



Cisco IOS XE Gibraltar 16.11.x (Catalyst 9300 スイッチ) セキュリティ設定ガイド

初版：2019年3月29日

最終更新：2020年1月30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。

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com go trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2019 Cisco Systems, Inc. All rights reserved.



目次

Short Description ?

第 1 章

パスワードおよび権限レベルによるスイッチ アクセスの制御	1
パスワードおよび権限によるスイッチ アクセスの制御の制約事項	1
可逆的パスワードタイプの制約事項とガイドライン	1
不可逆的パスワードタイプの制約事項とガイドライン	2
パスワードおよび権限によるスイッチアクセス制御に関する情報	2
不正アクセスの防止	3
デフォルトのパスワードおよび権限レベル設定	3
追加のパスワードセキュリティ	4
パスワードの回復	4
端末回線の Telnet 設定	5
ユーザ名とパスワードのペア	5
権限レベル	5
AES パスワード暗号化およびマスター暗号キー	6
パスワードおよび権限によるスイッチアクセスの設定方法	6
スタティック 有効 パスワードの設定または変更	6
暗号化によるイネーブルおよびイネーブル シークレット パスワードの保護	8
パスワード回復のディセーブル化	11
端末回線に対する Telnet パスワードの設定	12
ユーザ名とパスワードのペアの設定	13
コマンドの特権レベルの設定	15
回線のデフォルト特権レベルの変更	16
権限レベルへのログインおよび終了	17

暗号化事前共有キーの設定	18
パスワードおよび権限によるスイッチアクセスのモニター	19
パスワードおよび権限レベルによるスイッチアクセスの設定例	19
例：スタティック イネーブルパスワードの設定または変更	19
例：暗号化によるイネーブルおよびイネーブル シークレット パスワードの保護	20
例：端末回線に対する Telnet パスワードの設定	20
例：コマンドの権限レベルの設定	20
例：暗号化事前共有キーの設定	20
パスワードおよび権限によるスイッチアクセスの制御の機能履歴	21

第 2 章

認証の設定 23

認証の設定の前提条件	23
認証の設定に関する制約事項	23
認証について	24
認証の名前付き方式リスト	24
方式リストとサーバグループ	24
AAA によるログイン認証	25
イネーブルパスワードによるログイン認証	25
Kerberos によるログイン認証	26
ラインパスワードによるログイン認証	26
ローカルパスワードによるログイン認証	26
group RADIUS によるログイン認証	27
group TACACS によるログイン認証	27
グループ名によるログイン認証	27
AAA による PPP 認証	28
Kerberos による PPP 認証	28
ローカルパスワードによる PPP 認証	28
group RADIUS による PPP 認証	29
group TACACS による PPP 認証	29
グループ名による PPP 認証	29
PPP 要求の AAA スケーラビリティ	30

AAA による ARAP 認証	30
認可済みゲスト ログインを許可する ARAP 認証	30
ゲスト ログインを許可する ARAP 認証	31
ラインパスワードによる ARAP 認証	31
ローカルパスワードによる ARAP 認証	31
group RADIUS による ARAP 認証	31
group TACACS による ARAP 認証	32
グループ名による ARAP 認証	32
AAA による NASI 認証	33
イネーブルパスワードによる NASI 認証	33
group RADIUS による NASI 認証	33
group TACACS による NASI 認証	33
ラインパスワードによる NASI 認証	33
ローカルパスワードによる NASI 認証	34
グループ名による NASI 認証	34
ログイン入力にかける時間の指定	34
特権レベルでのパスワード保護	35
パスワードプロンプトに表示するテキストの変更	35
PPP セッションの二重認証	35
二重認証の機能	36
二重認証後のユーザー プロファイルへのアクセス	37
CHAP 認証または PAP 認証	38
PPP カプセル化の有効化	39
PAP または CHAP のイネーブル化	39
着信認証と発信認証	40
発信 PAP 認証のイネーブル化	41
PAP 認証要求の拒否	41
共通 CHAP パスワードの作成	41
CHAP 認証要求の拒否	41
ピアが認証されるまで CHAP 認証を遅延する	42
MS-CHAP の使用	42

ドメインストリッピング	43
認証の設定方法	44
AAA を使用したログイン認証の設定	44
AAA を使用した PPP 認証の設定	46
AAA を使用した ARAP 認証の設定	48
AAA を使用した NASI 認証の設定	50
ユーザー名が空のアクセス要求が RADIUS サーバーに送信されないようにする	52
AAA 認証のメッセージバナーの設定	53
ログインバナーの設定	53
Failed-Login バナーの設定	54
AAA パケット オブ ディスコネクトの設定	55
二重認証の設定	57
自動二重認証のイネーブル化	58
サーバー グループ レベルでのドメインストリッピングの設定	61
非 AAA 認証方式の設定	62
ラインパスワード保護の設定	62
ユーザー名認証の確立	64
MS-CHAP を使用した PPP 認証の定義	65
認証の設定例	67
例：方式リストの設定	67
例：RADIUS 認証	69
例：TACACS 認証	71
例：Kerberos 認証	72
例：AAA スケーラビリティ	72
例：AAA 認証のログインバナーおよび Failed-Login バナーの設定	74
例：AAA パケット オブ ディスコネクト サーバー キー	75
例：二重認証	75
例：二重認証による AAA のローカルホストの設定	76
例：第 1 段階の PPP 認証と許可に関する AAA サーバーの設定	76
例：第 2 段階の Per-User 認証と許可に関する AAA サーバーの設定	77
例：TACACS による設定完了	78

例：自動二重認証	81
認証設定の機能履歴	82

第 3 章

認可の設定 83

許可設定の前提条件	83
認可の設定の概要	84
認可の名前付き方式リスト	84
AAA 認可方式	85
認可方式	85
方式リストとサーバグループ	86
AAA 認可タイプ	87
承認タイプ	87
認可の属性値ペア	88
認可の設定方法	88
名前付き方式リストによる AAA 認可の設定	88
グローバル コンフィギュレーション コマンドの認可のディセーブル化	89
リバース Telnet の認可の設定	90
許可の設定例	91
例：TACACS 認可	91
例：RADIUS 許可	92
例：リバース Telnet 許可	93
認可の設定に関する追加情報	95
許可設定の機能履歴	95

第 4 章

アカウントिंगの設定 97

アカウントングを設定するための前提条件	97
アカウントングの設定の制約事項	98
アカウントングの設定に関する情報	98
アカウントングの名前付き方式リスト	98
方式リストとサーバグループ	99
AAA アカウントング方式	100

AAA アカウンティング タイプ	102
ネットワーク アカウンティング	102
EXEC アカウンティング	104
コマンド アカウンティング	106
接続 アカウンティング	106
システム アカウンティング	108
リソース アカウンティング	109
AAA アカウンティングの強化	111
AAA ブロードキャスト アカウンティング	111
AAA セッション MIB	111
アカウンティング属性と値のペア	112
AAA アカウンティングの設定方法	113
名前付き方式リストによる AAA アカウンティングの設定	113
スル ユーザ名セッション時のアカウンティング レコード生成の抑制	114
中間アカウンティング レコードの生成	115
定期的アカウンティング レコードを有効化する代替手段の設定	116
中間サービス アカウンティング レコードの生成	117
失敗したログインまたはセッションに対するアカウンティング レコードの生成	118
EXEC-Stop レコードよりも前のアカウンティング NETWORK-Stop レコードの指定	118
スイッチオーバー上のシステム アカウンティング レコードの抑制	119
AAA リソース失敗終了アカウンティングの設定	119
開始 - 終了レコードの AAA リソース アカウンティングの設定	119
AAA ブロードキャスト アカウンティング	120
DNIS による AAA ブロードキャスト アカウンティングの設定	120
AAA サーバーが到達不能な場合のデバイスとのセッションの確立	121
アカウンティングのモニタリング	121
アカウンティングのトラブルシューティング	122
AAA アカウンティングの設定例	122
例：名前付き方式リストの設定	122
例：AAA リソース アカウンティングの設定	124
例：AAA ブロードキャスト アカウンティングの設定	125

例：DNIS による AAA ブロードキャスト アカウンティングの設定 125

例：AAA セッション MIB 126

アカウンティングの設定に関するその他の参考資料 126

アカウンティングの設定の機能履歴 127

第 5 章

ローカル認証および許可の設定 129

ローカル認証および許可の設定方法 129

スイッチのローカル認証および許可の設定 129

ローカル認証および許可のモニタリング 131

ローカル認証および許可の機能履歴 131

第 6 章

TACACS+ の設定 133

TACACS+ の前提条件 133

TACACS+ の概要 134

TACACS+ およびスイッチ アクセス 134

TACACS+ の概要 134

TACACS+ の動作 136

方式リスト 136

TACACS+ 設定オプション 137

TACACS+ ログイン認証 137

特権 EXEC アクセスおよびネットワーク サービス用の TACACS+ 許可 137

TACACS+ Accounting 137

TACACS+ のデフォルト設定 138

TACACS+ を設定する方法 138

TACACS+ サーバホストの指定および認証キーの設定 138

TACACS+ ログイン認証の設定 140

特権 EXEC アクセスおよびネットワーク サービス用の TACACS+ 許可の設定 142

TACACS+ アカウンティングの起動 143

AAA サーバが到達不能な場合のデバイスとのセッションの確立 145

TACACS サーバグループの TACACS ソースインターフェイスの設定 145

TACACS+ のモニタリング 147

TACACS+ に関する追加情報 147

TACACS+ の機能の履歴 147

第 7 章

RADIUS の設定 149

RADIUS を設定するための前提条件 149

RADIUS の設定に関する制約事項 150

RADIUS に関する情報 151

RADIUS およびスイッチ アクセス 151

RADIUS の概要 151

RADIUS の動作 152

RADIUS 許可の変更 152

Change-of-Authorization 要求 154

CoA 要求応答コード 156

CoA 要求コマンド 157

セッション強制終了のスタック構成ガイドライン 160

RADIUS のデフォルト設定 161

RADIUS サーバ ホスト 161

RADIUS ログイン認証 162

AAA サーバグループ 162

AAA 許可 163

RADIUS アカウンティング 163

ベンダー固有の RADIUS 属性 163

ベンダー独自仕様の RADIUS サーバー通信 178

RADIUS の設定方法 179

RADIUS サーバ ホストの識別 179

RADIUS ログイン認証の設定 181

AAA サーバグループの定義 183

ユーザイネーブルアクセスおよびネットワーク サービスに関する RADIUS 許可の設定
184

RADIUS アカウンティングの起動 186

すべての RADIUS サーバの設定 186

ベンダー固有の RADIUS 属性を使用するデバイスの設定	188
ベンダー独自仕様の RADIUS サーバ通信に関するデバイスの設定	188
デバイス上での CoA の設定	190
RADIUS サーバグループの RADIUS ソースインターフェイスの設定	192
CoA 機能のモニタリング	194
RADIUS の機能の履歴	194

第 8 章
RadSec の設定 197

RadSec の設定に関する制限事項	197
RadSec に関する情報	197
RadSec の設定方法	198
RadSec over TLS の設定	198
TLS CoA の動的認可の設定	199
RadSec over DTLS の設定	200
DTLS CoA の動的認可の設定	202
RadSec のモニタリング	203
RadSec の設定例	203
例 : RadSec over TLS の設定	203
例 : TLS CoA の動的認可の設定	204
例 : RadSec over DTLS の設定	204
例 : DTLS CoA の動的認可の設定	204
RadSec 設定の機能履歴	204

第 9 章
Kerberos の設定 207

Kerberos の前提条件	207
Kerberos に関する情報	207
Kerberos とスイッチ アクセス	208
Kerberos の概要	208
Kerberos の動作	210
境界スイッチに対する認証の取得	211
KDC からの TGT の取得	211

ネットワーク サービスに対する認証の取得	211
Kerberos を設定する方法	211
Kerberos 設定の監視	212
Kerberos の機能履歴	212

第 10 章

MACsec の暗号化	213
MACsec 暗号化の前提条件	213
MACsec 暗号化の制約事項	213
MACsec 暗号化について	214
MACsec 暗号化の推奨事項	214
MACsec 暗号化の概要	215
Media Access Control Security と MACsec Key Agreement	216
MKA ポリシー	217
ポリシーマップアクションの定義	217
仮想ポート	217
MKA 統計情報	218
キー ライフタイムおよびヒットレス キー ロールオーバー	218
リプレイ保護ウィンドウ サイズ	219
MACsec、MKA、および 802.1x ホストモード	219
証明書ベースの MACsec 暗号化	220
中間スイッチの MACsec 接続	220
中間スイッチの MACsec 接続に関する制約事項	221
スイッチ間 MKA MACsec マストセキュアポリシー	221
MACsec Extended Packet Numbering (XPN)	222
ポートチャネルの MKA/MACsec	222
MACsec 暗号アナウンスメント	223
MACsec 暗号化の設定方法	223
MKA および MACsec の設定	224
MKA ポリシーの設定	224
スイッチからホストへの MACsec の暗号化設定	226
PSK を使用した MKA MACsec の設定	229

PSK を使用した MACsec MKA の設定	229
PSK を使用した、インターフェイスでの MACsec MKA の設定	230
証明書ベース MACsec 暗号化の設定	232
キー ペアの生成	232
SCEP による登録の設定	233
登録の手動設定	235
スイッチ間の MACsec の暗号化設定	237
MACsec XPN の設定	239
XPN の MKA ポリシーの設定	239
XPN MKA ポリシーをインターフェイスに適用する	240
ポートチャンネル用の MKA/MACsec の設定	241
PSK を使用したポートチャンネルの MKA/MACsec の設定	241
レイヤ 2 EtherChannel のポートチャンネル論理インターフェイスの設定	243
レイヤ 3 EtherChannel のポートチャンネル論理インターフェイスの設定	244
MACsec 暗号アナウンスメントの設定	245
セキュアアナウンスメントの MKA ポリシーの設定	245
セキュアアナウンスメントのグローバル設定	247
インターフェイスでの EAPOL アナウンスメントの設定	248
Cisco TrustSec MACsec の設定	249
手動モードでの Cisco TrustSec スイッチ間リンク セキュリティの設定	249
MACsec 暗号化の設定例	251
例：MKA および MACsec の設定	251
例：PSK を使用した MACsec MKA の設定	251
例：証明書ベース MACsec 暗号化を使用した MACsec MKA の設定	253
例：MACsec XPN の設定	253
例：PSK を使用したポートチャンネルの MACsec MKA の設定	255
例：MACsec 暗号アナウンスメントの設定	262
例：MKA 情報の表示	266
MACsec 暗号化に関する追加情報	272
MACsec 暗号化の機能情報	273

第 11 章

セキュア シェルの設定 275

セキュア シェルの設定 275

セキュア シェルを設定するための前提条件 275

セキュア シェルの設定に関する制約事項 276

セキュア シェルの設定について 276

SSH サーバ 276

SSH 統合クライアント 277

RSA 認証のサポート 277

SSH サーバ、統合クライアント、およびサポートされているバージョン 277

SSH 設定時の注意事項 278

セキュア シェルの設定方法 279

SSH を実行するためのデバイスの設定 279

SSH サーバーの設定 280

SSH クライアントの呼び出し 281

セキュア シェルの設定例 282

例：SSH サーバーの設定 282

例：SSH クライアントの呼び出し 282

例：SSH の確認 282

セキュア シェルに関するその他の参考資料 283

セキュア シェルの設定の機能履歴 284

第 12 章

セキュア シェルバージョン 2 サポート 285

セキュア シェルバージョン 2 サポートの前提条件 285

セキュア シェルバージョン 2 サポートの制約事項 286

セキュア シェルバージョン 2 サポートに関する情報 286

SSH バージョン 2 286

セキュア シェルバージョン 2 の機能拡張 287

セキュア シェルバージョン 2 の RSA キーに関する機能拡張 287

SSH およびスイッチ アクセス 288

SNMP トラップ生成 289

SSH キーボードインタラクティブ認証	289
セキュア シェルの設定方法	289
ホスト名およびドメイン名を使用した SSH バージョン 2 のデバイス設定	289
RSA キーペアを使用した SSH バージョン 2 のデバイス設定	291
RSA ベースのユーザ認証を実行するための Cisco SSH サーバの設定	292
RSA ベースのサーバ認証を実行するための Cisco IOS SSH サーバの設定	294
リモート デバイスとの暗号化セッションの開始	296
セキュア シェル接続のステータスの確認	297
セキュアシェルバージョン 2 のステータスの確認	298
セキュア シェルバージョン 2 のモニタリングと維持	300
セキュア シェルバージョン 2 サポートの設定例	303
例：セキュア シェルバージョン 2 の設定	303
例：セキュア シェルバージョン 1 および 2 の設定	303
例：リモート デバイスでの暗号化セッションの開始	303
例：SNMP トラップの設定	303
例：SSH キーボードインタラクティブ認証	303
例：クライアント側のデバッグの有効化	303
例：ブランク パスワードの変更による ChPass の有効化	304
例：ChPass の有効化および初回ログインでのパスワード変更	304
例：ChPass の有効化および 3 回ログインした後のパスワードの失効	305
例：SNMP のデバッグ	306
例：SSH のデバッグの強化	306
セキュア シェルバージョン 2 サポートの追加情報	307
セキュアシェルバージョン 2 サポートの機能履歴	308

 第 13 章

SSH File Transfer Protocol の設定	309
SSH File Transfer Protocol の前提条件	309
SSH File Transfer Protocol の制約事項	309
IPv6 を介した SSH サポートに関する情報	310
SSH File Transfer Protocol の概要	310
SSH File Transfer Protocol の設定方法	310

SFTP の設定	310
SFTP コピー操作の実行	311
IPv6 を介した SSH サポートの設定例	312
例：SSH File Transfer Protocol の設定	312
SSH File Transfer Protocol に関する追加情報	312
SSH File Transfer Protocol の機能履歴	312

第 14 章

SSH 認証の X.509v3 証明書 315

SSH 認証の X.509v3 証明書	315
SSH 認証の X.509v3 証明書の前提条件	315
SSH 認証の X.509v3 証明書の制約事項	315
SSH 認証用の X.509v3 証明書に関する情報	316
デジタル証明書	316
X.509v3 を使用したサーバーおよびユーザー認証	316
SSH 認証用の X.509v3 証明書の設定方法	316
サーバー認証にデジタル証明書を使用するための SSH サーバーの設定	316
ユーザー認証用のデジタル証明書を確認するための SSH サーバーの設定	317
デジタル証明書を使用したサーバーおよびユーザー認証の設定の確認	319
SSH 認証用の X.509v3 証明書の設定例	320
例：サーバー認証にデジタル証明書を使用するための SSH サーバーの設定	320
例：ユーザー認証用のデジタル証明書を確認するための SSH サーバーの設定	320
SSH 認証用の X.509v3 証明書の機能履歴	320

第 15 章

コモンクライテリア認定用の SSH アルゴリズム 323

コモンクライテリア認定用の SSH アルゴリズムに関する情報	323
コモンクライテリア認定用の SSH アルゴリズム	323
Cisco IOS SSH サーバー アルゴリズム	323
Cisco IOS SSH クライアント アルゴリズム	324
コモンクライテリア認定用の SSH アルゴリズムの設定方法	325
Cisco IOS SSH サーバーおよびクライアントの暗号キー アルゴリズムの設定	325
トラブルシューティングのヒント	326

Cisco IOS SSH サーバーおよびクライアントの MAC アルゴリズムの設定	327
トラブルシューティングのヒント	328
Cisco IOS SSH サーバーのホスト キー アルゴリズムの設定	328
トラブルシューティングのヒント	329
コモン クライテリア認定用の SSH アルゴリズムの設定例	329
例：Cisco IOS SSH サーバーの暗号キー アルゴリズムの設定	330
例：Cisco IOS SSH クライアントの暗号キー アルゴリズムの設定	330
例：Cisco IOS SSH サーバーの MAC アルゴリズムの設定	330
例：Cisco IOS SSH サーバーのホスト キー アルゴリズムの設定	330
コモン クライテリア認定用の SSH アルゴリズムの確認	330
コモン クライテリア認定用のセキュアシェルアルゴリズムの機能情報	331

 第 16 章

Secure Socket Layer HTTP の設定 333

Secure Socket Layer HTTP に関する情報	333
セキュア HTTP サーバおよびクライアントの概要	333
CA のトラストポイント	333
CipherSuite	335
SSL のデフォルト設定	336
SSL の設定時の注意事項	336
Secure Socket Layer HTTP の設定方法	337
CA のトラストポイントの設定	337
セキュア HTTP サーバの設定	339
セキュア HTTP クライアントの設定	342
セキュア HTTP サーバおよびクライアントのステータスのモニタリング	344
Secure Socket Layer HTTP に関するその他の参考資料	344
Secure Socket Layer HTTP の機能履歴	345

 第 17 章

IPv4 ACL 347

IPv4 アクセスコントロールリストの制約事項	347
IPv4 アクセスコントロールリストに関する情報	349
ACL の概要	349

アクセス コントロール エントリ	349
ACL でサポートされるタイプ	350
サポートされる ACL	350
ACL 優先順位	350
ポート ACL	351
ルータ ACL	352
VLAN マップ	353
ACE およびフラグメント化されたトラフィックとフラグメント化されていないトラフィック	353
ACL とスイッチ スタック	354
アクティブ スイッチおよび ACL の機能	354
スタック メンバおよび ACL の機能	354
アクティブ スイッチの障害および ACL	355
標準 IPv4 ACL および拡張 IPv4 ACL	355
IPv4 ACL スイッチでサポートされていない機能	355
アクセス リスト番号	355
番号付き標準 IPv4 ACL	356
番号付き拡張 IPv4 ACL	356
名前付き IPv4 ACL	357
ACL ロギング	358
ハードウェアおよびソフトウェアによる IP ACL の処理	359
VLAN マップの設定時の注意事項	359
VLAN マップとルータ ACL	360
VLAN マップとルータ ACL の設定時の注意事項	361
ACL の時間範囲	361
IPv4 ACL のインターフェイスに関する注意事項	362
IPv4 アクセスコントロールリストの設定方法	363
IPv4 ACL の設定	363
番号付き標準 ACL の作成	363
番号付き拡張 ACL の作成	364
名前付き標準 ACL の作成	368

名前付き拡張 ACL の作成	369
ACL の時間範囲の設定	370
端末回線への IPv4 ACL の適用	372
インターフェイスへの IPv4 ACL の適用	373
名前付き MAC 拡張 ACL の作成	374
レイヤ 2 インターフェイスへの MAC ACL の適用	375
VLAN マップの設定	377
VLAN への VLAN マップの適用	379
IPv4 ACL のモニタリング	380
IPv4 アクセスコントロールリストの設定例	381
小規模ネットワークが構築されたオフィス用の ACL	381
例：小規模ネットワークが構築されたオフィスの ACL	381
例：番号付き ACL	382
例：拡張 ACL	383
例：名前付き ACL	384
例：ACL ロギング	385
例：ACE およびフラグメント化されたトラフィックとフラグメント化されていないトラフィック	386
例：ACL での時間範囲を使用	387
例：IP ACL に適用される時間範囲	388
例：ACL へのコメントの挿入	388
例：パケットを拒否する ACL および VLAN マップの作成	389
例：パケットを許可する ACL および VLAN マップの作成	389
例：IP パケットのドロップおよび MAC パケットの転送のデフォルトアクション	389
例：MAC パケットのドロップおよび IP パケットの転送のデフォルトアクション	390
例：すべてのパケットをドロップするデフォルトアクション	391
例：ネットワークでの VLAN マップの使用	391
例：ワイヤリング クローゼットの設定	391
例：別の VLAN にあるサーバーへのアクセスの制限	393
例：別の VLAN にあるサーバーへのアクセスの拒否	393
IPv4 アクセスコントロールリストに関する追加情報	394

IPv4 アクセスコントロールリストの機能履歴 394

第 18 章

IPv6 ACL 397

IPv6 ACL の制限 397

IPv6 ACL の概要 398

IPv6 ACL の概要 398

サポートされる ACL 398

ACL のタイプ 399

ユーザー単位 IPv6 ACL 399

フィルタ ID IPv6 ACL 399

ダウンロード可能 IPv6 ACL 399

スイッチ スタックおよび IPv6 ACL 399

ACL 優先順位 399

VLAN マップ 400

他の機能およびスイッチとの相互作用 401

IPv6 ACL の設定方法 401

IPv6 ACL のデフォルト設定 402

IPv6 ACL の設定 402

インターフェイスへの IPv6 ACL の付加 405

VLAN マップの設定 406

VLAN への VLAN マップの適用 408

IPv6 ACL のモニタリング 409

IPv6 ACL の設定例 409

例：IPv6 ACL の作成 409

例：IPv6 ACL の表示 410

例：VLAN アクセスマップ設定の表示 410

IPv6 ACL の機能履歴 411

第 19 章

IP ソース ガードの設定 415

IP ソース ガードの概要 415

IP ソース ガード 415

スタティック ホスト用 IP ソース ガード	416
IP ソース ガードの設定時の注意事項	417
IP ソース ガードの設定方法	418
IP ソース ガードのイネーブル化	418
レイヤ2 アクセス ポートでのスタティック ホスト用 IP ソース ガードの設定	419
IP ソース ガードのモニタリング	421
IP ソース ガードの機能の履歴	421

第 20 章

ダイナミック ARP インспекションの設定	423
ダイナミック ARP インспекションの制約事項	423
ダイナミック ARP インспекションに関する情報	425
ダイナミック ARP インспекションの概要	425
インターフェイスの信頼状態とネットワーク セキュリティ	426
ARP パケットのレート制限	428
ARP ACL および DHCP スヌーピング エントリの相対的なプライオリティ	428
廃棄パケットのロギング	429
ダイナミック ARP インспекションのデフォルト設定	429
ARP ACL および DHCP スヌーピング エントリの相対的なプライオリティ	430
ダイナミック ARP インспекションの設定方法	430
非 DHCP 環境での ARP ACL の設定	430
DHCP 環境でのダイナミック ARP インспекションの設定	433
着信 ARP パケットのレート制限	435
ダイナミック ARP インспекション検証チェックの実行	437
DAI のモニタリング	439
DAI の設定の確認	439
ダイナミック ARP インспекションの機能履歴	440

第 21 章

IPv6 ファースト ホップ セキュリティの設定	441
IPv6 ファースト ホップ セキュリティの前提条件	441
IPv6 ファースト ホップ セキュリティの制約事項	441
IPv6 ファースト ホップ セキュリティに関する情報	442

IPv6 ファースト ホップ セキュリティの概要	442
IPv6 ファースト ホップ セキュリティの設定方法	444
IPv6 スヌーピング ポリシーの設定	444
インターフェイスへの IPv6 スヌーピングポリシーの適用	447
レイヤ 2 EtherChannel インターフェイスへの IPv6 スヌーピングポリシーの適用	449
VLAN への IPv6 スヌーピングポリシーのグローバル適用	450
IPv6 バインディング テーブルの内容の設定	451
IPv6 ネイバー探索インスペクションポリシーの設定	452
インターフェイスへの IPv6 ネイバー探索インスペクションポリシーの適用	454
レイヤ 2 EtherChannel インターフェイスへの IPv6 ネイバー探索インスペクションポリシーの適用	455
VLAN への IPv6 ネイバー探索インスペクションポリシーのグローバル適用	456
IPv6 ルータ アドバタイズメント ガード ポリシーの設定	457
インターフェイスへの IPv6 ルータ アドバタイズメント ガード ポリシーの適用	460
レイヤ 2 EtherChannel インターフェイスへの IPv6 ルータ アドバタイズメント ガード ポリシーの適用	461
VLAN への IPv6 ルータ アドバタイズメント ガード ポリシーのグローバル適用	462
IPv6 DHCP ガードポリシーの設定	463
インターフェイスまたはインターフェイス上の VLAN への IPv6 DHCP ガードポリシーの適用	466
レイヤ 2 EtherChannel インターフェイスへの IPv6 DHCP ガードポリシーの適用	467
VLAN への IPv6 DHCP ガードポリシーのグローバル適用	468
IPv6 ソース ガードの設定	469
インターフェイスへの IPv6 ソースガードポリシーの適用	470
レイヤ 2 EtherChannel インターフェイスへの IPv6 ソースガードポリシーの適用	471
IPv6 プレフィックス ガードの設定	472
インターフェイスへの IPv6 プレフィックス ガード ポリシーの適用	473
レイヤ 2 EtherChannel インターフェイスへの IPv6 プレフィックス ガード ポリシーの適用	474
IPv6 ファースト ホップ セキュリティの設定例	475
例：IPv6 DHCP ガードポリシーの設定	475
例：レイヤ 2 EtherChannel インターフェイスへの IPv6 ソースガードポリシーの適用	476

例：レイヤ 2 EtherChannel インターフェイスへの IPv6 プレフィックス ガード ポリシーの適用 476

IPv6 ファースト ホップ セキュリティに関する追加情報 476

IPv6 ファースト ホップ セキュリティの機能履歴 477

第 22 章

SISF ベースのデバイストラッキングの設定 479

SISF ベースのデバイストラッキングに関する情報 479

SISF ベースのデバイストラッキングの概要 479

SISF ベースのデバイストラッキングを有効にするオプション 480

レガシー IPDT と IPv6 スヌーピングから SISF ベースのデバイストラッキングへの移行 481

SISF ベースのデバイストラッキングの設定方法 483

SISF ベースのデバイストラッキングの手動による有効化 483

ターゲットへのデフォルト デバイス トラッキング ポリシーの適用 483

カスタム設定を使用したカスタム デバイス トラッキング ポリシーの作成 484

デバイス トラッキング ポリシーのインターフェイスへの適用 488

デバイス トラッキング ポリシーの VLAN への適用 489

Cisco IOS XE Fuji 16.9.x 以降のリリースでの SISF ベースのデバイストラッキングのプログラムによる有効化 491

トランクポートからのバインディングエントリの作成を停止するためのマルチスイッチネットワークの設定 492

SISF ベースのデバイストラッキングの設定例 494

例：Cisco IOS XE Everest 16.9.x 以降のリリースでの SISF ベースのデバイストラッキングのプログラムによる有効化 494

例：ターゲットでの IPv6 デバイストラッキングの無効化 497

例：VLAN 上の SVI に対する IPv6 の有効化（重複アドレスの問題を軽減するため） 497

例：IPv4 重複アドレスの問題の緩和 498

例：短いデバイストラッキング バインディング到達可能時間の回避 500

SISF の機能履歴 500

第 23 章

IEEE 802.1x ポートベースの認証の設定 503

IEEE 802.1x ポートベース認証の制約事項 503

IEEE 802.1x ポートベースの認証に関する情報 504

ポートベース認証プロセス	505
ポートベース認証の開始およびメッセージ交換	507
ポートベース認証方法	509
ユーザー単位 ACL および Filter-Id	510
許可ステートおよび無許可ステートのポート	510
802.1X のホスト モード	511
MAC 移動	512
MAC 置換	513
802.1x アカウンティング	513
802.1x アカウンティング属性値ペア	514
802.1x 準備状態チェック	515
スイッチと RADIUS サーバー間の通信	515
IEEE 802.1x 認証	515
802.1X 認証	515
ポートベース認証マネージャ CLI コマンド	517
802.1x 認証のデフォルト設定	517
ポートベース認証とスイッチ スタック	519
VLAN 割り当てを使用した 802.1x 認証	519
ユーザー単位 ACL を使用した 802.1x 認証	521
ダウンロード可能 ACL およびリダイレクト URL を使用した 802.1x 認証	522
VLAN ID ベース MAC 認証	524
MAC 認証バイパスを使用した IEEE 802.1x 認証	524
802.1x マルチ認証モード	526
ゲスト VLAN を使用した 802.1x 認証	528
制限付き VLAN を使用した 802.1x 認証	529
アクセス不能認証バイパスを使用した 802.1x 認証	530
VLAN 割り当て、ゲスト VLAN、制限付き VLAN、アクセス不能認証バイパス	533
802.1x クリティカル音声 VLAN	533
音声 VLAN ポートを使用した IEEE 802.1x 認証	534
VoL 機能を使用した IEEE 802.1x 認証	535
柔軟な認証の順序設定	536

Open1x 認証	536
マルチドメイン認証	537
Network Edge Access Topology を使用した 802.1x サプリカントおよびオーセンティケー タスイッチ	538
802.1x ユーザ ディストリビューション	540
802.1x ユーザ ディストリビューションの設定時の注意事項	541
Network Admission Control レイヤ 2 IEEE 802.1x 検証	541
音声認識 802.1x セキュリティ	542
コモン セッション ID	542
ポートあたりのデバイスの最大数	543
802.1x ポートベース認証の設定方法	543
802.1X 認証の設定	543
802.1x ポートベース認証の設定	544
定期的な再認証の設定	547
802.1x 違反モードの設定	549
待機時間の変更	551
スイッチからクライアントへの再送信時間の変更	552
スイッチからクライアントへのフレーム再送信回数の設定	554
ホスト モードの設定	555
MAC 移動のイネーブル化	557
MAC 置換のイネーブル化	558
802.1x アカウンティングの設定	559
802.1x 準備状態チェックの設定	561
スイッチ/RADIUS サーバー間通信の設定	562
再認証回数の設定	564
ゲスト VLAN の設定	565
制限付き VLAN の設定	567
制限付き VLAN の認証試行回数の設定	568
クリティカル音声 VLAN を使用した 802.1x アクセス不能認証バイパスの設定	569
WoL を使用した 802.1x 認証の設定	573
MAC 認証バイパスの設定	574

802.1x ユーザー ディストリビューションの設定	575
NAC レイヤ 2 802.1x 検証の設定	576
NEAT を使用したオーセンティケータ スイッチの設定	578
NEAT を使用したサブリカント スイッチの設定	580
ダウンロード可能 ACL およびリダイレクト URL を使用した 802.1x 認証の設定	582
ダウンロード可能な ACL の設定	583
ダウンロードポリシーの設定	584
VLAN ID ベース MAC 認証の設定	587
柔軟な認証順序の設定	587
Open1x の設定	589
ポート上での 802.1x 認証のディセーブル化	591
802.1x 認証設定のデフォルト値へのリセット	592
音声認識 802.1x セキュリティの設定	593
IEEE 802.1x ポートベースの認証の設定例	596
例：アクセス不能認証バイパスの設定	596
例：VLAN グループの設定	596
IEEE 802.1x ポートベースの認証統計情報とステータスのモニタリング	597
IEEE 802.1x ポートベースの認証の機能履歴	598

第 24 章

Web ベース認証	599
Web ベース認証の制約事項	599
Web ベース認証について	599
Web ベース認証の概要	599
デバイスのロール	601
ホストの検出	601
セッションの作成	602
認証プロセス	602
ローカル Web 認証バナー	603
Web 認証カスタマイズ可能な Web ページ	606
ガイドライン	606
認証プロキシ Web ページの注意事項	607

成功ログインに対するリダイレクト URL の注意事項	608
その他の機能と Web ベース認証の相互作用	609
ポート セキュリティ	609
LAN ポート IP	609
ゲートウェイ IP	609
ACL	609
コンテキストベース アクセス コントロール	609
EtherChannel	610
Web ベース認証の設定方法	610
デフォルトの Web ベース認証の設定	610
Web ベース認証の設定に関する注意事項と制約事項	610
認証ルールとインターフェイスの設定	612
AAA 認証の設定	614
スイッチ/RADIUS サーバー間通信の設定	616
HTTP サーバーの設定	618
認証プロキシ Web ページのカスタマイズ	619
成功ログインに対するリダイレクション URL の指定	621
Web ベース認証パラメータの設定	622
Web ベース認証ローカル バナーの設定	622
Web ベース認証キャッシュ エントリの削除	623
Web ベース認証の確認	624
Web ベース認証の機能履歴	624

第 25 章

ポート単位のトラフィック制御の設定	627
ポートベースのトラフィック制御	627
ポートベースのトラフィック制御に関する情報	627
ストーム制御	627
保護ポート	629
ポート ブロッキング	629
ポートベースのトラフィック制御の設定方法	630
ストーム制御およびしきい値レベルの設定	630

保護ポートの設定	633
インターフェイスでのフラッディング トラフィックのブロッキング	634
ポートベースのトラフィック制御に関するその他の関連資料	636
ポートベースのトラフィック制御の機能履歴	636

第 26 章

ポート セキュリティ 637

ポート セキュリティの前提条件	637
ポート セキュリティの制約事項	637
ポート セキュリティの概要	638
ポート セキュリティ	638
セキュア MAC アドレスのタイプ	638
MAC アドレス テーブルのデフォルト設定	638
MAC アドレス テーブルの作成	639
スティッキ セキュア MAC アドレス	639
セキュリティ違反	639
ポート セキュリティ エージング	641
ポート セキュリティとスイッチ スタック	641
デフォルトのポート セキュリティ設定	642
ポート セキュリティの設定時の注意事項	642
ポート セキュリティの設定方法	644
ポート セキュリティのイネーブル化および設定	644
ポート セキュリティ エージングのイネーブル化および設定	649
アドレス エージング タイムの変更	651
ポート セキュリティの監視	652
ポート セキュリティの設定例	652
ポート セキュリティの機能の履歴	653

第 27 章

コントロール プレーン ポリシングの設定 655

CoPP の制約事項	655
CoPP の概要	656
CoPP の概要	656

システム定義の CoPP の特徴	657
ユーザー設定可能な CoPP の特徴	665
ソフトウェアバージョンのアップグレードまたはダウングレード	666
ソフトウェアバージョンのアップグレードと CoPP	666
ソフトウェアバージョンのダウングレードと CoPP	667
CoPP の設定方法	667
CPU キューの有効化またはポリサー レートの変更	667
CPU キューの無効化	669
すべての CPU キューに対するデフォルトのポリサー レートの設定	671
CoPP の設定例	672
例：CPU キューの有効化または CPU キューのポリサー レートの変更	672
例：CPU キューの無効化	673
例：すべての CPU キューに対するデフォルトのポリサー レートの設定	673
CoPP のモニタリング	677
CoPP の機能の履歴	677

第 28 章

PKI での証明書の許可および失効の設定	681
PKI での証明書の許可および失効の設定	681
証明書の許可および失効に関する前提条件	681
証明書の許可および失効に関する制約事項	682
証明書の許可および失効に関する情報	682
PKI の許可	682
証明書ステータスのための PKI と AAA サーバーの統合	682
CRL または OCSP サーバー：証明書失効メカニズムの選択	685
許可または失効用に証明書ベースの ACL を使用する場合	687
PKI 証明書チェーンの検証	689
PKI で証明書の許可および失効を設定する方法	690
AAA サーバーとの PKI 統合の設定	690
PKI 証明書ステータス チェックの失効メカニズムの設定	695
証明書の許可および失効の設定	697
証明書チェーンの設定	706

PKI における証明書の許可および失効の設定例	708
PKI AAA 許可の設定および確認の例	708
例：失効メカニズムの設定	712
例：セントラルサイトにあるハブデバイスを証明書失効チェック用に設定	713
例：証明書の許可および失効の設定	718
例：証明書チェーン検証の設定	720
PKI での証明書の許可および失効の追加リファレンス	722
PKI での証明書の許可および失効の機能履歴	722



第 1 章

パスワードおよび権限レベルによるスイッチ アクセスの制御

- [パスワードおよび権限によるスイッチ アクセスの制御の制約事項 \(1 ページ\)](#)
- [パスワードおよび権限によるスイッチアクセス制御に関する情報 \(2 ページ\)](#)
- [パスワードおよび権限によるスイッチアクセスの設定方法 \(6 ページ\)](#)
- [パスワードおよび権限によるスイッチアクセスのモニター \(19 ページ\)](#)
- [パスワードおよび権限レベルによるスイッチアクセスの設定例, on page 19](#)
- [パスワードおよび権限によるスイッチアクセスの制御の機能履歴 \(21 ページ\)](#)

パスワードおよび権限によるスイッチアクセスの制御の制約事項

パスワードおよび権限によるスイッチアクセスの制御の制約事項は、次のとおりです。

- **boot manual** グローバルコンフィギュレーションコマンドを使用して、スイッチを手動で起動するように設定している場合は、パスワード回復をディセーブルにできません。このコマンドは、スイッチの電源の再投入後、ブートローダプロンプト (`switch:`) を表示させます。

可逆的パスワードタイプの制約事項とガイドライン

- パスワードタイプ 0 および 7 は、パスワードタイプ 6 に置き換えられます。したがって、コンソール、Telnet、SSH、WebUI、NETCONF への管理者ログインに使用されるパスワードタイプ 0 およびタイプ 7 は、パスワードタイプ 6 に移行する必要があります。CHAP、EAP などのローカル認証でユーザー名とパスワードがタイプ 0 およびタイプ 7 の場合、アクションは不要です。
- スタートアップコンフィギュレーションにタイプ 6 のパスワードがあり、タイプ 6 のパスワードがサポートされていないバージョンにダウングレードすると、デバイスからロックアウトされる可能性があります。

不可逆的パスワードタイプの制約事項とガイドライン

- ユーザ名シークレットパスワードタイプ5およびイネーブルシークレットパスワードタイプ5は、より強力なパスワードタイプ8または9に移行する必要があります。詳細については、「[暗号化によるイネーブルおよびイネーブルシークレットパスワードの保護 \(8 ページ\)](#)」を参照してください。
- デバイスのスタートアップコンフィギュレーションに複雑なタイプ9シークレット (\$14\$ で始まるパスワード) がある場合、ダウングレードは複雑なタイプ9シークレットがサポートされているリリースでのみ実行できます。複雑なタイプ9シークレットは、Cisco IOS XE Gibraltar 16.11.2 以降のリリースでサポートされます。スタートアップコンフィギュレーションに複雑なタイプ9シークレットが含まれており、Cisco IOS XE Gibraltar 16.11.2 より前のリリースにダウングレードすると、デバイスからロックアウトされます。複雑なタイプ9シークレット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、複雑なタイプ9シークレット (\$14\$ で始まるパスワード) またはタイプ5シークレット (\$1\$ で始まるパスワード) ではなく、タイプ9シークレット (\$9\$ で始まるパスワード) がスタートアップコンフィギュレーションに含まれていることを確認します。デバイスが、Cisco IOS XE Fuji 16.9.x、Cisco IOS XE Gibraltar 16.10.x、または Cisco IOS XE Gibraltar 16.11.x から Cisco IOS XE Gibraltar 16.12.x へアップグレードされると、タイプ5シークレットは複雑なタイプ9シークレット (\$14\$ で始まるパスワード) に自動変換されます。たとえば、`username user1 secret 5 1dNmW$7jWhqdtZ2qBVz2R4CSZZC0` は `username user1 secret 9 14dNmW$QykGZEEGmiEGrE$C9D/fD0czicOtgaZAa1CTa2sgygi0Leyw3/cLqPY426` に自動変換されます。デバイスがアップグレードされたら、特権 EXEC モードで **write memory** コマンドを実行し、複雑なタイプ9シークレットをスタートアップコンフィギュレーションに永続的に書き込みます。
- プレーンテキストパスワードは、不可逆的暗号化パスワードタイプ9に変換されます。



(注) これは、Cisco IOS XE Gibraltar 16.10.1 以降のリリースでサポートされています。

- シークレットパスワードタイプ4はサポートされていません。

パスワードおよび権限によるスイッチアクセス制御に関する情報

ここでは、パスワードおよび権限によるスイッチアクセス制御に関する情報を示します。

不正アクセスの防止

不正ユーザーによる、スイッチの再設定や設定情報の閲覧を防止できます。一般的には、ネットワーク管理者からスイッチへのアクセスを許可する一方、非同期ポートを用いてネットワーク外からダイヤルアップ接続するユーザーや、シリアルポートを通じてネットワーク外から接続するユーザー、またはローカルネットワーク内の端末またはワークステーションから接続するユーザーによるアクセスを制限します。

スイッチへの不正アクセスを防止するには、次のセキュリティ機能を1つまたは複数設定します。

- 最低限のセキュリティとして、各スイッチポートでパスワードおよび権限を設定します。このパスワードは、スイッチにローカルに保存されます。ユーザーがポートまたは回線を通じてスイッチにアクセスしようとするとき、ポートまたは回線に指定されたパスワードを入力してからでなければ、スイッチにアクセスできません。
- 追加のセキュリティレイヤとして、ユーザー名とパスワードをペアで設定できます。このペアはスイッチでローカルに保存されます。このペアは回線またはポートに割り当てられ、各ユーザを認証します。ユーザは認証後、スイッチにアクセスできます。権限レベルを定義している場合は、ユーザ名とパスワードの各ペアに特定の権限レベルを、対応する権利および権限とともに割り当てることもできます。
- ユーザ名とパスワードのペアを使用したいが、そのペアをローカルではなく中央のサーバに保存したい場合は、セキュリティ サーバ上のデータベースに保存できます。これにより、複数のネットワーク デバイスが同じデータベースを使用してユーザ認証情報を（必要に応じて許可情報も）得ることができます。
- また、失敗したログイン試行をログに記録するログイン拡張機能もイネーブルにすることもできます。ログイン拡張は、設定した回数のログインが失敗したあとに、それ以降のログイン試行をブロックするために設定することもできます。

デフォルトのパスワードおよび権限レベル設定

ネットワークで端末のアクセスコントロールを行う簡単な方法は、パスワードを使用して権限レベルを割り当てることです。パスワード保護によって、ネットワークまたはネットワーク デバイスへのアクセスが制限されます。権限レベルによって、ネットワーク デバイスにログイン後、ユーザがどのようなコマンドを使用できるかが定義されます。

次の表に、デフォルトのパスワードおよび権限レベル設定を示します。

Table 1: デフォルトのパスワードおよび権限レベル設定

機能	デフォルト設定
イネーブルパスワードおよび権限レベル	パスワードは定義されていません。デフォルトはレベル 15 です（特権 EXEC レベル）。パスワードは、コンフィギュレーション ファイル内では暗号化されていない状態です。

機能	デフォルト設定
イネーブル シークレット パスワードおよび権限レベル	パスワードは定義されていません。デフォルトはレベル 15 です (特権 EXEC レベル)。パスワードは、暗号化されてからコンフィギュレーション ファイルに書き込まれます。
回線パスワード	パスワードは定義されていません。

追加のパスワードセキュリティ

セキュリティレベルを強化するために、特にネットワークを超えるパスワードや Trivial File Transfer Protocol (TFTP; 簡易ファイル転送プロトコル) サーバーに保存されたパスワードについて、グローバル コンフィギュレーション コマンド **enable password** または **enable secret** を使用できます。コマンドの作用はどちらも同じです。このコマンドにより、暗号化されたパスワードを設定できます。特権 EXEC モード (デフォルト設定) または特定の権限レベルにアクセスするユーザは、このパスワードを入力する必要があります。

より高度な暗号化アルゴリズムが使用されるので、**enable secret** コマンドを使用することを推奨します。

enable secret コマンドを設定した場合、このコマンドは **enable password** コマンドよりも優先されます。同時に 2 つのコマンドを有効にはできません。

パスワードの暗号化をイネーブルにすると、ユーザー名パスワード、認証キーパスワード、イネーブル コマンド パスワード、コンソールおよび仮想端末回線パスワードなど、すべてのパスワードに適用されます。

パスワードの回復

スイッチに物理的にアクセスできるエンドユーザは、デフォルトで、スイッチの電源投入時にブートプロセスに割り込み、新しいパスワードを入力することによって、失われたパスワードを回復できます。

パスワード回復ディセーブル化機能では、この機能の一部をディセーブルにすることによりスイッチのパスワードへのアクセスを保護できます。この機能がイネーブルの場合、エンドユーザは、システムをデフォルト設定に戻すことに同意した場合に限り、ブートプロセスに割り込むことができます。パスワード回復をディセーブルにしても、ブートプロセスに割り込んでパスワードを変更できますが、コンフィギュレーション ファイル (config.text) および VLAN データベース ファイル (vlan.dat) は削除されます。

パスワード回復をディセーブルにする場合は、エンドユーザがブートプロセスに割り込んでシステムをデフォルトの状態に戻すような場合に備え、セキュア サーバにコンフィギュレーション ファイルのバックアップ コピーを保存しておくことを推奨します。スイッチ上でコンフィギュレーション ファイルのバックアップ コピーを保存しないでください。VTP (VLAN トランッキング プロトコル) トランスペアレント モードでスイッチが動作している場合は、VLAN データベース ファイルのバックアップ コピーも同様にセキュア サーバに保存してくだ

さい。スイッチがシステムのデフォルト設定に戻ったときに、XMODEM プロトコルを使用して、保存したファイルをスイッチにダウンロードできます。

パスワードの回復を再びイネーブルにするには、**no system disable password recovery switch number | all** グローバル コンフィギュレーション コマンドを使用します。

端末回線の Telnet 設定

初めてスイッチに電源を投入すると、自動セットアッププログラムが起動して IP 情報を割り当て、この後続けて使用できるようにデフォルト設定を作成します。さらに、セットアッププログラムは、パスワードによる Telnet アクセス用にスイッチを設定することを要求します。セットアッププログラムの実行中にこのパスワードを設定しなかった場合は、端末回線に対する Telnet パスワードを設定するときに設定できます。

ユーザ名とパスワードのペア

ユーザ名とパスワードのペアを設定できます。このペアはスイッチ上でローカルに保存されます。このペアは回線またはポートに割り当てられ、各ユーザを認証します。ユーザは認証後、スイッチにアクセスできます。権限レベルを定義している場合は、ユーザ名とパスワードの各ペアに特定の権限レベルを、対応する権利および権限とともに割り当てることもできます。

権限レベル

シスコデバイスでは、権限レベルを使用して、スイッチ動作の異なるレベルに対してパスワードセキュリティを提供します。デフォルトでは、Cisco IOS XE ソフトウェアは、パスワードセキュリティの2つのモード（権限レベル）で動作します。ユーザー EXEC（レベル 1）および特権 EXEC（レベル 15）です。各モードに、最大 16 個の階層レベルからなるコマンドを設定できます。複数のパスワードを設定することにより、ユーザグループ別に特定のコマンドへのアクセスを許可することができます。

回線の権限レベル

ユーザーは、回線にログインし、別の権限レベルを有効に設定することにより、**privilege level** ラインコンフィギュレーションコマンドを使用して設定された権限レベルを上書きできます。また、**disable** コマンドを使用することにより、権限レベルを引き下げることができます。上位の権限レベルのパスワードがわかっている場合、ユーザはそのパスワードを使用して上位の権限レベルをイネーブルにできます。回線の使用を制限するには、コンソール回線に高いレベルまたは権限レベルを指定してください。

たとえば、多くのユーザに **clear line** コマンドへのアクセスを許可する場合、レベル 2 のセキュリティを割り当て、レベル 2 のパスワードを広範囲のユーザに配布できます。また、**configure** コマンドへのアクセス制限を強化する場合は、レベル 3 のセキュリティを割り当て、そのパスワードを限られたユーザグループに配布することもできます。

コマンド権限レベル

コマンドをある権限レベルに設定すると、構文がそのコマンドのサブセットであるコマンドはすべて、そのレベルに設定されます。たとえば、**show ip traffic** コマンドをレベル 15 に設定すると、**show** コマンドと **show ip** コマンドは、異なるレベルに個別に設定しない限り、権限レベルは自動的に 15 に設定されます。

AES パスワード暗号化およびマスター暗号キー

強力で、反転可能な 128 ビットの高度暗号化規格 (AES) パスワード暗号化 (タイプ 6 暗号化ともいう) を有効にできます。タイプ 6 暗号化の使用を開始するには、AES パスワード暗号化機能を有効にし、パスワードを暗号化および復号するためのマスター暗号キーを設定します。

AES パスワード暗号化を有効にしてマスターキーを設定すると、タイプ 6 パスワード暗号化を無効にしない限り、サポートされているアプリケーションの既存および新規作成されたクリアテキストパスワードがすべて、タイプ 6 暗号化の形式で保存されます。また、既存の弱いすべての暗号化パスワードをタイプ 6 暗号化パスワードに変換するようにデバイスを設定することもできます。

AES パスワード暗号化機能とマスター暗号キーが設定されている場合、タイプ 0 および 7 のパスワードはタイプ 6 に自動変換できます。



- (注)
- ユーザー名パスワードのタイプ 6 の暗号化パスワードは、Cisco IOS XE Gibraltar 16.10.1 以降のリリースでサポートされています。パスワードタイプ 6 への自動変換は、Cisco IOS XE Gibraltar 16.11.1 以降のリリースでサポートされています。
 - タイプ 6 のユーザー名とパスワードには Cisco IOS XE Gibraltar 16.10.x と下位互換性があります。Cisco IOS XE Gibraltar 16.10.1 より前のリリースにダウングレードすると、タイプ 6 のユーザー名とパスワードは拒否されます。自動変換後、管理者パスワードがダウングレード中に拒否されないようにするには、管理者ログイン (管理アクセス) に使用されるパスワードを不可逆的なパスワードタイプに手動で移行します。

パスワードおよび権限によるスイッチアクセスの設定方法

スタティック 有効パスワードの設定または変更

イネーブルパスワードは、特権 EXEC モードへのアクセスを制御します。スタティックイネーブルパスワードを設定または変更するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **enable password** *password*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	enable password <i>password</i> Example: Device(config)# enable password secret321	特権 EXEC モードにアクセスするための新しいパスワードを定義するか、既存のパスワードを変更します。 デフォルトでは、パスワードは定義されません。 <i>password</i> には、1～25 文字の英数字のストリングを指定します。ストリングを数字で始めることはできません。大文字と小文字を区別し、スペースを使用できますが、先行スペースは無視されます。疑問符 (?) は、パスワードを作成する場合に、疑問符の前に Ctrl+v を入力すれば使用できます。たとえば、パスワード abc?123 を作成するときは、次のようになります。 a. abc を入力します。 b. Ctrl+v を入力します。 c. ?123 を入力します。 システムからイネーブルパスワードを入力するように求められた場合、疑問符の前に Ctrl+v を入力する必要はなく、パスワードのプロンプトにそのまま abc?123 と入力できます。
ステップ 4	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

暗号化によるイネーブルおよびイネーブル シークレット パスワードの保護

特権 EXEC モード（デフォルト）または指定された特権レベルにアクセスするためにユーザーが入力する必要がある暗号化パスワードを確立するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. 次のいずれかを使用します。
 - **enable password [level level] {unencrypted-password | encryption-type encrypted-password}**
 - **enable secret [level level] {unencrypted-password | encryption-type encrypted-password}**
4. **service password-encryption**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかを使用します。 <ul style="list-style-type: none"> • enable password [level level] {unencrypted-password encryption-type encrypted-password} • enable secret [level level] {unencrypted-password encryption-type encrypted-password} Example: Device(config)# enable password level 12 example123 または Device(config)# enable secret 9 \$9\$sMLBsTFXLnnHTk\$0L82	<ul style="list-style-type: none"> • 特権 EXEC モードにアクセスするための新しいパスワードを定義するか、既存のパスワードを変更します。 • シークレットパスワードを定義します。これは非可逆的な暗号化方式を使用して保存されます。 <ul style="list-style-type: none"> • （任意）<i>level</i> に指定できる範囲は 0 ～ 15 です。レベル 1 が通常のユーザ EXEC モード権限です。デフォルト レベルは 15 です（特権 EXEC モード権限）。 • <i>unencrypted-password</i> には、1 ～ 25 文字の英数字の文字列を指定します。ストリングを数字で始めることはできません。大文字と小文字を区別し、スペースを使用できま

	Command or Action	Purpose
		<p>すが、先行スペースは無視されます。デフォルトでは、パスワードは定義されません。</p> <ul style="list-style-type: none">• <i>encryption-type</i> の場合、enable password に使用可能なオプションはタイプ 0 と 7、enable secret に使用可能なオプションはタイプ 0、5、8、および 9 です。暗号化タイプを指定する場合は、暗号化されたパスワードを使用する必要があります。この暗号化パスワードは、別のスイッチの設定からコピーします。シークレット暗号化タイプ 9 はより安全であるため、アップグレードまたはダウングレード時に問題が発生しないように、タイプ 9 を選択することを推奨します。

	Command or Action	Purpose
		<p>Note</p> <ul style="list-style-type: none"> シークレットパスワードの暗号化タイプを指定しない場合、パスワードはタイプ 9 に自動的に変換されます。これは、Cisco IOS XE Gibraltar 16.10.1 以降のリリースで適用されます。 暗号化タイプを指定してクリアテキストパスワードを入力した場合は、エラーが発生します。 グローバル コンフィギュレーション モードで algorithm-type script コマンドを使用して、シークレットパスワードにタイプ 9 暗号化を手動で設定することもできます。次に例を示します。 <pre>Device(config)# username user1 algorithm-type script secret cisco</pre> <p>または</p> <pre>Device(config)# enable algorithm-type script secret cisco</pre> <p>特権 EXEC モードで write memory コマンドを実行し、タイプ 9 シークレットをスタートアップコンフィギュレーションに永続的に書き込みます。</p>
ステップ 4	<p>service password-encryption</p> <p>Example:</p> <pre>Device(config)# service password-encryption</pre>	<p>(任意) パスワードの定義時または設定の書き込み時に、パスワードを暗号化します。</p> <p>暗号化を行うと、コンフィギュレーションファイル内でパスワードが読み取り可能な形式になるのを防止できます。</p>
ステップ 5	<p>end</p> <p>Example:</p>	<p>グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

	Command or Action	Purpose
	Device (config)# end	

パスワード回復のディセーブル化

パスワードの回復をディセーブルにしてスイッチのセキュリティを保護するには、次の手順を実行します。

始める前に

パスワード回復をディセーブルにする場合は、エンドユーザがブートプロセスに割り込んでシステムをデフォルトの状態に戻すような場合に備え、セキュアサーバにコンフィギュレーションファイルのバックアップコピーを保存しておくことを推奨します。スイッチ上でコンフィギュレーションファイルのバックアップコピーを保存しないでください。VTP (VLAN トランッキング プロトコル) トランスペアレント モードでスイッチが動作している場合は、VLAN データベース ファイルのバックアップコピーも同様にセキュアサーバに保存してください。スイッチがシステムのデフォルト設定に戻ったときに、XMODEM プロトコルを使用して、保存したファイルをスイッチにダウンロードできます。

手順の概要

1. **enable**
2. **configure terminal**
3. **system disable password recovery switch {all | <1-9>}**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	system disable password recovery switch {all <1-9>} 例： Device (config)# system disable password recovery switch all	パスワード回復をディセーブルにします。 <ul style="list-style-type: none"> • <i>all</i> : スタック内のスイッチで設定を行います。 • <i><1-9></i> : 選択したスイッチ番号で設定を行います。

	コマンドまたはアクション	目的
		この設定は、フラッシュメモリの中で、ブートローダおよび Cisco IOS イメージがアクセスできる領域に保存されますが、ファイルシステムには含まれません。また、ユーザーがアクセスすることはできません。
ステップ 4	end 例 : Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

次のタスク

disable password recovery を削除するには、**no system disable password recovery switch all** グローバル コンフィギュレーション コマンドを使用します。

端末回線に対する Telnet パスワードの設定

接続された端末回線に対する Telnet パスワードを設定するには、ユーザー EXEC モードで次の手順を実行します。

Before you begin

- エミュレーション ソフトウェアを備えた PC またはワークステーションをスイッチ コンソール ポートに接続するか、または PC をイーサネット管理ポートに接続します。
- コンソール ポートのデフォルトのデータ特性は、9600 ボー、8 データ ビット、1 ストップ ビット、パリティなしです。コマンドラインプロンプトが表示されるまで、Return キーを何回か押す必要があります。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty 0 98**
4. **password *password***
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。

	Command or Action	Purpose
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	line vty 0 98 Example: Device(config)# line vty 0 98	Telnet セッション（回線）の数を設定し、ライン コンフィギュレーション モードを開始します。 コマンド対応 device では、最大 99 のセッションが可能です。0 および 98 は、可能なすべての 99 Telnet セッションを設定していることを意味します。
ステップ 4	password password Example: Device(config-line)# password abcxyz543	1 つまたは複数の回線に対応する Telnet パスワードを設定します。 <i>password</i> には、1 ～ 25 文字の英数字のストリングを指定します。ストリングを数字で始めることはできません。大文字と小文字を区別し、スペースを使用できますが、先行スペースは無視されます。デフォルトでは、パスワードは定義されません。
ステップ 5	end Example: Device(config-line)# end	特権 EXEC モードに戻ります。

ユーザ名とパスワードのペアの設定

ユーザー名とパスワードのペアを設定するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **username name [privilege level] { password encryption-type password }**
4. 次のいずれかを使用します。
 - **line console 0**
 - **line vty 0 98**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example:	特権 EXEC モードを有効にします。

	Command or Action	Purpose
	Device> enable	プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	username name [privilege level] { password encryption-type password} Example: Device(config)# username adamsample privilege 1 password secret456 Device(config)# username 111111111111 mac attribute	各ユーザのユーザ名、権限レベル、パスワードを設定します。 <ul style="list-style-type: none"> • <i>name</i> には、ユーザー ID を 1 ワードで指定するか、または MAC アドレスを指定します。スペースと引用符は使用できません。 • ユーザ名と MAC フィルタの両方に対し、最大 12000 のクライアントを個別に設定できます。 • (任意) <i>level</i> には、アクセス権を得たユーザーに設定する権限レベルを指定します。指定できる範囲は 0 ~ 15 です。レベル 15 では特権 EXEC モードでのアクセスが可能です。レベル 1 では、ユーザ EXEC モードでのアクセスとなります。 • <i>encryption-type</i> に、暗号化されていないパスワードが後ろに続く場合は 0 を入力します。非表示パスワードが後ろに続く場合は 7 を入力します。暗号化されたパスワードが後ろに続く場合は 6 を入力します。 • <i>password</i> には、デバイスにアクセスするためにユーザーが入力する必要があるパスワードを指定します。パスワードは 1 ~ 25 文字で、埋め込みスペースを使用でき、username コマンドの最後のオプションとして指定します。
ステップ 4	次のいずれかを使用します。 <ul style="list-style-type: none"> • line console 0 • line vty 0 98 Example: Device(config)# line console 0 または Device(config)# line vty 0 98	ライン コンフィギュレーション モードを開始し、コンソールポート (回線 0) または VTY 回線 (回線 0 ~ 98) を設定します。

	Command or Action	Purpose
ステップ 5	end Example: Device(config-line)# end	回線コンフィギュレーションモードを終了します。 続いて、特権 EXEC モードに戻ります。

コマンドの特権レベルの設定

コマンドの権限レベルを設定するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **privilege mode level level command**
4. **enable password level level password**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	privilege mode level level command Example: Device(config)# privilege exec level 14 configure	コマンドの特権レベルを設定します。 <ul style="list-style-type: none"> • <i>mode</i> には、グローバル コンフィギュレーションモードの場合は configure を、EXEC モードの場合は exec を、インターフェイス コンフィギュレーションモードの場合は interface を、ライン コンフィギュレーションモードの場合は line をそれぞれ入力します。 • <i>level</i> の範囲は 0 ~ 15 です。レベル 1 が通常のユーザー EXEC モード権限です。レベル 15 は、enable パスワードによって許可されるアクセスレベルです。 • <i>command</i> には、アクセスを制限したいコマンドを指定します。

	Command or Action	Purpose
ステップ 4	enable password level <i>level password</i> Example: <pre>Device(config)# enable password level 14 SecretPswd14</pre>	権限レベルをイネーブルにするためのパスワードを指定します。 <ul style="list-style-type: none"> • <i>level</i> の範囲は 0 ~ 15 です。レベル 1 が通常のユーザー EXEC モード権限です。 • <i>password</i> には、1 ~ 25 文字の英数字のストリングを指定します。ストリングを数字で始めることはできません。大文字と小文字を区別し、スペースを使用できますが、先行スペースは無視されます。デフォルトでは、パスワードは定義されません。
ステップ 5	end Example: <pre>Device(config)# end</pre>	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

回線のデフォルト特権レベルの変更

指定した回線のデフォルトの権限レベルを変更するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty *line***
4. **privilege exec level *level***
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: <pre>Device> enable</pre>	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: <pre>Device# configure terminal</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	line vty <i>line</i> Example:	アクセスを制限する仮想端末回線を選択します。

	Command or Action	Purpose
	Device(config)# line vty 10	
ステップ 4	privilege exec level level Example: Device(config-line)# privilege exec level 15	回線のデフォルト特権レベルを変更します。 level の範囲は 0 ～ 15 です。レベル 1 が通常のユーザー EXEC モード権限です。レベル 15 は、 enable パスワードによって許可されるアクセス レベルです。
ステップ 5	end Example: Device(config-line)# end	回線コンフィギュレーションモードを終了します。続いて、特権 EXEC モードに戻ります。

What to do next

ユーザーは、回線にログインし、別の権限レベルを有効に設定することにより、**privilege level** ラインコンフィギュレーションコマンドを使用して設定された権限レベルを上書きできます。また、**disable** コマンドを使用することにより、権限レベルを引き下げることができます。上位の権限レベルのパスワードがわかっている場合、ユーザーはそのパスワードを使用して上位の権限レベルをイネーブルにできます。回線の使用を制限するには、コンソール回線に高いレベルまたは権限レベルを指定してください。

権限レベルへのログインおよび終了

指定した権限レベルにログインする、または指定した権限レベルを終了するには、ユーザー EXEC モードで次の手順を実行します。

SUMMARY STEPS

1. **enable level**
2. **disable level**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable level Example: Device> enable 15	指定された特権レベルにログインします。 この例では、レベル 15 は特権 EXEC モードです。 level に指定できる範囲は 0 ～ 15 です。
ステップ 2	disable level Example:	指定した特権レベルを終了します。

	Command or Action	Purpose
	Device# <code>disable 1</code>	この例では、レベル 1 はユーザー EXEC モードです。 <i>level</i> に指定できる範囲は 0 ~ 15 です。

暗号化事前共有キーの設定

暗号化事前共有キーを設定するには、次の手順を実行します。

手順の概要

1. `enable`
2. `configure terminal`
3. `key config-key password-encrypt [text]`
4. `password encryption aes`
5. `end`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> <code>enable</code>	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# <code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	key config-key password-encrypt [text] 例： Device(config)# <code>key config-key password-encrypt</code>	タイプ 6 の暗号キーをプライベート NVRAM に保存します。 <ul style="list-style-type: none"> • (Enter キーを使用して) インタラクティブにキーボード操作を行う場合、暗号キーがすでに存在すれば、Old key、New key、Confirm key という 3 つのプロンプトが表示されます。 • インタラクティブにキーボード操作を行う場合、暗号キーが存在しなければ、New key、Confirm key という 2 つのプロンプトが表示されます。 • すでに暗号化されているパスワードを削除しようとする、次のプロンプトが表示されます。

	コマンドまたはアクション	目的
		WARNING: All type 6 encrypted keys will become unusable. Continue with master key deletion? [yes/no]:"
ステップ 4	password encryption aes 例 : Device(config)# password encryption aes	暗号化事前共有キーのイネーブル化
ステップ 5	end 例 : Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

パスワードおよび権限によるスイッチアクセスのモニター

表 2: 特権レベル情報を表示するためのコマンド

コマンド	情報
show privilege	権限レベルの設定を表示します。

パスワードおよび権限レベルによるスイッチアクセスの設定例

例 : スタティック イネーブルパスワードの設定または変更

次に、イネーブルパスワードを `11u2c3k4y5` に変更する例を示します。パスワードは暗号化されず、レベル 15 のアクセスが与えられます (従来の特権 EXEC モードアクセス)。

```
Device> enable
Device# configure terminal
Device(config)# enable password 11u2c3k4y5
Device(config)# end
```

例：暗号化によるイネーブルおよびイネーブルシークレットパスワードの保護

次に、権限レベル 2 に対して暗号化パスワード `9sMLBsTFXLnnHTk$0L82` を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# enable secret level 2 9 $9$sMLBsTFXLnnHTk$0L82
Device(config)# end
```

例：端末回線に対する Telnet パスワードの設定

次に、Telnet パスワードを `let45me67in89` に設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# line vty 10
Device(config-line)# password let45me67in89
Device(config-line)# end
```

例：コマンドの権限レベルの設定

ここで、`configure` コマンドを権限レベル 14 に設定する方法、レベル 14 のコマンドを使用する場合にユーザーが入力するパスワードとして `SecretPswd14` を定義する方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# privilege exec level 14 configure
Device(config)# enable password level 14 SecretPswd14
Device(config)# end
```

例：暗号化事前共有キーの設定

以下に、タイプ 6 の事前共有キーに暗号化を行った場合の設定例を示します。この中には、ユーザに対して表示されるプロンプトやメッセージも含まれています。

```
Device> enable
Device# configure terminal
Device(config)# password encryption aes
Device(config)# key config-key password-encrypt
New key:
Confirm key:
Device(config)#
01:46:40: TYPE6_PASS: New Master key configured, encrypting the keys with
the new master key
Device(config)# end
```

パスワードおよび権限によるスイッチ アクセスの制御の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	パスワードおよび権限によるスイッチ アクセスの制御	パスワード保護によって、ネットワークまたはネットワークデバイスへのアクセスが制限されます。権限レベルによって、ネットワークデバイスにログイン後、ユーザがどのようなコマンドを使用できるかが定義されます。
Cisco IOS XE Gibraltar 16.11.1	タイプ0およびタイプ7のユーザー名とパスワードのタイプ6への自動変換	このリリース以降は、タイプ0および7のユーザー名とパスワードをタイプ6に自動変換できます。

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



第 2 章

認証の設定

認証は、選択したセキュリティプロトコルに応じてログイン/パスワードダイアログ、チャレンジ/レスポンス、メッセージングサポート、および暗号化などのユーザーの識別方法を提供します。認証は、ユーザーに対してネットワークとネットワークサービスへのアクセスを許可する前に、ユーザーの識別を行う方法です。

- [認証の設定の前提条件 \(23 ページ\)](#)
- [認証の設定に関する制約事項 \(23 ページ\)](#)
- [認証について \(24 ページ\)](#)
- [認証の設定方法 \(44 ページ\)](#)
- [認証の設定例 \(67 ページ\)](#)
- [認証設定の機能履歴 \(82 ページ\)](#)

認証の設定の前提条件

認証の実装は、認証、許可、およびアカウンティング (AAA) 認証と非認証方式に分かれています。シスコでは、可能であれば AAA セキュリティ サービスを試用して認証を実装することを推奨します。

認証の設定に関する制約事項

- 設定できる AAA 方式リストの数は 250 です。
- **acct-port** キーワードを使用してアカウンティング要求と異なる UDP 宛先ポートに、および非標準オプションの有無に関係なく **auth-port** キーワードを使用して認証要求の UDP 宛先ポートに同じ RADIUS サーバーの IP アドレスを設定した場合、RADIUS サーバーは非標準オプションを受け入れません。

認証について

認証の名前付き方式リスト

まず認証方式の名前付きリストを定義して AAA 認証を設定し、その名前付きリストを各種インターフェイスに適用します。この方式リストは、認証のタイプと実行順序を定義したものです。定義されたいずれかの認証方式を実行するには、この方式リストを特定のインターフェイスに適用する必要があります。唯一の例外は、デフォルトの方式リスト（「default」という名前）です。デフォルトの方式リストは、明示的に定義された名前付きの方式リストを持つインターフェイスを除くすべてのインターフェイスに自動的に適用されます。定義済みの方式リストは、デフォルトの方式リストに優先します。

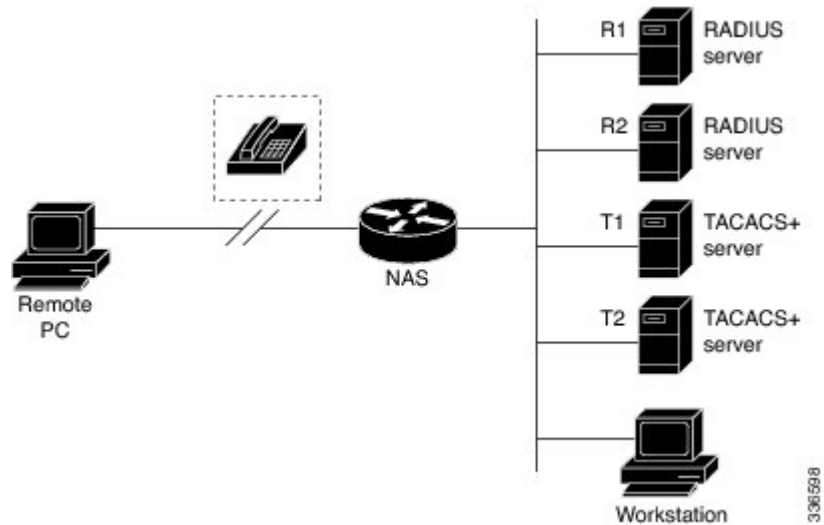
方式リストとは、ユーザー認証のために照会される認証方式を記述したシーケンシャルリストです。方式リストを使用すると、認証に使用するセキュリティプロトコルを1つまたは複数指定できるため、最初の方式が失敗した場合に備えて認証のバックアップシステムを確保できます。シスコソフトウェアは、ユーザーを認証するため、リストに記載されている最初の方式が使用されます。その方式で応答に失敗した場合、シスコソフトウェアは、方式リストに記載されている次の認証方式を選択します。このプロセスは、方式リストのいずれかの認証方式と通信に成功するか、定義されているすべての方式が試行されるまで継続されます。

このソフトウェアでは、前の方式からの応答がない場合にだけ、リストの次の認証方式で認証が試行される、という点に注意してください。このサイクルのいずれかの時点で認証に失敗した場合、つまりセキュリティサーバーまたはローカルユーザー名データベースからユーザーアクセスを拒否する応答があった場合には、許可プロセスが停止し、それ以上の認証方式は試行されません。

方式リストとサーバグループ

サーバグループは、方式リストに使用する既存の RADIUS または TACACS+ サーバ ホストをグループ化する方法の1つです。次の図に、4台のセキュリティサーバー（R1 と R2 は RADIUS サーバ、T1 と T2 は TACACS+ サーバ）が設置された一般的な AAA ネットワーク設定を示します。R1 と R2 で RADIUS サーバのグループを構成します。T1 と T2 で TACACS+ サーバのグループを構成します。

図 1: 一般的な AAA ネットワーク設定



サーバーグループを使用して、設定したサーバーホストのサブセットを指定し、特定のサービスに使用します。たとえば、サーバーグループを使用すると、R1 および R2 を一つのサーバーグループとして定義し、T1 および T2 を別のサーバーグループとして定義できます。また、認証ログインの方式リストに R1 および T1 を指定し、PPP 認証の方式リストに R2 および T2 を指定することもできます。

サーバーグループには、1台のサーバに対して複数のホストエントリを含めることができます。エントリごとに固有の識別情報を設定します。固有の識別情報は、IP アドレスと UDP ポート番号の組み合わせで構成されます。これにより、RADIUS ホストとして定義されているさまざまなポートが、固有の AAA サービスを提供できるようになります。つまり、この固有識別情報を使用して、ある IP アドレスに位置する 1 台のサーバ上に複数の UDP ポートが存在する場合、それぞれの UDP ポートに対して RADIUS 要求を送信できます。1 台の RADIUS サーバ上にある異なる 2 つのホストエントリが 1 つのサービス（認証など）に設定されている場合、設定されている 2 番目のホストエントリは最初のホストエントリのフェールオーバーバックアップとして動作します。この例の場合、最初のホストエントリがアカウントサービスを提供に失敗すると、同じデバイスに設定されている 2 番目のホストエントリを使用してアカウントサービスを提供するように、ネットワークアクセスサーバが試行します（試行される RADIUS ホストエントリの順番は、設定されている順序に従います）。

サーバーグループの設定および着信番号識別サービス（DNIS）番号に基づくサーバーグループの設定の詳細については、「RADIUS の設定」または「TACACS+ の設定」を参照してください。

AAA によるログイン認証

イネーブルパスワードによるログイン認証

認証方式としてイネーブルパスワードを指定するには、**enable** キーワードを指定して **aaa authentication login** コマンドを使用します。たとえば、他の方式リストが定義されていない場

合にログイン時のユーザー認証方式としてイネーブルパスワードを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default enable
```

ログイン認証方式としてイネーブルパスワードを使用するには、イネーブルパスワードを定義しておく必要があります。イネーブルパスワードの定義の詳細については、「パスワードおよび権限レベルによるスイッチアクセスの制御」を参照してください。

Kerberos によるログイン認証

Kerberos による認証は、他のほとんどの認証方式とは異なり、ユーザーのパスワードはリモート アクセス サーバーに送信されません。ネットワークにログインするリモート ユーザーは、ユーザー名の指定を求められます。ユーザーのエントリがキー発行局 (KDC) に存在する場合は、そのユーザーのパスワードを含む暗号化されたチケット認可チケット (TGT) が作成され、デバイスに送信されます。次に、ユーザーにパスワードの入力が求められ、デバイスではそのパスワードで TGT の復号が試行されます。復号に成功すると、ユーザーは認証され、デバイス上にあるユーザーのクレデンシャルキャッシュに TGT が保存されます。

krb5 は KINIT プログラムを使用しますが、デバイスを認証するために、ユーザーが KINIT プログラムを実行して TGT を取得する必要はありません。これは、Cisco IOS XE の Kerberos 実装のログイン手順に KINIT が統合されているためです。

ログイン認証方式として Kerberos を指定するには、**krb5** キーワードを指定して **aaa authentication login** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にログイン時のユーザー認証方式として Kerberos を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default krb5
```

ログイン認証方式として Kerberos を使用するには、Kerberos セキュリティ サーバーとの通信をイネーブルにしておく必要があります。Kerberos サーバーとの通信を確立する方法の詳細については、「Kerberos の設定」の章を参照してください。

ラインパスワードによるログイン認証

ログイン認証方式としてラインパスワードを指定するには、**line** キーワードを指定して **aaa authentication login default** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にログイン時のユーザー認証方式としてラインパスワードを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default line
```

ログイン認証方式としてラインパスワードを使用するには、ラインパスワードを定義しておく必要があります。

ローカルパスワードによるログイン認証

シスコデバイスが認証にローカルユーザー名データベースを使用するように指定するには **aaa authentication login default** コマンドに **local** キーワードを指定して使用します。たとえば、他

の方式リストが定義されていない場合にログイン時のユーザー認証方式としてローカルユーザー名データベースを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default local
```

group RADIUS によるログイン認証

ログイン認証方式として RADIUS を指定するには、**group radius** を指定して **aaa authentication login default** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にログイン時のユーザー認証方式として RADIUS を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default group radius
```

ログイン認証方式として RADIUS を使用するには、RADIUS セキュリティ サーバーとの通信をイネーブルにしておく必要があります。RADIUS サーバーとの通信を確立する方法の詳細については、「RADIUS の設定」の章を参照してください。

アクセス要求内の RADIUS 属性 8

aaa authentication login コマンドを使用して RADIUS を指定し、NAS から IP アドレスを要求するようにログインホストを設定すると、グローバル コンフィギュレーション モードで **radius-server attribute 8 include-in-access-req** コマンドを使用して、**access-request** パケットで属性 8 (Framed-IP-Address) を送信できます。このコマンドによって、ユーザー認証の前に、NAS から RADIUS サーバーに対してユーザー IP アドレスのヒントを提供できます。

group TACACS によるログイン認証

ログイン認証方式として TACACS+ を指定するには、**group tacacs+** を指定して **aaa authentication login default** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にログイン時のユーザー認証方式として TACACS+ を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default group tacacs+
```

ログイン認証方式として TACACS+ を使用するには、TACACS+ セキュリティ サーバーとの通信をイネーブルにしておく必要があります。TACACS+ サーバーとの通信を確立する方法の詳細については、「TACACS+ の設定」の章を参照してください。

グループ名によるログイン認証

ログイン認証方式として使用する RADIUS または TACACS+ サーバーのサブセットを指定するには、**group group-name** 方式を指定して **aaa authentication login default** コマンドを使用します。グループ名とそのグループのメンバを指定して定義するには、**aaa group server** コマンドを使用します。たとえば、**aaa group server** コマンドを使用して、**group loginrad** のメンバを最初に定義します。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius loginrad
Device(config-sg-radius)# server 172.16.2.3
Device(config-sg-radius)# server 172.16.2.17
```

```
Device(config-sg-radius)# server 172.16.2.32
Device(config-sg-radius)# end
```

