3	**guestshell**
	例：
	`switch# guestshell`
	`guestshell:~$`
		Guest Shell にアクセスします。

ステップ 4	**sudo rpm -Uvh** /bootflash/*filename*
	例：
	`guestshell:~$ sudo rpm -Uvh /bootflash/bash-4.2-r8.x86_64.rpm`
	`Preparing...`
	`########################################### [100%]`
	`1:bash`
	`########################################### [100%]`
	`update-alternatives: Linking //bin/sh to /bin/bash`
		Guest Shell の既存の Bash ファイルをアップグレードします。

ステップ 5	**rpm -qa | grep bash**
	例：
	`guestshell:~$ rpm -qa | grep bash`
	`bash-4.2-r8.x86_64`
		Bash ファイルの新しいバージョンが正常にインストールされたことを確認します。

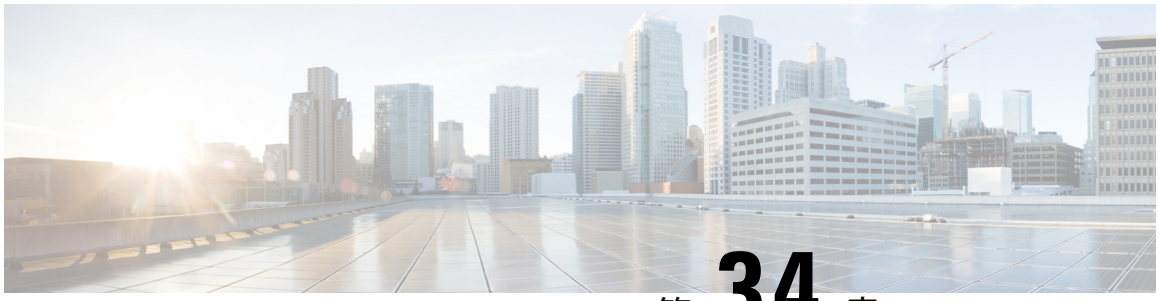
ステップ 6	**guestshell sync**
	例：
	`switch# guestshell sync`
	`Access to the guest shell will be temporarily disabled while it synchronizes contents to`
		デュアルスーパーバイザ システムでは、スイッチオーバーを実行する前に、スタンバイ スーパーバイザに対して rootfs を Bash SMU バージョンと同期します。このコマンドを実行しない場合は、スーパー

	コマンドまたはアクション	目的
	standby. Are you sure you want to continue? (y/n) [n] y dt-n9k3-1# 2014 Oct 7 05:00:01 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-INSTALL_STATE: Deactivating virtual service 'guestshell+' dt-n9k3-1# 2014 Oct 7 05:00:06 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Successfully deactivated virtual service 'guestshell+' 2014 Oct 7 05:00:12 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Successfully deactivated virtual service 'guestshell+' ; Starting sync to standby sup 2014 Oct 7 05:00:32 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-MOVE_STATE: Successfully synced virtual service 'guestshell+' ; Activating 2014 Oct 7 05:00:32 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Activating virtual service 'guestshell+' 2014 Oct 7 05:00:56 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Successfully activated virtual service 'guestshell+'	バイザのスイッチオーバー後にこの手順を繰り返す必要があります。 **（注）** 新しい Bash ファイルは、Guest Shell のリブート後または Guest Shell の無効化+有効化後に保持されます。ただし、Guest Shell の破棄と有効化の後に、Guest Shell Bash SMU パッケージ ファイルを再インストールする必要があります。

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
ソフトウェア アップグレード	『*Cisco Nexus 9000* シリーズ *NX-OS* ソフトウェア アップグレードおよびダウングレード ガイド』



第 34 章

コンフィギュレーションの置換の実行

この章は、次の項で構成されています。

- コンフィギュレーションの置換とコミットタイムアウトについて （663 ページ）
- 概要 （664 ページ）
- コンフィギュレーションの置換に関する注意事項と制限事項 （666 ページ）
- コンフィギュレーションの置換の推奨ワークフロー （668 ページ）
- コンフィギュレーションの置換の実行 （670 ページ）
- コンフィギュレーションの置換の確認 （673 ページ）
- コンフィギュレーションの置換の例 （673 ページ）

コンフィギュレーションの置換とコミットタイムアウトについて

コンフィギュレーションの置換機能を使用すると、デバイスをリロードすることなく Cisco Nexus スイッチの実行コンフィギュレーションをユーザ指定のコンフィギュレーションに置換できます。コンフィギュレーション自体でリロードが必要な場合にのみ、デバイスのリロードが必要になることがあります。ユーザが提供する実行コンフィギュレーションファイルは、実行ファイルのコピーを使用して取得する必要があります。**copy file: to running** と異なり、コンフィギュレーションの置換機能はマージ操作ではありません。この機能では、実行コンフィギュレーション全体が、ユーザによって提供される新しいコンフィギュレーションに置換されます。コンフィギュレーションの置換に障害がある場合は、元のコンフィギュレーションがスイッチで復元されます。Cisco NX-OS リリース 9.3(1) から、**best-effort** オプションが導入されました。このオプションを使用すると、コマンドでエラーが発生した場合でも、設定の置換によって完全なパッチが実行され、元の設定はスイッチに復元されません。

コミットタイムアウト機能を使用すると、コンフィギュレーションの置換操作の実行に成功した後に以前のコンフィギュレーションにロールバックすることができます。コミットタイマーの期限が切れると、ロールバック操作は自動的に開始されます。

（注）
- Cisco NX-OS デバイスで受信済みの有効な実行コンフィギュレーションを提供する必要があります。部分コンフィギュレーションにすることはできません。

概要

設定置換機能には、次の操作手順があります。

- コンフィギュレーションの置換では、Cisco Nexus スイッチの現在の実行コンフィギュレーションとユーザ指定のコンフィギュレーションとの間の違いをインテリジェントに計算し、2 ファイルの差異のパッチ ファイルを生成します。コンフィギュレーション コマンドのセットが含まれているこのパッチ ファイルは表示できます。
- コンフィギュレーションの置換では、実行中のコマンドと同様にパッチ ファイルのコンフィギュレーション コマンドが適用されます。
- コンフィギュレーションは、次の状況下で以前の実行コンフィギュレーションにロールバックまたは復元されます。
  - パッチ ファイルが適用された後、コンフィギュレーションに不一致がある場合。
  - コミット タイムアウトを使用してコンフィギュレーション操作を実行し、コミット タイマーが期限切れになった場合。
- ベストエフォート オプションが使用されている場合、設定は以前の実行コンフィギュレーションにロールバックされず、復元もされません。このオプションを使用すると、コマンドでエラーが発生した場合でも、設定の置換によって完全なパッチが実行され、以前の設定にロールバックされません。
- **show config-replace log exec** コマンドを使用すると、エラーが発生したコンフィギュレーションそのものを表示できます。
- スイッチを元のコンフィギュレーションに復元するときにエラーが発生しても復元操作は中断されません。復元操作は、残りのコンフィギュレーションを続行します。復元操作中にエラーが発生したコマンドを一覧表示するには、**show config-replace log exec** コマンドを使用します。
- タイマーの期限が切れる前に **configure replace commit** コマンドを入力した場合、コミット タイマーは停止し、コンフィギュレーションの置換機能によって適用されているユーザ指定のコンフィギュレーションでスイッチが稼働します。
- コミット タイマーの期限が切れると、以前のコンフィギュレーションへのロールバックは自動的に開始されます。
- Cisco NX-OS リリース 9.3(1) では、セマンティック検証のサポートが設定の置換に追加されました。このセマンティック検証は、設定置換の事前チェックの一部として実行されます。パッチは、セマンティック検証が成功した場合にのみ適用されます。パッチファイル

を適用すると、コンフィギュレーションの置換によって検証プロセスがトリガーされます。コンフィギュレーションの置換は、検証プロセスで、実行コンフィギュレーションとユーザー構成ファイルを比較します。不一致がある場合、デバイスは元のコンフィギュレーションに復元されます。

コンフィギュレーションの置換と実行コンフィギュレーションへのファイルのコピーとの違いは、次のとおりです。

コンフィギュレーションの置換	ファイルのコピー
configure replace <*target-url*> 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、置換ファイルには存在しないコマンドは削除されます。また、現在の実行コンフィギュレーションに追加する必要があるコマンドも追加されます。	**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。
configure replace <*target-url*> コマンドの交換ファイルには、完全な Cisco NX-OS コンフィギュレーション ファイルを使用する必要があります。	**copy** <*source-url*> **running-config** コマンドのコピー元ファイルとして、部分コンフィギュレーション ファイルを使用できます。

コンフィギュレーションの置換の利点

コンフィギュレーションの置換の利点は次のとおりです。

- スイッチをリロードしたり、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 ファイルをユーザ指定のコンフィギュレーション ファイルと置換できます。その結果、システムのダウンタイムが減少します。
- 保存済みの Cisco NX-OS コンフィギュレーションの状態に戻すことができます。
- 追加や削除が必要なコマンドだけが影響を受ける場合、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簡素化されます。その他のサービスおよび変更されていないコンフィギュレーションには影響しません。
- コミットタイムアウト機能を設定すると、コンフィギュレーションの置換操作が成功したときでも以前のコンフィギュレーションにロールバックすることができます。

コンフィギュレーションの置換に関する注意事項と制限事項

コンフィギュレーションの置換機能には、コンフィギュレーションに関する次のガイドラインと制限事項があります。

- 設定置換機能は、Cisco Nexus 3000 シリーズおよび Cisco Nexus 9000 シリーズ スイッチでサポートされています。
- コンフィギュレーションの置換、チェックポイント、ロールバック操作、または実行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のコピーを同時に実行できるのは、1 ユーザだけです。複数の Telnet、SSH または NX-API セッション経由の操作などのパラレル操作はサポートされていません。複数のコンフィギュレーションの置換またはロールバック要求はシリアル化され、たとえば、最初の要求の完了後にのみ、2 番目の要求の処理が開始されます。
- コミットタイマーの実行中に別のコンフィギュレーションの置換操作を開始することはできません。**configure replace commit** コマンドを使用してタイマーを停止するか、またはコミットタイマーの期限が切れるまで待機してから別のコンフィギュレーションの置換操作を開始する必要があります。
- **system default switchport shutdown** または **no system default switchport shutdown** を **configure replace bootflash:target_config_file** コマンドとともに使用する場合、ユーザーは、すべてのスイッチポートインターフェイスの target_config_file に目的のポート ステート（shutdown または no shutdown）ステートメントが存在することを確認する必要があります。
- コンフィギュレーションの置換操作を正常に行うには、ターゲット コンフィギュレーション ファイルの ACL のすべての ACE エントリにシーケンス番号が存在する必要があります。
- Cisco NX-OS Release 9.3（6）以降では、**service exclude-bootconfig** の設定によって**boot nxos** イメージ設定を、**show running-config**、**show startup-config**、 **copy running-config** *filename*、および **copy startup-config** *filename* コマンドで除外できます。
- コミットタイムアウト機能は、コミットタイムアウトを使用してコンフィギュレーションの置換操作を実行する場合にのみ開始されます。タイマーの値の範囲は 30 ～ 3600 秒です。
- ユーザ指定のコンフィギュレーション ファイルは、Cisco NX-OS デバイスから取得（copy run file）された有効な show running-configuration の出力である必要があります。このコンフィぎゅーレーションは部分コンフィギュレーションにすることはできず、user admin などの必須コマンドが含まれている必要があります。
- ソフトウェア バージョン違いで生成された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の置換操作を実行することは、操作が失敗する可能性があるため推奨されません。ソフトウェアバージョンの変更があるたびに新しいコンフィギュレーションファイルを再生成する必要があります。

- Multichassis EtherChannel トランク（MCT）設定を仮想ピアリンク設定と置き換えようとした場合、コンフィギュレーションの置換操作はサポートされません。物理 MCT はイーサネットを介した CFS 配信モードを使用し、仮想ピアリンクは IP を介した CFS 配信モードを使用するため、この操作は許可されません。
- コンフィギュレーションの置換操作が進行中の場合、他のセッションからはコンフィギュレーションを変更しないことを推奨します。操作が失敗する可能性があります。
- コンフィギュレーションの置換機能については、次の点に注意してください。
  - Cisco NX-OS リリース 9.3(5) 以降では、FEX インターフェイス コンフィギュレーションの設定置換（CR）がサポートされています。FEX のプロビジョニングは CR ではサポートされていません。プロビジョニングされた FEX インターフェイスの設定は、CR を使用して変更できます。
  - FEX ライン カードがオフラインの場合、コンフィギュレーションの置換機能は動作しません。
  - -R ライン カード搭載の Cisco Nexus 9500 プラットフォーム スイッチでは、コンフィギュレーションの置換機能はサポートされません。
  - Cisco NX-OS リリース 9.3 (5) 以降では、設定置換機能がポート プロファイルでサポートされています。
  - コンフィギュレーションの置換機能は、configure terminal モード コマンドで**のみ**サポートされます。configure profile、configure jobs、およびその他のモードはサポートされていません。
  - Cisco NX-OS リリース 9.3(5) 以降では、ジョブの設定モードがサポートされています。スケジューラ ジョブ コマンドを含むコンフィギュレーション ファイルは、コンフィギュレーションの置換に使用できます。
  - Cisco NX-OS リリース 9.3(4) 以降では、ブレークアウト インターフェイス コンフィギュレーションの設定置換機能がサポートされています。
  - 実行コンフィギュレーションに **feature-set mpls** または **mpls static range** コマンドが含まれていて、MPLS なしでコンフィギュレーションに移動しようとしたり、ラベルの範囲を変更する場合、コンフィギュレーションの置換機能が失敗することがあります。
  - コンフィギュレーションの置換機能は、自動設定をサポートしていません。
- コンフィギュレーションの置換機能が適用されるライン カードがオフラインである場合、コンフィギュレーションの置換操作は失敗します。
- 設定置換機能を使用してITDを変更する前に、ITD サービスをシャットダウンする必要があります（**shutdown**）。
- ユーザ コンフィギュレーションからのメンテナンス モードへの移行はサポートされていません。

- メンテナンス モードから **configure replace** コマンドを使用すると、次の警告でユーザの確認が求められます。

```
Warning: System is in maintenance mode. Please ensure user config won't inadvertently
 revert back config in maintenance mode profile.
            Do you wish to proceed anyway? (y/n)  [n]
```

- <*non-interactive*> オプションを使用してメンテナンス モードから **configure replace** コマンドを使用することはサポートされています。デフォルトでは、*yes* のユーザ確認を受けてから進行します。
- シーケンス番号は、CLI **ip community-list** および **ip as-path access-list** コマンドに必須です。シーケンス番号を指定しないと、構成の置換操作は失敗します。
- コンフィギュレーションを適用するために Cisco NX-OS デバイスをリロードする必要がある場合、これらのコンフィギュレーションをリロードしてからコンフィギュレーションの置換操作を行う必要があります。
- ユーザ指定のコンフィギュレーション ファイルでのコマンドの順序は、Cisco Nexus スイッチの実行コンフィギュレーションでのこれらのコマンドと同じにする必要があります。
- CR を使用してスイッチの実行コンフィギュレーションを置き換える必要があるユーザ コンフィギュレーション ファイルは、新しいコマンドを設定した後、スイッチの実行コンフィギュレーションから生成する必要があります。ユーザ コンフィギュレーション ファイルは、CLI コマンドを使用して手動で編集しないでください。また、コンフィギュレーション コマンドのシーケンスを変更しないでください。
- セマンティック検証は、4 ギガビット メモリ プラットフォームではサポートされていません。
- 異なるバージョンの機能が実行コンフィギュレーションとユーザ コンフィギュレーションに存在する場合（VRRPv2 と VRRPv3 など）、セマンティック検証オプションが期待どおりに機能しません。この問題は既知の制限です。
- 「verify-only」モードでは、TCAM 依存の設定はエラーをスローせず、成功する場合があります。ただし、実際の CR 操作では失敗する可能性があります。これを回避するには、CR を実行する前に TCAM カービング設定を適用してリロードすることをお勧めします。
- Cisco NX-OS リリース 10.3(1)F 以降、構成の置換機能は機能アプリ ホスティングをサポートしません。

コンフィギュレーションの置換の推奨ワークフロー

コンフィギュレーションの置換の推奨されるワークフローを次に示します。

（注）
- このワークフローは、候補構成でも同じである必要があります。
- 候補構成のデフォルト構成はサポートされていません。

1. Cisco Nexus シリーズ デバイスで最初にコンフィギュレーションを適用してコンフィギュレーション ファイルを生成してから、コンフィギュレーション ファイルとして **show running-configuration** 出力を使用します。このファイルを使用して、必要に応じてコンフィギュレーションを変更します。次に、この生成または更新されたコンフィギュレーション ファイルを使用して、コンフィギュレーションの置換を実行します。

（注） ソフトウェア バージョンの変更があるたびにコンフィギュレーション ファイルを再生成する必要があります。異なるソフトウェア バージョンで生成されたコンフィギュレーション ファイルを使用してコンフィギュレーションの置換操作を実行することは推奨されません。

2. **configure replace** <*file*> **show-patch** コマンドを実行してパッチ ファイルを表示し、確認します。この手順は任意です。
3. 構成の置換ファイルを実行するか、**commit-timeout** <*time*>機能をスキップします。要件に基づいて、次の手順のいずれかを実行できます。
   - コンフィギュレーションの置換で実行されるコマンドをコンソールに表示するには、**configure replace** <*file*> **verbose** を実行します。
   - **configure replace** [**bootflash**/**scp**/**sftp**] <*user-configuration-file*> **verbose commit-timeout** <*time*> コマンドを実行して、コミット時間を構成します。
4. **configure replace commit** コマンドを実行し、コミット タイマーを停止します。この手順は、コミットタイムアウト機能でコンフィギュレーションの置換操作を実行している場合に必要です。
5. コンフィギュレーションのセマンティック検証を含むプレチェックをコンフィギュレーションの置換で実行します。エラーがある場合、コンフィギュレーションの置換操作は失敗します。失敗したコンフィギュレーションの詳細を表示するには、**show config-replace log verify** コマンドを使用します。パッチファイルを適用すると、コンフィギュレーションの置換によって検証プロセスがトリガーされます。コンフィギュレーションの置換は、検証プロセスで、実行コンフィギュレーションとユーザー構成ファイルを比較します。不一致がある場合、デバイスは元のコンフィギュレーションに復元されます。不一致のコンフィギュレーションを表示するには、**show config-replace log verify** コマンドを使用します。
6. Cisco NX-OS リリース9.3(1) では、次のコンフィギュレーションの置換操作を実行できます。
   - セマンティック検証およびベストエフォートモードなしのコンフィギュレーションの置換。
   - セマンティック検証なし、ベストエフォートモードありのコンフィギュレーションの置換。
   - セマンティック検証あり、ベストエフォートモードなしのコンフィギュレーションの置換。

- セマンティック検証およびベストエフォートモードありのコンフィギュレーションの置換。

コンフィギュレーションの置換の実行

コンフィギュレーションの置換を実行するには、次の操作を行います。

始める前に

現在の構成ファイルと候補構成ファイルの IP アドレスに競合がないことを確認します。IP アドレスの競合の例は、現在の構成ファイルの eth インターフェイス 1/53 で 172.16.0.1/24 を設定し、eth 1/53 で 172.16.0.1/24 と 192.168.0.1/24 を使用してポートチャネル 30 を設定したとします。候補構成ファイル内。候補構成ファイルの構成置換を実行すると、IP アドレスの競合が発生します。

手順の概要

1. **configure replace** { < *uri_local* > | < *uri_remote* > } [**verbose** | **show-patch**]
2. **configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **show-patch**
3. **configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **verbose**
4. **configure replace** <*user-configuration-file*> [**best-effort**]
5. **configure replace** <*user-configuration-file*> [**verify-and-commit**]
6. **configure replace** <*user-configuration-file*> [**verify-only**]
7. （任意） **configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **verbose commit-timeout** <*time*>
8. （任意） **configure replace** [**commit**]
9. （任意） **configure replace** [**bootflash**/**scp**/**sftp**] <*user-configuration-file*> *non-interactive*
10. （任意） **configure replace** <*user-configuration-file*> [**on-failure reload** [**save-startup-config**]]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**configure replace** { < *uri_local* > | < *uri_remote* > } [**verbose** | **show-patch**]	コンフィギュレーションの置換を実行します。コンフィギュレーションの置換の進行中にセッションを通じてコンフィギュレーションを変更すると、コンフィギュレーションの置換操作は失敗します。1 つのコンフィギュレーション要求がすでに進行中であるときにコンフィギュレーションの置換要求を送信すると、要求はシリアル化されます。

	コマンドまたはアクション	目的
ステップ 2	**configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **show-patch**	実行コンフィギュレーションとユーザ指定のコンフィギュレーションの違いを表示します。 （注） • このコマンドでは、プレーン テキスト パスワードは暗号化されません。 • このコマンドは、CLI snmp-server traps コマンドの構成置換が成功した後でも、パッチを表示できます。
ステップ 3	**configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **verbose**	スイッチのコンフィギュレーションを、ユーザが提供する新しいユーザ コンフィギュレーションに置換します。コンフィギュレーションの置換は常にアトミックです。
ステップ 4	**configure replace** <*user-configuration-file*> [**best-effort**]	スイッチの設定を新しいユーザ設定に置き換え、セマンティック検証による設定の置き換えを有効にします。 best-effort オプションを使用すると、コマンドでエラーが発生した場合でも設定の置換によって完全なパッチが実行され、以前の設定がロールバックされないようになります。 Cisco NX-OS リリース 10.5(1)F 以降、コンフィギュレーション置換機能は、Cisco Nexus 9300-FX2/FX3/GX シリーズ スイッチのバッチ ACL コンフィギュレーションをサポートします。 **ベストエフォート**モードが有効になっている場合、バッチ構成内で障害が発生すると、その特定のバッチ内の構成セット全体がスキップされます。
ステップ 5	**configure replace** <*user-configuration-file*> [**verify-and-commit**]	スイッチの設定を新しいユーザ設定に置き換え、セマンティック検証による設定の置き換えを有効にします。 verify-and-commit オプションは、セマンティック検証を有効にするために使用されます。パッチは、完全なパッチのセマンティック検証に合格した場合にのみ実行されます。 ベストエフォート オプション、verify-and-commit オプション、または両方のオプションを同時に使用できます。

	コマンドまたはアクション	目的
ステップ 6	**configure replace** <*user-configuration-file*> [**verify-only**]	パッチのみを表示し、パッチでセマンティック検証を実行し、結果を表示します。パッチはシステムに適用されません。
ステップ 7	（任意） **configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **verbose commit-timeout** <*time*>	コミット時間を秒単位で設定します。タイマーは、コンフィギュレーションの置換操作が正常に完了した後に開始されます。
ステップ 8	（任意） **configure replace** [**commit**]	コミットタイマーを停止し、コンフィギュレーションの置換設定を続行します。 （注） この手順は、コミットタイムアウト機能を設定している場合にのみ適用されます。 （注） 以前のコンフィギュレーションにロールバックするには、コミット タイマーの期限が切れるまで待機する必要があります。タイマーの期限が切れると、スイッチは自動的に以前のコンフィギュレーションにロールバックされます。
ステップ 9	（任意） **configure replace** [**bootflash**/**scp**/**sftp**] <*user-configuration-file*> *non-interactive*	メンテナンス モードでは、ユーザ プロンプトはありません。デフォルトでは、**yes** のユーザ確認を受けてからロールバックが進行します。非インタラクティブ オプションは、メンテナンス モードでのみ使用できます。
ステップ 10	（任意） **configure replace** <*user-configuration-file*> [**on-failure reload** [**save-startup-config**]]	このオプションは、設定の置換操作が失敗し、以前の実行コンフィギュレーションへの復元も失敗した場合に、以前に保存したスタートアップ コンフィギュレーションにスイッチをリロードします。 このオプションは、次の場合には影響しません。 • 設定の置換が正常に行われました • 設定の置換が失敗するが、以前の実行コンフィギュレーションへの復元は成功する （注） [**save-startup-config**] : このオプションは、設定の置換操作を開始する前に、実行コンフィギュレーションをスタートアップに保存します。

コンフィギュレーションの置換の確認

コンフィギュレーションの置換とそのステータスをチェックして確認するには、表に記載されているコマンドを使用します。

表 *27*：コンフィギュレーションの置換の確認

コマンド	目的
configure replace [**bootflash**/**scp**/**sftp**]<*user-configuration-file*] **show-patch**	実行コンフィギュレーションとユーザ指定のコンフィギュレーションの違いを表示します。
show config-replace log exec	実行したすべてのコンフィギュレーションと失敗したコンフィギュレーションのログを表示します。エラーの場合、そのコンフィギュレーションに対してエラー メッセージが表示されます。
show config-replace log verify	失敗したコンフィギュレーションをエラーメッセージとともに表示します。成功したコンフィギュレーションは表示されません。
show config-replace status	コンフィギュレーションの置換操作のステータス（進行中、成功、失敗など）を表示します。コミットタイムアウト機能を設定している場合、コミットとタイマーのステータスに加え、コミット タイムアウトの残り時間も表示されます。

コンフィギュレーションの置換の例

以下のコンフィギュレーションの置換の設定例を参照してください。

- **configure replace bootflash:** <*file*> **show-patch** CLI コマンドを使用して、実行コンフィギュレーションとユーザ指定のコンフィギュレーションの違いを表示します。

```
switch(config)# configure replace bootflash:<file> show-patch
Collecting Running-Config
Converting to checkpoint file
#Generating Rollback Patch
!!
no role name abc
```

- **configure replace bootflash:** <*file*> **verbose** CLI コマンドを使用して、スイッチの実行コンフィギュレーション全体をユーザコンフィギュレーションに置換します。

```
switch(config)# configure replace bootflash:<file> verbose
Collecting Running-Config
```

```
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallelly may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
==========================================================
config t
no role name abc
==========================================================
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

Sample Example with adding of BGP configurations.
 switch(config)# sh run | section bgp
 switch(config)# sh file bootflash:file | section bgp
feature bgp
router bgp 1
    address-family ipv4 unicast
    neighbor 1.1.1.1
switch(config)#
switch(config)# configure replace bootflash:file verbose
Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallelly may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
==========================================================
config t
feature bgp
router bgp 1
address-family ipv4 unicast
neighbor 1.1.1.1
==========================================================
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

switch(config)# sh run | section bgp
feature bgp
router bgp 1
  address-family ipv4 unicast
  neighbor 1.1.1.1

Sample Example with ACL
 switch(config)# configure replace bootflash:run_1.txt
 Collecting Running-Config
 Generating Rollback patch for switch profile
 Rollback Patch is Empty
 Note: Applying config parallelly may fail Rollback verification
 Collecting Running-Config
 #Generating Rollback Patch
 Executing Rollback Patch
 ==========================================================
 config t
 no ip access-list nexus-50-new-xyz
 ip access-list nexus-50-new-xyz-jkl-abc
 10 remark Newark
 20 permit ip 17.31.5.0/28 any
```

```
30 permit ip 17.34.146.193/32 any
40 permit ip 17.128.199.0/27 any
50 permit ip 17.150.128.0/22 any
==========================================================
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

switch(config)#


switch(config)# show run aclmgr | sec nexus-50-new-xyz-jkl-abc
ip access-list nexus-50-new-xyz-jkl-abc
  10 remark Newark
  20 permit ip 17.31.5.0/28 any
  30 permit ip 17.34.146.193/32 any
  40 permit ip 17.128.199.0/27 any
  50 permit ip 17.150.128.0/22 any
```

- **configure replace bootflash:user-config.cfg verify-only** CLI コマンドを使用して、パッチを意味的に生成および確認します。

```
switch(config)# configure replace bootflash:user-config.cfg verify-only

Version match between user file and running configuration.
Pre-check for User config PASSED
Collecting Running-Config
Converting to checkpoint file
Generating Rollback Patch
Validating Patch
==========================================================
`config t `
`interface Ethernet1/1`
`shutdown`
`no switchport trunk allowed vlan`
`no switchport mode`
`no switchport`
`exit`
Skip non dme command for CR validation
`interface Vlan1`
`shutdown`
`interface Ethernet1/1`
`shutdown`
`no switchport`
`ip address 1.1.1.1/24`
`exit`
Skip non dme command for CR validation
==========================================
Patch validation completed successful
switch(config)#
```

- パッチでセマティック検証を実行した後、**configure replace bootflash:user-config.cfg best-effort verify-and-commit** CLI コマンドを使用して、スイッチの実行コンフィギュレーションを特定のユーザ コンフィギュレーションに置き換えます。

```
switch(config)# configure replace bootflash:user-config.cfg best-effort
verify-and-commit

 Version match between user file and running configuration.
Pre-check for User config PASSED
ADVISORY: Config Replace operation started...
Modifying running configuration from another VSH terminal in parallel
```

```
is not recommended, as this may lead to Config Replace failure.

Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Collecting Running-Config
Generating Rollback Patch

Validating Patch
Patch validation completed successful
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Rollback Patch

Configure replace completed successfully. Please run 'show config-replace log exec'
 to see if there is any configuration that requires reload to take effect.

switch(config)#
```

- **show config-replace log exec** CLI コマンドを使用して、実行したコンフィギュレーションと、存在する場合はエラーをすべて確認します。

```
switch(config)# show config-replace log exec
Operation            : Rollback to Checkpoint File
Checkpoint file name : .replace_tmp_28081
Scheme               : tmp
Rollback done By     : admin
Rollback mode        : atomic
Verbose              : enabled
Start Time           : Wed, 06:39:34 25 Jan 2017
-------------------------------------------------------------------------------

time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS
End Time             : Wed, 06:39:47 25 Jan 2017
Rollback Status      : Success

Executing Patch:
----------------
switch#config t
switch#no role name abc
```

- **show config-replace log verify** CLI コマンドを使用して、存在する場合は失敗したコンフィギュレーションを確認します。

```
switch(config)# show config-replace log verify
Operation            : Rollback to Checkpoint File
Checkpoint file name : .replace_tmp_28081
Scheme               : tmp
Rollback done By     : admin
Rollback mode        : atomic
Verbose              : enabled
Start Time           : Wed, 06:39:34 25 Jan 2017
End Time             : Wed, 06:39:47 25 Jan 2017
Status               : Success

Verification patch contains the following commands:
--------------------------------------------------
!!
! No changes
-------------------------------------------------------------------------------
```

```
time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS
```