このコマンドにより、172.16.2.3、172.16.2.17、172.16.2.32 の RADIUS サーバーがグループ *loginrad* のメンバとして指定されます。

他の方式リストが定義されていない場合にログイン時のユーザー認証方式として **group loginrad** を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default group loginrad
```

ログイン認証方式としてグループ名を使用するには、RADIUS または TACACS+ セキュリティ サーバーとの通信をイネーブルにしておく必要があります。RADIUS サーバーとの通信を確立する方法の詳細については、「RADIUS の設定」の章を参照してください。TACACS+ サーバーとの通信を確立する方法の詳細については、「TACACS+ の設定」の章を参照してください。

AAA による PPP 認証

Kerberos による PPP 認証

PPP を実行するインターフェイスで使用する認証方式として Kerberos を指定するには、**krb5** キーワードを指定して **aaa authentication ppp default Device** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にユーザー認証方式として Kerberos を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication ppp default krb5
```

PPP 認証方式として Kerberos を使用するには、Kerberos セキュリティ サーバーとの通信をイネーブルにしておく必要があります。Kerberos サーバーとの通信を確立する方法の詳細については、「Kerberos の設定」の章を参照してください。



(注) Kerberos ログイン認証は、PPP PAP 認証とだけ連携します。

ローカルパスワードによる PPP 認証

シスコデバイスが認証にローカルユーザー名データベースを使用するように指定するには **aaa authentication ppp default** コマンドに **local** キーワードを指定して使用します。たとえば、他の方式リストが定義されていない場合に、PPP を実行する回線に使用するユーザー認証方式としてローカルユーザー名データベースを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication ppp default local
```

group RADIUS による PPP 認証

ログイン認証方式として RADIUS を指定するには、**aaa authentication ppp default group radius** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にログイン時のユーザー認証方式として RADIUS を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication ppp default group radius
```

PPP 認証方式として RADIUS を使用するには、RADIUS セキュリティ サーバーとの通信をイネーブルにしておく必要があります。RADIUS サーバーとの通信を確立する方法の詳細については、「RADIUS の設定」の章を参照してください。

アクセス要求内の RADIUS 属性 44

aaa authentication ppp default group radius コマンドを使用して、RADIUS をログイン認証方式として指定すると、グローバル コンフィギュレーション モードで **radius-server attribute 44 include-in-access-req** コマンドを使用して **access-request** パケットで属性 44 (Acct-Session-ID) を送信するようにデバイスを設定できます。このコマンドによって、RADIUS デーモンはコールを開始から終了まで追跡できます。

group TACACS による PPP 認証

ログイン認証方式として TACACS+ を指定するには、**aaa authentication ppp default group tacacs+** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にログイン時のユーザー認証方式として TACACS+ を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication ppp default group tacacs+
```

PPP 認証方式として TACACS+ を使用するには、TACACS+ セキュリティ サーバーとの通信をイネーブルにしておく必要があります。TACACS+ サーバーとの通信を確立する方法の詳細については、「TACACS+ の設定」の章を参照してください。

グループ名による PPP 認証

ログイン認証方式として使用する RADIUS または TACACS+ サーバーのサブセットを指定するには、**group group-name** 方式を指定して **aaa authentication ppp default** コマンドを使用します。グループ名とそのグループのメンバを指定して定義するには、**aaa group server** コマンドを使用します。たとえば、**aaa group server** コマンドを使用して、**group ppprad** のメンバを最初に定義します。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius ppprad
Device(config-sg-radius)# server 172.16.2.3
Device(config-sg-radius)# server 172.16.2.17
Device(config-sg-radius)# server 172.16.2.32
Device(config-sg-radius)# end
```

このコマンドにより、172.16.2.3、172.16.2.17、172.16.2.32 の RADIUS サーバーがグループ **ppprad** のメンバとして指定されます。

他の方式リストが定義されていない場合にログイン時のユーザー認証方式として **group ppprad** を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication ppp default group ppprad
```

PPP 認証方式としてグループ名を使用するには、RADIUS または TACACS+ セキュリティ サーバーとの通信をイネーブルにしておく必要があります。RADIUS サーバーとの通信を確立する方法の詳細については、「RADIUS の設定」の章を参照してください。TACACS+ サーバーとの通信を確立する方法の詳細については、「TACACS+ の設定」の章を参照してください。

PPP 要求の AAA スケーラビリティ

ネットワークアクセスサーバー (NAS) の PPP マネージャによって割り当てられた複数のバックグラウンドプロセスを設定およびモニターして、AAA 認証要求と認可要求に対応できます。AAA スケーラビリティ機能によって、PPP に対する AAA 要求を処理するために使用される複数のプロセスを設定できるようになります。つまり、同時に認証または認可できるユーザー数が増えます。

PPP に対する AAA 要求を処理するために、特定の数のバックグラウンドプロセスを割り当てるには、グローバル 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。

```
Device(config)# aaa processes 5000
```

引数 *number* には、PPP に対する AAA 認証要求と認可要求を処理するために確保するバックグラウンドプロセス数を定義します。また、1 ~ 2147483647 の任意の値を設定できます。PPP マネージャが PPP に対する要求を処理する方法のため、この引数には、同時に認証できる新規ユーザーの数も定義します。この引数は、いつでも増減できます。



(注) 追加バックグラウンドプロセスの割り当ては、コストが高くなる可能性があります。PPP に対する AAA 要求を処理できるバックグラウンドプロセスの最小数を設定してください。

AAA による ARAP 認証

認可済みゲスト ログインを許可する ARAP 認証

ユーザーが EXEC に正常にログイン済みの場合にだけ、ゲストログインを許可するには、**auth-guest** キーワードを指定して **aaa authentication arap default** コマンドを使用します。この方式は ARAP 認証方式リストの先頭に指定する必要がありますが、この方式が成功しなかった場合は引き続き他の方式を試行できます。たとえば、認証のデフォルト方式として、すべての認可済みゲストログイン（つまり、EXEC にログイン済みのユーザーによるログイン）を許可し、その方式が失敗した場合にだけ RADIUS を使用するには、次のコマンドを入力します。

```
Device(config)# aaa authentication arap default auth-guest group radius
```



- (注) AAA を初期化すると、デフォルトで ARAP によるゲスト ログインはディセーブルになります。ゲストログインを許可するには、**guest** キーワードまたは **auth-guest** キーワードを指定して **aaa authentication arap {authentication-list | default}** コマンドを使用する必要があります。

ゲスト ログインを許可する ARAP 認証

ゲストログインを許可するには、**guest** キーワードを指定して **aaa authentication arap {default | authentication-list}** コマンドを使用します。この方式は ARAP 認証方式リストの先頭に指定する必要がありますが、この方式が成功しなかった場合は引き続き他の方式を試行できます。たとえば、認証のデフォルト方式としてすべてのゲストログインを許可し、その方式が失敗した場合にだけ RADIUS を使用するには、次のコマンドを入力します。

```
Device(config)# aaa authentication arap default guest group radius
```

ラインパスワードによる ARAP 認証

認証方式としてラインパスワードを指定するには、**line** キーワードを指定して **aaa authentication arap {default | authentication-list}** コマンドを使用します。たとえば、他の方式リストが定義されていない場合に、ARAP ユーザー認証方式としてラインパスワードを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication arap default line
```

ARAP 認証方式としてラインパスワードを使用するには、ラインパスワードを定義しておく必要があります。

ローカルパスワードによる ARAP 認証

Cisco デバイスが認証にローカルユーザー名データベースを使用するように指定するには **aaa authentication arap {default | authentication-list}** コマンドに **local** キーワードを指定して使用します。たとえば、他の方式リストが定義されていない場合に、ARAP ユーザー認証方式としてローカルユーザー名データベースを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication arap default local
```

group RADIUS による ARAP 認証

ARAP 認証方式として RADIUS を指定するには、**group radius method** を指定して **aaa authentication arap {default | authentication-list}** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にログイン時のユーザー認証方式として RADIUS を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication arap default group radius
```

ARAP 認証方式として RADIUS を使用する前に、RADIUS セキュリティ サーバーとの通信をイネーブルにしておく必要があります。RADIUS サーバーとの通信を確立する方法の詳細については、「RADIUS の設定」の章を参照してください。

group TACACS による ARAP 認証

ARAP 認証方式として TACACS+ を指定するには、**group tacacs+ method** を指定して **aaa authentication arap {default | authentication-list}** コマンドを使用します。たとえば、他の方式リストが定義されていない場合にログイン時のユーザー認証方式として TACACS+ を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication arap default group tacacs+
```

ARAP 認証方式として TACACS+ を使用するには、TACACS+ セキュリティ サーバーとの通信をイネーブルにしておく必要があります。TACACS+ サーバーとの通信を確立する方法の詳細については、「TACACS+ の設定」の章を参照してください。

グループ名による ARAP 認証

ARAP 認証方式として使用する RADIUS または TACACS+ サーバーのサブセットを指定するには、**group group-name** 方式を指定して **aaa authentication arap {default | authentication-list}** コマンドを使用します。グループ名とそのグループのメンバを指定して定義するには、**aaa group server** コマンドを使用します。たとえば、**aaa group server** コマンドを使用して、**group araprad** のメンバを最初に定義します。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius araprad
Device(config-sg-radius)# server 172.16.2.3
Device(config-sg-radius)# server 172.16.2.17
Device(config-sg-radius)# server 172.16.2.32
Device(config-sg-radius)# end
```

このコマンドにより、172.16.2.3、172.16.2.17、172.16.2.32 の RADIUS サーバーがグループ **araprad** のメンバとして指定されます。

他の方式リストが定義されていない場合にログイン時のユーザー認証方式として **group araprad** を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication arap default group araprad
```

ARAP 認証方式としてグループ名を使用するには、RADIUS または TACACS+ セキュリティ サーバーとの通信をイネーブルにしておく必要があります。RADIUS サーバーとの通信を確立する方法の詳細については、「RADIUS の設定」の章を参照してください。TACACS+ サーバーとの通信を確立する方法の詳細については、「TACACS+ の設定」の章を参照してください。

AAAによるNASI認証

イネーブルパスワードによるNASI認証

認証方式としてイネーブルパスワードを指定するには、キーワード **enable** を指定して **aaa authentication nasi** コマンドを使用します。たとえば、他の方式リストが定義されていない場合に、NASIユーザー認証方式としてイネーブルパスワードを指定するには、次のコマンドを使用します。

```
Device(config)# aaa authentication nasi default enable
```

認証方式としてイネーブルパスワードを使用するには、イネーブルパスワードを定義しておく必要があります。

group RADIUSによるNASI認証

NASI認証方式としてRADIUSを指定するには、**group radius** 方式を指定して **aaa authentication nasi** コマンドを使用します。たとえば、他の方式リストが定義されていない場合に、NASIユーザー認証方式としてRADIUSを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication nasi default group radius
```

NASI認証方式としてRADIUSを使用するには、RADIUSセキュリティサーバーとの通信をイネーブルにしておく必要があります。

group TACACSによるNASI認証

NASI認証方式としてTACACS+を指定するには、**group tacacs+** キーワードを指定して **aaa authentication nasi** コマンドを使用します。たとえば、他の方式リストが定義されていない場合に、NASIユーザー認証方式としてTACACS+を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication nasi default group tacacs+
```

認証方式としてTACACS+を使用するには、TACACS+セキュリティサーバーとの通信をイネーブルにしておく必要があります。

ラインパスワードによるNASI認証

認証方式としてラインパスワードを指定するには、**line** キーワードを指定して **aaa authentication nasi** コマンドを使用します。たとえば、他の方式リストが定義されていない場合に、NASIユーザー認証方式としてラインパスワードを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication nasi default line
```

NASI認証方式としてラインパスワードを使用するには、ラインパスワードを定義しておく必要があります。

ローカルパスワードによる NASI 認証

シスコデバイスが認証情報にローカルユーザー名データベースを使用するように指定するには **aaa authentication nasi** コマンドに **local** キーワードを指定して使用します。たとえば、他の方式リストが定義されていない場合に、NASI ユーザー認証方式としてローカルユーザー名データベースを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication nasi default local
```

グループ名による NASI 認証

NASI 認証方式として使用する RADIUS または TACACS+ サーバーのサブセットを指定するには、**group group-name** 方式を指定して **aaa authentication nasi** コマンドを使用します。グループ名とそのグループのメンバを指定して定義するには、**aaa group server** コマンドを使用します。たとえば、**aaa group server** コマンドを使用して、**group nasirad** のメンバを最初に定義します。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius nasirad
Device(config-sg-radius)# server 172.16.2.3
Device(config-sg-radius)# server 172.16.2.17
Device(config-sg-radius)# server 172.16.2.32
Device(config-sg-radius)# end
```

このコマンドにより、172.16.2.3、172.16.2.17、172.16.2.32 の RADIUS サーバーがグループ *nasirad* のメンバとして指定されます。

他の方式リストが定義されていない場合にログイン時のユーザー認証方式として **group nasirad** を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication nasi default group nasirad
```

NASI 認証方式としてグループ名を使用するには、RADIUS または TACACS+ セキュリティサーバーとの通信をイネーブルにしておく必要があります。

ログイン入力にかかる時間の指定

timeout login response コマンドを使用すると、ログイン入力（ユーザー名やパスワードなど）がタイムアウトするまでの待機時間を指定できます。デフォルトのログイン値は 30 秒です。**timeout login response** コマンドを使用して、1～300 秒のタイムアウト値を指定できます。30 秒というデフォルトのログインタイムアウト値を変更するには、ライン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。

```
Device(config-line)# timeout login response 30
```


特権レベルでのパスワード保護

ユーザーが特権 EXEC コマンドレベルにアクセスできるかどうかを判断するときに使用する一連の認証方式を作成するには、**aaa authentication enable default** コマンドを使用します。最大 4 つの認証方式を指定できます。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。すべての方式でエラーが返されても引き続き認証を行うように指定するには、コマンドラインの最後の方式として **none** を指定します。

グローバル 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。

```
Device(config)# authentication enable default radius
```

または

```
Device(config)# authentication enable default tacacs
```

パスワード プロンプトに表示するテキストの変更

Cisco IOS XE ソフトウェアからユーザーに対してパスワードの入力を求めるときに表示されるデフォルトテキストを変更するには、**aaa authentication password-prompt** コマンドを使用します。このコマンドによって、イネーブルパスワードと、リモートセキュリティ サーバーから提供されていないログインパスワードのパスワード プロンプトが変更されます。このコマンドの **no** 形式を使用すると、パスワード プロンプトが次のデフォルト値に戻ります。

```
Password:
```

aaa authentication password-prompt コマンドでは、リモートの TACACS+ サーバーまたは RADIUS サーバーから提供されるダイアログは変更されません。

aaa authentication password-prompt コマンドは、RADIUS をログイン方式として使用するときには機能します。RADIUS サーバーに到達不能の場合でも、コマンドで定義されたパスワード プロンプトが表示されます。**aaa authentication password-prompt** コマンドは、TACACS+ では機能しません。TACACS+ は、NAS に対して、ユーザーに表示するパスワード プロンプトを提供します。TACACS+ サーバーが到達可能な場合、NAS はそのサーバーからパスワード プロンプトを受け取り、**aaa authentication password-prompt** コマンドで定義したプロンプトではなく、受け取ったプロンプトを使用します。TACACS+ サーバーが到達不能の場合、**aaa authentication password-prompt** コマンドで定義したパスワード プロンプトが使用される可能性があります。

グローバル 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。

```
Device(config)# aaa authentication password-prompt "Enter your password now:"
```

PPP セッションの二重認証

PPP セッションを認証できるのは、PAP または CHAP の単一の認証方法を使用した場合だけです。二重認証方式の場合、ネットワーク アクセス権を得るには、リモートユーザーが (CHAP または PAP 認証後に) 認証の第 2 段階に合格する必要があります。

この第2段階（「二重」）の認証には、ユーザーがパスワードを知っている必要がありますが、ユーザーのリモートホストにパスワードは保存されません。そのため、第2段階の認証は、ホストではなくユーザーに固有です。その結果、リモートホストから情報が盗まれた場合でも有効な、追加のセキュリティレベルが実現します。さらに、ユーザー別にネットワーク特権をカスタマイズできるため、柔軟性も高くなります。

第2段階の認証には、CHAPではサポートされないトークンカードなど、ワンタイムパスワードを使用できます。ワンタイムパスワードを使用している場合、ユーザーパスワードが盗まれても盗用者の役に立ちません。

二重認証の機能

二重認証を使用する場合、2つの認証/認可段階があります。この2つの段階は、リモートユーザーがダイヤルインした後、およびPPPセッションが開始された後に発生します。

第1段階では、ユーザーがリモートホスト名を使用してログインしてCHAP（またはPAP）がリモートホストを認証し、次にPPPがAAAとネゴシエートしてリモートホストを認可します。このプロセスで、リモートホストに関連付けられたネットワークアクセス特権は、そのユーザーに関連付けられます。



(注) ローカルホストに対してTelnet接続だけを許可するように、この第1段階ではネットワーク管理者が認可を制限することを推奨します。

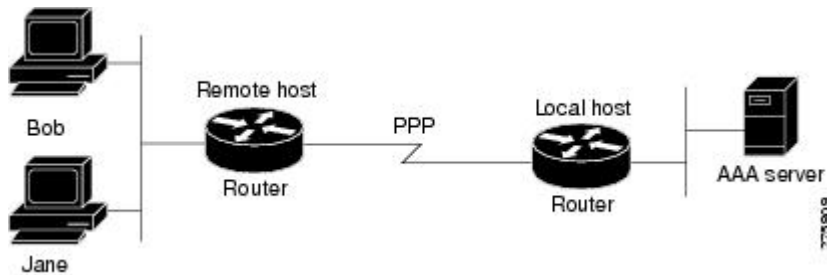
第2段階では、リモートユーザーが、認証を受けるネットワークアクセスサーバーに対してTelnetを送信する必要があります。リモートユーザーがログインする場合、AAAログイン認証を使用してユーザーを認証する必要があります。次に、AAAを使用して再度許可を受けるために、**access-profile** コマンドを入力する必要があります。この認可が完了すると、ユーザーは二重に認証され、ユーザー別のネットワーク特権に従ってネットワークにアクセスできるようになります。

システム管理者は、セキュリティサーバーで適切なパラメータを設定することで、各認証段階の後にリモートユーザーが保持するネットワーク特権を決定します。二重認証を使用するには、**access-profile** コマンドを発行してアクティブ化する必要があります。



注意 複数のホストがネットワーク アクセス サーバーに対して PPP 接続を共有する場合、二重認証によって望ましくない状況が発生することがあります（次の図を参照）。まず、ユーザー Bob が PPP セッションを開始し、ネットワーク アクセス サーバーで二重認証をアクティブにした場合（次の図を参照）、Bob の PPP セッションが期限切れになるまで、他のすべてのユーザーは Bob と同じネットワーク 特権を持つこととなります。この問題が発生するのは、PPP セッション時に Bob の認可プロファイルがネットワーク アクセス サーバーのインターフェイスに適用され、他のユーザーからの PPP トラフィックに Bob が確立した PPP セッションが使用されるためです。第 2 に、Bob が PPP セッションを開始して二重認証をアクティブにし、（Bob の PPP セッションが期限切れになる前に）別のユーザー Jane が **access-profile** コマンドを実行する場合（または、Jane がネットワーク アクセス サーバーに Telnet を送信し、**autocommand access-profile** が実行された場合）、再度許可が発生し、Jane の許可プロファイルがインターフェイスに適用され、Bob のプロファイルは置換されます。その結果、Bob の PPP トラフィックの不通や中止が発生することや、Bob が本来は持っていないレベルの特権が Bob に付与されることがあります。

図 2: 危険性を伴うトポロジ: 複数のホストがネットワーク アクセス サーバーに対する PPP 接続を共有



二重認証後のユーザー プロファイルへのアクセス

二重認証で、リモートユーザーがローカルホスト名を使用してローカルホストに対する PPP リンクを確立すると、リモートホストは CHAP（または PAP）認証されます。CHAP（または PAP）認証後、PPP は AAA とネゴシエートして、リモートホストに関連付けられたネットワーク アクセス 特権をユーザーに割り当てます（この段階の特権では、ユーザーがローカルホストに接続するには Telnet 接続を必須にするという制限を付けることを推奨します）。

ユーザーが二重認証の第 2 段階を開始する必要があるため、ローカルホストに対して Telnet 接続を確立する場合、ユーザーは個人の名とパスワード（CHAP または PAP のユーザー名とパスワードとは異なります）を入力します。この処理の結果、個人の名とパスワードに従って AAA 認証が発生します。ただし、ローカルホストに関連付けられた初期の権限が有効です。ローカルホストに関連付けられた権限は、**access-profile** コマンドを使用して、ユーザープロファイルのユーザー用に定義されている権限で置き換えられるか、結合されます。

二重認証後にユーザープロファイルにアクセスするには、EXEC 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。

```
Device> access-profile merge ignore-sanity-checks
```

autocommandとして実行するように **access-profile** コマンドを設定した場合、リモートユーザーのログイン後に自動的に実行されます。

CHAP 認証または PAP 認証

ISP のダイヤル ソリューションに使用されている最も一般的なトランスポートプロトコルの 1 つは、ポイントツーポイント プロトコル (PPP) です。従来、リモートユーザーはアクセスサーバーにダイヤルインして、PPP セッションを開始していました。PPP のネゴシエート後は、リモートユーザーは ISP ネットワークに接続され、そしてインターネットに接続されます。

ISP はアクセスサーバーへの接続を顧客に限定したいため、リモートユーザーはアクセスサーバーに対して認証を受けてから、PPP セッションを開始する必要があります。通常、リモートユーザーは、アクセスサーバーからのプロンプトに応じてユーザー名とパスワードを入力して、認証を受けます。これは実行可能なソリューションですが、管理が困難で、リモートユーザーにとっても面倒です。

よりよいソリューションは、PPP に組み込まれた認証プロトコルを使用することです。この場合、リモートユーザーはアクセスサーバーにダイヤルインし、アクセスサーバーと PPP の最小サブセットを開始します。この操作で、ISP のネットワークに対するアクセス権はリモートユーザーに付与されません。単に、アクセスサーバーがリモートデバイスと通話できるだけです。

現在、PPP は 2 つの認証プロトコルをサポートします。パスワード認証プロトコル (PAP) およびチャレンジハンドシェイク認証プロトコル (CHAP) の 2 つです。いずれも RFC 1334 で規定され、同期インターフェイスと非同期インターフェイスでサポートされます。PAP または CHAP を介する認証は、サーバーからのプロンプトを受けてユーザー名とパスワードを入力する方法と同等です。CHAP の場合、接続の間にリモートユーザーのパスワードは送信されないため、より安全性が高いと考えられます。

(PAP 認証または CHAP 認証の有無に関係なく) PPP はダイヤルアウト ソリューションでもサポートされます。アクセスサーバーがダイヤルアウト機能を使用するのは、アクセスサーバーからリモートデバイスに対してコールを開始し、PPP などのトランスポートプロトコルを起動しようとするときです。



(注) CHAP または PAP を使用するには、PPP カプセル化を実行する必要があります。

インターフェイスで CHAP をイネーブルにし、リモートデバイスがそのインターフェイスに接続しようすると、アクセスサーバーからリモートデバイスに CHAP パケットが送信されます。CHAP パケットは、リモートデバイスに応答するように要求または「チャレンジ」します。チャレンジパケットは、ローカルデバイスの ID、ランダム番号、およびホスト名から構成されます。

リモートデバイスは、チャレンジパケットを受信すると、ID、リモートデバイスのパスワード、およびランダム番号を連結し、リモートデバイスのパスワードを使用してすべてを暗号化

します。リモートデバイスは、その結果を、暗号化プロセスで使用されたパスワードに関連付けられた名前とともにアクセス サーバーに返信します。

アクセス サーバーがその応答を受信すると、受信した名前を使用して、ユーザー データベースに保存されているパスワードを取得します。取得したパスワードは、暗号化プロセスで使用されたリモート デバイスと同じパスワードです。アクセス サーバーは、新しく取得したパスワードを使用して、連結された情報を暗号化します。その結果が応答パケットで送信された結果と一致する場合、認証は成功です。

CHAP 認証を使用する利点は、リモート デバイスのパスワードがクリア テキストで送信されないことです。結果として、他のデバイスによるパスワード盗用や、ISP のネットワークに対する不正アクセスの取得を回避できます。

CHAP トランザクションが発生するのは、リンクが確立したときだけです。アクセス サーバーは、以降のコール中にパスワードを要求しません（ただし、ローカルデバイスは、コール中に他のデバイスからこのような要求があった場合、応答する可能性があります）。

PAP を有効にすると、アクセスサーバーに接続しようとするリモートデバイスは、認証要求を送信する必要があります。認証要求に指定されているユーザー名とパスワードが受け入れられた場合、Cisco IOS XE ソフトウェアから認証の確認応答が送信されます。

CHAP または PAP をイネーブルにすると、アクセス サーバーは、ダイヤルインするリモートデバイスからの認証を必須にするようになります。イネーブルにしたプロトコルをリモートデバイスがサポートしていない場合、コールはドロップされます。

CHAP または PAP を使用するには、次のタスクを実行する必要があります。

- PPP カプセル化をイネーブルにします。
- インターフェイスで CHAP または PAP をイネーブルにします。
- CHAP の場合、認証が必須の各リモート システムについて、ホスト名の認証および秘密（パスワード）を設定します。

PPP カプセル化の有効化

PPP カプセル化をイネーブルにするには、インターフェイス コンフィギュレーション モードで次のコマンドを使用します。

```
Device(config-if)# encapsulation ppp
```

このコマンドはインターフェイスで PPP を有効にします。

PAP または CHAP のイネーブル化

PPP カプセル化として設定されているインターフェイスで、CHAP 認証または PAP 認証をイネーブルにするには、インターフェイス コンフィギュレーション モードで次のコマンドを使用します。

```
Device(config-if)# ppp authentication chap pap
```

サポートされる認証プロトコルと、使用順序を定義します。このコマンドの *protocol1* と *protocol2* は、CHAP、MS-CHAP、および PAP のプロトコルを示します。PPP 認証は、まず *protocol1* に

指定された最初の認証方式を使用して試行されます。認証に *protocol1* を使用できない場合は、次に設定されているプロトコルを使用して認証のネゴシエーションを行います。

インターフェイスで **ppp authentication chap** を設定する場合、そのインターフェイスで PPP 接続を開始するすべての受信コールは、CHAP を使用して認証される必要があります。同様に、**ppp authentication pap** を設定する場合、PPP 接続を開始するすべての受信コールは、PAP を使用して認証される必要があります。**ppp authentication chap pap** を設定する場合、アクセスサーバーは、CHAP を使用して PPP セッションを開始するすべての受信コールを認証しようとします。リモートデバイスが CHAP をサポートしない場合、アクセスサーバーは PAP を使用してコールを認証しようとします。リモートデバイスが CHAP も PAP もサポートしない場合、認証は失敗し、コールはドロップされます。**ppp authentication pap chap** を設定する場合、アクセスサーバーは、PAP を使用して PPP セッションを開始するすべての受信コールを認証しようとします。リモートデバイスが PAP をサポートしない場合、アクセスサーバーは CHAP を使用してコールを認証しようとします。リモートデバイスがいずれのプロトコルもサポートしない場合、認証は失敗し、コールはドロップされます。**callin** キーワードを指定して **ppp authentication** コマンドを設定すると、アクセスサーバーは、リモートデバイスがコールを開始した場合にだけ、リモートデバイスの認証を行います。

認証方式リストと **one-time** キーワードを使用できるのは、AAA を有効にした場合だけです。TACACS または拡張 TACACS を有効にしている場合は、使用できません。**ppp authentication** コマンドを使用して認証方式リストの名前を指定すると、PPP は、指定した方式リストに定義されている方式を使用して、接続を認証しようとします。AAA をイネーブルにし、名前で定義されている方式リストがない場合、PPP は、デフォルトに定義されている方式を使用して接続を認証しようとします。**one-time** キーワードを指定して **ppp authentication** コマンドを使用すると、認証中にワンタイムパスワードをサポートできます。

if-needed キーワードを使用できるのは、TACACS または拡張 TACACS を使用している場合だけです。**if-needed** キーワードを指定して **ppp authentication** コマンドを使用することは、現在のコール期間中にリモートデバイスがまだ認証されていない場合にだけ、PPP が PAP または CHAP を介してリモートデバイスを認証することを示します。リモートデバイスが、標準のログイン手順で認証を受け、EXEC プロンプトから PPP を開始した場合、**ppp authentication chap if-needed** がインターフェイスで設定されていれば、PPP は CHAP を介して認証しません。



注意 **aaa authentication ppp** コマンドを使用して設定されていない *list-name* を使用する場合、その回線での PPP は無効になります。

着信認証と発信認証

PPP は双方向の認証をサポートしています。通常、リモートデバイスがアクセスサーバーにダイヤルインするときは、それが許可されているアクセスであることをリモートデバイスが証明するように、アクセスサーバーから要求されます。これは着信認証と呼ばれます。同時に、リモートデバイスは、身元を証明するようにアクセスサーバーに要求することもできます。これは発信認証と呼ばれます。また、アクセスサーバーは、リモートデバイスに対してコールを開始するときにも、発信認証を実行します。

発信 PAP 認証のイネーブル化

発信 PAP 認証をイネーブルにするには、インターフェイス コンフィギュレーション モードで次のコマンドを使用します。

```
Device(config-if)# ppp pap sent-username username1 password password1
```

アクセスサーバーからリモートデバイスに対してコールを開始する場合は常に、またはアウトバウンド認証のためにリモートデバイスの要求に応答する必要がある場合は、**ppp pap sent-username** コマンドで指定されたユーザー名とパスワードを使用して自身を認証します。

PAP 認証要求の拒否

ピアからの PAP 認証要求を拒否するには（つまり、すべてのコールで PAP 認証をディセーブルにするには）、インターフェイス コンフィギュレーション モードで次のコマンドを使用します。

```
Device(config-if)# ppp pap refuse
```

refuse キーワードが使用されない場合、デバイスはピアから受信した PAP 認証チャレンジを拒否しません。

共通 CHAP パスワードの作成

リモート CHAP 認証の場合、不明なピアからのチャレンジに応じて使用する共通の CHAP シークレットパスワードを作成するようにデバイスを設定できます。たとえば、新しい（つまり、不明な）デバイスが追加されたデバイス（別のベンダーの、または古いバージョンの Cisco IOS XE ソフトウェアを実行しているデバイス）のロータリーを呼び出します。**ppp chap password** コマンドを使用すると、任意のダイヤラインターフェイスまたは非同期グループインターフェイスで、複数のユーザー名およびパスワード コンフィギュレーション コマンドをこのコマンドの単一のコピーで置換できます。

デバイスのコレクションに発信するデバイスが、共通の CHAP シークレットパスワードを設定できるようにするには、インターフェイス コンフィギュレーション モードで次のコマンドを使用します。

```
Device(config-if)# ppp chap password secret
```

CHAP 認証要求の拒否

ピアからの CHAP 認証要求を拒否するには（つまり、すべてのコールで CHAP 認証をディセーブルにするには）、インターフェイス コンフィギュレーション モードで次のコマンドを使用します。

```
Device(config-if)# ppp chap refuse calling
```

calling キーワードが使用されると、デバイスは、ピアから受信した CHAP 認証チャレンジへの応答を拒否します。ただし、デバイスが送信する CHAP チャレンジに対しては、ピアが応答することを必須とします。

（**ppp pap sent-username** コマンドを使用して）アウトバウンド PAP が有効になっている場合、拒否パケットの認証方式として、PAP が使用されます。

ピアが認証されるまで CHAP 認証を遅延する

CHAP 認証を要求するピアがデバイスから認証を受けるまで、デバイスがこのピアを認証しないように指定するには、インターフェイス コンフィギュレーション モードで次のコマンドを使用します。

```
Device(config-if)# ppp chap wait secret
```

このコマンド（デフォルト）により、CHAP 認証を要求するピアがデバイスから認証を受けるまで、デバイスがこのピアを認証しないように指定します。**no ppp chap wait** コマンドにより、デバイスが認証チャレンジに対して即時に応答するように指定されます。

MS-CHAP の使用

マイクロソフト チャレンジハンドシェイク認証プロトコル（MS-CHAP）は、Microsoft バージョンの CHAP であり、RFC 1994 の拡張です。標準バージョンの CHAP と同様に、MS-CHAP は PPP 認証に使用されます。この場合、Microsoft Windows NT または Microsoft Windows 95 を使用する PC と、ネットワーク アクセス サーバーとして動作する Cisco デバイスまたはアクセス サーバーとの間に認証が発生します。

MS-CHAP と標準の CHAP の違いは次のとおりです。

- MS-CHAP をイネーブルにするには、LCP オプション 3 の Authentication Protocol で、CHAP Algorithm 0x80 をネゴシエートします。
- MS-CHAP 応答パケットは、Microsoft Windows NT 3.5 および 3.51、Microsoft Windows 95、および Microsoft LAN Manager 2.x と互換性を持つように設計されたフォーマットです。このフォーマットを使用する場合、オーセンティケータは、クリアパスワードまたは可逆的に暗号化されたパスワードを保存する必要はありません。
- MS-CHAP には、オーセンティケータが制御する認証リトライ メカニズムがあります。
- MS-CHAP には、オーセンティケータが制御するチャレンジパスワードメカニズムがあります。
- MS-CHAP には、Failure パケット メッセージ フィールドで返される「reason-for failure」コードセットが定義されています。

実装したセキュリティ プロトコルに応じて、AAA セキュリティ サービスの有無にかかわらず、MS-CHAP による PPP 認証を使用できます。AAA をイネーブルにしている場合、MS-CHAP を使用する PPP 認証は、TACACS+ および RADIUS の両方と併用できます。次の表に、RADIUS が MS-CHAP をサポートできるベンダー固有 RADIUS 属性（IETF Attribute 26）を示します。

表 3: MS-CHAP 用のベンダー固有 RADIUS 属性

ベンダーID 番号	ベンダータイ プ 番号	ベンダー固有属性	説明
311	11	MSCHAP-Challenge	ネットワーク アクセスサーバが MS-CHAP ユーザに送信するチャレンジが含まれます。これは、Access-Request パケットと Access-Challenge パケットの両方で使用できます。
211	11	MSCHAP-Response	PPP MS-CHAP ユーザがチャレンジに対する応答で提供するレスポンス値が含まれます。Access-Request パケットでしか使用されません。この属性は、PPP CHAP ID と同じです

ドメインストリッピング

AAA ブロードキャスト アカウンティング機能を有効にすると、アカウンティング情報を複数の AAA サーバーに同時に送信できます。つまり、アカウンティング情報を 1 つまたは複数の AAA サーバーに同時にブロードキャストすることが可能です。この機能を使用すると、プライベートおよびパブリック AAA サーバーにアカウント情報を送信できます。この機能では、音声アプリケーションによる課金情報も提供されます。

ドメインストリッピング機能を使用すると、ドメインストリッピングをサーバーグループレベルで設定できます。

サーバー単位のグループ コンフィギュレーションはグローバル コンフィギュレーションを上書きします。ドメインストリッピングが、グローバルではイネーブルではないがサーバーグループでイネーブルになっている場合、そのサーバーグループに対してのみイネーブルになります。また、Virtual Routing and Forwarding (VRF) 固有のドメインストリッピングがグローバルで設定されていて、別の VRF のドメインストリッピングがサーバーグループで設定されている場合、ドメインストリッピングは両方の VRF でイネーブルになります。VRF の設定は、サーバーグループ コンフィギュレーションモードから取得されます。サーバーグループ コンフィギュレーションがグローバル コンフィギュレーションモードでディセーブルになっているが、サーバーグループ コンフィギュレーションモードで使用可能である場合、サーバーグループ コンフィギュレーションモードでのすべての設定が適用可能です。

ドメインストリッピングおよびブロードキャスト アカウンティングを設定した後で、設定ごとに別個のアカウンティングレコードを作成できます。

domain-stripping コマンドと **directed-request** コマンドの両方が有効になっている場合、ドメインストリッピングが優先され、ダイレクトリクエスト機能は動作しません。

認証の設定方法

AAA を使用したログイン認証の設定

AAA セキュリティ サービスにより、さまざまなログイン認証方式を容易に実行できるようになります。 **aaa authentication login** コマンドを使用すると、サポートされているログイン認証方式のいずれを使用するかに関係なく、AAA 認証が有効になります。 **aaa authentication login** コマンドを使用すると、ログイン時に試行する認証方式リストを 1 つまたは複数作成できます。これらのリストは、**login authentication** ライン コンフィギュレーション コマンドによって適用されます。

AAA を使用してログイン認証を設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login** {default | list-name} method1[method2...]
5. **line** [aux | console | tty | vty] line-number [ending-line-number]
6. **login authentication** {default | list-name}
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA をイネーブルにします。
ステップ 4	aaa authentication login {default list-name} method1[method2...]	ローカルな認証リストを作成します。

	コマンドまたはアクション	目的
	例 : Device(config)# aaa authentication login default local	
ステップ 5	line [aux console tty vty] line-number [ending-line-number] 例 : Device(config)# line vty 1	認証リストを適用する回線について、ライン コンフィギュレーションモードを開始します。
ステップ 6	login authentication {default list-name} 例 : Device(config-line)# login authentication default	1 つの回線または複数回線に認証リストを適用します。
ステップ 7	end 例 : Device(config-line)# end	回線コンフィギュレーションモードを終了します。続いて、特権 EXEC モードに戻ります。

次のタスク

list-name は、作成するリストを指定するときを使用される名前、文字列を使用できます。*method* 引数は、認証アルゴリズムが試行する実際の方式を指します。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。すべての方式でエラーが返されても引き続き認証を行うように指定するには、コマンドラインの最後の方式として **none** を指定します。

たとえば、（この例では）TACACS+ サーバーでエラーが返されても引き続き認証を行うように指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default group tacacs+ none
```



(注) **none** キーワードを指定すると、すべてのユーザーがログイン認証に成功するため、認証のバックアップ方式としてだけ使用してください。

login authentication コマンドにリストが指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルト認証方式リストは、自動的にすべてのインターフェイスに適用されます。

たとえば、ログイン時のユーザー認証のデフォルト方式として RADIUS を指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication login default group radius
```

AAA を使用した PPP 認証の設定

AAA セキュリティ サービスにより、PPP を実行するシリアルインターフェイスに使用できるさまざまな認証方式の実行が容易になります。 **aaa authentication ppp** コマンドを使用すると、サポートされている PPP 認証方式のいずれを使用するかに関係なく、AAA 認証が有効になります。

PPP を使用してシリアル回線に AAA 認証方式を設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication ppp**{default | list-name} method1[method2...]
5. **interface** interface-type interface-number
6. **ppp authentication** {protocol1 [protocol2...]} [if-needed] {default | list-name} [callin] [one-time][optional]
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA をイネーブルにします。
ステップ 4	aaa authentication ppp {default list-name} method1[method2...]	ローカルな認証リストを作成します。

	コマンドまたはアクション	目的
	例 : Device(config)# aaa authentication ppp-auth default local	
ステップ 5	interface interface-type interface-number 例 : Device(config)# interface gigabitethernet 0/1/0	認証リストを適用するインターフェイスについて、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 6	ppp authentication {protocol1 [protocol2...]} [if-needed] {default list-name} [callin] [one-time][optional] 例 : Device(config)# ppp authentication ms-chap ppp-auth	1 つの回線または複数回線に認証リストを適用します。このコマンドの <i>protocol1</i> と <i>protocol2</i> は、CHAP、MS-CHAP、および PAP のプロトコルを示します。PPP 認証は、まず <i>protocol1</i> に指定された最初の認証方式を使用して試行されます。認証に <i>protocol1</i> を使用できない場合は、次に設定されているプロトコルを使用して認証のネゴシエーションを行います。
ステップ 7	end 例 : Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

次のタスク

aaa authentication ppp コマンドを使用して、PPP を介して認証を試行するときに使用する認証方式のリストを 1 つまたは複数作成します。これらのリストは、**ppp authentication** ライン コンフィギュレーション コマンドによって適用されます。

名前付きリストが **ppp authentication** コマンドに指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状況で使用される方式を指定します。

たとえば、ユーザー認証のデフォルト方式としてローカルユーザー名データベースを指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication ppp default local
```

list-name は、作成するリストを指定するときに使用される名前です。任意の文字列を使用できます。**method** 引数は、認証アルゴリズムが試行する実際の方式を指します。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。すべての方式でエラーが返されても引き続き認証を行うように指定するには、コマンドラインの最後の方式として **none** を指定します。

たとえば、（この例では）TACACS+ サーバーでエラーが返されても引き続き認証を行うように指定するには、次のコマンドを入力します。

```
Device(config)# aaa authentication ppp default group tacacs+ none
```



(注) **none** を指定するとすべてのユーザーが認証に成功してログインできるようになるため、認証のバックアップ方式として使用する必要があります。

AAA を使用した ARAP 認証の設定

aaa authentication arap コマンドを使用して、AppleTalk Remote Access Protocol (ARAP) ユーザーがデバイスにログインを試行するとき使用する認証方式のリストを1つまたは複数作成できます。これらのリストは、**arap authentication** ラインコンフィギュレーションコマンドで使用されます。

グローバル 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。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication arap**
5. **line number**
6. Device(config-line)# **autoselect arap**
7. **autoselect during-login**
8. **arap authentication list-name**
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例：	AAA をイネーブルにします。

	コマンドまたはアクション	目的
	Device(config)# aaa new-model	
ステップ 4	aaa authentication arap 例： Device(config)# aaa authentication arap 例： ARAP ユーザーに対する認証をイネーブルにします。	
ステップ 5	line number 例： Device(config)# line 1	(任意) ライン コンフィギュレーション モードに変更します。
ステップ 6	Device(config-line)# autoselect arap 例： Device(config-line)# auto-select arap	(任意) ARAP の自動選択をイネーブルにします。
ステップ 7	autoselect during-login 例： Device(config-line)# autoselect during-login	(任意) ユーザー ログイン時に ARAP セッションを自動的に開始します。
ステップ 8	arap authentication list-name 例： Device(config-line)# arap authentication arap-authen	(任意 : default が aaa authentication arap コマンドに使用されている場合は不要) 回線上の ARAP に対する TACACS+ 認証を有効にします。
ステップ 9	end 例： Device(config-line)# end	ライン コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

次のタスク

list-name は、作成するリストを指定するときに使用される名前で、任意の文字列を使用できます。*method* 引数は、認証アルゴリズムが試行する方式の実際のリストを指します。試行は入力されている順序で行われます。

名前付きリストが **arap authentication** コマンドに指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状況で使用される方式を指定します。

追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。すべての方式でエラーが返されても引き続き認証を行うように指定するには、コマンドラインの最後の方式として **none** を指定します。



(注) **none** を指定するとすべてのユーザーのログインが認証されるようになるため、認証のバックアップ方式として使用する必要があります。

たとえば、ARAP とともに使用するデフォルトの AAA 認証方式リストを作成するには、次のコマンドを使用します。

```
Device(config)# aaa authentication arap default if-needed none
```

ARAP に同じ認証方式リストを作成し、リストに *MIS-access* と名前を付けるには、次のコマンドを入力します。

```
Device(config)# aaa authentication arap MIS-access if-needed none
```

AAA を使用した NASI 認証の設定

aaa authentication nasi コマンドを使用して、NetWare Asynchronous Services Interface (NASI) ユーザーがデバイスにログインを試行するときに使用する認証方式のリストを1つまたは複数作成できます。これらのリストは、**nasi authentication line** コンフィギュレーション コマンドで使用されます。

AAA を使用して NASI 認証を設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication nasi**
5. **line number**
6. **nasi authentication list-name**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA をグローバルに有効にします。
ステップ 4	aaa authentication nasi 例： Device(config)# aaa authentication nasi	NASI ユーザーに対する認証をイネーブルにします。
ステップ 5	line number 例： Device(config)# line 4	（任意： aaa authentication nasi コマンドで default が使用されている場合は不要。）ライン コンフィギュレーション モードを開始します。
ステップ 6	nasi authentication list-name 例： Device(config-line)# nasi authentication nasi-authen	（任意： aaa authentication nasi コマンドで default が使用されている場合は不要。）回線で NASI の認証を有効にします。
ステップ 7	end 例： Device(config-line)# end	ライン コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

次のタスク

list-name は、作成するリストを指定するときに使用される名前です。任意の文字列を使用できます。*method* 引数は、認証アルゴリズムが試行する方式の実際のリストを指します。試行は入力されている順序で行われます。

aaa authentication nasi コマンドに名前付きリストが指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状態で使用される方式を指定します。

追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。すべての方式でエラーが返されても引き続き認証を行うように指定するには、コマンドラインの最後の方式として **none** を指定します。



(注) **none** を指定するとすべてのユーザーのログインが認証されるようになるため、認証のバックアップ方式として使用する必要があります。

ユーザー名が空のアクセス要求が RADIUS サーバーに送信されないようにする

次の設定手順では、ユーザー名が空のアクセス要求が RADIUS サーバーに送信されないようにする方法について説明します。この機能により、RADIUS サーバーとの不要なやりとりを回避でき、RADIUS ログの量を少なくすることができます。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication suppress null-username**
5. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	aaa new-model 例 : Device(config)# configure terminal	AAA をグローバルに有効にします。
ステップ 4	aaa authentication suppress null-username 例 : Device(config)# aaa authentication suppress null-username	ユーザー名が空のアクセス要求が RADIUS サーバーに送信されないようにします。
ステップ 5	end 例 : Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

AAA 認証のメッセージバナーの設定

AAA は、設定可能でパーソナライズされたログインおよび failed-login バナーの使用をサポートします。ユーザーが AAA を使用して認証を受けるシステムにログインする場合、および何らかの理由で認証が失敗した場合に表示されるメッセージバナーを設定できます。

ログインバナーの設定

ユーザーがログインするときに表示されるメッセージを設定する（デフォルトのログインメッセージを置き換える）には、次のタスクを実行します。

始める前に

ログインバナーを作成するには、デリミタを設定する必要があります。設定することで、続くテキスト文字列をバナーとして表示する必要があることがシステムに通知されます。次に、テキスト文字列自体を設定する必要があります。デリミタは、バナーの末尾を示すために、テキストストリングの末尾で繰り返されます。デリミタには、拡張 ASCII 文字セットの任意の 1 文字を使用できます。ただし、デリミタとして定義した文字は、バナー用のテキスト文字列には使用できません。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication banner delimiter string delimiter**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA をイネーブルにします。
ステップ 4	aaa authentication banner delimiter string delimiter 例： Device(config)# aaa authentication banner *Unauthorized use is prohibited.*	パーソナライズされたログイン バナーを作成します。
ステップ 5	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Failed-Login バナーの設定

ユーザーログインが失敗したときに表示されるメッセージを設定する（デフォルトの `failed-login` メッセージを置き換える）には、次のタスクを実行します。

始める前に

`failed-login` バナーを作成するには、デリミタを設定する必要があります。設定することで、続くテキスト文字列をバナーとして表示する必要があることがシステムに通知されます。次に、テキスト文字列自体を設定する必要があります。デリミタは、`failed-login` バナーの末尾を示すために、テキスト スtring の末尾で繰り返されます。デリミタには、拡張 ASCII 文字セットの任意の 1 文字を使用できます。ただし、デリミタとして定義した文字は、バナーを構成するテキスト スtring には使用できません。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication fail-message *delimiter string delimiter***
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA をイネーブルにします。
ステップ 4	aaa authentication fail-message <i>delimiter string delimiter</i> 例： Device(config)# aaa authentication fail-message *Failed login. Try again.*	ユーザー ログインが失敗したときに表示されるメッセージを作成します。
ステップ 5	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

AAA パケット オブ ディスコネクトの設定

特定のセッション属性が指定された場合、パケットオブディスコネクト (POD) によってネットワーク アクセス サーバー (NAS) の接続が終了されます。UNIX ワークステーション上にある POD クライアントでは、AAA から取得したセッション情報を使用して、ネットワーク アクセス サーバーで実行されている POD サーバーに接続解除パケットを送信します。NAS では、

1つまたは複数の一致するキー属性を含む任意の着信ユーザーセッションを終了します。必要なフィールドがない場合、または完全一致が見つからない場合、要求は拒否されます。

PODを設定するには、グローバルコンフィギュレーションモードで次のタスクを実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa accounting network default start-stop radius**
4. **aaa accounting delay-start**
5. **aaa pod server server-key string**
6. **radius server name non-standard**
7. **address {ipv4 | ipv6} hostname**
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa accounting network default start-stop radius 例： Device(config)# aaa accounting network default start-stop radius	AAA アカウンティング レコードをイネーブルにします。
ステップ 4	aaa accounting delay-start 例： Device(config)# aaa accounting delay-start	(任意) POD パケットで使用できるように、Framed-IP-Address が割り当てられるまで、開始アカウンティング レコードの生成を遅延します。
ステップ 5	aaa pod server server-key string 例： Device(config)# aaa pod server server-key xyz123	POD の受信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 6	radius server <i>name</i> non-standard 例 : Device(config)# radius server radser	RADIUS サーバーを設定し、RADIUS サーバー コンフィギュレーション モードを開始します。
ステップ 7	address { ipv4 ipv6 } <i>hostname</i> 例 : Device(config-radius-server)# address ipv4 radius-host	RADIUS ホストを設定します。
ステップ 8	end 例 : Device(config-radius-server)# end	RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

二重認証の設定

二重認証を設定するには、次の手順を実行します。

1. **aaa-new model** グローバル コンフィギュレーション コマンドを使用して、AAA を有効にします。
2. **aaa authentication** コマンドを使用して、ログインおよび PPP 認証方式リストを使用するようにネットワークアクセスサーバーを設定します。次に、これらの方式リストを適切な回線やインターフェイスに適用します。
3. **aaa authorization** コマンドを使用して、ログイン時の AAA ネットワーク許可を設定します。
4. セキュリティプロトコルパラメータ（たとえば、RADIUS または TACACS+）を設定します。
5. セキュリティサーバーで、ユーザーがローカルホストに接続できるアクセスコントロールリストの AV ペアを使用するには、Telnet 接続を確立する必要があります。
6. （任意）autocommand として **access-profile** コマンドを設定します。autocommand を設定すると、リモートユーザーは、個人のユーザープロファイルに関連付けられた許可済み権限にアクセスするために、手動で **access-profile** コマンドを入力する必要はなくなります。



(注) **access-profile** コマンドが autocommand として設定されている場合でも、二重認証を完了するには、ユーザーがローカルホストに Telnet を送信し、ログインする必要があります。

ユーザー固有の許可ステートメントを作成する場合、次の規則に従います（これらの規則は、**access-profile** コマンドのデフォルトの動作に関連します）。

- セキュリティ サーバーでアクセス コントロール リストの AV ペアを設定する場合、有効な AV ペアを使用します。
- リモートユーザーがインターフェイスの既存の許可（第 2 段階の認証/許可の前に存在する許可）を使用し、異なるアクセスコントロールリスト（ACL）を持つようにするには、ユーザー固有の許可定義で ACL AV ペアだけを指定します。この方法は、デフォルトの認可プロファイルを設定してリモート ホストに適用し、ACL はユーザー別に適用する場合などに有効です。
- これらのユーザー固有の許可ステートメントを後でインターフェイスに適用すると、ユーザーの許可に使用する **access-profile** コマンドの実行形式によって、既存のインターフェイス設定に追加することや、既存のインターフェイス設定を置き換えることができます。許可ステートメントを設定する前に、**access-profile** コマンドの機能について理解する必要があります。
- ISDN または Multilink PPP を使用する予定がある場合、ローカル ホストで仮想テンプレートも設定する必要があります。

自動二重認証のイネーブル化

自動二重認証を実装することで、ユーザーにとって二重認証プロセスが容易になります。自動二重認証は、二重認証が持つセキュリティ上の利点をすべて備えています。リモートユーザーにとってよりシンプルでユーザーフレンドリなインターフェイスです。二重認証の場合、ユーザー認証の第 2 レベルは、ユーザーがネットワーク アクセス サーバーまたはルータに Telnet に送信し、ユーザー名とパスワードを入力したときに完了します。自動二重認証の場合、ユーザーがネットワーク アクセス サーバーに Telnet を送信する必要はありません。その代わりに、ユーザー名とパスワードまたは Personal Identification Number (PIN) の入力を求めるダイアログボックスが表示されます。自動二重認証機能を使用するには、対応するクライアントアプリケーションがリモート ユーザー ホストで実行されている必要があります。



(注) 自動二重認証は、既存の二重認証機能と同様に、Multilink PPP ISDN 接続専用です。自動二重認証は、X.25 や SLIP など他のプロトコルとは併用できません。

自動二重認証は、既存の二重認証機能の強化です。自動二重認証を設定するには、まず次の手順を実行して二重認証を設定する必要があります。

1. **aaa-new model** グローバル コンフィギュレーション コマンドを使用して、AAA を有効にします。
2. **aaa authentication** コマンドを使用して、ログインおよび PPP 認証方式リストを使用するようにネットワークアクセスサーバーを設定します。次に、これらの方式リストを適切な回線やインターフェイスに適用します。

3. **aaa authorization** コマンドを使用して、ログイン時の AAA ネットワーク許可を設定します。
4. セキュリティプロトコルパラメータ（たとえば、RADIUS または TACACS+）を設定します。
5. セキュリティサーバーで、ユーザーがローカルホストに接続できるアクセスコントロールリストの AV ペアを使用するには、Telnet 接続を確立する必要があります。
6. autocommand として **access-profile** コマンドを設定します。autocommand を設定すると、リモートユーザーは、個人のユーザープロファイルに関連付けられた許可済み権限にアクセスするために、手動で **access-profile** コマンドを入力する必要はなくなります。



(注) **access-profile** コマンドが autocommand として設定されている場合でも、二重認証を完了するには、ユーザーがローカルホストに Telnet を送信し、ログインする必要があります。

ユーザー固有の許可ステートメントを作成する場合、次の規則に従います（これらの規則は、**access-profile** コマンドのデフォルトの動作に関連します）。

- セキュリティサーバーでアクセスコントロールリストの AV ペアを設定する場合、有効な AV ペアを使用します。
- リモートユーザーがインターフェイスの既存の認可（第 2 段階の認証/認可の前に存在する認可）を使用し、異なるアクセスコントロールリスト（ACL）を持つようにするには、ユーザー固有の認可定義で ACL AV ペアだけを指定します。この方法は、デフォルトの認可プロファイルを設定してリモートホストに適用し、ACL はユーザー別に適用する場合などに有効です。
- これらのユーザー固有の許可ステートメントを後でインターフェイスに適用すると、ユーザーの許可に使用する **access-profile** コマンドの実行方法によって、既存のインターフェイス設定に追加することや、既存のインターフェイス設定を置き換えることができます。許可ステートメントを設定する前に、**access-profile** コマンドの機能について理解する必要があります。
- ISDN または Multilink PPP を使用する予定がある場合、ローカルホストで仮想テンプレートも設定する必要があります。

自動二重認証の設定

自動二重認証を設定するには、次のタスクを実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip trigger-authentication [timeout seconds] [port number]**
4. **interface type number**

5. **ip trigger-authentication**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip trigger-authentication [timeout seconds] [port number] 例： Device(config)# ip trigger-authentication timeout 120	二重認証の自動化をイネーブルにします。
ステップ 4	interface type number 例： Device(config)# interface gigabitethernet 1/0/17	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 5	ip trigger-authentication 例： Device(config-if)# ip trigger-authentication	自動二重認証をインターフェイスに適用します。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

自動二重認証のトラブルシューティング

自動二重認証の問題を解決するには、特権 EXEC モードで次のコマンドを使用します。

手順の概要

1. **enable**

2. **show ip trigger-authentication**
3. **clear ip trigger-authentication**
4. **debug ip trigger-authentication**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	show ip trigger-authentication 例： Device# show ip trigger-authentication	自動二重認証が試行され、成功または失敗したりモートホストのリストが表示されます。
ステップ 3	clear ip trigger-authentication 例： Device# clear ip trigger-authentication	自動二重認証が試行されたリモートホストのリストをクリアします（これは、 show ip trigger-authentication コマンドで表示されるテーブルをクリアします）。
ステップ 4	debug ip trigger-authentication 例： Device# debug ip trigger-authentication	自動二重認証に関する debug の出力が表示されます。

サーバー グループ レベルでのドメインストリッピングの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa group server radius server-name**
4. **domain-stripping [strip-suffix word] [right-to-left] [prefix-delimiter word] [delimiter word]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa group server radius server-name 例： Device(config)# aaa group server radius rad1	RADIUS サーバを追加し、サーバグループ RADIUS コンフィギュレーション モードを開始します。 • <i>server-name</i> 引数には、RADIUS サーバー グループ名を指定します。
ステップ 4	domain-stripping [strip-suffix word] [right-to-left] [prefix-delimiter word] [delimiter word] 例： Device(config-sg-radius)# domain-stripping delimiter username@example.com	サーバー グループ レベルでドメイン ストリッピングを設定します。
ステップ 5	end 例： Device(config-sg-radius)# end	サーバー グループ RADIUS コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

非 AAA 認証方式の設定

ラインパスワード保護の設定

このタスクは、パスワードを入力し、パスワードチェック処理を確立することで、端末回線にアクセス コントロールを提供するために使用します。



(注) ラインパスワード保護を設定し、TACACS または拡張 TACACS を設定する場合、TACACS のユーザー名とパスワードの方が、ラインパスワードよりも優先されます。まだセキュリティポリシーを実装していない場合、AAA を使用することを推奨します。

手順の概要

1. **enable**
2. **configure terminal**
3. **line [aux | console | tty | vty] line-number [ending-line-number]**
4. **password password**
5. **login**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>line [aux console tty vty] line-number [ending-line-number]</p> <p>例 :</p> <pre>Device(config)# line console 0</pre>	<p>ライン コンフィギュレーション モードを開始します。</p>
ステップ 4	<p>password password</p> <p>例 :</p> <pre>Device(config-line)# secret word</pre>	<p>回線の端末または他のデバイスにパスワードを割り当てます。パスワードチェッカでは大文字と小文字が区別され、スペースを使用できます。たとえば、パスワード「Secret」とパスワード「secret」は異なるパスワードです。また、「two words」は有効なパスワードです。</p>
ステップ 5	<p>login</p> <p>例 :</p> <pre>Device(config-line)# login</pre>	<p>ログイン時のパスワードチェックをイネーブルにします。</p> <p>このコマンドの no 形式を使用してパスワードチェックを無効にすると、ラインパスワード検証を無効にできます。</p> <p>(注) login コマンドによって変更されるのはユーザー名および特権レベルだけであり、シェルは実行されません。したがって、autocommand は実行されません。この状況で autocommand を実行するには、Telnet セッションをデバイスに復帰 (ループバック) させる必要があります。この方法で autocommand を実装する場合は、デバイスがセキュアな Telnet セッションを使用するように設定されていることを確認してください。</p>
ステップ 6	<p>end</p> <p>例 :</p>	<p>回線 コンフィギュレーション モードを終了します。続いて、特権 EXEC モードに戻ります。</p>

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

ユーザー名認証の確立

ユーザー名ベースの認証システムを作成できます。これは、次のような場合に役立ちます。

- TACACS をサポートしないネットワークに、TACACS のようなユーザー名と暗号化されたパスワード認証システムを提供する場合
- 特殊なケース（たとえば、アクセスリストの確認、パスワードの確認なし、ログイン時の `autocommand` の実行、「エスケープなし」の状況など）に備えたログインを提供する場合

ユーザー名認証を確立するには、次のタスクを実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかを実行します。
 - **username name [nopassword | password password | password encryption-type encrypted password]**
 - **username name [access-class number]**
4. **username name [privilege level]**
5. **username name [autocommand command]**
6. **username name [noescape] [nohangup]**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかを実行します。 • username name [nopassword password password password encryption-type encrypted password]	暗号化されたパスワードを使用してユーザー名認証を確立します。 または

	コマンドまたはアクション	目的
	<ul style="list-style-type: none"> • username name [access-class number] 例 : Device(config)# username superuser password superpassword password 7 encrypted-password Device(config)# username user1 access-class access-user	(任意) アクセスリストによるユーザー名認証を確立します。
ステップ 4	username name [privilege level] 例 : Device(config)# username user1 privilege 5	(任意) ユーザーの特権レベルを設定します。
ステップ 5	username name [autocommand command] 例 : Device(config)# username user1 autocommand show users	(任意) 自動実行されるコマンドを指定します。
ステップ 6	username name [noescape] [nohangup] 例 : Device(config)# username user1 noescape	(任意) 「エスケープなし」のログイン環境を設定します。
ステップ 7	end 例 : Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

次のタスク

キーワード **noescape** を指定すると、ユーザーは接続先のホストでエスケープ文字を使用できなくなります。**nohangup** 機能を使用すると、**autocommand** の使用後に接続が解除されません。



注意 **service password-encryption** コマンドを有効にしない限り、設定のパスワードはクリアテキストで表示されます。

MS-CHAP を使用した PPP 認証の定義

MS-CHAP を使用して PPP 認証を定義するには、インターフェイス コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **encapsulation ppp**
4. **interface** *type number*
5. **ppp authentication ms-chap** [**if-needed**] [*list-name* | **default**] [**callin**] [**one-time**]
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	encapsulation ppp 例： Device(config)# encapsulation ppp	PPP カプセル化をイネーブルにします。
ステップ 4	interface <i>type number</i> 例： Device(config)# interface gigabitethernet 1/0/2	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 5	ppp authentication ms-chap [if-needed] [<i>list-name</i> default] [callin] [one-time] 例： Device(config-if)# ppp authentication ms-chap default callin	MS-CHAP を使用して PPP 認証を定義します。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

次のタスク

あるインターフェイスで **ppp authentication ms-chap** を設定する場合、PPP 接続を開始するそのインターフェイスに着信するすべてのコールは、MS-CHAP を使用して認証する必要があります。**callin** キーワードを指定して **ppp authentication** コマンドを設定すると、アクセスサーバーは、リモートデバイスがコールを開始した場合にだけ、リモートデバイスの認証を行います。

認証方式リストと **one-time** キーワードを使用できるのは、AAA を有効にした場合だけです。TACACS または拡張 TACACS を有効にしている場合は、使用できません。**ppp authentication** コマンドを使用して認証方式リストの名前を指定すると、PPP は、指定した方式リストに定義されている方式を使用して、接続を認証しようとします。AAA をイネーブルにし、名前で定義されている方式リストがない場合、PPP は、デフォルトに定義されている方式を使用して接続を認証しようとします。**one-time** キーワードを指定して **ppp authentication** コマンドを使用すると、認証中にワンタイムパスワードをサポートできます。

if-needed キーワードを使用できるのは、TACACS または拡張 TACACS を使用している場合だけです。**if-needed** キーワードを指定して **ppp authentication** コマンドを使用することは、現在のコール期間中にリモートデバイスがまだ認証されていない場合にだけ、PPP が MS-CHAP を介してリモートデバイスを認証することを示します。リモートデバイスが、標準のログイン手順で認証を受け、EXEC プロンプトから PPP を開始した場合、**ppp authentication chap if-needed** が設定されていれば、PPP は MS-CHAP を介して認証しません。



- (注) MS-CHAP を使用する PPP 認証と、ユーザー名認証を併用する場合、ローカル ユーザー名/パスワードデータベースに MS-CHAP シークレットを含める必要があります。

認証の設定例

例：方式リストの設定

たとえば、システム管理者が、すべてのインターフェイスに同じ認証方式を使用して PPP 接続を認証する、というセキュリティ ソリューションを決定したとします。RADIUS グループでは、まず認証情報のために R1 に接続し、応答がない場合、R2 に接続します。R2 が応答しない場合、TACACS+ グループの T1 に接続し、T1 が応答しない場合、T2 に接続します。すべての指定したサーバーが応答しなかった場合、認証はアクセスサーバー自体のローカルユーザー名データベースで行われます。このソリューションを実装するには、システム管理者が次のコマンドを入力してデフォルトの方式リストを作成します。

```
Device> enable
Device# configure terminal
Device(config)# aaa authentication ppp default group radius group tacacs+ local
Device(config)# exit
```

この例では、「default」が方式リストの名前です。この方式リストにプロトコルを含める場合、名前の後に、照会される順で指定します。デフォルトのリストは、すべてのインターフェイスに自動的に適用されます。

リモートユーザーがネットワークにダイヤルインしようとする、ネットワークアクセスサーバーは、まず R1 に認証情報を照会します。ユーザーが R1 から認証されると、R1 からネットワークアクセスサーバーに対して PASS 応答が発行され、ユーザーはネットワークにアクセスできるようになります。R1 から FAIL 応答が返されると、ユーザーはアクセスを拒否され、セッションは終了します。R1 が応答しない場合、ネットワークアクセスサーバーでは ERROR として処理され、認証情報について R2 に照会されます。このパターンは、ユーザーが認証または拒否されるか、セッションが終了するまで、残りの指定した方式について続行されます。

FAIL 応答は ERROR とまったく異なる点に注意してください。FAIL とは、適用可能な認証データベースに含まれる、認証の成功に必要な基準をユーザーが満たしていないことを示します。認証は FAIL 応答で終了します。ERROR とは、認証の照会に対してサーバーが応答しなかったことを示します。そのため、認証は試行されません。ERROR が検出された場合にだけ、認証方式リストに定義されている次の認証方式が AAA によって選択されます。

たとえば、システム管理者が、1つのインターフェイス、または一部のインターフェイスにだけ方式リストを適用するとします。この場合、システム管理者は名前付き方式リストを作成し、その名前付きリストを対象のインターフェイスに適用します。次に、システム管理者が、インターフェイス 3 にだけ適用する認証方式を実装する場合の例を示します。

```
Device> enable
Device# configure terminal
Device(config)# Device(config)#
Device(config)# aaa authentication ppp server-group1 group radius group tacacs+ local none
Device(config)# interface gigabitethernet 1/0/3
Device(config-if)# ppp authentication chap server-group1
Device(config-if)# end
```

この例では、「apple」が方式リストの名前です。また、この方式リストに含まれるプロトコルは、名前の後に、実行する順で指定されています。方式リストを作成すると、該当するインターフェイスに適用されます。AAA および PPP 認証コマンド両方の方式リスト名 (apple) は一致する必要があります。

次の例では、システム管理者がサーバーグループを使用し、PPP 認証の場合は R2 および T2 だけが有効であることを指定します。この場合、管理者は、メンバがそれぞれ R2 (172.16.2.7) と T2 (172.16.2.77) であるサーバーグループを定義する必要があります。この例では、RADIUS サーバーグループ「rad2only」は **aaa group server** コマンドを使用して次のように定義されます。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius rad2only
Device(config-sg-radius)# server 172.16.2.7
Device(config-sg-radius)# end
```

TACACS+ サーバーグループ「tac2only」は、**aaa group server** コマンドを使用して次のように定義されます。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server tacacs+ tac2only
Device(config-sg-tacacs)# server 172.16.2.77
Device(config-sg-tacacs)# end
```

次に、管理者はサーバー グループを使用して PPP 認証を適用します。この例では、PPP 認証用のデフォルト方式リストは **group rad2only**、**group tac2only**、**local** の順序に従います。

```
Device> enable
Device# configure terminal
Device(config)# aaa authentication ppp default group rad2only group tac2only local
Device(config)# exit
```

VTY 回線で方式リストを設定する場合、対応する方式リストを AAA に追加する必要があります。次の例は、VTY 回線の下に方式リストを設定する方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# line vty 0 4
Device(config-line)# authorization commands 15 auth1
Device(config-line)# exit
```

次の例は、AAA で方式リストを設定する方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization commands 15 auth1 group tacacs+
Device(config)# exit
```

VTY 回線で方式リストを設定しない場合、デフォルトの方式リストを AAA に追加する必要があります。次の例は、方式リストを使用しない VTY 設定を示しています。

```
Device> enable
Device# configure terminal
Device(config)# line vty 0 4
Device(config-line)# end
```

次の例は、デフォルトの方式リストを設定する方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization commands 15 default group tacacs+
Device(config)# exit
```

例：RADIUS 認証

ここでは、RADIUS を使用する 2 つの設定例を紹介します。

次に、RADIUS を使用して認証および認可を行うようにルータを設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa authentication login radius-login group radius local
```

```

Device(config)# aaa authentication ppp radius-ppp if-needed group radius
Device(config)# aaa authorization exec default group radius if-authenticated
Device(config)# aaa authorization network default group radius
Device(config)# line 3
Device(config-line)# login authentication radius-login
Device(config-line)# exit
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ppp authentication radius-ppp
Device(config-if)# end

```

この RADIUS 認証および認可設定のサンプル行は、次のように定義されます。

- `aaa authentication login radius-login group radius local` コマンドを実行すると、ルータは、ログインプロンプトで認証に RADIUS を使用するように設定されます。RADIUS がエラーを返すと、ユーザーはローカル データベースを使用して認証されます。
- `aaa authentication ppp radius-ppp if-needed group radius` コマンドを実行すると、ユーザーがまだログインしていない場合、Cisco IOS XE ソフトウェアは CHAP または PAP による PPP 認証を使用するように設定されます。EXEC 施設がユーザーを認証すると、PPP 認証は実行されません。
- `aaa authorization exec default group radius if-authenticated` コマンドを実行すると、`autocommand` や特権レベルなど、EXEC 認可時に使用される情報について、RADIUS データベースに照会されます。ただし、ユーザーの認証が成功した場合にだけ、権限が付与されます。
- `aaa authorization network default group radius` コマンドを実行すると、ネットワーク認可、アドレス割り当て、および他のアクセス リストについて RADIUS に照会されます。
- `login authentication radius-login` コマンドを使用すると、ライン 3 について `radius-login` 方式リストが有効になります。
- `ppp authentication radius-ppp` コマンドを使用すると、シリアルインターフェイス 0 について `radius-ppp` 方式リストが有効になります。

次に、ユーザー名とパスワードの入力を求め、その内容を確認し、ユーザーの EXEC レベルを認可し、特権レベル 2 の認可方式として指定するように、ルータを設定する例を示します。この例では、ユーザー名プロンプトにローカルユーザー名を入力すると、そのユーザー名が認証に使用されます。

ローカルデータベースを使用してユーザーが認証されると、RADIUS 認証からのデータは保存されないため、RADIUS を使用する EXEC 認可は失敗します。また、この方式リストではローカルデータベースを使用して `autocommand` を検索します。`autocommand` がない場合、ユーザーは EXEC ユーザーになります。次に、ユーザーが特権レベル 2 に設定されているコマンドを発行しようとする、TACACS+ を使用してコマンドの認可が試行されます。

```

Device> enable
Device# configure terminal
Device(config)# aaa authentication login default group radius local
Device(config)# aaa authorization exec default group radius local
Device(config)# aaa authorization command 2 default group tacacs+ if-authenticated
Device(config)# radius server radserver
Device(config-sg-radius)# address ipv4 10.2.3.1
Device(config-sg-radius)# exit
Device(config)# radius-server attribute 44 include-in-access-req

```

```
Device(config)# radius-server attribute 8 include-in-access-req
Device(config)# end
```

この RADIUS 認証および認可設定のサンプル行は、次のように定義されます。

- `aaa authentication login default group radius local` コマンドにより、RADIUS（RADIUS が応答しない場合はルータのローカル ユーザー データベース）がユーザー名およびパスワードを確認するように指定します。
- `aaa authorization exec default group radius local` コマンドにより、RADIUS を使用してユーザーが認証される場合、ユーザーの EXEC レベルの設定に RADIUS 認証情報を使用するように指定します。RADIUS 情報が使用されない場合、このコマンドにより、EXEC 認可にローカル ユーザー データベースが使用されるように指定します。
- `aaa authorization command 2 default group tacacs+ if-authenticated` コマンドにより、すでにユーザーの認証が成功している場合、特権レベル 2 に設定されているコマンドに TACACS+ 認可を指定します。
- `radius-server attribute 44 include-in-access-req` コマンドにより、`access-request` パケットで RADIUS 属性 44（Acct-Session-ID）を送信します。
- `radius-server attribute 8 include-in-access-req` コマンドにより、`access-request` パケットで RADIUS 属性 8（Framed-IP-Address）を送信します。

例：TACACS 認証

次に、PPP 認証に使用するセキュリティ プロトコルとして TACACS+ を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication ppp test group tacacs+ local
Device(config)# interface gigabitethernet 1/1/2
Device(config-if)# ppp authentication chap pap test
Device(config-if)# exit
Device(config)# tacacs server server1
Device(config-server-tacacs)# address ipv4 192.0.2.3
Device(config-server-tacacs)# key key1
Device(config-server-tacacs)# end
```

この TACACS+ 認証設定のサンプル行は、次のように定義されます。

- `aaa new-model` コマンドは、AAA セキュリティ サービスをイネーブルにします。
- `aaa authentication` コマンドにより、PPP を実行するシリアルインターフェイスに使用する方式リスト「test」を定義します。キーワード `group tacacs+` は、TACACS+ を介して認証を実行することを示します。認証中に TACACS+ から何らかのエラーが返される場合、キーワード `local` は、ネットワーク アクセス サーバー上のローカルデータベースを使用して認証が試行されることを示します。
- `interface` コマンドにより、回線を選択します。
- `ppp authentication` コマンドにより、この回線に test 方式リストを適用します。

- **address ipv4** コマンドにより、TACACS+ デーモンが 192.0.2.3 という IP アドレスを持っていると指定します。
- **key** コマンドにより、共有暗号キーが「key1」になるように定義します。

次に、PPP に AAA 認証を設定する例を示します。

```
Device(config)# aaa authentication ppp default if-needed group tacacs+ local
```

この例のキーワード **default** は、デフォルトですべてのインターフェイスに PPP 認証が適用されることを示します。**if-needed** キーワードは、ユーザーが ASCII ログイン手順を介してすでに認証済みの場合、PPP は不要なので、スキップできることを示します。認証が必要な場合、**group tacacs+** キーワードは、TACACS+ を介して認証が実行されることを示します。認証中に TACACS+ から何らかのエラーが返される場合、キーワード **local** は、ネットワーク アクセス サーバー上のローカル データベースを使用して認証が試行されることを示します。

次に、PAP に同じ認証アルゴリズムを作成し、「default」ではなく「MIS-access」の方式リストを呼び出す例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa authentication ppp MIS-access if-needed group tacacs+ local
Device(config)# interface gigabitethernet 1/1/2
Device(config)# ppp authentication pap MIS-access
Device(config)# end
```

この例では、リストはどのインターフェイスにも適用されないため（自動的にすべてのインターフェイスに適用されるデフォルトリストとは異なります）、管理者は **interface** コマンドを使用して、この認証スキームを適用するインターフェイスを選択する必要があります。次に、管理者は **ppp authentication** コマンドを使用して、選択したインターフェイスにこの方式リストを適用する必要があります。

例 : Kerberos 認証

ログイン認証方式として Kerberos を指定するには、次のコマンドを使用します。

```
Device> enable
Device# configure terminal
Device(config)# aaa authentication login default krb5
Device(config)# end
```

PPP に Kerberos 認証を指定するには、次のコマンドを使用します。

```
Device> enable
Device# configure terminal
Device(config)# aaa authentication ppp default krb5
Device(config)# end
```

例 : AAA スケーラビリティ

次に、セキュリティプロトコルとして RADIUS による AAA を使用する一般的なセキュリティ設定例を示します。この例では、ネットワーク アクセス サーバーは、16 バックアッププロセスを割り当てて PPP に対する AAA 要求を処理するように設定されています。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# radius server radserver
Device(config-sg-radius)# address ipv4 radius-host
Device(config-sg-radius)# key myRaDiUSpassWoRd
Device(config-sg-radius)# exit
Device(config)# radius-server configure-nas
Device(config)# username root password ALongPassword
Device(config)# aaa authentication ppp dialins group radius local
Device(config)# aaa authentication login admins local
Device(config)# aaa authorization network default group radius local
Device(config)# aaa accounting network default start-stop group radius
Device(config)# aaa processes 16
Device(config)# line 1 16
Device(config-line)# autoselect ppp
Device(config-line)# autoselect during-login
Device(config-line)# login authentication admins
Device(config-line)# modem dialin
Device(config-line)# exit
Device(config)# interface gigabitethernet 1/2/0
Device(config-if)# group-range 1 16
Device(config-if)# encapsulation ppp
Device(config-if)# ppp authentication pap dialins
Device(config-if)# end
```

この RADIUS AAA 設定のサンプル行は、次のように定義されます。

- **aaa new-model** コマンドは、AAA ネットワーク セキュリティ サービスをイネーブルにします。
- **address ipv4 {hostname | host-address}** コマンドは RADIUS サーバーホストの名前を定義します。
- **key** コマンドは、ネットワーク アクセス サーバーと RADIUS サーバー ホストの間の共有秘密テキスト文字列を定義します。
- **radius-server configure-nas** コマンドは、デバイスが最初に起動したときに、シスコルータまたはアクセスサーバーがスタティックルートと IP プール定義について RADIUS サーバーに照会するように定義します。
- **username** コマンドはユーザー名とパスワードを定義します。これらの情報は、PPP パスワード認証プロトコル (PAP) の発信元身元確認に使用されます。
- **aaa authentication ppp dialins group radius local** コマンドで、まず RADIUS 認証を指定する認証方式リスト「dialins」を定義します。次に、(RADIUS サーバーが応答しない場合) PPP を使用するシリアル回線でローカル認証が使用されます。
- **aaa authentication login admins local** コマンドは、ログイン認証に別の方式リスト「admins」を定義します。
- **aaa authorization network default group radius local** コマンドは、アドレスと他のネットワーク パラメータを RADIUS ユーザーに割り当てるために使用されます。
- **aaa accounting network default start-stop group radius** コマンドは、PPP の使用状況を追跡します。

- **aaa processes** コマンドにより、PPP に対する AAA 要求を処理するために 16 個のバックグラウンドプロセスを割り当てます。
- **line** コマンドはコンフィギュレーション モードをグローバル コンフィギュレーションからライン コンフィギュレーションに切り替え、設定対象の回線を指定します。
- **autoselect ppp** コマンドは、選択した回線上で PPP セッションを自動的に開始できるようにします。
- **autoselect during-login** コマンドを使用すると、Return キーを押さずにユーザ名およびパスワードのプロンプトが表示されます。ユーザがログインすると、autoselect 機能（この場合は PPP）が開始します。
- **login authentication admins** コマンドは、ログイン認証に「admins」方式リストを適用します。
- **modem dialin** コマンドは、選択した回線に接続されているモデムを設定し、着信コールだけを受け入れるようにします。
- **interface group-async** コマンドは、非同期インターフェイス グループを選択して定義します。
- **group-range** コマンドは、インターフェイス グループ内のメンバー非同期インターフェイスを定義します。
- **encapsulation ppp** コマンドは、指定のインターフェイスに使用されるカプセル化方式として PPP を設定します。
- **ppp authentication pap dialins** コマンドは「dialins」方式リストを指定したインターフェイスに適用します。

例：AAA 認証のログインバナーおよび Failed-Login バナーの設定

次に、ユーザーがシステムにログインするときに表示されるログインバナー（この場合、「Unauthorized Access Prohibited」というフレーズ）を設定する例を示します。アスタリスク (*) はデリミタとして使用されます。RADIUS はデフォルト ログイン認証方式として指定されます。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication banner *Unauthorized Access Prohibited*
Device(config)# aaa authentication login default group radius
Device(config)# end
```

この設定によって、次のログインバナーが表示されます。

```
Unauthorized Access Prohibited
Username:
```

次の例では、ユーザーがシステムにログインしようとして失敗すると表示される Failed-Login バナー（この場合、「Failed login. Try again」というフレーズ）を設定する方法を示します。ア