- **show config-replace status** CLI コマンドを使用して、コンフィギュレーションの置換のステータスを確認します。

```
switch(config)# show config-replace status
Last operation : Rollback to file
Details:
  Rollback type: atomic replace_tmp_28081
  Start Time: Wed Jan 25 06:39:28 2017
  End Time: Wed Jan 25 06:39:47 2017
  Operation Status: Success
switch(config)#
```

スイッチから生成された設定の代わりに手動で作成された設定を使用すると、[置換の設定（Configure Replace）] が失敗することがあります。失敗の原因として考えられるのは、show running configuration に示されていないデフォルト設定の潜在的な違いです。次の例を参照してください。

power redundancy コマンドがデフォルトのコマンドである場合、デフォルトの設定では表示されません。ただし、**show run all** コマンドを使用すると表示されます。次の例を参照してください。

```
switch# show run all

!Command: show running-config all
!Running configuration last done at: Tue Nov 12 11:07:44 2019
!Time: Tue Nov 12 11:16:09 2019

version 9.3(1) Bios:version 05.39
power redundancy-mode ps-redundant
no hardware module boot-order reverse
no license grace-period
<snip>
hostname n9k13
```

電源冗長コマンドは、show running configuration コマンド出力には表示されません。次の例を参照してください。

```
!Command: show running-config
!Running configuration last done at: Tue Nov 12 11:07:44 2019
!Time: Tue Nov 12 11:17:24 2019

version 9.3(1) Bios:version 05.39
hostname n9k13
```

設定置換のユーザ コンフィギュレーションに **power redundancy-mode ps-redundant** コマンドが追加された場合。検証/コミットが失敗する可能性があります。次の例を参照してください。

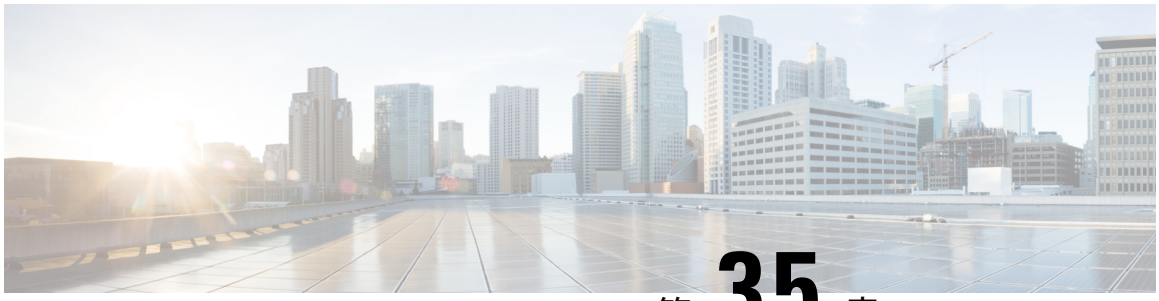
```
switch# show file bootflash:test

!Command: show running-config
!Running configuration last done at: Tue Nov 12 10:56:49 2019
!Time: Tue Nov 12 11:04:57 2019

version 9.3(1) Bios:version 05.39
power redundancy-mode ps-redundant
hostname n9k13
```

power redundancy-mode ps-redundant コマンドは、設定置換の後の show running には表示されません。したがって、「欠落」と見なされ、CR は失敗します。次に例を示します。

```
switch# config replace bootflash:test verify-and-commit

Version match between user file and running configuration.
Pre-check for User config PASSED
ADVISORY: Config Replace operation started...
Modifying running configuration from another VSH terminal in parallel
is not recommended, as this may lead to Config Replace failure.

Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Collecting Running-Config
.Generating Rollback Patch

Validating Patch
Patch validation completed successful
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Rollback Patch
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Patch for verification
Verification failed, Rolling back to previous configuration
Collecting Running-Config
Cleaning up switch-profile buffer
Generating Rollback patch for switch profile
Executing Rollback patch for switch profiles. WARNING - This will change the
configuration of switch profiles and will also affect any peers if configured
Collecting Running-Config
Generating Rollback Patch
Rollback Patch is Empty
Rolling back to previous configuration is successful

Configure replace failed. Use 'show config-replace log verify' or 'show config-replace
log exec' to see reasons for failure

n9k13# show config-replace log verify
Operation : Config-replace to user config
Checkpoint file name : .replace_tmp_31849
Scheme : tmp
Cfg-replace done By : agargula
Cfg-replace mode : atomic
Verbose : disabled
Start Time : Tue, 11:20:59 12 Nov 2019
Start Time UTC : Tue, 10:20:59 12 Nov 2019
------------------------------------------
End Time : Tue, 11:21:28 12 Nov 2019
End Time UTC : Tue, 10:21:28 12 Nov 2019
Status : Failed

Verification patch contains the following commands:
--------------------------------------------------
!!
Configuration To Be Added Missing in Running-config
==================================================
!
power redundancy-mode ps-redundant
```

```
Undo Log
--------------------------------------------------------------------------------
End Time : Tue, 11:21:32 12 Nov 2019
End Time UTC : Tue, 10:21:32 12 Nov 2019
Status : Success
n9k13#
```

上記の例では、CR は欠落しているデフォルトのコマンドを考慮します。



第 35 章

ロールバックの設定

この章では、Cisco NX-OS デバイスでロールバックを設定する方法について説明します。

この章は、次の内容で構成されています。

- ロール バックについて （681 ページ）
- ロールバックの前提条件 （683 ページ）
- ロールバックの注意事項と制約事項 （683 ページ）
- ロールバックのデフォルト設定 （684 ページ）
- ロールバックの設定 （684 ページ）
- ロールバック コンフィギュレーションの確認 （686 ページ）
- ロールバックの設定例 （687 ページ）
- その他の参考資料 （687 ページ）

ロール バックについて

ロールバックを使用すると、Cisco NX-OS コンフィギュレーションのスナップショットまたはユーザ チェックポイントを使用して、デバイスをリロードしなくても、いつでもそのコンフィギュレーションをデバイスに再適用できます。権限のある管理者であれば、チェックポイントで設定されている機能について専門的な知識がなくても、ロールバック機能を使用して、そのチェックポイント コンフィギュレーションを適用できます。

Cisco NX-OS は、システムのチェックポイントを自動的に作成します。ユーザまたはシステムのチェックポイントのいずれかを使用して、ロールバックを実行できます。

いつでも、現在の実行コンフィギュレーションのチェックポイント コピーを作成できます。Cisco NX-OS はこのチェックポイントを ASCII ファイルとして保存するので、将来、そのファイルを使用して、実行コンフィギュレーションをチェックポイント コンフィギュレーションにロールバックできます。複数のチェックポイントを作成すると、実行コンフィギュレーションのさまざまなバージョンを保存できます。

実行コンフィギュレーションをロールバックするとき、次のロールバック タイプを発生させることができます。

- atomic：エラーが発生しなかった場合に限り、ロールバックを実装します。

- best-effort：ロールバックを実装し、エラーがあってもスキップします。
- stop-at-first-failure：エラーが発生した場合は中止されるロールバックを実装します。

デフォルトのロールバック タイプは atomic です。

チェックポイントコンフィギュレーションにロールバック可能になった時点で、現在の実行コンフィギュレーションに適用される変更を確認してから、ロールバック操作にコミットできます。ロールバック操作時にエラーが発生した場合は、操作を取り消すか、またはエラーを無視してロールバック操作を続行するかを選択できます。操作を取り消した場合、Cisco NX-OS はエラーが発生するまでに、すでに適用した変更のリストを提示します。これらの変更は手動で処理する必要があります。

システム チェックポイントの自動生成

Cisco NX-OS ソフトウェアは、コンフィギュレーション情報が消失しないよう、システム チェックポイントを自動的に生成します。システム チェックポイントは次のイベントによって生成されます。

- **no feature** コマンドで、有効になっている機能を無効にする
- **no router bgp** コマンドや **no ip pim sparse-mode** コマンドで、レイヤ 3 プロトコルのインスタンスを削除する
- 機能のライセンスの有効期限が切れる

これらのイベントのいずれかによってシステム コンフィギュレーションの変更が生じると、この機能ソフトウェアによって、システム チェックポイントが作成されます。これを使用すると、以前のシステム コンフィギュレーションへロールバックできます。システムで生成されたチェックポイント ファイルの名前は「system-」で始まり、機能名が含まれています。たとえば、EIGRP 機能を最初にディセーブルにすると、システムは、system-fm-__inst_1__eigrp という名前のチェックポイントを作成します。

高可用性

checkpoint または checkpoint checkpoint_name コマンドを使用してチェックポイントが作成されるときは必ず、チェックポイントはスタンバイ ユニットと同期されます。

ロールバックではチェックポイント操作の状況を記憶しています。このためチェックポイント操作が中断された場合、およびシステムが不整合の状態になった場合には、ロールバック操作を続行する前に、ロールバックでチェックポイント操作（スタンバイ ユニットへのチェックポイントの同期化）を完了できます。

チェックポイント ファイルは、プロセスのリスタート後またはスーパーバイザのスイッチオーバー後も引き続き使用できます。プロセスの再起動中またはスーパーバイザのスイッチオーバー中に中断された場合でも、操作を続行する前にチェックポイントが正常に完了します。スーパーバイザのスイッチオーバーでは、チェックポイントは新しいアクティブ ユニットで完了します。

ロールバック操作中にプロセスの再起動またはスーパーバイザのスイッチオーバーが生じた場合は、再起動またはスイッチオーバーが完了した後で、ロールバックが以前の状態から再開し、正常に終了します。

仮想化のサポート

Cisco NX-OS は実行コンフィギュレーションのチェックポイントを作成します。異なるチェック ポイント コピーを作成できます。

ロールバックの前提条件

ロール バックを設定するには、network-admin のユーザ権限が必要です。

ロールバックの注意事項と制約事項

ロールバックに関する設定時の注意事項および制約事項は、次のとおりです。

- 作成できるチェックポイント コピーの最大数は 10 です。
- チェックポイント ファイル名の長さは、最大 80 文字です。
- チェックポイントのファイル名の先頭を *system* にすることはできません。
- チェックポイントのファイル名の先頭を *auto* にすることができます。
- チェックポイントのファイル名を、*summary* または *summary* の略語にすることができます。
- チェックポイント、ロールバック、または実行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のコピーを同時に実行できるのは、1 ユーザだけです。
- システムで **write erase** および **reload** コマンドを実行すると、チェックポイントは削除されます。**clear checkpoint database** コマンドを使用すると、すべてのチェックポイント ファイルを削除できます。
- 異なるソフトウェアバージョン間でのチェックポイントのロールバックはサポートされていませんが、ユーザは自己判断でロールバックを実行し、best-effort モードでエラーから回復できます。
- ブートフラッシュでチェックポイントを作成した場合、ロールバックの実行前は実行システム コンフィギュレーションとの違いは実行できず、「変更なし」と報告されます。
- **checkpoint** および **checkpoint***checkpoint_name* コマンドを使用して作成されるチェックポイントは、スイッチオーバーの直後に出現します。
- チェック ポイントは、リロードの前に **write erase** コマンドを発行しない限り、リロードの直後に出現します。

- ブートフラッシュ時のファイルへのロールバックは、**checkpoint** *checkpoint_name* コマンドを使用して作成されたファイルでのみサポートされます。他のASCIIタイプのファイルではサポートされません。
- チェックポイントの名前は一意にする必要があります。以前に保存したチェックポイントを同じ名前で上書きすることはできません。
- ロールバックは自動設定のコンテキストではサポートされません。チェックポイントは自動設定を保存しません。したがって、ロールバックを実行した後、対応する自動設定は存在しないことになります。
- ロールバック操作中にインターフェイスに複数のポートVLANマッピングを設定すると、ロールバック機能が失敗します。

ロールバックのデフォルト設定

次の表に、ロールバック パラメータのデフォルト設定を示します。

パラメータ	デフォルト
ロールバック タイプ	アトミック

ロールバックの設定

（注） Cisco NX-OS コマンドは Cisco IOS コマンドと異なる場合があるので注意してください。

チェックポイントの作成

設定には、最大 10 個のチェック ポイントを作成できます。

手順の概要

1. [**no**] **checkpoint** {[*cp-name*] [**description** *descr*] | **file** *file-name* }
2. （任意） **show checkpoint** *cp-name* [**all**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	[**no**] **checkpoint** {[*cp-name*] [**description** *descr*] | **file** *file-name* } 例： `switch# checkpoint stable`	ユーザ チェックポイント名またはファイルのいずれかに対して、実行中のコンフィギュレーションのチェックポイントを作成します。チェックポイント名には最大 80 文字の任意の英数字を使用できますが、スペースを含めることはできません。チェックポイント名を指定しなかった場合、Cisco NX-OS はチェックポイント名を user-checkpoint-*number* に設定します。ここで *number* は 1 ～ 10 の値です。 description には、スペースも含めて最大 80 文字の英数字を指定できます。 **checkpoint** コマンドの **no** 形式を使用すると、チェックポイント名を削除できます。**delete** コマンドを使用して、チェックポイント ファイルを削除できます。
ステップ 2	（任意） **show checkpoint** *cp-name* [**all**] 例： `switch# show checkpoint stable`	チェックポイント名の内容を表示します。

ロールバックの実装

チェックポイント名またはファイルにロールバックを実装できます。ロールバックを実装する前に、現在のコンフィギュレーションまたは保存されているコンフィギュレーションを参照しているソースと宛先のチェックポイント間の差異を表示できます。

（注） atomic ロールバック中に設定を変更すると、ロールバックは失敗します。

手順の概要

1. **show diff rollback-patch** { **checkpoint** *src-cp-name* | **running-config** | **startup-config** | **file** *source-file*} { **checkpoint** *dest-cp-name* | **running-config** | **startup-config** | **file** *dest-file*}
2. **rollback running-config** {**checkpoint** *cp-name* | **file** *cp-file*} [**atomic** | **best-effort** | **stop-at-first-failure**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**show diff rollback-patch** { **checkpoint** *src-cp-name* | **running-config** | **startup-config** | **file** *source-file*} { **checkpoint** *dest-cp-name* | **running-config** | **startup-config** | **file** *dest-file*} 例： `switch# show diff rollback-patch checkpoint stable running-config`	ソースと宛先のチェックポイント間の差異を表示します。
ステップ 2	**rollback running-config** {**checkpoint** *cp-name* | **file** *cp-file*} [**atomic** | **best-effort** | **stop-at-first-failure**] 例： `switch# rollback running-config checkpoint stable`	指定されたチェックポイント名またはファイルへのロールバックを作成します。次のロールバック タイプを実装できます。 • atomic：エラーが発生しなかった場合に限り、ロールバックを実装します。 • best-effort：ロールバックを実装し、エラーがあってもスキップします。 • stop-at-first-failure：エラーが発生した場合は中止されるロールバックを実装します。 デフォルトは atomic です。 次に、ユーザ チェックポイント名に対するロールバックを実装する例を示します。

ロールバック コンフィギュレーションの確認

ロールバックのコンフィギュレーション情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show checkpoint *name* [**all**]	チェックポイント名の内容を表示します。
show checkpoint all [**user** | **system**]	すべてのチェック ポイントの内容を表示します。表示されるチェックポイントを、ユーザまたはシステムで生成されるチェックポイントに限定できます。
show checkpoint summary [**user** | **system**]	すべてのチェック ポイントの一覧を表示します。表示されるチェックポイントを、ユーザまたはシステムで生成されるチェックポイントに限定できます。

コマンド	目的
show diff rollback-patch { **checkpoint** *src-cp-name* | **running-config** | **startup-config** | **file** *source-file*} { **checkpoint** *dest-cp-name* | **running-config** | **startup-config** | **file** *dest-file*}	ソースと宛先のチェックポイント間の差異を表示します。
show rollback log [**exec** | **verify**]	ロールバック ログの内容を表示します。

すべてのチェックポイント ファイルを削除するには、**clear checkpoint database** コマンドを使用します。

ロールバックの設定例

次に、チェックポイント ファイルを作成して、ユーザ チェックポイント名に対する best-effort ロールバックを実装する例を示します。

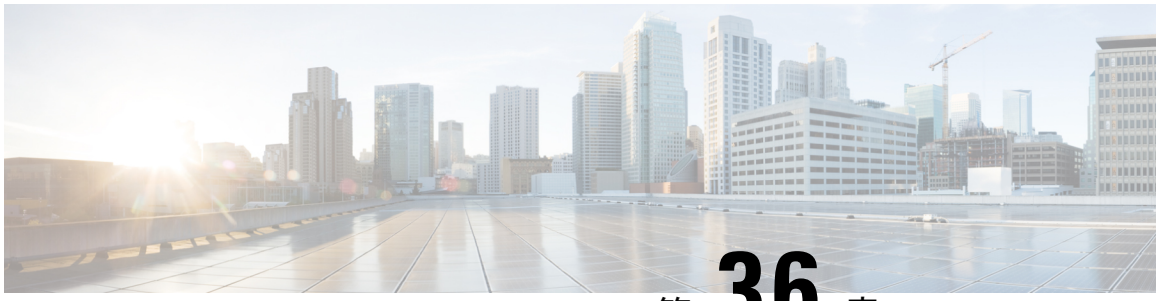
```
checkpoint stable
rollback running-config checkpoint stable best-effort
```

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
コンフィギュレーション ファイル	『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』

関連資料



第 36 章

候補構成の完全性チェック

本章では、候補構成の完全性チェックの方法について説明します。

この章は、次の項で構成されています。

- 候補構成について （689 ページ）
- 候補構成の完全性チェックの注意事項と制限事項 （689 ページ）
- 候補構成の完全性チェックの実行 （690 ページ）
- 完全性チェックの例 （691 ページ）

候補構成について

候補構成は、実行構成のサブセットです。実行構成は、追加、変更、または削除を行わずに、実行構成内に候補構成が存在するかどうかを確認します。

候補構成の完全性を確認するには、次のコマンドを使用します。

- `show diff running-config`
- `show diff startup-config`

CLI の詳細については、候補構成の完全性チェックの実行 （690 ページ） を参照してください。

候補構成の完全性チェックの注意事項と制限事項

候補構成の完全性チェックには、次の注意事項と制限事項があります。

- Cisco NX-OS リリース 10.2(3)F 以降、すべての Cisco Nexus スイッチに候補構成の完全性チェック オプションが導入されました。
- 部分構成ではなく、完全な実行構成の入力として完全性チェックを実行する必要がある場合は、**partial** キーワードを使用しないことをお勧めします。
- 生成された実行構成に表示される行番号は、内部で生成されたものであるため、候補構成とは一致しません。

- 実行構成と候補構成に違いがある場合、インラインで出力表示されます。
- 候補ファイルの構成ブロック全体が新たに追加されたものである場合、生成される実行構成の最後に追加されます。
- 候補設定に SNMP または AAA ユーザー CLI とクリアテキスト パスワードがある場合、ユーザーがすでに設定されている場合でも、SNMP ユーザーは diff として表示されます。

候補構成の完全性チェックの実行

完全性チェックを実行するには、次のコマンドを実行します。

始める前に

（注） 完全性チェックを実行する前に、実行構成と候補構成が同じイメージバージョンに属していることを確認してください。

手順の概要

1. **show diff running-config** *file_url* [**unified**]
2. **show diff startup-config** *file_url* [**unified**]

手順の詳細

手順

	コマンドまたはアクション	目的
ステップ 1	**show diff running-config** *file_url* [**unified**] 例： `switch# show diff running-config bootflash:candidate.cfg partial unified`	実行構成とユーザーが指定した候補構成の違いを表示します。 • *file_url*： と比較するファイルのパス。 • **unified**：実行構成とユーザー構成の違いを統一された形式で表示します。
ステップ 2	**show diff startup-config** *file_url* [**unified**] 例： `switch# show diff startup-config bootflash:candidate.cfg unified`	スタートアップ構成とユーザーが指定した候補構成の違いを表示します。 • *file_url*： と比較するファイルのパス。 • **unified**：スタートアップ構成とユーザー構成の違いを統一された形式で表示します。

完全性チェックの例

実行構成と候補構成の間に相違点はない

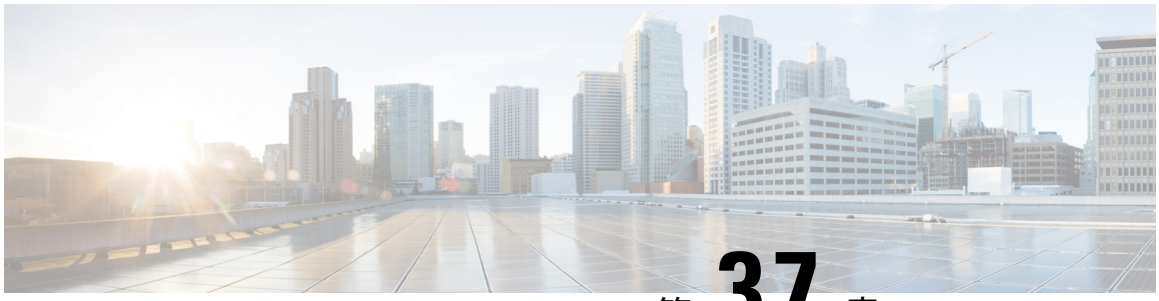
```
switch# show diff running-config bootflash:base_running.cfg
switch#
```

実行構成と候補構成の間の相違点

```
switch# show diff running-config bootflash:modified-running.cfg unified
--- running-config
+++ User-config
@@ -32,11 +32,11 @@

interface Ethernet1/1
   mtu 9100
   link debounce time 0
   beacon
-  ip address 2.2.2.2/24
+  ip address 1.1.1.1/24
   no shutdown

interface Ethernet1/2

interface Ethernet1/3
switch#
```

実行構成と部分候補構成の間の相違点

```
switch# show file bootflash:intf_vlan.cfg
interface Vlan101
  no shutdown
  no ip redirects
  ip address 1.1.2.1/24 secondary
  ip address 1.1.1.1/24
switch#
switch# show diff running-config bootflash:intf_vlan.cfg partial unified
--- running-config
+++ User-config
@@ -3897,10 +3883,14 @@
   mtu 9100
   ip access-group IPV4_EDGE in
   ip address 2.2.2.12/26 tag 54321

 interface Vlan101
+ no shutdown
+ no ip redirects
+ ip address 1.1.2.1/24 secondary
+ ip address 1.1.1.1/24

 interface Vlan102
   description Vlan102
   no shutdown
   mtu 9100
switch#
```



第 37 章

安全に消去するを実行します

- 安全に消去する（Secure Erase）機能に関する情報 （693 ページ）
- 安全な消去を実行するための前提条件 （694 ページ）
- 安全な消去の注意事項と制約事項 （694 ページ）
- 安全な消去の設定 （694 ページ）

安全に消去する（Secure Erase）機能に関する情報

Cisco NX-OS リリース 10.2(2)F 以降、Nexus 9000 スイッチのすべての顧客情報を消去する 安全に消去する（Secure Erase）機能が導入されました。Secure Erase は、Return Merchandise Authorization（RMA）、アップグレードまたは交換、またはシステムのサポート終了により製品が削除された状態で、Cisco NX-OS デバイス上のすべての識別可能な顧客情報を削除する操作です。

Cisco Nexus 9000 スイッチは、ストレージを消費して、システム ソフトウェア イメージ、スイッチ設定、ソフトウェア ログ、および動作履歴を保存します。これらの領域には、ネットワーク アーキテクチャや設計に関する詳細などの顧客固有の情報や、データ盗難の潜在的な標的が含まれている可能性があります。

安全に消去するプロセスは、次の 2 つのシナリオで使用されます。

- デバイスの返品許可（RMA）：RMA のためにデバイスをシスコに返送する必要がある場合は、そのデバイスの RMA 証明書を取得する前に、お客様固有のデータをすべて削除してください。
- 侵害を受けたデバイスのリカバリ：デバイスに保存されているキーマテリアルまたはクレデンシャルが侵害を受けた場合は、デバイスを初期設定にリセットし、デバイスを再設定してください。

（注） 安全に消去する機能では、外部ストレージのコンテンツは消去されません。

デバイスがリロードされて工場出荷時設定にリセットされ、EoR シャーシ モジュールがパワーダウン モードになります。工場出荷時設定にリセットすると、デバイスはすべての構成、ログ、およびストレージ情報を消去します。

安全な消去を実行するための前提条件

- 安全な消去操作を実行する前に、すべてのソフトウェアイメージ、構成、および個人データがバックアップされていることを確認してください。
- 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てください。
- 安全な消去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。

安全な消去の注意事項と制約事項

- FX3 または FX3S または FX3P スイッチは、TOR および FEX モードでサポートされます。安全な消去が FEX モードで実行された場合、スイッチは安全な消去操作後に TOR モードで起動します。
- 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。
- 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。

  トップ オブ ラック スイッチとスーパーバイザ モジュールは、ローダー プロンプトに戻ります。

  行端スイッチ モジュールは、電源が切断された状態になります。

  fex の安全な消去を構成すると、出荷時設定へのリセットが開始され、fex 構成が削除されます。

  fex コンソールを使用してモニタリングされる fex 安全な消去。失敗した場合は、再起動して fex を起動し、安全な消去を再度開始します。
- Cisco NX-OS リリース 10.4(1)F 以降、安全な消去は N9K-C9332D-H2R スイッチでサポートされています。

安全な消去の設定

RMA に発送する前に必要なデータをすべて削除するには、次のコマンドを使用して安全な消去を設定します。

コマンド	目的
factory-reset module*mod* 例： `switch(config)# factory-reset [module <3>]`	**all** オプションを有効にしてコマンドを使用してください。**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。 fex の消去を保護するには、**factory-resetfex** [*allfex_no*] を使用します。 • 一度にすべての fex を安全に消去するには、オプション **all** を使用します。 （注） 安全な消去操作を開始する前に、fex が Active-Active シナリオにないことを確認してください。 オプション **mod** を使用して、起動構成をリセットします。 • top-of-rack（ToR; トップオブラック）スイッチの場合、コマンドは **factory-reset** または **factory-reset module 1** です。 • トップ オブ ラック スイッチの LXC モードでは、コマンドは **factory-reset module 1** または **27** です。 • 行末のモジュール スイッチの場合、コマンドは [{**module** <*module*> [**bypass-secure-erase**] [**preserve-image**]}] です。 Cisco NX-OS リリース 10.2(3) 以降、factory-reset コマンドで次のオプションがサポートされています。 • **bypass-secure-erase**：このオプションは、安全なデータ削除が必要ない場合に使用します（ストレージの再パーティションと再フォーマットのみ）。 **preserve-image**：このオプションは、実行中のイメージを保持し、消去操作の完了後に自動起動します。 工場出荷時の状態へのリセットプロセスが正常に完了すると、スイッチがリブートして、電源が切れます。

（注） 並行の安全な消去操作はサポートされていません。単一の EoR シャーシ内の複数のモジュールを消去する場合、推奨される順序は、ライン カード、ファブリック、スタンバイ スーパーバイザ、システム コントローラ、アクティブ スーパーバイザです。

その安全な消去イメージを起動して、データ ワイプをトリガーできます。

次に、安全な消去による工場出荷時リセット コマンドを設定するための出力例を示します。

```
FX2-2- switch#
FX2-2- switch# show fex
FEX              FEX              FEX              FEX
Number           Description      State            Model
Serial
------------------------------------------------------------------------------
109              FEX0109          Online           N2K-C2348TQ-10GE
FOC1816R0F2
110              FEX0110          Online           N2K-C2348TQ-10G-E
FOC2003R1SQ

FX2-2-switch# factory-reset fex all
!!!! WARNING:
This command will perform factory-reset of all FEX modules !!!!
The factory reset operation will erase ALL persistent storage on the specified FEX module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable. Please, proceed
with caution and understanding that this operation cannot be undone and will leave the
system in a fresh-from-factory state.
!!!! WARNING !!!!

Do you want to continue? (y/n) [n] y
Initiating factory-reset for the FEX: 109 --- SUCCESS!!
FEX: 109 is reloading for the reset operation to proceed.
Factory reset may take time...
Please, wait and do not power off the FEX...
Trying to remove the FEX:109 config !!!
Initiating factory-reset for the FEX: 110 --- SUCCESS!!
FEX: 110 is reloading for the reset operation to proceed.
Factory reset may take time...
Please, wait and do not power off the FEX...
Trying to remove the FEX:110 config !!!
Successfully removed FEX:110 config. !!!
```