スタリスク (*) はデリミタとして使用されます。RADIUS はデフォルト ログイン認証方式として指定されます。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication banner *Unauthorized Access Prohibited*
Device(config)# aaa authentication fail-message *Failed login. Try again.*
Device(config)# aaa authentication login default group radius
Device(config)# end
```

この設定によって、次のログインバナーおよび Failed-Login バナーが表示されます。

```
Unauthorized Access Prohibited
Username:
Password:
Failed login. Try again.
```

例：AAA パケットオブディスコネクト サーバー キー

次に、パケットオブディスコネクト (POD) を設定する例を示します。その結果、特定のセッション属性が指定されると、ネットワーク アクセス サーバー (NAS) の接続が終了します。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication ppp default radius
Device(config)# aaa accounting network default start-stop radius
Device(config)# aaa accounting delay-start
Device(config)# aaa pod server server-key xyz123
Device(config)# radius server non-standard
Device(config-sg-radius)# address ipv4 10.2.1.1
Device(config-sg-radius)# key rad123
Device(config-sg-radius)# end
```

例：二重認証

ここでは、二重認証に使用できる設定例を示します。実際のネットワークおよびセキュリティ要件によっては、この例とは大幅に異なる可能性があります。



- (注) 設定例には、特定の IP アドレスと他の特定の情報が含まれます。この情報は説明のための例であり、実際の設定には異なる IP アドレス、異なるユーザー名とパスワード、異なる認可ステートメントを使用します。

例：二重認証による AAA のローカルホストの設定

次の 2 つの例では、PPP とログイン認証、およびネットワークと EXEC 認可に AAA を使用するようにローカルホストを設定する方法を示します。例はそれぞれ RADIUS の例と TACACS+ の例です。

いずれの例でも、先頭の 3 行で AAA を設定し、特定のサーバーを AAA サーバーとして設定しています。続く 2 行で PPP およびログイン認証に AAA を設定し、最後の 2 行でネットワークおよび EXEC 認可を設定します。最後の行が必要なのは、**access-profile** コマンドを **autocommand** として実行する場合だけです。

次に、RADIUS AAA サーバーを使用するデバイス設定の例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# radius server radsrver
Device(config-sg-radisr)# address ipv4 securesrver
Device(config-sg-radius)# key myradiuskey
Device(config-sg-radius)# exit
Device(config)# aaa authentication ppp default group radius
Device(config)# aaa authentication login default group radius
Device(config)# aaa authorization network default group radius
Device(config)# aaa authorization exec default group radius
Device(config)# end
```

次に、TACACS+ サーバーを使用するデバイス設定の例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# tacacs server server1
Device(config-server-tacacs)# address ipv4 192.0.2.3
Device(config-server-tacacs)# key mytacacskey
Device(config-server-tacacs)# exit
Device(config)# aaa authentication ppp default group tacacs+
Device(config)# aaa authentication login default group tacacs+
Device(config)# aaa authorization network default group tacacs+
Device(config)# aaa authorization exec default group tacacs+
Device(config)# end
```

例：第 1 段階の PPP 認証と許可に関する AAA サーバーの設定

次に、AAA サーバーでの設定例を示します。また、RADIUS 用の AAA 設定例の一部を示します。

TACACS+ サーバーも同様に設定できます（「TACACS による設定完了の例」を参照してください）。

この例では、二重認証の第 1 段階で CHAP によって認証される「hostx」というリモートホストに関する認証/認可を定義します。ACL AV ペアは、リモートホストによる Telnet 接続をローカルホストに制限しています。ローカルホストの IP アドレスは 10.0.0.2 です。

次に、RADIUS 用の AAA サーバーの設定例を一部示します。

```
hostx Password = "welcome"
      User-Service-Type = Framed-User,
```

```

Framed-Protocol = PPP,
cisco-avpair = "lcp:interface-config=ip unnumbered fastethernet 0",
cisco-avpair = "ip:inacl#3=permit tcp any 172.21.114.0 0.0.0.255 eq telnet",
cisco-avpair = "ip:inacl#4=deny icmp any any",
cisco-avpair = "ip:route#5=10.0.0.0 255.0.0.0",
cisco-avpair = "ip:route#6=10.10.0.0 255.0.0.0",
cisco-avpair = "ipx:inacl#3=deny any"

```

例：第2段階の Per-User 認証と許可に関する AAA サーバーの設定

ここでは、RADIUSサーバーでのAAA設定例の一部を示します。これらの設定では、ユーザー名が「user1」のユーザーの認証と許可を定義します。このユーザーは、二重認証の第2段階でユーザー認証されます。

TACACS+サーバーも同様に設定できます

3つの例は、**access-profile** コマンドの3つの各形式で使用できるRADIUS AAA設定の例を示します。

最初の例は、**access-profile** コマンドのデフォルトの形式（キーワードなし）で機能するAAA設定例の一部を示します。1つのACL AVペアのみが定義されます。また、この例では **autocommand** として **access-profile** コマンドも設定します。

```

user1 Password = "welcome"
      User-Service-Type = Shell-User,
      cisco-avpair = "shell:autocmd=access-profile"
      User-Service-Type = Framed-User,
      Framed-Protocol = PPP,
      cisco-avpair = "ip:inacl#3=permit tcp any host 10.0.0.2 eq telnet",
      cisco-avpair = "ip:inacl#4=deny icmp any any"

```

2番目の例は、**access-profile** コマンドの **access-profile merge** 形式で機能するAAA設定例の一部を示します。また、この例では **autocommand** として **access-profile merge** コマンドも設定します。

```

user1 Password = "welcome"
      User-Service-Type = Shell-User,
      cisco-avpair = "shell:autocmd=access-profile merge"
      User-Service-Type = Framed-User,
      Framed-Protocol = PPP,
      cisco-avpair = "ip:inacl#3=permit tcp any any"
      cisco-avpair = "ip:route=10.0.0.0 255.255.0.0",
      cisco-avpair = "ip:route=10.1.0.0 255.255.0.0",
      cisco-avpair = "ip:route=10.2.0.0 255.255.0.0"

```

3番目の例は、**access-profile** コマンドの **access-profile replace** 形式で機能するAAA設定例の一部を示します。また、この例では **autocommand** として **access-profile replace** コマンドも設定します。

```

user1 Password = "welcome"
      User-Service-Type = Shell-User,
      cisco-avpair = "shell:autocmd=access-profile replace"
      User-Service-Type = Framed-User,
      Framed-Protocol = PPP,
      cisco-avpair = "ip:inacl#3=permit tcp any any",
      cisco-avpair = "ip:inacl#4=permit icmp any any",
      cisco-avpair = "ip:route=10.10.0.0 255.255.0.0",

```

```
cisco-avpair = "ip:route=10.11.0.0 255.255.0.0",
cisco-avpair = "ip:route=10.12.0.0 255.255.0.0"
```

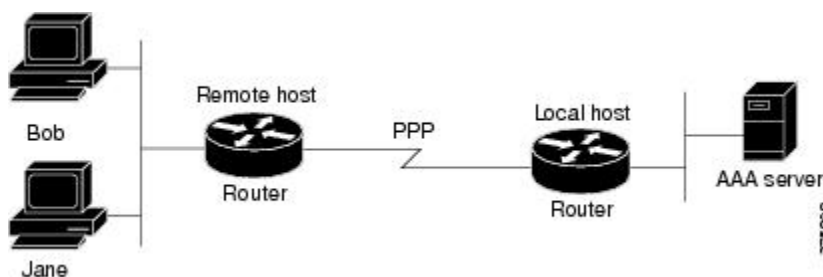
例：TACACSによる設定完了

この例では、リモートホスト（二重認証の第1段階で使用）および特定のユーザー（二重認証の第2段階で使用）の両方向けの、TACACS+ 認可プロファイルの設定を示します。

この設定例は、リモートホスト「hostx」および3ユーザー（ユーザー名が「user_default」、「user_merge」、および「user_replace」）のTACACS+サーバー上にある認証/許可プロファイルを示します。これら3つのユーザー名の設定は、**access-profile** コマンドの3種類のフォームに対応する異なる設定を示しています。また、3つのユーザー設定は、**access-profile** コマンドの各形式について **autocommand** の設定方法も示しています。

次の図に、トポロジを示します。図の後に、TACACS+ 設定ファイルの例を示します。

図 3: 二重認証のトポロジ例



この設定例は、リモートホスト「hostx」および3ユーザー（ユーザー名が「user_default」、「user_merge」、および「user_replace」）のTACACS+サーバー上にある認証/許可プロファイルを示します。

```
key = "mytacacskey"
default authorization = permit
#-----Remote Host (BRI)-----
#
# This allows the remote host to be authenticated by the local host
# during fist-stage authentication, and provides the remote host
# authorization profile.
#
#-----
user = hostx
{
  login = cleartext "welcome"
  chap = cleartext "welcome"
  service = ppp protocol = lcp {
    interface-config="ip unnumbered fastethernet 0"
  }
  service = ppp protocol = ip {
    # It is important to have the hash sign and some string after
    # it. This indicates to the NAS that you have a per-user
    # config.
    inacl#3="permit tcp any 172.21.114.0 0.0.0.255 eq telnet"
    inacl#4="deny icmp any any"
    route#5="10.0.0.0 255.0.0.0"
    route#6="10.10.0.0 255.0.0.0"
  }
}
```

```

        service = ppp protocol = ipx {
            # see previous comment about the hash sign and string, in protocol = ip
            inacl#3="deny any"
        }
    }
#----- "access-profile" default user "only acls" -----
#
# Without arguments, access-profile removes any access-lists it can find
# in the old configuration (both per-user and per-interface), and makes sure
# that the new profile contains ONLY access-list definitions.
#
#-----
user = user_default
{
    login = cleartext "welcome"
    chap = cleartext "welcome"
    service = exec
    {
        # This is the autocommand that executes when user_default logs in.
        autocmd = "access-profile"
    }
    service = ppp protocol = ip {
        # Put whatever access-lists, static routes, whatever
        # here.
        # If you leave this blank, the user will have NO IP
        # access-lists (not even the ones installed prior to
        # this)!
        inacl#3="permit tcp any host 10.0.0.2 eq telnet"
        inacl#4="deny icmp any any"
    }
    service = ppp protocol = ipx {
        # Put whatever access-lists, static routes, whatever
        # here.
        # If you leave this blank, the user will have NO IPX
        # access-lists (not even the ones installed prior to
        # this)!
    }
}
#----- "access-profile merge" user -----
#
# With the 'merge' option, first all old access-lists are removed (as before),
# but then (almost) all AV pairs are uploaded and installed. This will allow
# for uploading any custom static routes, sap-filters, and so on, that the user
# may need in his or her profile. This needs to be used with care, as it leaves
# open the possibility of conflicting configurations.
#
#-----
user = user_merge
{
    login = cleartext "welcome"
    chap = cleartext "welcome"
    service = exec
    {
        # This is the autocommand that executes when user_merge logs in.
        autocmd = "access-profile merge"
    }
    service = ppp protocol = ip
    {
        # Put whatever access-lists, static routes, whatever
        # here.
        # If you leave this blank, the user will have NO IP
        # access-lists (not even the ones installed prior to
        # this)!
        inacl#3="permit tcp any any"
    }
}

```

```

route#2="10.0.0.0 255.255.0.0"
route#3="10.1.0.0 255.255.0.0"
route#4="10.2.0.0 255.255.0.0"
}
service = ppp protocol = ipx
{
    # Put whatever access-lists, static routes, whatever
    # here.
    # If you leave this blank, the user will have NO IPX
    # access-lists (not even the ones installed prior to
    # this)!
}
}
----- "access-profile replace" user -----
#
# With the 'replace' option, ALL old configuration is removed and ALL new
# configuration is installed.
#
# One caveat: access-profile checks the new configuration for address-pool and
# address AV pairs. As addresses cannot be renegotiated at this point, the
# command will fail (and complain) when it encounters such an AV pair.
# Such AV pairs are considered to be "invalid" for this context.
-----
user = user_replace
{
    login = cleartext
    t
    "
welcome
"

    chap = cleartext "welcome"
    service = exec
    {
        # This is the autocommand that executes when user_replace logs in.
        autocmd = "access-profile replace"
    }
    service = ppp protocol = ip
    {
        # Put whatever access-lists, static routes, whatever
        # here.
        # If you leave this blank, the user will have NO IP
        # access-lists (not even the ones installed prior to
        # this)!
        inacl#3="permit tcp any any"
        inacl#4="permit icmp any any"
        route#2="10.10.0.0 255.255.0.0"
        route#3="10.11.0.0 255.255.0.0"
        route#4="10.12.0.0 255.255.0.0"
    }
    service = ppp protocol = ipx
    {
        # put whatever access-lists, static routes, whatever
        # here.
        # If you leave this blank, the user will have NO IPX
        # access-lists (not even the ones installed prior to
        # this)!
    }
}
}

```

例：自動二重認証

次に、自動二重認証が設定された設定ファイル全体の例を示します。自動二重認証に適用されるコンフィギュレーションコマンドは、2つのアスタリスク（**）を使用した記述よりも優先されます。

```
Current configuration:
!
version 16.10
no service password-encryption
!
hostname myrouter
!
!
! **The following AAA commands are used to configure double authentication:
!
! **The following command enables AAA:
aaa new-model
! **The following command enables user authentication via the RADIUS AAA server:
!
aaa authentication login default none
aaa authentication ppp default group radius
! **The following command causes the remote user's authorization profile to be
! downloaded from the AAA server to the router when required:
!
aaa authorization network default group radius
!
enable password mypassword
!
ip host blue 172.21.127.226
ip host green 172.21.127.218
ip host red 172.21.127.114
ip domain-name example.com
ip name-server 172.16.2.75
!
!
interface GigabitEthernet0/0/0
 ip address 172.21.127.186 255.255.255.248
 no ip route-cache
 no ip mroute-cache
 no keepalive
 ntp disable
 no cdp enable
!
interface Virtual-Template1
 ip unnumbered loopback0
 no ip route-cache
 no ip mroute-cache
!
! **The following command specifies that device authentication occurs via PPP CHAP:
 ppp authentication chap
!
router eigrp 109
 network 172.21.0.0
 no auto-summary
!
ip default-gateway 172.21.127.185
no ip classless
ip route 172.21.127.114 255.255.255.255 172.21.127.113
! **Virtual profiles are required for double authentication to work:
virtual-profile virtual-template 1
```

```

dialer-list 1 protocol ip permit
no cdp run
! **The following command defines where the TACACS+ AAA server is:
tacacs server server1
address ipv4 172.16.57.35
! **The following command defines the key to use with TACACS+ traffic (required):
key mytacacskey
snmp-server community public RO
!
line con 0
  exec-timeout 0 0
  login authentication console
line aux 0
  transport input all
line vty 0 4
  exec-timeout 0 0
  password lab
!
end

```

認証設定の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	AAA Authentication	認証は、選択したセキュリティプロトコルに応じてログイン/パスワードダイアログ、チャレンジ/レスポンス、メッセージングサポート、および暗号化などのユーザーの識別方法を提供します。認証は、ユーザーに対してネットワークとネットワーク サービスへのアクセスを許可する前に、ユーザーの識別を行う方法です。

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



第 3 章

認可の設定

AAA 認可を使用すると、ユーザーが利用できるサービスを制限できます。AAA 認可をイネーブルにすると、ネットワーク アクセス サーバーはユーザーのプロファイルから取得した情報を使用して、ユーザーの設定を設定します。このプロファイルは、ローカル ユーザー データベースまたはセキュリティ サーバーにあります。認可が完了すると、ユーザー プロファイルの情報で許可されているサービスであれば、ユーザーは要求したサービスに対するアクセス権を付与されます。

- [許可設定の前提条件 \(83 ページ\)](#)
- [認可の設定の概要 \(84 ページ\)](#)
- [認可の設定方法 \(88 ページ\)](#)
- [許可の設定例 \(91 ページ\)](#)
- [認可の設定に関する追加情報 \(95 ページ\)](#)
- [許可設定の機能履歴 \(95 ページ\)](#)

許可設定の前提条件

名前付き方式リストを使用して認証を設定する前に、まず、次のタスクを実行する必要があります。

- ネットワークアクセスサーバーの認証、許可、およびアカウントिंग (AAA) を有効にします。
- AAA 認証を設定します。一般的に、認可は認証後に実行し、認証が適切に動作することに依存します。AAA 認証の設定方法については、「[認証の設定](#)」モジュールを参照してください。
- RADIUS または TACACS+ 認可を発行している場合、RADIUS または TACACS+ セキュリティ サーバーの特性を定義します。シスコのネットワーク アクセス サーバーを設定して RADIUS セキュリティサーバーと通信する方法の詳細については、「[RADIUS の設定](#)」の章を参照してください。シスコのネットワーク アクセス サーバーを設定して TACACS+ セキュリティサーバーと通信する方法の詳細については、「[TACACS+ の設定](#)」モジュールを参照してください。

- ローカル認可を発行している場合、**username** コマンドを使用して、特定のユーザーに関連付けられている権限を定義します。

認可の設定の概要

認可の名前付き方式リスト

許可方式リストによって、許可の実行方法とこれらの方式の実行順序が定義されます。方式リストは、順に照会する認可方式（RADIUS または TACACS+ など）を記述した指定リストです。方式リストを使用すると、許可に使用するセキュリティプロトコルを1つ以上指定できるため、最初の方式が失敗した場合のバックアップシステムを確保できます。Cisco IOS XE ソフトウェアでは、特定のネットワークサービスについてユーザーを許可するために最初の方式が使用されます。その方式が応答しない場合、リストの次の方式が選択されます。このプロセスは、リストのいずれかの認可方式と通信に成功するか、定義されているすべての方式が試行されるまで継続されます。



- (注) Cisco IOS XE ソフトウェアでは、前の方式からの応答がない場合にのみ、リストの次の許可方式が試行されます。このサイクルの任意の時点で許可が失敗した場合（つまり、セキュリティサーバーまたはローカルユーザー名データベースからユーザーサービスの拒否応答が返される場合）、許可プロセスは停止し、その他の許可方式は試行されません。

方式リストは、要求した認可タイプに固有です。

- Commands** : ユーザーが実行する EXEC モードコマンドに適用されます。コマンドの認可は、特定の特権レベルに関連付けられた、グローバル コンフィギュレーション コマンドなどのすべての EXEC モードコマンドについて、認可を試行します。
- EXEC** : ユーザー EXEC ターミナルセッションに関連付けられた属性に適用されます。
- Network** : ネットワーク接続に適用します。これには、PPP、SLIP、または ARAP 接続が含まれます。
- Reverse Access** : リバース Telnet セッションに適用されます。

方式の指定リストを作成すると、指定した許可タイプに対して特定の許可方式リストが定義されます。

定義されると、方式リストを特定の回線またはインターフェイスに適用してから、定義済み方式のいずれかを実行する必要があります。唯一の例外は、デフォルトの方式リスト（「default」という名前）です。名前付き方式リストを指定せずに、特定の許可タイプ用の **aaa authorization** コマンドが発行されると、名前付き方式リストが明示的に定義されている場合を除いて、すべてのインターフェイスまたは回線にデフォルトの方式リストが自動的に適用されます。（定義済みの方式リストは、デフォルトの方式リストに優先します）。デフォルトの方式リストが定義されていない場合、デフォルトでローカル認可が実行されます。

AAA 認可方式

AAA は 5 種類の認可方式をサポートしています。

- **TACACS+** : ネットワークアクセスサーバーは、TACACS+セキュリティデーモンと許可情報を交換します。TACACS+ 認可は、属性値ペアを関連付けることでユーザーに特定の権限を定義します。属性ペアは適切なユーザーとともに TACACS+セキュリティサーバーのデータベースに保存されます。
- **If-Authenticated** : ユーザーが認証に成功した場合、ユーザーは要求した機能にアクセスできます。
- **None** : ネットワークアクセスサーバーは、認可情報を要求しません。認可は、この回線/インターフェイスで実行されません。
- **Local** : ルータまたはアクセスサーバーは、**username** コマンドの定義に従って、ローカルデータベースに問い合わせ、たとえばユーザーに固有の権限を許可します。ローカルデータベースを介して制御できるのは、一部の機能だけです。
- **RADIUS** : ネットワークアクセスサーバーは RADIUS セキュリティサーバーからの許可情報を要求します。RADIUS 認可では、属性を関連付けることでユーザーに固有の権限を定義します。属性は適切なユーザーとともに RADIUS サーバー上のデータベースに保存されます。



- (注) CSCuc32663 では、パスワードおよび認可ログは、TACACS+、LDAP、または RADIUS セキュリティサーバーへ送信される前にマスクされます。マスクされていない情報を TACACS+、LDAP または RADIUS セキュリティサーバーに送信するには、**aaa authorization commands visible-keys** コマンドを使用します。

認可方式

ネットワークアクセスサーバーから TACACS+セキュリティサーバーを介して認可情報を要求するには、**group tacacs+ method** キーワードを指定して **aaa authorization** コマンドを使用します。TACACS+セキュリティサーバーを使用して認可を設定する詳細な方法については、「TACACS+ の設定」の章を参照してください。TACACS+ サーバーが、PPP や ARA などのネットワークサービスの使用を認可できるようにする例については、「TACACS 認可の例」を参照してください。

ユーザーが認証済みであれば、要求した機能へのアクセスを許可するには、**if-authenticated method** キーワードを指定して **aaa authorization** コマンドを使用します。この方式を選択する場合は、すべての要求した機能は、認証済みユーザーに自動的に許可されます。

特定のインターフェイスまたは回線から認可を実行したくない場合があります。指定した回線またはインターフェイスで許可動作を停止するには、**none method** キーワードを使用します。この方式を選択すると、すべてのアクションについて認可はディセーブルになります。

ローカル許可を選択するには（つまり、ルータまたはアクセスサーバーがローカルユーザーデータベースに問い合わせ、ユーザーが使用可能な機能を決定する場合）、**local method** キーワードを指定して **aaa authorization** コマンドを使用します。ローカル許可に関連する機能は、**username** グローバル コンフィギュレーション コマンドを使用して定義します。許可されている機能のリストについては、「認証の設定」の章を参照してください。

ネットワークアクセスサーバーから RADIUS セキュリティサーバーを介して許可を要求するには、**radius method** キーワードを使用します。RADIUS セキュリティサーバーを使用して認可を設定する詳細な方法については、「RADIUS の設定」の章を参照してください。

ネットワークアクセスサーバーから RADIUS セキュリティサーバーを介して許可を要求するには、**group radius method** キーワードを指定して **aaa authorization** コマンドを使用します。RADIUS セキュリティサーバーを使用して認可を設定する詳細な方法については、「RADIUS の設定」の章を参照してください。RADIUS サーバーがサービスを認可できるようにする例については、「RADIUS 認可の例」を参照してください。



(注) SLIP の認可方式リストは、関連インターフェイスで PPP に設定されているすべての方式に従います。特定のインターフェイスに定義および適用されるリストがない場合（または PPP 設定が指定されていない場合）、認可のデフォルト設定が適用されます。

方式リストとサーバグループ

サーバグループは、方式リストに使用する既存の RADIUS または TACACS+ サーバ ホストをグループ化する方法の 1 つです。次の図に、4 台のセキュリティサーバ（R1 と R2 は RADIUS サーバ、T1 と T2 は TACACS+ サーバ）が設置された一般的な AAA ネットワーク設定を示します。R1 と R2 で RADIUS サーバのグループを構成します。T1 と T2 で TACACS+ サーバのグループを構成します。

サーバグループを使用して、設定したサーバホストのサブセットを指定し、特定のサービスに使用します。たとえば、サーバグループを使用すると、R1 および R2 を別のサーバグループとして定義し、T1 および T2 を別のサーバグループとして定義できます。つまり、R1 と T1 を方式リストに指定できるか、または R2 と T2 を方式リストに指定できます。そのため、RADIUS および TACACS+ のリソースを割り当てる場合の柔軟性が高くなります。

サーバグループには、1 台のサーバに対して複数のホストエントリを含めることができます。エントリごとに固有の識別情報を設定します。固有の識別情報は、IP アドレスと UDP ポート番号の組み合わせで構成されます。これにより、RADIUS ホストとして定義されているさまざまなポートが、固有の AAA サービスを提供できるようになります。つまり、この固有識別情報を使用して、ある IP アドレスに位置する 1 台のサーバ上に複数の UDP ポートが存在する場合、それぞれの UDP ポートに対して RADIUS 要求を送信できます。1 台の RADIUS サーバ上にある異なる 2 つのホストエントリが 1 つのサービス（認可など）に設定されている場合、設定されている 2 番目のホストエントリは最初のホストエントリのフェールオーバーバックアップとして動作します。この例の場合、最初のホストエントリがアカウントングサービスの提供に失敗すると、同じデバイスに設定されている 2 番目のホストエントリを使用してア

カウンティングサービスを提供するように、ネットワークアクセスサーバーが試行します（試行される RADIUS ホスト エントリの順番は、設定されている順序に従います）。

サーバー グループの設定および DNIS 番号に基づくサーバー グループの設定の詳細については、「RADIUS の設定」または「TACACS+ の設定」の章を参照してください。

AAA 認可タイプ

Cisco IOS XE ソフトウェアは、5 種類の認可をサポートしています。

- **Commands** : ユーザーが実行する EXEC モードコマンドに適用されます。コマンドの認可は、特定の特権レベルに関連付けられた、グローバル コンフィギュレーション コマンドなどのすべての EXEC モードコマンドについて、認可を試行します。
- **EXEC** : ユーザー EXEC ターミナルセッションに関連付けられた属性に適用されます。
- **Network** : ネットワーク接続に適用します。これには、PPP、SLIP、または ARAP 接続が含まれます。
- **Reverse Access** : リバース Telnet セッションに適用されます。
- **Configuration** : AAA サーバーからのコンフィギュレーションのダウンロードに適用されません。
- **IP Mobile** : IP モバイルサービスの許可に適用されます。

承認タイプ

名前付き認可方式リストは、指定される認可の種類によって変わります。

ユーザー別に固有のセキュリティポリシーを適用する認可をイネーブにする方式リストを作成するには、**auth-proxy** キーワードを使用します。認証プロキシ機能の詳細については、このガイドの「Traffic Filtering and Firewalls」の部の「Configuring Authentication Proxy」を参照してください。

すべてのネットワーク関連サービス要求（SLIP、PPP、PPP NCP、ARAP など）について認可を有効にする方式リストを作成するには、**network** キーワードを使用します。

ユーザーが EXEC シェルを実行できるかどうかを認可で決定できるように方式リストを作成するには、**exec** キーワードを使用します。

特定の特権レベルに関連付けられた個々の EXEC コマンドについて認可を有効にする方式リストを作成するには、**commands** キーワードを使用します。これにより、指定されたコマンドレベル（0～15）に関連付けられているすべてのコマンドを認可できます。

リバース Telnet 機能について認可を有効にする方式リストを作成するには、**reverse-access** キーワードを使用します。

Cisco IOS XE ソフトウェアでサポートされている認可のタイプの詳細については、「AAA 認可タイプ」を参照してください。

認可の属性値ペア

RADIUS および TACACS+ の認可はいずれも、セキュリティサーバーのデータベースに保存されている属性を処理することで、ユーザーに固有の権限を定義します。RADIUS と TACACS+ のいずれも、属性はセキュリティサーバーに定義され、ユーザーに関連付けられ、ユーザーの接続に適用されるネットワーク アクセス サーバーに送信されます。

サポートされる RADIUS 属性のリストについては、「RADIUS 属性の概要および RADIUS IETF 属性」の章を参照してください。サポートされる TACACS+ の AV ペアのリストについては、「TACACS+ の設定」の章を参照してください。

認可の設定方法

名前付き方式リストによる AAA 認可の設定

名前付き方式リストを使用して AAA 認可を設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa authorization** {auth-proxy | network | exec | commands level | reverse-access | configuration | ipmobile} {default | list-name} [method1 [method2...]]
4. 次のいずれかを実行します。
 - **line** [aux | console | tty | vty] line-number [ending-line-number]
 - **interface** interface-type interface-number
5. 次のいずれかを実行します。
 - **authorization** {arap | commands level | exec | reverse-access} {default | list-name}
 - **ppp authorization** {default | list-name}
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	Device# configure terminal	
ステップ 3	aaa authorization {auth-proxy network exec commands level reverse-access configuration ipmobile} {default list-name} [method1 [method2...]] 例 : Device(config)# aaa authorization auth-proxy default	特定の認可タイプの認可方式リストを作成し、認可をイネーブルにします。
ステップ 4	次のいずれかを実行します。 <ul style="list-style-type: none"> • line [aux console tty vty] line-number [ending-line-number] • interface interface-type interface-number 例 : Device(config)# line 1 Device(config)# interface gigabitethernet 0/1/1	認可方式リストを適用する回線について、ライン コンフィギュレーション モードを開始します。 または、認可方式リストを適用するインターフェイスについて、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 5	次のいずれかを実行します。 <ul style="list-style-type: none"> • authorization {arap commands level exec reverse-access} {default list-name} • ppp authorization {default list-name} 例 : Device(config-line)# authorization commands default Device(config-if)# ppp authorization default	1つの回線または複数回線に認可リストを適用します。 または、1つのインターフェイスまたは複数インターフェイスに認可リストを適用します。
ステップ 6	end 例 : Device(config-line)# end Device(config-if)# end	回線コンフィギュレーションモードを終了します。続いて、特権 EXEC モードに戻ります。 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、特権 EXEC モードに戻ります。

グローバル コンフィギュレーション コマンドの認可のディセーブル化

commands キーワードを指定して **aaa authorization** コマンドを使用すると、その特権レベルに関連付けられているすべての EXEC モードコマンド（グローバル コンフィギュレーション コマンドを含む）に対して許可が試行されます。一部の EXEC レベル コマンドと同じコンフィ

ギューレーション コマンドもあるため、認可プロセスが混乱する可能性があります。 **no aaa authorization config-commands** を使用すると、ネットワーク アクセス サーバーがコンフィギュレーション コマンド認可の試行を停止します。

すべてのグローバルコンフィギュレーションコマンドについて AAA 認可をディセーブルにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
Device(config)# no aaa authorization config-commands	すべてのグローバル コンフィギュレーション コマンドについて認可をディセーブルにします。

コンソール上で AAA 認可をディセーブルにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。



- (注) デフォルトでコンソールの AAA 認可はディセーブルです。コンソールで AAA 許可が有効になっている場合は、AAA の設定段階で **no aaa authorization console** コマンドを設定して無効にします。ユーザー認証用のコンソールでは AAA をディセーブルにする必要があります。

コマンド	目的
Device(config)# no aaa authorization console	コンソールでの認証を無効にします。

リバース Telnet の認可の設定

Telnet は、リモートターミナル接続に使用される標準ターミナルエミュレーションプロトコルです。通常、ネットワーク アクセス サーバーに（主にダイヤルアップ接続経由で）ログインし、Telnet を使用してそのネットワーク アクセス サーバーから他のネットワーク デバイスにアクセスします。ただし、場合によっては、リバース Telnet セッションを確立する必要があります。リバース Telnet セッションでは、反対方向の Telnet 接続（つまり、ネットワーク内部から、ネットワーク周辺にあるネットワーク アクセス サーバーに対する接続）が確立されます。その接続によって、ネットワーク アクセス サーバーに接続しているモデムや他のデバイスへのアクセスを取得します。リバース Telnet は、ユーザーがネットワーク アクセス サーバーに接続されているモデム ポートに Telnet を送信できるようにすることで、ユーザーにダイヤルアウト機能を提供します。

リバース Telnet を介してアクセスできるポートのアクセス権を制御することが重要です。適切に制御しないと、たとえば、不正ユーザーがモデムに自由にアクセスし、着信コールをトラップして迂回させたり、不正な宛先にコールを送信したりする可能性があります。

リバース Telnet 時の認証は、Telnet 用の標準の AAA ログイン手順を介して実行されます。通常、Telnet またはリバース Telnet セッションを確立するには、ユーザーはユーザー名とパスワードを指定する必要があります。リバース Telnet 認可は、認証に加えて認可を必須にするこ

とで、追加（任意）レベルのセキュリティを提供します。リバース Telnet 認可をイネーブルにすることで、標準の Telnet ログイン手順を介してユーザー認証を完了した後に、RADIUS または TACACS+ を使用して、そのユーザーが非同期ポートにリバース Telnet アクセスを実行できるかどうかを認可できます。

リバース Telnet 認可には次の利点があります。

- リバース Telnet アクティビティを実行しているユーザーに、リバース Telnet を使用して特定の非同期ポートにアクセスする権限を付与することで、追加レベルの保護を実現しています。
- リバース Telnet 認可を管理できる（アクセス リスト以外の）代替方式があります。

ネットワーク アクセス サーバーが TACACS+ または RADIUS サーバーからの認可情報を要求するように設定してから、ユーザーによるリバース Telnet セッションの確立を許可するには、グローバル 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
<pre>Device(config)# aaa authorization reverse-access method1 [method2 ...]</pre>	<p>ネットワーク アクセス サーバーが認可情報を要求するように設定してから、ユーザーによるリバース Telnet セッションの確立を許可します。</p>

この機能によって、ネットワーク アクセス サーバーは、セキュリティ サーバー（RADIUS または TACACS+）からリバース Telnet 認可情報を要求できます。セキュリティ サーバー上のユーザーに固有のリバース Telnet 特権を設定する必要があります。

許可の設定例

例：TACACS 認可

次に、TACACS+ サーバーを使用して、PPP や ARA などのネットワーク サービスの使用を認可する例を示します。TACACS+ サーバーが使用不能の場合、または認可プロセス中にエラーが発生した場合、フォールバック方式（none）はすべての認可要求を許可することです。

```
Device(config)# aaa authorization network default group tacacs+ none
```

次に、TACACS+ を使用してネットワークの認可を許可する例を示します。

```
Device(config)# aaa authorization network default group tacacs+
```

次に、同じ許可を提供し、「mci」と「att」というアドレスプールも作成する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa authorization network default group tacacs+
Device(config)# interface gigabitethernet 01/1/
```

```
Device(config-if)# ip address-pool local
Device(config-if)# exit
Device(config)# ip local-pool mci 172.16.0.1 172.16.0.255
Device(config)# ip local-pool att 172.17.0.1 172.17.0.255
Device(config-if)# end
```

これらのアドレス プールは、TACACS デーモンによって選択できます。デーモンの設定例を次に示します。

```
user = mci_customer1 {
    login = cleartext "some password"
    service = ppp protocol = ip {
        addr-pool=mci
    }
}
user = att_customer1 {
    login = cleartext "some other password"
    service = ppp protocol = ip {
        addr-pool=att
    }
}
```

例：RADIUS 許可

次に、RADIUS を使用して認可を行うようにルータを設定する方法の例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization exec default group radius if-authenticated
Device(config)# aaa authorization network default group radius
Device(config)# radius server ip
Device(config-radius-server)# key sharedkey
Device(config-radius-server)# end
```

この RADIUS 認可設定のサンプル行は、次のように定義されます。

- **aaa authorization exec default group radius if-authenticated** コマンドで、ネットワークアクセスサーバーが RADIUS サーバーに接続して、ユーザーのログイン時にユーザーが EXEC シェルを起動する権限があるかどうかを決定するように設定します。ネットワークアクセスサーバーが RADIUS サーバーに接続するときにエラーが発生した場合、フォールバック方式は、ユーザーが適切に認証されていると CLI の起動を許可します。

返される RADIUS 情報を使用して、その接続に適用される autocommand または接続アクセスリストを指定できます。

- **aaa authorization network default group radius** コマンドにより、RADIUS を介するネットワーク許可を設定します。この操作は、アドレス割り当ての管理、アクセスリストのアプリケーション、および他の多様なユーザー別の数量に使用できます。



(注) この例ではフォールバック方式を指定していないため、何らかの理由で認可に失敗すると、RADIUS サーバーからの応答はありません。

例：リバース Telnet 許可

次に、ネットワーク アクセス サーバーが TACACS+ セキュリティ サーバーから認可情報を要求してから、ユーザーによるリバース Telnet セッションの確立を許可する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication login default group tacacs+
Device(config)# aaa authorization reverse-access default group tacacs+
Device(config)# tacacs server server1
Device(config-server-tacacs)# address ipv4 172.31.255.0
Device(config-server-tacacs)# timeout 90
Device(config-server-tacacs)# key sharedkey
Device(config-server-tacacs)# end
```

この TACACS+ リバース Telnet 認可設定のサンプル行は、次のように定義されます。

- **aaa new-model** コマンドは AAA を有効にします。
- **aaa authentication login default group tacacs+** コマンドで、ログイン時のユーザー認証のデフォルト方式として TACACS+ を指定します。
- リバース Telnet セッションを確立しようとしているときに、**aaa authorization reverse-access default group tacacs+** コマンドで、ユーザー認可の方式として TACACS+ を指定します。
- **tacacs server** コマンドで、TACACS+ サーバーを識別します。
- **timeout** コマンドで、ネットワーク アクセス サーバーが TACACS+ サーバーの応答を待機する期間を設定します。
- **key** コマンドで、ネットワーク アクセス サーバーと TACACS+ デーモン間のすべての TACACS+ 通信に使用される暗号キーを定義します。

次に、ネットワーク アクセス サーバー「maple」上のポート tty2、およびネットワーク アクセス サーバー「oak」上のポート tty5 に対するリバース Telnet アクセス権をユーザー pat に付与する汎用の TACACS+ サーバーを設定する例を示します。

```
user = pat
  login = cleartext lab
  service = raccess {
    port#1 = maple/tty2
    port#2 = oak/tty5
```



(注) この例では、「maple」と「oak」には、DNS 名またはエイリアスではなく、ネットワーク アクセス サーバーのホスト名が設定されています。

次に、TACACS+ サーバー (CiscoSecure) を設定して、ユーザー pat にリバース Telnet アクセス権を付与する例を示します。

```
user = pat
profile_id = 90
profile_cycle = 1
member = Tacacs_Users
```

```

service=shell {
default cmd=permit
}
service=raccess {
allow "c2511e0" "tty1" \.*"
refuse \.*" \.*" \.*"
password = clear "goaway"

```



(注) CiscoSecure は、バージョン 2.1(x)～バージョン 2.2(1) のコマンドライン インターフェイスを使用して、リバース Telnet だけをサポートしています。

空の「service=raccess {}」句は、リバース Telnet のネットワーク アクセス サーバー ポートに対して無条件のアクセス権をユーザーに許可しています。「service=raccess」句が存在しない場合、ユーザーはリバース Telnet のすべてのポートに対してアクセスを拒否されます。

TACACS+ の設定の詳細については、「TACACS+ の設定」の章を参照してください。CiscoSecure の設定の詳細については、『CiscoSecure Access Control Server User Guide』の version 2.1(2) 以降を参照してください。

次に、ネットワーク アクセス サーバーが RADIUS セキュリティ サーバーから認可を要求してから、ユーザーによるリバース Telnet セッションの確立を許可する例を示します。

```

Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication login default group radius
Device(config)# aaa authorization reverse-access default group radius
Device(config)# radius server ip
Device(config-radius-server)# key sharedkey
Device(config-radius-server)# address ipv4 172.31.255.0 auth-port 1645 acct-port 1646
Device(config-radius-server)# end

```

この RADIUS リバース Telnet 認可設定のサンプル行は、次のように定義されます。

- **aaa new-model** コマンドは AAA を有効にします。
- **aaa authentication login default group radius** コマンドで、ログイン時のユーザー認証のデフォルト方式として RADIUS を指定します。
- リバース Telnet セッションを確立しようとしているときに、**aaa authorization reverse-access default group radius** コマンドで、ユーザー認可の方式として RADIUS を指定します。
- **radius** コマンドで、RADIUS サーバーを指定します。
- **key** コマンドで、ネットワークアクセスサーバーと RADIUS デーモン間のすべての RADIUS 通信に使用される暗号キーを定義します。

次に、ネットワーク アクセス サーバー「maple」上のポート tty2 で、ユーザー「pat」にリバース Telnet アクセス権を付与する RADIUS サーバーに要求を送信する例を示します。

```

Username = "pat"
Password = "goaway"
User-Service-Type = Shell-User
cisco-avpair = "raccess:port#1=maple/tty2"

```

構文「`raccess:port=any/any`」で、リバース Telnet のネットワーク アクセス サーバー ポートに対して無条件のアクセス権をユーザーに許可します。「`raccess:port={nasname }/{tty number }`」句がユーザー プロファイルにない場合、ユーザーはすべてのポートでリバース Telnet へのアクセスを拒否されます。

RADIUS の設定の詳細については、「RADIUS の設定」の章を参照してください。

認可の設定に関する追加情報

シスコのテクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカルサポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

許可設定の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	AAA 許可	AAA 認可を使用すると、ユーザーが利用できるサービスを制限できます。AAA 認可をイネーブルにすると、ネットワーク アクセス サーバーはユーザーのプロファイルから取得した情報を使用して、ユーザーの設定を設定します。このプロファイルは、ローカルユーザー データベースまたはセキュリティ サーバーにあります。認可が完了すると、ユーザー プロファイルの情報で許可されているサービスであれば、ユーザーは要求したサービスに対するアクセス権を付与されます。

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



第 4 章

アカウントティングの設定

AAA アカウントティング機能を使用すると、ユーザーがアクセスするサービス、およびユーザーが消費するネットワーク リソース量を追跡できます。AAA アカウントティングをイネーブルにすると、ネットワーク アクセス サーバーから TACACS+ または RADIUS セキュリティ サーバー（実装しているセキュリティ手法によって異なります）に対して、アカウントティング レコードの形式でユーザー アクティビティがレポートされます。各アカウントティング レコードにはアカウントティングの Attribute-Value (AV) ペアが含まれ、レコードはセキュリティ サーバーに格納されます。このデータを分析して、ネットワーク管理、クライアント課金、および監査に利用できます。

- [アカウントティングを設定するための前提条件](#) (97 ページ)
- [アカウントティングの設定の制約事項](#) (98 ページ)
- [アカウントティングの設定に関する情報](#) (98 ページ)
- [AAA アカウントティングの設定方法](#) (113 ページ)
- [AAA アカウントティングの設定例](#) (122 ページ)
- [アカウントティングの設定に関するその他の参考資料](#) (126 ページ)
- [アカウントティングの設定の機能履歴](#) (127 ページ)

アカウントティングを設定するための前提条件

次のタスクを実行してから、名前付き方式リストを使用してアカウントティングを設定します。

- ネットワークアクセスサーバで AAA を有効にするには、グローバル コンフィギュレーション モードで **aaa new-model** コマンドを使用します。
- RADIUS または TACACS+ 認可が発行されている場合、RADIUS または TACACS+ セキュリティ サーバの特性を定義します。Cisco ネットワーク アクセス サーバを設定して RADIUS セキュリティ サーバと通信する方法の詳細については、「RADIUS の設定」モジュールを参照してください。Cisco ネットワーク アクセス サーバを設定して TACACS+ セキュリティ サーバと通信する方法の詳細については、「TACACS+ の設定」モジュールを参照してください。

アカウントティングの設定の制約事項

- アカウントティング情報は、最大 4 台の AAA サーバにのみ同時送信できます。

アカウントティングの設定に関する情報

アカウントティングの名前付き方式リスト

認証および認可方式リストと同様に、アカウントティングの方式リストには、アカウントティングの実行方法とその方式を実行するシーケンスが定義されています。

アカウントティングの名前付き方式リストには、特定のセキュリティプロトコルを指定し、アカウントティングサービスの特定の行またはインターフェイスに使用できます。唯一の例外は、デフォルトの方式リスト（「default」という名前）です。デフォルトの方式リストは、明示的に定義された名前付きの方式リストを持つインターフェイスを除くすべてのインターフェイスに自動的に適用されます。定義済みの方式リストは、デフォルトの方式リストに優先します。

方式リストは、シーケンスで照会されるアカウントティング方式（RADIUS、TACACS+ など）を説明する単なる名前付きリストです。方式リストでは、アカウントティングに1つまたは複数のセキュリティプロトコルを指定できます。そのため、最初の方式が失敗した場合に備えてアカウントティングのバックアップシステムを確保できます。Cisco IOS ソフトウェアでは、リストされている最初の方式を使用して、アカウントティングをサポートします。その方式が応答しない場合、リストされている次のアカウントティング方式が選択されます。このプロセスは、リストのいずれかのアカウントティング方式と通信に成功するか、定義されているすべての方式が試行されるまで継続されます。



- (注) Cisco IOS ソフトウェアでは、前の方式で応答が得られない場合にのみ、リストされている次のアカウントティング方式でアカウントティングが試行されます。このサイクルの任意の時点でアカウントティングが失敗した場合（つまり、セキュリティサーバーからユーザーアクセスの拒否応答が返される場合）、アカウントティングプロセスは停止し、その他のアカウントティング方式は試行されません。

アカウントティング方式リストは、要求されるアカウントティングの種類によって変わります。AAA は、次の 7 種類のアカウントティングをサポートしています。

- **Network** : パケットやバイトカウントなど、すべての PPP、SLIP、または ARAP セッションに関する情報を提供します。
- **EXEC** : ネットワークアクセスサーバのユーザ EXEC ターミナルセッションに関する情報を提供します。

- **Commands** : ユーザが発行する EXEC モードコマンドに関する情報を提供します。コマンドアカウントティングは、特定の特権レベルに関連付けられた、グローバルコンフィギュレーションコマンドなどのすべての EXEC モードコマンドについて、アカウントティングレコードを生成します。
- **Connection** : Telnet、ローカルエリアトランスポート (LAT)、TN3270、パケットアセンブラ/ディスアセンブラ (PAD)、rlogin などのネットワークアクセスサーバから行われたすべてのアウトバンド接続に関する情報を提供します。
- **System** : システムレベルのイベントに関する情報を提供します。
- **Resource** : ユーザ認証に成功したコールの「開始」および「終了」レコードを提供します。また、認証に失敗したコールの「終了」レコードを提供します。
- **VRRS** : Virtual Router Redundancy Service (VRRS) に関する情報を提供します。



(注) システムアカウントティングは、名前付きアカウントティングリストを使用しません。システムアカウントティングのデフォルトリストだけを定義できます。

方式指定リストが作成されると、指定したアカウントティングタイプのアカウントティング方式のリストが定義されます。

アカウントティング方式リストを特定の回線またはインターフェイスに適用してから、定義済み方式のいずれかを実行する必要があります。唯一の例外は、デフォルトの方式リスト（「default」という名前）です。名前付き方式リストを指定せずに、特定のアカウントティングタイプに対して **aaa accounting** コマンドを発行すると、明示的に名前付き方式リストが定義されている場合を除き、すべてのインターフェイスまたは回線にデフォルトの方式リストが自動的に適用されます（定義した方式リストは、デフォルトの方式リストよりも優先されます）。デフォルトの方式リストが定義されていない場合、アカウントティングは実行されません。

ここでは、次の内容について説明します。

方式リストとサーバグループ

サーバグループは、方式リストに使用する既存の RADIUS または TACACS+ サーバホストをグループ化する方法の 1 つです。次の図に、4 台のセキュリティサーバ (R1 と R2 は RADIUS サーバ、T1 と T2 は TACACS+ サーバ) が設置された一般的な AAA ネットワーク設定を示します。R1 と R2 は RADIUS サーバのグループから構成されます。T1 と T2 は TACACS+ サーバのグループから構成されます。

Cisco IOS ソフトウェアでは、RADIUS および TACACS+ サーバ設定はグローバルです。サーバグループを使用して、設定済みのサーバホストのサブセットを指定できます。このようなサーバグループは、特定のサービスに使用できます。たとえば、サーバグループを使用すると、R1 と R2 を個別のサーバグループ (SG1 と SG2) として定義し、T1 と T2 を個別のサーバグループ (SG3 と SG4) として定義できます。つまり、R1 と T1 (SG1 と SG3) または R2 と T2 (SG2 と SG4) を方式リストに指定することができます。そのため、RADIUS および TACACS+ のリソースを割り当てる場合の柔軟性が高くなります。

サーバグループには、1台のサーバに対して複数のホストエントリを含めることができます。エントリごとに固有の識別情報を設定します。固有の識別情報は、IPアドレスとUDPポート番号の組み合わせで構成されます。これにより、RADIUSホストとして定義されているさまざまなポートが、固有のAAAサービスを提供できるようになります。つまり、この固有識別情報を使用して、1台のサーバ上に複数のUDPポートが存在する場合、同じIPアドレスからそれぞれのUDPポートに対してRADIUS要求を送信できます。1台のRADIUSサーバ上にある異なる2つのホストエントリが1つのサービス（アカウントングなど）に設定されている場合、設定されている2番目のホストエントリは最初のホストエントリのフェールオーバーバックアップとして動作します。この例を使用して、最初のホストエントリがアカウントングサービスの提供に失敗した場合、ネットワークアクセスサーバは、同じデバイスに設定されている2番目のホストエントリに対してアカウントングサービスを試行します（RADIUSホストエントリは、設定順に試行されます）。

サーバグループの設定および着信番号識別サービス（DNIS）番号に基づくサーバグループの設定の詳細については、「RADIUSの設定」または「TACACS+の設定」を参照してください。

AAA アカウントング方式

次の2つのアカウントング方式がサポートされます。

- TACACS+：ネットワークアクセスサーバは、アカウントングレコードの形式でTACACS+セキュリティサーバに対してユーザアクティビティを報告します。各アカウントングレコードは、アカウントングAVペアが含まれ、セキュリティサーバ上で保管されます。
- RADIUS：ネットワークアクセスサーバは、アカウントングレコードの形式でRADIUSセキュリティサーバに対してユーザアクティビティを報告します。各アカウントングレコードは、アカウントングAVペアが含まれ、セキュリティサーバ上で保管されます。



(注) パスワードおよびアカウントングログは、TACACS+またはRADIUSセキュリティサーバへ送信される前にマスクされます。マスクされていない情報をTACACS+またはRADIUSセキュリティサーバに送信するには、**aaa accounting commands visible-keys** コマンドを使用します。

アカウントングレコードの種類

最小限のアカウントングの場合、**stop-only** キーワードを使用します。このキーワードによって、要求されたユーザプロセスの終了時に、終了レコードアカウントング通知を送信するように、指定した方式（RADIUSまたはTACACS+）に指示します。詳細なアカウントング情報が必要な場合、**start-stop** キーワードを使用して、要求されたイベントの開始時には開始アカウントング通知、そのイベントの終了時には修理用アカウントング通知を送信します。この回線またはインターフェイスですべてのアカウントングアクティビティを終了するには、**none** キーワードを使用します。

アカウントング方式

次の表に、サポートされるアカウントング方式を示します。

表 4: AAA アカウントティング方式

キーワード	Description
group radius	アカウントティングにすべての RADIUS サーバーのリストを使用します。
group tacacs+	アカウントティングにすべての TACACS+ サーバーのリストを使用します。
group <i>group-name</i>	<i>group-name</i> サーバーグループで定義したように、アカウントティングのための RADIUS サーバーまたは TACACS+ サーバーのサブセットを使用します。

method 引数は、認証アルゴリズムが試行する実際の方式を指します。追加の認証方式は、直前の方式で（失敗した場合ではなく）エラーが返された場合にのみ使用されます。他のすべての方式がエラーを返しても、認証に成功したことを指定するには、コマンドで追加の方式を指定します。たとえば、TACACS+ 認証がエラーを返す場合に認証のバックアップ方式として RADIUS を指定する `acct_tac1` という方式リストを作成するには、次のコマンドを入力します。

```
aaa accounting network acct_tac1 stop-only group tacacs+ group radius
```

名前付きリストが `aaa accounting` コマンドに指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルト認証方式リストは、自動的にすべてのインターフェイスに適用されます。

たとえば、ログイン時のユーザー認証のデフォルト方式として RADIUS を指定するには、次のコマンドを入力します。

```
aaa accounting network default stop-only group radius
```

AAA アカウントティングは、次の方式をサポートします。

- **group tacacs** : ネットワークアクセスサーバーからアカウントティング情報を TACACS+ セキュリティサーバーに送信するには、**group tacacs+ method** キーワードを使用します。
- **group radius** : ネットワークアクセスサーバーからアカウントティング情報を RADIUS セキュリティサーバーに送信するには、**group radius method** キーワードを使用します。



(注) SLIP のアカウントティング方式リストは、関連インターフェイスで PPP に設定されているすべての方式に従います。特定のインターフェイスに定義および適用されるリストがない場合（または PPP 設定が指定されていない場合）、アカウントティングのデフォルト設定が適用されます。

- **group group-name** : RADIUS または TACACS+ サーバーのサブセットを指定して、アカウントティング方式として使用するには、**group group-name** 方式を指定して **aaa accounting** コマンドを使用します。グループ名とそのグループのメンバを指定して定義するには、**aaa group server** コマンドを使用します。たとえば、**aaa group server** コマンドを使用して、**group loginrad** のメンバを最初に定義します。

```
aaa group server radius loginrad
server 172.16.2.3
server 172.16.2.17
server 172.16.2.32
```

このコマンドにより、172.16.2.3、172.16.2.17、172.16.2.32 の RADIUS サーバーが **group loginrad** のメンバとして指定されます。

他の方式リストが定義されていない場合、ネットワークアカウントティングの方式として **group loginrad** を指定するには、次のコマンドを入力します。

```
aaa accounting network default start-stop group loginrad
```

アカウントティング方式としてグループ名を使用するには、事前に RADIUS または TACACS+ セキュリティ サーバーとの通信をイネーブルにする必要があります。

AAA アカウンティング タイプ

ネットワーク アカウンティング

ネットワーク アカウンティングは、パケットやバイト カウントなど、すべての PPP、SLIP、または ARAP セッションに関する情報を提供します。

次に、EXEC セッションを介して着信する PPP ユーザの RADIUS ネットワーク アカウンティング レコードに含まれる情報の例を示します。

```
Wed Jun 27 04:44:45 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 5
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Start
Acct-Authentic = RADIUS
Service-Type = Exec-User
Acct-Session-Id = "0000000D"
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"
```

```
Wed Jun 27 04:45:00 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 5
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Start
Acct-Authentic = RADIUS
```

```

Service-Type = Framed
Acct-Session-Id = "0000000E"
Framed-IP-Address = "10.1.1.2"
Framed-Protocol = PPP
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identififier = "172.16.25.15"
Wed Jun 27 04:47:46 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 5
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Framed
Acct-Session-Id = "0000000E"
Framed-IP-Address = "10.1.1.2"
Framed-Protocol = PPP
Acct-Input-Octets = 3075
Acct-Output-Octets = 167
Acct-Input-Packets = 39
Acct-Output-Packets = 9
Acct-Session-Time = 171
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identififier = "172.16.25.15"
Wed Jun 27 04:48:45 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 5
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "408"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Exec-User
Acct-Session-Id = "0000000D"
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identififier = "172.16.25.15"

```

次に、最初にEXECセッションを開始したPPPユーザのTACACS+ネットワークアカウントティングレコードに含まれる情報の例を示します。

```

Wed Jun 27 04:00:35 2001 172.16.25.15 username1 tty4 562/4327528
starttask_id=28 service=shell
Wed Jun 27 04:00:46 2001 172.16.25.15 username1 tty4 562/4327528
starttask_id=30 addr=10.1.1.1 service=ppp
Wed Jun 27 04:00:49 2001 172.16.25.15 username1 tty4 408/4327528
updattask_id=30 addr=10.1.1.1 service=ppp protocol=ip addr=10.1.1.1
Wed Jun 27 04:01:31 2001 172.16.25.15 username1 tty4 562/4327528 stoptask_id=30
addr=10.1.1.1 service=ppp protocol=ip addr=10.1.1.1
bytes_in=2844 bytes_out=1682 paks_in=36
paks_out=24 elapsed_time=51
Wed Jun 27 04:01:32 2001 172.16.25.15 username1 tty4 562/4327528 stoptask_id=28
service=shell elapsed_time=57

```



(注) アカウンティングパケットレコードの正確なフォーマットは、セキュリティサーバデーモンに応じて変わります。

次に、`autoselect` を介して着信する PPP ユーザの RADIUS ネットワーク アカウンティング レコードに含まれる情報の例を示します。

```
Wed Jun 27 04:30:52 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 3
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Start
Acct-Authentic = RADIUS
Service-Type = Framed
Acct-Session-Id = "0000000B"
Framed-Protocol = PPP
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"
```

```
Wed Jun 27 04:36:49 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 3
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Framed
Acct-Session-Id = "0000000B"
Framed-Protocol = PPP
Framed-IP-Address = "10.1.1.1"
Acct-Input-Octets = 8630
Acct-Output-Octets = 5722
Acct-Input-Packets = 94
Acct-Output-Packets = 64
Acct-Session-Time = 357
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"
```

次に、`autoselect` を介して着信する PPP ユーザの TACACS+ ネットワーク アカウンティング レコードに含まれる情報の例を示します。

```
Wed Jun 27 04:02:19 2001 172.16.25.15 username1 Async5 562/4327528
starttask_id=35 service=ppp
Wed Jun 27 04:02:25 2001 172.16.25.15 username1 Async5 562/4327528
updatetask_id=35 service=ppp protocol=ip addr=10.1.1.2
Wed Jun 27 04:05:03 2001 172.16.25.15 username1 Async5 562/4327528 stoptask_id=35
service=ppp protocol=ip addr=10.1.1.2
bytes_in=3366 bytes_out=2149 paks_in=42
paks_out=28 elapsed_time=164
```

EXEC アカウンティング

EXEC アカウンティングは、ネットワーク アクセス サーバ上にあるユーザ EXEC ターミナル セッション（ユーザシェル）に関する情報を提供します。たとえば、ユーザ名、日付、開始時刻と終了時刻、アクセスサーバの IP アドレス、および（ダイヤルインユーザの場合）発信元の電話番号などです。

次に、ダイヤルインユーザの RADIUS EXEC アカウンティング レコードに含まれる情報の例を示します。

```

Wed Jun 27 04:26:23 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 1
  User-Name = "username1"
  Client-Port-DNIS = "4327528"
  Caller-ID = "5622329483"
  Acct-Status-Type = Start
  Acct-Authentic = RADIUS
  Service-Type = Exec-User
  Acct-Session-Id = "00000006"
  Acct-Delay-Time = 0
  User-Id = "username1"
  NAS-Identifier = "172.16.25.15"
Wed Jun 27 04:27:25 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 1
  User-Name = "username1"
  Client-Port-DNIS = "4327528"
  Caller-ID = "5622329483"
  Acct-Status-Type = Stop
  Acct-Authentic = RADIUS
  Service-Type = Exec-User
  Acct-Session-Id = "00000006"
  Acct-Session-Time = 62
  Acct-Delay-Time = 0
  User-Id = "username1"
  NAS-Identifier = "172.16.25.15"

```

次に、ダイヤルインユーザの TACACS+ EXEC アカウンティング レコードに含まれる情報の例を示します。

```

Wed Jun 27 03:46:21 2001      172.16.25.15      username1      tty3      5622329430/4327528
  start
  task_id=2      service=shell
Wed Jun 27 04:08:55 2001      172.16.25.15      username1      tty3      5622329430/4327528
  stop
  task_id=2      service=shell      elapsed_time=1354

```

次に、Telnet ユーザの RADIUS EXEC アカウンティング レコードに含まれる情報の例を示します。

```

Wed Jun 27 04:48:32 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 26
  User-Name = "username1"
  Caller-ID = "10.68.202.158"
  Acct-Status-Type = Start
  Acct-Authentic = RADIUS
  Service-Type = Exec-User
  Acct-Session-Id = "00000010"
  Acct-Delay-Time = 0
  User-Id = "username1"
  NAS-Identifier = "172.16.25.15"
Wed Jun 27 04:48:46 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 26
  User-Name = "username1"

```

```

Caller-ID = "10.68.202.158"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Exec-User
Acct-Session-Id = "00000010"
Acct-Session-Time = 14
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

```

次に、Telnet ユーザの TACACS+ EXEC アカウントング レコードに含まれる情報の例を示します。

```

Wed Jun 27 04:06:53 2001      172.16.25.15      username1      tty26      10.68.202.158
starttask_id=41      service=shell
Wed Jun 27 04:07:02 2001      172.16.25.15      username1      tty26      10.68.202.158
stoptask_id=41      service=shell      elapsed_time=9

```

コマンドアカウントング

コマンドアカウントングは、ネットワーク アクセス サーバで実行される各特権レベルの EXEC シェル コマンドに関する情報を提供します。各コマンドアカウントング レコードには、その特権レベルで実行されるコマンド、各コマンドが実行された日時、および実行したユーザのリストが含まれます。

次に、特権レベル 1 の TACACS+ コマンドアカウントング レコードに含まれる情報の例を示します。

```

Wed Jun 27 03:46:47 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=3      service=shell      priv-lvl=1      cmd=show version <cr>
Wed Jun 27 03:46:58 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=4      service=shell      priv-lvl=1      cmd=show interfaces Ethernet
0 <cr>
Wed Jun 27 03:47:03 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=5      service=shell      priv-lvl=1      cmd=show ip route <cr>

```

次に、特権レベル 15 の TACACS+ コマンドアカウントング レコードに含まれる情報の例を示します。

```

Wed Jun 27 03:47:17 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=6      service=shell      priv-lvl=15      cmd=configure terminal <cr>
Wed Jun 27 03:47:21 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=7      service=shell      priv-lvl=15      cmd=interface Serial 0 <cr>
Wed Jun 27 03:47:29 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=8      service=shell      priv-lvl=15      cmd=ip address 10.1.1.1
255.255.255.0 <cr>

```



(注) Cisco の RADIUS 実装は、コマンドアカウントングをサポートしていません。

接続アカウントング

接続アカウントングは、Telnet、LAT、TN3270、PAD、rlogin などのネットワーク アクセス サーバから行われるすべての発信接続に関する情報を提供します。

次に、発信 Telnet 接続の RADIUS 接続アカウントティング レコードに含まれる情報の例を示します。

```
Wed Jun 27 04:28:00 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 2
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "5622329477"
Acct-Status-Type = Start
Acct-Authentic = RADIUS
Service-Type = Login
Acct-Session-Id = "00000008"
Login-Service = Telnet
Login-IP-Host = "10.68.202.158"
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identififier = "172.16.25.15"
```

```
Wed Jun 27 04:28:39 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 2
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "5622329477"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Login
Acct-Session-Id = "00000008"
Login-Service = Telnet
Login-IP-Host = "10.68.202.158"
Acct-Input-Octets = 10774
Acct-Output-Octets = 112
Acct-Input-Packets = 91
Acct-Output-Packets = 99
Acct-Session-Time = 39
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identififier = "172.16.25.15"
```

次に、発信 Telnet 接続の TACACS+ 接続アカウントティング レコードに含まれる情報の例を示します。

```
Wed Jun 27 03:47:43 2001          172.16.25.15      username1  tty3      5622329430/4327528
start      task_id=10      service=connection      protocol=telnet  addr=10.68.202.158
cmd=telnet username1-sun
Wed Jun 27 03:48:38 2001          172.16.25.15      username1  tty3      5622329430/4327528
stop      task_id=10      service=connection      protocol=telnet  addr=10.68.202.158
cmd=telnet username1-sun      bytes_in=4467  bytes_out=96      paks_in=61      paks_out=72
elapsed_time=55
```

次に、発信 rlogin 接続の RADIUS 接続アカウントティング レコードに含まれる情報の例を示します。

```
Wed Jun 27 04:29:48 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 2
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "5622329477"
Acct-Status-Type = Start
```

```

Acct-Authentic = RADIUS
Service-Type = Login
Acct-Session-Id = "0000000A"
Login-Service = Rlogin
Login-IP-Host = "10.68.202.158"
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

Wed Jun 27 04:30:09 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 2
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "5622329477"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Login
Acct-Session-Id = "0000000A"
Login-Service = Rlogin
Login-IP-Host = "10.68.202.158"
Acct-Input-Octets = 18686
Acct-Output-Octets = 86
Acct-Input-Packets = 90
Acct-Output-Packets = 68
Acct-Session-Time = 22
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

```

次に、発信 rlogin 接続の TACACS+ 接続アカウンティング レコードに含まれる情報の例を示します。

```

Wed Jun 27 03:48:46 2001      172.16.25.15      username1  tty3      5622329430/4327528
start  task_id=12      service=connection  protocol=rlogin  addr=10.68.202.158
cmd=rlogin  username1-sun /user  username1
Wed Jun 27 03:51:37 2001      172.16.25.15      username1  tty3      5622329430/4327528
stop   task_id=12      service=connection  protocol=rlogin  addr=10.68.202.158
cmd=rlogin  username1-sun /user  username1  bytes_in=659926  bytes_out=138   paks_in=2378
paks_
out=1251      elapsed_time=171

```

次に、発信 LAT 接続の TACACS+ 接続アカウンティング レコードに含まれる情報の例を示します。

```

Wed Jun 27 03:53:06 2001      172.16.25.15      username1  tty3      5622329430/4327528
start  task_id=18      service=connection  protocol=lat     addr=VAX        cmd=lat
VAX
Wed Jun 27 03:54:15 2001      172.16.25.15      username1  tty3      5622329430/4327528
stop   task_id=18      service=connection  protocol=lat     addr=VAX        cmd=lat
VAX  bytes_in=0      bytes_out=0      paks_in=0      paks_out=0      elapsed_time=6

```

システム アカウンティング

システム アカウンティングは、すべてのシステムレベル イベント（たとえば、システムのリブート時やアカウンティングのオン/オフ時）に関する情報を提供します。

次のアカウンティング レコードは、AAA アカウンティングがオフになったことを示す一般的な TACACS+ システム アカウンティング レコード サーバを示します。

```
Wed Jun 27 03:55:32 2001      172.16.25.15      unknown unknown unknown start task_id=25
service=system
event=sys_acct reason=reconfigure
```



(注) アカウントティングパケットレコードの正確なフォーマットは、TACACS+デーモンに応じて変わります。

次のアカウントティングレコードは、AAAアカウントティングがオンになったことを示すTACACS+システムアカウントティングレコードを示します。

```
Wed Jun 27 03:55:22 2001      172.16.25.15      unknown unknown unknown stop task_id=23
service=system
event=sys_acct reason=reconfigure
```

リソース アカウントティング

シスコが採用している AAA アカウントティングでは、ユーザー認証を通過したコールに対する「開始」レコードと「終了」レコードがサポートされます。ユーザー認証の一部として認証に失敗したコールの「終了」レコードを生成する追加機能もサポートされます。このようなレコードは、ネットワークを管理およびモニタするアカウントティングレコードを採用する場合に必要です。

ここでは、次の内容について説明します。

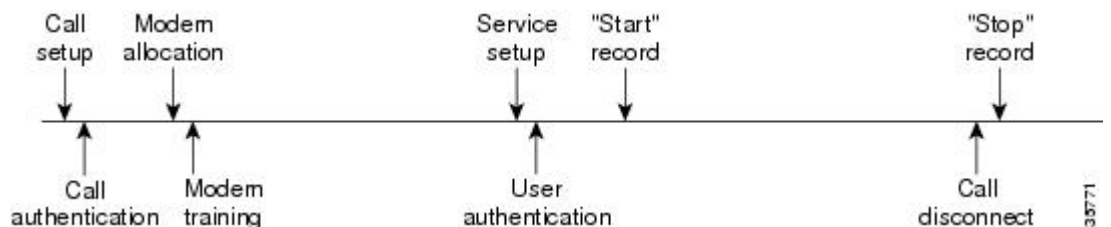
AAA リソース失敗終了アカウントティング

AAA リソース失敗終了アカウントティングの前には、コール設定シーケンスのユーザー認証段階に到達できなかったコールについて、アカウントティングレコードを提供する方式がありませんでした。このようなレコードは、ネットワークおよびその卸売りの顧客を管理およびモニタするアカウントティングレコードを採用する場合に必要です。

この機能によって、ユーザー認証に到達しなかったコールの「終了」アカウントティングレコードが生成されます。「終了」レコードは、コール設定の時点から生成されます。ユーザー認証に成功したすべてのコールは、従来と同様に動作します。つまり、追加のアカウントティングレコードは確認されません。

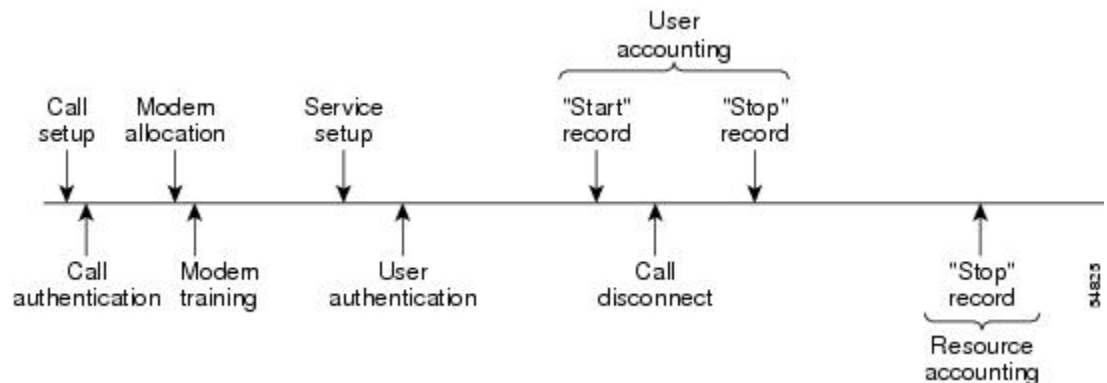
次の図に、通常のコールフローで、AAA リソース失敗終了アカウントティングを有効にしていないコールシーケンスを示します。

図4:通常のフローでAAAリソース失敗終了アカウントティングを有効にしないモデムダイヤルインコール設定シーケンス



次の図に、通常のコールフローで、AAA リソース失敗終了アカウントングを有効にしたコールシーケンスを示します。

図 5: 通常のフローで AAA リソース失敗終了アカウントングを有効にしたモデムダイヤルインコール設定シーケンス



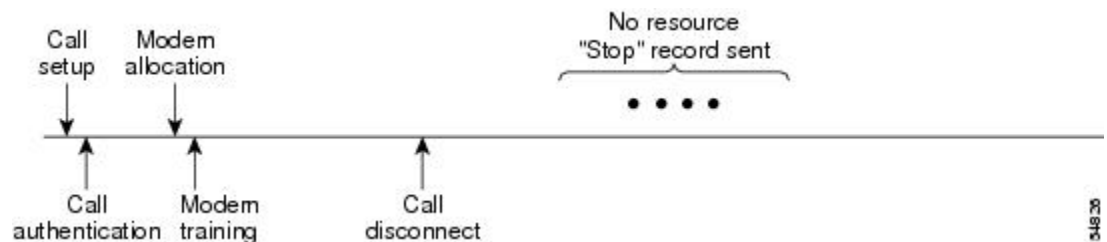
次の図に、ユーザー認証前にコールの接続解除が発生し、AAA リソース失敗終了アカウントングを有効にしたコール設定シーケンスを示します。

図 6: ユーザー認証前にコールの接続解除が発生し、AAA リソース失敗終了アカウントングを有効にしたモデムダイヤルインコール設定シーケンス



次の図に、ユーザー認証前にコールの接続解除が発生し、AAA リソース失敗終了アカウントングを有効にしていないコール設定シーケンスを示します。

図 7: ユーザー認証前にコールの接続解除が発生し、AAA リソース失敗終了アカウントングをイネーブルにしていないモデムダイヤルインコール設定シーケンス



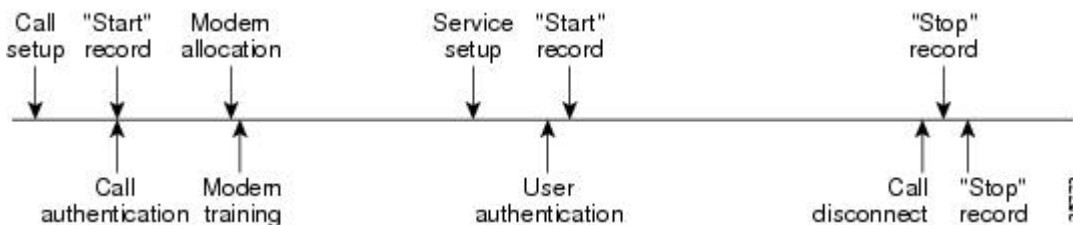
開始 - 終了レコードの AAA リソース アカウンティング

開始 - 終了レコードの AAA リソース アカウンティングは、各コール設定時に「開始」レコードを送信し、コールの接続解除時に対応する「終了」レコードを送信する機能をサポートしています。この機能は、アカウントングレコードなどを報告するデータの発信元の1つから、卸売りの顧客を管理およびモニターするために使用できます。

この機能を使用すると、コール設定およびコールの接続解除の「開始-終了」アカウントティングレコードは、デバイスに対するリソース接続の進行状況を追跡します。個別のユーザー認証「開始-終了」アカウントティングレコードが、ユーザー管理の進行状況を追跡します。これら2セットのアカウントティングレコードは、そのコールで固有のセッションIDを使用して相互リンクされます。

次の図は、AAA リソース開始-終了アカウントティングを有効にしたコール設定シーケンスを示します。

図 8: リソース開始-終了アカウントティングを有効にしたモデムダイヤルインコール設定シーケンス



AAA アカウントティングの強化

AAA ブロードキャストアカウントティング

AAA ブロードキャストアカウントティングを有効にすると、アカウントティング情報を複数のAAA サーバに同時に送信できます。つまり、アカウントティング情報を1つまたは複数のAAA サーバに同時にブロードキャストすることが可能です。この機能を使用すると、サービスプロバイダーは自社使用のプライベートAAA サーバやエンドユーザのAAA サーバにアカウントティング情報を送信できるようになります。この機能では、音声アプリケーションによる課金情報も提供されます。

ブロードキャストは、RADIUS または TACACS+ サーバのグループに使用できます。また、各サーバグループは、他のグループとは関係なく、フェールオーバーの場合のバックアップサーバを定義できます。

したがって、サービスプロバイダーとそのエンドユーザは、アカウントティングサーバに異なるプロトコル (RADIUS または TACACS+) を使用できます。また、サービスプロバイダーとそのエンドユーザは、それぞれ単独でバックアップサーバを指定することもできます。音声アプリケーションについては、独自のフェールオーバーシーケンスを持つ個別のグループを介して、冗長的なアカウントティング情報を単独で管理できます。

AAA セッション MIB

ユーザがAAAセッションMIB機能を使用すると、簡易ネットワーク管理プロトコル (SNMP) を使用して自身の認証済みクライアント接続をモニタおよび終了できます。そのクライアントのデータが提示されるため、RADIUS または TACACS+ サーバから報告されるAAAアカウントティング情報に直接関連付けることができます。AAAセッションMIBは、次の情報を提供します。

- 各 AAA 機能の統計情報 (`show radius statistics` コマンドと併用する場合)

- AAA 機能を提供するサーバのステータス
- 外部 AAA サーバの ID
- (アイドル時間などの) リアルタイム情報 (アクティブ コールを終了するかどうかを評価する SNMP ネットワークが使用する追加基準を提供します)

次の表に、認証済みクライアントと AAA セッション MIB 機能との接続をモニタおよび終了するために使用できる SNMP ユーザエンド データ オブジェクトを示します。

表 5: SNMP エンドユーザ データ オブジェクト

SessionId	AAA アカウントング プロトコルに使用されるセッション ID (RADIUS 属性 44 (Acct-Session-ID) から報告される値と同じ)
UserId	ユーザ ログイン ID または (ログインが使用できない場合) 長さがゼロの文字列
IpAddr	セッションの IP アドレスまたは (IP アドレスが適用されない場合、または使用できない場合) 0.0.0.0
IdleTime	セッションがアイドルになってからの経過時間
Disconnect	そのクライアントとの接続を解除するために使用されるセッション終了オブジェクト
CallId	コール トラッカー レコードが保存した、このアカウントング セッションに対応するエントリ インデックス

次の表に、システム別に SNMP を使用する AAA セッション MIB 機能から提供される AAA の概要情報を示します。

表 6: SNMP AAA セッションの概要

ActiveTableEntries	現在アクティブなセッションの数
ActiveTableHighWaterMark	システムが最後に再インストールされてからの同時接続セッションの最大数
TotalSessions	システムが最後に再インストールされてからのセッションの合計数
DisconnectedSessions	システムが最後に再インストールされてから接続解除されたセッションの合計数

アカウントング属性と値のペア

ネットワーク アクセス サーバは、TACACS+ AV のペアまたは RADIUS 属性 (実装しているセキュリティ方式によって異なります) に定義されたアカウントング機能をモニタします。

AAA アカウントティングの設定方法

名前付き方式リストによる AAA アカウントティングの設定

名前付き方式リストを使用して AAA アカウントティングを設定するには、次の手順を実行します。



(注) システム アカウントティングは、名前付き方式リストを使用しません。システム アカウントティングの場合、デフォルトの方式リストだけを定義します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa accounting** {system | network | exec | connection | commands level} {default | list-name} {start-stop | stop-only | none} [method1 [method2...]]
4. 次のいずれかを実行します。
 - **line** [aux | console | tty | vty] line-number [ending-line-number]
 - **interface** interface-type interface-number
5. 次のいずれかを実行します。
 - **accounting** {arap | commands level | connection | exec} {default | list-name}
 - **ppp accounting** {default | list-name}
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa accounting {system network exec connection commands level} {default list-name} {start-stop stop-only none} [method1 [method2...]]	アカウントティング方式リストを作成し、アカウントティングを有効にします。引数 list-name は、作成し

	コマンドまたはアクション	目的
	<p>例 :</p> <pre>Device(config)# aaa accounting system default start-stop</pre>	<p>たリストに名前を付けるときに使用される文字列です。</p>
ステップ 4	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> • line [aux console tty vty] <i>line-number</i> [<i>ending-line-number</i>] • interface <i>interface-type interface-number</i> <p>例 :</p> <pre>Device(config)# line aux line1</pre>	<p>アカウントング方式リストを適用する回線について、ライン コンフィギュレーション モードを開始します。</p> <p>または</p> <p>アカウントング方式リストを適用するインターフェイスについて、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。</p>
ステップ 5	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> • accounting {arap commands level connection exec} {default <i>list-name</i>} • ppp accounting{default <i>list-name</i>} <p>例 :</p> <pre>Device(config-line)# accounting arap default</pre>	<p>1つの回線または複数回線にアカウントング方式リストを適用します。</p> <p>または</p> <p>1つのインターフェイスまたは複数インターフェイスにアカウントング方式リストを適用します。</p>
ステップ 6	<p>end</p> <p>例 :</p> <pre>Device(config-line)# end</pre>	<p>(任意) ライン コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

ヌルユーザ名セッション時のアカウントングレコード生成の抑制

AAA アカウントングをアクティブにすると、Cisco IOS ソフトウェアは、システム上のすべてのユーザにアカウントングレコードを発行します。このとき、プロトコル変換のためユーザ名文字列がヌルになっているユーザも含まれます。この例では、**aaa authentication login method-list none** コマンドが適用される回線で着信するユーザがそれに該当します。関連付けられているユーザ名がないセッションについて、アカウントングレコードが生成されないようにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
<pre>Device(config)# aaa accounting suppress null-username</pre>	ユーザ名文字列がヌルのユーザについて、アカウントティングレコードが生成されないようにします。

中間アカウントティング レコードの生成

アカウントティング サーバに定期的な中間アカウントティング レコードを送信できるようにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
<pre>Device(config)# aaa accounting update [newinfo] [periodic] <i>number</i></pre>	アカウントティング サーバに送信される定期的な中間アカウントティングレコードをイネーブルにします。

aaa accounting update コマンドをアクティブにすると、Cisco IOS ソフトウェアによってシステム上のすべてのユーザーの中間アカウントティングレコードが発行されます。 **newinfo** キーワードを使用した場合は、レポートする新しいアカウントティング情報が発生するたびに、中間アカウントティングレコードがアカウントティングサーバに送信されます。たとえば、IPCP がリモートピアとの間で IP アドレスのネゴシエーションを完了したときなどです。中間アカウントティングレコードには、リモートピアに使用されるネゴシエート済み IP アドレスが含まれます。

キーワード **periodic** と一緒に使用した場合は、*number* 引数による定義に基づいて、中間アカウントングレコードが定期的送信されます。中間アカウントングレコードには、中間アカウントングレコードが送信される時間までに、そのユーザについて記録されたすべてのアカウントング情報が含まれます。



注意 多数のユーザがネットワークにログインしている場合には、**aaa accounting update periodic** コマンドを使用すると、重度の輻輳が発生する可能性があります。

定期的アカウントングレコードを有効化する代替手段の設定

次の代替手段を使用して、アカウントングサーバーに送信される定期的中間アカウントングレコードをイネーブルにできます。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa accounting network default**
4. **action-type {none | start-stop [periodic {disable | interval *minutes*}] | stop-only}**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa accounting network default 例： Device(config)# aaa accounting network default	すべてのネットワーク関連のサービス要求のデフォルトのアカウントングを設定し、アカウントング方式リスト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 4	action-type {none start-stop [periodic {disable interval <i>minutes</i>}] stop-only} 例： Device(cfg-acct-mlist)# action-type start-stop	アカウントングレコードに対して実行されるアクションのタイプを指定します。 • (任意) periodic キーワードは、定期的なアカウントングアクションを示します。

	コマンドまたはアクション	目的
	例： <pre>periodic interval 5</pre>	<ul style="list-style-type: none"> • interval キーワードは、定期的なアカウントティング間隔を指定します。 • value 引数は、アカウントティング更新レコードの間隔を指定します（分単位）。 • disable キーワードは、定期的なアカウントティングを無効にします。
ステップ 5	end 例： <pre>Device(cfg-acct-mlist)# end</pre>	アカウンティング方式リストコンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

中間サービス アカウンティング レコードの生成

このタスクを実行して、サブスクライバに対する定期的な間隔での中間サービスアカウントティングレコードの生成をイネーブルにします。

始める前に

ユーザー サービス プロファイルの RADIUS 属性 85 は設定済みの中間の間隔値よりも常に優先されます。RADIUS 属性 85 は、ユーザー サービス プロファイル内にある必要があります。詳細については、RADIUS 属性の概要および RADIUS IETF 属性の機能のドキュメントを参照してください。



- (注) RADIUS 属性 85 がユーザー サービス プロファイル内にない場合、中間アカウントティングレコードの生成で設定された中間の間隔値がサービスの中間アカウントティングレコードに使用されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **subscriber service accounting interim-interval *minutes***
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	Device> enable	
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	subscriber service accounting interim-interval minutes 例： Device(config)# subscriber service accounting interim-interval 10	サブスクライバに対する定期的な間隔での中間サービス アカウンティング レコードの生成をイネーブにします。 <i>minutes</i> 引数は、アカウントング更新レコードを送信する定期的な間隔を 1 ~ 71582 分で示します。
ステップ 4	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

失敗したログインまたはセッションに対するアカウントングレコードの生成

AAA アカウンティングをアクティブにすると、Cisco IOS XE ソフトウェアは、ログイン認証に失敗したシステム ユーザー、またはログイン認証には成功しても何らかの理由で PPP ネゴシエーションに失敗したユーザーのアカウントングレコードを生成しません。

ログイン時またはセッションネゴシエーション中の認証に失敗したユーザーについて、アカウントング終了レコードを生成するように指定するには、グローバルコンフィギュレーションモードで次のコマンドを使用します。

コマンドまたはアクション	目的
aaa accounting send stop-record authentication failure	ログイン時またはセッションネゴシエーション中の認証に失敗したユーザについて、「終了」レコードを生成します。

EXEC-Stop レコードよりも前のアカウントング NETWORK-Stop レコードの指定

EXEC 終了セッションを開始する PPP ユーザーの場合、EXEC-stop レコードの前に、NETWORK レコードを生成するように指定できます。特定のサービスについて顧客に課金する場合など、状況によっては、ネットワークの開始レコードと終了レコードを一緒に保持する方が望ましいことがあります。その際、基本的に、EXEC の開始メッセージと終了メッセージのフレーム

ワーク内に「ネスト」にします。たとえば、PPPを使用するユーザーダイヤルインによって、EXEC-start、NETWORK-start、EXEC-stop、NETWORK-stop というレコードを作成できます。ネットワーク アカウンティング レコードをネストにすることで、NETWORK-stop レコードは NETWORK-start メッセージ (EXEC-start、NETWORK-start、NETWORK-stop、EXEC-stop) に従います。

ユーザーセッションのアカウントティングレコードをネストするには、グローバル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。

コマンドまたはアクション	目的
aaa accounting nested	ネットワーク アカウンティングレコードをネストします。

スイッチオーバー上のシステム アカウンティング レコードの抑制

スイッチオーバー中のシステム アカウンティングオンおよびアカウントティングオフ メッセージを抑制するには、グローバル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。

コマンドまたはアクション	目的
aaa accounting redundancy suppress system-records	スイッチオーバー中のシステムアカウントティングレコードを抑制します。

AAA リソース失敗終了アカウントティングの設定

リソース失敗終了アカウントティングを有効にするには、グローバルコンフィギュレーションモードで次のコマンドを使用します。

コマンド	目的
<pre>Device(config)# aaa accounting resource method-list stop-failure group server-group</pre>	<p>ユーザー認証に到達しないコールについて、「終了」レコードを生成します。</p> <p>(注) この機能を設定する前に、アカウントティングを設定するための前提条件 (97 ページ) のセクションに記載されている作業を実行し、ネットワーク アクセス サーバー上で SNMP を有効にしてください。</p>

開始 - 終了レコードの AAA リソース アカウンティングの設定

開始 - 終了レコードのフルリソース アカウンティングをイネーブルにするには、グローバルコンフィギュレーションモードで次のコマンドを使用します。

コマンド	目的
<pre>Device(config)# aaa accounting resource method-list start-stop group server-group</pre>	<p>各コール設定時に「開始」レコードを送信し、コールの接続解除時に対応する「終了」レコードを送信する機能をサポートします。</p> <p>(注) この機能を設定する前に、アカウントिंगを設定するための前提条件 (97 ページ) のセクションに記載されている作業を実行し、ネットワークアクセスサーバー上でSNMPを有効にしてください。</p>

AAA ブロードキャスト アカウンティング

AAA ブロードキャスト アカウンティングを有効にすると、アカウントング情報を複数の AAA サーバに同時に送信できます。つまり、アカウントング情報を 1 つまた複数の AAA サーバに同時にブロードキャストすることが可能です。この機能を使用すると、サービスプロバイダーは自社使用のプライベート AAA サーバやエンドユーザの AAA サーバにアカウントング情報を送信できるようになります。この機能では、音声アプリケーションによる課金情報も提供されます。

ブロードキャストは、RADIUS または TACACS+ サーバのグループに使用できます。また、各サーバグループは、他のグループとは関係なく、フェールオーバーの場合のバックアップサーバを定義できます。

したがって、サービスプロバイダーとそのエンドユーザは、アカウントングサーバに異なるプロトコル (RADIUS または TACACS+) を使用できます。また、サービスプロバイダーとそのエンドユーザは、それぞれ単独でバックアップサーバを指定することもできます。音声アプリケーションについては、独自のフェールオーバーシーケンスを持つ個別のグループを介して、冗長的なアカウントング情報を単独で管理できます。

DNIS による AAA ブロードキャスト アカウンティングの設定

AAA ブロードキャスト アカウンティングを設定するには、グローバルコンフィギュレーションモードで `aaa dnis map accounting network` コマンドを使用します。

コマンド	目的
<pre>Device(config)# aaa dnis map dnis-number accounting network [start-stop stop-only none] [broadcast] <i>method1</i> [<i>method2...</i>]</pre>	<p>DNIS によるアカウントティングの設定を許可します。このコマンドは、グローバルの aaa accounting コマンドよりも優先されます。</p> <p>複数の AAA サーバに対するアカウントティングレコードの送信をイネーブルにします。各グループの最初のサーバに対し、アカウントティングレコードを同時に送信します。最初のサーバが使用できない場合はフェールオーバーが発生し、そのグループ内に定義されているバックアップサーバが使用されます。</p>

AAA サーバーが到達不能な場合のデバイスとのセッションの確立

AAA サーバーが到達不能の場合に、デバイスとの間にコンソールセッションを確立するには、グローバル 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。

コマンドまたはアクション	目的
no aaa accounting system guarantee-first	<p>aaa accounting system guarantee-first コマンドは、システムアカウントを最初のレコードとして保証します。これは、デフォルトの条件です。</p> <p>状況によっては、システムの再ロードが完了するまで（3分よりも長くかかる可能性があります）、ユーザーがコンソールまたは Telnet 接続でセッションを開始できない可能性があります。この問題を解決するには、no aaa accounting system guarantee-first コマンドを使用します。</p>

アカウントティングのモニタリング

RADIUS または TACACS+ アカウントティングの場合、特定の **show** コマンドは存在しません。ログインしているユーザーに関する情報を表示するアカウントティングレコードを取得するには、特権 EXEC モードで次のコマンドを使用します。

コマンドまたはアクション	目的
show accounting	ネットワークでアクティブなアカウント可能なイベントの表示を許可し、アカウントティングサーバでデータが損失した場合に情報を収集できます。

アカウントINGのトラブルシューティング

アカウントING情報の問題を解決するには、特権 EXEC モードで次のコマンドを使用します。

コマンドまたはアクション	目的
debug aaa accounting	説明の義務があるイベントが発生したときに、その情報を表示します。

AAA アカウントINGの設定例

例：名前付き方式リストの設定

次に、RADIUS サーバーから AAA サービスを提供するためにシスコ デバイス（AAA および RADIUS セキュリティサーバーとの通信で有効）を設定する例を示します。RADIUS サーバーが応答に失敗すると、認証情報と認可情報についてローカルデータベースへの照会が行われ、アカウントING サービスは TACACS+ サーバーによって処理されます。

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication login admins local
Device(config)# aaa authentication ppp dialins group radius local
Device(config)# aaa authorization network network1 group radius local
Device(config)# aaa accounting network network2 start-stop group radius group tacacs+
Device(config)# username root password ALongPassword
Device(config)# tacacs server server1
Device(config-server-tacacs)# address ipv4 172.31.255.0
Device(config-server-tacacs)# key goaway
Device(config-server-tacacs)# exit
Device(config)# radius server isp
Device(config-sg-radius)# key myRaDiUSpassWoRd
Device(config-sg-radius)# exit
Device(config)# interface group-async 1
Device(config-if)# group-range 1 16
Device(config-if)# encapsulation ppp
Device(config-if)# ppp authentication chap dialins
Device(config-if)# ppp authorization network1
Device(config-if)# ppp accounting network2
Device(config-if)# exit
Device(config)# line 1 16
Device(config-line)# autoselect ppp
Device(config-line)# autoselect during-login
Device(config-line)# login authentication admins
Device(config-line)# modem dialin
Device(config-line)# end
```