以下に fex ログの例を示します。

```
FX2-2-switch# 2021
FEX console logs:
================
bgl-ads-4157:138> telnet 10.127.118.15 2007
Trying 10.127.118.15...
Connected to 10.127.118.15.
Escape character is '^]'.

fex-109#
fex-109# [129266.313614] writing reset reason 9, Factory-reset requested by abc
[129266.391801] Restarting system - Factory-reset requested by abc [9]
U-Boot 2011.12 (Jun 25 2014 - 16:28:41) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
```

```
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Golden image
U-boot retry count 0
Jump to upgradeable image at 0xefd20040
U-Boot 2011.12 (Jun 25 2014 - 16:19:54) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Upgradeable image
DRAM: Configuring DDR for 666.667 MT/s data rate
Time-out count = 480
DDR configuration get done
1 GiB (DDR3, 32-bit, CL=6, ECC on)
Memory test from 0x40000 to 0x1fdfffff
Data line test..... OK
Address line test..... OK
OK
Flash: 288 MiB
L2: 256 KB enabled
Set dbglevel to its default value (0x1)
PCIe1: Root Complex of mini PCIe SLOT, x1, regs @ 0xffe0a000
PCIe1: Bus 00 - 01
PCIe2: Root Complex of PCIe SLOT, no link, regs @ 0xffe09000
PCIe2: Bus 02 - 02
Net: eTSEC1, eTSEC3
Hit Ctrl-L to stop autoboot: 0
WARN: user forced bootcmd="run sysboot"
.. WARNING: adjusting available memory to 30000000
## Booting kernel from Legacy Image at 01000000 ...
Image Name: Linux-2.6.27.47
Created: 2015-11-20 10:22:39 UTC
Image Type: PowerPC Linux Kernel Image (gzip compressed)
Data Size: 8936305 Bytes = 8.5 MiB
Load Address: 00000000
Entry Point: 00000000
Verifying Checksum ... OK
## Flattened Device Tree blob at 00c00000
Booting using the fdt blob at 0x00c00000
Uncompressing Kernel Image ... OK
Loading Device Tree to 03ffb000, end 03fffe82 ... OK
setup_arch: bootmem
mpc85xx_fex_setup_arch()
arch: exit
[0.436112] Host controller irq 17
[0.477490] pci 0000:00:00.0: ignoring class b20 (doesn't match header type 01)
[0.566841] Assign root port irq 17 for 0000:00:00.0
[2.210329] Enabling all PCI devices
[2.802226] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
```

```
[2.975494] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
[3.889037]
[3.889041] Watchdog init<0>
Mount failed for selinuxfs on /selinux: No such file or directory
INIT: version 2.86 booting
Setting system clock: [ OK ]
Mounting all filesystems: [ OK ]
/sbin/dhclient-script: configuration for eth1 not found. Continuing with defaults.
/etc/sysconfig/network-scripts/network-functions: line 78: eth1: No such file or directory
Mounting system image: [ OK ]
Unpacking system image: [ OK ]
Uncompressing system image: [ OK ]
Loading system image: [ OK ]
net.ipv4.ip_forward = 0
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_no_pmtu_disc = 1
Starting internet superserver: inetd [ OK ]
net.core.rmem_max = 524288
net.core.wmem_max = 524288
net.core.rmem_default = 524288
net.core.wmem_default = 524288
net.core.somaxconn = 1024
net.core.netdev_max_backlog = 1024
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
[23.255118] Device eth0 configured with sgmii interface
Non issu restart
[24.151321]
[24.151327] base_addr is 26524<0>
Secure erase requested! Please, do not power off module!
Starting the secure erase. !!
This may take time. Please wait !!
>>>> Wiping all storage devices ...
[28.706882] NX-OS starts punching watchdog
grep: Backu: No such file or directory
+++ Starting mtd secure erase for the partition /dev/mtd2 +++
Erasing /dev/mtd2 ...
Erasing 128 Kibyte @ 17e0000 -- 99 % complete.
---> SUCCESS
Writing random data onto /dev/mtd2
Filling /dev/mtd2 using random data ...
Erasing blocks: 192/192 (100%)
Writing data: 24576k/24576k (100%)
Verifying data: 24576k/24576k (100%)
---> SUCCESS
Erasing /dev/mtd2 ...
Erasing 128 Kibyte @ 17e0000 -- 99 % complete.
---> SUCCESS
+++ Skipping cmos secure erase +++
>>>> Done
```

```
+++ Skipping nvram secure erase +++
>>>> Done
>>>> Iniatilzing system to factory defaults ...
+++ Starting init-system +++
Initializing /dev/mtd5
/isan/bin/mount_jffs2.sh: line 68: ${LOG_FILE}: ambiguous [ 651.954326] Restarting system.
U-Boot 2011.12 (Jun 25 2014 - 16:28:41) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Golden image
U-boot retry count 1
Jump to upgradeable image at 0xefd20040
U-Boot 2011.12 (Jun 25 2014 - 16:19:54) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Upgradeable image
DRAM: Configuring DDR for 666.667 MT/s data rate
Time-out count = 480
DDR configuration get done
1 GiB (DDR3, 32-bit, CL=6, ECC on)
Memory test from 0x40000 to 0x1fdfffff
Data line test.................. OK
Address line test.............. OK
OK
Flash: 288 MiB
L2: 256 KB enabled
Set dbglevel to its default value (0x1)
PCIe1: Root Complex of mini PCIe SLOT, x1, regs @ 0xffe0a000
PCIe1: Bus 00 - 01
PCIe2: Root Complex of PCIe SLOT, no link, regs @ 0xffe09000
PCIe2: Bus 02 - 02
Net: eTSEC1, eTSEC3
Hit Ctrl-L to stop autoboot: 0
WARN: user forced bootcmd="run sysboot"
.. WARNING: adjusting available memory to 30000000
## Booting kernel from Legacy Image at 01000000 ...
Image Name: Linux-2.6.27.47
Created: 2015-11-20 10:22:39 UTC
Image Type: PowerPC Linux Kernel Image (gzip compressed)
Data Size: 8936305 Bytes = 8.5 MiB
Load Address: 00000000
Entry Point: 00000000
Verifying Checksum ... OK
## Flattened Device Tree blob at 00c00000
Booting using the fdt blob at 0x00c00000
Uncompressing Kernel Image ... OK
```

```
Loading Device Tree to 03ffb000, end 03fffe82 ... OK
setup_arch: bootmem
mpc85xx_fex_setup_arch()
arch: exit
[ 0.436112] Host controller irq 17
[ 0.477490] pci 0000:00:00.0: ignoring class b20 (doesn't match header type 01)
[ 0.566841] Assign root port irq 17 for 0000:00:00.0
[ 2.210556] Enabling all PCI devices
[ 2.804559] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
[ 2.975502] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
[ 3.889014]
[ 3.889018] Watchdog init<0>
Mount failed for selinuxfs on /selinux: No such file or directory
INIT: version 2.86 booting
Setting system clock: [ OK ]
Mounting all filesystems: [ OK ]
/sbin/dhclient-script: configuration for eth1 not found. Continuing with defaults.
/etc/sysconfig/network-scripts/network-functions: line 78: eth1: No such file or directory
Mounting system image: [ OK ]
Unpacking system image: [ OK ]
Uncompressing system image: [ OK ]
Loading system image: [ OK ]
net.ipv4.ip_forward = 0
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_no_pmtu_disc = 1
Starting internet superserver: inetd [ OK ]
net.core.rmem_max = 524288
net.core.wmem_max = 524288
net.core.rmem_default = 524288
net.core.wmem_default = 524288
net.core.somaxconn = 1024
net.core.netdev_max_backlog = 1024
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or
directory
[ 22.630994] Device eth0 configured with sgmii interface
Non issu restart
[ 23.535827]
[ 23.535832] base_addr is 26524<0>
INIT: Entering runlevel: 3
fex login: Sorry, user root is not allowed to execute '/sbin/sysctl -q -w vm.drop_caches=3'
 as root on fex.
[ 28.090052] NX-OS starts punching watchdog
fex login:
```

次に、モジュールで安全な消去による工場出荷時リセットコマンドを設定するための出力例を示します。

```
switch# factory-reset [all | module <mod>]
switch# factory-reset [module <3>]
!!!! WARNING !!!!
The factory reset operation will erase ALL persistent storage on the specified module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken to render data non-recoverable. Please, proceed with caution and
 understanding that this operation cannot be undone and will leave the system in a
fresh-from-factory state.
!!!! WARNING !!!!
Continue? (y/n) [n] y
A module reload is required for the reset operation to proceed. Please, wait...
…truncated…
Secure erase requested! Please, do not power off module!
>>>> Wiping all storage devices ...
+++ Starting mmc secure erase for /dev/mmcblk0 +++
*** Please, wait - this may take several minutes ***
\
---> SUCCESS
+++ Starting SSD secure erase for /dev/sda +++
*** Please, wait - this may take several minutes ***
\
---> SUCCESS
+++ Starting cmos secure erase +++
\
---> SUCCESS
>>>> Done
+++ Starting nvram secure erase +++
\
---> SUCCESS
>>>> Done
```

次に、LC で安全な消去による工場出荷時リセット コマンドを設定するための出力ログの例を示します。

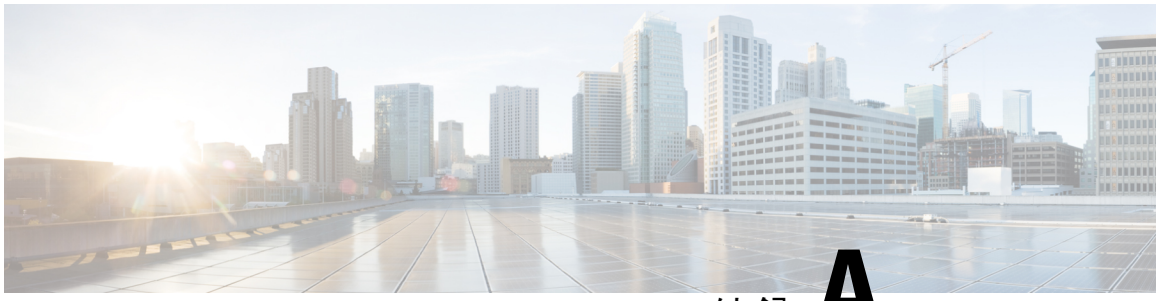
```
switch# show mod
Mod    Ports        Module-Type                         Model         Status
----   ------     -------------------------------  -------------  ---------
1        32          32x40/100G Ethernet Module    N9K-X9732C-FX    ok
22        0           4-slot Fabric Module         N9K-C9504-FM-E   ok
24        0           4-slot Fabric Module         N9K-C9504-FM-E   ok
26        0           4-slot Fabric Module         N9K-C9504-FM-E   ok
27        0           Supervisor Module            N9K-SUP-B+       active *
28        0           Supervisor Module            N9K-SUP-B+       ha-standby
29        0           System Controller            N9K-SC-          active
30        0           System Controller            N9K-SC-          standby

Mod        Sw              Hw         Slot
----------------------------------------
1         10.2(1.196)     0.1070     LC1
22        10.2(1.196)     1.2        FM2
24        10.2(1.196)     1.2        FM4
26        10.2(1.196)     1.1        FM6
27        10.2(1.196)     1.0        SUP1
28        10.2(1.196)     1.2        SUP2
29        10.2(1.196)     1.4        SC1
30        10.2(1.196)     1.4        SC2

switch#
switch# factory-reset mod 1
!!!! WARNING !!!!
The factory reset operation will erase ALL persistent storage on the specified module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable.
Please, proceed with
```

```
caution and understanding that this operation cannot be undone and will leave the system
 in
a fresh-from-factory state.
!!!! WARNING !!!!
Continue? (y/n) [n] y
A module reload is required for the reset operation to proceed. Please, wait...
reloading module 1 ...
.................................................................
SUCCESS! All persistent storage devices detected on the specified module have been purged.

switch#
switch# show mod
Mod            Ports      Module-Type                        Model              Status
-------      ---------   ----------------------------    ---------------    ------------
1              32        32x40/100G Ethernet Module    N9K-X9732C-FX       powered-dn
22              0            4-slot Fabric Module      N9K-C9504-FM-E        ok
24              0            4-slot Fabric Module      N9K-C9504-FM-E        ok
26              0            4-slot Fabric Module      N9K-C9504-FM-E        ok
27              0            Supervisor Module         N9K-SUP-B+            active *
28              0           Supervisor Module         N9K-SUP-B+            ha-standby
29              0            System Controller         N9K-SC-A              active
30              0            System Controller         N9K-SC-A              standby

Mod     Power-Status         Reason
------  ----------------    ------------
1         powered-dn        Configured Power down

Mod     Sw                 Hw         Slot
22      10.2(1.196)        1.2        FM2
24      10.2(1.196)        1.2        FM4
26      10.2(1.196)        1.1        FM6
27      10.2(1.196)        1.0        SUP1
28      10.2(1.196)        1.2        SUP2
29      10.2(1.196)        1.4        SC1
switch#
```

次に、modでの安全な消去による工場出荷時リセットコマンドを設定した場合の出力ログの例を示します。

```
switch# factory-reset mod 26
!!!! WARNING !!!!
The factory reset operation will erase ALL persistent storage on the specified module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable.
Please, proceed with
caution and understanding that this operation cannot be undone and will leave the system
 in
a fresh-from-factory state.
!!!! WARNING !!!!
Continue? (y/n) [n] y
A module reload is required for the reset operation to proceed. Please, wait...
reloading module 26 ...
.............................................................................
..........................
.............................................................................
..........................
.............................................................................
..........................
.............................................................................
..........................
.............................................................................
..........................
.............................................................................
..........................
```

```
.....................................................................
SUCCESS! All persistent storage devices detected on the specified module have been
cleared.
>>>> Please, note - multiple write passes were required to remove data from one or more
 devices. <<<<

switch# show mod

Mod       Ports      Module-Type                  Model             Status
------  --------    ---------------------------  -------------    ------------
1         32         32x40/100G Ethernet Module  N9K-X9732C-FX     powered-dn
22         0         4-slot Fabric Module        N9K-C9504-FM-E     ok
24         0         4-slot Fabric Module        N9K-C9504-FM-E     ok
26         0         4-slot Fabric Module        N9K-C9504-FM-E    powered-dn

Mod         Power-Status       Reason
------     ---------------    ------------------------
1            powered-dn        Configured Power down
26           powered-dn        Configured Power down

Mod    Sw            Hw            Slot
----  ------        ---------    -----------
22    10.2(1.196)    1.2            FM2
24    10.2(1.196)    1.2            FM4
27    10.2(1.196)    1.0            SUP1
28    10.2(1.196)    1.2            SUP2
29    10.2(1.196)    1.4            SC1
switch#
```



付録 A

Cisco NX-OS システム管理でサポートされている IETF RFC

Cisco NX-OS でサポートされているシステム管理に関する IETF RFC は次のとおりです。

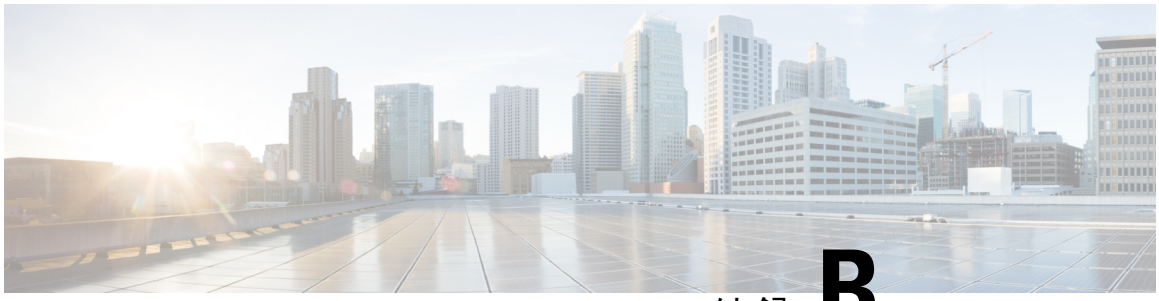
• Cisco NX-OS システム管理でサポートされている IETF RFC （705 ページ）

Cisco NX-OS システム管理でサポートされている IETF RFC

Cisco NX-OS でサポートされているシステム管理に関する IETF RFC は次のとおりです。

RFC	タイトル
RFC 2819	『*Remote Network Monitoring Management Informa*
RFC 3411 および RFC 3418	『*An Architecture for Describing Simple Network Man (SNMP) Management Frameworks*』