この RADIUS AAA 設定のサンプル行は、次のように定義されます。

- **aaa new-model** コマンドは、AAA ネットワーク セキュリティ サービスをイネーブルにします。

- **aaa authentication login admins local** コマンドは、ログイン認証に方式リスト「admins」を定義します。
- **aaa authentication ppp dialins group radius local** コマンドで、認証方式リスト「dialins」を定義します。このリストは、最初にRADIUS認証を指定して、次に（RADIUSサーバーが応答しない場合）PPPを使用してシリアル回線上でローカル認証が使用されます。
- **aaa authorization network network1 group radius local** コマンドで、「network1」というネットワーク許可方式リストを定義します。これにより、PPPを使用してシリアル回線上でRADIUS許可を使用するよう指定されます。RADIUSサーバーが応答に失敗すると、ローカルネットワークの認可が実行されます。
- **aaa accounting network network2 start-stop group radius group tacacs+** コマンドで、「network2」というネットワークアカウントティング方式リストを定義します。これにより、PPPを使用してシリアル回線上でRADIUSアカウントティングサービス（この場合、特定のイベントに対する開始レコードと終了レコード）を使用するよう指定されます。RADIUSサーバーが応答に失敗すると、アカウントティングサービスはTACACS+サーバーによって処理されます。
- **username** コマンドはユーザ名とパスワードを定義します。これらの情報は、PPPパスワード認証プロトコル（PAP）の発信元身元確認に使用されます。
- **tacacs server** コマンドはTACACS+サーバーホストの名前を定義します。
- **key** コマンドは、ネットワークアクセスサーバーとTACACS+サーバーホストの間の共有秘密テキスト文字列を定義します。
- **radius server** コマンドはRADIUSサーバーホストの名前を定義します。
- **key** コマンドは、ネットワークアクセスサーバーとRADIUSサーバーホストの間の共有秘密テキスト文字列を定義します。
- **interface group-async** コマンドは、非同期インターフェイスグループを選択して定義します。
- **group-range** コマンドは、インターフェイスグループ内のメンバー非同期インターフェイスを定義します。
- **encapsulation ppp** コマンドは、指定のインターフェイスに使用されるカプセル化方式としてPPPを設定します。
- **ppp authentication chap dialins** コマンドは、PPP認証方式としてチャレンジハンドシェイク認証プロトコル（CHAP）を選択し、指定したインターフェイスに「dialins」方式リストを適用します。
- **ppp authorization network1** コマンドによって、blue1ネットワーク許可方式リストが、指定したインターフェイスに適用されます。
- **ppp accounting network2** コマンドによって、red1ネットワークアカウントティング方式リストが、指定したインターフェイスに適用されます。

- **line** コマンドはコンフィギュレーション モードをグローバル コンフィギュレーションからライン コンフィギュレーションに切り替え、設定対象の回線を指定します。
- **autoselect ppp** コマンドは、選択した回線上で PPP セッションを自動的に開始できるように Cisco IOS XE ソフトウェアを設定します。
- **autoselect during-login** コマンドを使用すると、Return キーを押さずにユーザ名およびパスワードのプロンプトが表示されます。ユーザがログインすると、autoselect 機能（この場合は PPP）が開始します。
- **login authentication admins** コマンドは、ログイン認証に admins 方式リストを適用します。
- **modem dialin** コマンドは、選択した回線に接続されているモデムを設定し、着信コールだけを受け入れるようにします。

show accounting コマンドを使用すると、前述の設定に関する出力が次のように生成されます。

```
Active Accounted actions on tty1, User username2 Priv 1
Task ID 5, Network Accounting record, 00:00:52 Elapsed
task_id=5 service=ppp protocol=ip address=10.0.0.98
```

次の表に、前述の出力に含まれるフィールドについて説明します。

表 7: **show accounting** のフィールドの説明

フィールド	説明
Active Accounted actions on	ユーザがログインに使用する端末回線またはインターフェイス名
User	ユーザの ID。
Priv	ユーザの特権レベル。
Task ID	各アカウントングセッションの固有識別情報
Accounting Record	アカウントングセッションタイプ
Elapsed	このセッションタイプの期間 (hh:mm:ss)
attribute=value	このアカウントングセッションに関連付けられている AV ペア

例：AAA リソース アカウンティングの設定

次に、リソース失敗終了アカウントング、および開始 - 終了レコード機能のリソース アカウンティングを設定する例を示します。

```
!Enable AAA on your network access server.
Device(config)# aaa new-model
!Enable authentication at login and list the AOL string name to use for login
authentication.
Device(config)# aaa authentication login AOL group radius local
!Enable authentication for ppp and list the default method to use for PPP authentication.
Device(config)# aaa authentication ppp default group radius local
```

```
!Enable authorization for all exec sessions and list the AOL string name to use for
authorization.
Device(config)# aaa authorization exec AOL group radius if-authenticated
!Enable authorization for all network-related service requests and list the default
method
to use for all network-related authorizations.
Device(config)# aaa authorization network default group radius if-authenticated
!Enable accounting for all exec sessions and list the default method to use for all
start-stop
accounting services.
Device(config)# aaa accounting exec default start-stop group radius
!Enable accounting for all network-related service requests and list the default method
to use
for all start-stop accounting services.
Device(config)# aaa accounting network default start-stop group radius
!Enable failure stop accounting.
Device(config)# aaa accounting resource default stop-failure group radius
!Enable resource accounting for start-stop records.
Device(config)# aaa accounting resource default start-stop group radius
```

例：AAA ブロードキャスト アカウンティングの設定

次に、グローバル **aaa accounting** コマンドを使用して、ブロードキャストアカウンティングを有効にする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius isp
Device(config-sg-radius)# server 10.0.0.1
Device(config-sg-radius)# server 10.0.0.2
Device(config-sg-radius)# exit
Device(config)# aaa group server tacacs+ isp_customer
Device config-sg-tacacs+)# server 172.0.0.1
Device config-sg-tacacs+)# exit
Device(config)# aaa accounting network default start-stop broadcast group isp group
isp_customer
Device(config)# tacacs server server1
Device(config-server-tacacs)# address ipv4 172.0.0.1
Device(config-server-tacacs)# key key2
Device(config-server-tacacs)# end
```

broadcast キーワードによって、ネットワーク接続に関する「開始」および「終了」アカウンティングレコードが、グループ **isp** ではサーバー 10.0.0.1 に、グループ **isp_customer** ではサーバー 172.0.0.1 に同時送信されます。サーバ 10.0.0.1 が使用できなくなると、サーバ 10.0.0.2 へのフェールオーバーが行われます。サーバ 172.0.0.1 が使用できなくなっても、グループ **isp_customer** にはバックアップサーバが設定されていないため、フェールオーバーは行われません。

例：DNIS による AAA ブロードキャスト アカウンティングの設定

次に、グローバル **aaa dnis map accounting network** コマンドを使用して、DNIS ごとのブロードキャストアカウンティングを有効にする例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa group server radius isp
Device(config-sg-radius)# server 10.0.0.1
```

例：AAA セッション MIB

```
Device(config-sg-radius)# server 10.0.0.2
Device(config-sg-radius)# exit
Device(config)# aaa group server tacacs+ isp_customer
Device config-sg-tacacs+)# server 172.0.0.1
Device config-sg-tacacs+)# exit
Device(config)# aaa dnis map enable
Device(config)# aaa dnis map 7777 accounting network start-stop broadcast group isp group
isp_customer
Device(config)# tacacs server server1
Device(config-server-tacacs)# address ipv4 172.0.0.1
Device(config-server-tacacs)# key key_2
Device(config-server-tacacs)# end
```

broadcast キーワードによって、DNIS 番号 7777 のネットワーク接続コールに関する「開始」および「終了」アカウントングレコードが、グループ **isp** ではサーバー 10.0.0.1 に、グループ **isp_customer** ではサーバー 172.0.0.1 に同時送信されます。サーバ 10.0.0.1 が使用できなくなると、サーバ 10.0.0.2 へのフェールオーバーが行われます。サーバ 172.0.0.1 が使用できなくなっても、グループ **isp_customer** にはバックアップサーバが設定されていないため、フェールオーバーは行われません。

例：AAA セッション MIB

次に、AAA セッション MIB 機能を設定して、PPP ユーザの認証済みクライアント接続を解除する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication ppp default group radius
Device(config)# aaa authorization network default group radius
Device(config)# aaa accounting network default start-stop group radius
Device(config)# aaa session-mib disconnect
Device(config)# end
```

アカウントングの設定に関するその他の参考資料

ここでは、アカウントングの設定機能に関する関連資料について説明します。

MIB

MIB	MIB のリンク
<ul style="list-style-type: none"> • CISCO-AAA-SESSION-MIB 	選択したプラットフォーム、Cisco IOS XE ソフトウェア リリース、および機能セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

シスコのテクニカル サポート

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

アカウントティングの設定の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	AAA ブロードキャスト アカウンティング	AAA ブロードキャスト アカウンティングを有効にすると、アカウントティング情報を複数の AAA サーバーに同時に送信できます。つまり、アカウントティング情報を1つまた複数の AAA サーバーに同時にブロードキャストすることが可能です。
Cisco IOS XE Everest 16.5.1a	AAA セッション MIB	ユーザーが AAA セッション MIB 機能を使用すると、SNMP を使用して自身の認証済みクライアント接続をモニターおよび終了できます。
Cisco IOS XE Everest 16.5.1a	接続アカウントティング	接続アカウントティングは、Telnet、ローカルエリアトランスポート、TN3270、Packet Assembler/disassembler (PAD)、rlogin など、ネットワークアクセスサーバーからのアウトバウンド接続すべてに関する情報を提供します。

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	AAA 中間アカウントティング	AAA 中間アカウントティングにより、レポートする必要がある新しいアカウントティング情報が発生するたびに、または定期的に、アカウントティングサーバーに中間アカウントティングレコードを送信できます。

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



第 5 章

ローカル認証および許可の設定

- [ローカル認証および許可の設定方法](#) (129 ページ)
- [ローカル認証および許可のモニタリング](#) (131 ページ)
- [ローカル認証および許可の機能履歴](#) (131 ページ)

ローカル認証および許可の設定方法

スイッチのローカル認証および許可の設定

ローカルモードでAAAを実装するようにスイッチを設定すると、サーバーがなくても動作するようにAAAを設定できます。この場合、スイッチは認証および許可の処理を行います。この設定ではアカウントिंग機能は使用できません。



Note AAA方式を使用してHTTPアクセスに対しスイッチのセキュリティを確保するには、**ip http authentication aaa** グローバル コンフィギュレーション コマンドでスイッチを設定する必要があります。AAA認証を設定しても、AAA方式を使用したHTTPアクセスに対しスイッチのセキュリティは確保しません。

ローカルモードでAAAを実装するようにスイッチを設定して、サーバーがなくても動作するようにAAAを設定するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default local**
5. **aaa authorization exec default local**
6. **aaa authorization network default local**
7. **username name [privilege level] { password encryption-type password }**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model Example: Device(config)# aaa new-model	AAA をイネーブルにします。
ステップ 4	aaa authentication login default local Example: Device(config)# aaa authentication login default local	ローカルユーザ名データベースを使用するログイン認証を設定します。 default キーワードにより、ローカルユーザデータベース認証がすべてのポートに適用されます。
ステップ 5	aaa authorization exec default local Example: Device(config)# aaa authorization exec default local	ユーザの AAA 許可を設定し、ローカルデータベースを確認して、そのユーザに EXEC シェルの実行を許可します。
ステップ 6	aaa authorization network default local Example: Device(config)# aaa authorization network default local	ネットワーク関連のすべてのサービス要求に対してユーザ AAA 許可を設定します。
ステップ 7	username name [privilege level] { password encryption-type password} Example: Device(config)# username your_user_name privilege 1 password 7 secret567	ローカルデータベースを入力し、ユーザ名ベースの認証システムを設定します。 ユーザごとにコマンドを繰り返し入力します。 <ul style="list-style-type: none"> name には、ユーザー ID を 1 ワードで指定します。スペースと引用符は使用できません。 (任意) level には、アクセス権を得たユーザーに設定する権限レベルを指定します。指定でき

	Command or Action	Purpose
		<p>る範囲は0～15です。レベル15では特権EXECモードでのアクセスが可能です。レベル0では、ユーザEXECモードでのアクセスとなります。</p> <ul style="list-style-type: none"> • <i>encryption-type</i> には、暗号化されていないパスワードが後ろに続く場合は0を、暗号化されたパスワードが後ろに続く場合は7を指定します。 • <i>password</i> には、ユーザーがスイッチにアクセスする場合に入力する必要があるパスワードを指定します。パスワードは1～25文字で、埋め込みスペースを使用でき、username コマンドの最後のオプションとして指定します。
ステップ 8	<p>end</p> <p>Example:</p> <p>Device (config) # end</p>	<p>グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

ローカル認証および許可のモニタリング

ローカル認証および許可の設定を表示するには、**show running-config** コマンドを特権 EXEC モードで使用します。

ローカル認証および許可の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	ローカル認証および許可	ローカルモードで AAA を実装するようにデバイスを設定すると、サーバがなくても動作するように AAA を設定できます。

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



第 6 章

TACACS+ の設定

- [TACACS+ の前提条件](#) (133 ページ)
- [TACACS+ の概要](#) (134 ページ)
- [TACACS+ を設定する方法, on page 138](#)
- [TACACS+ のモニタリング](#) (147 ページ)
- [TACACS+ に関する追加情報](#) (147 ページ)
- [TACACS+ の機能の履歴](#) (147 ページ)

TACACS+ の前提条件

TACACS+によるスイッチアクセスのセットアップと設定の前提条件は、次のとおりです（示されている順序で実行する必要があります）。

1. スイッチに TACACS+ サーバー アドレスとスイッチを設定します。
2. 認証キーを設定します。
3. TACACS+ サーバでステップ 2 からキーを設定します。
4. 認証、許可、アカウントिंग (AAA) をイネーブルにする。
5. ログイン認証方式リストを作成します。
6. 端末回線にリストを適用します。
7. 認証およびアカウントिंग方式のリストを作成します。

TACACS+によるスイッチアクセスの制御の前提条件は、次のとおりです。

- スイッチ上で TACACS+ 機能を設定するには、設定済みの TACACS+ サーバーにアクセスする必要があります。また、通常 LINUX または Windows ワークステーション上で稼働する TACACS+ デーモンのデータベースで管理されている TACACS+ サービスにもアクセスする必要があります。
- スイッチで TACACS+ を使用するには、TACACS+ デーモン ソフトウェアが稼働するシステムが必要です。

- TACACS+ を使用するには、それをイネーブルにする必要があります。
- 許可は、使用するスイッチでイネーブルにする必要があります。
- ユーザは TACACS+ 許可に進む前に、まず TACACS+ 認証を正常に完了する必要があります。
- このセクションに記載されている AAA コマンドのいずれかを使用するには、まず **aaa new-model** コマンドを使用して AAA をイネーブルにする必要があります。
- 最低限、TACACS+ デーモンを維持するホスト（1つまたは複数）を特定し、TACACS+ 認証の方式リストを定義する必要があります。また、任意で TACACS+ 許可およびアカウントिंगの方式リストを定義できます。
- 方式リストは実行される認証のタイプと実行順序を定義します。このリストを特定のポートに適用してから、定義済み認証方式を実行する必要があります。唯一の例外はデフォルトの方式リスト（偶然に *default* と名前が付けられている）です。デフォルトの方式リストは、名前付き方式リストを明示的に定義されたインターフェイスを除いて、自動的にすべてのポートに適用されます。定義済みの方式リストは、デフォルトの方式リストに優先します。
- TACACS+ を使用して認証を行った場合は、TACACS+ を使用して特権 EXEC アクセスを許可します。
- 認証に TACACS+ を使用しなかった場合は、ローカル データベースを使用します。

TACACS+ の概要

TACACS+ およびスイッチ アクセス

ここでは、TACACS+ について説明します。TACACS+ は詳細なアカウントング情報を提供し、認証と許可のプロセスを柔軟に管理します。TACACS+ は、認証、許可、アカウントング（AAA）機能により拡張されており、TACACS+ をイネーブルにするには AAA コマンドを使用する必要があります。

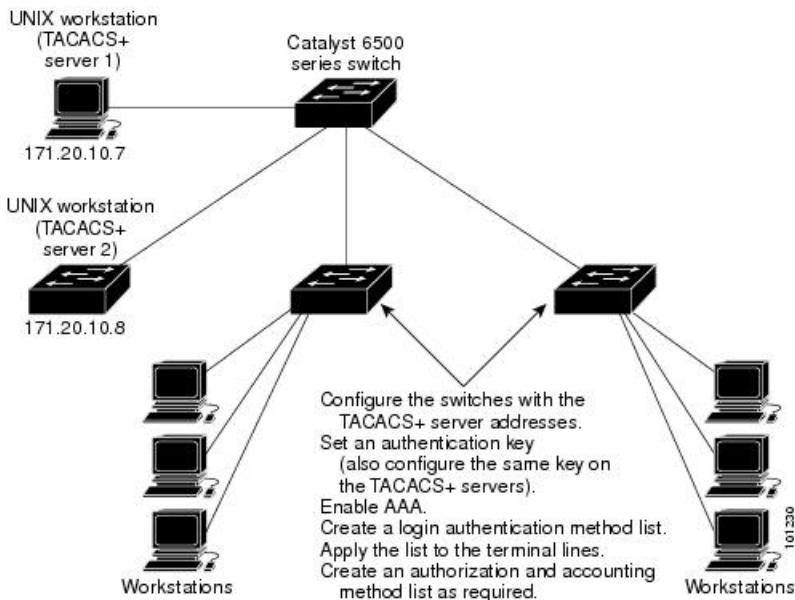
TACACS+ の概要

TACACS+ は、スイッチにアクセスしようとするユーザーの検証を集中的に行うセキュリティアプリケーションです。

TACACS+ では、独立したモジュラ型の認証、許可、アカウントング機能が提供されます。TACACS+ では、単一のアクセス コントロール サーバ（TACACS+ デーモン）が各サービス（認証、許可、およびアカウントング）を別個に提供します。各サービスを固有のデータベースに結合し、デーモンの機能に応じてそのサーバまたはネットワークで使用できる他のサービスを使用できます。

TACACS+ の目的は、1 つの管理サービスから複数のネットワーク アクセス ポイントを管理する方式を提供することです。スイッチは、他の Cisco ルータやアクセス サーバとともにネットワーク アクセス サーバにできます。

Figure 9: 一般的な TACACS+ ネットワーク構成



TACACS+ は、AAA セキュリティ サービスによって管理され、次のようなサービスを提供します。

- 認証：ログインおよびパスワードダイアログ、チャレンジおよび応答、メッセージサポートによって認証の完全制御を行います。

認証機能は、ユーザとの対話を実行できます（たとえば、ユーザ名とパスワードが入力された後、自宅の住所、母親の旧姓、サービスタイプ、社会保険番号などのいくつかの質問をすることによりユーザを試します）。TACACS+ 認証サービスは、ユーザ画面にメッセージを送信することもできます。たとえば、会社のパスワード有効期間ポリシーに従い、パスワードの変更の必要があることをユーザに通知することもできます。

- 許可：autocommand、アクセスコントロール、セッション期間、プロトコルサポートの設定といった、ユーザーセッション時のユーザー機能についてきめ細かく制御します。また、TACACS+ 許可機能によって、ユーザーが実行できるコマンドを制限することもできます。
- アカウンティング：課金、監査、およびレポートに使用する情報を収集して TACACS+ デーモンに送信します。ネットワークの管理者は、アカウンティング機能を使用して、セキュリティ監査のためにユーザの活動状況をトラッキングしたり、ユーザ課金用の情報を提供したりできます。アカウンティングレコードには、ユーザ ID、開始時刻および終了時刻、実行されたコマンド（PPP など）、パケット数、およびバイト数が含まれます。

TACACS+ プロトコルは、スイッチと TACACS+ デーモン間の認証を行い、スイッチと TACACS+ デーモン間のプロトコル交換をすべて暗号化することによって機密保持を実現します。

TACACS+ の動作

ユーザーが、TACACS+ を使用しているスイッチに対して簡易 ASCII ログインを試行し、認証が必要になると、次のプロセスが発生します。

1. 接続が確立されると、スイッチは TACACS+ デーモンに接続してユーザー名プロンプトを取得し、これをユーザーに表示します。ユーザーがユーザー名を入力すると、スイッチは TACACS+ デーモンに接続してパスワードプロンプトを取得します。スイッチによってパスワードプロンプトが表示され、ユーザーがパスワードを入力すると、そのパスワードが TACACS+ デーモンに送信されます。

TACACS+ によって、デーモンとユーザとの間の対話が可能になり、デーモンはユーザを認証できるだけの情報を取得できるようになります。デーモンは、ユーザ名とパスワードの組み合わせを入力するよう求めますが、ユーザの母親の旧姓など、その他の項目を含めることもできます。

2. スイッチは、最終的に TACACS+ デーモンから次のいずれかの応答を得ます。
 - **ACCEPT** : ユーザーが認証され、サービスを開始できます。許可を必要とするようにスイッチが設定されている場合は、この時点で許可処理が開始されます。
 - **REJECT** : ユーザーは認証されません。TACACS+ デーモンに応じて、ユーザーはアクセスを拒否されるか、ログインシーケンスを再試行するように求められます。
 - **ERROR** : デーモンによる認証サービスのある時点で、またはデーモンとスイッチの間のネットワーク接続においてエラーが発生しました。ERROR 応答が表示された場合は、スイッチは、通常別の方法でユーザーを認証しようとします。
 - **CONTINUE** : ユーザーは、さらに認証情報の入力を求められます。

認証後、スイッチで許可がイネーブになっている場合、ユーザーは追加の許可フェーズに入ります。ユーザは TACACS+ 許可に進む前に、まず TACACS+ 認証を正常に完了する必要があります。

3. TACACS+ 許可が必要な場合は、再び TACACS+ デーモンに接続し、デーモンが **ACCEPT** または **REJECT** の許可応答を返します。ACCEPT 応答が返された場合は、その応答に、そのユーザおよびそのユーザがアクセスできるサービスの、EXEC または NETWORK セッション宛ての属性の形式でデータが含まれています。
 - Telnet、セキュア シェル (SSH)、rlogin、または特権 EXEC サービス
 - 接続パラメータ (ホストまたはクライアントの IP アドレス、アクセスリスト、およびユーザタイムアウトを含む)

方式リスト

方式リストによって、ユーザの認証、許可、またはアカウント維持のための順序と方式を定義します。方式リストを使用して、使用するセキュリティプロトコルを1つまたは複数指定できるので、最初の方式が失敗した場合のバックアップシステムが確保されます。ソフトウェア

は、リスト内の最初の方式を使用してユーザの認証、許可、アカウントの維持を行います。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の方式を選択します。このプロセスは、リスト内の方式による通信が成功するか、方式リストの方式をすべて試し終わるまで続きます。

TACACS+ 設定オプション

認証用に1つのサーバーを使用することも、また、既存のサーバーホストをグループ化するために AAA サーバー グループを使用するように設定することもできます。サーバをグループ化して設定済みサーバホストのサブセットを選択し、特定のサービスにそのサーバを使用できます。サーバグループは、グローバルサーバホストリストとともに使用され、選択されたサーバホストの IP アドレスのリストが含まれています。

TACACS+ ログイン認証

方式リストは、ユーザ認証のためクエリ送信を行う手順と認証方式を記述したものです。認証に使用する1つまたは複数のセキュリティプロトコルを指定できるので、最初の方式が失敗した場合のバックアップシステムが確保されます。ソフトウェアは、リスト内の最初の方式を使用してユーザを認証します。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の認証方式を選択します。このプロセスは、リスト内の認証方式による通信が成功するか、定義された方式をすべて試し終わるまで繰り返されます。この処理のある時点で認証が失敗した場合（つまり、セキュリティサーバまたはローカルのユーザ名データベースがユーザアクセスを拒否すると応答した場合）、認証プロセスは停止し、それ以上認証方式が試行されることはありません。

特権 EXEC アクセスおよびネットワーク サービス用の TACACS+ 許可

AAA 許可によってユーザが使用できるサービスが制限されます。AAA 許可がイネーブルに設定されていると、スイッチはユーザーのプロファイルから取得した情報を使用します。このプロファイルは、ローカルのユーザーデータベースまたはセキュリティサーバ上にあり、ユーザーのセッションを設定します。ユーザは、ユーザプロファイル内の情報で認められている場合に限り、要求したサービスのアクセスが認可されます。

TACACS+ Accounting

AAA アカウンティング機能は、ユーザがアクセスしたサービスと、消費したネットワーク リソース量をトラッキングします。AAA アカウンティングをイネーブルにすると、スイッチはユーザーの活動状況をアカウンティング レコードの形式で TACACS+ セキュリティサーバに報告します。各アカウンティング レコードにはアカウンティングの Attribute-Value (AV) ペアが含まれ、レコードはセキュリティサーバに格納されます。このデータを、ネットワーク管理、クライアント請求、または監査のために分析できます。

TACACS+ のデフォルト設定

TACACS+ および AAA は、デフォルトではディセーブルに設定されています。

セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して TACACS+ を設定することはできません。TACACS+ をイネーブルに設定した場合、CLI を通じてスイッチにアクセスするユーザを認証できます。



Note TACACS+ の設定は CLI を使用して行いますが、TACACS+ サーバは権限レベル 15 に設定された HTTP 接続を許可します。

TACACS+ を設定する方法

ここでは、TACACS+ をサポートするようにスイッチを設定する方法について説明します。

TACACS+ サーバホストの指定および認証キーの設定

TACACS+ サーバホストを特定し、認証キーを設定するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **tacacs server *server-name***
4. **address {*ipv4* | *ipv6*} *ip address***
5. **key [*encryption-type*] [*key-string*]**
6. **exit**
7. **aaa new-model**
8. **aaa group server tacacs+ *group-name***
9. **server name *server-name***
10. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。

	Command or Action	Purpose
ステップ 3	tacacs server <i>server-name</i> Example: Device(config)# tacacs server yourserver	TACACS+ サーバを維持する IP ホストを特定します。このコマンドを複数回入力して、優先ホストのリストを作成します。ソフトウェアは、指定された順序でホストを検索します。 <i>server-name</i> にはサーバ名を指定します。
ステップ 4	address { <i>ipv4</i> <i>ipv6</i> } <i>ip address</i> Example: Device(config-server-tacacs)# address ipv4 10.0.1.12	TACACS サーバの IP アドレスを設定します。
ステップ 5	key [<i>encryption-type</i>] [<i>key-string</i>] Example: Device(config-server-tacacs)# key 0 auth-key	アクセスサーバと TACACS+ デーモンとのすべての TACACS+ 通信に使用される認証暗号キーを設定します。この暗号化キーは、TACACS+ デーモンで使用するキーと一致している必要があります。 <i>encryption-type</i> はオプションで、何も指定されていない場合はクリアテキストと見なされます。暗号化されていないキーが後ろに続くよう指定するには、 0 を入力します。暗号化されたキーが後ろに続くよう指定するには、 6 を入力します。隠れたキーが後ろに続くよう指定するには、 7 を入力します。
ステップ 6	exit Example: Device(config-server-tacacs)# exit	TACACS サーバモードを終了して、グローバル コンフィギュレーション モードを開始します。
ステップ 7	aaa new-model Example: Device(config)# aaa new-model	AAA をイネーブルにします。
ステップ 8	aaa group server tacacs+ <i>group-name</i> Example: Device(config)# aaa group server tacacs+ your_server_group	(任意) グループ名を使用して AAA サーバグループを定義し、サーバグループ コンフィギュレーション モードを開始します。
ステップ 9	server name <i>server-name</i> Example: Device(config-sg-tacacs)# server name yourserver	(任意) 特定の TACACS+ サーバを定義済みサーバグループに関連付けます。AAA サーバグループの TACACS+ サーバごとに、このステップを繰り返します。 グループの各サーバは、ステップ 3 で定義済みのものでなければなりません。
ステップ 10	end Example:	サーバグループ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

	Command or Action	Purpose
	Device (config-sg-tacacs) # end	

TACACS+ ログイン認証の設定

TACACS+ ログイン認証を設定するには、次の手順を実行します。

Before you begin

AAA 認証を設定するには、認証方式の名前付きリストを作成してから、各種ポートにそのリストを適用します。



Note AAA 方式を使用して HTTP アクセスに対しデバイスのセキュリティを確保するには、**ip http authentication aaa** グローバル コンフィギュレーション コマンドでデバイスを設定する必要があります。AAA 認証を設定しても、AAA 方式を使用した HTTP アクセスに対しデバイスのセキュリティは確保されません。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login {default | list-name} method1 [method2...]**
5. **line [console | tty | vty] line-number [ending-line-number]**
6. **login authentication {default | list-name}**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model Example:	AAA をイネーブルにします。

	Command or Action	Purpose
	Device (config)# aaa new-model	
ステップ 4	aaa authentication login {default list-name} method1 [method2...] Example: Device (config)# aaa authentication login default tacacs+ local	<p>ログイン認証方式リストを作成します。</p> <ul style="list-style-type: none"> • login authentication コマンドにリストが指定されていない場合に使用するデフォルトのリストを作成するには、default キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。 • <i>list-name</i> には、作成するリストの名前として使用する文字列を指定します。 • <i>method1...</i> には、認証アルゴリズムが試行する実際の方式を指定します。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。 <p>次のいずれかの方式を選択します。</p> <ul style="list-style-type: none"> • <i>enable</i> : イネーブルパスワードを認証に使用します。この認証方式を使用するには、あらかじめ enable password グローバル コンフィギュレーション コマンドを使用してイネーブルパスワードを定義しておく必要があります。 • <i>group tacacs+</i> : TACACS+ 認証を使用します。この認証方式を使用するには、あらかじめ TACACS+ サーバーを設定しておく必要があります。 • <i>line</i> : 回線パスワードを認証に使用します。この認証方式を使用するには、あらかじめ回線パスワードを定義しておく必要があります。password password ライン コンフィギュレーション コマンドを使用します。 • <i>local</i> : ローカルユーザー名データベースを認証に使用します。データベースにユーザ名情報を入力しておく必要があります。username password グローバル コンフィギュレーション コマンドを使用します。 • <i>local-case</i> : 大文字と小文字が区別されるローカルユーザー名データベースを認証に使用しま

	Command or Action	Purpose
		<p>す。 username name password グローバル コンフィギュレーションコマンドを使用して、ユーザー名情報をデータベースに入力する必要があります。</p> <ul style="list-style-type: none"> • <i>none</i> : ログインに認証を使用しません。
ステップ 5	<p>line [console tty vty] line-number [ending-line-number]</p> <p>Example:</p> <pre>Device(config)# line 2 4</pre>	<p>ライン コンフィギュレーションモードを開始し、認証リストを適用する回線を設定します。</p>
ステップ 6	<p>login authentication {default list-name}</p> <p>Example:</p> <pre>Device(config-line)# login authentication default</pre>	<p>1つの回線または複数回線に認証リストを適用します。</p> <ul style="list-style-type: none"> • default を指定する場合は、aaa authentication login コマンドで作成したデフォルトのリストを使用します。 • <i>list-name</i> には、aaa authentication login コマンドで作成したリストを指定します。
ステップ 7	<p>end</p> <p>Example:</p> <pre>Device(config-line)# end</pre>	<p>回線コンフィギュレーションモードを終了します。続いて、特権 EXEC モードに戻ります。</p>

特権 EXEC アクセスおよびネットワーク サービス用の TACACS+ 許可の設定

aaa authorization グローバル コンフィギュレーションコマンドと **tacacs+** キーワードを使用すると、ユーザのネットワークアクセスを特権 EXEC モードに制限するパラメータを設定できます。



Note 許可が設定されていても、CLI を使用してログインし、認証されたユーザに対しては、許可は省略されます。

特権 EXEC アクセスおよびネットワーク サービスに関する TACACS+ 許可を指定するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authorization network authorization-list tacacs+**
4. **aaa authorization exec default tacacs+**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa authorization network authorization-list tacacs+ Example: Device(config)# aaa authorization network list1 tacacs+	ネットワーク関連のすべてのサービス要求に対してユーザー TACACS+ 認可を行うことを設定します。
ステップ 4	aaa authorization exec default tacacs+ Example: Device(config)# aaa authorization exec default tacacs+	ユーザーの特権 EXEC アクセスに対してユーザー TACACS+ 許可を行うことを設定します。 exec キーワードを指定すると、ユーザープロファイル情報（ autocommand 情報など）が返される場合があります。
ステップ 5	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

TACACS+ アカウンティングの起動

TACACS+ アカウンティングを開始するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **aaa accounting network authorization-liststart-stop tacacs+**
4. **aaa accounting exec** デフォルト **start-stop tacacs+**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa accounting network authorization-liststart-stop tacacs+ Example: Device(config)# aaa accounting network list1 start-stop tacacs+	ネットワーク関連のすべてのサービス要求について、TACACS+ アカウンティングをイネーブルにします。
ステップ 4	aaa accounting exec デフォルト start-stop tacacs+ Example: Device(config)# aaa accounting exec default start-stop tacacs+	TACACS+ アカウンティングをイネーブルにして、特権 EXEC プロセスの最初に記録開始アカウンティング通知、最後に記録停止通知を送信します。
ステップ 5	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

What to do next

AAA サーバーが到達不能な場合にデバイスとのセッションを確立するには、**aaa accounting system guarantee-first** コマンドを使用します。これは、最初のレコードとしてシステムアカウンティングを保証します（これがデフォルトの条件です）。場合によっては、システムがリロードされるまでコンソールまたは端末接続でセッションを開始できない場合があります。システムのリロードにかかる時間は 3 分を超えることがあります。

ルータのリロード時に AAA サーバが到達不能な場合に、ルータとのコンソールセッションまたは Telnet セッションを確立するには、**no aaa accounting system guarantee-first** コマンドを使用します。

AAA サーバが到達不能な場合のデバイスとのセッションの確立

AAA サーバが到達不能な場合にデバイスとのセッションを確立するには、**aaa accounting system guarantee-first** コマンドを使用します。これは、最初のレコードとしてシステムアカウントリングを保証します（これがデフォルトの条件です）。場合によっては、システムがリロードされるまでコンソールまたは端末接続でセッションを開始できない場合があります。システムのリロードにかかる時間は 3 分を超えることがあります。

デバイスのリロード時に AAA サーバが到達不能な場合に、デバイスとのコンソールセッションまたは Telnet セッションを確立するには、**no aaa accounting system guarantee-first** コマンドを使用します。

TACACS サーバグループの TACACS ソースインターフェイスの設定

TACACS ソースインターフェイスは、次のいずれかの方法で、TACACS サーバグループの下で設定できます。

- **ip tacacs source-interface *interface-name*** コマンドを使用して、TACACS サーバグループの下に TACACS ソースインターフェイスを設定します。
- **vrf *vrf-name*** コマンドを使用して、TACACS サーバグループの下に VRF を設定してから、**ip tacacs source interface *interface-name* vrf *vrf-name*** コマンドを使用して、設定した VRF をソースインターフェイスにグローバルに関連付けます。

両方のメソッドが設定されている場合、サーバグループ設定の下ではソースインターフェイスが優先されます。

TACACS サーバグループの下で TACACS ソースインターフェイスを設定するには、次の手順を実行します。

始める前に

VRF ルーティングテーブルを設定し、VRF をインターフェイスに関連付ける必要があります

手順の概要

1. **enable**
2. **configure terminal**
3. **{ ip | ipv6 } tacacs source-interface *interface-number* vrf *vrf-name***
4. **aaa group server tacacs *group_name***
5. **ip vrf forwarding *vrf-name***
6. **{ ip | ipv6 } tacacs source-interface *interface-number***
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	{ ip ipv6 } tacacs source-interface interface-number vrf vrf-name 例： Device(config)# ip tacacs source-interface GigabitEthernet1/0/23 vrf vrf17	すべての発信 TACACS パケットに対して、TACACS に、指定したインターフェイスの IP アドレスを強制的に使用させ、VRF ごとに仕様を有効にします。 <ul style="list-style-type: none"> • <i>interface-name</i> : TACACS+ がすべての発信パケットに使用するインターフェイスの名前を指定します。 • <i>vrf vrf-name</i> : VRF ごとの設定を指定します。
ステップ 4	aaa group server tacacs group_name 例： Device(config-sg-tacacs+) # aaa group server tacacs rad-grp	異なる TACACS サーバホストを別々のリストとメソッドにグループ化し、server-group 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 vrf forwarding vrf-name 例： Device(config-sg-tacacs+) # ip vrf forwarding vrf17	(任意) インターフェイスに VRF を設定します。
ステップ 6	{ ip ipv6 } tacacs source-interface interface-number 例： Device(config-sg-tacacs+) # ip tacacs source-interface loopback0	(任意) TACACS+ グループサーバからのすべての発信 TACACS パケットに対して、TACACS+ に、指定したインターフェイスの IP アドレスを強制的に使用させます。 <i>interface-name</i> : TACACS+ がすべての発信パケットに使用するインターフェイスの名前を指定します。
ステップ 7	end 例： Device(config-sg-tacacs+) # end	特権 EXEC モードに戻ります。

TACACS+ のモニタリング

表 8: TACACS+ 情報を表示するためのコマンド

コマンド	目的
show tacacs	TACACS+ サーバの統計情報を表示します。

TACACS+ に関する追加情報

関連資料

関連項目	マニュアル タイトル
AAA の設定	『セキュリティ コンフィギュレーション ガイド』の「認証の設定」、「許可の設定」、および「アカウントिंगの設定」を参照してください。

シスコのテクニカル サポート

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

TACACS+ の機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	TACACS+	TACACS+ は、認証および認可プロセスについて詳細なアカウント情報と柔軟な管理コントロールを提供します。TACACS+は、AAA を介して実装され、AAA コマンドを使用するのみイネーブルにできます。

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



第 7 章

RADIUS の設定

- RADIUS を設定するための前提条件 (149 ページ)
- RADIUS の設定に関する制約事項 (150 ページ)
- RADIUS に関する情報 (151 ページ)
- RADIUS の設定方法 (179 ページ)
- CoA 機能のモニタリング (194 ページ)
- RADIUS の機能の履歴 (194 ページ)

RADIUS を設定するための前提条件

ここでは、RADIUS によるdevice アクセスの制御の前提条件を示します。

全般：

- この章のいずれかのコンフィギュレーションコマンドを使用するには、RADIUSおよび認証、許可、ならびにアカウントिंग (AAA) をイネーブルにする必要があります。
- RADIUS は、AAA を介して実装され、AAA コマンドを使用してのみイネーブルにできません。
- **aaa new-model** グローバル コンフィギュレーション コマンドを使用して、AAA をイネーブルにします。
- **aaa authentication** グローバル コンフィギュレーション コマンドを使用して、RADIUS 認証の方式リストを定義します。
- **line** および **interface** コマンドを使用して、使用する定義済みの方式リストをイネーブルにします。
- 最低限、RADIUS サーバソフトウェアが稼働するホスト (1 つまたは複数) を特定し、RADIUS 認証の方式リストを定義する必要があります。また、任意でRADIUS 許可およびアカウントिंगの方式リストを定義できます。
- device上で RADIUS 機能の設定を行う前に、RADIUS サーバにアクセスし、サーバを設定する必要があります。

- RADIUS ホストは、通常、シスコ（Cisco Secure Access Control Server バージョン 3.0）、Livingston、Merit、Microsoft、または他のソフトウェアプロバイダーの RADIUS サーバソフトウェアが稼働しているマルチユーザシステムです。詳細については、RADIUS サーバのマニュアルを参照してください。
- Change-of-Authorization (CoA) インターフェイスを使用するには、スイッチにセッションがすでに存在している必要があります。CoA を使用すると、セッションの識別と接続解除要求を実行できます。アップデートは、指定されたセッションにだけ作用します。

RADIUS の動作：

- ユーザは RADIUS 許可に進む前に、まず RADIUS 認証を正常に完了する必要があります（イネーブルに設定されている場合）。
- RADIUS over IPv6 構成の場合、ユーザーは **ipv6 unicast-routing** コマンドを有効にして、IPv6 ユニキャストルーティングを有効にする必要があります。

RADIUS の設定に関する制約事項

ここでは、RADIUS による device アクセスの制御の制約事項について説明します。

全般：

- セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して RADIUS を設定することはできません。

RADIUS は次のネットワーク セキュリティ状況には適していません。

- マルチプロトコルアクセス環境。RADIUS は、AppleTalk Remote Access (ARA)、NetBIOS Frame Control Protocol (NBFCP)、NetWare Asynchronous Services Interface (NASI)、または X.25 PAD 接続をサポートしません。
- スイッチ間またはルータ間状態。RADIUS は、双方向認証を行いません。RADIUS は、他社製のデバイスが認証を必要とする場合に、あるデバイスから他社製のデバイスへの認証に使用できます。
- 各種のサービスを使用するネットワーク。RADIUS は、一般に 1 人のユーザを 1 つのサービス モデルにバインドします。

RADIUS に関する情報

RADIUS およびスイッチ アクセス

この項では、RADIUS をイネーブルにし、設定する方法について説明します。RADIUS を使用すると、アカウントの詳細を取得したり、認証および許可プロセスの柔軟な管理制御を実現できます。

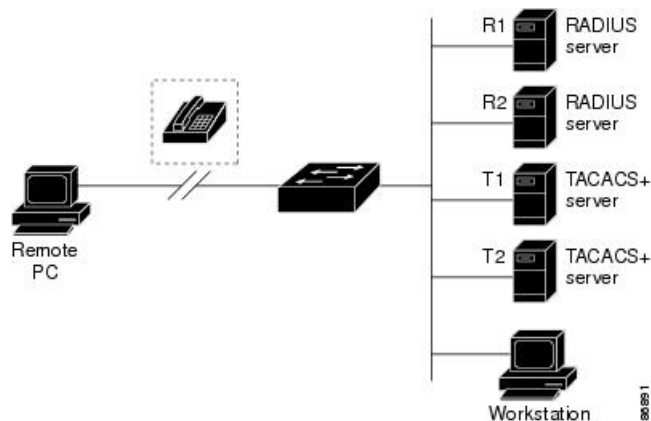
RADIUS の概要

RADIUS は、不正なアクセスからネットワークのセキュリティを保護する分散クライアント/サーバシステムです。RADIUS クライアントは、サポート対象のシスコ デバイス上で稼働します。クライアントは中央の RADIUS サーバに認証要求を送ります。中央の RADIUS サーバにはすべてのユーザ認証情報、ネットワーク サービス アクセス情報が登録されています。

RADIUS は、アクセスのセキュリティが必要な、次のネットワーク環境で使用します。

- それぞれが RADIUS をサポートする、マルチベンダー アクセス サーバによるネットワーク。たとえば、複数のベンダーのアクセスサーバが、1つの RADIUS サーバベースセキュリティ データベースを使用します。複数ベンダーのアクセス サーバからなる IP ベースのネットワークでは、ダイヤルインユーザは RADIUS サーバを通じて認証されます。RADIUS サーバは、Kerberos セキュリティシステムで動作するようにカスタマイズされています。
- アプリケーションが RADIUS プロトコルをサポートするターンキー ネットワーク セキュリティ環境。たとえば、スマートカードアクセスコントロールシステムを使用するアクセス環境。
- すでに RADIUS を使用中のネットワーク。RADIUS クライアント装備のシスコ device をネットワークに追加できます。これが TACACS+ サーバへの移行の最初のステップとなることもあります。下の図「RADIUS サービスから TACACS+ サービスへの移行」を参照してください。

Figure 10: RADIUS サービスから TACACS+ サービスへの移行



- ユーザが1つのサービスにしかアクセスできないネットワーク。RADIUS を使用すると、ユーザのアクセスを1つのホスト、Telnet などの1つのユーティリティ、または IEEE 802.1x などのプロトコルを使用するネットワークに制御できます。このプロトコルの詳細については、「IEEE 802.1x ポートベースの認証の設定」を参照してください。
- リソース アカウンティングが必要なネットワーク。RADIUS 認証または許可とは別個に RADIUS アカウンティングを使用できます。RADIUS アカウンティング機能によって、サービスの開始および終了時点でデータを送信し、このセッション中に使用されるリソース（時間、パケット、バイトなど）の量を表示できます。インターネット サービス プロバイダーは、RADIUS アクセス コントロールおよびアカウンティング ソフトウェアのフリーウェアバージョンを使用して、特殊なセキュリティおよび課金に対するニーズを満たすこともできます。

RADIUS の動作

RADIUS サーバによってアクセス コントロールされる device に、ユーザがログインおよび認証を試みると、次のイベントが発生します。

1. ユーザ名およびパスワードの入力を要求するプロンプトが表示されます。
2. ユーザ名および暗号化されたパスワードが、ネットワーク経由で RADIUS サーバに送信されます。
3. ユーザは、RADIUS サーバから次のいずれかの応答を受信します。
 - ACCEPT : ユーザーが認証されたことを表します。
 - REJECT : ユーザーの認証が失敗し、ユーザー名およびパスワードの再入力が必要されるか、またはアクセスが拒否されます。
 - CHALLENGE : ユーザーに追加データを要求します。
 - CHALLENGE PASSWORD : ユーザーは新しいパスワードを選択するように要求されます。

ACCEPT または REJECT 応答には、特権 EXEC またはネットワーク許可に使用する追加データがバンドルされています。ACCEPT または REJECT パケットには次の追加データが含まれます。

- Telnet、SSH、rlogin、または特権 EXEC サービス
- 接続パラメータ（ホストまたはクライアントの IP アドレス、アクセスリスト、およびユーザ タイムアウトを含む）

RADIUS 許可の変更

RADIUS 許可の変更 (CoA) は、認証、認可、およびアカウンティング (AAA) セッションの属性を認証された後に変更するためのメカニズムを提供します。AAA でユーザー、またはユー

ザグループのポリシーが変更された場合、管理者は、AAA サーバーから Cisco Secure Access Control Server (ACS) などの RADIUS CoA パケットを送信し、認証を再初期化して新しいポリシーを適用することができます。このセクションでは、使用可能なプリミティブおよびそれらの CoA での使用方法を含む、RADIUS インターフェイスの概要について説明します。

- Change-of-Authorization 要求
- CoA 要求応答コード
- CoA 要求コマンド
- セッション再認証
- セッション強制終了のスタック構成ガイドライン

標準 RADIUS インターフェイスは通常、ネットワークに接続しているデバイスから要求が送信され、クエリーが送信されたサーバーが応答するプルモデルで使用されます。シスコ デバイスは、RFC 5176 で規定された（通常はプッシュモデルで使用される）RADIUS CoA 拡張機能をサポートし、外部の AAA またはポリシーサーバーからのセッションを動的に再設定できるようにします。

シスコ デバイスは、次のセッション単位の CoA 要求をサポートしています。

- セッション再認証
- セッションの終了
- ポート シャットダウンでのセッション終了
- ポート バウンスでのセッション終了

この機能は、Cisco Secure Access Control Server (ACS) 5.1 に統合されています。

シスコ デバイスで、RADIUS インターフェイスはデフォルトで有効に設定されています。ただし、次の属性については、一部の基本的な設定が必要になります。

- セキュリティおよびパスワード：このガイドの「スイッチへの不正アクセスの防止」を参照してください。
- アカウンティング：このガイドの「スイッチベース認証の設定」の章の「RADIUS アカウンティングの起動」の項を参照してください。

Cisco IOS XE ソフトウェアは、RFC 5176 で定義されている RADIUS CoA の拡張をサポートします。この拡張は、一般に、外部 AAA またはポリシーサーバーからのセッションの動的な再構成を可能にするプッシュモデルで使用されます。セッションの特定、セッションの終了、ホストの再認証、ポートのシャットダウン、およびポートバウンスでは、セッションごとの CoA 要求がサポートされます。このモデルは、次のように、1つの要求 (CoA-Request) と2つの考えられる応答コードで構成されます。

- CoA acknowledgement (ACK) [CoA-ACK]
- CoA nonacknowledgement (NAK) [CoA-NAK]

要求は CoA クライアント（通常は AAA またはポリシー サーバー）から開始されて、リスナーとして動作するデバイスに転送されます。

次の表は、Identity-Based Networking Services でサポートされている RADIUS CoA コマンドとベンダー固有属性（VSA）を示します。すべての CoA コマンドには、デバイスと CoA クライアント間のセッション ID が含まれている必要があります。

Table 9: Identity-Based Networking Services でサポートされている RADIUS CoA コマンド

CoA コマンド	シスコの VSA
Activate service	Cisco:Avpair="subscriber:command=activate-service" Cisco:Avpair="subscriber:service-name=<service-name>" Cisco:Avpair="subscriber:precedence=<precedence-number>" Cisco:Avpair="subscriber:activation-mode=replace-all"
Deactivate service	Cisco:Avpair="subscriber:command=deactivate-service" Cisco:Avpair="subscriber:service-name=<service-name>"
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"
Session query	Cisco:Avpair="subscriber:command=session-query"
Session reauthenticate	Cisco:Avpair="subscriber:command=reauthenticate" Cisco:Avpair="subscriber:reauthenticate-type=last" または Cisco:Avpair="subscriber:reauthenticate-type=rerun"
Session terminate	これは、VSA を必要としない、標準の接続解除要求です。
Interface template	Cisco:AVpair="interface-template-name=<interfacetemplate>"

Change-of-Authorization 要求

Change of Authorization (CoA) 要求は、RFC 5176 に記載されているように、プッシュ モデルで使用するによって、セッション識別、ホスト再認証、およびセッション終了を行うことができます。このモデルは、1つの要求 (CoA-Request) と2つの可能な応答コードで構成されています。

- CoA acknowledgment (ACK) [CoA-ACK]
- CoA non-acknowledgment (NAK) [CoA-NAK]

要求は CoA クライアント（通常は RADIUS またはポリシー サーバー）から発信されて、リスナーとして動作するスイッチに送信されます。

RFC 5176 規定

Disconnect Request メッセージは Packet of Disconnect (POD) とも呼ばれますが、セッション終了に対してスイッチでサポートされています。

次の表に、この機能でサポートされている IETF 属性を示します。

Table 10: サポートされている IETF 属性

属性番号	属性名
24	状態
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

次の表に、Error-Cause 属性で取ることができる値を示します。

Table 11: Error-Cause の値

値	説明
21	削除された残留セッション コンテキスト
22	無効な EAP パケット (無視)
41	サポートされていない属性
42	見つからない属性
43	NAS 識別情報のミスマッチ
44	無効な要求
45	サポートされていないサービス
46	サポートされていない拡張機能
47	無効な属性値
31	管理上の禁止
32	ルート不可能な要求 (プロキシ)
33	セッション コンテキストが検出されない
34	セッション コンテキストが削除できない

値	説明
35	その他のプロキシ処理エラー
36	リソースが使用不可能
37	要求が発信された
38	マルチセッションの選択がサポートされていない

CoA 要求応答コード

CoA 要求応答コードを使用すると、スイッチにコマンドを伝達できます。

RFC 5176 で定義されている CoA 要求応答コードの packets の形式は、コード、ID、長さ、オーセンティケータ、およびタイプ、長さ、値 (TLV) 形式の属性から構成されます。属性フィールドは、シスコのベンダー固有属性 (VSA) を送信するために使用します。

セッションの識別

特定のセッションに向けられた切断と CoA 要求については、スイッチは 1 つ以上の次の属性に基づいて、セッションを検索します。

- Acct-Session-Id (IETF 属性 #44)
- Audit-Session-Id VSA (シスコの VSA)
- Calling-Station-Id (ホスト MAC アドレスを含む IETF 属性 #31)
- 次のいずれかの IPv6 属性。
 - Framed-IPv6-Prefix (IETF 属性 #97) および Framed-Interface-Id (IETF 属性 #96)。ともに RFC 3162 に従った完全な IPv6 アドレスを作成する
 - Framed-IPv6-Address
- プレーン IP アドレス (IETF 属性 #8)

CoA メッセージに含まれるすべてのセッション ID 属性がそのセッションと一致しない限り、スイッチは「Invalid Attribute Value」エラーコード属性を含む Disconnect-NAK または CoA-NAK を返します。

複数のセッション ID 属性がメッセージに含まれる場合は、すべての属性がセッションと一致しなければなりません。そうでない場合は、スイッチが Disconnect - negative acknowledgement (NAK) または CoA -NAK と、「Invalid Attribute Value」エラーコードを返します。

RFC 5176 で定義されている CoA 要求コードの packets の形式は、コード、ID、長さ、オーセンティケータ、およびタイプ、長さ、値 (TLV) 形式の属性から構成されます。

0

1

2

3

```

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|   Code   | Identifier |           Length           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Authenticator                                     |
|                                                                                                                                 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Attributes ... |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

属性フィールドは、シスコのベンダー固有属性（VSA）を送信するために使用します。

特定の適用ポリシーを対象とする CoA 要求の場合、上記のセッション ID 属性のいずれかがメッセージに含まれていると、デバイスはエラーコードが「Invalid Attribute Value」の CoA-NAK を返します。

CoA ACK 応答コード

許可ステートの変更成功した場合は、肯定確認応答（ACK）が送信されます。CoA ACK 内で返される属性は CoA 要求によって異なり、個々の CoA コマンドで検討されます。

CoA NAK 応答コード

否定応答（NAK）は許可ステートの変更に失敗したことを示し、エラーの理由を示す属性を含めることができます。CoA が成功したかを確認するには、**show** コマンドを使用します。

CoA 要求コマンド

Table 12: サポートされる CoA コマンド

コマンド	シスコの VSA
1	
Reauthenticate host	Cisco:Avpair="subscriber:command=reauthenticate"
Terminate session	これは、VSA を要求しない、標準の接続解除要求です。
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"

¹ すべての CoA コマンドには、デバイスと CoA クライアント間のセッション ID が含まれている必要があります。

セッション再認証

不明な ID またはポストチャを持つホストがネットワークに加入して、制限されたアクセス許可プロファイル（たとえば、ゲスト VLAN）に関連付けられると、AAA サーバーは通常、セッション再認証要求を生成します。再認証要求は、クレデンシャルが不明である場合にホストが適切な認証グループに配置されることを許可します。

セッション認証を開始するために、AAA サーバーは `Cisco:Avpair="subscriber:command=reauthenticate"` の形式で Cisco VSA と 1 つ以上のセッション ID 属性を含む標準 CoA 要求メッセージを送信します。

現在のセッションステートは、メッセージに対するスイッチの応答を決定します。セッションが現在、IEEE 802.1x によって認証されている場合、スイッチは EAPOL (LAN 経由の拡張認証プロトコル) RequestId メッセージをサーバーに送信することで応答します。

現在、セッションが MAC 認証バイパス (MAB) で認証されている場合は、スイッチはサーバーにアクセス要求を送信し、初期正常認証で使用されるものと同じ ID 属性を渡します。

スイッチがコマンドを受信した際にセッション認証が実行中である場合は、スイッチはプロセスを終了し、認証シーケンスを再開し、最初に試行されるように設定された方式で開始します。

セッションがまだ認証されていない、あるいはゲスト VLAN、クリティカル VLAN、または同様のポリシーで認証されている場合は、再認証メッセージがアクセス コントロール方式を再開し、最初に試行されるように設定された方式で開始します。セッションの現在の許可は、再認証によって異なる認証結果になるまで維持されます。

スイッチ スタックでのセッションの再認証

スイッチ スタックでセッション再認証メッセージを受信すると、次の動作が発生します。

- 確認応答 (ACK) を戻す前に、再認証の必要性がチェックされます。
- 適切なセッションで再認証が開始されます。
- 認証が成功または失敗のいずれかで完了すると、再認証をトリガーする信号がスタック メンバから削除されます。
- 認証の完了前にアクティブスイッチに障害が発生すると、(後で削除される) 元のコマンドに基づいたアクティブスイッチの切り替え後、再認証が開始されます。
- ACK の送信前にアクティブスイッチに障害が発生した場合、新たなアクティブスイッチでは、再送信コマンドが新しいコマンドとして扱われます。

セッションの終了

セッションを終了させる 3 種類の CoA 要求があります。CoA 接続解除要求は、ホストポートをディセーブルにせずにセッションを終了します。このコマンドを使用すると、指定されたホストのオーセンティケータ ステート マシンが再初期化されますが、そのホストのネットワークへのアクセスは制限されません。

ホストのネットワークへのアクセスを制限するには、`Cisco:Avpair="subscriber:command=disable-host-port"` VSA の設定で CoA 要求を使用します。このコマンドは、ネットワーク上で障害を引き起こしたと認識されているホストがある場合に便利であり、そのホストに対してネットワークアクセスをただちにブロックする必要があります。ポートへのネットワーク アクセスを復旧する場合は、非 RADIUS メカニズムを使用して再びイネーブルにします。

プリンタなどのサブリカントを持たないデバイスが新しい IP アドレスを取得する必要がある場合（たとえば、VLAN 変更後）は、ポートバウンスでホストポート上のセッションを終了します（ポートを一時的にディセーブルした後、再びイネーブルにする）。

CoA 接続解除要求

このコマンドは標準の接続解除要求です。セッションが見つからない場合、デバイスは「Session Context Not Found」エラーコード属性を使用して Disconnect-NAK メッセージを返します。セッションが見つかった場合、デバイスはセッションを終了します。セッションが完全に削除されると、デバイスは Disconnect-ACK を返します。

デバイスがクライアントに接続解除 ACK を返す前にスタンバイデバイスにフェールオーバーする場合は、クライアントから要求が再送信される際に、新しいアクティブデバイス上でそのプロセスが繰り返されます。再送信後もセッションが見つからない場合は、Disconnect-ACK と「Session Context Not Found」エラーコード属性が送信されます。

CoA 要求：ホストポートのディセーブル化

RADIUS サーバーの CoA disable port コマンドを実行すると、セッションをホストしている認証ポートが管理的にシャットダウンされます。その結果、セッションは終了します。このコマンドは、ホストがネットワーク上で問題を起こしていることを把握し、ホストのネットワークアクセスを即座にブロックする必要がある場合に便利です。ポートのネットワークアクセスを復元するには、非RADIUS メカニズムを使用して再びイネーブルにします。このコマンドは、次の新しいベンダー固有属性（VSA）が含まれている標準 CoA 要求メッセージで伝達されます。

```
Cisco:Avpair="subscriber:command=disable-host-port"
```

このコマンドはセッション指向であるため、「セッション ID」セクションに示されている 1 つ以上のセッション ID 属性とともに使用する必要があります。セッションが見つからない場合、デバイスは「Session Context Not Found」エラーコード属性を使用して CoA-NAK メッセージを返します。このセッションがある場合は、デバイスはホストポートを無効にし、CoA-ACK メッセージを返します。

デバイスが CoA-ACK をクライアントに返す前にデバイスに障害が発生した場合、クライアントから要求が再送信されると、新しいアクティブデバイス上でそのプロセスが繰り返されます。デバイスが CoA-ACK メッセージをクライアントに返した後で障害が発生したが、操作が完了していない場合、その操作は新しいアクティブデバイスで再開されます。



Note 再送信コマンドの後に接続解除要求が失敗すると、（接続解除ACKが送信されていない場合に）チェンジオーバー前にセッションが正常終了するか、または元のコマンドが実行されてスタンバイデバイスがアクティブになるまでの間に発生した他の方法（たとえば、リンク障害）によりセッションが終了することがあります。

CoA 要求：バウンスポート

RADIUS サーバーの CoA bounce port が RADIUS サーバーから送信されると、認証ポートでリンクのフラップが発生します。その結果、このポートに接続している 1 つまたは複数のホスト

から、DHCP の再ネゴシエーションが開始されます。この状況は、VLAN の変更があり、この認証ポートに関する変化を検出するメカニズムがないデバイス（プリンタなど）がエンドポイントの場合に発生する可能性があります。CoA bounce port は、次の新しい VSA を含む標準の CoA-Request メッセージで伝達されます。

```
Cisco:Avpair="subscriber:command=bounce-host-port"
```

このコマンドはセッション指向であるため、1 つ以上のセッション ID 属性とともに使用する必要があります。セッションが見つからない場合、デバイスは「Session Context Not Found」エラーコード属性を使用して CoA-NAK メッセージを返します。このセッションがある場合は、デバイスはホストポートを 10 秒間無効にし、再び有効にし（ポートバウンス）、CoA-ACK を返します。

デバイスが CoA-ACK をクライアントに返す前にデバイスに障害が発生した場合、クライアントから要求が再送信されると、新しいアクティブ デバイス上でそのプロセスが繰り返されます。デバイスが CoA-ACK メッセージをクライアントに返した後で障害が発生したが、操作が完了していない場合、その操作は新しいアクティブデバイスで再開されます。

セッション強制終了のスタック構成ガイドライン

スイッチ スタックでは、CoA 接続解除要求メッセージに必要な特別な処理はありません。

CoA 要求バウンス ポートのスタック構成ガイドライン

bounce-port コマンドのターゲットはポートではなくセッションのため、セッションが見つからなかった場合、コマンドは実行できません。

アクティブスイッチで Auth Manager コマンドハンドラが有効な **bounce-port** コマンドを受信すると、CoA-ACK メッセージを返す前に次の情報が確認されます。

- ポート バウンスの必要性
- ポート ID（ローカルセッション コンテキストで検出された場合）

スイッチで、ポート バウンスが開始されます（ポートが 10 秒間ディセーブルになり、再びネーブルにされます）。

ポートバウンスが正常に実行された場合、ポートバウンスをトリガーした信号がスタンバイスイッチから削除されます。

ポートバウンスの完了前にアクティブスイッチに障害が発生すると、（後で削除される）元のコマンドに基づいたアクティブスイッチの切り替え後、ポートバウンスが開始されます。

CoA-ACK メッセージの送信前にアクティブスイッチに障害が発生した場合、新たなアクティブスイッチでは、再送信コマンドが新しいコマンドとして扱われます。

CoA 要求ディセーブル ポートのスタック構成ガイドライン

disable-port コマンドのターゲットはポートではなくセッションのため、セッションが見つからなかった場合、コマンドは実行できません。

アクティブスイッチで Auth Manager コマンドハンドラが有効な **disable-port** コマンドを受信すると、CoA-ACK メッセージを返す前に次の情報が確認されます。

- ポート ディセーブルの必要性
- ポート ID (ローカルセッション コンテキストで検出された場合)

スイッチで、ポートをディセーブルする操作が試行されます。

ポートを無効にする操作が正常に実行された場合、ポートを無効にする操作をトリガーした信号がスタンバイスイッチから削除されます。

ポートを無効にする操作の完了前にアクティブスイッチに障害が発生すると、(後で削除される) 元のコマンドに基づいたアクティブスイッチの切り替え後、ポートが無効にされます。

CoA-ACK メッセージの送信前にアクティブスイッチに障害が発生した場合、新たなアクティブスイッチでは、再送信コマンドが新しいコマンドとして扱われます。

RADIUS のデフォルト設定

RADIUS および AAA は、デフォルトではディセーブルに設定されています。

セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して RADIUS を設定することはできません。RADIUS を有効にすると、CLI 経由でデバイスにアクセスするユーザーを認証できます。

RADIUS サーバホスト

デバイスと RADIUS サーバー間の通信には、次の要素が関係します。

- ホスト名または IP アドレス
- 認証の宛先ポート
- アカウンティングの宛先ポート
- キー文字列
- タイムアウト時間
- 再送信回数

RADIUS セキュリティ サーバは、ホスト名または IP アドレス、ホスト名と特定の UDP ポート番号、または IP アドレスと特定の UDP ポート番号によって特定します。IP アドレスと UDP ポート番号の組み合わせによって、一意の ID が作成され、特定の AAA サービスを提供する RADIUS ホストとして個々のポートを定義できます。この一意の ID を使用することによって、同じ IP アドレスにあるサーバ上の複数の UDP ポートに、RADIUS 要求を送信できます。

同じ RADIUS サーバー上の異なる 2 つのホスト エントリに同じサービス (たとえばアカウンティング) を設定した場合、2 番めに設定したホスト エントリは、最初に設定したホスト エントリのフェールオーバーバックアップとして動作します。この例では、最初のホスト エントリがアカウンティング サービスを提供できなかった場合、デバイスは

「%RADIUS-4-RADIUS_DEAD」メッセージを表示し、その後、同じデバイス上で 2 番目に設

定されたホストエントリでアカウントिंगサービスを試みます (RADIUS ホスト エントリは、設定した順序に従って試行されます)。

RADIUS サーバーとデバイスは、共有秘密テキスト文字列を使用して、パスワードの暗号化および応答の交換を行います。RADIUS で AAA セキュリティ コマンドを使用するように設定するには、RADIUS サーバーデーモンが稼働するホストと、そのホストがデバイスと共有する秘密テキスト (キー) 文字列を指定する必要があります。

タイムアウト、再送信回数、および暗号キーの値は、すべての RADIUS サーバに対してグローバルに設定することもできますし、サーバ単位で設定することもできます。また、グローバルな設定とサーバ単位での設定を組み合わせることもできます。

RADIUS ログイン認証

AAA 認証を設定するには、認証方式の名前付きリストを作成してから、各種ポートにそのリストを適用します。方式リストは実行される認証のタイプと実行順序を定義します。このリストを特定のポートに適用してから、定義済み認証方式を実行する必要があります。唯一の例外は、デフォルトの方式リストです。デフォルトの方式リストは、名前付き方式リストを明示的に定義されたインターフェイスを除いて、自動的にすべてのポートに適用されます。

方式リストは、ユーザ認証のためクエリ送信を行う手順と認証方式を記述したものです。認証に使用する 1 つまたは複数のセキュリティプロトコルを指定できるので、最初の方式が失敗した場合のバックアップシステムが確保されます。ソフトウェアは、リスト内の最初の方式を使用してユーザを認証します。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の認証方式を選択します。このプロセスは、リスト内の認証方式による通信が成功するか、定義された方式をすべて試し終わるまで繰り返されます。この処理のある時点で認証が失敗した場合 (つまり、セキュリティサーバまたはローカルのユーザ名データベースがユーザアクセスを拒否すると応答した場合)、認証プロセスは停止し、それ以上認証方式が試行されることはありません。

AAA サーバグループ

既存のサーバホストを認証用にグループ化するため、AAA サーバグループを使用するようにデバイスを設定できます。設定済みのサーバホストのサブセットを選択して、それを特定のサービスに使用します。サーバグループは、選択されたサーバホストの IP アドレスのリストを含むグローバルなサーバホストリストとともに使用されます。

サーバグループには、同じサーバの複数のホストエントリを含めることもできますが、各エントリが一意的 ID (IP アドレスと UDP ポート番号の組み合わせ) を持っていることが条件です。この場合、個々のポートをそれぞれ特定の AAA サービスを提供する RADIUS ホストとして定義できます。この一意の ID を使用することによって、同じ IP アドレスにあるサーバ上の異なる UDP ポートに、RADIUS 要求を送信できます。同じ RADIUS サーバ上の異なる 2 つのホストエントリに同じサービス (たとえばアカウントिंग) を設定した場合、2 番めに設定したホストエントリは、最初に設定したホストエントリのフェールオーバー バックアップとして動作します。最初のホストエントリがアカウントングサービスの提供に失敗すると、ネットワーク アクセス サーバは同じデバイスに設定されている 2 番目のホストエントリを使

用してアカウントリング サービスを提供するように試行します。（試行される RADIUS ホスト エントリの順番は、設定されている順序に従います）。

AAA 許可

AAA 許可によってユーザが利用できるサービスが制限されます。AAA 許可が有効になると、デバイスはユーザのプロファイルから取得した情報を使用します。このプロファイルは、ローカルのユーザデータベースまたはセキュリティサーバ上にあり、ユーザのセッションを設定します。ユーザは、ユーザプロファイル内の情報で認められている場合に限り、要求したサービスのアクセスが認可されます。

RADIUS アカウンティング

AAA アカウンティング機能は、ユーザが使用したサービスと、消費したネットワーク リソース量を追跡します。AAA アカウンティングを有効にすると、デバイスはユーザアクティビティをアカウントングレコードの形式で RADIUS セキュリティサーバに報告します。各アカウントングレコードにはアカウントングの Attribute-Value (AV) ペアが含まれ、レコードはセキュリティサーバに格納されます。これらのデータは、ネットワーク管理、クライアントへの課金、または監査のために後で分析できます。

ベンダー固有の RADIUS 属性

Internet Engineering Task Force (IETF) ドラフト規格に、ベンダー固有の属性 (属性 26) を使用して、デバイスと RADIUS サーバ間でベンダー固有の情報を通信するための方式が定められています。各ベンダーは、Vendor-Specific Attribute (VSA) を使用することによって、一般的な用途には適さない独自の拡張属性をサポートできます。シスコが実装する RADIUS では、この仕様で推奨されるフォーマットを使用して、ベンダー固有のオプションを1つサポートしています。シスコのベンダー ID は 9 であり、サポート対象のオプションはベンダータイプ 1 (名前は *cisco-avpair*) です。この値は、次のフォーマットのストリングです。

```
protocol : attribute sep value *
```

protocol は、特定の認証タイプに使用するシスコのプロトコル属性の値です。*attribute* および *value* は、シスコの TACACS+ 仕様で定義されている適切な属性値 (AV) ペアです。*sep* は、必須の属性の場合は =、任意指定の属性の場合は * です。TACACS+ 認証で使用できるすべての機能は、RADIUS でも使用できます。

たとえば、次の AV ペアにより、IP 認証中 (PPP の IPCP アドレス割り当て中) には、シスコの「multiple named IP address pools」機能がアクティブになります。

```
cisco-avpair= "ip:addr-pool=first"
```

「*」を挿入すると、AV ペア「ip:addr-pool=first」は省略可能になります。任意の AV ペアを省略可能にすることができます。

```
cisco-avpair= "ip:addr-pool*first"
```

次に、ネットワーク アクセス サーバからユーザがログインしたときに、すぐに EXEC コマンドを実行する方法の例を示します。

```
cisco-avpair= "shell:priv-lvl=15"
```

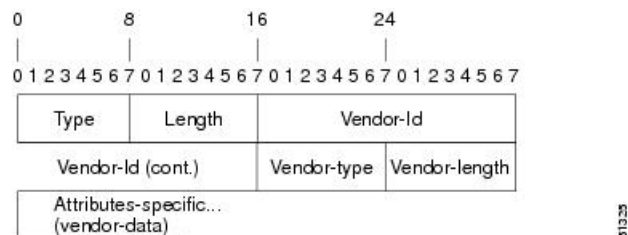
他のベンダーにも、それぞれ独自のベンダー ID、オプション、および対応する VSA があります。ベンダー ID および VSA の詳細については、RFC 2138 『Remote Authentication Dial-In User Service (RADIUS)』を参照してください。

属性 26 には、次の 3 つの要素が含まれています。

- タイプ
- 長さ
- スtring (またはデータ)
 - Vendor-ID
 - Vendor-Type
 - Vendor-Length
 - Vendor-Data

次の図は、属性 26 の「背後で」カプセル化される VSA のパケット形式を示します。

図 11: 属性 26 の背後でカプセル化される VSA



(注) VSA の形式はベンダーが指定します。Attribute-Specific フィールド (Vendor-Data と呼ばれる) は、ベンダーによるその属性の定義によって異なります。

次の表に、「ベンダー固有 RADIUS IETF 属性テーブル」(次の 2 番目の表) で表示される重要なフィールドを示します。これは、サポート対象のベンダー固有 RADIUS 属性 (IETF 属性 26) を表示します。

表 13: ベンダー固有属性表のフィールドの説明

フィールド	説明
番号	次の表に示されるすべての属性は、IETF 属性 26 の拡張です。
ベンダー固有のコマンドコード	特定のベンダーの識別に使用する定義されたコード。コード 9 は Cisco VSA、311 は Microsoft VSA、529 は Ascend VSA を定義します。

フィールド	説明
サブタイプ番号	属性 ID 番号。この番号は、属性 26 の背後でカプセル化される「2 番目のレイヤ」の ID 番号であること以外、IETF 属性の ID 番号に似ています。
属性	属性の ASCII スtring 名。
説明	属性の説明。

表 14: ベンダー固有 RADIUS IETF 属性

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
MS-CHAP 属性				
26	311	1	MSCHAP-Response	PPP MS-CHAP ユーザが チャレンジに対する応 答で提供するレスポ ンス値が含まれます。 Access-Request パケッ トでしか使用されませ ん。この属性は、PPP CHAP ID と同じです (RFC 2548)
26	311	11	MSCHAP-Challenge	ネットワーク アクセス サーバが MS-CHAP ユーザに送信するチャ レンジが含まれます。 これは、Access-Request パケットと Access-Challenge パケッ トの両方で使用できま す。(RFC 2548)
VPDN 属性				
26	9	1	l2tp-cm-local-window-size	L2TP 制御メッセージの 最大受信ウィンドウ サ イズを指定します。こ の値は、トンネルの確 立中にピアにアドバタ イズされます。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	1	l2tp-drop-out-of-order	正しくない順序で受信したデータパケットをドロップして、シーケンス番号を順守します。これは受信した場合の処理方法であって、データパケット上でシーケンス番号が送信されるわけではありません。
26	9	1	l2tp-hello-interval	hello キープアライブインターバルの秒数を指定します。ここで指定した秒数、トンネルでデータが送信されないと、hello パケットが送信されます。
26	9	1	l2tp-hidden-avp	イネーブルにすると、L2TP 制御メッセージで、大文字小文字を区別する AVP にスクランブルがかけられるか、または非表示になります。
26	9	1	l2tp-nosession-timeout	タイムアウトおよびシャットダウンまでに、セッションなしでトンネルがアクティブのままになる秒数を指定します。
26	9	1	tunnel-tos-reflect	LNS でトンネルに入るパケットに対して、IP ToS フィールドを各ペイロードパケットの IP ヘッダーからトンネルパケットの IP ヘッダーにコピーします。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	1	l2tp-tunnel-authen	この属性を設定すると、L2TP トンネル認証が実行されます。
26	9	1	l2tp-tunnel-password	L2TP トンネル認証および AVP 隠蔽に使用される共有秘密。
26	9	1	l2tp-udp-checksum	これは認可属性で、L2TP がデータ パケットに対して UDP チェックサムを実行する必要があるかどうかを定義します。有効な値は「yes」と「no」です。デフォルトは「no」です。
Store and Forward Fax 属性				
26	9	3	Fax-Account-Id-Origin	mmoip aaa receive-id コマンドまたは mmoip aaa send-id コマンドについて、システム管理者によって定義されたものとしてアカウント ID の発信元を示します。
26	9	4	Fax-Msg-Id=	Store and Forward Fax 機能によって割り当てられた一意のファクスメッセージ識別番号を示します。
26	9	5	Fax-Pages	このファクスセッション中に送信または受信したページ数を示します。このページ数には、カバー ページも含まれます。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	6	Fax-Coverpage-Flag	カバー ページがこの ファクスセッションの オフランプゲートウェ イで生成されたかどう かを示します。true は カバー ページが生成さ れたことを示します。 false はカバー ページが 生成されなかったこと を意味します。
26	9	7	Fax-Modem-Time	モデムがファクスデー タを送信した時間 (x)、およびファクス セッションの合計時間 (y) を秒単位で示しま す。これには、fax-mail および PSTN 時間が x/y の形式で含まれます。 たとえば、10/15 は送信 時間が 10 秒で、合計 ファクスセッションが 15 秒であったことを示 します。
26	9	8	Fax-Connect-Speed	この fax-mail が最初に 送信または受信された 時点のモデム速度を示 します。有効値は、 1200、4800、9600、お よび 14400 です。
26	9	9	Fax-Recipient-Count	このファクス送信の受 信者数を示します。E メールサーバがセッ ションモードをサポ ートするまで、この数字 は 1 にする必要があります。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	10	Fax-Process-Abort-Flag	ファクスセッションが中断したこと、または正常に終了したことを示します。true はセッションが中断したことを示します。false はセッションが成功したことを示します。
26	9	11	Fax-Dsn-Address	DSN の送信先のアドレスを示します。
26	9	12	Fax-Dsn-Flag	DSN がイネーブルにされているかどうかを示します。true は DSN がイネーブルにされていることを示します。false は DSN がイネーブルにされていないことを示します。
26	9	13	Fax-Mdn-Address	MDN の送信先のアドレスを示します。
26	9	14	Fax-Mdn-Flag	メッセージ配信通知 (MDN) がイネーブルにされているかどうかを示します。true は MDN がイネーブルにされていることを示します。false は MDN がイネーブルにされていないことを示します。
26	9	15	Fax-Auth-Status	このファクスセッションに対する認証が成功したかどうかを示します。このフィールドに対する有効値は、success、failed、bypassed、または unknown です。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	16	Email-Server-Address	オンランプ fax-mail メッセージを処理する Eメールサーバの IP ア ドレスを示します。
26	9	17	Email-Server-Ack-Flag	オンランプ ゲートウェ イが fax-mail メッセー ジを受け入れる E メール サーバから肯定確認 応答を受信したことを 示します。
26	9	18	Gateway-Id	ファクスセッションを 処理したゲートウェイ の名前を示します。名 前は、 hostname.domain-name という形式で表示され ます。
26	9	19	Call-Type	ファクスのアクティビ ティのタイプを、fax receive または fax send のどちらかで記述しま す。
26	9	20	Port-Used	この fax-mail の送受信 いずれかに使用される Cisco AS5300 のスロッ ト/ポート番号を示しま す。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	21	Abort-Cause	ファクスセッションが中断した場合、中断の信号を送信したシステムコンポーネントを示します。中断する可能性のあるシステムコンポーネントには、FAP (Fax Application Process)、TIFF (TIFFリーダーまたはTIFFライター)、fax-mailクライアント、fax-mailサーバー、ESMTPクライアント、ESMTPサーバーなどがあります。
H323 属性				
26	9	23	Remote-Gateway-ID (h323-remote-address)	リモートゲートウェイのIPアドレスを示します。
26	9	24	Connection-ID (h323-conf-id)	会議IDを識別します。
26	9	25	Setup-Time (h323-setup-time)	以前、グリニッジ標準時 (GMT) およびズールタイムと呼ばれていた協定世界時 (UTC) でのこの接続のセットアップ時間を示します。
26	9	26	Call-Origin (h323-call-origin)	ゲートウェイに対するコールの発行元を示します。有効値は、 originating および terminating です (回答)。
26	9	27	Call-Type (h323-call-type)	コールのレグタイプを示します。使用可能な値は telephony と VoIP です。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	28	Connect-Time (h323-connect-time)	このコール レッグの UTC での接続時間を示 します。
26	9	29	Disconnect-Time (h323-disconnect-time)	このコール レッグが UTC で接続解除された 時間を示します。
26	9	30	Disconnect-Cause (h323-disconnect-cause)	Q.931 仕様によって、 接続がオフラインにさ れた理由を示します。
26	9	31	Voice-Quality (h323-voice-quality)	コールの音声品質に影 響する Impairment Factor (ICPIF) を指定しま す。
26	9	33	Gateway-ID (h323-gw-id)	下位のゲートウェイの 名前を示します。
大規模のダイヤルアウト属性				
26	9	1	callback-dialstring	コールバックに使用す るダイヤリング文字列 を定義します。
26	9	1	data-service	説明はありません。
26	9	1	dial-number	ダイヤルする番号を定 義します。
26	9	1	force-56	チャンネルの 64 K すべ てが使用可能に見える場 合でも、ネットワーク アクセスサーバが 56 K の部分のみを使用する かどうかを指定しま す。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	1	map-class	ユーザ プロファイルに、ダイヤルアウトするネットワーク アクセス サーバ上で同じ名前のマップクラスで設定される情報の参照を許可します。
26	9	1	send-auth	CLID 認証に続く、username-password 認証で使用するプロトコル (PAP または CHAP) を定義します。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	1	send-name	

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
				<p>PPP 名前認証。PAP に適用する場合、インターフェイスで ppp pap sent-name password コマンドは設定しないでください。PAP の場合、アウトバウンド認証の PAP ユーザ名および PAP パスワードとして、</p> <p>「preauth:send-name」および 「preauth:send-secret」が使用されます。CHAP の場合、</p> <p>「preauth:send-name」は、アウトバウンド認証だけでなく、インバウンド認証にも使用されます。CHAP インバウンドの場合、NAS は発信元のボックスへのチャレンジパケットに、</p> <p>「preauth:send-name」で定義された名前を使用します。</p> <p>(注) send-name 属性は時間の経過とともに変わっています。最初は、現在 send-name および remote-name 属性の両方で提供されている機能を実行していました。</p>

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
				remote-name 属性が追加 されたた め、 send-name 属性は現在 の動作に制 限されてい ます。
26	9	1	send-secret	PPP パスワード認証。 ベンダー固有属性 (VSA) の場合、アウ トバウンド認証の PAP ユーザ名および PAP パ スワードとして、 「preauth:send-name」お よび 「preauth:send-secret」 が使用されます。CHAP アウトバウンドの場 合、 「preauth:send-name」と 「preauth:send-secret」 の両方が応答パケット で使用されます。
26	9	1	remote-name	大規模のダイヤルアウ トで使用するリモート ホストの名前を提供し ます。ダイヤラは、大 規模のダイヤルアウト のリモート名が認証さ れた名前と一致するこ を確認し、偶発的な ユーザ RADIUS 設定ミ スから保護します (有 効な電話番号にダイヤ ルしたが誤ったデバイ スに接続されるなどの ミスです)。
その他の属性				

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	2	Cisco-NAS-Port	<p>NAS-Port アカウンティングに追加的なベンダー固有属性 (VSA) を指定します。追加的な NAS-Port 情報を属性値ペア (AVPair) の形式で指定するには、radius-server vsa send グローバル コンフィギュレーションコマンドを使用します。</p> <p>(注) この VSA は、通常アカウンティングで使用されますが認証 (Access-Request) パケットで使用される場合もあります。</p>
26	9	1	min-links	MLP に対するリンクの最小数を設定します。
26	9	1	proxyacl#<n>	ダウンロード可能なユーザ プロファイル (ダイナミック ACL) を、認証プロキシを使用して設定でき、これにより設定されたインターフェイスのトラフィックの通過を許可するよう、認証を設定できます。

番号	ベンダー固有の 企業コード	サブタイプ番号	属性	説明
26	9	1	spi	登録中にホーム エージェントがモバイルノードの認証で必要とする認証情報を伝送します。この情報は、 ip mobile secure host <addr> コンフィギュレーションコマンドと同じ構文です。基本的に、この文字列に続く残りのコンフィギュレーションコマンドはそのまま含まれます。これにはセキュリティパラメータ インデックス (SPI)、キー、認証アルゴリズム、認証モード、およびリプレイ保護タイムスタンプ範囲が含まれています。

ベンダー独自仕様の RADIUS サーバー通信

RADIUS に関する IETF ドラフト規格では、デバイスと RADIUS サーバー間でベンダー独自仕様の情報を通信する方式について定められていますが、RADIUS 属性セットを独自に機能拡張しているベンダーもあります。Cisco IOS XE ソフトウェアは、ベンダー独自仕様の RADIUS 属性のサブセットをサポートしています。

前述のように、（ベンダー固有か IETF ドラフト 準拠かに関係なく）RADIUS を設定するには、RADIUS サーバー デモンを実行するホストと、デバイスと共有する秘密テキストストリングを指定する必要があります。RADIUS ホストおよび秘密テキスト文字列を指定するには、**radius server** グローバル コンフィギュレーション コマンドを使用します。

RADIUS の設定方法

RADIUS サーバホストの識別

デバイスと通信するすべての RADIUS サーバーにこのような設定をグローバルに適用するには、**radius-server timeout**、**radius-server retransmit**、および **key string** という 3 つの固有なグローバル コンフィギュレーション コマンドを使用します。

既存のサーバ ホストを認証用にグループ化するため、AAA サーバー グループを使用するようにデバイスを設定できます。

RADIUS サーバ上でも、いくつかの値を設定する必要があります。device の IP アドレス、およびサーバと device の双方で共有するキー スtring などの設定値です。

サーバ単位で RADIUS サーバとの通信を設定するには、次の手順を実行します。

Before you begin

device 上にグローバルな機能とサーバ単位での機能（タイムアウト、再送信回数、およびキー コマンド）を設定した場合、サーバ単位で設定したタイムアウト、再送信回数、およびキーに関するコマンドは、グローバルに設定したタイムアウト、再送信回数、およびキーに関するコマンドを上書きします。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius server** *server name*
4. **address** {*ipv4* | *ipv6*} *ip address* { **auth-port** *port number* | **acct-port** *port number*}
5. **key string**
6. **retransmit** *value*
7. **timeout** *seconds*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。

	Command or Action	Purpose
ステップ 3	radius server <i>server name</i> Example: Device(config)# radius server <i>rsim</i>	RADIUS サーバ設定の名前を Protected Access Credential (PAC) のプロビジョニング用に指定し、RADIUS サーバ設定モードを開始します。
ステップ 4	address { <i>ipv4 ipv6</i> } <i>ip address</i> { auth-port <i>port number</i> acct-port <i>port number</i> } Example: Device(config-radius-server)# address <i>ipv4</i> <i>124.2.2.12</i> auth-port <i>1612</i>	(任意) RADIUS サーバーのパラメータを指定します。 auth-port <i>port-number</i> には、認証要求の UDP 宛先ポートを指定します。デフォルトは 1645 です。指定できる範囲は 0 ~ 65536 です。 acct-port <i>port-number</i> には、認証要求の UDP 宛先ポートを指定します。デフォルトは 1646 です。
ステップ 5	key string Example: Device(config-radius-server)# key <i>rad123</i>	(任意) key string には、デバイスと RADIUS サーバーで動作する RADIUS デーモン間で使用される認証と暗号キーを指定します。 Note キーは、RADIUS サーバーで使用する暗号化キーに一致するテキストストリングでなければなりません。必ず radius server コマンドの最終項目としてキーを設定してください。先頭のスペースは無視されますが、キーの中間および末尾のスペースは使用されます。キーにスペースを使用する場合は、引用符がキーの一部である場合を除き、引用符でキーを囲まないでください。
ステップ 6	retransmit value Example: Device(config-radius-server)# retransmit <i>10</i>	(任意) サーバが応答しない、または応答が遅い場合に、RADIUS 要求をリセットする回数を指定します。指定できる範囲は 1 ~ 100 です。この設定は、 radius-server retransmit グローバル コンフィギュレーション コマンドによる設定を上書きします。
ステップ 7	timeout seconds Example: Device(config-radius-server)# timeout <i>60</i>	(任意) device が要求を再送信する前に RADIUS サーバからの応答を待機する時間間隔を指定します。指定できる範囲は 1 ~ 1000 です。この設定は、 radius-server timeout グローバル コンフィギュレーション コマンドによる設定を上書きします。
ステップ 8	end Example:	RADIUS サーバー コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

	Command or Action	Purpose
	Device (config-radius-server) # end	

RADIUS ログイン認証の設定

RADIUS ログイン認証を設定するには、次の手順を実行します。

Before you begin

AAA 方式を使用して HTTP アクセスに対しデバイスのセキュリティを確保するには、**ip http authentication aaa** グローバル コンフィギュレーション コマンドを設定する必要があります。AAA 認証を設定しても、AAA 方式を使用した HTTP アクセスに対しデバイスのセキュリティは確保されません。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login {default | list-name} method1 [method2...]**
5. **line [console | tty | vty] line-number [ending-line-number]**
6. **login authentication {default | list-name}**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model Example: Device (config) # aaa new-model	AAA をイネーブルにします。
ステップ 4	aaa authentication login {default list-name} method1 [method2...] Example:	ログイン認証方式リストを作成します。 • login authentication コマンドにリストが指定されていない場合に使用するデフォルトのリスト

Command or Action	Purpose
<pre>Device(config)# aaa authentication login default local</pre>	<p>を作成するには、default キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。</p> <ul style="list-style-type: none"> • <i>list-name</i> には、作成するリストの名前として使用する文字列を指定します。 • <i>method1...</i> には、認証アルゴリズムが試行する実際の方式を指定します。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。 <p>次のいずれかの方式を選択します。</p> <ul style="list-style-type: none"> • <i>enable</i> : イネーブルパスワードを認証に使用します。この認証方式を使用するには、あらかじめ enable password グローバルコンフィギュレーションコマンドを使用してイネーブルパスワードを定義しておく必要があります。 • <i>group radius</i> : RADIUS 認証を使用します。この認証方式を使用するには、あらかじめ RADIUS サーバーを設定しておく必要があります。 • <i>line</i> : 回線パスワードを認証に使用します。この認証方式を使用するには、あらかじめ回線パスワードを定義しておく必要があります。 password password ラインコンフィギュレーションコマンドを使用します。 • <i>local</i> : ローカルユーザー名データベースを認証に使用します。データベースにユーザー名情報を入力しておく必要があります。 username name password グローバルコンフィギュレーションコマンドを使用します。 • <i>local-case</i> : 大文字と小文字が区別されるローカルユーザー名データベースを認証に使用します。 username password グローバルコンフィギュレーションコマンドを使用して、ユーザー名情報をデータベースに入力する必要があります。

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>none</i> : ログインに認証を使用しません。
ステップ 5	line [console tty vty] <i>line-number</i> [<i>ending-line-number</i>] Example: Device(config)# line 1 4	ライン コンフィギュレーション モードを開始し、認証リストを適用する回線を設定します。
ステップ 6	login authentication {default <i>list-name</i> } Example: Device(config-line)# login authentication default	1 つの回線または複数回線に認証リストを適用します。 <ul style="list-style-type: none"> • default を指定する場合は、aaa authentication login コマンドで作成したデフォルトのリストを使用します。 • <i>list-name</i> には、aaa authentication login コマンドで作成したリストを指定します。
ステップ 7	end Example: Device(config-line)# end	ラインコンフィギュレーションモードを終了して、特権 EXEC モードを開始します。

AAA サーバグループの定義

定義したグループサーバに特定のサーバを対応付けるには、**server** グループ サーバ コンフィギュレーション コマンドを使用します。サーバを IP アドレスで特定することもできますし、任意指定の **auth-port** および **acct-port** キーワードを使用して複数のホストインスタンスまたはエントリを特定することもできます。

AAA サーバグループを定義するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius server** *name*
4. **address** {**ipv4** | **ipv6**} {*ip-address* | *hostname*} **auth-port** *port-number* **acct-port** *port-number*
5. **key** *string*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example:	特権 EXEC モードを有効にします。

	Command or Action	Purpose
	Device> enable	プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	radius server name Example: Device(config)# radius server ISE	RADIUS サーバの設定の名前を Protected Access Credential (PAC) のプロビジョニング用に指定し、RADIUS サーバ設定モードを開始します。 deviceは、IPv6 対応の RADIUS をサポートしています。
ステップ 4	address {ipv4 ipv6} {ip-address hostname} auth-port port-number acct-port port-number Example: Device(config-radius-server)# address ipv4 10.1.1.1 auth-port 1645 acct-port 1646	RADIUS サーバのアカウントingおよび認証パラメータの IPv4 アドレスを設定します。
ステップ 5	key string Example: Device(config-radius-server)# key cisco123	デバイスと RADIUS サーバとの間におけるすべての RADIUS 通信用の認証および暗号キーを指定します。
ステップ 6	end Example: Device(config-radius-server)# end	RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ユーザイネーブル アクセスおよびネットワーク サービスに関する RADIUS 許可の設定



Note 許可が設定されていても、CLI を使用してログインし、認証されたユーザに対しては、許可は省略されます。

ユーザ特権アクセスおよびネットワーク サービスに関する RADIUS 許可を設定するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authorization network authorization-listradius**

4. `aaa authorization exec authorization-listradius`
5. `end`

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa authorization network authorization-listradius Example: Device(config)# aaa authorization network list1 radius	ネットワーク関連のすべてのサービス要求に対して、ユーザが RADIUS 許可を受けるように device を設定します。
ステップ 4	aaa authorization exec authorization-listradius Example: Device(config)# aaa authorization exec list1 radius	ユーザに特権 EXEC のアクセス権限がある場合、ユーザが RADIUS 許可を受けるように device を設定します。 exec キーワードを指定すると、ユーザープロファイル情報 (autocommand 情報など) が返される場合があります。
ステップ 5	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

What to do next

aaa authorization グローバル コンフィギュレーション コマンドと **radius** キーワードを使用すると、ユーザのネットワーク アクセスを特権 EXEC モードに制限するパラメータを設定できます。

aaa authorization exec radius local コマンドは、次の許可パラメータを設定します。

- RADIUS を使用して認証を行った場合は、RADIUS を使用して特権 EXEC アクセスを許可します。
- 認証に RADIUS を使用しなかった場合は、ローカル データベースを使用します。

RADIUS アカウンティングの起動

RADIUS アカウンティングを開始するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting network *accounting-list*start-stop radius**
4. **aaa accounting exec *accounting-list*start-stop radius**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa accounting network <i>accounting-list</i>start-stop radius Example: Device(config)# aaa accounting network start-stop radius	ネットワーク関連のあらゆるサービス要求に関して、RADIUS アカウンティングをイネーブルにします。
ステップ 4	aaa accounting exec <i>accounting-list</i>start-stop radius Example: Device(config)# aaa accounting exec acc-list start-stop radius	RADIUS アカウンティングをイネーブルにして、特権 EXEC プロセスの最初に記録開始アカウンティング通知、最後に記録停止通知を送信します。
ステップ 5	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

すべての RADIUS サーバの設定

すべての RADIUS サーバーを設定するには、特権 EXEC モードで次の手順を実行します。

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **radius server** *server name*
4. **key** *string*
5. **retransmit** *retries*
6. **timeout** *seconds*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	radius server <i>server name</i> Example: Device(config)# radius server <i>rsim</i>	RADIUS サーバ設定の名前を Protected Access Credential (PAC) のプロビジョニング用に指定し、RADIUS サーバ設定モードを開始します。
ステップ 4	key <i>string</i> Example: Device(config-radius-server)# key <i>your_server_key</i>	スイッチとすべての RADIUS サーバ間で共有されるシークレット テキスト ストリングを指定します。 Note キーは、RADIUS サーバで使用する暗号化キーに一致するテキスト ストリングでなければなりません。先頭のスペースは無視されますが、キーの中間および末尾のスペースは使用されます。キーにスペースを使用する場合は、引用符がキーの一部である場合を除き、引用符でキーを囲まないでください。
ステップ 5	retransmit <i>retries</i> Example: Device(config-radius-server)# retransmit 5	スイッチが RADIUS 要求をサーバに再送信する回数を指定します。デフォルトは 3 です。指定できる範囲は 1 ~ 1000 です。
ステップ 6	timeout <i>seconds</i> Example: Device(config-radius-server)# timeout 3	スイッチが RADIUS 要求に対する応答を待って、要求を再送信するまでの時間 (秒) を指定します。デフォルトは 5 秒です。指定できる範囲は 1 ~ 1000 です。
ステップ 7	end Example: Device(config-radius-server)# end	RADIUS サーバー コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

ベンダー固有の RADIUS 属性を使用するデバイスの設定

ベンダー固有の RADIUS 属性を設定するには、次の手順を実行します。

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `radius-server vsa send [accounting | authentication]`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	<code>enable</code> Example: Device> <code>enable</code>	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	<code>configure terminal</code> Example: Device# <code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<code>radius-server vsa send [accounting authentication]</code> Example: Device(config)# <code>radius-server vsa send accounting</code>	device が VSA (RADIUS IETF 属性 26 で定義) を認識して使用できるようにします。 <ul style="list-style-type: none"> • (任意) 認識されるベンダー固有属性の集合を アカウンティング属性だけに限定するには、accounting キーワードを使用します。 • (任意) 認識されるベンダー固有属性の集合を 認証属性だけに限定するには、authentication キーワードを使用します。 キーワードを指定せずにこのコマンドを入力すると、アカウンティングおよび認証のベンダー固有属性の両方が使用されます。
ステップ 4	<code>end</code> Example: Device(config)# <code>end</code>	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

ベンダー独自仕様の RADIUS サーバ通信に関するデバイスの設定

ベンダー独自仕様の RADIUS サーバ通信を設定するには、次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius server** *server name*
4. **address** { **ipv4** | **ipv6** } *ip address*
5. **non-standard**
6. **key** *string*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	radius server <i>server name</i> Example: Device(config)# radius server <i>rsim</i>	RADIUS サーバ設定の名前を Protected Access Credential (PAC) のプロビジョニング用に指定し、RADIUS サーバ設定モードを開始します。
ステップ 4	address { ipv4 ipv6 } <i>ip address</i> Example: Device(config-radius-server)# address ipv4 <i>172.24.25.10</i>	(任意) RADIUS サーバの IP アドレスを指定します。
ステップ 5	non-standard Example: Device(config-radius-server)# non-standard	RADIUS サーバが RADIUS ベンダー独自の実装を使用していることを示します。
ステップ 6	key <i>string</i> Example: Device(config-radius-server)# key <i>rad123</i>	デバイスとベンダー独自仕様の RADIUS サーバーとの間で使用される共有秘密テキスト文字列を指定します。デバイスと RADIUS サーバーはこのテキスト文字列を使用してパスワードを暗号化し、応答を交換します。
ステップ 7	end Example:	RADIUS サーバーモードを終了し、特権 EXEC モードを開始します。

	Command or Action	Purpose
	Device (config-radius-server) # end	

デバイス上での CoA の設定

CoA を device で設定するには、次の手順を実行します。この手順は必須です。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa server radius dynamic-author**
5. **client** {*ip-address* | *name*} [*vrf vrfname*] [*server-key string*]
6. **server-key** [0 | 7] *string*
7. **port** *port-number*
8. **auth-type** {*any* | *all* | *session-key*}
9. **ignore server-key**
10. **exit**
11. **authentication command bounce-port ignore**
12. **authentication command disable-port ignore**
13. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model Example: Device (config) # aaa new-model	AAA をイネーブルにします。
ステップ 4	aaa server radius dynamic-author Example: Device (config) # aaa server radius dynamic-author	デバイスを認証、許可、アカウントिंग (AAA) サーバーとして設定して外部ポリシーサーバーとの通信を容易にし、ダイナミック許可ローカルサーバー 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 5	client { <i>ip-address</i> <i>name</i> } [vrf <i>vrfname</i>] [server-key <i>string</i>] Example: Device (config-locsvr-da-radius)# client client1 vrf vrf1	デバイスが CoA を受け取り、要求を取り外す RADIUS クライアントを指定します。
ステップ 6	server-key [0 7] <i>string</i> Example: Device (config-locsvr-da-radius)# server-key your_server_key	RADIUS キーをデバイスと RADIUS クライアントとの間で共有されるように設定します。
ステップ 7	port <i>port-number</i> Example: Device (config-locsvr-da-radius)# port 25	設定された RADIUS クライアントから RADIUS 要求をデバイスが受信するポートを指定します。
ステップ 8	auth-type { <i>any</i> <i>all</i> <i>session-key</i> } Example: Device (config-locsvr-da-radius)# auth-type any	deviceが RADIUS クライアントに使用する許可のタイプを指定します。 クライアントは、許可用に設定されたすべての属性と一致していなければなりません。
ステップ 9	ignore server-key Example: Device (config-locsvr-da-radius)# ignore server-key	(任意) サーバーキーを無視するように device を設定します。
ステップ 10	exit Example: Device (config-locsvr-da-radius)# exit	ダイナミック認可ローカル サーバー 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 11	authentication command bounce-port ignore Example: Device (config)# authentication command bounce-port ignore	(任意) CoA 要求を無視して、セッションをホスティングするポートを一時的にディセーブルにするように device を設定します。ポートを一時的にディセーブルにする目的は、VLAN の変更が発生しても、その変更を検出するサブリカントがエンドポイント上にない場合に、ホストから DHCP 再ネゴシエーションを行わせることです。

	Command or Action	Purpose
ステップ 12	authentication command disable-port ignore Example: Device(config)# authentication command disable-port ignore	(任意) セッションをホスティングしているポートを管理上のシャットダウン状態にするよう要求する非標準コマンドを無視するようにdeviceを設定します。ポートをシャットダウンすると、セッションが終了します。 ポートを再びイネーブルにするには、標準の CLI または SNMP コマンドを使用します。
ステップ 13	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

RADIUS サーバグループの RADIUS ソースインターフェイスの設定

RADIUS source-interface は、次のいずれかの方法で RADIUS サーバグループの下に設定できます。

- **ip radius source-interface interface-name** コマンドを使用して、RADIUS サーバグループの下に RADIUS source-interface を設定します。
- **vrf vrf-name** コマンドを使用して、RADIUS サーバグループの下に VRF を設定してから **ip radius source interface interface-name vrf vrf-name** コマンドを使用して、設定した VRF を source-interface にグローバルに関連付けます。

両方のメソッドが設定されている場合、サーバグループ設定の下ではソースインターフェイスが優先されます。

RADIUS サーバグループの下で RADIUS source-interface を設定するには、次の手順を実行します。

始める前に

VRF ルーティングテーブルを設定し、VRF をインターフェイスに関連付ける必要があります

手順の概要

1. **enable**
2. **configure terminal**
3. **{ ip | ipv6 } radius source-interface interface-number vrf vrf-name**
4. **aaa group server radius group_name**
5. **ip vrf forwarding vrf-name**
6. **{ ip | ipv6 } radius source-interface interface-number**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	{ ip ipv6 } radius source-interface interface-number vrf vrf-name 例： Device(config)# ip radius source-interface GigabitEthernet1/0/23 vrf vrf17	すべての発信 RADIUS パケットに対して、RADIUS に、指定されたインターフェイスの IP アドレスを強制的に使用させ、Per VRF に基づいて仕様を有効にします。 <ul style="list-style-type: none"> • interface-name : RADIUS がすべての発信パケットに使用するインターフェイスの名前を指定します。 • vrf vrf-name : VRF ごとの設定を指定します。
ステップ 4	aaa group server radius group_name 例： Device(config-sg-radius)# aa group server radius rad-grp	異なる RADIUS サーバホストを別々のリストと方式にグループ化し、 server-group 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 vrf forwarding vrf-name 例： Device(config-sg-radius)# ip vrf forwarding vrf17	(任意) インターフェイスに VRF を設定します。
ステップ 6	{ ip ipv6 } radius source-interface interface-number 例： Device(config-sg-radius)# ip radius source-interface loopback0	(任意) RADIUS グループサーバからのすべての発信 RADIUS パケットに対して、RADIUS に、指定されたインターフェイスの IP アドレスを強制的に使用させます。 <i>interface-name</i> : RADIUS がすべての発信パケットに使用するインターフェイスの名前を指定します。
ステップ 7	end 例： Device(config-sg-radius)# end	特権 EXEC モードに戻ります。

CoA 機能のモニタリング

表 15: 特権 EXEC 表示コマンド

コマンド	目的
show aaa attributes protocol radius	RADIUS コマンドの AAA 属性を表示します。

表 16: グローバルトラブルシューティング コマンド

コマンド	目的
debug radius	RADIUS のトラブルシューティングを行うための情報を表示します。
debug aaa coa	CoA 処理のトラブルシューティングを行うための情報を表示します。
debug aaa pod	POD パケットのトラブルシューティングを行うための情報を表示します。
debug aaa subsys	POD パケットのトラブルシューティングを行うための情報を表示します。
debug cmdhd[detail error events]	コマンドヘッダーのトラブルシューティングを行うための情報を表示します。

RADIUS の機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	RADIUS	RADIUS は、不正なアクセスからネットワークのセキュリティを保護する分散クライアント/サーバシステムです。RADIUS クライアントは、サポート対象のシスコデバイス上で稼働します。クライアントは中央の RADIUS サーバに認証要求を送ります。中央の RADIUS サーバにはすべてのユーザ認証情報、ネットワーク サービスアクセス情報が登録されています。

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



第 8 章

RadSec の設定

この章では、RadSec over Transport Layer Security (TLS) および Datagram Transport Layer Security (DTLS) サーバーを設定する方法について説明します。

- [RadSec の設定に関する制限事項 \(197 ページ\)](#)
- [RadSec に関する情報 \(197 ページ\)](#)
- [RadSec の設定方法 \(198 ページ\)](#)
- [RadSec のモニタリング \(203 ページ\)](#)
- [RadSec の設定例 \(203 ページ\)](#)
- [RadSec 設定の機能履歴 \(204 ページ\)](#)

RadSec の設定に関する制限事項

RadSec 機能には、次のような制限事項が適用されます。

- RADIUS クライアントは、エフェメラルポートを送信元ポートとして使用します。この送信元ポートは、UDP、Datagram Transport Layer Security (DTLS)、および Transport Layer Security (TLS) に同時に使用できません。
- 設定の制限はありませんが、AAA サーバグループ下のサーバーに同じタイプ (TLS のみまたは DTLS のみ) を使用することを推奨します。
- RadSec は、IPv4 接続でのみサポートされます。

RadSec に関する情報

RadSec は、安全なトンネルを介して転送される RADIUS サーバー上で暗号化サービスを提供します。RadSec over TLS および DTLS は、クライアントサーバとデバイスサーバーの両方に実装されています。クライアント側が RADIUS AAA を制御するのに対し、デバイス側は認可変更 (CoA) を制御します。

次のパラメータを設定できます:

- 個々のクライアント固有のアイドルタイムアウト、クライアントトラストポイント、およびサーバートラストポイント。
- グローバル CoA 固有の TLS または DTLS リスニングポートおよび対応するソースインターフェイスのリスト。



(注) 特定のサーバに対して TLS または DTLS を無効にするには、RADIUS サーバの設定モードで **no tls** または **no dtls** コマンドを使用します。

RadSec の設定方法

次のセクションでは、RadSec の設定を構成するさまざまな作業について説明します。

RadSec over TLS の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **radius server radius-server-name**
4. **tls [connectiontimeout connection-timeout-value] [idletimeout idle-timeout-value] [ip {radius source-interface interface-name |vrf forwarding forwarding-table-name}] [port port-number] [retries number-of-connection-retries] [trustpoint {client trustpoint name | server trustpoint name}]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	radius server radius-server-name 例： Device(config)# radius server R1	RADIUS サーバ設定の名前を Protected Access Credential (PAC) のプロビジョニング用に指定し、RADIUS サーバ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 4	<p>tls [connectiontimeout <i>connection-timeout-value</i>] [idletimeout <i>idle-timeout-value</i>] [ip {radius source-interface <i>interface-name</i> vrf forwarding forwarding-table-name}] [port <i>port-number</i>] [retries <i>number-of-connection-retries</i>] [trustpoint {client <i>trustpoint name</i> server <i>trustpoint name</i>}]</p> <p>例 :</p> <pre>Device(config-radius-server)# tls connectiontimeout 10 Device(config-radius-server)# tls idletimeout 75 Device(config-radius-server)# tls retries 15 Device(config-radius-server)# tls ip radius source-interface GigabitEthernet 1/0/1 Device(config-radius-server)# tls ip vrf forwarding table-1 Device(config-radius-server)# tls port 10 Device(config-radius-server)# tls trustpoint client TP-self-signed-721943660 Device(config-radius-server)# tls trustpoint server isetp</pre>	<p>TLS パラメータを設定します。次のパラメータを設定できます:</p> <ul style="list-style-type: none"> • connectiontimeout : TLS 接続タイムアウト値を設定します。デフォルトは 5 秒です。 • idletimeout : TLS アイドルタイムアウト値を設定します。デフォルトは 60 秒です。 • ip : IP 送信元パラメータを設定します。 • port : TLS ポート番号を設定します。デフォルトは 2083 です。 • retries : TLS 接続再試行の回数を設定します。デフォルトは 5 分です。 • trustpoint : クライアントとサーバーに TLS トラストポイントを設定します。クライアントとサーバーの TLS トラストポイントが同じ場合、トラストポイント名も両方で同じである必要があります。
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device(config-radius-server)# end</pre>	<p>RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

TLS CoA の動的認可の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa server radius dynamic-author**
4. **client** {**ip-addr** | **hostname**} [**tls** [**client-tp** *client-tp-name*] [**idletimeout** *idletimeout-interval*] [**server-tp** *server-tp-name*] | **vrf** *vrf-id*]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <ul style="list-style-type: none"> • プロンプトが表示されたらパスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa server radius dynamic-author 例： Device(config)# aaa server radius dynamic-author	ダイナミック認証ローカル サーバー コンフィギュレーション モードを入力し、デバイスが認可変更 (CoA) を受け入れ、要求を取り外す RADIUS クライアントを指定します。デバイスを AAA サーバーとして設定し、外部ポリシーサーバーとの連携を促進します。
ステップ 4	client {ip-addr hostname} [tls [client-tp client-tp-name] [idletimeout idletimeout-interval] [server-tp server-tp-name] vrf vrf-id] 例： Device(config-locsvr-da-radius)# client 10.104.49.14 tls idletimeout 100 client-tp tls_ise server-tp tls_client	AAA サーバー クライアントの IP アドレスまたはホスト名を設定します。次のオプションのパラメータを設定できます。 <ul style="list-style-type: none"> • tls : クライアントの TLS を有効にします。 • client-tp : クライアントのトラストポイントを設定します。 • idletimeout : TLS アイドルタイムアウト値を設定します。 • server-tp : サーバーのトラストポイントを設定します。 • vrf : クライアントの仮想ルーティングおよび転送 (VRF) ID を設定します。
ステップ 5	end 例： Device(config-locsvr-da-radius)# end	ダイナミック認証ローカル サーバー コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

RadSec over DTLS の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **radius server radius-server-name**
4. **dtls [connectiontimeout connection-timeout-value] [idletimeout idle-timeout-value] [ip {radius source-interface interface-name | vrf forwarding forwarding-table-name}] [port port-number] [retries number-of-connection-retries] [trustpoint {client trustpoint name | server trustpoint name}]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	radius server radius-server-name 例： Device(config)# radius server R1	RADIUS サーバー設定の名前を Protected Access Credential (PAC) のプロビジョニング用に指定し、RADIUS サーバー設定モードを開始します。
ステップ 4	dtls [connectiontimeout connection-timeout-value] [idletimeout idle-timeout-value] [ip {radius source-interface interface-name vrf forwarding forwarding-table-name}] [port port-number] [retries number-of-connection-retries] [trustpoint {client trustpoint name server trustpoint name}] 例： Device(config-radius-server)# dtls connectiontimeout 10 Device(config-radius-server)# dtls idletimeout 75 Device(config-radius-server)# dtls retries 15 Device(config-radius-server)# dtls ip radius source-interface GigabitEthernet 1/0/1 Device(config-radius-server)# dtls ip vrf forwarding table-1 Device(config-radius-server)# dtls port 10 Device(config-radius-server)# dtls trustpoint client TP-self-signed-721943660 Device(config-radius-server)# dtls trustpoint server isetp	DTLS パラメータを設定します。次のパラメータを設定できます。 <ul style="list-style-type: none"> • connectiontimeout : DTLS 接続タイムアウト値を設定します。デフォルトは 5 秒です。 • idletimeout : DTLS アイドルタイムアウト値を設定します。デフォルトは 60 秒です。 • ip : IP 送信元パラメータを設定します。 • port : DTLS ポート番号を設定します。デフォルトは 2083 です。 • retries : DTLS 接続再試行の回数を設定します。デフォルトは 5 分です。 • trustpoint : クライアントとサーバーに DTLS トラストポイントを設定します。クライアントとサーバーの DTLS トラストポイントが同じ場合、トラストポイント名も両方で同じである必要があります。
ステップ 5	end 例： Device(config-radius-server)# end	RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

DTLS CoA の動的認可の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa server radius dynamic-author**
4. **client {ip-addr | hostname} [dtls [client-tp client-tp-name] [idletimeout idletimeout-interval] [server-tp server-tp-name] | vrf vrf-id]**
5. **dtls {ip radius source-interface interface-name | port radius-dtls-server-port-number}**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa server radius dynamic-author 例： Device(config)# aaa server radius dynamic-author	ダイナミック認可ローカル サーバー コンフィギュレーション モードを開始し、デバイスが認可変更 (CoA) を受け入れ、要求を取り外す RADIUS クライアントを指定します。デバイスを AAA サーバーとして設定し、外部ポリシーサーバーとの連携を促進します。
ステップ 4	client {ip-addr hostname} [dtls [client-tp client-tp-name] [idletimeout idletimeout-interval] [server-tp server-tp-name] vrf vrf-id] 例： Device(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 100 client-tp dtls_ise server-tp dtls_client	AAA サーバー クライアントの IP アドレスまたはホスト名を設定します。次のオプションのパラメータを設定できます。 • dtls : クライアントの DTLS を有効にします。 • client-tp : クライアントのトラストポイントを設定します。 • idletimeout : DTLS アイドルタイムアウト値を設定します。 • server-tp : サーバーのトラストポイントを設定します。 • vrf : クライアントの仮想ルーティングおよび転送 (VRF) ID を設定します。

	コマンドまたはアクション	目的
ステップ 5	dtls {ip radius source-interface interface-name port radius-dtls-server-port-number} 例 : Device(config-locsvr-da-radius)# dtls ip radius source-interface GigabitEthernet 1/0/24 Device(config-locsvr-da-radius)# dtls port 100	RADIUS CoA サーバーを設定します。次のパラメータを設定できます: <ul style="list-style-type: none"> • ip radius source-interface interface-name : RADIUS CoA サーバーの送信元アドレスのインターフェイスを指定します。 • port radius-dtls-server-port-number : ローカル DTLS RADIUS サーバーがリスンするポートを指定します。
ステップ 6	end 例 : Device(config-locsvr-da-radius)# end	ダイナミック認証ローカル サーバー コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

RadSec のモニタリング

次のコマンドを使用して、TLS および DTLS サーバーの統計を監視します。

表 17: TLS および DTLS サーバー統計コマンドの監視

コマンド	目的
show aaa servers	TLS および DTLS サーバーに関連する情報を表示します。
clear aaa counters servers radius {server id all}	RADIUS TLS 固有または DTLS 固有の統計情報をクリアします。
debug radius radsec	RADIUS RadSec デバッグを有効にします。

RadSec の設定例

次の例は、RadSec の設定を理解するのに役立ちます。

例 : RadSec over TLS の設定

```
Device> enable
Device# configure terminal
Device(config)# radius server R1
Device(config-radius-server)# tls connectiontimeout 10
Device(config-radius-server)# tls idletimeout 75
Device(config-radius-server)# tls retries 15
```

例 : TLS CoA の動的認可の設定

```

Device(config-radius-server) # tls ip radius source-interface GigabitEthernet 1/0/1
Device(config-radius-server) # tls ip vrf forwarding table-1
Device(config-radius-server) # tls port 10
Device(config-radius-server) # tls trustpoint client TP-self-signed-721943660
Device(config-radius-server) # tls trustpoint server isetp
Device(config-radius-server) # end

```

例 : TLS CoA の動的認可の設定

```

Device> enable
Device# configure terminal
Device(config)# aaa server radius dynamic-author
Device(config-locsvr-da-radius)# client 10.104.49.14 tls idletimeout 100
client-tp tls_ise server-tp tls_client
Device(config-locsvr-da-radius)# dtls port 100
Device(config-locsvr-da-radius)# end

```

例 : RadSec over DTLS の設定

```

Device> enable
Device# configure terminal
Device(config)# radius server R1
Device(config-radius-server)# dtls connectiontimeout 10
Device(config-radius-server)# dtls idletimeout 75
Device(config-radius-server)# dtls retries 15
Device(config-radius-server)# dtls ip radius source-interface GigabitEthernet 1/0/1
Device(config-radius-server)# dtls ip vrf forwarding table-1
Device(config-radius-server)# dtls port 10
Device(config-radius-server)# dtls trustpoint client TP-self-signed-721943660
Device(config-radius-server)# dtls trustpoint server isetp
Device(config-radius-server)# end

```

例 : DTLS CoA の動的認可の設定

```

Device> enable
Device# configure terminal
Device(config)# aaa server radius dynamic-author
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 100
client-tp dtls_ise server-tp dtls_client
Device(config-locsvr-da-radius)# dtls ip radius source-interface GigabitEthernet 1/0/24
Device(config-locsvr-da-radius)# dtls port 100
Device(config-locsvr-da-radius)# end

```

RadSec 設定の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.6.1	RadSec over DTLS の設定	RadSec over DTLS は、安全なトンネルを介して転送される RADIUS サーバー上で暗号化サービスを提供します。
Cisco IOS XE Fuji 16.9.1	RadSec over TLS の設定	RadSec over TLS は、安全なトンネルを介して転送される RADIUS サーバー上で暗号化サービスを提供します。

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



第 9 章

Kerberos の設定

- [Kerberos の前提条件](#) (207 ページ)
- [Kerberos に関する情報](#) (207 ページ)
- [Kerberos を設定する方法, on page 211](#)
- [Kerberos 設定の監視](#) (212 ページ)
- [Kerberos の機能履歴](#) (212 ページ)

Kerberos の前提条件

次に、Kerberos を使用してスイッチ アクセスを制御するための前提条件を示します。

- リモート ユーザーがネットワーク サービスに対して認証を得るには、Kerberos レルム内のホストと KDC を設定し、ユーザーとネットワーク サービスの両方に通信を行い、相互に認証させる必要があります。これを実現するには、互いの識別が必要です。KDC 上の Kerberos データベースにホストのエントリを追加し、Kerberos レルム内のすべてのホストに KDC が生成した KEYTAB ファイルを追加します。また、KDC データベースにユーザー用のエントリも作成します。
- Kerberos サーバーには、ネットワーク セキュリティ サーバーとして設定されていて、Kerberos プロトコルを用いてユーザーを認証できるスイッチを使用できます。

ホストおよびユーザーのエントリを追加または作成する場合の注意事項は次のとおりです。

- Kerberos プリンシパル名はすべて小文字でなければなりません。
- Kerberos インスタンス名はすべて小文字でなければなりません。
- Kerberos レルム名はすべて大文字でなければなりません。

Kerberos に関する情報

ここでは、以下の Kerberos に関する情報を提供します。

Kerberos とスイッチ アクセス

ここでは、Kerberos セキュリティ システムをイネーブルにして設定する方法について説明します。Kerberos セキュリティ システムは、信頼できるサードパーティを使用してネットワーク リソースに対する要求を認証します。



Note Kerberos の設定例では、信頼できるサードパーティを、Kerberos をサポートし、ネットワーク セキュリティ サーバーとして設定され、Kerberos プロトコルを使用してユーザーを認証するスイッチとすることができます。

Kerberos の概要

Kerberos はマサチューセッツ工科大学 (MIT) が開発した秘密キーによるネットワーク 認証 プロトコルです。データ暗号規格 (DES) という暗号化アルゴリズムを暗号化と認証に使用し、ネットワーク リソースに対する要求を認証します。Kerberos は、信頼できるサードパーティという概念を使ってユーザーとサービスに対してセキュリティの検証を実行します。この信頼できるサードパーティをキー発行局 (KDC) と呼びます。

Kerberos は、ユーザーが誰であるか、そのユーザーが使用しているネットワーク サービスは何であるかを検証します。これを実行するために、KDC (つまり信頼できる Kerberos サーバー) がユーザーにチケットを発行します。これらのチケットには有効期限があり、ユーザー クレデンシャルのキャッシュに保存されます。Kerberos サーバーは、ユーザー名やパスワードの代わりにチケットを使ってユーザーとネットワーク サービスを認証します。



Note Kerberos サーバーには、ネットワーク セキュリティ サーバーとして設定されていて、Kerberos プロトコルを用いてユーザーを認証できるのであれば、どのスイッチも使用できます。

Kerberos のクレデンシャル発行スキームでは、*single logon* という手順を使用します。この手順では、ユーザーを 1 回認証すると、ユーザー クレデンシャルが有効な間は (他のパスワードの暗号化を行わずに) セキュア認証が可能になります。

このソフトウェア リリースは Kerberos 5 に対応しています。Kerberos 5 では、すでに Kerberos 5 を使用している組織が、(UNIX サーバーや PC などの) 他のネットワーク ホストが使用している KDC 上の Kerberos 認証データベースを使用できます。

Kerberos は次のネットワーク サービスをサポートしています。

- Telnet
- rlogin
- rsh

次の表に、一般的な Kerberos 関連用語とその定義を示します。

Table 18: Kerberos の用語

用語	定義
認証	ユーザーやサービスが他のサービスに対して自分自身の身元を証明する手順。たとえば、クライアントはスイッチに対して認証を得て、スイッチは他のスイッチに対して認証を得ます。
許可	ユーザーがネットワークやスイッチにおいてどのような権限を有しており、またどのような動作を実行できるかを、スイッチが識別する手段
クレデンシヤル	認証チケット (TSG ² 、サービスクレデンシヤルなど) を表す総称。Kerberos クレデンシヤルで、ユーザーまたはサービスの ID を検証します。ネットワーク サービスがチケットを発行した Kerberos サーバーを信頼することにした場合、ユーザー名やパスワードを再入力する代わりにこれを使用できます。証明書の有効期限は、8 時間がデフォルトの設定です。
インスタンス	<p>Kerberos プリンシパルの承認レベル ラベル。ほとんどの Kerberos プリンシパルは、<code>user@REALM</code> という形式です (たとえば、<code>smith@EXAMPLE.COM</code>)。Kerberos インスタンスのある Kerberos プリンシパルは、<code>user/instance@REALM</code> という形式です (たとえば、<code>smith/admin@EXAMPLE.COM</code>)。Kerberos インスタンスは、認証が成功した場合のユーザーの承認レベルを指定するために使用できます。各ネットワーク サービスのサーバーは、Kerberos インスタンスの許可マッピングを適用し実行できますが、必須ではありません。</p> <p>Note Kerberos プリンシパル名およびインスタンス名はすべて小文字でなければなりません。</p> <p>Note Kerberos レルム名はすべて大文字でなければなりません。</p>
KDC ³	ネットワーク ホストで稼働する Kerberos サーバーおよびデータベースプログラムで構成されるキー発行局
Kerberos 対応	Kerberos クレデンシヤルのインフラストラクチャをサポートするために変更されたアプリケーションやサービスのことを指す用語
Kerberos レルム	<p>Kerberos サーバーに登録されたユーザー、ホスト、およびネットワーク サービスで構成されるドメイン。Kerberos サーバーを信頼して、ユーザーまたはネットワーク サービスに対する別のユーザーまたはネットワーク サービスの ID を検証します。</p> <p>Note Kerberos レルム名はすべて大文字でなければなりません。</p>

用語	定義
Kerberos サーバー	ネットワーク ホストで稼働しているデーモン。ユーザーおよびネットワーク サービスはそれぞれ Kerberos サーバーに ID を登録します。ネットワーク サービスは Kerberos サーバーにクエリーを送信して、他のネットワーク サービスの認証を得ます。
KEYTAB ⁴	ネットワーク サービスが KDC と共有するパスワード。Kerberos 5 以降のバージョンでは、ネットワーク サービスは KEYTAB を使って暗号化されたサービス クレデンシヤルを暗号解除して認証します。Kerberos 5 よりも前のバージョンでは、KEYTAB は SRVTAB ⁵ と呼ばれます。
プリンシパル	Kerberos ID とも呼ばれ、Kerberos サーバーに基づき、ユーザーが誰であるか、サービスが何であるかを表します。 Note Kerberos プリンシパル名はすべて小文字でなければなりません。
サービス クレデンシヤル	ネットワーク サービスのクレデンシヤル。KDC からクレデンシヤルが発行されると、ネットワーク サービスと KDC が共有するパスワードで暗号化されます。ユーザー TGT ともパスワードを共有します。
SRVTAB	ネットワーク サービスが KDC と共有するパスワード。SRVTAB は、Kerberos 5 以降のバージョンでは KEYTAB と呼ばれています。
TGT	身分証明書のこと、KDC が認証済みユーザーに発行するクレデンシヤル。TGT を受け取ったユーザーは、KDC が示した Kerberos レルム内のネットワーク サービスに対して認証を得ることができます。

² チケット認可チケット

³ キー発行局

⁴ キー テーブル

⁵ サーバー テーブル

Kerberos の動作

Kerberos サーバーには、ネットワーク セキュリティサーバーとして設定されていて、Kerberos プロトコルを使用してリモートユーザーを認証できるデバイスを使用できます。Kerberos をカスタマイズする方法はいくつかありますが、ネットワーク サービスにアクセスしようとするリモートユーザーは、3つのセキュリティ レイヤを通過しないとネットワーク サービスにアクセスできません。

デバイスを Kerberos サーバーとして使用してネットワーク サービスで認証されるには、リモートユーザーは次の手順を実行する必要があります。

境界スイッチに対する認証の取得

ここでは、リモート ユーザーが通過しなければならない最初のセキュリティ レイヤについて説明します。ユーザーは、まず境界スイッチに対して認証を得なければなりません。リモート ユーザーが境界スイッチに対して認証を得る場合、次のプロセスが発生します。

1. ユーザーが境界スイッチに対して、Kerberos 未対応の Telnet 接続を開始します。
2. ユーザー名とパスワードの入力を求めるプロンプトをスイッチが表示します。
3. スイッチが、このユーザーの TGT を KDC に要求します。
4. KDC がユーザー ID を含む暗号化された TGT をスイッチに送信します。
5. スイッチは、ユーザーが入力したパスワードを使って TGT の暗号解除を試行します。
 - 暗号解除に成功した場合は、ユーザーはスイッチに対して認証を得ます。
 - 暗号解除に成功しない場合は、ユーザー名とパスワードを再入力 (Caps Lock または NumLock のオン/オフに注意) するか、別のユーザー名とパスワードを入力してステップ 2 の手順を繰り返します。

Kerberos 未対応の Telnet セッションを開始し、境界スイッチの認証を得ているリモート ユーザーはファイアウォールの内側にいますが、ネットワーク サービスにアクセスするには、KDC から直接認証を得る必要があります。ユーザーが KDC から認証を得なければならないのは、KDC が発行する TGT はスイッチに保存されており、ユーザーがこのスイッチにログオンしないかぎり、追加の認証に使用できないからです。

KDC からの TGT の取得

ここでは、リモート ユーザーが通過しなければならない 2 番めのセキュリティ レイヤについて説明します。ユーザーは、ネットワーク サービスにアクセスするために、このレイヤで KDC の認証を得て、KDC から TGT を取得しなければなりません。

ネットワーク サービスに対する認証の取得

ここでは、リモート ユーザーが通過しなければならない 3 番めのセキュリティ レイヤについて説明します。TGT を取得したユーザーは、このレイヤで Kerberos レalm内のネットワーク サービスに対して認証を得なければなりません。

Kerberos を設定する方法

Kerberos 認証済みサーバー/クライアント システムを設定する手順は、次のとおりです。

- Kerberos コマンドを使用して KDC を設定します。
- Kerberos プロトコルを使用するようにスイッチを設定します。

Kerberos 設定の監視

Kerberos 設定を表示するには、次のコマンドを使用します。

- **show running-config**
- **show kerberos creds** : 現在のユーザーの認定証キャッシュに含まれる認定証を一覧表示します。
- **clear kerberos creds** : 転送済みの認定証を含め、現在のユーザーの認定証キャッシュに含まれるすべての認定証を破棄します。

Kerberos の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	Kerberos	Kerberos はマサチューセッツ工科大学 (MIT) が開発した秘密キーによるネットワーク認証プロトコルです。データ暗号規格 (DES) という暗号化アルゴリズムを暗号化と認証に使用し、ネットワーク リソースに対する要求を認証します。Kerberos は、信頼できるサードパーティという概念を使ってユーザーとサービスに対してセキュリティの検証を実行します。

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



第 10 章

MACsec の暗号化

- [MACsec 暗号化の前提条件 \(213 ページ\)](#)
- [MACsec 暗号化の制約事項 \(213 ページ\)](#)
- [MACsec 暗号化について \(214 ページ\)](#)
- [MACsec 暗号化の設定方法 \(223 ページ\)](#)
- [MACsec 暗号化の設定例 \(251 ページ\)](#)
- [MACsec 暗号化に関する追加情報 \(272 ページ\)](#)
- [MACsec 暗号化の機能情報 \(273 ページ\)](#)

MACsec 暗号化の前提条件

証明書ベース MACsec の前提条件

- 認証局 (CA) サーバーがネットワークに設定されていることを確認します。
- CA 証明書を生成します。
- Cisco Identity Services Engine (ISE) リリース 2.0 が設定されていることを確認します。
- 両方の参加デバイス (CA サーバーと Cisco Identity Services Engine (ISE)) が Network Time Protocol (NTP) を使用して同期されていることを確認します。時間がすべてのデバイスで同期されていないと、証明書は検証されません。
- 802.1x 認証と AAA がデバイスに設定されていることを確認します。

MACsec 暗号化の制約事項

- C9300-48UXM および C9300-48UN スイッチモデルの MACsec は、最初の 16 のダウンリンク ネットワーク ポートとすべてのアップリンク ネットワーク モジュール ポートでのみサポートされます。MACsec は、C9300-48UXM および C9300-48UN スイッチモデルの最後の 32 個のダウンリンク ネットワーク ポートではサポートされません。

- MACsec 設定は、EtherChannel ポートではサポートされません。代わりに、EtherChannel の個々のメンバポートに MACsec 設定を適用できます。MACsec 設定を削除するには、最初に EtherChannel からメンバポートをバンドル解除してから、個々のメンバポートから削除する必要があります。
- 証明書ベースの MACsec は、アクセスセッションがクローズドとして、またはマルチホストモードで設定されている場合にのみサポートされます。他のコンフィギュレーションモードはサポートされていません。
- MACsec Key Agreement (MKA) はハイアベイラビリティではサポートされません。
- MKA を使用した MACsec は、ポイントツーポイントリンクでのみサポートされます。
- GCM-AES-256 および XPN 暗号スイート (GCM-AES-XPN-128 および GCM-AES-XPN-256) は、Network Advantage ライセンスでのみサポートされます。
- MACsec 暗号アナウンスメントは、MACsec 拡張パケット番号 (XPN) 暗号およびスイッチ間 MACsec 接続ではサポートされません。
- MACsec XPN 暗号スイートは、スイッチからホストへの MACsec 接続ではサポートされていません
- **dot1q tag vlan native** コマンドがグローバルレベルで設定されている場合、トランクポートでの dot1x 再認証は失敗します。
- MACsec XPN 暗号スイートは、機密性オフセットを含む機密性保護を提供しません。
- Precision Time Protocol (PTP) を備えた MACsec はサポートされません。
- MACsec は、Locator ID Separation Protocol (LISP) インターフェイスおよび Cisco Software-Defined Access (SD-Access) ソリューションではサポートされません。
- MACsec はマルチキャスト VPN (mVPN) ではサポートされません。
- MACsec は、SD-Access の展開ではサポートされていません。
- **should-secure** アクセスモードは、PSK 認証を使用するスイッチ間ポートでのみサポートされます。

MACsec 暗号化について

以降のセクションでは、MACsec 暗号化に関する情報を示します。

MACsec 暗号化の推奨事項

ここでは、MACsec 暗号化の設定に関する推奨事項を示します。

- スイッチとホスト間の接続では、機密性 (暗号化) オフセットを 0 として使用します。

- 双方向フォワーディングおよび検出 (BFD) タイマー値は、10 Gbps ポートでは 750 ミリ秒、10 Gbps を超える速度のポートでは 1.25 秒として使用します。
- アクティブセッションの MKA ポリシーまたは MACsec 設定を変更した後、ポートで **shutdown** コマンドを実行し、**no shutdown** コマンドを実行して、変更がアクティブセッションに適用されるようにします。
- 40Gbps 以上のポート速度には、Extended Packet Numbering (XPN) 暗号スイートを使用します。
- 接続アソシエーションキー (CAK) キー再生成オーバーラップタイマーを 30 秒以上に設定します。
- 10Gbps を超えるポート速度には、Cisco TrustSec Security Association Protocol (SAP) MACsec 暗号化を使用しないでください。
- どのインターフェイスでも、Cisco TrustSec SAP とアップリンク MKA の両方を同時に有効にしないでください。
- MACsec MKA 暗号化を使用することをお勧めします。

MACsec 暗号化の概要

MACsec は 2 台の MACsec 対応デバイス間のパケットの認証と暗号化の IEEE 802.1AE 規格です。Catalyst スイッチは、スイッチとホストデバイス間の暗号化に、スイッチからホストへのリンクでの MACsec Key Agreement (MKA) による 802.1AE 暗号化をサポートします。また、スイッチは、Cisco TrustSec ネットワーク デバイス アドミッション コントロール (NDAC)、Security Association Protocol (SAP) および MKA ベースのキー交換プロトコルを使用して、スイッチ間 (ネットワーク間デバイス) セキュリティの MACsec 暗号化をサポートします。



- (注) スイッチ間 MACsec が有効な場合、EAP-over-LAN (EAPOL) パケットを除くすべてのトラフィックが暗号化されます。

リンク層セキュリティはスイッチ間のパケット認証とスイッチ間の MACsec 暗号化の両方を含みます (暗号化は任意です)。リンク層セキュリティは、SAP ベースの MACsec でサポートされます。

表 19: スイッチ ポートの MACsec サポート

Connections	MACsec のサポート
Switch-to-Host	MACsec MKA の暗号化
Switch-to-Switch	MACsec MKA の暗号化 (推奨) Cisco TrustSec SAP

Cisco TrustSec と Cisco SAP はスイッチ間のリンクにのみ使用され、PC や IP フォンなどのエンドホストに接続されたスイッチポートではサポートされません。MKA は、スイッチからホストへのリンクとスイッチ間リンクでサポートされます。ホスト側のリンクは、IEEE 802.1x の有無にかかわらず異種デバイスを扱うために、一般に柔軟な認証順序を使用し、オプションで MKA ベースの MACsec 暗号化を使用できます。Cisco NDAC および SAP は、コンパクトなスイッチがワイヤリングクローゼットの外側にセキュリティを拡張するために使用する、ネットワーク エッジアクセス トポロジ (NEAT) と相互排他的です。

Media Access Control Security と MACsec Key Agreement

802.1AE で定義された MACsec では、暗号化キー入力のためにアウトオブバンド方式を使用することによって、有線ネットワーク上で MAC レイヤの暗号化を実現します。MACsec Key Agreement (MKA) プロトコルでは、必要なセッションキーを提供し、必要な暗号化キーを管理します。MKA と MACsec は、802.1x 拡張認証プロトコル (EAP-TLS) または事前共有キー (PSK) フレームワークを使用した認証に成功した後に実装されます。

MACsec を使用するスイッチでは、MKA ピアに関連付けられたポリシーに応じて、MACsec フレームまたは非 MACsec フレームを許可します。MACsec フレームは暗号化され、整合性チェック値 (ICV) で保護されます。スイッチは MKA ピアからフレームを受信すると、MKA によって提供されたセッションキーを使用してこれらのフレームを暗号化し、正しい ICV を計算します。スイッチはこの ICV をフレーム内の ICV と比較します。一致しない場合は、フレームが破棄されます。また、スイッチは現在のセッションキーを使用して、ICV を暗号化し、セキュアなポート (セキュアな MAC サービスを MKA ピアに提供するために使用されるアクセス ポイント) を介して送信されたフレームに追加します。

MKA プロトコルは、基礎となる MACsec プロトコルで使用される暗号キーを管理します。MKA の基本的な要件は 802.1x-REV で定義されています。MKA プロトコルでは 802.1x を拡張し、相互認証の確認によってピアを検出し、MACsec 秘密キーを共有してピアで交換されるデータを保護できます。

EAP フレームワークでは、新しく定義された EAP-over-LAN (EAPOL) パケットとして MKA を実装します。EAP 認証では、データ交換で両方のパートナーで共有されるマスターセッションキー (MSK) を生成します。EAP セッション ID を入力すると、セキュアな接続アソシエーションキー名 (CKN) が生成されます。スイッチは、アップリンクおよびダウンリンクの両方のオーセンティケータとして機能します。また、ダウンリンクのキーサーバーとして機能します。これによってランダムなセキュアアソシエーションキー (SAK) が生成され、クライアントパートナーに送信されます。クライアントはキーサーバーではなく、単一の MKA エンティティであるキーサーバーとだけ対話できます。キーの派生と生成の後で、スイッチは定期的にトランスポートをパートナーに送信します。デフォルトの間隔は 2 秒間です。

EAPOL プロトコルデータユニット (PDU) のパケット本体は、MACsec Key Agreement PDU (MKPDU) と呼ばれます。MKA セッションと参加者は、MKA ライフタイム (6 秒間) が経過しても参加者から MKPDU を受信していない場合に削除されます。たとえば、MKA ピアが接続を解除した場合、スイッチ上の参加者は MKA ピアから最後の MKPDU を受信した後、6 秒間経過するまで MKA の動作を継続します。



- (注) MKPDU の整合性チェック値 (ICV) インジケータはオプションです。トラフィックが暗号化されている場合、ICV はオプションではありません。

MKA ポリシー

定義済みの MKA ポリシーをインターフェイスに適用すると、インターフェイス上で MKA がイネーブルになります。MKA ポリシーを削除すると、そのインターフェイス上で MKA がディセーブルになります。次のオプションを設定可能です。

- 16 ASCII 文字未満のポリシー名。
- 物理インターフェイスごとの 0 バイト、30 バイト、または 50 バイトの機密保持 (暗号化) オフセット。

ポリシーマップアクションの定義

ここでは、ポリシーマップアクションとその定義について説明します。

- **Activate** : サービステンプレートをセッションに適用します。
- **Authenticate** : セッションの認証を開始します。
- **Authorize** : セッションを明示的に許可します。
- **Set-domain** : クライアントのドメインを明示的に設定します。
- **Terminate** : 実行中のメソッドを終了し、セッションに関連付けられているすべてのメソッドの詳細を削除します。
- **Deactivate** : セッションに適用されたサービステンプレートを削除します。適用されない場合、アクションは実行されません。
- **Set-timer** : タイマーを開始し、セッションに関連付けます。タイマーが期限切れになると、開始する必要があるアクションを処理できます。
- **Authentication-restart** : 認証を再開します。
- **Clear-session** : セッションを削除します。
- **Pause** : 認証を一時停止します。

残りのアクションについては説明の必要はなく、認証に関連したものです。

仮想ポート

仮想ポートは、1つの物理ポート上の複数のセキュアな接続アソシエーションに使用します。各接続アソシエーション (ペア) は仮想ポートを表します。スイッチ間では、物理ポートごとに1つの仮想ポートのみを指定できます。スイッチとホスト間では、物理ポートごとに最大2つの仮想ポートを指定でき、一方の仮想ポートはデータ VLAN の一部にできます。もう一方

は音声 VLAN に対してパケットを外部的にタグ付けする必要があります。同じポートで同じ VLAN 内のセキュアなセッションとセキュアでないセッションを同時にホストすることはできません。この制限のため、802.1x マルチ認証モードはサポートされません。

この制限の例外は、マルチホストモードで最初の MACsec サブリカントが正常に認証され、スイッチに接続されたハブに接続される場合です。ハブに接続された非 MACsec ホストでは、マルチホストモードであるため、認証なしでトラフィックを送信できます。最初にクライアントが成功した後、他のクライアントでは認証が必要ないため、マルチホストモードの使用は推奨しません。

仮想ポートは、接続アソシエーションの任意の ID を表し、MKA プロトコル外では意味を持ちません。仮想ポートは個々の論理ポート ID に対応します。仮想ポートの有効なポート ID は 0x0002 ~ 0xFFFF です。各仮想ポートは、16 ビットのポート ID に連結された物理インターフェイスの MAC アドレスに基づいて、一意のセキュアチャネル ID (SCI) を受け取ります。

MKA 統計情報

一部の MKA カウンタはグローバルに集約され、その他のカウンタはグローバルとセッション単位の両方で更新されます。また、MKA セッションのステータスに関する情報も取得できます。詳細については、[例：MKA 情報の表示 \(266 ページ\)](#) を参照してください。

キー ライフタイムおよびヒットレス キー ロールオーバー

MACsec キーチェーンには、キー ID とオプションのライフタイムが設定された複数の事前共有キー (PSK) を含めることができます。キーのライフタイムには、キーが期限切れになる時刻が指定されます。ライフタイム設定が存在しない場合は、無期限のデフォルトライフタイムが使用されます。ライフタイムが設定されている場合、ライフタイムの期限が切れた後に、MKA はキーチェーン内の次に設定された事前共有キーにロールオーバーします。キーのタイムゾーンは、ローカルまたは UTC を指定できます。デフォルトのタイムゾーンは UTC です。

キーチェーン内に 2 番目のキーを設定し、最初のキーのライフタイムを設定することで、同じキーチェーン内の次のキーにロールオーバーできます。最初のキーのライフタイムが期限切れになると、リスト内の次のキーに自動的にロールオーバーします。同一のキーがリンクの両側で同時に設定されている場合、キーのロールオーバーはヒットレスになります。つまり、キーはトラフィックを中断せずにロールオーバーされます。

すべての参加デバイスで、MACsec キーチェーンを Network Time Protocol (NTP) を使用して同期し、同じタイムゾーンを使用する必要があります。参加しているすべてのデバイスが同期されていない場合、接続アソシエーションキー (CAK) のキー再生成はすべてのデバイスで同時に開始されません。



(注) キーのライフタイムは、ヒットレス キー ロールオーバーを実現するためにオーバーラップする必要があります。

リプレイ保護ウィンドウ サイズ

リプレイ保護は、リプレイ攻撃に対抗するためにMACsecにより提供される機能です。暗号化された各パケットには一意のシーケンス番号が割り当てられ、シーケンスはリモートエンドで確認されます。メトロイーサネットサービスプロバイダーネットワークを介して送信されるフレームは、順序が変更されることが多くあります。これは、ネットワーク内で使用されている優先順位付けとロードバランシングのメカニズムによるものです。

フレームの順序が変更されるプロバイダーネットワーク上でMACsecの使用をサポートするには、リプレイウィンドウが必要です。ウィンドウ内のフレームは順不同で受信できますが、リプレイ保護されません。デフォルトのウィンドウサイズは0で、厳密な受信順序が適用されません。リプレイウィンドウのサイズは、0～ $2^{32}-1$ の範囲で設定できます。XPN暗号スイートの場合、最大リプレイウィンドウサイズは $2^{30}-1$ で、より大きなウィンドウサイズが設定されている場合、ウィンドウサイズは $2^{30}-1$ に制限されます。暗号スイートが非XPN暗号スイートに変更された場合、制限はなく、設定されたウィンドウサイズが使用されます。

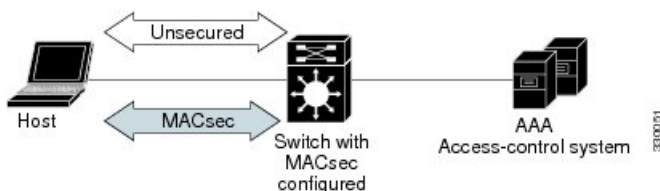
MACsec、MKA、および802.1x ホストモード

MACsecとMKAプロトコルは、802.1xシングルホストモード、マルチホストモード、またはマルチドメイン認証(MDA)モードで使用できます。マルチ認証モードはサポートされません。

シングルホストモード

次の図に、MKAを使用して、MACsecで1つのEAP認証済みセッションをセキュアにする方法を示します。

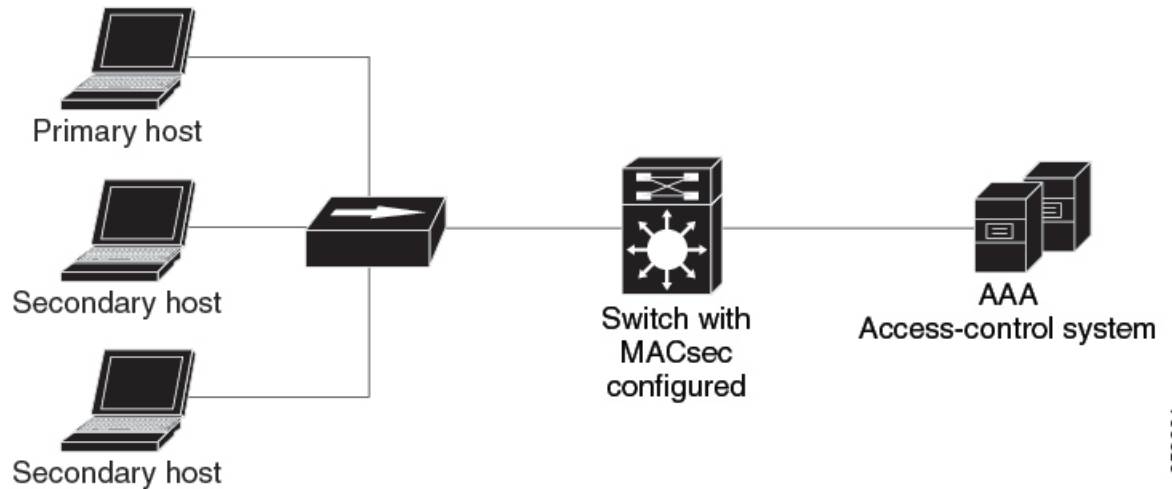
図 12: セキュアなデータセッションでのシングルホストモードのMACsec



マルチホストモード

標準の(802.1x REVではない)802.1xマルチホストモードでは、1つの認証に基づいてポートが開いているか、閉じられています。1人のユーザー(プライマリセキュアクライアントサービスのクライアントホスト)が認証される場合は、同じポートに接続されているホストに同じレベルのネットワークアクセスが提供されます。セカンダリホストがMACsecサブリカントの場合、認証できず、トラフィックフローは発生しません。非MACsecホストであるセカンダリホストは、マルチホストモードであるため、認証なしでネットワークにトラフィックを送信できます。次の図に、標準のマルチホスト非セキュアモードにおけるMACsecを示します。

図 13: マルチホストモードの MACsec : 非セキュア



253664



(注) マルチホストモードは推奨されていません。これは最初にクライアントが成功した後、他のクライアントでは認証が必要ないことから、安全性が低いからです。

マルチドメインモード

標準の (802.1x REV ではない) 802.1x マルチドメインモードでは、1つの認証に基づいてポートが開いているか、閉じられています。プライマリユーザー (データドメインの PC) が認証されると、同じレベルのネットワークアクセスが同じポートに接続されているドメインに提供されます。セカンダリユーザーが MACsec サプリカントの場合、認証できず、トラフィックフローは発生しません。非 MACsec ホストであるセカンダリユーザー (音声ドメインの IP フォン) は、マルチドメインモードであるため、認証なしでネットワークにトラフィックを送信できます。

証明書ベースの MACsec 暗号化

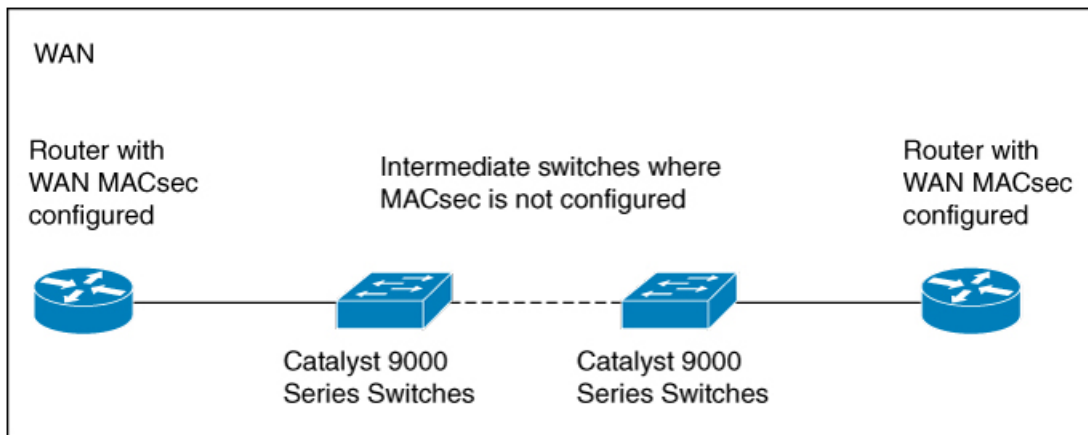
証明書ベースの MACsec 暗号化を使用して、デバイスのスイッチ間ポート間で MACsec MKA を設定できます。証明書ベースの MACsec 暗号化は相互認証を許可し、MSK (マスターセッションキー) を取得します。そのキーから、MKA 操作の接続アソシエーションキー (CAK) が取得されます。デバイスの証明書は、AAA サーバーへの認証用に、証明書ベースの MACsec 暗号化を使用して伝送されます。

中間スイッチの MACsec 接続

Cisco IOS XE Gibraltar 16.10.1 以前は、Cisco Catalyst 9000 シリーズスイッチとして中間スイッチで WAN MACsec が設定されているエンドデバイス間の MACsec 接続はサポートされていませんでした。MACsec が中間スイッチに設定されていない状態でエンドデバイスに WAN MACsec を設定すると、暗号化されたパケットはドロップされました。ASIC に ClearTag 機能が実装さ

れている場合、スイッチは MACsec ヘッダーを解析せずに暗号化されたパケットを転送します。以下のトポロジは、暗号化されたパケットが L2 スイッチングの中間スイッチを介してどのように転送されるかを示しています。