Cisco NX-OS システム管理でサポートされている IETF RFC



付録 B

Embedded Event Manager システム イベントおよび設定例

この付録では、Embedded Event Manager（EEM）システム ポリシー、イベント、およびポリシーのコンフィギュレーション例について説明します。

この付録は、次の項で構成されています。

- EEM システム ポリシー （707 ページ）
- EEM イベント （711 ページ）
- EEM ポリシーの設定例 （712 ページ）

EEM システム ポリシー

次の表に、Embedded Event Manager（EEM）のシステム ポリシーを示します。

イベント	説明（Description）
__BootupPortLoopback	CallHome を実行し、影響があるポートのエラーを無効にして、GOLD "BootupPortLoopback" テストに1 回連続で失敗した場合は、その後影響を受けたポートでのエラー テストを記録します。
__PortLoopback	CallHome を実行し、Syslog、OBFL、または例外ログにエラーを記録し、GOLD "PortLoopback" テストに 10 回連続で失敗した場合は、その後影響を受けたポートでの HM テストをディセーブルにします。
__RewriteEngineLoopback	CallHome を実行し、Syslog、OBFL、または例外ログにエラーを記録し、GOLD "RewriteEngine" テストに 10 回連続で失敗した場合は、その後影響を受けたポートでの HM テストをディセーブルにします。

Embedded Event Manager システム イベントおよび設定例

イベント	説明（**Description**）
__asicmem	GOLD "AsicMemory" テストに失敗した場合には、CallHome を実行し、エラーを記録します。テストの失敗の原因となる問題は一時的なものである可能性があるため、カーネルパニックによるリカバリ リロードを試行します。 （注） テストが失敗したときにカーネルパニックを回避するには、EEM システム ポリシーを上書します。
__asic_register_check	CallHome を実行し、エラーを記録し、GOLD "ASICRegisterCheck" テストに 20 回連続で失敗した場合は、その後その ASIC デバイスおよびインスタンスの HM テストをディセーブルにします。
__compact_flash	CallHome を実行し、エラーを記録し、GOLD "CompactFlash" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__crypto_device	CallHome を実行し、GOLD "CryptoDevice" テストに失敗するとエラーを記録します。
__eobc_port_loopback	CallHome を実行し、GOLD "EOBCPortLoopback" テストに失敗するとエラーを記録します。
__ethpm_debug_1	アクション：なし
__ethpm_debug_2	アクション：なし
__ethpm_debug_3	アクション：なし
__ethpm_debug_4	アクション：なし
__ethpm_link_flap	420 秒間隔でリンク フラップが 30 を超えています。アクション：エラー。ポートをディセーブルにします。
__external_compact_flash	CallHome を実行し、エラーを記録し、GOLD "ExternalCompactFlash" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。

Embedded Event Manager システム イベントおよび設定例

イベント	説明（Description）
__fpgareg	GOLD "FpgaRegTest" テストに 20 回連続で失敗した場合は、CallHome を実行し、エラーを記録し、その後 HM テストをディセーブルにします。テストの失敗の原因となる問題は一時的なものである可能性があるため、カーネルパニックによるリカバリ リロードを試行します。 （注） テストが失敗したときにカーネルパニックを回避するには、EEM システム ポリシーを上書します。
__L2ACLRedirect	L2ACLRedirect テストを 10 回連続で失敗した場合は、CallHome を実行し、エラーを記録し、その後 HM テストをディセーブルにします。テストの失敗の原因となる問題は一時的なものである可能性があるため、カーネルパニックによるリカバリ リロードを試行します。 （注） テストが失敗したときにカーネルパニックを回避するには、EEM システム ポリシーを上書します。
__lcm_module_failure	2 度電源を切って入れ直し、電源を切ります。
__management_port_loopback	CallHome を実行し、GOLD "ManagementPortLoopback" テストに失敗するとエラーを記録します。
__nvram	CallHome を実行し、エラーを記録し、GOLD "NVRAM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__pfm_fanabsent_all_systemfan	両方のファン トレイ（f1 と f2）が 2 分間存在しない場合シャットダウンします。
__pfm_fanbad_all_systemfan	ファンで障害が発生した場合シスログに記録します。
__pfm_fanbad_any_singlefan	ファンで障害が発生した場合シスログに記録します。

Embedded Event Manager システム イベントおよび設定例

イベント	説明（Description）
__pfm_power_over_budget	不十分な電力超過バジェットに対するシスログ警告
__pfm_tempev_major	TempSensor メジャーしきい値アクション：シャットダウン
__pfm_tempev_minor	TempSensor マイナーしきい値アクション：シスログ
__primary_bootrom	CallHome を実行し、エラーを記録し、GOLD "PrimaryBootROM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__pwr_mgmt_bus	CallHome を実行し、エラーを記録し、GOLD "PwrMgmtBus" テストに 20 回連続で失敗した場合は、モジュールまたはスパインカードの HM テストをディセーブルにします。
__real_time_clock	CallHome を実行し、エラーを記録し、GOLD "RealTimeClock" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__secondary_bootrom	CallHome を実行し、エラーを記録し、GOLD "SecondaryBootROM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__spine_control_bus	CallHome を実行し、エラーを記録し、GOLD "SpineControlBus" テストに 20 回連続で失敗した場合は、そのモジュールまたはスパインカードの HM テストをディセーブルにします。
__standby_fabric_loopback	CallHome を実行し、エラーを記録し、10 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__status_bus	CallHome を実行し、エラーを記録し、GOLD "StatusBus" テストに 5 回連続で失敗した場合は、その後 HM テストをディセーブルにします。

イベント	説明（Description）
__system_mgmt_bus	Call Home を実行し、エラーを記録し、GOLD "SystemMgmtBus" テストに 20 回連続で失敗した場合は、そのファンまたは電源の HM テストを無効にします。
__usb	Call Home を実行し、GOLD "USB" テストに失敗するとエラーを記録します。

EEM イベント

次の表は、デバイスで使用できる EEM イベントについて説明します。

EEM イベント	説明
application	アプリケーション固有のイベントをパブリッシュします。
cli	ワイルドカードを使用したパターンを照合する CLI コマンドが入力されます。
counter	EEM カウンタが指定された値または範囲に達します。
fanabsent	システム ファン トレイがありません。
fanbad	システム ファンで障害が生成されます。
fib	ユニキャスト FIB のルートまたは TCAM の使用状況をモニタします。
Gold	GOLD テスト失敗条件がヒットします。
インターフェイス	インターフェイス カウンタがしきい値を超えます。
メモリ	使用可能なシステム メモリがしきい値を超えます。
両側面）	指定したモジュールが、選択したステータスになります。
module-failure	モジュール障害が生成されます。
なし	指定されたイベントがないポリシー イベントを実行します。

EEM イベント	説明
oir	活性挿抜が発生します。
policy-default	デフォルトのパラメータおよびしきい値が、上書きするシステム ポリシーのイベントに使用されます。
poweroverbudget	プラットフォーム ソフトウェアが電力バジェット条件を検出します。
snmp	SNMP オブジェクト ID（OID）の状態が変化します。
storm-control	プラットフォーム ソフトウェアがイーサネット パケット ストーム条件を検出します。
syslog	syslog メッセージを監視し、ポリシーの検索文字列に基づいてポリシーを呼び出します。
sysmgr	システムマネージャがイベントを生成します。
温度	システムの温度レベルがしきい値を超えます。
timer	指定された時間に到達します。
トラック	トラッキング対象オブジェクトの状態が変化します。

EEM ポリシーの設定例

CLI イベントの設定例

インターフェイス シャットダウンのモニタリング

インターフェイスのシャットダウンをモニタする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# event manager applet monitorShutdown
switch(config-applet)#
switch(config-applet)# description “Monitors interface shutdown.”
switch(config-applet)# event cli match “conf t; interface *; shutdown”
switch(config-applet)# action 1.0 cli show interface e 3/1
switch(config)# copy running-config startup-config
```

（注） EEM ポリシーの一部として入力された **show** コマンドの出力は、「eem_archive_」というプレフィックスが付加されたテキスト ファイルとして logflash にアーカイブされます。アーカイブされている出力を表示するには、**show file logflash:eem_archive_***n* コマンドを使用します。

モジュール パワーダウンのモニタリング

モジュールのパワーダウンをモニタする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# event manager applet monitorPoweroff
switch(config-applet)#
switch(config-applet)# description "Monitors module power down."
switch(config-applet)# event cli match "conf t; poweroff *"
switch(config-applet)# action 1.0 cli show module
switch(config)# copy running-config startup-config
```

ロールバックを開始するトリガーの追加

ロールバックを開始するトリガーを追加する例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)#
switch(config)# event manager applet rollbackTrigger
switch(config-applet)#
switch(config-applet)# description "Rollback trigger."
switch(config-applet)# event cli match "rollback *"
switch(config-applet)# action 1.0 cli copy running-config bootflash:last_config
switch(config)# copy running-config startup-config
```

メジャーしきい値を上書き（無効化）する設定例

メジャーしきい値に達したときにシャットダウンを防ぐ方法

メジャーしきい値に達したことによるシャットダウンを防ぐ例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

One Bad センサーの無効化

センサー 3 で障害が発生した場合（他のセンサーに影響なし）に、モジュール 2 でセンサー 3 だけをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

複数の不良センサーを無効にする方法

モジュール 2 のセンサー 5、6、7 で障害が発生した場合（他のセンサーに影響なし）に、これらのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 5 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 6 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

モジュール全体の上書き（無効化）

誤動作するモジュール 2 をディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 threshold major
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

複数のモジュールおよびセンサーの上書き（無効）

誤動作するモジュール 2 のセンサー 3、4、7 とモジュール 3 のすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 4 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 3 threshold major
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

1 つのセンサーを有効にして、すべてのモジュールの残りのセンサーをすべて無効にする方法

モジュール 9 のセンサー 4 を除くすべてのモジュールのすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 4 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

複数のセンサーを有効にして、すべてのモジュールの残りのセンサーをすべて無効にする方法

モジュール 9 のセンサー 4、6、7 を除くすべてのモジュールのすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 4 threshold major
```

```
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 6 threshold major
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 7 threshold major
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
```

1 つのモジュールのすべてのセンサーを有効にして、残りのモジュールのすべてのセンサーを無効にする方法

モジュール 9 のすべてのセンサーを除く残りのモジュールのすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

モジュールのセンサーを組み合わせて有効にして、残りのモジュールのすべてのセンサーを無効にする方法

モジュール 2 の センサー 3、4、7 とモジュール 3 のすべてのセンサーを除くすべてのモジュールのすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 4 threshold major
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet5 override __pfm_tempev_major
switch(config-applet)# event temperature module 3 threshold major
switch(config-applet)# action 5 policy-default
switch(config-applet)# end
```

ファントレイ取り外しのためのシャットダウンを上書き（無効化）するコンフィギュレーション例

1 つまたは複数のファン トレイ取り外しのためのシャットダウンの上書き（無効）

1つまたは複数（またはすべて）のファン トレイを取り外せるように、シャットダウンを無効にする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
```

指定したファン トレイを取り外すためのシャットダウンの上書き（無効）

指定したファントレイ（ファントレイ3）を取り外せるように、シャットダウンを無効にする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config) no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config)# end
```

指定した複数のファン トレイを取り外すためのシャットダウンの上書き（無効化）

指定した複数のファントレイ（ファン トレイ2、3、4）を取り外せるように、シャットダウンを無効にする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 4 time 60
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config)# end
```

1 つを除くすべてのファンを取り外すためのシャットダウンの上書き（無効）

1 つ（ファン トレイ 2）を除くすべてのファン トレイを取り外せるように、シャットダウンを無効にする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

ファン トレイの指定したセットを除くファン トレイを取り外すためのシャットダウンの上書き（無効）

指定したファン トレイのセット（ファン トレイ 2、3、4）を除くファンを取り外せるように、シャットダウンを無効にする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2,3,4 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

ファン トレイのセットから 1 台を除くすべてのファン トレイを取り外すためのシャットダウンの上書き（無効）

指定したファン トレイのセット（ファン トレイ 2、3、4）の 1 台を除くすべてのファン トレイを取り外せるように、シャットダウンを無効にする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 4 time 60
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
```

補足ポリシーを作成するコンフィギュレーション例

ファン トレイが存在しないイベントの補足ポリシーの作成

event fanabsent コマンドを使用して、補足ポリシーを作成する例を示します。

[**no**] **event fanabsent** [**fan** *fan-tray-number*] **time** *time-interval*

ファン トレイ 1 が 60 秒間存在しない場合に、デフォルトのポリシーに加えて、ポリシー myappletname とアクション 3 を実行する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname
switch(config-applet)# event fanabsent fan 1 time 60
switch(config-applet)# action 3 cli "show env fan"
switch(config-applet)# end
```

温度しきい値イベントの補足ポリシーの作成

event temperature コマンドを使用して、補足ポリシーを作成する例を示します。

[**no**] **event temperature** [**mod** *module-number*] [**sensor** *sensor-number*] **threshold** {**major** | **minor** | **any**}

モジュール 2 のセンサー 3 で温度がマイナーしきい値を超えた場合に、デフォルトのポリシーに加えて、ポリシー myappletname とアクション 1 を実行する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname
switch(config-applet)# event temperature module 2 sensor 3 threshold minor
switch(config-applet)# action 1 cli "show environ temperature"
switch(config-applet)# end
```

電力のバジェット超過ポリシーの設定例

電力のバジェット超過ポリシーは、使用可能な電力がゼロ未満に低下し、前に起動されたモジュールを起動状態で維持できなくなった場合に開始します。デフォルトのアクションでは、ユーザに電力のバジェット超過が発生したことを通知する syslog を出力します。

利用可能な電力が赤（負）のゾーンから回復するまでモジュールの電源を落とす追加アクションをイネーブルにできます。

モジュールのシャットダウン

モジュールを指定しない場合、電力のバジェット超過シャットダウンはスロット 1 から始まり、電力が赤（負）のゾーンから回復するまでモジュールをシャットダウンします。空のスロットやスーパーバイザ、スタンバイ スーパーバイザ、スパイン、クロスバーを含むスロットは飛ばされます。

利用可能な電力がゼロ未満に低下した場合に、モジュール 1 からモジュールをシャットダウンする例を示します。

```
switch# configure terminal
switch(config)# event manager applet <myappletname4a> override __pfm_power_over_budget
```

```
switch(config-applet)# event poweroverbudget
switch(config-applet)# action 4 overbudgetshut
switch(config-applet)# end
```

指定された一連のモジュールのシャットダウン

電力のバジェット超過アクションによって、電力が赤（負）のゾーンから回復するまでシャットダウンされるモジュールのリストを指定できます。空のスロットやスーパーバイザ、スタンバイ スーパーバイザ、スパイン、クロスバーを含むスロットは飛ばされます。

利用可能な電力がゼロ未満に低下した場合に、指定されたモジュールのリスト（1、2、7、8）からモジュールをシャットダウンする例を示します。

```
switch# configure terminal
switch(config)# event manager applet <myappletname4b> override __pfm_power_over_budget
switch(config-applet)# event poweroverbudget
switch(config-applet)# action 5 overbudgetshut module 1,2,7,8
switch(config-applet)# end
```

シャットダウンするモジュールを選択する設定例

デフォルトでシャットダウンに非上書きモジュールを選択するポリシーの使用

メジャーしきい値を超えた場合に、デフォルトで非上書きモジュールをシャットダウンするよう選択するポリシーを使用する例を示します。