図 14: ClearTag MACsec のトポロジ: MACsec が中間スイッチで設定されていない



中間スイッチの MACsec 接続に関する制約事項

- Catalyst 9000 シリーズ スイッチを WAN MACsec がルータに設定されている中間スイッチとして使用するホップバイホップ MACsec 暗号化はサポートされていません。
- 中間スイッチが Catalyst 9000 シリーズ スイッチのルータに設定された WAN MACsec は、レイヤ 3 VPN ではサポートされません。
- 中間スイッチが Catalyst 9000 シリーズ スイッチのルータに設定された WAN MACsec では、should-secure モードのみで Cisco Discovery Protocol ネイバーが表示されます。

スイッチ間 MKA MACsec マストセキュアポリシー

Cisco IOS XE Fuji 16.8.1a 以降、入力と出力の両方で must-secure のサポートが有効になります。MKA および SAP では、Must-secure がサポートされています。must-secure を有効にすると、EAPoL トラフィックのみが暗号化されません。他のトラフィックは暗号化されます。暗号化されないパケットはドロップされます。



(注) デフォルトでは、Must-secure モードが有効になっています。

Cisco IOS XE Fuji 16.8.1a よりも前のリリースでは、MKA と SAP で should-secure がサポートされていました。should-secure を有効にすると、ピアが MACsec に設定されている場合はデータトラフィックが暗号化され、それ以外の場合はクリアテキストで送信されます。

MACsec Extended Packet Numbering (XPN)

各 MACsec フレームには 32 ビット パケット番号 (PN) が含まれており、特定のセキュリティアソシエーションキー (SAK) に対して一意です。PN が枯渇すると ($2^{31}-1$ の 75% に達した後)、SAK キーが再生成されてデータプレーンキーが更新されます。40 Gb/s などの高容量リンクの場合は数秒以内に PN が枯渇し、コントロールプレーンに対する SAK キーの頻繁な再生成が必要になります。XPN が使用されている場合、 $2^{63}-1$ の 75% に達した後、MACsec フレームの PN は 64 ビット値であるため、PN が枯渇するまで数年を要します。これにより、高速リンクで頻繁な SAK キー再生成が発生しなくなります。MKA/MACsec の XPN 機能により、大容量リンクで発生する可能性のある頻繁な SAK キー再生成が不要になります。XPN は、40 Gb/s、100 Gb/s などの高速リンクでの FIPS/CC コンプライアンスの必須要件です。



(注) MACsec XPN は、スイッチ間ポートでのみサポートされます。

XPN では次のキー再生成が可能です。

- ボリュームベースのキー再生成**：頻繁な SAK キー再生成が発生しないようにするために、定義された MKA ポリシーの下で GCM-AES-XPN-128 または GCM-AES-XPN-256 暗号スイートを使用して XPN を設定できます。これらの暗号スイートを使用すると、1 つの SAK で 2^{32} 以上のフレームを保護できます。XPN では、64 ビット値の PN がサポートされています。MACsec フレームには最下位 32 ビットのみが含まれ、最上位 32 ビットはピア自身、つまり送信側と受信側のピアの両方により維持されます。それぞれのピアの LAPN (許容される最小パケット番号) の MSB (最上位ビット) が設定され、MACsec フレームで受信した PN 値の MSB が 0 の場合、PN の最上位 32 ビットが受信側で増分されます。したがって、送信側と受信側の両方のピアが、MACsec フレーム構造を変更せずに同じ PN 値を維持します。
- 時間ベースのキー再生成**：SAK キー再生成を手動で設定するために、タイマーベースのキー再生成がサポートされており、指定された間隔で SAK キー再生成を開始することができます。インターフェイスに適用される定義済み MKA ポリシーの SAK キー再生成間隔を設定するには、MKA ポリシーコンフィギュレーションモードで **sak rekey interval time-interval** コマンドを使用します。

ポートチャネルの MKA/MACsec

MKA/MACsec は、ポートチャネルのポートメンバで設定できます。ポートチャネルのポートメンバ間で MKA セッションが確立されるため、MKA/MACsec はポートチャネルに依存しません。



- (注) ポートチャネルの一部として形成される EtherChannel リンクは、合同または異種のいずれかです。つまり、リンクは MACsec セキュアまたは非 MACsec セキュアのいずれかになります。ポートチャネルの一方のポートメンバが MACsec に設定されていない場合でも、ポートメンバ間の MKA セッションが確立されます。

ポートチャネルのセキュリティを強化するために、すべてのメンバポートで MKA/MACsec を有効にすることをお勧めします。

MACsec 暗号アナウンスメント

暗号アナウンスメントを使用すると、サブリカントとオーセンティケータは、それぞれの MACsec 暗号スイート機能を相互にアナウンスできます。サブリカントとオーセンティケータの両方が、サポートされる最大の共通 MACsec 暗号スイートを計算し、MKA セッションのキー情報と同じものを使用します。



- (注) MKA ポリシーで設定されている MACsec 暗号スイート機能だけが、オーセンティケータからサブリカントにアナウンスされます。

EAPOL アナウンスメントには 2 つのタイプがあります。

- 非セキュアアナウンスメント (EAPOL PDU) : 非セキュアアナウンスメントは、MACsec 暗号スイート機能を非セキュアな方法で伝送する EAPOL アナウンスメントです。これらのアナウンスメントは、認証の前に MKA セッションに使用するキーの幅を決定するために使用されます。
- セキュアアナウンスメント (MKPDU) : セキュアアナウンスメントは、以前は非セキュアアナウンスメントで共有されていた MACsec 暗号スイート機能を再検証します。

セッションが認証されると、EAPOL アナウンスメントを介して受信されたピア機能がセキュアアナウンスメントで再検証されます。機能に不一致がある場合、MKA セッションは切断されます。



- (注) サブリカントとオーセンティケータ間の MKA セッションは、両方に設定された MACsec 暗号スイート機能が共通の暗号スイートにならない場合でも切断されません。

MACsec 暗号化の設定方法

以降のセクションでは、MACsec 暗号化を構成するさまざまなタスクに関する情報を示します。

MKA および MACsec の設定

デフォルトでは、MACsec は無効です。MKA ポリシーは設定されていません。

MKA ポリシーの設定

MKA プロトコルポリシーを作成するには、特権 EXEC モードで次の手順を実行します。MKA では 802.1x をイネーブルにすることも必要であることに注意してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **mka policy *policy-name***
4. **key-server priority**
5. **include-icv-indicator**
6. **macsec-cipher-suite {*gcm-aes-128* | *gcm-aes-256*}**
7. **confidentiality-offset *offset-value***
8. **ssci-based-on-sci**
9. **end**
10. **show mka policy**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mka policy <i>policy-name</i> 例： Device(config)# mka policy mka_policy	MKA ポリシーを指定して、MKA ポリシーコンフィギュレーション モードを開始します。ポリシー名の長さは最大で 16 文字です。

	コマンドまたはアクション	目的
		<p>(注) MKA ポリシー内のデフォルトの MACsec 暗号スイートは常に「GCM-AES-128」です。デバイスが「GCM-AES-128」および「GCM-AES-256」の両方の暗号方式をサポートしている場合は、ユーザー定義の MKA ポリシーを定義して使用し、必要に応じて、128 および 256 ビット両方の暗号を含めるか、または 256 ビットのみ暗号を含めることを強くお勧めします。</p>
ステップ 4	key-server priority 例： Device(config-mka-policy)# key-server priority 200	<p>MKA キーサーバオプションを設定し、優先順位を設定します (0 ~ 255 の値)。</p> <p>(注) キーサーバプライオリティの値を 255 に設定した場合、ピアはキーサーバになることはできません。キーサーバの優先順位の値は MKA PSK に対してのみ有効です。MKA EAPTLS に対しては有効ではありません。</p>
ステップ 5	include-icv-indicator 例： Device(config-mka-policy)# include-icv-indicator	<p>MKPDU の ICV インジケータを有効にします。ICV インジケータを無効にするには、このコマンドの no 形式を使用します。</p>
ステップ 6	macsec-cipher-suite {gcm-aes-128 gcm-aes-256} 例： Device(config-mka-policy)# macsec-cipher-suite gcm-aes-128	<p>128 ビットまたは 256 ビット暗号化により SAK を取得するための暗号スイートを設定します。</p>
ステップ 7	confidentiality-offset offset-value 例： Device(config-mka-policy)# confidentiality-offset 0	<p>各物理インターフェイスに機密性 (暗号化) オフセットを設定します。</p> <p>(注) オフセット値は、0、30、または 50 を指定できます。クライアントで Anyconnect を使用している場合は、オフセット 0 を使用することをお勧めします。</p>
ステップ 8	ssci-based-on-sci 例： Device(config-mka-policy)# ssci-based-on-sci	<p>(任意) Secure Channel Identifier (SCI) 値に基づいて Short Secure Channel Identifier (SSCI) 値を計算します。SCI 値が高いほど、SSCI 値は低くなります。</p>

	コマンドまたはアクション	目的
ステップ 9	end 例： Device(config-mka-policy)# end	MKA ポリシー コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 10	show mka policy 例： Device# show mka policy	MKA ポリシー設定情報を表示します。

スイッチからホストへの MACsec の暗号化設定

音声用に 1 つの MACsec セッションとデータ用に 1 つの MACsec セッションが存在するインターフェイスで MACsec を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface type number**
4. **switchport access vlan vlan-id**
5. **switchport mode access**
6. **macsec**
7. **authentication event linksec fail action authorize vlan vlan-id**
8. **authentication host-mode multi-domain**
9. **authentication linksec policy must-secure**
10. **authentication port-control auto**
11. **authentication periodic**
12. **authentication timer reauthenticate**
13. **authentication violation protect**
14. **mka policy policy-name**
15. **dot1x pae authenticator**
16. **spanning-tree portfast**
17. **end**
18. **show authentication session interface interface-id details**
19. **show macsec interface interface-id**
20. **show mka sessions**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードをイネーブルにします。 • プロンプトが表示されたら、パスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface type number 例： Device(config)# interface GigabitEthernet 1/0/1	MACsec 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。インターフェイスは物理インターフェイスでなければなりません。
ステップ 4	switchport access vlan vlan-id 例： Device(config-if)# switchport access vlan 1	このポートのアクセス VLAN を設定します。
ステップ 5	switchport mode access 例： Device(config-if)# switchport mode access	インターフェイスをアクセス ポートとして設定します。
ステップ 6	macsec 例： Device(config-if)# macsec	インターフェイス上で 802.1ae MACsec をイネーブルにします。macsec コマンドを使用すると、スイッチからホストへのリンクでのみ MKA MACsec が有効になります。
ステップ 7	authentication event linksec fail action authorize vlan vlan-id 例： Device(config-if)# authentication event linksec fail action authorize vlan 1	(任意) 認証の試行に失敗した後で、ポート上の制限付き VLAN を許可することによって、ユーザー証明書が認識されない認証リンク セキュリティの問題をスイッチが処理することを指定します。
ステップ 8	authentication host-mode multi-domain 例： Device(config-if)# authentication host-mode multi-domain	ホストと音声デバイスの両方が、802.1x で許可されたポート上で認証されるように、ポート上の認証マネージャ モードを設定します。設定されていない場合、デフォルトのホストモードはシングルです。
ステップ 9	authentication linksec policy must-secure 例： Device(config-if)# authentication linksec policy must-secure	LinkSec セキュリティ ポリシーを設定して、ピアを利用できる場合に、MACsec でセッションをセキュアにします。設定されていない場合、デフォルト値は <i>should secure</i> です。
ステップ 10	authentication port-control auto 例： Device(config-if)# authentication port-control auto	ポートでの 802.1x 認証をイネーブルにします。スイッチとクライアント間の認証交換に基づいてポートが許可ステートまたは無許可ステートに変わります。
ステップ 11	authentication periodic 例：	(任意) このポートの再認証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
	Device(config-if) # authentication periodic	
ステップ 12	authentication timer reauthenticate 例： Device(config-if) # authentication timer reauthenticate	(任意) 1 から 65535 までの値 (秒) を入力します。サーバーから再認証タイムアウト値を取得します。デフォルトの再認証時間は 3600 秒です。
ステップ 13	authentication violation protect 例： Device(config-if) # authentication violation protect	新しいデバイスがポートに接続された場合、または最大数のデバイスがポートに接続された後に新しいデバイスがそのポートに接続された場合に、予期しない着信 MAC アドレスを破棄するようポートを設定します。設定されていない場合、デフォルトではポートをシャットダウンします。
ステップ 14	mka policy policy-name 例： Device(config-if) # mka policy mka_policy	既存の MKA プロトコル ポリシーをインターフェイスに適用し、インターフェイス上で MKA をイネーブルにします。MKA ポリシーを設定しなかった場合 (mka policy グローバル コンフィギュレーション コマンドを入力して)。
ステップ 15	dot1x pae authenticator 例： Device(config-if) # dot1x pae authenticator	ポートを 802.1x ポートアクセスエンティティ (PAE) オーセンティケータとして設定します。
ステップ 16	spanning-tree portfast 例： Device(config-if) # spanning-tree portfast	関連するすべての VLAN 内のインターフェイスで、スパンニングツリー Port Fast をイネーブルにします。Port Fast 機能がイネーブルの場合、インターフェイスはブロッキングステートからフォワーディングステートに直接移行します。その際に、中間のスパンニングツリーステートは変わりません
ステップ 17	end 例： Device(config) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 18	show authentication session interface interface-id details 例： Device# show authentication session interface GigabitEthernet 1/0/1	許可されたセッションのセキュリティステータスの詳細を確認します。
ステップ 19	show macsec interface interface-id 例： Device# show macsec interface GigabitEthernet 1/0/1	インターフェイスの MACsec ステータスを確認します。

	コマンドまたはアクション	目的
ステップ 20	show mka sessions 例 : Device# show mka sessions	確立された MKA セッションを確認します。

PSK を使用した MKA MACsec の設定

PSK を使用した MACsec MKA の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **key chain key-chain-name macsec**
4. **key hex-string**
5. **cryptographic-algorithm** {*aes-128-cmac* | *aes-256-cmac*}
6. **key-string** { [0/6/7] *pwd-string* | *pwd-string*}
7. **lifetime local** [*start timestamp* {*hh::mm::ss* | *day* | *month* | *year*}] [**duration** *seconds* | *end timestamp* {*hh::mm::ss* | *day* | *month* | *year*}]
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	key chain key-chain-name macsec 例 : Device(config)# key chain keychain1 macsec	キー チェーンを設定して、キー チェーン コンフィギュレーション モードを開始します。
ステップ 4	key hex-string 例 : Device(config-key-chain)# key 1000	キーチェーン内の各キーの固有識別子を設定し、キーチェーンのキー 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
		(注) 128ビット暗号化の場合は、1～32文字の16進数キー文字列を使用します。256ビット暗号化の場合は、64文字の16進数キー文字列を使用します。
ステップ 5	cryptographic-algorithm { <i>aes-128-cmac</i> / <i>aes-256-cmac</i> } 例： Device(config-key-chain)# cryptographic-algorithm aes-128-cmac	128ビットまたは256ビット暗号による暗号化認証アルゴリズムを設定します。
ステップ 6	key-string { [0/6/7] <i>pwd-string</i> / <i>pwd-string</i> } 例： Device(config-key-chain)# key-string 12345678901234567890123456789012	キー文字列のパスワードを設定します。16進数の文字のみを入力する必要があります。
ステップ 7	lifetime local [<i>start timestamp {hh::mm::ss / day / month / year}</i>] [<i>duration seconds</i> <i>end timestamp {hh::mm::ss / day / month / year}</i>] 例： Device(config-key-chain)# lifetime local 12:12:00 July 28 2016 12:19:00 July 28 2016	事前共有キーの有効期間を設定します。
ステップ 8	end 例： Device(config-key-chain)# end	キーチェーンコンフィギュレーションモードを終了して、特権 EXEC モードに戻ります。

PSK を使用した、インターフェイスでの MACsec MKA の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **macsec network-link**
5. **mka policy** *policy-name*
6. **mka pre-shared-key key-chain** *key-chain name*
7. **macsec replay-protection window-size** *frame number*
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface type number 例： Device(config-if)# interface GigabitEthernet 0/0/0	インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	macsec network-link 例： Device(config-if)# macsec network-link	インターフェイス上で MACsec をイネーブルにします。
ステップ 5	mka policy policy-name 例： Device(config-if)# mka policy mka_policy	MKA ポリシーを設定します。
ステップ 6	mka pre-shared-key key-chain key-chain name 例： Device(config-if)# mka pre-shared-key key-chain key-chain-name	MKA 事前共有キーのキーチェーン名を設定します。
ステップ 7	macsec replay-protection window-size frame number 例： Device(config-if)# macsec replay-protection window-size 10	リプレイ保護の MACsec ウィンドウサイズを設定します。
ステップ 8	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

次のタスク

セッションの実行中に MKA PSK が設定されたインターフェイスで MKA ポリシーを変更することは推奨されません。ただし、変更が必要な場合は、次のようにポリシーを再設定する必要があります。

1. **no macsec network-link** コマンドを使用して、各参加ノードの macsec network-link 設定を削除し、既存のセッションを無効にします。

2. **mka policy policy-name** コマンドを使用して、各参加ノードのインターフェイスで MKA ポリシーを設定します。
3. **macsec network-link** コマンドを使用して、各参加ノードで新しいセッションを有効にします。

証明書ベース MACsec 暗号化の設定

ポイントツーポイントリンクで MKA による MACsec を設定するには、次のタスクを実行します。

- 証明書登録の設定
 - キー ペアの生成
 - SCEP 登録の設定
 - 証明書の手動設定
- 認証ポリシーの設定
- 証明書ベース MACsec 暗号化プロファイルと IEEE 802.1x ログイン情報の設定
- インターフェイスで証明書ベース MACsec 暗号化を使用する MACsec MKA の設定

キー ペアの生成

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	crypto key generate rsa label label-name general-keys modulus size 例： Device(config)# crypto key generate rsa label general-keys modulus 2048	署名および暗号化用に RSA キーペアを作成します。 label キーワードを使用すると、各キーペアにラベルを割り当てることもできます。このラベルは、キーペアを使用するトラストポイントによって参照されます。ラベルを割り当てなかった場合、キーペアには <Default-RSA-Key> というラベルが自動的に付けられます。

	コマンドまたはアクション	目的
		追加のキーワードを使用しない場合、このコマンドは汎用 RSA キーペアを 1 つ生成します。係数が指定されていない場合は、デフォルトのキー係数である 1024 が使用されます。その他の係数サイズを指定するには、 <code>modulus</code> キーワードを使用します。
ステップ 4	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 5	show authentication session interface interface-id 例： Device# show authentication session interface gigabitethernet 0/1/1	許可されたセッションのセキュリティステータスを確認します。

SCEP による登録の設定

Simple Certificate Enrollment Protocol (SCEP) は、HTTP を使用して認証局 (CA) または登録局 (RA) と通信する、シスコが開発した登録プロトコルです。SCEP は、要求および証明書の送受信に最も一般的に使用される方式です。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none">パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	crypto pki trustpoint server name 例： Device(config)# crypto pki trustpoint ka	トラストポイントおよび設定された名前を宣言して、CA トラストポイントコンフィギュレーションモードを開始します。
ステップ 4	enrollment url url name pem 例： Device(ca-trustpoint)# enrollment url http://url:80	デバイスが証明書要求を送信する CA の URL を指定します。 URL 内の IPv6 アドレスは括弧で囲む必要があります。たとえば、 <code>http://[2001:DB8:1:1::1]:80</code> です。 <code>pem</code> キーワードは、証明書要求に Privacy Enhanced Mail (PEM) の境界を追加します。

	コマンドまたはアクション	目的
ステップ 5	rsakeypair label 例： Device(ca-trustpoint)# rsakeypair exampleCAkeys	証明書に関連付けるキー ペアを指定します。 (注) rsakeypair 名は、信頼ポイント名と一致している必要があります。
ステップ 6	serial-number none 例： Device(ca-trustpoint)# serial-number none	none キーワードは、証明書要求にシリアル番号が含まれないことを指定します。
ステップ 7	ip-address none 例： Device(ca-trustpoint)# ip-address none	none キーワードは、証明書要求に IP アドレスが含まれないことを指定します。
ステップ 8	revocation-check crl 例： Device(ca-trustpoint)# revocation-check crl	ピアの証明書が取り消されていないことを確認する方法として CRL を指定します。
ステップ 9	auto-enroll percent regenerate 例： Device(ca-trustpoint)# auto-enroll 90 regenerate	<p>自動登録をイネーブルにします。これにより、クライアントは CA から自動的にロールオーバー証明書を要求できます。</p> <p>自動登録がイネーブルでない場合、証明書の失効時にクライアントを手動で PKI に再登録する必要があります。</p> <p>デフォルトでは、デバイスのドメイン ネーム システム (DNS) 名だけが証明書に含められます。</p> <p>現行の証明書の有効期間が指定のパーセンテージに達したときに、新しい証明書が要求されるように指定するには、percent 引数を使用します。</p> <p>名前付きのキーがすでに存在する場合でも、証明書の新しいキーを生成するには、regenerate キーワードを使用します。</p> <p>ロールオーバー中のキー ペアがエクスポート可能な場合、新しいキーペアもエクスポート可能です。次のコメントがトラストポイントコンフィギュレーションに表示され、キー ペアがエクスポート可能かどうかを示されます。「! RSA key pair associated with trustpoint is exportable.」</p> <p>新しいキー ペアは、セキュリティ上の問題に対処するために生成することを推奨します。</p>

	コマンドまたはアクション	目的
ステップ 10	exit 例： Device(ca-trustpoint)# exit	CA トラストポイント コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 11	crypto pki authenticate name 例： Device(config)# crypto pki authenticate myca	CA 証明書を取得して、認証します。
ステップ 12	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 13	show crypto pki certificate trustpoint name 例： Device# show crypto pki certificate ka	信頼ポイントの証明書に関する情報を表示します。

登録の手動設定

CA が SCEP をサポートしない場合、またはルータと CA 間のネットワーク接続が不可能な場合。手動での証明書登録を設定するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none">パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	crypto pki trustpoint server name 例： Device# crypto pki trustpoint ka	トラストポイントおよび設定された名前を宣言して、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 4	enrollment url url-name 例： Device(ca-trustpoint)# enrollment url http://url:80	デバイスが証明書要求を送信する CA の URL を指定します。 URL 内の IPv6 アドレスは括弧で囲む必要があります。たとえば、 <code>http://[2001:DB8:1:1::1]:80</code> です。 <code>pem</code> キーワードは、証明書要求に Privacy Enhanced Mail (PEM) の境界を追加します。

	コマンドまたはアクション	目的
ステップ 5	rsa keypair label 例： Device(ca-trustpoint)# rsa keypair exampleCAkeys	証明書に関連付けるキー ペアを指定します。
ステップ 6	serial-number none 例： Device(ca-trustpoint)# serial-number none	証明書要求にシリアル番号が含まれないことを指定します。
ステップ 7	ip-address none 例： Device(ca-trustpoint)# ip-address none	none キーワードは、証明書要求に IP アドレスが含まれないことを指定します。
ステップ 8	revocation-check crl 例： Device(ca-trustpoint)# revocation-check crl	ピアの証明書が取り消されていないことを確認する方法として CRL を指定します。
ステップ 9	exit 例： Device(ca-trustpoint)# exit	CA トラストポイントコンフィギュレーションモードを終了し、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 10	crypto pki authenticate name 例： Device(config)# crypto pki authenticate myca	CA 証明書を取得して、認証します。
ステップ 11	crypto pki enroll name 例： Device(config)# crypto pki enroll myca	証明書要求を生成し、証明書サーバーにコピーおよびペーストするために要求を表示します。 プロンプトが表示されたら、登録情報を入力します。たとえば、証明書要求にデバイスの FQDN および IP アドレスを含めるかどうかを指定します。 コンソール端末に対して証明書要求を表示するかについても選択できます。 必要に応じて、Base 64 符号化証明書を PEM ヘッダーを付けて、または付けずに表示します。
ステップ 12	crypto pki import name certificate 例： Device(config)# crypto pki import myca certificate	許可された証明書を取得するコンソール端末で、TFTP によって証明書をインポートします。 デバイスは、拡張子が「.req」から「.crt」に変更されたことを除いて、要求の送信に使用した同じファイル名を使用して、許可された証明書を TFTP によって取得しようと試みます。用途キー証明書の場、拡張子「-sign.crt」および「-encr.crt」が使用されます。

	コマンドまたはアクション	目的
		<p>デバイスは、受信したファイルを解析して証明書を検証し、証明書をスイッチの内部証明書データベースに挿入します。</p> <p>(注) 一部の CA は、証明書要求の用途キー情報を無視し、汎用目的の証明書を発行します。ご使用の CA が証明書要求の用途キー情報を無視する場合は、汎用目的の証明書だけをインポートしてください。ルータは、生成される 2 つのキーペアのいずれも使用しません。</p>
ステップ 13	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 14	show crypto pki certificate trustpoint name 例： Device# show crypto pki certificate ka	信頼ポイントの証明書に関する情報を表示します。

スイッチ間の MACsec の暗号化設定

証明書ベース MACsec 暗号化を使用して MKA MACsec をインターフェイスに適用するには、次のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface type number 例： Device(config)# interface gigabitethernet 0/2/1	MACsec 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。インターフェイスは物理インターフェイスでなければなりません。
ステップ 4	macsec network-link 例：	インターフェイス上で MACsec をイネーブルにします。

	コマンドまたはアクション	目的
	<code>Device(config-if) # macsec network-link</code>	
ステップ 5	authentication periodic 例： <code>Device(config-if) # authentication periodic</code>	このポートの再認証をイネーブルにします。
ステップ 6	authentication timer reauthenticate interval 例： <code>Device(config-if) # authentication timer reauthenticate interval</code>	再認証間隔を設定します。
ステップ 7	access-session host-mode multi-host 例： <code>Device(config-if) # access-session host-mode multi-host</code>	ホストにインターフェイスへのアクセスを許可します。
ステップ 8	access-session closed 例： <code>Device(config-if) # access-session closed</code>	インターフェイスへの事前認証アクセスを防止します。
ステップ 9	access-session port-control auto 例： <code>Device(config-if) # access-session port-control auto</code>	ポートの認可状態を設定します。
ステップ 10	dot1x pae both 例： <code>Device(config-if) # dot1x pae both</code>	ポートを 802.1X ポートアクセス エンティティ (PAE) のサブリカントおよびオーセンティケータとして設定します。
ステップ 11	dot1x credentials profile 例： <code>Device(config-if) # dot1x credentials profile</code>	802.1x クレデンシャルプロファイルをインターフェイスに割り当てます。
ステップ 12	end 例： <code>Device(config-if) # end</code>	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、特権 EXEC モードに戻ります。
ステップ 13	show macsec interface interface-id 例： <code>Device# show macsec interface GigabitEthernet 1/0/1</code>	インターフェイスの MACsec の詳細を表示します。

MACsec XPN の設定

XPN の MKA ポリシーの設定

MKA ポリシーで XPN を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **mka policy *policy-name***
4. **macsec-cipher-suite { *gcm-aes-128* | *gcm-aes-256* | *gcm-aes-xpn-128* | *gcm-aes-xpn-256* }**
5. **sak-rekey interval *time-interval***
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mka policy <i>policy-name</i> 例： Device(config)# mka policy mka_policy	MKA ポリシーを指定して、MKA ポリシー コンフィギュレーション モードを開始します。ポリシー名の長さは最大で 16 文字です。 (注) MKA ポリシー内のデフォルトの MACsec 暗号スイートは常に「GCM-AES-128」です。デバイスが「GCM-AES-128」および「GCM-AES-256」の両方の暗号方式をサポートしている場合は、ユーザー定義の MKA ポリシーを定義して使用し、必要に応じて、128 および 256 ビット両方の暗号を含めるか、または 256 ビットのみを含めることを強くお勧めします。
ステップ 4	macsec-cipher-suite { <i>gcm-aes-128</i> <i>gcm-aes-256</i> <i>gcm-aes-xpn-128</i> <i>gcm-aes-xpn-256</i> } 例：	XPN 用の 128 ビットおよび 256 ビット暗号により SAK を取得するための暗号スイートを設定します。

XPN MKA ポリシーをインターフェイスに適用する

	コマンドまたはアクション	目的
	Device(config-mka-policy) # macsec-cipher-suite gcm-aes-xpn-256	
ステップ 5	sak-rekey interval time-interval 例： Device(config-mka-policy) # sak-rekey interval 50	(任意) SAK キー再生成間隔を秒単位で設定します。範囲は 30 ~ 65535 です。デフォルトでは、SAK キー再生成間隔は、インターフェイス速度に応じて自動的に発生します。 SAK キー再生成タイマーを停止するには、このコマンドの no 形式を使用します。
ステップ 6	end 例： Device(config-mka-policy) # end	MKA ポリシー コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

XPN MKA ポリシーをインターフェイスに適用する

XPN MKA ポリシーをインターフェイスに適用するには、次のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-name 例： Device(config) # interface gigabitethernet 1/0/1	MACsec 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。インターフェイスは物理インターフェイスでなければなりません。
ステップ 4	mka policy policy-name 例： Device(config-if) # mka policy mka-xpn-policy	XPNMKA プロトコルポリシーをインターフェイスに適用します。
ステップ 5	end 例： Device(config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ポートチャネル用の MKA/MACsec の設定

PSK を使用したポートチャネルの MKA/MACsec の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **macsec network-link**
5. **mka policy *policy-name***
6. **mka pre-shared-key key-chain *key-chain-name***
7. **macsec replay-protection window-size *frame number***
8. **channel-group *channel-group-number* mode {auto | desirable} | {active | passive} | {on}**
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config-if)# interface gigabitethernet 1/0/3	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	macsec network-link 例： Device(config-if)# macsec network-link	インターフェイス上で MACsec をイネーブルにします。レイヤ 2 およびレイヤ 3 ポートチャネルをサポートします。
ステップ 5	mka policy <i>policy-name</i> 例： Device(config-if)# mka policy mka_policy	MKA ポリシーを設定します。
ステップ 6	mka pre-shared-key key-chain <i>key-chain-name</i> 例： Device(config-if)# mka pre-shared-key key-chain key-chain-name	MKA 事前共有キーのキーチェーン名を設定します。 (注) MKA 事前共有キーは、物理インターフェイスまたはサブインターフェイスで設定できますが、両方で設定することはできません。

	コマンドまたはアクション	目的
ステップ 7	macsec replay-protection window-size <i>frame number</i> 例 : Device(config-if)# macsec replay-protection window-size 0	リプレイ保護の MACsec ウィンドウサイズを設定します。
ステップ 8	channel-group <i>channel-group-number</i> mode { auto desirable } { active passive } { on } 例 : Device(config-if)# channel-group 3 mode auto active on	チャンネルグループ内にポートを設定し、モードを設定します。 (注) インターフェイスで MACsec を設定しないと、チャンネルグループのポートを設定できません。このステップの前に、ステップ 3、4、5、および 6 のコマンドを設定する必要があります。 channel-number の指定できる範囲は 1 ~ 4096 です。ポートチャネルがない場合は、このチャンネルグループに関連付けられたポートチャネルが自動的に作成されます。モードは、以下のキーワードのいずれかを選択します。 <ul style="list-style-type: none"> • auto : PAgP デバイスが検出された場合に限り、PAgP を有効にします。ポートをパッシブ ネゴシエーション ステートにします。この場合、ポートは受信する PAgP パケットに応答しますが、PAgP パケットネゴシエーションを開始することはありません。 (注) EtherChannel メンバが、スイッチスタックにある異なるスイッチのメンバである場合、auto キーワードはサポートされません。 • desirable : 無条件に PAgP を有効にします。ポートをアクティブ ネゴシエーション ステートにします。この場合、ポートは PAgP パケットを送信することによって、相手ポートとのネゴシエーションを開始します。 (注) EtherChannel メンバが、スイッチスタックにある異なるスイッチのメンバである場合、desirable キーワードはサポートされません。 • on : PAgP または LACP を使用せずにポートが強制的にチャンネル化されます。on モードでは、EtherChannel が存在するのは、on モードのポー

	コマンドまたはアクション	目的
		<p>トグループが、onモードの別のポートグループに接続する場合だけです。</p> <ul style="list-style-type: none"> • active : LACP デバイスが検出された場合に限り、LACP を有効にします。ポートをアクティブ ネゴシエーション ステートにします。この場合、ポートは LACP パケットを送信することによって、相手ポートとのネゴシエーションを開始します。 • passive : ポート上で LACP を有効にして、ポートをパッシブ ネゴシエーション ステートにします。この場合、ポートは受信する LACP パケットに応答しますが、LACP パケットネゴシエーションを開始することはありません。
ステップ 9	end 例 : Device(config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

レイヤ 2 EtherChannel のポートチャネル論理インターフェイスの設定

レイヤ 2 EtherChannel 用のポートチャネルインターフェイスを作成するには、次の作業を行います。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *channel-group-number*
4. **switchport**
5. **switchport mode** {access | trunk}
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	interface port-channel <i>channel-group-number</i> 例： Device(config)# interface port-channel 1	ポートチャネルインターフェイスを作成し、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) ポートチャネルインターフェイスを削除するには、このコマンドの no 形式を使用します。
ステップ 4	switchport 例： Device(config-if)# switchport	レイヤ 3 モードになっているインターフェイスを、レイヤ 2 設定のレイヤ 2 モードに切り替えます。
ステップ 5	switchport mode { <i>access</i> <i>trunk</i> } 例： Device(config-if)# switchport mode access	すべてのポートをスタティックアクセスポートとして同じ VLAN に割り当てるか、またはトランクとして設定します。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

レイヤ 3 EtherChannel のポートチャネル論理インターフェイスの設定

レイヤ 3 EtherChannel 用のポートチャネルインターフェイスを作成するには、次の作業を行います。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no switchport**
5. **ip address** *ip-address subnet-mask*
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	Device# configure terminal	
ステップ 3	interface interface-id 例 : Device(config)# interface gigabitethernet 1/0/2	インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	no switchport 例 : Device(config-if)# no switchport	レイヤ 2 モードになっているインターフェイスを、レイヤ 3 設定用にレイヤ 3 モードに切り替えます。
ステップ 5	ip address ip-address subnet-mask 例 : Device(config-if)# ip address 10.2.2.3 255.255.255.254	EtherChannel に IP アドレスおよびサブネットマスクを割り当てます。
ステップ 6	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

MACsec 暗号アナウンスメントの設定

以降のセクションでは、MACsec 暗号アナウンスを設定するためのさまざまなタスクに関する情報を示します。

セキュアアナウンスメントの MKA ポリシーの設定

MKA プロトコルポリシーを作成して MKPDU でセキュアアナウンスメントを有効にするには、特権 EXEC モードで次の手順を実行します。デフォルトでは、セキュアアナウンスメントは無効になっています。

手順の概要

1. **enable**
2. **configure terminal**
3. **mka policy policy-name**
4. **key-server priority**
5. **send-secure-announcements**
6. **macsec-cipher-suite {gcm-aes-128 | gcm-aes-256}**
7. **end**
8. **show mka policy**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mka policy policy-name 例： Device(config)# mka policy mka_policy	MKA ポリシーを指定して、MKA ポリシー コンフィギュレーションモードを開始します。ポリシー名の長さは最大で 16 文字です。 (注) MKA ポリシーのデフォルトの MACsec 暗号スイートは GCM-AES-128 です。デバイスが GCM-AES-128 および GCM-AES-256 の両方の暗号方式をサポートしている場合は、ユーザー定義の MKA ポリシーを定義して使用し、必要に応じて、128 および 256 ビット両方の暗号を含めるか、または 256 ビットの暗号のみを含めることを推奨します。
ステップ 4	key-server priority 例： Device(config-mka-policy)# key-server priority 200	MKA キーサーバー オプションを設定し、0～255 の間で優先順位を設定します。 (注) キーサーバーの優先順位の値を 255 に設定した場合、ピアはキーサーバーになることはできません。キーサーバーの優先順位の値は MKA PSK に対してのみ有効です。これは MKA EAP-TLS には適用されません。
ステップ 5	send-secure-announcements 例： Device(config-mka-policy)# send-secure-announcements	セキュアアナウンスメントの送信を有効にします。セキュアアナウンスメントの送信を無効にするには、このコマンドの no 形式を使用します。デフォルトでは、セキュアアナウンスメントは無効になっています。
ステップ 6	macsec-cipher-suite {gcm-aes-128 gcm-aes-256} 例： Device(config-mka-policy)# macsec-cipher-suite gcm-aes-128	128 ビットまたは 256 ビット暗号化により SAK を取得するための暗号スイートを設定します。

	コマンドまたはアクション	目的
ステップ 7	end 例： Device(config-mka-policy)# end	MKA ポリシー コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 8	show mka policy 例： Device# show mka policy	MKA ポリシーを表示します。

セキュアアナウンスメントのグローバル設定

特権 EXEC モードから始めて、次の手順に従って、すべての MKA ポリシーにわたって安全なアナウンスメントをグローバルに有効にします。

手順の概要

1. **enable**
2. **configure terminal**
3. **mka defaults policy send-secure-announcements**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mka defaults policy send-secure-announcements 例： Device(config)# mka defaults policy send-secure-announcements	MKA ポリシーを介した MKPDU でのセキュアアナウンスメントの送信を有効にします。デフォルトでは、セキュアアナウンスメントは無効になっています。
ステップ 4	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

インターフェイスでの EAPOL アナウンスメントの設定

インターフェイスで EAPOL アナウンスメントを設定するには、特権 EXEC モードで開始し、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **eapol announcement**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 1/0/1	MACsec 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。インターフェイスは物理インターフェイスでなければなりません。
ステップ 4	eapol announcement 例： Device(config-if)# eapol announcement	EAPOL アナウンスメントを有効にします。EAPOL アナウンスメントを無効にするには、コマンドの no 形式を使用します。デフォルトでは、EAPOL アナウンスメントは無効になっています。
ステップ 5	end 例： Device(config-if)# configure terminal	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、特権 EXEC モードに戻ります。

Cisco TrustSec MACsec の設定

手動モードでの Cisco TrustSec スイッチ間リンク セキュリティの設定

始める前に

インターフェイスの Cisco TrustSec を手動で設定する場合は、次のような使用上の注意事項、および制約事項を考慮してください。

- SAP パラメータが定義されていない場合、Cisco TrustSec カプセル化または暗号化は行われません。
- SAP 動作モードとして GCM を選択すると、シスコの MACsec 暗号化ソフトウェア ライセンスが必要です。必要なライセンスなしで GCM を選択した場合、インターフェイスはリンク ダウン状態になります。
- これらの保護レベルは、SAP の Pairwise Master Key (sap pmk) を設定する場合にサポートされます。
 - SAP が設定されていない：保護は行われません。
 - **sap mode-list gcm-encrypt gmac no-encap**：保護が望ましいが必須ではない。
 - **sap mode-list gcm-encrypt gmac**：機密性が推奨され、完全性は必須。保護はサブリカントの設定に応じてサブリカントによって選択されます。
 - **sap mode-list gmac**：完全性のみ。
 - **sap mode-list gcm-encrypt**：機密性が必須。
 - **sap mode-list gmac gcm-encrypt**：完全性が必須であり推奨される。機密性は任意。
- MKA から Cisco TrustSec SAP (またはその逆) に設定を変更する前に、インターフェイスの設定を削除することを推奨します。

別の Cisco TrustSec デバイスへのインターフェイスで Cisco TrustSec を手動で設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface *interface-id***
3. **cts manual**
4. **sap pmk *key* [**mode-list** *mode1* [*mode2* [*mode3* [*mode4*]]]]**
5. **no propagate sgt**
6. **exit**
7. **end**
8. **show cts interface [*interface-id* | **brief** | **summary**]**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface interface-id 例： Device(config)# interface tengigabitethernet 1/1/2	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	cts manual 例： Device(config-if)# cts manual	Cisco TrustSec 手動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	sap pmk key [mode-list mode1 [mode2 [mode3 [mode4]]]] 例： Device(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt no-encap	<p>(任意) SAP の Pairwise Master Key (PMK) と動作モードを設定します。Cisco TrustSec の手動モードでは、SAP はデフォルトでディセーブルになっています。</p> <ul style="list-style-type: none"> • key : 文字数が偶数個で最大 32 文字の 16 進値。 <p>SAP 動作モードのオプションは次のとおりです。</p> <ul style="list-style-type: none"> • gcm-encrypt : 認証および暗号化 <ul style="list-style-type: none"> (注) ソフトウェアライセンスが MACsec 暗号化をサポートする場合、MACsec の認証と暗号化にこのモードを選択します。 • gmac : 認証、暗号化なし • no-encap : カプセル化なし
ステップ 5	no propagate sgt 例： Device(config-if-cts-manual)# no propagate sgt	ピアが SGT を処理できない場合、このコマンドの no 形式を使用します。 no propagate sgt コマンドを使用すると、インターフェイスからピアに SGT が送信されなくなります。
ステップ 6	exit 例： Device(config-if-cts-manual)# exit	Cisco TrustSec 802.1x 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。
ステップ 7	end 例： Device(config-if)# end	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	show cts interface [<i>interface-id</i> brief summary]	(任意) TrustSec 関連のインターフェイス特性を表示して、設定を確認します。
ステップ 9	copy running-config startup-config 例 : Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

MACsec 暗号化の設定例

以降のセクションでは、MACsec 暗号化の設定例を示します。

例 : MKA および MACsec の設定

次に、MKA ポリシーを作成する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# mka policy mka_policy
Device(config-mka-policy)# key-server priority 200
Device(config-mka-policy)# macsec-cipher-suite gcm-aes-128
Device(config-mka-policy)# confidentiality-offset 30
Device(config-mka-policy)# ssci-based-on-sci
Device(config-mka-policy)#end
```

次は、インターフェイスに MACsec を設定する例です。

```
Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# switchport access vlan 1
Device(config-if)# switchport mode access
Device(config-if)# macsec
Device(config-if)# authentication event linksec fail action authorize vlan 1
Device(config-if)# authentication host-mode multi-domain
Device(config-if)# authentication linksec policy must-secure
Device(config-if)# authentication port-control auto
Device(config-if)# authentication periodic
Device(config-if)# authentication timer reauthenticate
Device(config-if)# authentication violation protect
Device(config-if)# mka policy mka_policy
Device(config-if)# dot1x pae authenticator
Device(config-if)# spanning-tree portfast
Device(config-if)# end
```

例 : PSK を使用した MACsec MKA の設定

次に、PSK を使用して、MKA MACsec を設定する例を示します。

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

```

Device(config)# key chain keychain1 macsec
Device(config-keychain)# key 1000
Device(config-keychain-key)# cryptographic-algorithm aes-128-cmac
Device(config-keychain-key)# key-string 12345678901234567890123456789012
Device(config-keychain-key)# lifetime local 12:12:00 July 28 2016 12:19:00 July 28 2016
Device(config-keychain-key)# end

```

次に、PSK を使用して、インターフェイスに MACsec MKA を設定する例を示します。

```

Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# mka policy mka_policy
Device(config-if)# mka pre-shared-key key-chain key-chain-name
Device(config-if)# macsec replay-protection window-size 10
Device(config-if)# end

```

MKA-PSK : CKN 動作の変更

Cisco IOS XE Fuji 16.8.1 リリース以降、MKA PSK セッションの場合、CKN は、固定の 32 バイトではなく、キーの 16 進文字列として設定されている CKN とまったく同じ文字列を使用します。

```

Device> enable
Device# configure terminal
Device(config)# key chain abc macsec
Device(config-keychain)# key 11
Device(config-keychain-key)# cryptographic-algorithm aes-128-cmac
Device(config-keychain-key)# key-string 12345678901234567890123456789013
Device(config-keychain-key)# lifetime local 12:21:00 Sep 9 2015 infinite
Device(config-keychain-key)# end

```

以下は、上記の設定に対する `show mka session` コマンドの出力例です。

```

Device# show mka session

Total MKA Sessions..... 1
Secured Sessions... 1
Pending Sessions... 0

```

Interface	Local-TxSCI	Policy-Name	Inherited	Key-Server
Port-ID	Peer-RxSCI	MACsec-Peers	Status	CKN
Et0/0	aabb.cc00.6600/0002	icv	NO	NO
2	aabb.cc00.6500/0002 1		Secured	11 *Note that the CKN key-string is exactly the same that has been configured for the key as hex-string.*

一方で CKN 動作が変更され、もう一方で CKN 動作が変更されていない 2 つのイメージ間の相互運用性の場合、キーの 16 進数文字列は 64 文字の 16 進数文字列である必要があります。この文字列は、CKN 動作が変更されたイメージを持つデバイスで動作するようにゼロパディングされている必要があります。次の例を参照してください。

CKN キー文字列の動作が変更されていない設定 :

```

Device# configure terminal
Device(config)# key chain abc macsec
Device(config-keychain)# key 11
Device(config-keychain-key)# cryptographic-algorithm aes-128-cmac

```


例：PSK を使用したポートチャネルの MACsec MKA の設定

```

Device(config-mka-policy)# key-server priority 0
Device(config-mka-policy)# macsec-cipher-suite gcm-aes-128
Device(config-mka-policy)# confidentiality-offset 0
Device(config-mka-policy)# exit
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# channel-group 2 mode on
Device(config-if)# macsec network-link
Device(config-if)# mka policy POLICY
Device(config-if)# mka pre-shared-key key-chain KC
Device(config-if)# exit
Device(config)# interface gigabitethernet 1/0/2
Device(config-if)# channel-group 2 mode on
Device(config-if)# macsec network-link
Device(config-if)# mka policy POLICY
Device(config-if)# mka pre-shared-key key-chain KC
Device(config-if)# end

```

レイヤ 2 EtherChannel 設定

デバイス 1

```

Device> enable
Device# configure terminal
Device(config)# interface port-channel 2
Device(config-if)# switchport
Device(config-if)# switchport mode trunk
Device(config-if)# no shutdown
Device(config-if)# end

```

デバイス 2

```

Device> enable
Device# configure terminal
Device(config)# interface port-channel 2
Device(config-if)# switchport
Device(config-if)# switchport mode trunk
Device(config-if)# no shutdown
Device(config-if)# end

```

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

```

Flags:  D - down          P - bundled in port-channel
        I - stand-alone  s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port

        A - formed by Auto LAG

```

```

Number of channel-groups in use: 1
Number of aggregators:          1

```

```

Group  Port-channel  Protocol  Ports
-----+-----+-----+-----
2      Po2 (RU)         -         Te1/0/1 (P)  Te1/0/2 (P)

```

レイヤ 3 EtherChannel 設定

デバイス 1

```

Device> enable
Device# configure terminal
Device(config)# interface port-channel 2
Device(config-if)# no switchport
Device(config-if)# ip address 10.25.25.3 255.255.255.0
Device(config-if)# no shutdown
Device(config-if)# end

```

デバイス 2

```

Device> enable
Device# configure terminal
Device(config)# interface port-channel 2
Device(config-if)# no switchport
Device(config-if)# ip address 10.25.25.4 255.255.255.0
Device(config-if)# no shutdown
Device(config-if)# end

```

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

```

Flags:  D - down          P - bundled in port-channel
        I - stand-alone  s - suspended
        H - Hot-standby (LACP only)
        R - Layer3       S - Layer2
        U - in use       f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port

        A - formed by Auto LAG

```

```

Number of channel-groups in use: 1
Number of aggregators:          1

```

```

Group  Port-channel  Protocol  Ports
-----+-----+-----+-----
2      Po2 (RU)         -         Te1/0/1 (P)  Te1/0/2 (P)

```

EtherChannel モード - LACP

次に、EtherChannel モードが LACP のデバイス 1 およびデバイス 2 の設定例を示します。

```
Device> enable
Device# configure terminal
Device(config)# key chain KC macsec
Device(config-key-chain)# key 1000
Device(config-key-chain)# cryptographic-algorithm aes-128-cmac
Device(config-key-chain)# key-string FC8F5B10557C192F03F60198413D7D45
Device(config-key-chain)# exit
Device(config)# mka policy POLICY
Device(config-mka-policy)# key-server priority 0
Device(config-mka-policy)# macsec-cipher-suite gcm-aes-128
Device(config-mka-policy)# confidentiality-offset 0
Device(config-mka-policy)# exit
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# channel-group 2 mode active
Device(config-if)# macsec network-link
Device(config-if)# mka policy POLICY
Device(config-if)# mka pre-shared-key key-chain KC
Device(config-if)# exit
Device(config)# interface gigabitethernet 1/0/2
Device(config-if)# channel-group 2 mode active
Device(config-if)# macsec network-link
Device(config-if)# mka policy POLICY
Device(config-if)# mka pre-shared-key key-chain KC
Device(config-if)# end
```

レイヤ 2 EtherChannel 設定

デバイス 1

```
Device> enable
Device# configure terminal
Device(config)# interface port-channel 2
Device(config-if)# switchport
Device(config-if)# switchport mode trunk
Device(config-if)# no shutdown
Device(config-if)# end
```

デバイス 2

```
Device> enable
Device# configure terminal
Device(config)# interface port-channel 2
Device(config-if)# switchport
Device(config-if)# switchport mode trunk
Device(config-if)# no shutdown
Device(config-if)# end
```

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

```
Flags:  D - down          P - bundled in port-channel
        I - stand-alone  s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
```


例 : MACsec 暗号アナウンスメントの設定

```

Number of aggregators:          1

Group  Port-channel  Protocol  Ports
-----+-----+-----+-----
2      Po2 (RU)        PAgP      Te1/1/1 (P)  Te1/1/2 (P)

```

アクティブな MKA セッションの表示

次に、すべてのアクティブな MKA セッションを表示します。

```
Device# show mka sessions interface Te1/0/1
```

```

=====
Interface      Local-TxSCI          Policy-Name          Inherited
Key-Server
Port-ID        Peer-RxSCI           MACsec-Peers        Status
CKN
=====
Te1/0/1        00a3.d144.3364/0025 POLICY                NO
NO
37             701f.539b.b0c6/0032 1                Secured
1000

```

例 : MACsec 暗号アナウンスメントの設定

次に、セキュアアナウンスメントの MKA ポリシーの設定例を示します。

```

Device> enable
Device# configure terminal
Device(config)# mka policy mka_policy
Device(config-mka-policy)# key-server 2
Device(config-mka-policy)# send-secure-announcements
Device(config-mka-policy)# macsec-cipher-suite gcm-aes-128confidentiality-offset 0
Device(config-mka-policy)# end

```

次に、セキュアアナウンスメントのグローバル設定例を示します。

```

Device> enable
Device# configure terminal
Device(config)# mka defaults policy send-secure-announcements
Device(config)# end

```

次に、インターフェイスでの EAPoL アナウンスメントの設定例を示します。

```

Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# eapol announcement
Device(config-if)# end

```

次に、EAPoL アナウンスメントが有効になっている `show running-config interface interface-name` コマンドの出力例を示します。

```
Device# show running-config interface GigabitEthernet 1/0/1
```



```

Latest SAK KI (KN)..... D46CBEC05D5D67594543CEAE00000001 (1)
Old SAK Status..... FIRST-SAK
Old SAK AN..... 0
Old SAK KI (KN)..... FIRST-SAK (0)

SAK Transmit Wait Time... 0s (Not waiting for any peers to respond)
SAK Retire Time..... 0s (No Old SAK to retire)

MKA Policy Name..... p2
Key Server Priority..... 2
Delay Protection..... NO
Replay Protection..... YES
Replay Window Size..... 0
Confidentiality Offset... 0
Algorithm Agility..... 80C201
Send Secure Announcement.. DISABLED
SAK Cipher Suite..... 0080C20001000001 (GCM-AES-128)
MACsec Capability..... 3 (MACsec Integrity, Confidentiality, & Offset)
MACsec Desired..... YES

# of MACsec Capable Live Peers..... 1
# of MACsec Capable Live Peers Responded.. 1

Live Peers List:
  MI                               MN           Rx-SCI (Peer)           KS Priority
-----
  38046BA37D7DA77E06D006A9  89560       c800.8459.e764/002a    10

Potential Peers List:
  MI                               MN           Rx-SCI (Peer)           KS Priority
-----

Dormant Peers List:
  MI                               MN           Rx-SCI (Peer)           KS Priority
-----

```

次に、セキュアアナウンスメントが無効になっている **show mka policy policy-name detail** コマンドの出力例を示します。

```

Device# show mka policy p2 detail

MKA Policy Configuration ("p2")
=====
MKA Policy Name..... p2
Key Server Priority.... 2
Confidentiality Offset. 0
Send Secure Announcement..DISABLED

```



```

Delay Protection..... NO
Replay Protection..... YES
Replay Window Size..... 0
Confidentiality Offset... 0
Algorithm Agility..... 80C201
Send Secure Announcement.. DISABLED
SAK Cipher Suite..... 0080C20001000001 (GCM-AES-128)
MACsec Capability..... 3 (MACsec Integrity, Confidentiality, & Offset)
MACsec Desired..... YES

```

```

# of MACsec Capable Live Peers..... 1
# of MACsec Capable Live Peers Responded.. 1

```

Live Peers List:

MI	MN	Rx-SCI (Peer)	KS Priority
38046BA37D7DA77E06D006A9	89560	c800.8459.e764/002a	10

Potential Peers List:

MI	MN	Rx-SCI (Peer)	KS Priority
----	----	---------------	-------------

Dormant Peers List:

MI	MN	Rx-SCI (Peer)	KS Priority
----	----	---------------	-------------

次に、**show mka policy** コマンドの出力例を示します。

```
Device# show mka policy
```

```
MKA Policy Summary...
```

Policy Name	KS Priority	Delay Protect	Replay Protect	Window Size	Conf Offset	Cipher Suite(s)	Interfaces Applied
DEFAULT POLICY	0	FALSE	TRUE	0	0	GCM-AES-128	
p1	1	FALSE	TRUE	0	0	GCM-AES-128	
p2	2	FALSE	TRUE	0	0	GCM-AES-128	Gi1/0/1

次に、**show mka policy policy-name** コマンドの出力例を示します。

```
Device# show mka policy p2
```

```
MKA Policy Summary...
```

Policy Name	KS Priority	Delay Protect	Replay Protect	Window Size	Conf Offset	Cipher Suite(s)	Interfaces Applied
p2	2	FALSE	TRUE	0	0	GCM-AES-128	Gi1/0/1

次に、**show mka policy policy-name detail** コマンドの出力例を示します。

```
Device# show mka policy p2 detail
```

```
MKA Policy Configuration ("p2")
```

```

=====
MKA Policy Name..... p2
Key Server Priority.... 2
Confidentiality Offset. 0
Send Secure Announcement..DISABLED
Cipher Suite(s)..... GCM-AES-128

```



```

SA Statistics
SAKs Generated..... 1
SAKs Rekeyed..... 0
SAKs Received..... 0
SAK Responses Received..... 1

MKPDU Statistics
MKPDUs Validated & Rx..... 89589
  "Distributed SAK"..... 0
  "Distributed CAK"..... 0
MKPDUs Transmitted..... 89600
  "Distributed SAK"..... 1
  "Distributed CAK"..... 0

MKA Error Counter Totals
=====
Session Failures
  Bring-up Failures..... 0
  Reauthentication Failures..... 0
  Duplicate Auth-Mgr Handle..... 0

SAK Failures
  SAK Generation..... 0
  Hash Key Generation..... 0
  SAK Encryption/Wrap..... 0
  SAK Decryption/Unwrap..... 0
  SAK Cipher Mismatch..... 0

CA Failures
  Group CAK Generation..... 0
  Group CAK Encryption/Wrap..... 0
  Group CAK Decryption/Unwrap..... 0
  Pairwise CAK Derivation..... 0
  CKN Derivation..... 0
  ICK Derivation..... 0
  KEK Derivation..... 0
  Invalid Peer MACsec Capability... 0

MACsec Failures
  Rx SC Creation..... 0
  Tx SC Creation..... 0
  Rx SA Installation..... 0
  Tx SA Installation..... 0

MKPDU Failures
  MKPDU Tx..... 0
  MKPDU Rx Validation..... 0
  MKPDU Rx Bad Peer MN..... 0
  MKPDU Rx Non-recent Peerlist MN.. 0

```

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

```

Device# show macsec interface HundredGigE 2/0/4

MACsec is enabled
  Replay protect : enabled
  Replay window : 0
  Include SCI : yes
  Use ES Enable : no
  Use SCB Enable : no
  Admin Pt2Pt MAC : forceTrue(1)
  Pt2Pt MAC Operational : no
  Cipher : GCM-AES-128
  Confidentiality Offset : 0

Capabilities

```

```
ICV length : 16
Data length change supported: yes
Max. Rx SA : 16
Max. Tx SA : 16
Max. Rx SC : 8
Max. Tx SC : 8
Validate Frames : strict
PN threshold notification support : Yes
Ciphers supported : GCM-AES-128
                   GCM-AES-256
                   GCM-AES-XPN-128
                   GCM-AES-XPN-256

Access control : must secure

Transmit Secure Channels
SCI : 3C5731BBB5850475
SC state : inUse(1)
Elapsed time : 7w0d
Start time : 7w0d
Current AN: 0
Previous AN: -
Next PN: 149757
SA State: inUse(1)
Confidentiality : yes
SAK Unchanged : yes
SA Create time : 00:04:41
SA Start time : 7w0d
SC Statistics
  Auth-only Pkts : 0
  Auth-only Bytes : 0
  Encrypted Pkts : 0
  Encrypted Bytes : 0
SA Statistics
  Auth-only Pkts : 0
  Auth-only Bytes : 0
  Encrypted Pkts : 149756
  Encrypted Bytes : 16595088

Port Statistics
Egress untag pkts 0
Egress long pkts 0

Receive Secure Channels
SCI : 3C5731BBB5C504DF
SC state : inUse(1)
Elapsed time : 7w0d
Start time : 7w0d
Current AN: 0
Previous AN: -
Next PN: 149786
RX SA Count: 0
SA State: inUse(1)
SAK Unchanged : yes
SA Create time : 00:04:39
SA Start time : 7w0d
SC Statistics
  Notvalid pkts 0
  Invalid pkts 0
  Valid pkts 0
  Late pkts 0
  Uncheck pkts 0
  Delay pkts 0
  UnusedSA pkts 0
```

```

NousingSA pkts 0
Validated Bytes 0
Decrypted Bytes 0
SA Statistics
Notvalid pkts 0
Invalid pkts 0
Valid pkts 149784
Late pkts 0
Uncheck pkts 0
Delay pkts 0
UnusedSA pkts 0
NousingSA pkts 0
Validated Bytes 0
Decrypted Bytes 16654544

Port Statistics
Ingress untag pkts 0
Ingress notag pkts 631726
Ingress badtag pkts 0
Ingress unknownSCI pkts 0
Ingress noSCI pkts 0
Ingress overrun pkts 0

```

MACsec 暗号化に関する追加情報

標準および RFC

標準/RFC	タイトル
IEEE 802.1AE-2006	<i>Media Access Control (MAC) セキュリティ</i>
IEEE 802.1X-2010	ポート ベースのネットワーク アクセス コントロール
IEEE 802.1AEbw-2013	<i>Media Access Control (MAC) セキュリティ (IEEE 802.1AE-2006 の修正) : Extended Packet Numbering (XPN)</i>
IEEE 802.1Xbx-2014	ポートベースのネットワーク アクセス コントロール (<i>IEEE 802.1X-2010</i> の修正)
RFC 4493	<i>AES-CMAC</i> アルゴリズム

シスコのテクニカル サポート

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

MACsec 暗号化の機能情報

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

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

表 20: MACsec 暗号化の機能情報

機能名	リリース	機能情報
MACsec の暗号化	Cisco IOS XE Everest 16.5.1a	MACsec は 2 台の MACsec 対応デバイス間のパケットの認証と暗号化の IEEE 802.1AE 規格です。Catalyst スイッチは、スイッチとホストデバイス間の MACsec Key Agreement (MKA) 暗号化による 802.1AE 暗号化をサポートします。



第 11 章

セキュア シェルの設定

・セキュア シェルの設定 (275 ページ)

セキュア シェルの設定

セキュア シェル (SSH) は、Berkeley の r ツールへのセキュアな置換を提供するアプリケーションおよびプロトコルです。プロトコルは標準の暗号メカニズムを使用してセッションの安全を確保します。アプリケーションは Berkeley の rexec および rsh ツールと同様に使用できます。2つのバージョンの SSH (SSH バージョン 1 と SSH バージョン 2) を使用できます。特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。SSH バージョン 2 については、「セキュアシェルバージョン 2 サポート」機能モジュールを参照してください。

セキュア シェルを設定するための前提条件



(注) 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

- SSH を動作させるには、スイッチに Rivest、Shamir、および Adleman (RSA) の公開キーと秘密キーのペアが必要です。これは SSH が必要なセキュア コピー プロトコル (SCP) も同様で、セキュアな転送を実現させるには、これらのキーのペアが必要です。
- デバイスに必要なイメージをダウンロードします。セキュア シェル (SSH) サーバは、IPsec (データ暗号規格 (DES) または 3DES) の暗号化ソフトウェアイメージを必要とします。SSH クライアントは、IPsec (DES または 3DES) の暗号化ソフトウェアイメージが必要です。
- グローバル コンフィギュレーション モードで **hostname** および **ip domain name** コマンドを使用して、デバイスのホスト名とホストドメインを設定します。
- デバイスの Rivest、Shamir and Adleman (RSA) キーペアを生成します。グローバル コンフィギュレーション モードで **crypto key generate rsa** コマンドを入力すると、このキーペアによって SSH とリモート認証が自動的に有効になります。



(注) RSA キーのペアを削除するには、**crypto key zeroize rsa** グローバル コンフィギュレーション コマンドを使用します。RSA キー ペアを削除すると、SSH サーバーは自動的にディセーブルになります。

- ローカルアクセスまたはリモートアクセス用にユーザー認証を設定します。認証、許可、アカウントिंग (AAA) の有無に関係なく、認証を設定できます。
- セキュア シェル (SSH) サーバーは、IPsec (データ暗号規格 (DES) または 3DES) の暗号化ソフトウェア イメージを必要とします。SSH クライアントは、IPsec (DES または 3DES) の暗号化ソフトウェア イメージが必要です。

セキュア シェルの設定に関する制約事項



(注) 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

- セキュアシェル (SSH) サーバーと SSH クライアントは、Data Encryption Standard (DES) (56 ビット) および 3DES (168 ビット) データ暗号化ソフトウェア イメージのみでサポートされます。DES ソフトウェア イメージの場合、使用できる暗号化アルゴリズムは DES だけです。3DES ソフトウェア イメージの場合、DES と 3DES の両方の暗号化アルゴリズムを使用できます。
- 実行シェルは、唯一サポートされるアプリケーションです。
- ログインバナーはセキュア シェルバージョン 1 ではサポートされません。セキュア シェルバージョン 2 ではサポートされています。
- SFTP サーバーはサポートされていません。

セキュア シェルの設定について

セキュアシェル (SSH) は、デバイスに対する安全なリモート接続を可能にするプロトコルです。SSH は、デバイスの認証時に強力な暗号化を行うことで、リモート接続について Telnet 以上のセキュリティを実現します。このソフトウェア リリースは、SSH バージョン 2 (SSHv2) をサポートします。

SSH サーバ



(注) 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

セキュアシェル (SSH) サーバー機能を使用すると、SSHクライアントはシスコデバイスとの間で、セキュアな暗号化された接続を確立できます。この接続は、インバウンドTelnet接続の機能と同様です。SSH以前は、セキュリティはTelnetのセキュリティに限定されていました。SSHをCiscoソフトウェアの認証と併用することで、強力な暗号化が可能になります。CiscoソフトウェアのSSHサーバーは、市販の一般的なSSHクライアントと相互運用できます。

SSH 統合クライアント



(注) 特に明記しない限り、「SSH」という用語は「SSHバージョン1」のみを意味します。

セキュアシェル (SSH) 統合クライアント機能は、SSHプロトコル上で動作し、デバイスの認証および暗号化を実現するアプリケーションです。SSHクライアントによって、シスコデバイスは別のシスコデバイスなどSSHサーバを実行するデバイスに対して、セキュアで暗号化された接続を実行できます。この接続は、接続が暗号化される点を除いてTelnetのアウトバウンド接続と同様の機能を提供します。SSHクライアントは、認証および暗号化により、保護されていないネットワーク上でもセキュアな通信ができます。

シスコソフトウェアのSSHクライアントは、市販の一般的なSSHサーバーと使用します。SSHクライアントは、Data Encryption Standard (DES)、3DES、およびパスワード認証の暗号をサポートします。ユーザー認証は、デバイスに対するTelnetセッションの認証と同様に実行されます。SSHがサポートするユーザー認証メカニズムには、Remote Authentication Dial-In User Service (RADIUS)、TACACS+、およびローカルに格納されたユーザー名とパスワードを使用した認証があります。



(注) SSHクライアント機能を使用できるのは、SSHサーバーがイネーブルの場合だけです。

RSA 認証のサポート

セキュアシェル (SSH) クライアントで使用できるRivest, Shamir, Adleman (RSA) 認証は、CiscoソフトウェアのSSHサーバーではデフォルトでサポートされていません。RSA認証サポートの詳細については、「セキュアシェルバージョン2サポート」の「RSAペアを使用したSSHバージョン2のデバイス設定」セクションを参照してください。

SSH サーバ、統合クライアント、およびサポートされているバージョン

セキュアシェル (SSH) 統合クライアント機能は、SSHプロトコル上で動作し、デバイスの認証および暗号化を実現するアプリケーションです。SSHクライアントによって、シスコデバイスは別のシスコデバイスなどSSHサーバを実行するデバイスに対して、セキュアで暗号化された接続を実行できます。この接続は、接続が暗号化される点を除いてTelnetのアウトバウンド接続と同様の機能を提供します。SSHクライアントは、認証および暗号化により、保護されていないネットワーク上でもセキュアな通信ができます。

SSH サーバおよび SSH 統合クライアントは、スイッチ上で実行されるアプリケーションです。SSH サーバは、このリリースでサポートされている SSH クライアントおよび、他社製の SSH クライアントと使用します。SSH クライアントは、市販の一般的な SSH サーバと連動します。SSH クライアントは、Data Encryption Standard (DES)、3DES、およびパスワード認証の暗号をサポートします。



(注) SSH クライアント機能を使用できるのは、SSH サーバがイネーブルの場合だけです。

ユーザ認証は、デバイスに対する Telnet セッションの認証と同様に実行されます。SSH は、次のユーザ認証方式もサポートします。

- TACACS+
- RADIUS
- ローカル認証および許可

SSH 設定時の注意事項

スイッチを SSH サーバまたは SSH クライアントとして設定する場合は、次の注意事項に従ってください。

- SSHv2 サーバは、SSHv1 サーバで生成される RSA キーのペアを使用できます（逆の場合も同様です）。
- SSH サーバがアクティブスイッチ上で動作しており、アクティブスイッチに障害が発生した場合、新しいアクティブスイッチは、以前のアクティブスイッチによって生成された RSA キーペアを使用します。
- **crypto key generate rsa** グローバル コンフィギュレーション コマンドを入力した後、CLI エラー メッセージが表示される場合、RSA キーペアは生成されていません。ホスト名およびドメインを再設定してから、**crypto key generate rsa** コマンドを入力してください。
- RSA キーのペアを生成する場合に、メッセージ「No host name specified」が表示されることがあります。このメッセージが表示された場合は、グローバルコンフィギュレーション モードで **hostname** コマンドを使用してホスト名を設定する必要があります。
- RSA キーのペアを生成する場合に、メッセージ「No domain specified」が表示されることがあります。このメッセージが表示された場合は、グローバル コンフィギュレーション モードで **ip domain name** コマンドを使用して IP ドメイン名を設定する必要があります。
- ローカル認証および許可の方法を設定する場合に、コンソール上で AAA がディセーブルにされていることを確認してください。

セキュア シェルの設定方法

SSH を実行するためのデバイスの設定

SSH を実行するようにデバイスをセットアップするには、次の手順を実行してください。

Before you begin

ローカルアクセスまたはリモートアクセス用にユーザ認証を設定します。この手順は必須です。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname *hostname***
4. **ip domain name *domain_name***
5. **crypto key generate rsa**
6. **exit**
7. **show ip ssh**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	hostname <i>hostname</i> Example: Device(config)# hostname your_hostname	device のホスト名および IP ドメイン名を設定します。 Note この手順を実行するのは、device を SSH サーバとして設定する場合だけです。
ステップ 4	ip domain name <i>domain_name</i> Example: Device(config)# ip domain name your_domain	device のホストドメインを設定します。
ステップ 5	crypto key generate rsa Example: Device(config)# crypto key generate rsa	device 上でローカルおよびリモート認証用に SSH サーバをイネーブルにし、RSA キーペアを生成します。device の RSA キーペアを生成すると、SSH が自動的にイネーブルになります。

	Command or Action	Purpose
		<p>最小モジュラス サイズは、1024 ビットにすることを推奨します。</p> <p>RSA キーのペアを生成する場合に、モジュラスの長さの入力を求められます。モジュラスが長くなるほど安全ですが、生成と使用に時間がかかります。</p> <p>Note この手順を実行するのは、deviceを SSH サーバとして設定する場合だけです。</p>
ステップ 6	exit Example: Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 7	show ip ssh Example: Device# show ip ssh	(任意) SSHサーバーが有効であることを確認し、SSH接続のバージョンおよび設定データを表示します。

SSH サーバーの設定



(注) 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh { time-out seconds | authentication-retries integer }**
4. **ip ssh rekey { time time | volume volume }**
5. **exit**
6. **show ip ssh**

手順の詳細

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

	コマンドまたはアクション	目的
	Device# configure terminal	
ステップ 3	ip ssh { time-out <i>seconds</i> authentication-retries <i>integer</i>} 例 : Device(config)# ip ssh time-out 30	セキュアシェル (SSH) 制御パラメータを設定します。 (注) このコマンドは、ユーザーに表示するパスワードプロンプトの回数を設定するためにも使用できます。この数値は、次の2つの値の低い方です。 <ul style="list-style-type: none"> • ssh -o numberofpasswordprompt コマンドを使用してクライアントから提案された値。 • ip ssh authentication-retries <i>integer</i> コマンドを使用してデバイスに設定された値に 1 を加えた値。
ステップ 4	ip ssh rekey { time <i>time</i> volume <i>volume</i>} 例 : Device(config)# ip ssh rekey time 108	(任意) SSH の時間ベースのキー再生成またはボリュームベースのキー再生成を設定します。
ステップ 5	exit 例 : Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show ip ssh 例 : Device# show ip ssh	(任意) SSH サーバーが有効であることを確認し、SSH 接続のバージョンおよび設定データを表示します。

SSH クライアントの呼び出し



(注) 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

セキュアシェル (SSH) クライアントを呼び出すには、次の作業を実行します。SSH クライアントはユーザー EXEC モードで実行されます。設定作業は特にありません。

手順の概要

1. **enable**
2. **ssh -l *username* -vrf *vrf-name* *ip-address***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	ssh -l username -vrf vrf-name ip-address 例： Device# ssh -l user1 -vrf vrf1 192.0.2.1	SSH クライアントを呼び出し、指定した仮想ルーティングおよび転送 (VRF) インスタンスの IP ホストまたはアドレスに接続します。

セキュア シェルの設定例

例：SSH サーバーの設定



(注) 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

次に、サーバーに設定されたセキュアシェル (SSH) 制御パラメータの例を示します。この例では、30 秒のタイムアウト間隔が指定されています。このタイムアウト間隔は、SSH ネゴシエーションフェーズで使用されます。

```
Device> enable
Device# configure terminal
Device(config)# ip ssh timeout 30
Device(config)# end
```

例：SSH クライアントの呼び出し



(注) 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

次の例では、指定された Virtual Routing and Forwarding (VRF) インスタンスの IP アドレス 192.0.2.1 に接続するためにセキュアシェル (SSH) クライアントが呼び出されています。

```
Device> enable
Device# ssh -l user1 -vrf vrf1 192.0.2.1
```

例：SSH の確認



(注) 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

セキュアシェル (SSH) サーバが有効であることを確認し、SSH接続のバージョンおよび設定データを表示するには、**show ip ssh** コマンドを使用します。次に、SSHがイネーブルの例を示します。

```
Device# show ip ssh

SSH Enabled - version 1.5
Authentication timeout: 120 secs; Authentication retries: 3
```

次に、SSHがディセーブルの例を示します。

```
Device# show ip ssh

%SSH has not been enabled
```

SSHサーバ接続のステータスを確認するには、**show ssh** コマンドを使用します。次に、SSHを有効にしたときのデバイス上のSSHサーバ接続の例を示します。

```
Device# show ssh

Connection      Version      Encryption State Username
0 1.5 3DES Session Started guest
```

次に、SSHがディセーブルの例を示します。