```
switch# configure terminal
switch(config)# event manager applet my5a1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet my5a2 override __pfm_tempev_major
switch(config-applet)# event temperature module 1-3 sensor 4 threshold major
switch(config-applet)# action 5 policy-default
switch(config-applet)# end
```

シャットダウンに非上書きモジュールを選択するパラメータ置き換えの使用

メジャーしきい値を超えた場合に、パラメータの置き換えを使用してシャットダウンする非上書きモジュールを選択する例を示します。

```
switch# configure terminal
switch(config)# event manager applet my5b1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet my5b2 override __pfm_tempev_major
switch(config-applet)# event temperature module 1-3 sensor 8 threshold major
switch(config-applet)# action 6 forceshut module my_module_list reset "temperature-sensor
 policy trigger"
switch(config-applet)# end
```

イベント マネージャ パラメータを作成するには、**event manager environment** コマンドを使用します。イベント マネージャ パラメータの値を表示するには、**show event manager environment all** コマンドを使用します。

活性挿抜イベントのコンフィギュレーション例

活性挿抜イベント（OIR）には、デフォルトのポリシーがありません。

event oir コマンドを使用して、OIR イベントを設定する例を示します。

event oir *device-type event-type* [*device-number*]+

device-type は、**fan**、**module** または **powersupply** です。

event-type は、**insert**、**remove**、または **anyoir**（装着または取り外し）です。

オプションの *device-number* では 1 台のデバイスを指定します。省略すると、すべてのデバイスが選択されます。

装着イベントを設定する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module insert
switch(config-applet)# action 1 syslog priority critical msg "OIR insert event: A Module
 is inserted"
```

取り外しイベントを設定する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module remove
switch(config-applet)# action 1 syslog priority critical msg "OIR remove event: A Module
 is removed"
```

ユーザ syslog を生成するコンフィギュレーション例

action syslog コマンドを使用して、ユーザ syslog を生成する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module remove
switch(config-applet)# action 1 syslog priority critical msg "Module is removed"
```

このイベントが発生すると、次の syslog が生成されます。

```
switch(config)# 2013 May 20 00:08:27 p1b-57 %$ VDC-1 %$ %EEM_ACTION-2-CRIT: "Module is
removed"
```

Syslog メッセージをモニタする設定例

次に、スイッチからの Syslog メッセージをモニタする例を示します。

```
switch(config)# event manager applet a1
switch(config-applet)# event syslog occurs 6 period 4294967 pattern "authentication
failed"
```

このイベントがトリガーされると、ポリシーで定義されているアクションが実行されます。

SNMP 通知の設定例

SNMP OID のポーリングによる EEM イベントの生成

スイッチの CPU 使用率を問い合わせるには、SNMP オブジェクト ID（OID）CISCO-SYSTEM-EXT-MIB::cseSysCPUUtilization が使用されます。

```
cseSysCPUUtilization OBJECT-TYPE
SYNTAX Gauge32 (0..100 )
UNITS "%"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The average utilization of CPU on the active supervisor."
::= { ciscoSysInfoGroup 1 }
```

10 秒間隔でポーリングされ、しきい値が 95 % の SNMP ODI を使用する例を示します。

```
switch# configure terminal
switch(config)# event manager applet test_policy
switch(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.305.1.1.1.0 get-type exact entry-op
 gt entry-val 95 exit-op lt exit-val 90 poll-interval 10
```

イベント ポリシーのイベントへの応答で SNMP 通知を送信

このタイプのコンフィギュレーションを使用して、重大なイベント トリガーで SNMP 通知を生成できます。

イベントマネージャのアプレットコンフィギュレーションモードからイベントに対してSNMP通知を送信する例を示します。

```
switch(config-applet)# action 1.1 snmp-trap intdata1 100 intdata2 300 strdata "CPU Hogging
 at switch1"
switch(config-applet)# action 1.1 snmp-trap intdata1 100 intdata2 300 strdata "Port
Failure eth9/1"
```

このコンフィギュレーションでは、スイッチから SNMP ホストに SNMP 通知（トラップ）を行います。SNMP ペイロードには、ユーザ定義フィールド intdata1、intdata2、および strdata の値が含まれます。

ポート トラッキングの設定例

1 つのポートの状態を別のポートの状態と一致させるように設定する例を示します（ポート トラッキング）。

イーサネット インターフェイス 1/2 によるイーサネット インターフェイス 3/23 のポート トラッキングを設定するには、次のステップに従います。

手順の概要

1. イーサネットインターフェイス 3/23 のステータスを追跡するオブジェクトを作成します。

2. トラッキング オブジェクトがシャットダウンされたらイーサネット インターフェイス 1/2 をシャットダウンする EEM イベントを設定します。
3. イーサネット インターフェイス 3/23 が起動したらイーサネット インターフェイス 1/2 を起動する EEM イベントを設定します。

手順の詳細

手順

ステップ 1 イーサネット インターフェイス 3/23 のステータスを追跡するオブジェクトを作成します。

例：

```
switch# configure terminal
switch(config)# track 1 interface ethernet 3/23
switch(config-track)# end
```

ステップ 2 トラッキング オブジェクトがシャットダウンされたらイーサネット インターフェイス 1/2 をシャットダウンする EEM イベントを設定します。

例：

```
switch(config)# event manager applet track_3_23_down
switch(config-applet)# event track 1 state down
switch(config-applet)# action 1 syslog msg EEM applet track_3_23_down shutting down port eth1/2 due
 to eth3/23 being down
switch(config-applet)# action 2 cli conf term
switch(config-applet)# action 3 cli interface ethernet 1/2
switch(config-applet)# action 4 cli shut
switch(config-applet)# end
```

ステップ 3 イーサネット インターフェイス 3/23 が起動したらイーサネット インターフェイス 1/2 を起動する EEM イベントを設定します。

例：

```
switch# configure terminal
switch(config)# event manager applet track_3_23_up
switch(config-applet)# event track 1 state up
switch(config-applet)# action 1 syslog msg EEM applet track_3_23_down bringing up port eth1/2 due
to eth3/23 being up
switch(config-applet)# action 2 cli conf term
switch(config-applet)# action 3 cli interface ethernet 1/2
switch(config-applet)# action 4 cli no shut
switch(config-applet)# end
```

EEM によって EEM ポリシーを登録する設定例

次に、EEM によって EEM ポリシーを登録する例を示します。

Embedded Event Manager システム イベントおよび設定例

基本的なスイッチ設定：

```
event manager applet vpc_check_peer_at_startup
event track 101 state up
action 1.0 cli copy bootflash:eem/user_script_policies/load_schedules running-config

feature scheduler

!!## 2 x dummy loopbacks are required ##!!
interface loopback 101
interface loopback 102

track 1 list boolean or
object 13
object 12
object 102
track 2 list boolean and
object 13
object 12
track 12 interface Ethernet 2/24 line-protocol
track 13 interface port-channel 3000 line-protocol
track 101 interface loopback 101 line-protocol
track 102 interface loopback 102 line-protocol
```

（注） この例では、ポート チャネル 3000 が vPC ピア リンクで、イーサネット 2/24 が vPC キープアライブ リンクです。

ブートフラッシュに次のファイルをコピーする必要があります。

- スーパーバイザのブートフラッシュに作成する必要がある、/eem/user_script_policies と呼ばれるディレクトリ。
- 次の 5 つのファイルを上記のディレクトリに作成してロードする必要があります。
  - load_schedules
  - remove_vpc_if_peer_failed
  - clean_up
  - unload_schedules
  - restore_vpc

load_schedules ファイルの設定

```
feature scheduler

configure terminal
scheduler job name vpc_check
configure terminal
event manager policy remove_vpc_if_peer_failed
end

configure terminal
scheduler job name clean_up
configure terminal
event manager policy clean_up
```

Embedded Event Manager システム イベントおよび設定例

```
end

configure terminal
scheduler job name trigger
configure terminal
interface loopback 102
shutdown
no shutdown
end

configure terminal
scheduler schedule name load_vpc_check
time start +00:00:04
job name vpc_check

scheduler schedule name trigger_vpc_check
time start +00:00:05
job name trigger

scheduler schedule name load_clean_up
time start +00:00:08
job name clean_up


scheduler schedule name trigger_clean_up
time start +00:00:10
job name trigger
```

remove_vpc_if_peer_failed ファイルの設定：

```
event manager applet remove_vpc_if_peer_failed
event track 1 state down
action 1.0 cli show run vpc > bootflash://sup-active/eem/user_script_policies/vpc_saved.cfg
action 2.0 cli show run vpc >
bootflash://sup-standby/eem/user_script_policies/vpc_saved.cfg
action 3.0 cli configure terminal
action 4.0 cli no feature vpc
action 5.0 syslog msg severity alert "##### WARNING!!!! PEER SWITCH FAILED TO COME ONLINE.
 VPC CONFIG REMOVED #####"
action 6.0 cli event manager policy restore_vpc
action 7.0 cli copy bootflash:eem/user_script_policies/unload_schedules running-config
action 8.0 cli no event manager applet remove_vpc_if_peer_failed
action 9.0 cli end
```

clean_up ファイルの設定：

```
event manager applet clean_up
event track 102 state up
action 1.0 cli configure terminal
action 2.0 cli no event manager applet remove_vpc_if_peer_failed
action 3.0 cli copy bootflash:eem/user_script_policies/unload_schedules running
action 4.0 cli no event manager applet clean_up
action 5.0 end
```

unload_schedules ファイルの設定：

```
no scheduler schedule name load_vpc_check
no scheduler schedule name trigger_vpc_check
no scheduler schedule name load_clean_up
no scheduler schedule name trigger_clean_up
no scheduler job name vpc_check
no scheduler job name trigger
```

Embedded Event Manager システム イベントおよび設定例

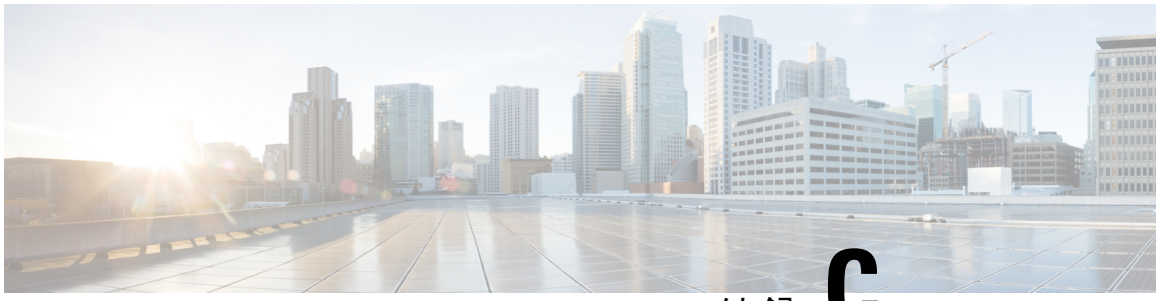
```
no scheduler job name clean_up
```

restore_vpc ファイルの設定：

```
event manager applet restore_vpc
event track 2 state up
action 1.0 cli copy bootflash:eem/user_script_policies/vpc_saved.cfg running-config
action 1.0 syslog priority alerts msg VPC PEER DETECTED. VPC CONFIG RESTORED
action 3.0 cli configure terminal
action 4.0 cli copy bootflash:eem/user_script_policies/unload_schedules running-config
action 5.0 cli no event manager applet restore_vpc
action 6.0 cli end
```

（注） severity キーワードは廃止され、次のパターンのみが許可されます。

[0-9 a-zA-Z][0-9 a-zA-Z]*[-_ ,:/0-9a-zA-Z]*



付録 C

Cisco NX-OS システム管理の設定制限

設定制限は『*Cisco Nexus 9000* シリーズ *NX-OS* 検証済みスケーラビリティ ガイド』にまとめられています。

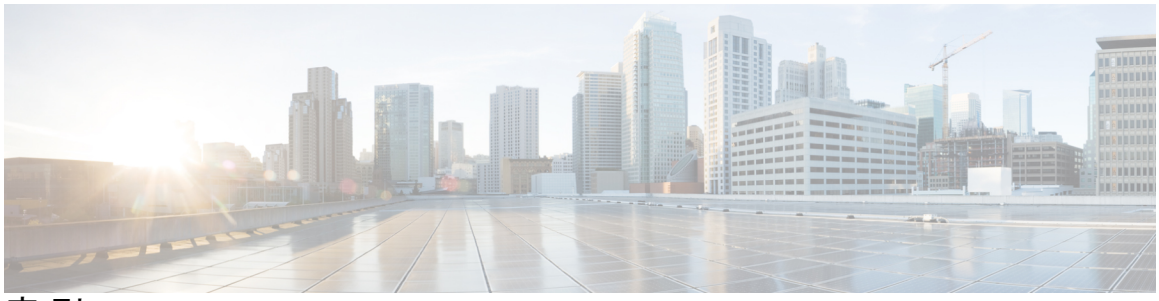
• Cisco NX-OS システム管理の設定制限 （727 ページ）

Cisco NX-OS システム管理の設定制限

Cisco NX-OS がサポートする機能には、設定の最大制限があります。一部の機能には、最大値以下の制限をサポートする設定があります。

設定制限は『*Cisco Nexus 9000* シリーズ *NX-OS* 検証済みスケーラビリティ ガイド』にまとめられています。

Cisco NX-OS システム管理の設定制限



索 引

数字

1G の ptp スケール 88

A

abort 40, 255
action 356, 363–364, 367
alert-group {Configuration | Diagnostic | EEM | Environmental | Inventory | License | Supervisor-Hardware | Syslog-group-port | System | Test} user-def-cmd 221

C

callhome 214, 216, 218–219, 221–222, 224–226, 228–230, 233
callhome send 234
callhome send configuration 234
callhome send diagnostic 234
callhome test 234
cdp advertise {v1 | v2} 176
cdp enable 174–175
cdp format device-id {mac-address | serial-number | system-name} 176
cdp holdtime 177
cdp timer 177
cfs ipv4 distribute 34
checkpoint 685
clear cdp table 177
clear checkpoint database 687
clear counters interface all 561
clear counters mpls strip 583
clear hardware rate-limiter sflow 561
clear cdp counters 177
clear lldp counters 501
clear logging logfile 199
clear logging onboard 410
clear mpls strip label dynamic 583
clear ntp session 167
clear ntp statistics 167
clear scheduler logfile 267
clear sflow statistics 561
モニタ セッションの全ての統計をクリア 444, 477
clear monitor session stats 444, 477
Clock protocol gnss 93
collect counter 516, 531
collect ip version 516, 531
collect timestamp sys-uptime 516, 531
collect transport tcp flags 516, 531
collect 514, 528–529
commit 35, 37, 40, 215–216, 218, 220–221, 223–225, 227–229, 255
設定の同期 34, 37, 39, 42
configure session 253–254
configure maintenance profile maintenance-mode 618
configure maintenance profile normal-mode 620
contract-id 214
copy ftp 643
copy sftp: 644
copy tftp 642
customer-id 214