```
Device# show ssh

%No SSH server connections running.
```

セキュア シェルに関するその他の参考資料

関連資料

関連項目	マニュアルタイトル
SSHバージョン2	『セキュリティコンフィギュレーションガイド』の「セキュアシェルバージョン2サポート」

シスコのテクニカル サポート

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

セキュアシェルの設定の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	セキュア シェル	SSH は、デバイスに対する安全なリモート接続を可能にするプロトコルです。SSH は、デバイスの認証時に強力な暗号化を行うことで、リモート接続について Telnet 以上のセキュリティを実現します。

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



第 12 章

セキュア シェルバージョン 2 サポート

セキュア シェルバージョン 2 サポート機能で、セキュア シェル (SSH) バージョン 2 を設定できます (SSH バージョン 1 サポートは、以前のシスコ ソフトウェア リリースに実装されていました)。SSH は、信頼性の高いトランスポート層の上部で実行され、強力な認証機能と暗号化機能を提供します。SSH では、信頼できる転送として定義されているのは TCP のみです。SSH で、ネットワーク上の他のコンピュータに安全にアクセスしたり、コマンドを安全に実行できます。SSH とともに提供されるセキュア コピー プロトコル (SCP) 機能で、ファイルを安全に転送できます。

- [セキュア シェルバージョン 2 サポートの前提条件 \(285 ページ\)](#)
- [セキュア シェルバージョン 2 サポートの制約事項 \(286 ページ\)](#)
- [セキュア シェルバージョン 2 サポートに関する情報 \(286 ページ\)](#)
- [セキュア シェルの設定方法 \(289 ページ\)](#)
- [セキュア シェルバージョン 2 サポートの設定例 \(303 ページ\)](#)
- [セキュア シェルバージョン 2 サポートの追加情報 \(307 ページ\)](#)
- [セキュア シェルバージョン 2 サポートの機能履歴 \(308 ページ\)](#)

セキュア シェルバージョン 2 サポートの前提条件

- SSH を設定する前に、ご使用のデバイスに必要なイメージがロードされていることを確認します。SSH サーバーには、ご使用のリリースに応じた k9 (Triple Data Encryption Standard [3DES]) ソフトウェア イメージが必要です。
- SSH バージョン 2 をサポートする SSH リモート デバイスを使用する必要があります。また、シスコ デバイスに接続する必要があります。
- SCP は、認証、認可、およびアカウンティング (AAA) によって正しく機能します。そのため、SSH サーバーで Secure Copy Protocol が有効になるようにデバイスで AAA を設定する必要があります。



- (注) SSHバージョン2サーバーとSSHバージョン2クライアントは、ご使用のリリースに応じてシスコソフトウェアでサポートされます（SSHクライアントはSSHバージョン1プロトコルとSSHバージョン2プロトコルの両方を実行します。SSHクライアントは、ご使用のリリースに応じてk9イメージでサポートされます）。

セキュア シェルバージョン2サポートの制約事項

- セキュア シェル (SSH) サーバーと SSH クライアントは、Triple Data Encryption Standard (3DES) ソフトウェア イメージでサポートされます。
- サポートされるアプリケーションは、実行シェル、remote コマンドの実行、Secure Copy Protocol (SCP) のみです。
- Rivest、Shamir、および Adleman (RSA) キー生成は SSH サーバー側の要件です。SSH クライアントとして動作するデバイスは、RSA キーを生成する必要がありません。
- RSA キー ペアのサイズは、768 ビット以上である必要があります。
- 次の機能はサポートされていません。
 - ポート フォワーディング。
 - Compression

セキュア シェルバージョン2サポートに関する情報

SSHバージョン2

セキュア シェルバージョン2サポート機能で、SSHバージョン2を設定できます。

SSHバージョン2サーバの設定は、SSHバージョン1の設定と同様です。**ip ssh version** コマンドは、設定するSSHバージョンを定義します。このコマンドを設定しない場合、デフォルトでSSHは互換モードで実行されます。バージョン1とバージョン2両方の接続が利用できます。



- (注) SSHバージョン1は、標準として定義されていないプロトコルです。未定義のプロトコル（バージョン1）にデバイスがフォールバックしないようにするには、**ip ssh version** コマンドを使用してバージョン2を指定する必要があります。

ip ssh rsa keypair-name コマンドを使用すると、設定した Rivest、Shamir、および Adleman (RSA) キーを使用して SSH 接続を実行できます。すでに、SSH は生成済みの最初の RSA

キーにリンクされています（つまり、最初の RSA キー ペアが生成された時点で SSH はイネーブルになっています）。この動作は存在していますが、**ip ssh rsa keypair-name** コマンドを使用してこの動作を行わないようにすることができます。**ip ssh rsa keypair-name** コマンドをキーペアの名前を指定して設定すると、SSH は、キーペアが存在する場合に有効になるか、キーペアを後で作成する場合は後から有効になります。このコマンドを使用して SSH をイネーブルにする場合、Cisco ソフトウェアの SSH バージョン 1 では必要な、ホスト名とドメイン名を設定する必要はありません。



(注) ログインバナーは SSH バージョン 2 でサポートされますが、セキュア シェルバージョン 1 ではサポートされません。

セキュア シェルバージョン2の機能拡張

SSH バージョン 2 の機能拡張には、Virtual Routing and Forwarding (VRF) -Aware SSH、SSH デバッグ機能拡張、および Diffie-Hellman (DH) グループ交換のサポートなどの追加機能がいくつか含まれています。



(注) VRF-Aware SSH 機能は、ご使用のリリースに応じてサポートされます。

Cisco SSH 実装では従来、768 ビット絶対値が使用されていましたが、DH グループ 14 (2048 ビット) およびグループ 16 (4096 ビット) 暗号化アプリケーションに対応するため、より大きなキーサイズの必要性が高まり、優先 DH グループを確立するクライアントとサーバー間のメッセージ交換が必要になっています。**ip ssh dh min size** コマンドは、SSH サーバー上のモジュラスサイズを設定します。これに加え、**ssh** コマンドが拡張され、SSH クライアント側のクライアントの VRF インスタンス名を IP アドレスとともに使用して、正しいルーティングテーブルを検索し、接続を確立する機能に、VRF 認識が追加されました。

SSH debug コマンドが修正され、デバッグが拡張されました。**debug ip ssh** コマンドは、デバッグプロセスを簡素化するために拡張されました。デバッグプロセスを簡素化する前、このコマンドでは、明確に必要なかどうかに関係なく SSH に関連するすべてのデバッグメッセージが印刷されました。この動作は依然として存在しますが、**debug ip ssh** コマンドをキーワードを指定して設定した場合、メッセージはキーワードで指定した情報に制限されます。

セキュア シェルバージョン2の RSA キーに関する機能拡張

Cisco SSH バージョン 2 は、キーボードインタラクティブ認証方式およびパスワードベースの認証方式をサポートしています。RSA キーの SSH バージョン 2 拡張機能は、クライアントとサーバ向けの RSA ベースの公開キー認証もサポートしています。

- ユーザー認証：RSA ベースのユーザー認証は、各ユーザーに関連付けられている秘密キー/公開キーのペアを認証に使用します。ユーザは秘密キー/公開キーのペアをクライアントで生成し、公開キーを Cisco SSH サーバで設定して、認証を完了します。

クレデンシャルの確立を試行する SSH ユーザは、秘密キーを使用して暗号化された署名を提示します。署名とユーザの公開キーは、認証のために SSH サーバに送信されます。SSH サーバでは、ユーザから提示された公開キーに対してハッシュを計算します。ハッシュは、サーバに一致するエントリがあるかどうかを判断するために使用されます。一致が見つかった場合、RSA ベースのメッセージ検証が公開キーを使用して実行されます。その結果、暗号化されたシグニチャに基づいて、ユーザのアクセスは認証されるか拒否されます。

- サーバ認証：SSH セッションの確立中に、Cisco SSH クライアントは、キー交換フェーズ中に使用できるサーバホストキーを使用して、SSH サーバを認証します。SSH サーバキーは、SSH サーバの識別に使用されます。これらのキーは SSH がイネーブルになるときに作成され、クライアント側で設定する必要があります。

サーバ認証の場合、Cisco SSH クライアントが各サーバにホストキーを割り当てる必要があります。クライアントがサーバとの間で SSH セッションを確立しようとする時、クライアントはキー交換メッセージの一部として、サーバの署名を受信します。厳密なホストキーのチェックフラグがクライアント側でイネーブルの場合、そのサーバに対応するホストキーエントリがあるかどうかクライアントで確認されます。一致が見つかったら、クライアントはサーバホストキーを使用して署名の検証を試行します。サーバの認証に成功すると、セッションの確立処理は続行します。失敗すると、処理は終了し、「Server Authentication Failed」というメッセージが表示されます。



- (注)
- 公開キーをサーバで格納する際、メモリを使用します。したがって、SSH サーバで設定できる公開キーの数は、1 ユーザに最大2つの公開キーを作成した場合10 ユーザ分に限られます。
 - シスコサーバは RSA ベースのユーザ認証をサポートしていますが、シスコクライアントは認証方式として公開キーを提案できません。RSA ベースの認証に対するオープンな SSH クライアントからの要求を Cisco サーバが受信した場合、サーバは認証要求を受け入れません。
 - サーバ認証の場合、サーバの RSA 公開キーを手動で設定し、Cisco SSH クライアント側で `ip ssh stricthostkeycheck` コマンドを設定します。

SSH およびスイッチ アクセス

セキュアシェル (SSH) は、デバイスに対する安全なリモート接続を可能にするプロトコルです。SSH は、デバイスの認証時に強力な暗号化を行うことで、リモート接続について Telnet 以上のセキュリティを実現します。このソフトウェアリリースは、SSH バージョン2 (SSHv2) をサポートします。

IPv6 の SSH 機能は IPv4 における機能と同じです。IPv6 の場合、SSH は IPv6 アドレスをサポートし、IPv6 トランスポート上において、リモート IPv6 ノードとのセキュリティ保護および暗号化された接続を有効化します。

SNMP トラップ生成

ご使用のリリースに応じて、簡易ネットワーク管理プロトコル (SNMP) トラップは、トラップが有効で SNMP デバッグがオンになっている場合、SSH セッションが終了した際に自動的に生成されます。



(注) **snmp-server host** コマンドを設定する場合、IP アドレスは、SSH (telnet) クライアントがあり、SSH サーバへの IP 接続が可能な PC のアドレスにする必要があります。

また、**debug snmp packet** コマンドを使用して SNMP デバッグを有効にし、トラップを表示する必要があります。トラップ情報には、送信バイト数や SSH セッションで使用されたプロトコルなどの情報が含まれます。

SSH キーボードインタラクティブ認証

SSH キーボードインタラクティブ認証機能は、SSH での汎用メッセージ認証とも呼ばれ、異なる種類の認証メカニズムを実装するために使用できる方式です。基本的に、現在サポートされている、ユーザの入力のみが必要な認証方式はすべて、この機能で実行することができます。この機能は自動的にイネーブルになります。

次の方式がサポートされています。

- Password
- サーバが送信するチャレンジに応答する番号またはストリングを印刷する SecurID およびハードウェア トークン
- プラグイン可能な認証モジュール (PAM)
- S/KEY (およびその他の使い捨てキー)

セキュア シェルの設定方法

ホスト名およびドメイン名を使用した SSH バージョン2 のデバイス設定

手順の概要

1. **enable**
2. **configure terminal**
3. **hostname *name***
4. **ip domain name *name***
5. **crypto key generate rsa**

6. `ip ssh [time-out seconds | authentication-retries integer]`
7. `ip ssh version [1 | 2]`
8. `exit`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	hostname name 例： Device(config)# hostname catalyst9k	デバイスのホスト名を設定します。
ステップ 4	ip domain name name 例： catalyst9k(config)# ip domain name example.com	デバイスのドメイン名を設定します。
ステップ 5	crypto key generate rsa 例： catalyst9k(config)# crypto key generate rsa	ローカルおよびリモート認証用に SSH サーバをイネーブルにします。
ステップ 6	ip ssh [time-out seconds authentication-retries integer] 例： catalyst9k(config)# ip ssh time-out 120	(任意) デバイス上で SSH 制御変数を設定します。
ステップ 7	ip ssh version [1 2] 例： catalyst9k(config)# ip ssh version 1	(任意) デバイスで実行する SSH のバージョンを指定します。
ステップ 8	exit 例： catalyst9k(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。 <ul style="list-style-type: none">• デフォルトホストに戻るには、no hostname コマンドを使用します。

RSA キー ペアを使用した SSH バージョン2 のデバイス設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh rsa keypair-name** *keypair-name*
4. **crypto key generate rsa** **usage-keys** **label** *key-label* **modulus** *modulus-size*
5. **ip ssh** [*time-out seconds* | **authentication-retries** *integer*]
6. **ip ssh version** **2**
7. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip ssh rsa keypair-name <i>keypair-name</i> 例： Device(config)# ip ssh rsa keypair-name sshkeys	SSH に使用する RSA キー ペアを指定します。 (注) シスコ デバイスには複数の RSA キー ペアを設定できます。
ステップ 4	crypto key generate rsa usage-keys label <i>key-label</i> modulus <i>modulus-size</i> 例： Device(config)# crypto key generate rsa usage-keys label sshkeys modulus 768	デバイスでローカルおよびリモート認証を行う SSH サーバを有効にします。 • SSH バージョン 2 では、絶対サイズは 768 ビット以上である必要があります。 (注) RSA キー ペアを削除するには、 crypto key zeroize rsa コマンドを使用します。RSA キー ペアを削除すると、SSH サーバは自動的に無効になります。
ステップ 5	ip ssh [<i>time-out seconds</i> authentication-retries <i>integer</i>] 例： Device(config)# ip ssh time-out 12	デバイス上で SSH 制御変数を設定します。
ステップ 6	ip ssh version 2 例：	デバイスで実行する SSH のバージョンを指定します。

	コマンドまたはアクション	目的
	Device(config)# ip ssh version 2	
ステップ 7	exit 例 : Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

RSA ベースのユーザ認証を実行するための Cisco SSH サーバの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **hostname name**
4. **ip domain name name**
5. **crypto key generate rsa**
6. **ip ssh pubkey-chain**
7. **username username**
8. **key-string**
9. **key-hash key-type key-name**
10. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	hostname name 例 : Device(config)# hostname host1	ホスト名を指定します。
ステップ 4	ip domain name name 例 : host1(config)# ip domain name name1	Cisco ソフトウェアで使用するデフォルトのドメイン名を定義し、不完全なホスト名のドメインを補完します。
ステップ 5	crypto key generate rsa 例 :	RSA キー ペアを生成します。

	コマンドまたはアクション	目的
	host1 (config) # crypto key generate rsa	
ステップ 6	ip ssh pubkey-chain 例 : host1 (config) # ip ssh pubkey-chain	SSH サーバ上のユーザおよびサーバ認証用に SSH-RSA キーを設定し、公開キー コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> サーバに保存されている RSA 公開キーが、クライアントに保存されている公開キーと秘密キーのペアを使用して検証されると、ユーザ認証は成功です。
ステップ 7	username username 例 : host1 (conf-ssh-pubkey) # username user1	SSH ユーザ名を設定し、公開キー ユーザ コンフィギュレーション モードを開始します。
ステップ 8	key-string 例 : host1 (conf-ssh-pubkey-user) # key-string	リモートピアの RSA 公開キーを指定し、公開キー データ コンフィギュレーション モードを開始します。 (注) オープン SSH クライアントから (言い換えると .ssh/id_rsa.pub ファイルから) 公開キー値を取得できます。
ステップ 9	key-hash key-type key-name 例 : host1 (conf-ssh-pubkey-data) # key-hash ssh-rsa key1	(任意) SSH キー タイプとバージョンを指定します。 <ul style="list-style-type: none"> 秘密キー/公開キーペアの設定では、キータイプを ssh-rsa にする必要があります。 key-string コマンドが設定されている場合に限りこの手順は任意です。 key-string コマンドと key-hash コマンドのいずれかを設定する必要があります。 (注) 公開キー ストリングのハッシュを計算するには、ハッシュ処理ソフトウェアを使用します。また、別のシスコデバイスからハッシュ値をコピーすることもできます。初めて公開キー データを入力する場合、 key-string コマンドを使用して公開キー データを入力することを推奨します。
ステップ 10	end 例 :	公開キー データ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	host1(conf-ssh-pubkey-data) # end	<ul style="list-style-type: none"> デフォルトホストに戻るには、no hostname コマンドを使用します。

RSA ベースのサーバ認証を実行するための Cisco IOS SSH サーバの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **hostname name**
4. **ip domain name name**
5. **crypto key generate rsa**
6. **ip ssh pubkey-chain**
7. **server server-name**
8. **key-string**
9. **exit**
10. **key-hash key-type key-name**
11. **end**
12. **configure terminal**
13. **ip ssh stricthostkeycheck**
14. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	hostname name 例： Device(config)# hostname host1	ホスト名を指定します。
ステップ 4	ip domain name name 例： host1(config)# ip domain name name1	Cisco ソフトウェアで使用するデフォルトのドメイン名を定義し、不完全なホスト名のドメインを補完します。

	コマンドまたはアクション	目的
ステップ 5	crypto key generate rsa 例： host1(config)# crypto key generate rsa	RSA キー ペアを生成します。
ステップ 6	ip ssh pubkey-chain 例： host1(config)# ip ssh pubkey-chain	SSH サーバ上のユーザおよびサーバ認証用に SSH-RSA キーを設定し、公開キー コンフィギュレーション モードを開始します。
ステップ 7	server server-name 例： host1(conf-ssh-pubkey)# server server1	デバイスでの公開キー認証について SSH サーバを有効にし、公開キー サーバ コンフィギュレーション モードを開始します。
ステップ 8	key-string 例： host1(conf-ssh-pubkey-server)# key-string	リモートピアの RSA 公開キーを指定し、公開キーデータ コンフィギュレーション モードを開始します。 (注) オープン SSH クライアントから（言い換えると .ssh/id_rsa.pub ファイルから）公開キー値を取得できます。
ステップ 9	exit 例： host1(conf-ssh-pubkey-data)# exit	公開キーデータ コンフィギュレーション モードを終了し、公開キー サーバ コンフィギュレーション モードを開始します。
ステップ 10	key-hash key-type key-name 例： host1(conf-ssh-pubkey-server)# key-hash ssh-rsa key1	(任意) SSH キー タイプとバージョンを指定します。 <ul style="list-style-type: none"> • 秘密キー/公開キーペアの設定では、キータイプを ssh-rsa にする必要があります。 • key-string コマンドが設定されている場合に限りこの手順は任意です。 • key-string コマンドと key-hash コマンドのいずれかを設定する必要があります。 (注) 公開キー スtring のハッシュを計算するには、ハッシュ処理ソフトウェアを使用します。また、別のシスコデバイスからハッシュ値をコピーすることもできます。初めて公開キーデータを入力する場合、 key-string コマンドを使用して公開キーデータを入力することを推奨します。

	コマンドまたはアクション	目的
ステップ 11	end 例： host1(conf-ssh-pubkey-server)# end	公開キー サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 12	configure terminal 例： host1# 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 13	ip ssh stricthostkeycheck 例： host1(config)# ip ssh stricthostkeycheck	サーバ認証が実行されることを確認します。 <ul style="list-style-type: none"> • 障害が発生すると、接続は終了します。 • デフォルトホストに戻るには、no hostname コマンドを使用します。
ステップ 14	end 例： host1(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

リモート デバイスとの暗号化セッションの開始



(注) 接続するデバイスは、シスコ ソフトウェアでサポートされる暗号化アルゴリズムを備えたセキュアシェル (SSH) サーバをサポートする必要があります。また、デバイスを有効にする必要はありません。SSH はディセーブルモードで実行できます。

手順の概要

1. **enable**
2. **ssh [-v {1 | 2}] [-c {aes128-ctr | aes192-ctr | aes256-ctr | aes128-cbc | 3des | aes192-cbc | aes256-cbc} | -I user-id | -I user-id:vrf-name number ip-address ip-address | -I user-id:rotary number ip-address | -m {hmac-md5-128 | hmac-md5-96 | hmac-sha1-160 | hmac-sha1-96} | -o numberofpasswordprompts n | -p port-num] {ip-addr | hostname} [command | -vrf]**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	<pre>ssh [-v {1 2} -c {aes128-ctr aes192-ctr aes256-ctr aes128-cbc 3des aes192-cbc aes256-cbc} -l user-id -l user-id:vrf-name number ip-address ip-address -l user-id:rotary number ip-address -m {hmac-md5-128 hmac-md5-96 hmac-sha1-160 hmac-sha1-96} -o numberofpasswordprompts n -p port-num] {ip-addr hostname} [command -vrf]</pre> <p>例 :</p> <pre>Device# ssh -v 2 -c aes256-ctr -m hmac-sha1-96 -l user2 10.76.82.24</pre>	リモート ネットワーク デバイスとの暗号化されたセッションを開始します。

セキュア シェル接続のステータスの確認

手順の概要

1. **enable**
2. **show ssh**
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<pre>enable</pre> <p>例 :</p> <pre>Device> enable</pre>	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	<pre>show ssh</pre> <p>例 :</p> <pre>Device# show ssh</pre>	SSH サーバ接続のステータスを表示します。
ステップ 3	<pre>exit</pre> <p>例 :</p> <pre>Device# exit</pre>	特権 EXEC モードを終了し、ユーザ EXEC モードに戻ります。

次の **show ssh** コマンドの出力例には、バージョン 1 およびバージョン 2 接続の複数の SSH バージョン 1 およびバージョン 2 接続のステータスが表示されています。

```
-----
Device# show ssh
```

```

Connection      Version Encryption      State      Username
0                1.5      3DES              Session started      lab
Connection Version Mode Encryption      Hmac      State
Username
1                2.0      IN      aes128-cbc      hmac-md5      Session started      lab
1                2.0      OUT      aes128-cbc      hmac-md5      Session started      lab
-----

```

次の **show ssh** コマンドの出力例には、バージョン2接続（バージョン1接続なし）の複数のSSHバージョン2およびバージョン1接続のステータスが表示されています。

```

-----
Device# show ssh

Connection Version Mode Encryption      Hmac      State
Username
1                2.0      IN      aes128-cbc      hmac-md5      Session started      lab
1                2.0      OUT      aes128-cbc      hmac-md5      Session started      lab
%No SSHv1 server connections running.
-----

```

次の **show ssh** コマンドの出力例には、バージョン2接続（バージョン1接続なし）の複数のSSHバージョン1およびバージョン2接続のステータスが表示されています。

```

-----
Device# show ssh

Connection      Version Encryption      State      Username
0                1.5      3DES              Session started      lab
%No SSHv2 server connections running.
-----

```

セキュアシェルバージョン2のステータスの確認

手順の概要

1. **enable**
2. **show ip ssh**
3. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	show ip ssh 例 : Device# show ip ssh	SSH のバージョンおよび設定データを表示します。
ステップ 3	exit 例 : Device# exit	特権 EXEC モードを終了し、ユーザ EXEC モードに戻ります。

例

次の **show ip ssh** コマンドの出力例には、有効な SSH のバージョン、認証タイムアウト値、およびバージョン 1 およびバージョン 2 接続の認証の再試行回数が表示されています。

```
-----
Device# show ip ssh

SSH Enabled - version 1.99
Authentication timeout: 120 secs; Authentication retries: 3
-----
```

次の **show ip ssh** コマンドの出力例には、有効な SSH のバージョン、認証タイムアウト値、およびバージョン 2 接続（バージョン 1 接続なし）の認証の再試行回数が表示されています。

```
-----
Device# show ip ssh

SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
-----
```

次の **show ip ssh** コマンドの出力例には、有効な SSH のバージョン、認証タイムアウト値、およびバージョン 1 接続（バージョン 2 接続なし）の認証の再試行回数が表示されています。

```
-----
Device# show ip ssh

3d06h: %SYS-5-CONFIG_I: Configured from console by console
SSH Enabled - version 1.5
Authentication timeout: 120 secs; Authentication retries: 3
-----
```

セキュア シェルバージョン2 のモニタリングと維持

手順の概要

1. **enable**
2. **debug ip ssh**
3. **debug snmp packet**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	debug ip ssh 例： Device# debug ip ssh	SSH のデバッグを有効にします。
ステップ 3	debug snmp packet 例： Device# debug snmp packet	デバイスによって送受信されたすべての SNMP パケットのデバッグを有効にします。

例

次の **debug ip ssh** コマンドの出力例は、接続が SSH バージョン 2 接続であることを示します。

```
Device# debug ip ssh

00:33:55: SSH1: starting SSH control process
00:33:55: SSH1: sent protocol version id SSH-1.99-Cisco-1.25
00:33:55: SSH1: protocol version id is - SSH-2.0-OpenSSH_2.5.2p2
00:33:55: SSH2 1: send: len 280 (includes padlen 4)
00:33:55: SSH2 1: SSH2_MSG_KEXINIT sent
00:33:55: SSH2 1: ssh_receive: 536 bytes received
00:33:55: SSH2 1: input: packet len 632
00:33:55: SSH2 1: partial packet 8, need 624, maclen 0
00:33:55: SSH2 1: ssh_receive: 96 bytes received
00:33:55: SSH2 1: partial packet 8, need 624, maclen 0
00:33:55: SSH2 1: input: padlen 11
00:33:55: SSH2 1: received packet type 20
00:33:55: SSH2 1: SSH2_MSG_KEXINIT received
00:33:55: SSH2: kex: client->server aes128-cbc hmac-md5 none
00:33:55: SSH2: kex: server->client aes128-cbc hmac-md5 none
00:33:55: SSH2 1: expecting SSH2_MSG_KEXDH_INIT
00:33:55: SSH2 1: ssh_receive: 144 bytes received
00:33:55: SSH2 1: input: packet len 144
00:33:55: SSH2 1: partial packet 8, need 136, maclen 0
00:33:55: SSH2 1: input: padlen 5
00:33:55: SSH2 1: received packet type 30
```

```
00:33:55: SSH2 1: SSH2_MSG_KEXDH_INIT received
00:33:55: SSH2 1: signature length 111
00:33:55: SSH2 1: send: len 384 (includes padlen 7)
00:33:55: SSH2: kex_derive_keys complete
00:33:55: SSH2 1: send: len 16 (includes padlen 10)
00:33:55: SSH2 1: newkeys: mode 1
00:33:55: SSH2 1: SSH2_MSG_NEWKEYS sent
00:33:55: SSH2 1: waiting for SSH2_MSG_NEWKEYS
00:33:55: SSH2 1: ssh_receive: 16 bytes received
00:33:55: SSH2 1: input: packet len 16
00:33:55: SSH2 1: partial packet 8, need 8, maclen 0
00:33:55: SSH2 1: input: padlen 10
00:33:55: SSH2 1: newkeys: mode 0
00:33:55: SSH2 1: received packet type 2100:33:55: SSH2 1: SSH2_MSG_NEWKEYS received
00:33:56: SSH2 1: ssh_receive: 48 bytes received
00:33:56: SSH2 1: input: packet len 32
00:33:56: SSH2 1: partial packet 16, need 16, maclen 16
00:33:56: SSH2 1: MAC #3 ok
00:33:56: SSH2 1: input: padlen 10
00:33:56: SSH2 1: received packet type 5
00:33:56: SSH2 1: send: len 32 (includes padlen 10)
00:33:56: SSH2 1: done calc MAC out #3
00:33:56: SSH2 1: ssh_receive: 64 bytes received
00:33:56: SSH2 1: input: packet len 48
00:33:56: SSH2 1: partial packet 16, need 32, maclen 16
00:33:56: SSH2 1: MAC #4 ok
00:33:56: SSH2 1: input: padlen 9
00:33:56: SSH2 1: received packet type 50
00:33:56: SSH2 1: send: len 32 (includes padlen 13)
00:33:56: SSH2 1: done calc MAC out #4
00:34:04: SSH2 1: ssh_receive: 160 bytes received
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #5 ok
00:34:04: SSH2 1: input: padlen 13
00:34:04: SSH2 1: received packet type 50
00:34:04: SSH2 1: send: len 16 (includes padlen 10)
00:34:04: SSH2 1: done calc MAC out #5
00:34:04: SSH2 1: authentication successful for lab
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #6 ok
00:34:04: SSH2 1: input: padlen 6
00:34:04: SSH2 1: received packet type 2
00:34:04: SSH2 1: ssh_receive: 64 bytes received
00:34:04: SSH2 1: input: packet len 48
00:34:04: SSH2 1: partial packet 16, need 32, maclen 16
00:34:04: SSH2 1: MAC #7 ok
00:34:04: SSH2 1: input: padlen 19
00:34:04: SSH2 1: received packet type 90
00:34:04: SSH2 1: channel open request
00:34:04: SSH2 1: send: len 32 (includes padlen 10)
00:34:04: SSH2 1: done calc MAC out #6
00:34:04: SSH2 1: ssh_receive: 192 bytes received
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #8 ok
00:34:04: SSH2 1: input: padlen 13
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: pty-req request
00:34:04: SSH2 1: setting TTY - requested: height 24, width 80; set: height 24,
width 80
00:34:04: SSH2 1: input: packet len 96
00:34:04: SSH2 1: partial packet 16, need 80, maclen 16
```

```
00:34:04: SSH2 1: MAC #9 ok
00:34:04: SSH2 1: input: padlen 11
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: x11-req request
00:34:04: SSH2 1: ssh_receive: 48 bytes received
00:34:04: SSH2 1: input: packet len 32
00:34:04: SSH2 1: partial packet 16, need 16, maclen 16
00:34:04: SSH2 1: MAC #10 ok
00:34:04: SSH2 1: input: padlen 12
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: shell request
00:34:04: SSH2 1: shell message received
00:34:04: SSH2 1: starting shell for vty
00:34:04: SSH2 1: send: len 48 (includes padlen 18)
00:34:04: SSH2 1: done calc MAC out #7
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #11 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #8
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #12 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #9
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #13 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #10
00:34:08: SSH2 1: ssh_receive: 48 bytes received
00:34:08: SSH2 1: input: packet len 32
00:34:08: SSH2 1: partial packet 16, need 16, maclen 16
00:34:08: SSH2 1: MAC #14 ok
00:34:08: SSH2 1: input: padlen 17
00:34:08: SSH2 1: received packet type 94
00:34:08: SSH2 1: send: len 32 (includes padlen 17)
00:34:08: SSH2 1: done calc MAC out #11
00:34:08: SSH2 1: ssh_receive: 48 bytes received
00:34:08: SSH2 1: input: packet len 32
00:34:08: SSH2 1: partial packet 16, need 16, maclen 16
00:34:08: SSH2 1: MAC #15 ok
00:34:08: SSH2 1: input: padlen 17
00:34:08: SSH2 1: received packet type 94
00:34:08: SSH2 1: send: len 32 (includes padlen 16)
00:34:08: SSH2 1: done calc MAC out #12
00:34:08: SSH2 1: send: len 48 (includes padlen 18)
00:34:08: SSH2 1: done calc MAC out #13
00:34:08: SSH2 1: send: len 16 (includes padlen 6)
00:34:08: SSH2 1: done calc MAC out #14
00:34:08: SSH2 1: send: len 16 (includes padlen 6)
00:34:08: SSH2 1: done calc MAC out #15
00:34:08: SSH1: Session terminated normally
```

セキュア シェルバージョン2 サポートの設定例

例：セキュア シェルバージョン2 の設定

```
Device> enable
Device# configure terminal
Device(config)# ip ssh version 2
Device(config)# end
```

例：セキュア シェルバージョン1 および2 の設定

```
Device> enable
Device# configure terminal
Device(config)# no ip ssh version
Device(config)# end
```

例：リモート デバイスでの暗号化セッションの開始

```
Device> enable
Device# ssh -v 2 -c aes256-cbc -m hmac-sha1-160 -l shaship 10.76.82.24
Device# exit
```

例：SNMP トラップの設定

次の例では、SNMPトラップの設定方法を示します。トラップ通知は、SSHセッションが終了すると自動的に生成されます。この例の 10.1.1.1 は SSH クライアントの IP アドレスです。

```
Device> enable
Device# configure terminal
Device(config)# snmp-server trap link switchover
Device(config)# snmp-server host 10.1.1.1 public tty
Device(config)# end
```

例：SSH キーボード インタラクティブ認証

例：クライアント側のデバッグの有効化

次の例では、クライアント側のデバッグがオンになっており、プロンプトの最大数が 6（SSH キーボードインタラクティブ認証方式のために 3 つ、パスワード認証方式のために 3 つ）になっています。

```
Password:
Password:
Password:
Password:
Password:
```

例：ブランクパスワードの変更による ChPass の有効化

```

Password: cisco123
Last login: Tue Dec 6 13:15:21 2005 from 10.76.248.213
user1@courier:~> exit
logout
[Connection to 10.76.248.200 closed by foreign host]
Device1# debug ip ssh client

SSH Client debugging is on

Device1# ssh -l lab 10.1.1.3

Password:
*Nov 17 12:50:53.199: SSH0: sent protocol version id SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: protocol version id is - SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: sent protocol version id SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: protocol version exchange successful
*Nov 17 12:50:53.203: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.335: SSH CLIENT0: key exchange successful and encryption on
*Nov 17 12:50:53.335: SSH2 CLIENT 0: using method keyboard-interactive
Password:
Password:
Password:
*Nov 17 12:51:01.887: SSH2 CLIENT 0: using method password authentication
Password:
Password: lab
Device2>

*Nov 17 12:51:11.407: SSH2 CLIENT 0: SSH2_MSG_USERAUTH_SUCCESS message received
*Nov 17 12:51:11.407: SSH CLIENT0: user authenticated
*Nov 17 12:51:11.407: SSH2 CLIENT 0: pty-req request sent
*Nov 17 12:51:11.411: SSH2 CLIENT 0: shell request sent
*Nov 17 12:51:11.411: SSH CLIENT0: session open

```

例：ブランクパスワードの変更による ChPass の有効化

次の例では、ChPass 機能が有効になっており、SSH キーボードインタラクティブ認証方式を使用してブランクパスワードが変更されています。TACACS+ アクセスコントロールサーバ (ACS) は、バックエンド AAA サーバとして使用されています。

```

Device> enable
Device1# ssh -l cisco 10.1.1.3

Password:
Old Password: cisco
New Password: cisco123
Re-enter New password: cisco123

Device2> exit

[Connection to 10.1.1.3 closed by foreign host]

```

例：ChPass の有効化および初回ログインでのパスワード変更

次の例では、ChPass 機能が有効になっており、TACACS+ ACS はバックエンドサーバとして使用されています。パスワードは、SSH キーボードインタラクティブ認証方式を使用して最初のログインで変更されています。

```

Device1> enable
Device1# ssh -l cisco 10.1.1.3

Password: cisco

```

```
Your password has expired.  
Enter a new one now.  
New Password: cisco123  
Re-enter New password: cisco123  
  
Device2> exit  
  
[Connection to 10.1.1.3 closed by foreign host]  
  
Device1# ssh -l cisco 10.1.1.3  
  
Password:cisco1  
Your password has expired.  
Enter a new one now.  
New Password: cisco  
Re-enter New password: cisco12  
The New and Re-entered passwords have to be the same.  
Try again.  
New Password: cisco  
Re-enter New password: cisco  
  
Device2>
```

例 : ChPass の有効化および3回ログインした後のパスワードの失効

次の例では、ChPass 機能が有効になっており、TACACS+ ACS はバックエンド AAA サーバとして使用されています。パスワードは、SSH キーボードインタラクティブ認証方式を使用して3回ログインした後に期限切れになります。

```
Device# ssh -l cisco. 10.1.1.3  
  
Password: cisco  
  
Device2> exit  
  
[Connection to 10.1.1.3 closed by foreign host]  
  
Device1# ssh -l cisco 10.1.1.3  
  
Password: cisco  
  
Device2> exit  
  
Device1# ssh -l cisco 10.1.1.3  
  
Password: cisco  
  
Device2> exit  
  
[Connection to 10.1.1.3 closed by foreign host]  
  
Device1# ssh -l cisco 10.1.1.3  
  
Password: cisco  
Your password has expired.  
Enter a new one now.  
New Password: cisco123  
Re-enter New password: cisco123  
  
Device2>
```

例 : SNMP のデバッグ

次に、**debug snmp packet** コマンドの出力例を示します。出力には、SSH セッションの SNMP トラップ情報が含まれます。

```
Device1# debug snmp packet

SNMP packet debugging is on
Device1# ssh -l lab 10.0.0.2
Password:

Device2# exit

[Connection to 10.0.0.2 closed by foreign host]
Device1#
*Jul 18 10:18:42.619: SNMP: Queuing packet to 10.0.0.2
*Jul 18 10:18:42.619: SNMP: V1 Trap, ent cisco, addr 10.0.0.1, gentrap 6, spectrap 1
local.9.3.1.1.2.1 = 6
tcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 4
ltcpConnEntry.5.10.0.0.1.22.10.0.0.2.55246 = 1015
ltcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 1056
ltcpConnEntry.2.10.0.0.1.22.10.0.0.2.55246 = 1392
local.9.2.1.18.2 = lab
*Jul 18 10:18:42.879: SNMP: Packet sent via UDP to 10.0.0.2

Device1#
```

例 : SSH のデバッグの強化

次に、**debug ip ssh detail** コマンドの出力例を示します。出力には、SSH プロトコルとチャネル要求に関するデバッグ情報が含まれます。

```
Device# debug ip ssh detail

00:04:22: SSH0: starting SSH control process
00:04:22: SSH0: sent protocol version id SSH-1.99-Cisco-1.25
00:04:22: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
00:04:22: SSH2 0: SSH2_MSG_KEXINIT sent
00:04:22: SSH2 0: SSH2_MSG_KEXINIT received
00:04:22: SSH2:kex: client->server enc:aes128-cbc mac:hmac-shal
00:04:22: SSH2:kex: server->client enc:aes128-cbc mac:hmac-shal
00:04:22: SSH2 0: expecting SSH2_MSG_KEXDH_INIT
00:04:22: SSH2 0: SSH2_MSG_KEXDH_INIT received
00:04:22: SSH2: kex_derive_keys complete
00:04:22: SSH2 0: SSH2_MSG_NEWKEYS sent
00:04:22: SSH2 0: waiting for SSH2_MSG_NEWKEYS
00:04:22: SSH2 0: SSH2_MSG_NEWKEYS received
00:04:24: SSH2 0: authentication successful for lab
00:04:24: SSH2 0: channel open request
00:04:24: SSH2 0: pty-req request
00:04:24: SSH2 0: setting TTY - requested: height 24, width 80; set: height 24, width
80
00:04:24: SSH2 0: shell request
00:04:24: SSH2 0: shell message received
00:04:24: SSH2 0: starting shell for vty
00:04:38: SSH0: Session terminated normally
```

次に、**debug ip ssh packet** コマンドの出力例を示します。出力には、SSH パケットに関するデバッグ情報が含まれます。


```

Device# debug ip ssh packet

00:05:43: SSH2 0: send:packet of length 280 (length also includes padlen of 4)
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: input: total packet length of 280 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 24 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 4 bytes
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: input: total packet length of 144 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 16 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 6 bytes
00:05:43: SSH2 0: signature length 143
00:05:43: SSH2 0: send:packet of length 448 (length also includes padlen of 7)
00:05:43: SSH2 0: send:packet of length 16 (length also includes padlen of 10)
00:05:43: SSH2 0: newkeys: mode 1
00:05:43: SSH2 0: ssh_receive: 16 bytes received
00:05:43: SSH2 0: input: total packet length of 16 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 8 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 10 bytes
00:05:43: SSH2 0: newkeys: mode 0
00:05:43: SSH2 0: ssh_receive: 52 bytes received
00:05:43: SSH2 0: input: total packet length of 32 bytes
00:05:43: SSH2 0: partial packet length(block size)16 bytes,needed 16 bytes, maclen 20
00:05:43: SSH2 0: MAC compared for #3 :ok

```

セキュア シェルバージョン2 サポートの追加情報

関連資料

関連項目	マニュアルタイトル
SSH バージョン 1	『セキュリティ コンフィギュレーションガイド』の「セキュアシェルの設定」

標準

標準	タイトル
IETF Secure Shell Version 2 Draft 規格	Internet Engineering Task Force の Web サイト

シスコのテクニカル サポート

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

セキュアシェルバージョン2 サポートの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	セキュア シェル バージョン 2 サポート	セキュア シェル バージョン 2 サポート機能を使用して、セキュア シェル (SSH) バージョン 2 を設定できます (SSH バージョン 1 のサポートは、以前の Cisco IOS ソフトウェアリリースで実装されていました)。SSH は、信頼性の高いトランスポート層の上部で実行され、強力な認証機能と暗号化機能を提供します。SSH バージョン 2 は、AES カウンタベース暗号化モードもサポートします。

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



第 13 章

SSH File Transfer Protocol の設定

セキュアシェル (SSH) には、SSHv2 で導入された新たな標準ファイル転送プロトコルである SSH File Transfer Protocol (SFTP) のサポートが含まれています。この機能は、デバイス設定またはデバイスイメージファイルをコピーするための安全で認証された方式を提供します。

- [SSH File Transfer Protocol の前提条件](#) (309 ページ)
- [SSH File Transfer Protocol の制約事項](#) (309 ページ)
- [IPv6 を介した SSH サポートに関する情報](#) (310 ページ)
- [SSH File Transfer Protocol の設定方法](#) (310 ページ)
- [IPv6 を介した SSH サポートの設定例](#) (312 ページ)
- [SSH File Transfer Protocol に関する追加情報](#) (312 ページ)
- [SSH File Transfer Protocol の機能履歴](#) (312 ページ)

SSH File Transfer Protocol の前提条件

- SSH を有効にする必要があります。
- `ip ssh source-interface interface-type interface-number` コマンドを設定する必要があります。

SSH File Transfer Protocol の制約事項

- SFTP サーバはサポートされていません。
- SFTP 起動はサポートされていません。
- `sftp` コマンドでの `install add` オプションはサポートされていません。

IPv6 を介した SSH サポートに関する情報

SSH File Transfer Protocol の概要

SFTP クライアント機能は SSH コンポーネントの一部として提供され、対応するデバイスで常に有効になっています。したがって、適切な権限を持つ SFTP サーバのユーザは、デバイスとの間でファイルをコピーできます。

SFTP クライアントは VRF 対応です。接続の試行時に特定の送信元インターフェイスに関連付けられた仮想ルーティングおよび転送（VRF）を使用するようにセキュア FTP クライアントを設定できます。

SSH File Transfer Protocol の設定方法

ここでは、SFTP の設定を構成するさまざまな作業について説明します。

SFTP の設定

次の操作を行ってください。

始める前に

SFTP クライアント側機能用にシスコ デバイスを設定するには、最初に **ip ssh source-interface interface-type interface-number** コマンドを設定する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh source-interface interface-type interface-number**
4. **exit**
5. **show running-config**
6. **debug ip sftp**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip ssh source-interface interface-type interface-number 例 : Device(config)# ip ssh source-interface GigabitEthernet 1/0/1	SSH セッションの送信元 IP を定義します。
ステップ 4	exit 例 : Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 5	show running-config 例 : Device# show running-config	(任意) SFTP クライアント側機能を表示します。
ステップ 6	debug ip sftp 例 : Device# debug ip sftp	(任意) SFTP デバッグを有効にします。

SFTP コピー操作の実行

ドメインネームシステム (DNS) が設定されている場合、SFTP コピーは対応するサーバの IP またはホスト名を取得します。SFTP コピー操作を実行するには、特権 EXEC モードで次のコマンドを使用します。

コマンド	目的
Device# copy ios-file-system:file sftp://user:pwd@server-ip//filepath または Device# copy ios-file-system: sftp:	ローカル Cisco IOS ファイルシステムからサーバにファイルをコピーします。 サーバのユーザ名、パスワード、IP アドレス、およびファイルパスを指定します。
Device# copy sftp://user:pwd@server-ip//filepath ios-file-system:file または Device# copy sftp: ios-file-system:	サーバからローカル Cisco IOS ファイルシステムにファイルをコピーします。 サーバのユーザ名、パスワード、IP アドレス、およびファイルパスを指定します。

IPv6 を介した SSH サポートの設定例

例：SSH File Transfer Protocol の設定

次に、SFTP のクライアント側機能を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip ssh source-interface gigabitethernet 1/0/1
Device(config)# exit
```

SSH File Transfer Protocol に関する追加情報

関連資料

関連項目	マニュアルタイトル
セキュアシェルバージョン 1 と 2 のサポート	『セキュリティコンフィギュレーションガイド』

シスコのテクニカル サポート

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

SSH File Transfer Protocol の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Gibraltar 16.10.1	SSH ファイル転送プロトコル	SSH には、SSHv2 で導入された新たな標準ファイル転送プロトコルである SFTP のサポートが含まれています。

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



第 14 章

SSH 認証の X.509v3 証明書

- [SSH 認証の X.509v3 証明書 \(315 ページ\)](#)

SSH 認証の X.509v3 証明書

SSH 認証の X.509v3 証明書機能は、サーバー内で X.509v3 デジタル証明書を使用し、セキュアシェル (SSH) サーバー側でユーザー認証を使用します。

このモジュールでは、デジタル証明書用のサーバおよびユーザ証明書プロファイルを設定する方法について説明します。

SSH 認証の X.509v3 証明書の前提条件

- SSH 認証の X.509v3 証明書機能では、**ip ssh server authenticate user** コマンドの代わりに **ip ssh server algorithm authentication** コマンドが導入されます。 **ip ssh server authenticate user** コマンドを使用すると、次の警告メッセージが表示されます。

```
Warning: SSH command accepted but this CLI will be deprecated soon.  
Please move to new CLI "ip ssh server algorithm authentication".  
Please configure "default ip ssh server authenticate user" to make the CLI ineffective.
```

default ip ssh server authenticate user コマンドを使用して、**ip ssh server authenticate user** コマンドを無効にします。その後、IOS セキュアシェル (SSH) サーバーは **ip ssh server algorithm authentication** コマンドを使用して起動します。

SSH 認証の X.509v3 証明書の制約事項

- SSH 認証の X.509v3 証明書機能の実装は、Cisco IOS XE セキュアシェル (SSH) サーバー側にのみ適用できます。
- SSH サーバーは、サーバーおよびユーザー認証について、x509v3-ssh-rsa アルゴリズムベースの証明書のみをサポートします。

SSH 認証用の X.509v3 証明書に関する情報

次に、デジタル証明書、およびサーバーとユーザーの認証について説明します。

デジタル証明書

認証の有効性は、公開署名キーとその署名者のアイデンティティとの関連の強さに依存します。X.509v3 形式 (RFC5280) のデジタル証明書は、アイデンティティの管理を実行するために使用されます。信頼できるルート証明機関とその中間証明機関による署名の連鎖によって、指定の公開署名キーと指定のデジタルアイデンティティがバインドされます。

公開キーインフラストラクチャ (PKI) のトラストポイントは、デジタル証明書の管理に役立ちます。証明書とトラストポイントを関連付けることによって、証明書を追跡できます。トラストポイントには、認証局 (CA)、さまざまなアイデンティティパラメータ、およびデジタル証明書に関する情報が含まれています。複数のトラストポイントを作成して、異なる証明書に関連付けることができます。

X.509v3 を使用したサーバーおよびユーザー認証

サーバー認証の場合、Cisco IOS XE セキュアシェル (SSH) サーバーが確認のためにそれ自体の証明書を SSH クライアントに送信します。このサーバー証明書は、サーバー証明書プロファイル (ssh-server-cert-profile-server コンフィギュレーションモード) で設定されたトラストポイントに関連付けられます。

ユーザー認証の場合、SSH クライアントが確認のためにユーザーの証明書を SSH サーバーに送信します。SSH サーバーは、サーバー証明書プロファイル (ssh-server-cert-profile-user コンフィギュレーションモード) で設定された公開キーインフラストラクチャ (PKI) トラストポイントを使用して、受信したユーザー証明書を確認します。

デフォルトでは、証明書ベースの認証が SSH サーバー端末でサーバーおよびユーザーに対して有効になります。

SSH 認証用の X.509v3 証明書の設定方法

ここでは、SSH 認証用の X.509v3 証明書の設定方法について説明します。

サーバー認証にデジタル証明書を使用するための SSH サーバーの設定

サーバー認証にデジタル証明書を使用するように SSH サーバーを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip ssh server algorithm hostkey {x509v3-ssh-rsa [ssh-rsa] ssh-rsa [x509v3-ssh-rsa]} 例： Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa	<p>ホスト キー アルゴリズムの順序を定義します。セキュアシェル (SSH) クライアントとネゴシエートされるのは、設定済みのアルゴリズムのみです。</p> <p>(注) IOS SSH サーバーには、1 つ以上の設定済みホスト キー アルゴリズムが必要です。</p> <ul style="list-style-type: none"> • ssh-rsa : 公開キーベース認証 • x509v3-ssh-rsa : 証明書ベース認証
ステップ 4	ip ssh server certificate profile 例： Device(config)# ip ssh server certificate profile	サーバー証明書プロファイルおよびユーザー証明書プロファイルを設定し、SSH 証明書プロファイル 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	server 例： Device(ssh-server-cert-profile)# server	サーバー証明書プロファイルを設定し、SSH サーバー証明書プロファイルのユーザー コンフィギュレーション モードを開始します。
ステップ 6	trustpoint sign PKI-trustpoint-name 例： Device(ssh-server-cert-profile-server)# trustpoint sign trust1	公開キー インフラストラクチャ (PKI) トラストポイントをサーバ証明書プロファイルにアタッチします。SSH サーバは、この PKI トラストポイントに関連付けられた証明書をサーバ認証に使用します。
ステップ 7	ocsp-response include 例： Device(ssh-server-cert-profile-server)# ocsp-response include	<p>(任意) Online Certificate Status Protocol (OCSP) の応答または OCSP ステージングをサーバ証明書と一緒に送信します。</p> <p>(注) デフォルトではこのコマンドの no 形式が設定されており、OCSP 応答はサーバ証明書と一緒に送信されません。</p>
ステップ 8	end 例： Device(ssh-server-cert-profile-server)# end	SSH サーバー証明書プロファイルのサーバー コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ユーザー認証用のデジタル証明書を確認するための SSH サーバーの設定

ユーザー認証にデジタル証明書を使用するように SSH サーバーを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip ssh server algorithm authentication {publickey keyboard password} 例： Device(config)# ip ssh server algorithm authentication publickey	ユーザ認証アルゴリズムの順序を定義します。セキュアシェル (SSH) クライアントとネゴシエートされるのは、設定済みのアルゴリズムのみです。 (注) <ul style="list-style-type: none"> • SSH サーバーには、1 つ以上の設定済みユーザー認証アルゴリズムが必要です。 • ユーザー認証に証明書方式を使用するには、publickey キーワードを設定する必要があります。 • ip ssh server algorithm authentication コマンドは ip ssh server authenticate user コマンドの代わりに使用します。
ステップ 4	ip ssh server algorithm publickey {x509v3-ssh-rsa [ssh-rsa] ssh-rsa [x509v3-ssh-rsa]} 例： Device(config)# ip ssh server algorithm publickey x509v3-ssh-rsa	公開キーアルゴリズムの順序を定義します。SSH クライアントによってユーザー認証に許可されるのは、設定済みのアルゴリズムのみです。 (注) <ul style="list-style-type: none"> • SSH クライアントには、1 つ以上の設定済み公開キーアルゴリズムが必要です。 • ssh-rsa : 公開キーベース認証 • x509v3-ssh-rsa : 証明書ベース認証
ステップ 5	ip ssh server certificate profile 例： Device(config)# ip ssh server certificate profile	サーバー証明書プロファイルおよびユーザー証明書プロファイルを設定し、SSH 証明書プロファイル コンフィギュレーション モードを開始します。
ステップ 6	user 例： Device(ssh-server-cert-profile)# user	ユーザ証明書プロファイルを設定し、SSH サーバ証明書プロファイルのユーザ 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 7	trustpoint verify PKI-trustpoint-name 例： Device (ssh-server-cert-profile-user) # trustpoint verify trust2	受信したユーザ証明書の確認に使用される公開キーインフラストラクチャ (PKI) トラストポイントを設定します。 (注) 同じコマンドを複数回実行することで、複数のトラストポイントを設定します。最大 10 のトラストポイントを設定できます。
ステップ 8	ocsp-response required 例： Device (ssh-server-cert-profile-user) # ocsp-response required	(任意) 受信したユーザ証明書による Online Certificate Status Protocol (OCSP) の応答の有無を要求します。 (注) デフォルトではこのコマンドの no 形式が設定されており、ユーザー証明書は OCSP 応答なしで受け入れられます。
ステップ 9	end 例： Device (ssh-server-cert-profile-user) # end	SSH サーバー証明書プロファイルのユーザー コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

デジタル証明書を使用したサーバーおよびユーザー認証の設定の確認

デジタル証明書を使用したサーバーおよびユーザー認証の設定を確認するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	show ip ssh 例： Device# show ip ssh SSH Enabled - version 1.99 Authentication methods:publickey,keyboard-interactive,password Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa Authentication timeout: 120 secs; Authentication retries: 3	現在設定されている認証方式を表示します。証明書ベース認証の使用を確認するには、x509v3-ssh-rsa アルゴリズムが設定済みのホストキーアルゴリズム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
Minimum expected Diffie Hellman key size : 1024 bits	

SSH 認証用の X.509v3 証明書の設定例

ここでは、デジタル証明書を使用したユーザーおよびサーバー認証の例を示します。

例：サーバー認証にデジタル証明書を使用するための SSH サーバーの設定

この例では、サーバー認証用のデジタル証明書を使用するための SSH サーバーの設定方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa
Device(config)# ip ssh server certificate profile
Device(ssh-server-cert-profile)# server
Device(ssh-server-cert-profile-server)# trustpoint sign trust1
Device(ssh-server-cert-profile-server)# end
```

例：ユーザー認証用のデジタル証明書を確認するための SSH サーバーの設定

この例では、ユーザー認証用のユーザーのデジタル証明書を確認するための SSH サーバーの設定方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm authentication publickey
Device(config)# ip ssh server algorithm publickey x509v3-ssh-rsa
Device(config)# ip ssh server certificate profile
Device(ssh-server-cert-profile)# user
Device(ssh-server-cert-profile-user)# trustpoint verify trust2
Device(ssh-server-cert-profile-user)# end
```

SSH 認証用の X.509v3 証明書の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	SSH 認証の X.509v3 証明書	SSH 認証の X.509v3 証明書機能は、サーバ内で X.509v3 デジタル証明書を使用し、SSH サーバ側でユーザ認証を使用します。

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



第 15 章

コモンクライテリア認定用の SSH アルゴリズム

- [コモンクライテリア認定用の SSH アルゴリズムに関する情報 \(323 ページ\)](#)
- [コモンクライテリア認定用の SSH アルゴリズムの設定方法 \(325 ページ\)](#)
- [コモンクライテリア認定用の SSH アルゴリズムの設定例 \(329 ページ\)](#)
- [コモンクライテリア認定用の SSH アルゴリズムの確認 \(330 ページ\)](#)
- [コモンクライテリア認定用のセキュアシェルアルゴリズムの機能情報 \(331 ページ\)](#)

コモンクライテリア認定用の SSH アルゴリズムに関する情報

ここでは、コモンクライテリア認定のセキュアシェル (SSH) アルゴリズム、Cisco IOS SSH サーバーアルゴリズム、および Cisco IOS SSH クライアントアルゴリズムについて説明します。

コモンクライテリア認定用の SSH アルゴリズム

セキュアシェル (SSH) 設定によって、Cisco IOS SSH サーバーおよびクライアントは、許可リストから設定されたアルゴリズムのネゴシエーションのみを許可することができます。リモートパーティが許可リストに含まれていないアルゴリズムのみを使用してネゴシエートしようとする、要求は拒否され、セッションは確立されません。

Cisco IOS SSH サーバー アルゴリズム

Cisco IOS セキュア シェル (SSH) サーバーは、次の順序で暗号化アルゴリズム (Advanced Encryption Standard カウンタ モード [AES-CTR]、AES 暗号ブロック連鎖 [AES-CBC]、Triple Data Encryption Standard [3DES]) をサポートします。

サポートされるデフォルトの暗号化の順序 :

1. aes128-ctr

2. aes192-ctr
3. aes256-ctr

サポートされるデフォルト以外の暗号化の順序：

1. aes128-cbc
2. aes192-cbc
3. aes256-cbc
4. 3des

Cisco IOS SSH クライアントは、次の順序でメッセージ認証コード (MAC) アルゴリズムをサポートします。

サポートされるデフォルトの HMAC の順序：

1. hmac-sha2-256
2. hmac-sha2-512
3. hmac-sha1
4. hmac-sha1-96

Cisco IOS SSH クライアントがサポートするホストキーアルゴリズムは 1 つのみで、CLI 設定は必要ありません。

サポートされるデフォルトのホストキーの順序：

1. x509v3-ssh-rsa
2. ssh-rsa

Cisco IOS SSH クライアント アルゴリズム

Cisco IOS セキュア シェル (SSH) クライアントは、次の順序で暗号化アルゴリズム (Advanced Encryption Standard カウンタ モード [AES-CTR]、AES 暗号ブロック連鎖 [AES-CBC]、Triple Data Encryption Standard [3DES]) をサポートします。

サポートされるデフォルトの暗号化の順序：

1. aes128-ctr
2. aes192-ctr
3. aes256-ctr

サポートされるデフォルト以外の暗号化の順序：

1. aes128-cbc
2. aes192-cbc
3. aes256-cbc

4. 3des

Cisco IOS SSH クライアントは、次の順序でメッセージ認証コード (MAC) アルゴリズムをサポートします。

サポートされるデフォルトの HMAC の順序 :

1. hmac-sha2-256
2. hmac-sha2-512
3. hmac-sha1
4. hmac-sha1-96

Cisco IOS SSH クライアントがサポートするホストキーアルゴリズムは1つのみで、CLI 設定は必要ありません。

サポートされるデフォルトのホストキーの順序 :

1. x509v3-ssh-rsa
2. ssh-rsa

コモンクライテリア認定用の SSH アルゴリズムの設定方法

ここでは、設定とトラブルシューティング方法に関する情報を提供します。

- Cisco IOS SSH サーバーおよびクライアントの暗号キーアルゴリズム
- Cisco IOS SSH サーバーおよびクライアントの MAC アルゴリズム
- Cisco IOS SSH サーバーのホストキーアルゴリズム

Cisco IOS SSH サーバーおよびクライアントの暗号キーアルゴリズムの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh {server | client} algorithm encryption {aes128-ctr | aes192-ctr | aes256-ctr | aes128-cbc | aes192-cbc | aes256-cbc | 3des-cbc }**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <p>パスワードを入力します (要求された場合)。</p>
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>
ステップ 3	<p>ip ssh {server client} algorithm encryption {aes128-ctr aes192-ctr aes256-ctr aes128-cbc aes192-cbc aes256-cbc 3des-cbc }</p> <p>例 :</p> <pre>Device(config)# ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr aes128-cbc 3des-cbc aes192-cbc aes256-cbc</pre> <pre>Device(config)# ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr aes128-cbc 3des-cbc aes192-cbc aes256-cbc</pre>	<p>SSH サーバーおよびクライアントでの暗号化アルゴリズムの順序を定義します。この順序は、アルゴリズムのネゴシエーション時に指定されます。</p> <p>(注) Cisco IOS SSH サーバーおよびクライアントには、1つ以上の設定済み暗号化アルゴリズムが必要です。</p> <p>(注) 以前設定したアルゴリズムのリストから1つのアルゴリズムを無効にするには、このコマンドの no 形式を使用します。複数のアルゴリズムを無効にするには、このコマンドの no 形式を異なるアルゴリズム名で複数回使用します。</p> <p>(注) デフォルト設定では、次に示すようにこのコマンドのデフォルト形式を使用します。</p> <pre>Device(config)# ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr aes128-cbc 3des-cbc aes192-cbc aes256-cbc</pre>
ステップ 4	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

トラブルシューティングのヒント

設定で最後の暗号化アルゴリズムを無効にしようとする、次のメッセージが表示されてコマンドが拒否されます。

% SSH command rejected: All encryption algorithms cannot be disabled

Cisco IOS SSH サーバーおよびクライアントの MAC アルゴリズムの設定

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合) 。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip ssh {server client} algorithm mac {hmac-sha2-256 hmac-sha2-512 hmac-sha1 hmac-sha1-96} 例 : Device(config)# ip ssh server algorithm mac hmac-sha2-256 hmac-sha2-512 hmac-sha1 hmac-sha1-96 Device(config)# ip ssh client algorithm mac hmac-sha2-256 hmac-sha2-512 hmac-sha1 hmac-sha1-96	SSH サーバーおよびクライアントでの MAC (メッセージ認証コード) アルゴリズムの順序を定義します。この順序は、アルゴリズムのネゴシエーション時に指定されます。 (注) Cisco IOS SSH サーバーおよびクライアントには、1つ以上の設定済みハッシュメッセージ認証コード (HMAC) アルゴリズムが必要です。 (注) 以前設定したアルゴリズムのリストから1つのアルゴリズムを無効にするには、このコマンドの no 形式を使用します。複数のアルゴリズムを無効にするには、このコマンドの no 形式を異なるアルゴリズム名で複数回使用します。 (注) デフォルト設定では、次に示すようにこのコマンドのデフォルト形式を使用します。 Device(config)# ip ssh server algorithm mac hmac-sha2-256 hmac-sha2-512 hmac-sha1 hmac-sha1-96

	コマンドまたはアクション	目的
ステップ 4	end 例： Device (config) # end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

トラブルシューティングのヒント

設定で最後の MAC アルゴリズムを無効にしようとする、次のメッセージが表示されてコマンドが拒否されます。

```
% SSH command rejected: All mac algorithms cannot be disabled
```

Cisco IOS SSH サーバーのホスト キー アルゴリズムの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh server algorithm hostkey {x509v3-ssh-rsa | ssh-rsa}**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip ssh server algorithm hostkey {x509v3-ssh-rsa ssh-rsa} 例： Device (config) # ip ssh server algorithm hostkey x509v3-ssh-rsa ssh-rsa	ホストキーアルゴリズムの順序を定義します。Cisco IOS セキュア シェル (SSH) クライアントとネゴシエートされるのは、設定済みのアルゴリズムのみです。

	コマンドまたはアクション	目的
		<p>(注) Cisco IOS SSH サーバーには、1つ以上の設定済みホストキーアルゴリズムが必要です。</p> <ul style="list-style-type: none"> • x509v3-ssh-rsa : X.509v3 証明書ベース認証 • ssh-rsa : 公開キーベース認証 <p>(注) 以前設定したアルゴリズムのリストから1つのアルゴリズムを無効にするには、このコマンドの no 形式を使用します。複数のアルゴリズムを無効にするには、このコマンドの no 形式を異なるアルゴリズム名で複数回使用します。</p> <p>(注) デフォルト設定では、次に示すようにこのコマンドのデフォルト形式を使用します。</p> <pre>Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa ssh-rsa</pre>
ステップ 4	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

トラブルシューティングのヒント

設定で最後のホストキーアルゴリズムを無効にしようとするすると、次のメッセージが表示されてコマンドが拒否されます。

```
% SSH command rejected: All hostkey algorithms cannot be disabled
```

コモンクライテリア認定用のSSHアルゴリズムの設定例

ここでは、コモン認定用のSSHアルゴリズムの設定例を示します。

例 : Cisco IOS SSH サーバーの暗号キー アルゴリズムの設定

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
aes128-cbc aes192-cbc aes256-cbc 3des
Device(config)# end
```

例 : Cisco IOS SSH クライアントの暗号キー アルゴリズムの設定

```
Device> enable
Device# configure terminal
Device(config)# ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
aes128-cbc aes192-cbc aes256-cbc 3des

Device(config)# end
```

例 : Cisco IOS SSH サーバーの MAC アルゴリズムの設定

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm mac hmac-sha2-256, hmac-sha2-512, hmac-sha1,
hmac-sha1-96
Device(config)# end
```

例 : Cisco IOS SSH サーバーのホスト キー アルゴリズムの設定

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa ssh-rsa
Device(config)# end
```

コモンクライテリア認定用の SSH アルゴリズムの確認

手順の概要

1. enable
2. show ip ssh

手順の詳細

ステップ 1 enable

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 show ip ssh

設定済みのセキュアシェル（SSH）暗号化、ホストキー、およびメッセージ認証コード（MAC）アルゴリズムを表示します。

例：

次の **show ip ssh** コマンドの出力例は、デフォルトの順序で設定された暗号化アルゴリズムを示しています。

```
Device# show ip ssh
```

```
Encryption Algorithms: aes128-ctr, aes192-ctr, aes256-ctr, aes128-cbc, aes192-cbc, aes256-cbc, 3des
```

次の **show ip ssh** コマンドの出力例は、デフォルトの順序で設定された MAC アルゴリズムを示しています。

```
Device# show ip ssh
```

```
MAC Algorithms: hmac-sha2-256, hmac-sha2-512, hmac-sha1, hmac-sha1-96
```

次の **show ip ssh** コマンドの出力例は、デフォルトの順序で設定されたホストキー アルゴリズムを示しています。

```
Device# show ip ssh
```

```
Hostkey Algorithms: x509v3-ssh-rsa, ssh-rsa
```

コモンクライテリア認定用のセキュアシェルアルゴリズムの機能情報

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	コモンクライテリア認定用のセキュアシェルアルゴリズム	コモンクライテリア認定用の SSH アルゴリズム機能によって、コモンクライテリア認定を取得したアルゴリズムのリストおよび順序が提供されます。このモジュールでは、認定されたアルゴリズムのリストに基づいて SSH 接続を制限できるように、セキュアシェル (SSH) サーバーおよびクライアントの暗号化、メッセージ認証コード (MAC)、およびホストキーアルゴリズムの設定方法について説明します。

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



第 16 章

Secure Socket Layer HTTP の設定

- [Secure Socket Layer HTTP に関する情報](#) (333 ページ)
- [Secure Socket Layer HTTP の設定方法](#) (337 ページ)
- [セキュア HTTP サーバおよびクライアントのステータスのモニタリング](#) (344 ページ)
- [Secure Socket Layer HTTP に関するその他の参考資料](#) (344 ページ)
- [Secure Socket Layer HTTP の機能履歴](#) (345 ページ)

Secure Socket Layer HTTP に関する情報

セキュア HTTP サーバおよびクライアントの概要

セキュア HTTP 接続の場合、HTTP サーバが送受信するデータは暗号化されてインターネットに送信されます。SSL 暗号化を伴う HTTP は、Web ブラウザからスイッチを設定するような機能に、セキュアな接続を提供します。シスコが実装するセキュア HTTP サーバおよび HTTP クライアントでは、アプリケーション層の暗号化に SSL バージョン 3.0 を使用します。HTTP over SSL は、HTTPS と省略されます (セキュアな接続の場合、URL が `http://` の代わりに `https://` で始まります)。

セキュア HTTP サーバ (スイッチ) の主な役割は、指定のポート (デフォルトの HTTPS ポートは 443) で HTTPS 要求を待ち受けて、HTTP 1.1 Web サーバへその要求を渡すことです。HTTP 1.1 サーバはその要求を処理して、セキュア HTTP サーバへ応答 (呼び出す) します。セキュア HTTP サーバは HTTP 1.1 サーバの代わりに、元の要求に応えます。

セキュア HTTP クライアント (Web ブラウザ) の主な役割は、Cisco IOS アプリケーション要求に応答して、そのアプリケーションが要求した HTTPS User Agent サービスを実行し、応答を (そのアプリケーションに) 返すことです。

CA のトラストポイント

認証局 (CA) は、要求を認可して参加するネットワーク デバイスに証明書を発行します。これらのサービスは、参加するデバイスに対する中央集約的なセキュリティ キーおよび証明書の管理を提供します。特定の CA サーバはトラストポイントと呼ばれます。

接続が実行されると、HTTPS サーバは、トラストポイントとなる特定の CA から得た X.509v3 の証明書を発行することで、セキュアな接続をクライアントに提供します。クライアント（通常、Web ブラウザ）は、その証明書の認証に必要な公開キーを保有しています。

セキュア HTTP 接続には、CA のトラストポイントを設定することを強く推奨します。HTTPS サーバを実行しているデバイスに CA のトラストポイントが設定されていないと、サーバは自身を認証して必要な RSA のキーのペアを生成します。自身で認証した（自己署名）証明書は適切なセキュリティではないので、接続するクライアントはその証明書が自己証明書であることを通知し、ユーザに接続の選択（確立または拒否）をさせる必要があります。この選択肢は内部ネットワーク トポロジ（テスト用など）に役立ちます。

CA のトラストポイントを設定していないと、セキュア HTTP 接続を有効にした場合、そのセキュア HTTP サーバ（またはクライアント）に対する一時的または永続的な自己署名証明書が自動的に生成されます。

- スイッチにホスト名とドメイン名が設定されていない場合、生成される自己署名証明書は一時的なものです。スイッチを再起動すると、この一時的な自己署名証明書は失われ、新たに自己署名証明書（一時的に）が割り当てられます。
- スイッチにホスト名とドメイン名が設定されている場合、生成される自己署名証明書は永続的なものです。この証明書は、スイッチを再起動しても、セキュア HTTP サーバを無効にしても有効のままです。そのため、再度セキュア HTTP 接続を有効にしたときに使用できます。



Note 認証局およびトラストポイントは、個々のデバイスで設定する必要があります。他のデバイスからコピーすると、それらはスイッチ上で無効になります。

新しい証明書を登録した場合、新しい設定の変更は、サーバが再起動するまで HTTPS サーバに適用されません。**reload** コマンドを使用して DCNM サーバを再起動できます。サーバを再起動すると、スイッチは新しい証明書の使用を開始します。

自己署名証明書が生成された場合、その情報は **show running-config** 特権 EXEC コマンドで出力できます。自己署名証明書を表示するコマンドの出力（**show running-config** コマンド）を例として一部示します。

```
Device# show running-config
Building configuration...

<output truncated>

crypto pki trustpoint TP-self-signed-3080755072
  enrollment selfsigned
  subject-name cn=IOS-Self-Signed-Certificate-3080755072
  revocation-check none
  rsakeypair TP-self-signed-3080755072
  !
  !
crypto ca certificate chain TP-self-signed-3080755072
  certificate self-signed 01
    3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
```

```
59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D
```

<output truncated>

自己署名証明書は、セキュア HTTP サーバを無効にして、**no crypto pki trustpoint**

TP-self-signed-30890755072 グローバル コンフィギュレーション コマンドを入力することで削除できます。その後、セキュア HTTP サーバを再度有効にすると、自己署名証明書が新たに生成されます。



Note *TP self-signed* の後ろに表示されている値は、デバイスのシリアル番号によって異なります。

オプションのコマンド (**ip http secure-client-auth**) を使用すると、HTTPS サーバがクライアントからの X.509v3 証明書を要求します。クライアントの認証は、サーバ自身の認証よりも高いセキュリティを提供します。

CipherSuite

CipherSuite は暗号化アルゴリズムおよびダイジェストアルゴリズムを指定して、SSL 接続に使用します。HTTPS サーバに接続すると、クライアントの Web ブラウザは、サポート対象の CipherSuite のリストを提供します。その後クライアントとサーバは、両方でサポートされている暗号化アルゴリズムで最適なものをリストから選択してネゴシエートします。たとえば、Netscape Communicator 4.76 は、米国のセキュリティ (RSA 公開キー暗号 MD2、MD5、RC2-CBC、RC4、DES-CBC、および DES-EDE3-CBC) をサポートしています。

最適な暗号化には、128 ビット暗号化をサポートするクライアントブラウザ (Microsoft Internet Explorer バージョン 5.5 以降または Netscape Communicator バージョン 4.76 以降など) が必要です。SSL_RSA_WITH_DES_CBC_SHA CipherSuite は、128 ビット暗号化を提供しないため、他の CipherSuite よりもセキュリティが低くなります。

CipherSuite は、よりセキュリティが高く、複雑になればなるほど、わずかですが処理時間が必要になります。次に、スイッチでサポートされる CipherSuite およびルータの処理負荷 (速さ) による CipherSuite のランク (速い順) を定義します。

1. SSL_RSA_WITH_DES_CBC_SHA : メッセージの暗号化に DES-CBC、およびメッセージダイジェストに SHA を使用した RSA のキー交換 (RSA 公開キー暗号化)
2. SSL_RSA_WITH_NULL_SHA : メッセージの暗号化に NULL、およびメッセージダイジェストに SHA を使用したキー交換 (SSL 3.0 専用)。
3. SSL_RSA_WITH_NULL_MD5 : メッセージの暗号化に NULL、およびメッセージダイジェストに MD5 を使用したキー交換 (SSL 3.0 専用)。
4. SSL_RSA_WITH_RC4_128_MD5 : RC4 128 ビット暗号化、およびメッセージダイジェストに MD5 を使用した RSA のキー交換

5. SSL_RSA_WITH_RC4_128_SHA : RC4 128 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換
6. SSL_RSA_WITH_3DES_EDE_CBC_SHA : メッセージの暗号化に 3DES と DES-EDE3-CBC、およびメッセージダイジェストに SHA を使用した RSA のキー交換 (RSA 公開キー暗号化)
7. SSL_RSA_WITH_AES_128_CBC_SHA : AES 128 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換 (SSL 3.0 専用)。
8. SSL_RSA_WITH_AES_256_CBC_SHA : AES 256 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換 (SSL 3.0 専用)。
9. SSL_RSA_WITH_AES_128_CBC_SHA : AES 128 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換 (SSL 3.0 専用)。
10. SSL_RSA_WITH_AES_256_CBC_SHA : AES 256 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換 (SSL 3.0 専用)。



Note Chrome の最新バージョンは 4 つの元の暗号スイートをサポートしません。そのため、Web GUI とゲストポータル両方へのアクセスが拒否されます。

(暗号化およびダイジェストアルゴリズムをそれぞれ指定して組み合わせた) RSA は、SSL 接続においてキーの生成および認証の両方に使用されます。これは、CA のトラストポイントが設定されているかどうかにかかわらず。

SSL のデフォルト設定

デフォルトの SSL 設定には次の注意事項が適用されます。

- 標準の HTTP サーバはイネーブルに設定されています。
- SSL はイネーブルに設定されています。
- CA のトラストポイントは設定されていません。
- 自己署名証明書は生成されていません。

SSL の設定時の注意事項

SSL をスイッチクラスタで使用すると、SSL セッションがクラスタ コマンドで終了します。クラスタメンバのスイッチは標準の HTTP で動作させる必要があります。

CA のトラストポイントを設定する前に、システムクロックが設定されていることを確認してください。クロックが設定されていないと、不正な日付により証明書が拒否されます。

スイッチスタック内のアクティブスイッチで、SSL セッションが終了します。

Secure Socket Layer HTTP の設定方法

CA のトラストポイントの設定

セキュア HTTP 接続には、CA のトラストポイントを正式に設定することを推奨します。CA のトラストポイントは、自己署名証明書より高いセキュリティがあります。

CA のトラストポイントを設定するには、特権 EXEC モードで次の手順を実行します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname** *hostname*
4. **ip domain name** *domain-name*
5. **crypto key generate rsa**
6. **crypto ca trustpoint** *name*
7. **enrollment url** *url*
8. **enrollment http-proxy** *host-name port-number*
9. **crl query** *url*
10. **primary** *name*
11. **exit**
12. **crypto ca authentication** *name*
13. **crypto ca enroll** *name*
14. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	hostname <i>hostname</i> Example: Device(config)# hostname <i>your_hostname</i>	スイッチのホスト名を指定します（以前ホスト名を設定していない場合のみ必須）。ホスト名はセキュリティ キーと証明書に必要です。
ステップ 4	ip domain name <i>domain-name</i> Example: Device(config)# ip domain name <i>your_domain</i>	スイッチの IP ドメイン名を指定します（以前 IP ドメイン名を設定していない場合のみ必須）。IP ドメイン名はセキュリティ キーと証明書に必要です。

	Command or Action	Purpose
ステップ 5	crypto key generate rsa Example: Device(config)# crypto key generate rsa	(任意) RSA キー ペアを生成します。RSA キーのペアは、スイッチの証明書を入手する前に必要です。RSA キーのペアは自動的に生成されます。必要であれば、このコマンドを使用してキーを再生成できます。
ステップ 6	crypto ca trustpoint name Example: Device(config)# crypto ca trustpoint your_trustpoint	CA のトラストポイントにローカルの設定名を指定して、CA トラストポイントコンフィギュレーションモードを開始します。
ステップ 7	enrollment url url Example: Device(ca-trustpoint)# enrollment url http://your_server:80	スイッチによる証明書要求の送信先の URL を指定します。
ステップ 8	enrollment http-proxy host-name port-number Example: Device(ca-trustpoint)# enrollment http-proxy your_host 49	(任意) HTTP プロキシサーバーを経由して CA から証明書を入手するようにスイッチを設定します。 <ul style="list-style-type: none"> • <i>host-name</i> には、CA を取得するために使用するプロキシサーバーを指定します。 • <i>port-number</i> には、CA にアクセスするために使用するポート番号を指定します。
ステップ 9	crl query url Example: Device(ca-trustpoint)# crl query ldap://your_host:49	ピアの証明書が取り消されていないかを確認するために、証明書失効リスト (CRL) を要求するようにスイッチを設定します。
ステップ 10	primary name Example: Device(ca-trustpoint)# primary your_trustpoint	(任意) トラストポイントが CA 要求に対してプライマリ (デフォルト) トラストポイントとして使用されるように指定します。 <ul style="list-style-type: none"> • <i>name</i> には、設定したトラストポイントを指定します。
ステップ 11	exit Example: Device(ca-trustpoint)# exit	CA トラストポイントコンフィギュレーションモードを終了し、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 12	crypto ca authentication name Example: Device(config)# crypto ca authentication your_trustpoint	CA の公開キーを取得して CA を認証します。ステップ 5 で使用した名前と同じものを使用します。

	Command or Action	Purpose
ステップ 13	<code>crypto ca enroll name</code> Example: Device(config)# <code>crypto ca enroll your_trustpoint</code>	指定した CA トラストポイントから証明書を取得します。このコマンドは、各 RSA キーのペアに対して 1 つの署名入りの証明書を要求します。
ステップ 14	<code>end</code> Example: Device(config)# <code>end</code>	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

セキュア HTTP サーバの設定

セキュア HTTP サーバを設定するには、特権 EXEC モードで次の手順を実行します。

Before you begin

証明に証明書の認証を使用する場合、前の手順を使用してスイッチの CA トラストポイントを設定してから、HTTP サーバを有効にする必要があります。CA のトラストポイントを設定していない場合、セキュア HTTP サーバを最初に有効にした時点で、自己署名証明書が生成されます。サーバを設定した後、標準およびセキュア HTTP サーバ両方に適用するオプション（パス、適用するアクセスリスト、最大接続数、またはタイムアウトポリシー）を設定できます。

Web ブラウザを使用してセキュア HTTP 接続を確認するには、`https://URL` を入力します（URL は IP アドレス、またはサーバ スイッチのホスト名）。デフォルトポート以外のポートを設定している場合、URL の後ろにポート番号も指定する必要があります。次に例を示します。



Note AES256_SHA2 はサポートされません。