D

destination-profile 216, 218, 220, 230
destination-profile {CiscoTAC-1 | full-txt-destination | short-txt-destination} alert-group 220
destination-profile {CiscoTAC-1 | full-txt-destination | short-txt-destination} email-addr 218
destination-profile {CiscoTAC-1 | full-txt-destination | short-txt-destination} http 218
destination-profile {CiscoTAC-1 | full-txt-destination | short-txt-destination} message-level 218
destination-profile {CiscoTAC-1 | full-txt-destination | short-txt-destination} message-size 218
destination-profile {CiscoTAC-1 | full-txt-destination | short-txt-destination} transport-method {email | http} 218
destination-profile name alert-group all 231
destination-profile 名前 email-address email-address 231
diagnostic bootup level {complete | minimal | bypass} 341
diagnostic clear result module 345
diagnostic monitor interval module 342
diagnostic monitor module 343
diagnostic ondemand action-on-failure {continue failure-count | stop} 344
diagnostic ondemand iteration 344
diagnostic test simulation 345
dir 641
dir bootflash: 645
dscp 517, 532

E

email-contact 214, 230
ERSPAN 473, 480–481
  ipv6 経由の接続先 481
    設定例 481
  宛先 480
    設定例 480
  宛先セッション 473
    ERSPAN の設定 473
  宛先セッションの設定 473
erspan ソースのモニタセッション 472
erspan-id 462
event cli 357
event counter 358
event fanabsent 358
event fanbad 358
event fib adjacency extra 358
event fib resource tcam usage 358
event fib route {extra | inconsistent | missing} 358
event manager applet 356, 367, 370
event manager environment 355
event manager policy 365
event memory {critical | minor | severe} 359
event module-failure 359
event none 359
event oir 360
event policy-default count 360
event poweroverbudget 360
event snmp 361
event storm-control 361
event syslog 361
event syslog {occurs | period | pattern | priority} 370
event syslog tag {occurs | period | pattern | priority} 370
event sysmgr memory 361
event sysmgr switchover count 361
event temperature 362
event timer 362
event track 362
exporter 518, 534

F

feature lldp 489
feature netflow 513, 527
feature ntp 157
feature ptp 88
feature scheduler 260
feature sflow 550
filter access-group 433

G

Guest Shell 同期 661
guestshell 660

H

hardware acl tap-agg 569
hardware multicast global-tx-span 439
hw-module logging onboard 407
hw-module logging onboard counter-stats 407
hw-module logging onboard cpuhog 407
hw-module logging onboard environmental-history 408
hw-module logging onboard error-stats 408
hw-module logging onboard interrupt-stats 408
hw-module logging onboard module 408
hw-module logging onboard obfl-logs 408

I

インポート 39
import running-config 39
install activate 646
install add bootflash 645
install add ftp 645
install add tftp 645
install add usb1 645
install add usb2 645
install commit 647, 649
install deactivate 648
install remove 649
ip access-list 254, 436, 471, 571
ip dscp 463
ip flow monitor 519–520, 522, 534–535
IP TTL 463
ip access-group 254
ip port access-group 573–574, 578
ipv6 flow monitor 520, 522
isolate 615

L

lldp chassis-id switch 489
lldp dcbx version 493
lldp holdtime 499
lldp receive 490, 498
lldp reinit 499
lldp timer 499
lldp tlv-select 500
lldp transmit 490, 498
logging console 183
logging event {link-status | trunk-status} {enable | default} 188
logging logfile 186
logging message interface type ethernet description 185
logging monitor 184
logging origin-id 185
logging source-interface Loopback 193
logging timestamp {microseconds | milliseconds | seconds} 191

logging server 192, 194

M

mac access-list 571
mac port access-group 573–574, 578
mac packet-classify 521
match datalink 514, 520, 530
match ip 515, 530
match ipv4 515, 530
match ipv6 515, 531
match transport 515, 531
mode tap-aggregation 575
monitor erspan origin ip-address 460
monitor session all shut 443, 464
monitor session all type erspan-source 461
monitor session 431, 443, 461, 464
mpls strip 577
mpls strip dest-mac 580
mpls strip label 579
mpls strip label-age 581
mtu 437

N

NetFlow 519, 522
  timeouts 522
  VLAN でのブリッジ 519
no duplicate-message throttle 228
no monitor session all shut 464
no monitor session 431, 461, 464
no scheduler job name 264
no shut 434, 463, 465
no snmp trap link-status 306
no snmp-server protocol enable 310
no switch-profile 42
no system mode maintenance 629
no system mode maintenance dont-generate-profile 629
no system mode maintenance on-reload reset-reason 626
no isolate 615
no shutdown 615
no system interface shutdown 615
ntp access-group {peer | serve | serve-only | query-only} 163
ntp logging 166
ntp master 157
ntp source 165
ntp source-interface 165
ntp authenticate 162
ntp authentication-key 161
ntp server 158
ntp trusted-key 161
ntp peer 159

O

option exporter-stats 517, 533
option interface-table 517, 533

P

permit 254, 571
permit ip 436, 471
permit udf 436, 471
phone-contact 214
ptp 94, 102
ptp announce {interval | timeout} 98
ptp clock-mode 90
ptp delay-request minimum interval 98–99
ptp device-type generalized-ptp 89, 93
ptp device-type ordinary-clock-grandmaster 89, 93
ptp domain 89–90
ptp priority1 90
ptp priority2 91
Ptp utc-offset 93
ptp vlan 99
ptp device-type boundary-clock 89, 93
PTP 同期間隔 99
python instance 615, 619

R

record 518, 534
reload 436, 439, 471
rmon alarm 323
rmon event 324
rmon hcalarm 323
rollback running-config {checkpoint | file} 686
run bash 658

S

scheduler aaa-authentication password 262
scheduler aaa-authentication username 262
show scheduler job name 263–264
scheduler schedule name 266
sflow agent-ip 558
sflow collector-ip 555
sFlow collector-port 557
sFlow counter-poll-interval 553
sflow data-source interface ethernet 559
sflow data-source interface port-channel 559
sflow max-datagram-size 554
sflow max-sampled-size 552
sflow sampling-rate 551
show callhome destination-profile 216, 219–220, 234
show callhome destination-profile profile 216, 219–220
show callhome transport 223, 226, 234
show call-home user-def-cmds 221, 234

show cdp all 177
show cdp entry {all | name} 177
show cdp global 177
show cdp interface 176–177
show cdp neighbors {device-id | interface} 177
show cdp neighbors detail 172
show checkpoint 685–686
show checkpoint all 686
show checkpoint all system 686
show checkpoint all user 686
show checkpoint summary 686
show checkpoint summary system 686
show checkpoint summary user 686
show configuration session 253–255
show configuration session status 255
show configuration session summary 256
show diagnostic bootup level 342, 345
show diagnostic content module 343, 346
show diagnostic description module 346
show diagnostic events 346
show diagnostic ondemand setting 346
show diagnostic result module 346
show diagnostic simulation module 346
show diagnostic status module 344, 346
show diff rollback-patch {checkpoint | running-config | startup-config | file} 686–687
show event manager environment 355, 370
show event manager environment all 355, 370
show event manager event-types 371
show event manager event-types all 371
show event manager event-types module 371
show event manager history events 371
show event manager policy-state 356, 367, 371
show event manager script system 371
show event manager script system all 371
show event manager system-policy 348, 354, 371
show event manager system-policy all 371
show feature 551
show flow cache 523, 536, 542
show flow exporter 523, 542
show flow interface 523, 542
show flow record netflow layer2-switched input 521, 524
show flow record 514, 524, 528, 530, 542
show hardware capacity 346
show install active 640, 646
show install committed 647
show install inactive 646, 648
show install log 646, 659
show install packages 658
show interface brief 631
show interface snmp-ifindex 306, 313
show lldp interface 490, 498, 500
show lldp neighbors detail 501
show lldp neighbors interface 501
show lldp timers 500
show lldp tlv-select 500
show lldp traffic 501
show lldp traffic interface 501
show logging nvram 199–200
show logging console 184, 200
show logging info 188, 200
show logging logfile 199–200
show logging level 190, 200
show logging logfile end-time 199–200
show logging logfile start-time 199–200
show logging module 189, 200
show logging monitor 184, 200
clear logging nvram 199
show logging nvram last 199–200
show logging onboard 409
show logging onboard boot-uptime 409
show logging onboard counter-stats 409
show logging onboard credit-loss 409
show logging onboard device-version 409
show logging onboard endtime 409
show logging onboard environmental-history 409
show logging onboard error-stats 409
show logging onboard exception-log 409
show logging onboard interrupt-stats 409
show logging onboard module 409
show logging onboard obfl-history 409
show logging onboard obfl-logs 409
show logging onboard stack-trace 409
show logging onboard starttime 410
show logging onboard status 410
show logging origin-id 186
show logging timestamp 191, 200
show maintenance on-reload reset-reasons 631
show maintenance profile 631
show maintenance profile maintenance-mode 618, 631
show maintenance profile normal-mode 620, 631
show maintenance timeout 631
show monitor 443
show monitor session all 434, 444, 465, 469, 476
show monitor session range 434, 444, 469, 476
show monitor session 434, 437–438, 444, 463, 469, 472–473, 476
show mpls strip Labels 582
show mpls strip labels all 582
show mpls strip labels dynamic 582
show mpls strip labels static 582
show ntp access-groups 164, 167
show ntp logging-status 166–167
show ntp peer-status 167
show ntp peers 159, 167
show ntp rts-update 167
show ntp source 167
show ntp source-interface 167
show ntp statistics {io | local | memory | peer {ipaddr | name}} 167
show ptp brief 100, 132
show ptp clock 132
show ptp clock foreign-masters-record 133
show ptp corrections 133

show ptp counters 133
show ptp parent 133
show ptp port interface 100
show ptp port interface ethernet 133
show ptp time-property 133
show qos dcbxp interface 501
show rmon {alarms | hcalarms} 323
show rmon alarms 324
show rmon events 324
show rmon hcalarms 324
show rmon logs 325
show rollback log 687
show rollback log exec 687
show rollback log verify 687
show run aclmgr 603
show run ofm 603
show running-config | include "scheduler aaa-authentication" 262
show running-config | include "system memory" 369
show running-config callhome 234
show running-config eem 348, 371
show running-config lldp 489, 500
show running-config mmode 631
show running-config monitor 463, 465, 476
show running-config netflow 524, 536, 542
show running-config ntp 157, 167
show running-config ptp 133
show running-config sflow 561
show running-config sflow all 561
show running-config snmp 314
show running-config switch-profile 43
show scheduler config 260, 267–268
show scheduler job 264, 268
show scheduler logfile 268
show scheduler schedule 268
show sflow 552–554, 556–561
show snapshots 621, 631
show snapshots compare 621, 623, 632
show snapshots dump 632
show snapshots sections 623, 632
show snmp 308, 314
show snmp community 314
show snmp context 309, 314
show snmp engineID 314
show snmp group 314
show snmp host 296, 314
show snmp source-interface 292, 295, 315
show snmp trap 315
show snmp user 285, 315
show startup-config callhome 234
show startup-config eem 371
show startup-config mmode 631
show startup-config monitor 463, 465, 477
show startup-config switch-profile 43
show switch-profile 36–38, 40, 43
show system mode 627, 629, 632
show tech-support callhome 234
show callhome 215, 224, 227, 234
show clock 640
フロー イベントを表示 543
show flow filter 542
show flow monitor 542
show flow profile 542
フロー システムを表示 542
show install log detail 659
show ip access-list acl_nam 603
show ip access-lists 571, 575
show logging last 199–200
show logging server 193–194, 200
show mac access-lists 571, 575
show module 346, 640
show monitor session all stats 444, 477
show monitor session stats 444, 477
show ntp authentication-keys 161, 167
show ntp authentication-status 162, 167
show ntp trusted-keys 161, 167
show process 561
show snmp session 314
トンネル プロファイルを表示します 603
shut 443, 465
site-id 215
sleep instance 615
snapshot create 621
snapshot delete 621
snapshot section add 623
snapshot section delete 623
snmp-server aaa-user cache-timeout 312
snmp-server context 309
snmp-server counter cache timeout 311
snmp-server enable traps 299
snmp-server enable traps aaa 299
snmp-server enable traps bgp 299
snmp-server enable traps bridge 300
snmp-server enable traps callhome 300
snmp-server enable traps config 300
snmp-server enable traps eigrp 300
snmp-server enable traps entity 301
snmp-server enable traps feature-control 301
snmp-server enable traps hsrp 301
snmp-server enable traps license 302
snmp-server enable traps link 302
snmp-server enable traps ospf 303
snmp-server enable traps rf 303
snmp-server enable traps rmon 303
snmp-server enable traps snmp 303
snmp-server enable traps stpx 304
snmp-server enable traps syslog 304
snmp-server enable traps sysmgr 304
snmp-server enable traps upgrade 304
snmp-server enable traps vtp 305
snmp-server globalEnforcePriv 286
snmp-server mib community-map 309

snmp-server name 284
snmp-server source-interface {traps | informs} 292
snmp-server source-interface traps 295
snmp-server tcp-session 307
snmp-server community 288–289
snmp-server contact 214, 308
snmp-server host 290, 292, 294, 296
snmp-server location 308
snmp-server user 286–287, 293
source-interface 233
SPAN セッション 506
　構成 506
statistics per-entry 571
storm-control action trap 305
streetaddress 214
switch-priority 215
switch-profile 34, 37, 39
switchport 35, 431, 521–522, 535, 572, 578
switchport monitor 431
sync-peer destination 40
sync-peers destination 35, 41
system memory-thresholds minor 368
system memory-thresholds threshold critical no-process-kill 369
system mode maintenance dont-generate-profile 626
system mode maintenance on-reload reset-reason 626
system interface shutdown 615

T

tag 356
template data timeout 517, 533
terminal event-manager bypass 364
terminal monitor 183
time daily 266
time monthly 266
time start 266
time start now 266
time start repeat 266
time weekly 266
transport email from 223
transport email mail-server 222
transport http proxy enable 225
transport http proxy server 225
transport http use-vrf 224, 232
transport udp 517, 532
transport email from callhome_email-address 231
transport email smtp-server hostname/ip-address port 465/587 use-vrf
　vrf-name 231
ユーザー名 passwd メールを転送 {cleartext|encrypted} 231

U

udf 435, 470

V

verify 37, 254
version 9 517, 533
vlan configuration 519
vrf 462

あ

宛先 516, 532
宛先 IP 462
宛先インターフェイス 433, 437–438

い

一致 514, 528–529
イネーブル化 229, 232
イベント 356, 367
イベント アプリケーション 357
イベント ゴールド モジュール 358
イベント モジュール 359
インターフェイスのインポート 39

し

システムモードメンテナンスシャットダウン 626
システムモードメンテナンスタイムアウト 626
シャットダウン 615
出力 NetFlow 525
ジョブ名 266
診断開始モジュール 344
診断停止モジュール 344

す

スケジューラ ログファイル サイズ 261

せ

設定例 480–481
　ERSPAN 480–481
　　ipv6 経由の接続先 481
　　宛先 480
説明 356, 367, 431, 461, 514, 517–518, 527, 529, 532–533

そ

送信元 517, 532

て

定期的なインベントリ通知 227
定期的なインベントリ通知 timeofday 227
定期的なインベントリ通知の間隔 227
転送電子メール返送先 223

は

送信元インターフェイス 437, 472

ふ

フロー エクスポータ 516, 532
flow monitor 518, 533
フロー レコード 514, 520, 522, 527, 529, 535

ほ

保存 253, 255

れ

レイヤ 2 スイッチド フロー モニタ 521

ろ

logging module 189
logging level 189, 191

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。