```
https://209.165.129:1026
```

または

```
https://host.domain.com:1026
```

アクセスリスト（IPv4 ACL のみ）を指定するための従来の `ip http access-class access-list-number` コマンドは廃止予定です。引き続きこのコマンドを使用して、HTTP サーバへのアクセスを許可するアクセスリストを指定できます。2 つの新しいコマンドは、IPv4 および IPv6 ACL を指定するためのサポートを有効にするために導入されました。これらは、IPv4 ACL を指定するための `ip http access-class ipv4 access-list-name | access-list-number` と、IPv6 ACL を指定するための `ip http access-class ipv6 access-list-name` です。警告メッセージの受信を防ぐために、新しい CLI の使用をお勧めします。

アクセスリストを指定する際は、次の考慮事項があります。

- 存在しないアクセスリストを指定すると、設定は実行されますが、次の警告メッセージを受信します。

ACL being attached does not exist, please configure it

- **ip http access-class ipv4 access-list-name | access-list-number** または **ip http access-class ipv6 access-list-name** を使用した場合には、アクセスリストがすでに **ip http access-class** を使用して設定されていた場合は、次の警告メッセージが表示されます。

Removing ip http access-class <access-list-number>

ip http access-class access-list-number と **ip http access-class ipv4 access-list-name | access-list-number** は同じ機能を共有しています。コマンドを実行するごとに、その前のコマンドのコンフィギュレーションは上書きされます。2つのコマンドの設定間の次の組み合わせによって、実行コンフィギュレーションへの影響が説明されます。

- **ip http access-class access-list-number** がすでに設定されている場合に、**ip http access-class ipv4 access-list-number** コマンドを使用して設定を行おうとした場合、**ip http access-class access-list-number** の設定は削除され、**ip http access-class ipv4 access-list-number** の設定が実行コンフィギュレーションに追加されます。
- **ip http access-class access-list-number** がすでに設定されている場合に、**ip http access-class ipv4 access-list-name** コマンドを使用して設定を行おうとした場合、**ip http access-class access-list-number** の設定は削除され、**ip http access-class ipv4 access-list-name** の設定が実行コンフィギュレーションに追加されます。
- **ip http access-class ipv4 access-list-number** がすでに設定されている場合に、**ip http access-class access-list-name** を使用して設定を行おうとした場合、**ip http access-class ipv4 access-list-number** の設定は削除され、**ip http access-class access-list-name** の設定が実行コンフィギュレーションに追加されます。
- **ip http access-class ipv4 access-list-name** がすでに設定されている場合に、**ip http access-class access-list-number** を使用して設定を行おうとした場合、**ip http access-class ipv4 access-list-name** の設定は削除され、**ip http access-class access-list-number** の設定が実行コンフィギュレーションに追加されます。

SUMMARY STEPS

1. **enable**
2. **show ip http server status**
3. **configure terminal**
4. **ip http secure-server**
5. **ip http secure-port port-number**
6. **ip http secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}**
7. **ip http secure-client-auth**
8. **ip http secure-trustpoint name**
9. **ip http path path-name**
10. **ip http access-class { ipv4 {access-list-number | access-list-name} | ipv6 {access-list-name} }**
11. **ip http max-connections value**
12. **ip http timeout-policy idle seconds life seconds requests value**
13. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	show ip http server status Example: Device# show ip http server status	(任意) HTTP サーバのステータスを表示して、セキュア HTTP サーバの機能がソフトウェアでサポートされているかどうかを判断します。出力で、次のラインのどちらかを確認してください。 HTTP secure server capability: Present または HTTP secure server capability: Not present
ステップ 3	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 4	ip http secure-server Example: Device(config)# ip http secure-server	HTTPS サーバがディセーブルの場合、イネーブルにします。HTTPS サーバは、デフォルトでイネーブルに設定されています。
ステップ 5	ip http secure-port port-number Example: Device(config)# ip http secure-port 443	(任意) HTTPS サーバに使用するポート番号を指定します。デフォルトのポート番号は 443 です。443 または 1025 ~ 65535 の範囲で指定できます。
ステップ 6	ip http secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]} Example: Device(config)# ip http secure-ciphersuite rc4-128-md5	(任意) HTTPS 接続の暗号化に使用する CipherSuite (暗号化アルゴリズム) を指定します。特定の CipherSuite を指定する理由がなければ、サーバとクライアントが、両方がサポートする CipherSuite でネゴシエートするように設定します。これがデフォルトです。
ステップ 7	ip http secure-client-auth Example: Device(config)# ip http secure-client-auth	(任意) HTTP サーバを設定して、接続処理の間、認証のために、クライアントからの X.509v3 証明書を要求します。デフォルトでは、クライアントがサーバからの証明書を要求する設定になっていますが、サーバはクライアントを認証しないようになっています。
ステップ 8	ip http secure-trustpoint name Example: Device(config)# ip http secure-trustpoint your_trustpoint	X.509v3 セキュリティ証明書の取得およびクライアントの証明書接続の認証に使用する CA のトラストポイントを指定します。

	Command or Action	Purpose
		Note このコマンドの使用は、前の手順に従って CA のトラストポイントをすでに設定しているという前提を踏まえて説明しています。
ステップ 9	ip http path <i>path-name</i> Example: Device(config)# ip http path /your_server:80	(任意) HTML ファイルのベースとなる HTTP パスを設定します。パスは、ローカルシステムにある HTTP サーバファイルの場所を指定します (通常、システムのフラッシュメモリを指定します)。
ステップ 10	ip http access-class { ipv4 { <i>access-list-number</i> <i>access-list-name</i> } ipv6 { <i>access-list-name</i> } } Example: Device(config)# ip http access-class ipv4 4	(任意) HTTP サーバへのアクセスの許可に使用するアクセスリストを指定します。
ステップ 11	ip http max-connections <i>value</i> Example: Device(config)# ip http max-connections 4	(任意) HTTP サーバへの同時最大接続数を指定します。値は 10 以上にすることを推奨します。これは、UI が想定どおりに機能するために必要な値です。
ステップ 12	ip http timeout-policy idle <i>seconds</i> life <i>seconds</i> requests <i>value</i> Example: Device(config)# ip http timeout-policy idle 120 life 240 requests 1	(任意) 指定の状況下における、HTTP サーバへの接続最大時間を指定します。 <ul style="list-style-type: none">• idle : データの受信がないか、応答データが送信できない場合の最大時間。指定できる範囲は 1 ~ 600 秒です。デフォルト値は 180 秒 (3分) です。• life : 接続を確立している最大時間。指定できる範囲は 1 ~ 86400 秒 (24時間) です。デフォルト値は 180 秒です。• requests : 永続的な接続で処理される要求の最大数。最大値は 86400 です。デフォルトは 1 です。
ステップ 13	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

セキュア HTTP クライアントの設定

セキュア HTTP クライアントを設定するには、特権 EXEC モードで次の手順を実行します。

Before you begin

標準の HTTP クライアントおよびセキュア HTTP クライアントは常にイネーブルです。証明書の認証にはセキュア HTTP クライアントの証明書が必要です。次の手順では、前の手順で CA のトラストポイントをスイッチに設定していることを前提にしています。CA のトラストポイントが設定されておらず、リモートの HTTPS サーバーがクライアントの認証を要求した場合、セキュア HTTP クライアントへの接続は失敗します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http client secure-trustpoint *name***
4. **ip http client secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip http client secure-trustpoint <i>name</i> Example: Device(config)# ip http client secure-trustpoint your_trustpoint	(任意) リモートの HTTP サーバがクライアント認証を要求した場合に使用する、CA のトラストポイントを指定します。このコマンドの使用は、前の手順を使用して CA のトラストポイントをすでに設定しているという前提を踏まえて説明しています。クライアント認証が必要ない場合、またはプライマリのトラストポイントがすでに設定されている場合は、このコマンドは任意です。
ステップ 4	ip http client secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]} Example: Device(config)# ip http client secure-ciphersuite rc4-128-md5	(任意) HTTPS 接続の暗号化に使用する CipherSuite (暗号化アルゴリズム) を指定します。特定の CipherSuite を指定する理由がなければ、サーバとクライアントが、両方がサポートする CipherSuite でネゴシエートするように設定します。これがデフォルトです。
ステップ 5	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

セキュア HTTP サーバおよびクライアントのステータスのモニタリング

SSL セキュアサーバおよびクライアントのステータスをモニタするには、次の表の特権 EXEC コマンドを使用します。

表 21: SSL セキュアサーバおよびクライアントのステータスを表示するコマンド

コマンド	目的
<code>show ip http client secure status</code>	セキュア HTTP クライアントの設定を表示します。
<code>show ip http server secure status</code>	セキュア HTTP サーバの設定を表示します。
<code>show running-config</code>	セキュア HTTP 接続に対して生成された自己署名証明書を表示します。

Secure Socket Layer HTTP に関するその他の参考資料

関連資料

関連項目	マニュアルタイトル
証明機関	「Configuring Certification Authority Interoperability」

シスコのテクニカル サポート

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

Secure Socket Layer HTTP の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	Secure Socket Layer HTTP	シスコが実装するセキュア HTTP サーバおよび HTTP クライアントでは、アプリケーション層の暗号化に SSL バージョン 3.0 を使用します。セキュア HTTP 接続の場合、HTTP サーバが送受信するデータは暗号化されてインターネットに送信されます。

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



第 17 章

IPv4 ACL

- IPv4 アクセスコントロールリストの制約事項 (347 ページ)
- IPv4 アクセスコントロールリストに関する情報 (349 ページ)
- IPv4 アクセスコントロールリストの設定方法 (363 ページ)
- IPv4 ACL のモニタリング (380 ページ)
- IPv4 アクセスコントロールリストの設定例 (381 ページ)
- IPv4 アクセスコントロールリストに関する追加情報 (394 ページ)
- IPv4 アクセスコントロールリストの機能履歴 (394 ページ)

IPv4 アクセスコントロールリストの制約事項

一般的なネットワーク セキュリティ

次は、ACL によるネットワーク セキュリティの設定の制約事項です。

- 番号付き ACL で使用できるすべてのコマンドが名前付き ACL でも使用できるわけではありません。インターフェイスのパケットフィルタおよびルートフィルタ用の ACL では、名前を使用できます。また、VLAN マップでも名前を指定できます。
- 標準 ACL と拡張 ACL に同じ名前は使用できません。
- **appletalk** は、コマンドラインのヘルプストリングに表示されますが、**deny** および **permit** MAC アクセスリスト コンフィギュレーション モード コマンドの一致条件としてサポートされていません。
- ACL ワイルドカードは、ダウンストリーム クライアント ポリシーではサポートされていません。
- プロトコルの TCAM をプログラムしないインターフェイスと、アンロードされた ACL にスケール ACL を適用すると、他のプロトコルのトラフィックの既存の通常移動に影響を与える可能性があります。IPv6 および MAC アドレストラフィックにこの制限は適用されません。
- ルータ ACL は、CPU 生成トラフィックを含むすべてのタイプのトラフィックに適用されます。

- 出力方向の ACL ロギングは、デバイスのコントロールプレーンから生成されたパケットではサポートされません。
- 存続可能時間 (TTL) 分類は、ACL ではサポートされていません。
- ダウンロード可能な ACL に重複するエントリが含まれている場合、エントリは自動的にマージされません。その結果、802.1Xセッション許可は失敗します。ダウンロード可能な ACL が、同じポートのポートベースのエントリや名前ベースのエントリなど、重複するエントリなしで最適化されていることを確認します。

IPv4 ACL ネットワーク インターフェイス

次の制限事項が、ネットワーク インターフェイスへの IPv4 ACL に適用されます。

- インターフェイスへのアクセスを制御する場合、名前付き ACL または番号付き ACL を使用できます。
- VLAN に属しているレイヤ 2 インターフェイスに ACL を適用した場合、レイヤ 2 (ポート) ACL は VLAN インターフェイスに適用された入力方向のレイヤ 3 ACL、または VLAN に適用された VLAN マップよりも優先します。
- レイヤ 3 インターフェイスに ACL が適用され、スイッチ上でルーティングがイネーブルになっていない場合は、SNMP、Telnet、Web トラフィックなど、CPU で処理されるパケットだけがフィルタリングされます。
- パケットをフィルタリングするために `preauth_ipv4_acl` ACL が設定されている場合、ACL は認証後に削除されます。
- レイヤ 2 インターフェイスに ACL を適用する場合、ルーティングをイネーブルにする必要はありません。

レイヤ 2 インターフェイスの MAC ACL

MAC ACL を作成し、それをレイヤ 2 インターフェイスに適用すると、そのインターフェイスに着信する非 IP トラフィックをフィルタリングできます。MAC ACL を適用するときには、次の注意事項に留意してください。

- 同じレイヤ 2 インターフェイスには、IP アクセス リストと MAC アクセス リストを 1 つずつしか適用できません。IP アクセス リストは IP パケットだけをフィルタリングし、MAC アクセス リストは非 IP パケットをフィルタリングします。
- 1 つのレイヤ 2 インターフェイスに適用できる MAC アドレス リストは 1 つだけです。すでに MAC ACL が設定されているレイヤ 2 インターフェイスに MAC アクセス リストを適用すると、設定済みの ACL が新しい ACL に置き換えられます。



- (注) **mac access-group** インターフェイス コンフィギュレーション コマンドは、物理レイヤ 2 インターフェイスに適用される場合のみ有効です。このコマンドは、EtherChannel ポートチャネルでは使用できません。

IP アクセス リスト エントリ シーケンス番号

- この機能は、ダイナミックアクセスリスト、再帰アクセスリスト、またはファイアウォールアクセス リストをサポートしていません。

IPv4 アクセスコントロールリストに関する情報

ACL の概要

パケットフィルタリングは、ネットワークトラフィックを限定し、特定のユーザまたはデバイスによるネットワークの使用を制限するうえで役立ちます。ACLは、トラフィックをデバイスの通過時にフィルタリングし、パケットが指定されたインターフェイスを通過することを許可または拒否します。ACLは、パケットに適用される許可条件および拒否条件の順序付けられた集まりです。パケットがインターフェイスに着信すると、スイッチはパケット内のフィールドを適用される ACL と比較し、アクセスリストに指定された基準に基づいて、パケットが転送に必要な権限を持っているかどうかを確認します。アクセスリスト内の条件を1つずつ調べ、パケットをテストします。最初に一致した条件によって、スイッチがパケットを受け入れるか拒否するかが決定されます。スイッチは最初に一致した時点でテストを中止するので、リストに条件を指定する順序が重要です。一致する条件がない場合、スイッチはパケットを拒否します。スイッチは、制限条件がない場合はパケットを転送し、制限条件がある場合はパケットをドロップします。スイッチは、転送するすべてのパケット上で ACL を使用できます。

ネットワークに基本的なセキュリティを導入する場合は、デバイスにアクセスリストを設定します。ACLを設定しなければ、スイッチを通過するすべてのパケットがネットワークのあらゆる部分で許可される可能性があります。ACLを使用すると、ネットワークの場所ごとにアクセス可能なホストを制御したり、デバイスインターフェイスで転送またはブロックされるトラフィックの種類を決定できます。たとえば、電子メールトラフィックの転送を許可し、Telnetトラフィックの転送を拒否することもできます。

アクセスコントロール エントリ

ACLには、アクセスコントロールエントリ (ACE) の順序付けられたリストが含まれています。各 ACEには、*permit* または *deny* と、パケットが ACE と一致するために満たす必要のある一連の条件を指定します。*permit* または *deny* の意味は、ACL が使用されるコンテキストによって変わります。

ACL でサポートされるタイプ

デバイスは、IP ACL とイーサネット（MAC）ACL をサポートしています。

- IP ACL は、TCP、ユーザー データグラム プロトコル（UDP）、インターネット グループ 管理プロトコル（IGMP）、およびインターネット制御メッセージプロトコル（ICMP）などの IPv4 トラフィックをフィルタリングします。
- イーサネット ACL は非 IP トラフィックをフィルタリングします。

このデバイスは、Quality of Service（QoS）分類 ACL もサポートしています。

サポートされる ACL

スイッチでは、トラフィックをフィルタリングするために、次に示す 3 種類の ACL がサポートされています。

- ポート ACL は、レイヤ 2 インターフェイスに入るトラフィックをアクセス コントロール します。IPv4 と MAC どちらのアクセスリストタイプのどの方向に対してでも、レイヤ 2 インターフェイスにポート ACL を適用できます。
- ルータ ACL は、VLAN 間でルーティングされたトラフィックのアクセスを制御し、レイヤ 3 インターフェイスで特定の方向（インバウンドまたはアウトバウンド）に適用されます。
- VLAN ACL または VLAN マップはレイヤ 2 VLAN にのみ適用され、ブリッジされたトラフィックにのみ影響します。VLAN マップを使用すると、同じ VLAN 内のデバイス間で 転送されるトラフィックをフィルタリングできます。VLAN マップは、IPv4 のレイヤ 3 アドレスに基づいてアクセスコントロールするように設定されています。イーサネット ACE を使用すると MAC アドレスにより、サポートされていないプロトコルがアクセス コントロールされます。VLAN マップを VLAN に適用すると、VLAN に入るすべてのパケット（ルーテッドパケットまたはブリッジドパケット）が VLAN マップと照合されます。パケットは、スイッチポートを介して、または、ルーティングされたパケットの場合、ルーテッドポートを介して、VLAN に入ることができます。

ACL 優先順位

VLAN マップ、ポート ACL、およびルータ ACL が同じスイッチに設定されている場合、入力トラフィックの場合のフィルタの優先順位は上からポート ACL、VLAN マップ、およびルータ ACL です。出力トラフィックの場合、フィルタの優先順位は、ルータ ACL、VLAN マップ、ポート ACL です。

次の例で、簡単な使用例を説明します。

- 入力ポート ACL と VLAN マップが両方とも適用されている場合に、ポート ACL が適用されたポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。その他のパケットは、VLAN マップによってフィルタリングされます。

- スイッチ仮想インターフェイス（SVI）に入力ルータ ACL および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。他のポートで受信した着信のルーティング IP パケットには、ルータ ACL のフィルタが適用されます。他のパケットはフィルタリングされません。
- SVI に出力ルータ ACL および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。発信するルーティング IP パケットには、ルータ ACL のフィルタが適用されます。他のパケットはフィルタリングされません。
- SVI に VLAN マップ、入力ルータ ACL、および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL だけによってフィルタリングされます。他のポートで受信した着信のルーティング IP パケットには、VLAN マップおよびルータ ACL のフィルタが適用されます。他のパケットには、VLAN マップのフィルタだけが適用されます。
- SVI に VLAN マップ、出力ルータ ACL、および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL だけによってフィルタリングされます。発信するルーティング IP パケットには、VLAN マップおよびルータ ACL のフィルタが適用されます。他のパケットには、VLAN マップのフィルタだけが適用されます。

ポート ACL

ポート ACL は、スイッチのレイヤ 2 インターフェイスに適用される ACL です。ポート ACL は、物理インターフェイスおよび EtherChannel インターフェイス上でサポートされていますが、EtherChannel メンバーインターフェイスではサポートされていません。ポート ACL は、インバウンド方向とアウトバウンド方向のインターフェイスに適用できます。次のアクセスリストがサポートされています。

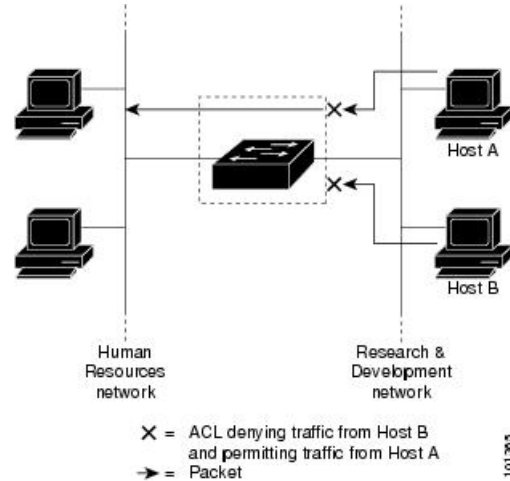
- 送信元アドレスを使用する IP アクセス リスト
- 送信元および宛先のアドレスと任意でプロトコル タイプ情報を使用できる拡張 IP アクセス リスト
- 送信元および宛先の MAC アドレスと任意でプロトコル タイプ情報を使用できる MAC 拡張アクセス リスト

スイッチは、インターフェイス上の ACL を調べ、パケットが ACL 内のエン트리とどのように一致するかに基づいてパケットの転送を許可または拒否します。このように、ACL がネットワークまたはネットワークの部分へのアクセスを制御します。

図 15: ACL によるネットワーク内のトラフィックの制御

次に、すべてのワークステーションが同じ VLAN にある場合にポート ACL を使用してネットワークへのアクセスを制御する例を示します。レイヤ 2 の着信方向に適用された ACL は、ホスト A がヒューマンリソース ネットワークにアクセスすることを許可しますが、ホスト B が

同一のネットワークにアクセスすることは拒否します。ポート ACL は、着信方向のレイヤ 2



インターフェイスだけに適用できます。

ポート ACL をトランク ポートに適用すると、ACL はそのトランク ポート上のすべての VLAN でトラフィックをフィルタリングします。ポート ACL を音声 VLAN ポートに適用すると、ACL はデータ VLAN と音声 VLAN の両方でトラフィックをフィルタリングします。

ポート ACL では、IP アクセスリストを使用して IP トラフィックをフィルタリングでき、MAC アドレスを使用して非 IP トラフィックをフィルタリングできます。同じレイヤ 2 インターフェイス上で IP トラフィックと非 IP トラフィックの両方をフィルタリングするには、そのインターフェイスに IP アクセスリストと MAC アクセスリストの両方を適用します。



- (注) レイヤ 2 インターフェイスに適用できるのは、IP アクセスリスト 1 つと MAC アクセスリスト 1 つだけです。すでに IP アクセスリストまたは MAC アクセスリストが設定されているレイヤ 2 インターフェイスに、新しい IP アクセスリストまたは MAC アクセスリストを適用すると、前に設定した ACL が新しい ACL に置き換わります。

ルータ ACL

VLAN へのレイヤ 3 インターフェイスであるスイッチ仮想インターフェイス (SVI)、物理層 3 インターフェイス、およびレイヤ 3 EtherChannel インターフェイスに、ルータ ACL を適用できます。ルータ ACL はインターフェイスの特定の方向 (着信または発信) に対して適用されます。1 つのインターフェイスの方向ごとに、ルータ ACL を 1 つ適用できます。

スイッチは、IPv4 トラフィックの次のアクセスリストをサポートしています。

- 標準 IP アクセスリストでは、照合操作に送信元アドレスを使用します。
- 拡張 IP アクセスリストは、送信元アドレス、宛先アドレス、およびオプションのプロトコルタイプ情報を使用して一致処理を行います。

ポート ACL の場合と同様、スイッチはインターフェイスに設定されている機能に関連付けられている ACL が照合されます。パケットがスイッチのインターフェイスに着信すると、その

インターフェイスに設定されているすべての着信機能に対応する ACL が照合されます。パケットがルーティングされてからネクストホップに転送されるまでの間に、出力インターフェイスに設定された発信機能に対応するすべての ACL が照合されます。

ACL は ACL 内のエントリとパケットの一致結果に応じて、パケット転送を許可するか、拒否するかを決めます。ACL を使用すると、ネットワーク全体またはネットワークの一部に対するアクセス コントロールが行えます。

VLAN マップ

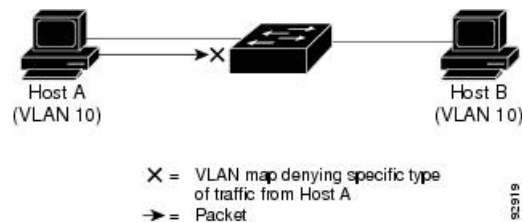
VLAN ACL または VLAN マップは、VLAN 内のネットワークトラフィックを制御するために使用されます。スイッチまたはスイッチ スタックの VLAN 内でブリッジングされるすべてのパケットに VLAN マップを適用できます。VACL は、セキュリティ パケット フィルタリング および特定の物理インターフェイスへのトラフィックのリダイレクトだけを目的としたものです。VACL は方向（入力または出力）で定義されることはありません。

すべての非 IP プロトコルは、MAC VLAN マップを使用して、MAC アドレスおよび Ethertype によってアクセス コントロールされます（IP トラフィックは、MAC VLAN マップではアクセス制御されません）。VLAN マップはスイッチを通過するパケットにだけ適用できます。ハブ上またはこのスイッチに接続された別のスイッチ上のホスト間のトラフィックには、VLAN マップを適用できません。

VLAN マップを使用すると、マップに指定されたアクションに基づいてパケットの転送が許可または拒否されます。

図 16: VLAN マップによるトラフィックの制御

次の図に、VLAN マップを適用して、特定のトラフィック タイプを VLAN 10 のホスト A から転送できないように設定する例を示します。各 VLAN には、VLAN マップを 1 つだけ適用で



きます。

ACE およびフラグメント化されたトラフィックとフラグメント化されていないトラフィック

IP パケットは、ネットワークを通過するときにフラグメント化されることがあります。その場合、TCP または UDP ポート番号や ICMP タイプおよびコードなどのレイヤ 4 情報は、パケットの最初の部分があるフラグメントだけに含まれます。他のフラグメントには、この情報はありません。

アクセス コントロール エントリ (ACE) には、レイヤ 4 情報をチェックしないため、すべてのパケットフラグメントに適用されるものがあります。レイヤ 4 情報を調べる ACE は、フラグメント化された IP パケットのほとんどのフラグメントに標準的な方法では適用できません。

フラグメントにレイヤ4情報が含まれておらず、ACEが一部のレイヤ4情報をチェックする場合、一致ルールは次のように変更されます。

- フラグメント内のレイヤ3情報（TCPやUDPなどのプロトコルタイプを含む）をチェックする許可ACEは、含まれていないレイヤ4情報の種類にかかわらず、フラグメントと一致すると見なされます。



(注) L4 Ops をともなう ACE の TCP では、フラグメント化パケットは RFC 1858 ごとにドロップします。

- レイヤ4情報をチェックする拒否ACEは、フラグメントにレイヤ4情報が含まれていない限り、フラグメントと一致しません。

ACL とスイッチ スタック

スイッチスタックのACLサポートは、スタンドアロンスイッチと同じです。ACLの構成情報は、スタック内のすべてのスイッチに送信されます。アクティブスイッチを含むスタック内のすべてのスイッチでは、情報が処理され、ハードウェアがプログラムされます。

アクティブ スイッチおよび ACL の機能

アクティブ スイッチにより、次の ACL 機能が実行されます。

- ACL 構成情報が処理され、情報がすべてのスタック メンバに送信されます。
- ACL 情報は、スタックに加入しているすべてのスイッチに配信されます。
- (たとえば、十分なハードウェアリソースがないなど) 何らかの理由で、ソフトウェアによってパケットが送信される必要がある場合、ACLをパケットに適用後にのみ、アクティブ スイッチによってパケットが転送されます。
- そのハードウェアは、処理する ACL 情報でプログラムされます。

スタック メンバおよび ACL の機能

スタック メンバにより、次の ACL 機能が実行されます。

- アクティブスイッチから ACL 情報を受信し、ハードウェアがプログラムされます。
- スタンバイ スイッチとして設定されたスタック メンバがアクティブ スイッチが失敗したイベント内のアクティブ スイッチ機能を実行します。

アクティブスイッチの障害および ACL

アクティブとスタンバイの両方のスイッチに ACL 情報があります。アクティブスイッチに障害が発生すると、スタンバイが役割を引き継ぎます。新しいアクティブスイッチにより、すべてのスタックメンバーに ACL 情報が配信されます。

標準 IPv4 ACL および拡張 IPv4 ACL

ACL は、許可条件と拒否条件の順序付けられた集まりです。デバイスは、アクセスリスト内の条件を1つずつ調べ、パケットをテストします。最初に一致した条件によって、デバイスがパケットを受け入れるか拒否するかが決定されます。デバイスは一致する最初の条件が見つかった時点でパケットのテストを停止するため、条件の順序が重要な意味を持ちます。一致する条件がない場合、デバイスはパケットを拒否します。

このソフトウェアは、IPv4 について次の ACL（アクセスリスト）をサポートします。

- 標準 IP アクセスリストでは、照合操作に送信元アドレスを使用します。
- 拡張 IP アクセスリストでは、照合操作に送信元アドレスと宛先アドレスを使用し、任意でプロトコルタイプ情報を使用して制御のきめ細かさを高めることもできます。

IPv4 ACL スイッチでサポートされていない機能

以下の ACL 関連の機能はサポートされていません。

- 非 IP プロトコル ACL
- IP アカウンティング
- 再帰 ACL およびダイナミック ACL はサポートされていません。

アクセスリスト番号

ACL を識別するために使用する番号は、作成するアクセスリストのタイプを表します。

次の一覧に、アクセスリスト番号と対応するアクセスリストタイプを挙げ、このスイッチでサポートされているかどうかを示します。このスイッチは、IPv4 標準アクセスリストおよび拡張アクセスリスト（1～199 および 1300～2699）をサポートします。

表 22: アクセスリスト番号

アクセスリスト番号	タイプ	サポートあり
1～99	IP 標準アクセスリスト	あり
100～199	IP 拡張アクセスリスト	あり
200～299	プロトコルタイプコードアクセスリスト	なし
300～399	DECnet アクセスリスト	なし

アクセス リスト番号	タイプ	サポートあり
400 ~ 499	XNS 標準アクセス リスト	なし
500 ~ 599	XNS 拡張アクセス リスト	なし
600 ~ 699	AppleTalk アクセス リスト	なし
700 ~ 799	48 ビット MAC アドレス アクセス リスト	なし
800 ~ 899	IPX 標準アクセス リスト	なし
900 ~ 999	IPX 拡張アクセス リスト	なし
1000 ~ 1099	IPX SAP アクセス リスト	なし
1100 ~ 1199	拡張 48 ビット MAC サマリー アドレス アクセス リスト	なし
1200 ~ 1299	IPX サマリー アドレス アクセス リスト	なし
1300 ~ 1999	IP 標準アクセス リスト (拡張範囲)	あり
2000 ~ 2699	IP 拡張アクセス リスト (拡張範囲)	あり

番号付き標準 ACL および番号付き拡張 ACL に加え、サポートされる番号を使用して名前付き標準 ACL および名前付き拡張 ACL も作成できます。標準 IP ACL の名前は 1 ~ 99 で、拡張 IP ACL の名前は 100 ~ 199 です。番号付きリストの代わりに名前付き ACL を使用することには、エントリを個別に削除できるという利点があります。

番号付き標準 IPv4 ACL

ACL を作成するときには、ACL の末尾にデフォルトで暗黙的な **deny** ステートメントが追加され、ACL の終わりに到達するまで一致する条件が見つからなかったすべてのパケットに適用されることに注意してください。標準アクセスリストでは、関連付けられた IP ホストアドレス ACL の指定からマスクを省略すると、0.0.0.0 がマスクと見なされます。

デバイスは、**host** 一致条件があるエントリと *don't care* マスク 0.0.0.0 を含む一致条件があるエントリがリストの先頭に移動し、0 以外の *don't care* マスクを含むエントリよりも前に位置するように、標準アクセスリストの順序を書き換えます。そのため、**show** コマンドの出力やコンフィギュレーションファイルでは、ACE が必ずしも入力されたとおりの順序で配置されません。

作成した番号付き標準 IPv4 ACL を VLAN、端末回線、またはインターフェイスに適用できます。

番号付き拡張 IPv4 ACL

標準 ACL では照合に送信元アドレスだけを使用しますが、拡張 ACL では、照合操作に送信元アドレスと宛先アドレスを使用でき、任意でプロトコルタイプ情報を使用して制御のきめ細か

さを高めることができます。番号付き拡張アクセスリストの ACE を作成するときには、作成した ACE がリストの末尾に追加されることに注意してください。番号付きリストでは、ACE の順序を変更したり、リスト内の特定の場所に対して ACE を追加または削除したりできません。

このデバイスは、ダイナミックまたはリフレクシブアクセスリストをサポートしていません。また、タイプオブサービス (ToS) の minimize-monetary-cost ビットに基づくフィルタリングもサポートしていません。

一部のプロトコルには、特定のパラメータやキーワードも適用されます。

拡張 TCP、UDP、ICMP、IGMP、またはその他の IP ACL を定義できます。また、このデバイスは以下の IP プロトコルをサポートします。



(注) ICMP エコー応答はフィルタリングできません。他の ICMP コードまたはタイプは、すべてフィルタリングできます。

これらの IP プロトコルがサポートされます。

- 認証ヘッダー プロトコル (**ahp**)
- カプセル化セキュリティペイロード (**esp**)
- Enhanced Interior Gateway Routing Protocol (**eigrp**)
- 総称ルーティングカプセル化 (**gre**)
- インターネット制御メッセージプロトコル (**icmp**)
- インターネット グループ管理プロトコル (**igmp**)
- すべての内部プロトコル (**ip**)
- IP-in-IP トンネリング (**ipinip**)
- KA9Q NOS 互換 IP over IP トンネリング (**nos**)
- Open Shortest Path First ルーティング (**ospf**)
- ペイロード圧縮プロトコル (**pcp**)
- プロトコル独立マルチキャスト (**pim**)
- 伝送制御プロトコル (**tcp**)
- ユーザ データグラム プロトコル (**udp**)

名前付き IPv4 ACL

IPv4 ACL を識別する手段として、番号ではなく英数字のストリング (名前) を使用できます。名前付き ACL を使用すると、デバイス上で番号付きアクセスリストの場合より多くの IPv4 アクセスリストを設定できます。アクセスリストの識別手段として名前を使用する場合のモード

とコマンド構文は、番号を使用する場合とは多少異なります。ただし、IP アクセス リストを使用するすべてのコマンドを名前付きアクセス リストで使用できるわけではありません。



- (注) 標準 ACL または拡張 ACL に指定する名前は、アクセスリスト番号のサポートされる範囲内の番号にすることもできます。つまり、標準の IP ACL の名前は 1~99 を指定できます。番号付きリストの代わりに名前付き ACL を使用することには、エントリを個別に削除できるという利点があります。

名前付き ACL を設定するときには、次の注意事項に留意してください。

- また、番号付き ACL も使用できます。
- 標準 ACL と拡張 ACL に同じ名前は使用できません。
- VLAN マップには、標準 ACL または拡張 ACL (名前付きまたは番号付き) を使用できません。

ACL ロギング

標準 IP アクセスリストによって許可または拒否されたパケットに関するログメッセージが、デバイスのソフトウェアによって表示されます。つまり、ACL と一致するパケットがあった場合は、そのパケットに関するログ通知メッセージがコンソールに送信されます。コンソールに表示されるメッセージのレベルは、syslog メッセージを管理する **logging console** コマンドで管理されます。



- (注) ACL ロギングは、Unicast Reverse Path Forwarding (uRPF) で使用される ACL ではサポートされません。ルータ ACL でのみサポートされます。



- (注) ルーティングはハードウェアで、ロギングはソフトウェアで実行されます。したがって、**log** キーワードを含む許可 (*permit*) または拒否 (*deny*) ACE と一致するパケットが多数存在する場合、ソフトウェアはハードウェアの処理速度に追いつくことができないため、一部のパケットはロギングされない場合があります。

ACL を起動した最初のパケットについては、ログメッセージがすぐに表示されますが、それ以降のパケットについては、5 分間の収集時間が経過してから表示またはロギングされます。ログメッセージにはアクセスリスト番号、パケットの許可または拒否に関する状況、パケットの送信元 IP アドレス、および直前の 5 分間に許可または拒否された送信元からのパケット数が示されます。



- (注) ログメッセージが多すぎて処理できない場合、または1秒以内に処理する必要があるログメッセージが複数ある場合、ログ設備ではログメッセージパケットの一部をドロップすることがあります。この動作によって、ログメッセージパケットが多すぎてデバイスがクラッシュすることを回避します。そのため、課金ツールや、アクセスリストと一致する数の正確な情報源としてログ設備を使用しないでください。

ハードウェアおよびソフトウェアによる IP ACL の処理

ACL 処理はハードウェアで実行されます。ハードウェアで ACL の設定を保存する領域が不足すると、そのインターフェイス上のすべてのパケットがドロップします。



- (注) デバイスまたはスタックメンバのリソース不足が原因でハードウェアに ACL を設定できない場合、影響を受けるのは、デバイスに着信した該当 VLAN 内のトラフィックだけです。

ルータ ACL の場合は、次の場合にパケットが CPU に送信されることがあります。

- **log** キーワードの使用
- ICMP 到達不能メッセージを生成する。

show ip access-lists 特権 EXEC コマンドを入力した場合、表示される一致カウントには、ハードウェアでアクセスが制御されるパケットは含まれません。スイッチドパケットおよびルーテッドパケットに関するハードウェアの ACL の基本的な統計情報を取得する場合は、**show platform software fed switch { switch_num | active | standby } acl counters hardware** 特権 EXEC コマンドを使用します。

ルータ ACL の機能は、次のとおりです。

- 標準 ACL および拡張 ACL（入力および出力）の許可アクションや拒否アクションをハードウェアで制御し、アクセスコントロールのセキュリティを強化します。
- *ip unreachable* が無効の場合、**log** を指定しないと、セキュリティ ACL の *deny* ステートメントと一致するフローがハードウェアによってドロップされます。許可ステートメントと一致するフローは、ハードウェアでスイッチングされます。
- ルータ ACL の ACE に **log** キーワードを追加すると、パケットのコピーが CPU に送信され、ログメッセージが行われます。ACE が許可ステートメントの場合も、パケットはハードウェアでスイッチングおよびルーティングされます。

VLAN マップの設定時の注意事項

VLAN マップは、VLAN 内でフィルタリングを制御する唯一の方法です。VLAN マップには方向の指定がありません。VLAN マップを使用して、特定の方向のトラフィックをフィルタリン

グするには、特定の送信元または宛先アドレスが指定された ACL を追加する必要があります。VLAN マップ内に該当パケットタイプ（IP または MAC）に対する `match` 句がある場合、デフォルトでは、マップ内のどのエントリにも一致しないパケットはドロップされます。該当パケットタイプに対する `match` コマンドがない場合、デフォルトでは、パケットが転送されません。

次は、VLAN マップ設定の注意事項です。

- インターフェイスでトラフィックを拒否するように設定された ACL がなく、VLAN マップが設定されていない場合、すべてのトラフィックが許可されます。
- 各 VLAN マップは一連のエントリで構成されます。VLAN マップのエントリの順序は重要です。デバイスに着信したパケットは、VLAN マップの最初のエントリに対してテストされます。一致した場合は、VLAN マップのその部分に指定されたアクションが実行されます。一致しなかった場合、パケットはマップ内の次のエントリに対してテストされます。
- 該当パケットタイプ（IP または MAC）に対する `match` 句が VLAN マップに 1 つまたは複数ある場合でも、パケットがそれらの `match` 句に一致しない場合、デフォルトではパケットがドロップされます。該当パケットタイプに対する `match` 句が VLAN マップ内にない場合、デフォルトではパケットが転送されます。
- VLAN マップのロギングはサポートされていません。
- レイヤ 2 インターフェイスに適用された IP アクセスリストまたは MAC アクセスリストがデバイスにあって、ポートが属する VLAN に VLAN マップを適用する場合、ポート ACL が VLAN マップよりも優先されます。
- ハードウェアに VLAN マップの設定を適用できない場合は、その VLAN 内のすべてのパケットがドロップします。

VLAN マップとルータ ACL

ブリッジングされたトラフィックおよびルーティングされたトラフィックの両方に対してアクセスコントロールを行うには、VLAN マップを単独で使用するか、またはルータ ACL と VLAN マップを組み合わせて使用します。入力と出力両方のルーテッド VLAN インターフェイスでルータ ACL を定義したり、ブリッジングされたトラフィックのアクセスをコントロールする VLAN マップを定義したりできます。

パケットフローが ACL 内 VLAN マップの `deny` ステートメントと一致した場合、ルータ ACL の設定に関係なく、パケットフローは拒否されます。



-
- (注) ルータ ACL を VLAN マップと組み合わせて使用し、ルータ ACL でのロギングを必要とするパケットが VLAN マップで拒否された場合、これらのパケットはロギングされません。
-

該当パケットタイプ（IP または MAC）に対する `match` 句が VLAN マップにある場合でも、パケットがそのタイプに一致しない場合、デフォルトではパケットがドロップされます。VLAN

マップ内に `match` 句がなく、アクションが指定されていない場合、どの VLAN マップ エントリとも一致しないパケットは転送されます。

VLAN マップとルータ ACL の設定時の注意事項

ここに記載された注意事項は、ルータ ACL および VLAN マップを同じ VLAN 上で使用する必要がある設定に適用されます。ルータ ACL および VLAN マップを異なる VLAN に割り当てる設定には、これらの注意事項は適用されません。

ルータ ACL および VLAN マップを同じ VLAN に設定する必要がある場合は、ルータ ACL と VLAN マップの両方の設定に関し、ここで説明する注意事項に従ってください。

- VLAN インターフェイス上の各方向（入力および出力）に VLAN マップおよびルータの ACL を 1 つずつに限り設定できます。
- 可能な限り、すべてのエントリのアクションが同一で、末尾のデフォルトアクションだけが反対のタイプとなるように ACL を記述します。次のいずれかの形式を使用して、ACL を記述します。

```
permit... permit... permit... deny ip any any
```

または

```
deny... deny... deny... permit ip any any
```

- ACL 内で複数のアクション（許可、拒否）を定義する場合は、それぞれのアクションタイプをまとめて、エントリ数を削減します。
- ACL 内にレイヤ 4 情報を指定しないでください。レイヤ 4 情報を追加すると、統合プロセスが複雑になります。ACL のフィルタリングが、full-flow（送信元 IP アドレス、宛先 IP アドレス、プロトコル、およびプロトコルポート）でなく、IP アドレス（送信元および宛先）に基づいて行われる場合に、最適な統合結果が得られます。可能な限り、IP アドレスには *don't care* ビットを使用してください。

IP ACE とレイヤ 4 情報を含む TCP/UDP/ICMP ACE が両方とも ACL 内に存在し、full-flow モードを指定する必要があるときは、レイヤ 4 ACE をリストの末尾に配置します。この結果、IP アドレスに基づくトラフィックのフィルタリングが優先されます。

ACL の時間範囲

time-range グローバル コンフィギュレーション コマンドを使用することによって、時刻および曜日に基づいて拡張 ACL を選択的に適用できます。まず、時間範囲の名前を定義し、その時間範囲内の時刻および日付または曜日を設定します。次に、ACL を適用してアクセス リストに制限を設定するときに時間範囲を入力します。時間範囲を使用すると、ACL の許可ステートメントまたは拒否ステートメントの有効期間（指定期間内や指定曜日など）を定義できます。**time-range** キーワードおよび引数については、名前付きおよび番号付き拡張 ACL タスクの表を参照してください。

時間範囲を使用するいくつかの利点を次に示します。

- アプリケーションなどのリソース（IP アドレスとマスクのペア、およびポート番号で識別）へのユーザ アクセスをより厳密に許可または拒否できます。
- ログ メッセージを制御できます。ACL エントリを使用して特定の時刻に関してのみトラフィックをロギングできるため、ピーク時間に生成される多数のログを分析しなくても、簡単にアクセスを拒否できます。

時間ベースのアクセス リストを使用すると、CPU に負荷が生じます。これは、アクセス リストの新規設定を他の機能や、ハードウェアメモリにロードされた結合済みの設定とマージする必要があるためです。そのため、複数のアクセスリストが短期間に連続して（互いに数分以内に）有効となるような設定とならないように注意する必要があります。



- (注) 時間範囲は、デバイスのシステムクロックに基づきます。したがって、信頼できるクロックソースが必要です。ネットワーク タイム プロトコル (NTP) を使用してデバイスクロックを同期させることを推奨します。

IPv4 ACL のインターフェイスに関する注意事項

ip access-group インターフェイス コンフィギュレーション コマンドをレイヤ 3 インターフェイス (SVI、レイヤ 3 EtherChannel、またはルーテッドポート) に適用するには、そのインターフェイスに IP アドレスが設定されている必要があります。レイヤ 3 アクセスグループは、CPU のレイヤ 3 プロセスによってルーティングまたは受信されるパケットをフィルタリングします。このグループは、VLAN 内でブリッジングされるパケットに影響を与えません。

インバウンド ACL の場合、パケットの受信後デバイスはパケットを ACL と照合します。ACL がパケットを許可する場合、デバイスはパケットの処理を継続します。ACL がパケットを拒否する場合、デバイスはパケットを廃棄します。

アウトバウンド ACL の場合、パケットを受信し制御対象インターフェイスにルーティングした後、デバイスはパケットを ACL と照合します。ACL がパケットを許可した場合は、デバイスはパケットを送信します。ACL がパケットを拒否する場合、デバイスはパケットを廃棄します。

デフォルトでは、パケットが廃棄された場合は、その原因が入力インターフェイスの ACL または発信インターフェイスの ACL のいずれであっても、常に入力インターフェイスから ICMP 到達不能メッセージが送信されます。ICMP 到達不能メッセージは通常、入力インターフェイス 1 つにつき、0.5 秒ごとに 1 つだけ生成されます。ただし、この設定は **ip icmp rate-limit unreachable** グローバル コンフィギュレーション コマンドを使用して変更できます。

未定義の ACL をインターフェイスに適用すると、デバイスは ACL がインターフェイスに適用されていないと判断し、すべてのパケットを許可します。ネットワークセキュリティのために未定義の ACL を使用する場合は、このような結果が生じることに注意してください。

IPv4 アクセスコントロールリストの設定方法

IPv4 ACL の設定

このスイッチで IP ACL を使用する手順は次のとおりです。

手順の概要

1. アクセス リストの番号または名前とアクセス条件を指定して、ACL を作成します。
2. その ACL をインターフェイスまたは端末回線に適用します。標準および拡張 IP ACL を VLAN マップに適用することもできます。

手順の詳細

ステップ 1 アクセス リストの番号または名前とアクセス条件を指定して、ACL を作成します。

ステップ 2 その ACL をインターフェイスまたは端末回線に適用します。標準および拡張 IP ACL を VLAN マップに適用することもできます。

番号付き標準 ACL の作成

番号付き標準 ACL を作成するには、次の手順に従ってください。

手順の概要

1. `enable`
2. `configure terminal`
3. `access-list access-list-number {deny | permit} source source-wildcard]`
4. `end`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例： Device> <code>enable</code>	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	<code>configure terminal</code> 例： Device# <code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	<p>access-list <i>access-list-number</i> {deny permit} <i>source source-wildcard</i>]</p> <p>例 :</p> <pre>Device(config)# access-list 2 deny your_host</pre>	<p>送信元アドレスとワイルドカードを使用して標準 IPv4 アクセス リストを定義します。</p> <p><i>access-list-number</i> には、1 ~ 99 または 1300 ~ 1999 の 10 進数を指定します。</p> <p>条件が一致した場合にアクセスを拒否する場合は deny を指定し、許可する場合は permit を指定します。</p> <p><i>source</i> には、パケットの送信元となるネットワークまたはホストのアドレスを次の形式で指定します。</p> <ul style="list-style-type: none"> ドット付き 10 進表記による 32 ビット長の値。 キーワード any は 0.0.0.0 255.255.255.255 という <i>source</i> および <i>source-wildcard</i> の省略形です。 <i>source-wildcard</i> を入力する必要はありません。 キーワード host は送信元および <i>source</i> 0.0.0.0 の <i>source-wildcard</i> の省略形です。 <p>(任意) <i>source-wildcard</i> は、ワイルドカード ビットを送信元アドレスに適用します。</p> <p>(注) ログギングは、レイヤ3 インターフェイスに割り当てられた ACL でだけサポートされます。</p>
ステップ 4	<p>end</p> <p>例 :</p> <pre>Device(config)# end</pre>	<p>グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

番号付き拡張 ACL の作成

番号付き拡張 ACL を作成するには、次の手順に従ってください。

手順の概要

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*]
4. **access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**established**] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*]

5. `access-list access-list-number {deny | permit} udp source source-wildcard [operator port] destination destination-wildcard [operator port] [precedence precedence] [tos tos] [fragments] [time-range time-range-name] [dscp dscp]`
6. `access-list access-list-number {deny | permit} icmp source source-wildcard destination destination-wildcard [icmp-type | [[icmp-type icmp-code] | [icmp-message]]] [precedence precedence] [tos tos] [fragments] [time-range time-range-name] [dscp dscp]`
7. `access-list access-list-number {deny | permit} igmp source source-wildcard destination destination-wildcard [igmp-type] [precedence precedence] [tos tos] [fragments] [time-range time-range-name] [dscp dscp]`
8. `end`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	access-list access-list-number {deny permit} protocol source source-wildcard destination destination-wildcard [precedence precedence] [tos tos] [fragments] [time-range time-range-name] [dscp dscp] 例： Device(config)# access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log	拡張 IPv4 アクセス リストおよびアクセス条件を定義します。 <i>access-list-number</i> には、100 ～ 199 または 2000 ～ 2699 の 10 進数を指定します。 条件が一致した場合にパケットを拒否する場合は deny を指定し、許可する場合は permit を指定します。 source、source-wildcard、destination、および destination-wildcard の値は、次の形式で指定します。 <ul style="list-style-type: none"> • ドット付き 10 進表記による 32 ビット長の値。 • 0.0.0.0 255.255.255.255 (任意のホスト) を表すキーワード any。 • 単一のホスト 0.0.0.0 を表すキーワード host。 その他のキーワードはオプションであり、次の意味を持ちます。 <ul style="list-style-type: none"> • precedence : パケットを 0 ～ 7 の番号または名前で指定する優先度と一致させる場合に入力します。指定できる値は、routine (0)、priority (1)、immediate (2)、flash (3)、

	コマンドまたはアクション	目的
		<p>flash-override (4) 、 critical (5) 、 internet (6) 、 network (7) です。</p> <ul style="list-style-type: none"> • fragments : 2つ目以降のフラグメントを確認する場合に入力します。 • tos : パケットを 0 ～ 15 の番号または名前で指定するサービスタイプレベルと一致させる場合に入力します。指定できる値は、 normal (0) 、 max-reliability (2) 、 max-throughput (4) 、 min-delay (8) です。 • time-range : 時間範囲の名前を指定します。 • dscp : パケットを 0 ～ 63 の番号で指定する DSCP 値と一致させる場合に入力します。または、指定できる値のリストを表示するには、疑問符 (?) を使用します。 <p>(注) dscp 値を入力する場合は、 tos または precedence を入力できません。 dscp を入力せずに tos と precedence の両方の値を入力できます。</p>
ステップ 4	<p>access-list <i>access-list-number</i> {deny permit} tcp <i>source source-wildcard</i> [<i>operator port</i>] <i>destination destination-wildcard</i> [<i>operator port</i>] [established] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>] [<i>flag</i>]</p> <p>例 :</p> <pre>Device(config)# access-list 101 permit tcp any any eq 500</pre>	<p>拡張 TCP アクセス リストおよびアクセス条件を定義します。</p> <p>次に示す例外を除き、拡張 IPv4 ACL に対して説明するパラメータと同じパラメータを使用します。</p> <p>(任意) <i>operator</i> および <i>port</i> を入力すると、送信元ポート (<i>source source-wildcard</i> の後に入力した場合) または宛先ポート (<i>destination destination-wildcard</i> の後に入力した場合) が比較されます。演算子の候補には、eq (次の値に等しい) 、 gt (次の値より大きい) 、 lt (次の値より小さい) 、 neq (次の値に等しくない) 、 および range (次の範囲) があります。演算子にはポート番号を指定する必要があります (range の場合は 2 つのポート番号をスペースで区切って指定する必要があります) 。</p> <p><i>port</i> には、10 進数 (0 ～ 65535) のポート番号または TCP ポート名を入力します。TCP をフィルタリングするときには、TCP ポートの番号または名前だけを使用します。</p>

	コマンドまたはアクション	目的
		<p>他のオプションのキーワードの意味は次のとおりです。</p> <ul style="list-style-type: none"> • established : 確立された接続と照合する場合に入力します。このキーワードは、ack または rst フラグでの照合と同じ機能を果たします。 • flag : 指定された TCP ヘッダービットを基準にして照合します。入力できるフラグは、ack (確認応答)、fin (終了)、psh (プッシュ)、rst (リセット)、syn (同期)、または urg (緊急) です。
ステップ 5	<p>access-list <i>access-list-number</i> {deny permit} udp <i>source source-wildcard</i> [<i>operator port</i>] <i>destination destination-wildcard</i> [<i>operator port</i>] [precedence precedence] [tos tos] [fragments] [time-range time-range-name] [dscp dscp]</p> <p>例 :</p> <pre>Device(config)# access-list 101 permit udp any any eq 100</pre>	<p>(任意) 拡張 UDP アクセス リストおよびアクセス条件を定義します。</p> <p>UDP パラメータは TCP の説明にあるパラメータと同じです。ただし、[<i>operator</i> [<i>port</i>]] ポート番号またはポート名は、UDP ポートの番号または名前を指定する必要があります。また、UDP では、flag キーワードと established キーワードは無効です。</p>
ステップ 6	<p>access-list <i>access-list-number</i> {deny permit} icmp <i>source source-wildcard destination destination-wildcard</i> [<i>icmp-type</i> [[<i>icmp-type icmp-code</i>] [<i>icmp-message</i>]]] [precedence precedence] [tos tos] [fragments] [time-range time-range-name] [dscp dscp]</p> <p>例 :</p> <pre>Device(config)# access-list 101 permit icmp any any 200</pre>	<p>拡張 ICMP アクセス リストおよびアクセス条件を定義します。</p> <p>ICMP パラメータは拡張 IPv4 ACL の IP プロトコルの説明にあるパラメータとほとんど同じですが、ICMP メッセージタイプおよびコードパラメータが追加されています。オプションのキーワードの意味は次のとおりです。</p> <ul style="list-style-type: none"> • icmp-type : ICMP メッセージタイプでフィルタリングする場合に入力します。指定できる値の範囲は、0 ~ 255 です。 • icmp-code : ICMP パケットを ICMP メッセージコードタイプでフィルタリングする場合に入力します。指定できる値の範囲は、0 ~ 255 です。 • icmp-message : ICMP パケットを ICMP メッセージタイプ名または ICMP メッセージタイプとコード名でフィルタリングする場合に入力します。
ステップ 7	<p>access-list <i>access-list-number</i> {deny permit} igmp <i>source source-wildcard destination destination-wildcard</i> [<i>igmp-type</i>] [precedence precedence] [tos tos] [fragments] [time-range time-range-name] [dscp dscp]</p>	<p>(任意) 拡張 IGMP アクセス リストおよびアクセス条件を定義します。</p> <p>IGMP パラメータは拡張 IPv4 ACL の IP プロトコルの説明にあるパラメータとほとんど同じですが、次</p>

	コマンドまたはアクション	目的
	例 : Device(config)# access-list 101 permit igmp any any 14	に示すオプションのパラメータが追加されています。 <i>igmp-type</i> : IGMPメッセージタイプと照合するには、0～15の番号またはメッセージ名 (dvmrp 、 host-query 、 host-report 、 pim 、または trace) を入力します。
ステップ 8	end 例 : Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

名前付き標準 ACL の作成

名前を使用して標準 ACL を作成するには、次の手順に従ってください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list standard name**
4. 次のいずれかを使用します。
 - **deny** {source [source-wildcard] | host source | any} [log]
 - **permit** {source [source-wildcard] | host source | any} [log]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip access-list standard name 例 : Device(config)# ip access-list standard 20	名前を使用して標準 IPv4 アクセスリストを定義し、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。 名前には、1～99の番号を使用できます。

	コマンドまたはアクション	目的
ステップ 4	<p>次のいずれかを使用します。</p> <ul style="list-style-type: none"> • deny {<i>source</i> [<i>source-wildcard</i>] host <i>source</i> any} [log] • permit {<i>source</i> [<i>source-wildcard</i>] host <i>source</i> any} [log] <p>例 :</p> <pre>Device(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255</pre> <p>または</p> <pre>Device(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0</pre>	<p>アクセスリストコンフィギュレーションモードで、パケットを転送するかドロップするのかを決定する 1 つ以上の拒否条件または許可条件を指定します。</p> <ul style="list-style-type: none"> • host <i>source</i> : 送信元および送信元ワイルドカードの値である <i>source</i> 0.0.0.0。 • any : 送信元および送信元ワイルドカードの値である 0.0.0.0 255.255.255.255。
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device(config-std-nacl)# end</pre>	<p>アクセスリスト コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

名前付き拡張 ACL の作成

名前を使用して拡張 ACL を作成するには、次の手順に従ってください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *name*
4. {**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | **host** *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p>enable</p> <p>例 :</p> <pre>Device> enable</pre>	<p>特権 EXEC モードを有効にします。</p> <p>プロンプトが表示されたらパスワードを入力します。</p>
ステップ 2	<p>configure terminal</p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>グローバル コンフィギュレーション モードを開始します。</p>

	コマンドまたはアクション	目的
ステップ 3	ip access-list extended name 例： Device(config)# ip access-list extended 150	名前を使用して拡張 IPv4 アクセスリストを定義し、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。 名前には、100 ~ 199 の番号を使用できます。
ステップ 4	{deny permit} protocol {source [source-wildcard] host source any} {destination [destination-wildcard] host destination any} [precedence precedence] [tos tos] [established] [log] [time-range time-range-name] 例： Device(config-ext-nacl)# permit 0 any any	アクセスリスト コンフィギュレーション モードで、許可条件または拒否条件を指定します。 log キーワードを使用すると、違反を含むアクセスリストのログメッセージを取得できます。 <ul style="list-style-type: none"> • host source : 送信元および送信元ワイルドカードの値である <i>source</i> 0.0.0.0。 • host destination : 宛先および宛先ワイルドカードの値である <i>destination</i> 0.0.0.0。 • any : 送信元および送信元ワイルドカード、または宛先および宛先ワイルドカードの値である 0.0.0.0 255.255.255.255。
ステップ 5	end 例： Device(config-ext-nacl)# end	アクセスリスト コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

拡張 ACL を作成するときには、ACL の末尾にデフォルトで暗黙的な deny ステートメントが追加され、ACL の終わりに到達するまで一致する条件が見つからなかったすべてのパケットに適用されることに注意してください。標準 ACL では、関連付けられた IP ホストアドレス アクセスリストの指定からマスクを省略すると、0.0.0.0 がマスクと見なされます。

ACL の作成後に追加したエントリは、リストの末尾に追加されます。ACL エントリを特定の ACL に選択的に追加できません。ただし、**no permit** および **no deny** アクセスリスト コンフィギュレーション モード コマンドを使用すると、名前付き ACL からエントリを削除できます。

番号付き ACL ではなく名前付き ACL を使用する理由の 1 つとして、名前付き ACL では行を選択して削除できることがあります。

次のタスク

作成した名前付き ACL は、インターフェイスまたは VLAN に適用できます。

ACL の時間範囲の設定

ACL の時間範囲パラメータを設定するには、次の手順に従ってください。

手順の概要

1. **enable**
2. **configure terminal**
3. **time-range** *time-range-name*
4. 次のいずれかを使用します。
 - **absolute** [*start time date*] [*end time date*]
 - **periodic** *day-of-the-week hh:mm to [day-of-the-week] hh:mm*
 - **periodic** { *weekdays* | *weekend* | *daily* } *hh:mm to hh:mm*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	time-range <i>time-range-name</i> 例： Device(config)# time-range workhours	作成する時間範囲には意味のある名前 (<i>workhours</i> など) を割り当て、時間範囲コンフィギュレーションモードを開始します。名前にスペースや疑問符を含めることはできません。また、文字から始める必要があります。
ステップ 4	次のいずれかを使用します。 <ul style="list-style-type: none"> • absolute [<i>start time date</i>] [<i>end time date</i>] • periodic <i>day-of-the-week hh:mm to [day-of-the-week] hh:mm</i> • periodic { <i>weekdays</i> <i>weekend</i> <i>daily</i> } <i>hh:mm to hh:mm</i> 例： Device(config-time-range)# absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006 または Device(config-time-range)# periodic weekdays 8:00 to 12:00	適用対象の機能がいつ動作可能になるかを指定します。 <ul style="list-style-type: none"> • 時間範囲には、absolute ステートメントを 1 つだけ使用できます。複数の absolute ステートメントを設定した場合は、最後に設定したステートメントだけが実行されます。 • 複数の periodic ステートメントを入力できます。たとえば、平日と週末に異なる時間を設定できます。
ステップ 5	end 例： Device(config-time-range)# end	時間範囲コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

次のタスク

複数の項目をそれぞれ異なる時間に有効にする場合は、上記の手順を繰り返してください。

端末回線への IPv4 ACL の適用

番号付き ACL を使用して、1 つまたは複数の端末回線へのアクセスを制御できます。端末回線には名前付き ACL を適用できません。すべての仮想端末回線にユーザが接続する可能性があるため、すべてに同じ制限を設定する必要があります。

仮想端末回線と ACL に指定されたアドレス間の着信接続および発信接続を制限するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **line [console | vty] line-number**
4. **access-class access-list-number {in | out}**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	line [console vty] line-number 例： Device(config)# line console 0	設定する回線を指定し、インライン コンフィギュレーション モードを開始します。 <ul style="list-style-type: none"> • console : コンソール端末回線を指定します。コンソールポートは DCE です。 • vtty : リモートコンソールアクセス用の仮想端末を指定します。 <i>line-number</i> は、回線タイプを指定する場合に、設定する連続グループ内で最初の回線番号です。指定できる範囲は 0 ~ 16 です。

	コマンドまたはアクション	目的
ステップ 4	access-class <i>access-list-number</i> { in out } 例： Device(config-line)# access-class 10 in	(デバイスへの) 特定の仮想端末回線とアクセスリストに指定されたアドレス間の着信接続および発信接続を制限します。
ステップ 5	end 例： Device(config-line)# end	回線コンフィギュレーションモードを終了します。続いて、特権 EXEC モードに戻ります。

インターフェイスへの IPv4 ACL の適用

ここでは、IPv4 ACL をネットワーク インターフェイスへ適用する方法について説明します。インターフェイスへのアクセスを制御する管理には、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip access-group** {*access-list-number* | *name*} {**in** | **out**}
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet1/0/1	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。 インターフェイスには、レイヤ 2 インターフェイス (ポート ACL) またはレイヤ 3 インターフェイス (ルータ ACL) を指定できます。
ステップ 4	ip access-group { <i>access-list-number</i> <i>name</i> } { in out } 例：	指定されたインターフェイスへのアクセスを制御します。

	コマンドまたはアクション	目的
	Device(config-if)# ip access-group 2 in	
ステップ 5	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

名前付き MAC 拡張 ACL の作成

VLAN またはレイヤ 2 インターフェイスで非 IPv4 トラフィックをフィルタリングするには、MAC アドレスおよび名前付き MAC 拡張 ACL を使用します。その手順は、他の名前付き拡張 ACL を設定する場合と同様です。

名前付き MAC 拡張 ACL を作成するには、次の手順に従ってください。

手順の概要

1. **enable**
2. **configure terminal**
3. **mac access-list extended name**
4. **{deny | permit} {any | host source MAC address | source MAC address mask} {any | host destination MAC address | destination MAC address mask} [type mask | lsap lsap mask | aarp | amber | dec-spanning | decnet-iv | diagnostic | dsm | etype-6000 | etype-8042 | lat | larc-sca | mop-console | mop-dump | msdos | mumps | netbios | vines-echo | vines-ip | xns-idp | 0-65535] [cos cos]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mac access-list extended name 例 : Device(config)# mac access-list extended mac1	名前を使用して MAC 拡張アクセス リストを定義します。

	コマンドまたはアクション	目的
ステップ 4	<p>{deny permit} {any host source MAC address source MAC address mask} {any host destination MAC address destination MAC address mask} [type mask lsap lsap mask aarp amber dec-spanning decnet-iv diagnostic dsm etype-6000 etype-8042 lat larc-sca mop-console mop-dump msdos mumps netbios vines-echo vines-ip xns-idp 0-65535] [cos cos]</p> <p>例 :</p> <pre>Device(config-ext-macl)# deny any any decnet-iv</pre> <p>または</p> <pre>Device(config-ext-macl)# permit any any</pre>	<p>拡張 MAC アクセス リスト コンフィギュレーション モードでは、すべての送信元 MAC アドレス、マスク付き送信元 MAC アドレス、または特定のhostの送信元 MAC アドレスと、anyの宛先 MAC アドレス、マスク付き宛先 MAC アドレス、または特定の宛先 MAC アドレスに、permit または deny を指定します。</p> <p>(任意) 次のオプションを入力することもできます。</p> <ul style="list-style-type: none"> • type mask : Ethernet II または SNAP でカプセル化されたパケットの任意の EtherType 番号。10 進数、16 進数、または 8 進数で表記できます。一致検査の前に、任意で指定できる <i>don't care</i> ビットのマスクが EtherType に適用されます。 • lsap lsap mask : IEEE 802.2 でカプセル化されたパケットの LSAP 番号。10 進数、16 進数、または 8 進数で表記できます。任意で <i>don't care</i> ビットのマスクを指定できます。 • aarp amber dec-spanning decnet-iv diagnostic dsm etype-6000 etype-8042 lat larc-sca mop-console mop-dump msdos mumps netbios vines-echo vines-ip xns-idp : 非 IP プロトコル。 • cos cos : プライオリティを設定する 0 ~ 7 の IEEE 802.1Q CoS 番号。
ステップ 5	<p>end</p> <p>例 :</p> <pre>Device(config-ext-macl)# end</pre>	<p>拡張 MAC アクセスリスト コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>

レイヤ2 インターフェイスへの MAC ACL の適用

レイヤ2 インターフェイスへのアクセスを制御するために MAC アクセス リストを適用するには、次の手順を実行します。

手順の概要

1. enable
2. configure terminal

3. **interface** *interface-id*
4. **mac access-group** {*name*} {**in** | **out**}
5. **end**
6. **show mac access-group** [**interface** *interface-id*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet1/0/2	特定の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。 指定するインターフェイスは物理レイヤ2インターフェイス（ポート ACL）でなければなりません。
ステップ 4	mac access-group { <i>name</i> } { in out }	MAC アクセス リストを使用して、指定されたインターフェイスへのアクセスを制御します。 ポート ACL はアウトバウンドおよびインバウンド方向でサポートされます。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show mac access-group [interface <i>interface-id</i>] 例： Device# show mac access-group interface gigabitethernet1/0/2	そのインターフェイスまたはすべてのレイヤ2インターフェイスに適用されている MAC アクセス リストを表示します。

デバイスは、パケットを受信すると、インバウンド ACL とパケットを照合します。ACL がパケットを許可する場合、デバイスはパケットの処理を継続します。ACL がパケットを拒否する場合、デバイスはパケットを廃棄します。未定義の ACL をインターフェイスに適用すると、デバイスは ACL がインターフェイスに適用されていないものとして、すべてのパケットを許可します。ネットワーク セキュリティのために未定義の ACL を使用する場合は、このような結果が生じることに注意してください。

VLAN マップの設定

VLAN マップを作成して、1つまたは複数の VLAN に適用するには、次のステップを実行します。

始める前に

VLAN に適用する標準 IPv4 ACL または拡張 IP ACL、または名前付き MAC 拡張 ACL を作成します。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan access-map name [number]**
4. **match {ip | mac} address {name | number} [name | number]**
5. IP パケットまたは非 IP パケットを（既知の 1 MAC アドレスのみを使って）指定し、1 つ以上の ACL（標準または拡張）とそのパケットを照合するには、次のコマンドのいずれかを入力します。
 - **action { forward }**
 Device (config-access-map) # **action forward**
 - **action { drop }**
 Device (config-access-map) # **action drop**
6. **exit**
7. **vlan filter mapname vlan-list list**
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan access-map name [number] 例： Device (config) # vlan access-map map1 20	VLAN マップを作成し、名前と、任意で番号を付けます。番号は、マップ内のエントリのシーケンス番号です。 同じ名前の VLAN マップを作成すると、10 ずつ増加する番号が順に割り当てられます。マップを変更

	コマンドまたはアクション	目的
		<p>または削除するときは、該当するマップエントリの番号を入力できます。</p> <p>VLAN マップでは、特定の permit または deny キーワードを使用しません。VLAN マップを使用してパケットを拒否するには、パケットを照合する ACL を作成して、アクションをドロップに設定します。ACL 内の permit は、一致するという意味です。ACL 内の deny は、一致しないという意味です。</p> <p>このコマンドを入力すると、アクセス マップ コンフィギュレーション モードに変わります。</p>
ステップ 4	<p>match {ip mac} address {name number} [name number]</p> <p>例 :</p> <pre>Device(config-access-map) # match ip address ip2</pre>	<p>1 つまたは複数の標準または拡張アクセス リストに対してパケットを照合します (IP または MAC アドレスを使用)。パケットの照合は、対応するプロトコルタイプのアクセス リストに対してだけ行われます。IP パケットは、標準または拡張 IP アクセスリストに対して照合されます。非 IP パケットは、名前付き MAC 拡張アクセス リストに対してだけ照合されます。</p> <p>(注) パケットタイプ (IP または MAC) に対する match 句が VLAN マップに設定されている場合で、そのマップアクションがドロップの場合は、そのタイプに一致するすべてのパケットがドロップされます。match 句が VLAN マップになく、設定されているアクションがドロップの場合は、すべての IP およびレイヤ 2 パケットがドロップされます。</p>
ステップ 5	<p>IP パケットまたは非 IP パケットを (既知の 1 MAC アドレスのみを使って) 指定し、1 つ以上の ACL (標準または拡張) とそのパケットを照合するには、次のコマンドのいずれかを入力します。</p> <ul style="list-style-type: none"> • action { forward } <pre>Device(config-access-map) # action forward</pre> <ul style="list-style-type: none"> • action { drop } <pre>Device(config-access-map) # action drop</pre>	<p>マップエントリに対するアクションを設定します。</p>
ステップ 6	<p>exit</p> <p>例 :</p> <pre>Device(config-access-map) # exit</pre>	<p>アクセスマップ コンフィギュレーション モードを終了して、グローバルコンフィギュレーションモードに戻ります。</p>

	コマンドまたはアクション	目的
ステップ 7	vlan filter mapname vlan-list list 例： Device(config)# vlan filter map1 vlan-list 20-22	VLAN マップを 1 つまたは複数の VLAN に適用します。 list には単一の VLAN ID (22)、連続した範囲 (10 ~ 22)、または VLAN ID のストリング (12、22、30) を指定できます。カンマやハイフンの前後にスペースを挿入することもできます。
ステップ 8	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

VLAN への VLAN マップの適用

VLAN マップを 1 つまたは複数の VLAN に適用するには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan filter mapname vlan-list list**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan filter mapname vlan-list list 例： Device(config)# vlan filter map 1 vlan-list 20-22	VLAN マップを 1 つまたは複数の VLAN に適用します。 list には単一の VLAN ID (22)、連続した範囲 (10 ~ 22)、または VLAN ID のストリング (12、22、30) を指定できます。カンマやハイフンの前後にスペースを挿入することもできます。
ステップ 4	end 例：	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

コマンドまたはアクション	目的
Device(config)# end	

IPv4 ACL のモニタリング

デバイスに設定されている ACL、およびインターフェイスと VLAN に適用された ACL を表示して IPv4 ACL をモニターできます。

ip access-group インターフェイス コンフィギュレーション コマンドを使用して、レイヤ 2 またはレイヤ 3 インターフェイスに ACL を適用した場合は、そのインターフェイスのアクセスグループを表示できます。また、レイヤ 2 インターフェイスに適用された MAC ACL も表示できます。この情報を表示するには、次の表に記載された特権 EXEC コマンドを使用します。

表 23: アクセスリストおよびアクセスグループを表示するコマンド

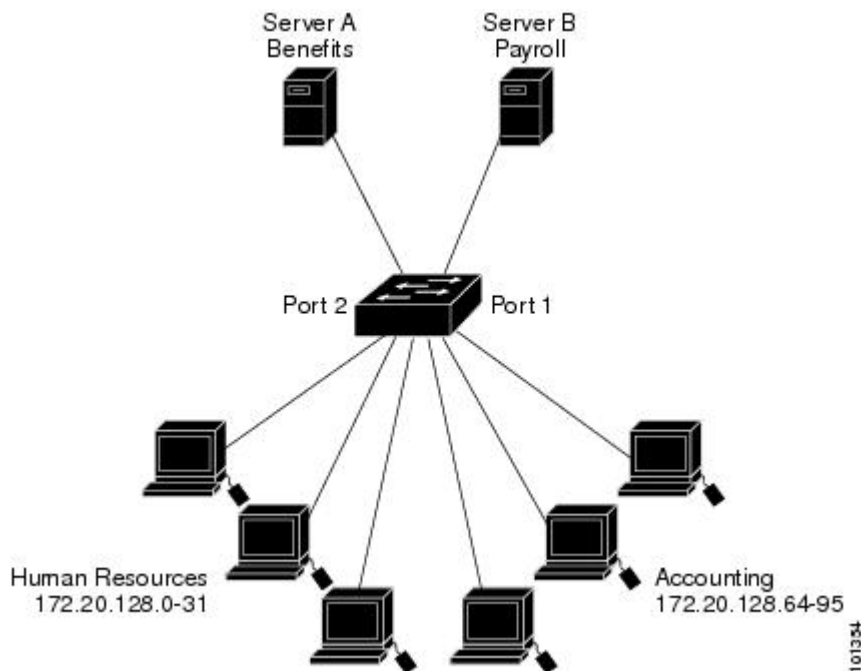
コマンド	目的
show access-lists [<i>number</i> <i>name</i>]	最新の IP および MAC アドレス アクセスリストの全体や、または特定のアクセスリスト（番号付きまたは名前付き）を表示します。
show ip access-lists [<i>number</i> <i>name</i>]	最新の IP アクセスリスト全体、または特定の IP アクセスリスト（番号付きまたは名前付き）を表示します。
show ip interface <i>interface-id</i>	インターフェイスの詳細設定およびステータスを表示します。ネットワークになっているインターフェイスに、 ip access-group インターフェイス コンフィギュレーション コマンドを使用して ACL が適用された場合は、アクセスグループも表示に含まれます。
show running-config [<i>interface interface-id</i>]	デバイスまたは指定されたインターフェイスのコンフィギュレーション ファイルの内容（設定されたすべての MAC および IP アドレス、アクセスリストや、どのアクセスグループがインターフェイスに適用されたか）を表示します。
show mac access-group [<i>interface interface-id</i>]	すべてのレイヤ 2 インターフェイスまたは指定されたレイヤ 2 インターフェイスに適用されている MAC アクセスリストを表示します。

IPv4 アクセスコントロールリストの設定例

小規模ネットワークが構築されたオフィス用の ACL

図 17: ルータ ACL によるトラフィックの制御

次に、小規模ネットワークが構築されたオフィス環境を示します。ルーテッドポート2に接続されたサーバー A には、すべての従業員がアクセスできる収益などの情報が格納されています。ルーテッドポート1に接続されたサーバー B には、機密扱いの給与支払いデータが格納されています。サーバー A にはすべてのユーザーがアクセスできますが、サーバー B にアクセスできるユーザーは制限されています。



ルータ ACL を使用して上記のように設定するには、次のいずれかの方法を使用します。

- 標準 ACL を作成し、ポート 1 からサーバーに着信するトラフィックをフィルタリングします。
- 拡張 ACL を作成し、サーバーからポート 1 に着信するトラフィックをフィルタリングします。

例：小規模ネットワークが構築されたオフィスの ACL

次に、標準 ACL を使用してポートからサーバー B に着信するトラフィックをフィルタリングし、経理部の送信元アドレス 172.20.128.64 ~ 172.20.128.95 から送信されるトラフィックだけ

を許可する例を示します。この ACL は、指定された送信元アドレスを持つルーテッドポート 1 から送信されるトラフィックに適用されます。

```
Device> enable
Device# configure terminal
Device(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Device(config)# exit
Device# show access-lists
```

```
Standard IP access list 6
 10 permit 172.20.128.64, wildcard bits 0.0.0.31
```

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 6 out
Device(config-if)# end
```

次に、拡張 ACL を使用してサーバー B からポートに着信するトラフィックをフィルタリングし、任意の送信元アドレス（この場合はサーバー B）から経理部の宛先アドレス 172.20.128.64 ~ 172.20.128.95 に送信されるトラフィックだけを許可する例を示します。この ACL は、ルーテッドポート 1 に着信するトラフィックに適用され、指定の宛先アドレスに送信されるトラフィックだけを許可します。拡張 ACL を使用する場合は、送信元および宛先情報の前に、プロトコル（IP）を入力する必要があります。

```
Device(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Device(config)# exit
Device# show access-lists
```

```
Extended IP access list 106
 10 permit ip any 172.20.128.64 0.0.0.31
```

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 106 in
Device(config-if)# end
```

例：番号付き ACL

次の例のネットワーク 10.0.0.0 は、2 番目のオクテットがサブネットを指定するクラス A ネットワークです。つまり、サブネットマスクは 255.255.0.0 です。ネットワーク 10.0.0.0 アドレスの 3 番目および 4 番目のオクテットで特定のホストを指定します。アクセスリスト 2 を使用して、サブネット 48 のアドレスを 1 つ許可し、同じサブネットの他のアドレスはすべて拒否します。このアクセスリストの最終行は、ネットワーク 10.0.0.0 の他のすべてのサブネット上のアドレスが許可されることを示します。この ACL は、ポートに着信するパケットに適用されます。

```
Device> enable
Device# configure terminal
Device(config)# access-list 2 permit 10.48.0.3
Device(config)# access-list 2 deny 10.48.0.0 0.0.255.255
Device(config)# access-list 2 permit 10.0.0.0 0.255.255.255
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group 2 in
Device(config-if)# end
```

例：拡張 ACL

次の例の先頭行は、1023 よりも大きい宛先ポートへの着信 TCP 接続を許可します。2 行目で、ホスト 172.16.0.0 の Simple Mail Transfer Protocol (SMTP) ポートへの着信 TCP 接続を許可しています。3 番目の行は、エラー フィードバック用の着信 ICMP メッセージを許可します。

```
Device> enable
Device# configure terminal
Device(config)# access-list 102 permit tcp any 172.16.0.0 0.0.255.255 gt 1023
Device(config)# access-list 102 permit tcp any host 172.16.1.2 eq 25
Device(config)# access-list 102 permit icmp any any
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# ip access-group 102 in
Device(config-if)# end
```

次の例では、インターネットに接続されたネットワークがあり、そのネットワーク上の任意のホストがインターネット上の任意のホストと TCP 接続を確立できるようにする場合を想定しています。ただし、IP ホストからは、専用メールホストのメール (SMTP) ポートを除き、ネットワーク上のホストと TCP 接続を確立できないようにします。

SMTP は、接続の一端では TCP ポート 25、もう一端ではランダムなポート番号を使用します。接続している間は、同じポート番号が使用されます。インターネットから着信するメールパケットの宛先ポートは 25 です。安全なネットワークシステムでは常にポート 25 でのメール接続が使用されているため、着信サービスは個別に制御されます。

```
Device> enable
Device# configure terminal
Device(config)# access-list 102 permit tcp any 172.16.0.0 0.0.255.255 eq 23
Device(config)# access-list 102 permit tcp any 172.16.0.0 0.0.255.255 eq 25
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip access-group 102 in
Device(config-if)# end
```

次の例では、ネットワークはアドレスが 172.16.0.0 のクラス B ネットワークで、メールホストのアドレスは 172.16.1.2 です。**established** キーワードは、確立された接続を表示する TCP 専用のキーワードです。TCP データグラムに ACK または RST ビットが設定され、パケットが既存の接続に属していることが判明すると、一致と見なされます。ギガビットイーサネットインターフェイス 1 は、デバイスをインターネットに接続するインターフェイスです。

```
Device> enable
Device# configure terminal
Device(config)# access-list 102 permit tcp any 172.16.0.0 0.0.255.255 established
Device(config)# access-list 102 permit tcp any host 172.16.1.2 eq 25
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip access-group 102 in
Device(config-if)# end
```

例：名前付き ACL

名前付き標準 ACL および名前付き拡張 ACL の作成

次に、*Internet_filter* という名前の標準 ACL および *marketing_group* という名前の拡張 ACL を作成する例を示します。*Internet_filter* ACL は、送信元アドレス 10.2.3.4 から送信されるすべてのトラフィックを許可します。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list standard Internet_filter
Device(config-ext-nacl)# permit 10.2.3.4
Device(config-ext-nacl)# exit
Device(config-ext-nacl)# end
```

marketing_group ACL は、宛先アドレスとワイルドカードの値 172.16.0.0/24 への任意の TCP Telnet トラフィックを許可し、その他の TCP トラフィックを拒否します。ICMP トラフィックを許可し、任意の送信元から、宛先ポートが 1024 より小さい 172.16.0.0/24 宛先アドレスへ送信される UDP トラフィックを拒否します。それ以外のすべての IP トラフィックを拒否して、結果を示すログが表示されます。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended marketing_group
Device(config-ext-nacl)# permit tcp any 172.16.0.0 0.0.255.255 eq telnet
Device(config-ext-nacl)# deny tcp any any
Device(config-ext-nacl)# permit icmp any any
Device(config-ext-nacl)# deny udp any 172.16.0.0 0.0.255.255 lt 1024
Device(config-ext-nacl)# deny ip any any log
Device(config-ext-nacl)# end
```

Internet_filter ACL は発信トラフィックに適用され、*marketing_group* ACL はレイヤ 3 ポートの着信トラフィックに適用されます。

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet3/0/2
Device(config-if)# no switchport
Device(config-if)# ip address 10.0.5.1 255.255.255.0
Device(config-if)# ip access-group Internet_filter out
Device(config-if)# ip access-group marketing_group in
Device(config-if)# end
```

名前付き ACL からの個別 ACE の削除

次に、名前付きアクセスリスト *border-list* から ACE を個別に削除する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended border-list
Device(config-ext-nacl)# no permit ip host 10.1.1.3 any
Device(config-ext-nacl)# end
```

例：ACL ロギング

ルータ ACL では、2 種類のロギングがサポートされています。**log** キーワードを指定すると、エントリと一致するパケットに関するログ通知メッセージがコンソールに送信されます。

log-input キーワードを指定すると、ログ エントリに入力インターフェイスが追加されます。

次の例では、名前付き標準アクセス リスト *stan1* は 10.1.1.0 0.0.0.255 からのトラフィックを拒否し、その他のすべての送信元からのトラフィックを許可します。**log** キーワードも指定されています。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list standard stan1
Device(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Device(config-std-nacl)# permit any log
Device(config-std-nacl)# exit
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group stan1 in
Device(config-if)# end
Device# show logging

Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
  Console logging: level debugging, 37 messages logged
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 37 messages logged
  File logging: disabled
  Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

次に、名前付き拡張アクセス リスト *ext1* によって、任意の送信元から 10.1.1.0 0.0.0.255 への ICMP パケットを許可し、すべての UDP パケットを拒否する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended ext1
Device(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Device(config-ext-nacl)# deny udp any any log
Device(config-ext-nacl)# exit
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip access-group ext1 in
Device(config)# end
```

次に、拡張 ACL のログの例を示します。

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1
packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8
packets
```

IP ACL のすべてのロギング エントリは %SEC-6-IPACCESSLOGDP で開始します。エントリの形式は、一致した ACL やアクセス エントリの種類に応じて若干異なります。

次に、**log-input** キーワードを指定した場合の出力メッセージの例を示します。

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1 0001.42ef.a400)
->
10.1.1.61 (0/0), 1 packet
```

log キーワードを指定した場合、同様のパケットに関するログ メッセージには入力インターフェイス情報が含まれません。

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0),
1 packet
```

例：ACE およびフラグメント化されたトラフィックとフラグメント化されていないトラフィック

次のコマンドで構成され、フラグメント化された3つのパケットに適用されるアクセスリスト 102 を例にとって説明します。

```
Device> enable
Device# configure terminal
Device(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Device(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Device(config)# access-list 102 permit tcp any host 10.1.1.2
Device(config)# access-list 102 deny tcp any any
Device(config)# end
```



(注) 最初の2つの ACE には宛先アドレスの後に *eq* キーワードがありますが、これは既知の TCP 宛先ポート番号がそれぞれシンプル メール転送プロトコル (SMTP) および Telnet と一致するかどうかをチェックすることを意味します。

- パケット A は、ホスト 10.2.2.2 のポート 65000 からホスト 10.1.1.1 の SMTP ポートに送信される TCP パケットです。このパケットがフラグメント化された場合、レイヤ 4 情報がすべて揃っているため、完全なパケットである場合と同じように最初のフラグメントが最初の ACE (*permit*) と一致します。残りのフラグメントも最初の ACE と一致します。これは、それらのフラグメントに SMTP ポート情報が含まれていなくても、最初の ACE が適用されたときにレイヤ 3 情報だけをチェックするからです。この例の情報は、パケットが TCP であることと、宛先が 10.1.1.1 であることです。
- パケット B は、ホスト 10.2.2.2 のポート 65001 からホスト 10.1.1.2 の Telnet ポートに送信されます。このパケットがフラグメント化された場合、レイヤ 3 情報とレイヤ 4 情報がすべて揃っているため、最初のフラグメントが2つめの ACE (*deny*) と一致します。残りのフラグメントは、レイヤ 4 情報が含まれていないため、2つめの ACE と一致しません。残りのフラグメントは3つめの ACE (*permit*) と一致します。

最初のフラグメントが拒否されたため、ホスト 10.1.1.2 は完全なパケットを再構成できず、その結果、パケット B は拒否されます。ただし、以降の許可されたフラグメントがネット

ワークの帯域幅を使用し、ホスト 10.1.1.2 がパケットを再構成しようとするときにホストのリソースが消費されます。

- フラグメント化されたパケット C は、ホスト 10.2.2.2 のポート 65001 からホスト 10.1.1.3 のポート ftp に送信されます。このパケットがフラグメント化された場合、最初のフラグメントが 4 つめの ACE (deny) と一致します。ACE はレイヤ 4 情報をチェックせず、すべてのフラグメントのレイヤ 3 情報に宛先がホスト 10.1.1.3 であることが示され、前の permit ACE は異なるホストをチェックしていたため、他のフラグメントもすべて 4 つめの ACE と一致します。

例：ACLでの時間範囲を使用

次の例に、*workhours*（営業時間）の時間範囲および会社の休日（2006年1月1日）を設定し、設定を確認する例を示します。

```
Device# show time-range

time-range entry: new_year_day_2003 (inactive)
  absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
  periodic weekdays 8:00 to 12:00
  periodic weekdays 13:00 to 17:00
```

時間範囲を適用するには、時間範囲を実装できる拡張 ACL 内に時間範囲名を入力します。次に、拡張アクセスリスト 188 を作成して確認する例を示します。このアクセスリストでは、定義された休業時間中はすべての送信元からすべての宛先への TCP トラフィックを拒否し、営業時間中はすべての TCP トラフィックを許可します。

```
Device> enable
Device# configure terminal
Device(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Device(config)# access-list 188 permit tcp any any time-range workhours
Device(config)# exit
Device# show access-lists

Extended IP access list 188
  10 deny tcp any any time-range new_year_day_2006 (inactive)
  20 permit tcp any any time-range workhours (inactive)
```

次に、名前付き ACL を使用して同じトラフィックを許可および拒否する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended deny_access
Device(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Device(config-ext-nacl)# exit
Device(config)# ip access-list extended may_access
Device(config-ext-nacl)# permit tcp any any time-range workhours
Device(config-ext-nacl)# end
Device# show ip access-lists

Extended IP access list lpip_default
  10 permit ip any any
Extended IP access list deny_access
  10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
  10 permit tcp any any time-range workhours (inactive)
```

例：IP ACL に適用される時間範囲

次に、月曜日から金曜日の午前 8 時～午後 6 時（18 時）の間、IP の HTTP トラフィックを拒否する例を示します。UDP トラフィックは、土曜日および日曜日の正午～午後 8 時（20 時）の間だけ許可されます。

```
Device> enable
Device# configure terminal
Device(config)# time-range no-http
Device(config)# periodic weekdays 8:00 to 18:00
Device(config)# time-range udp-yes
Device(config)# periodic weekend 12:00 to 20:00
Device(config)# ip access-list extended strict
Device(config-ext-nacl)# deny tcp any any eq www time-range no-http
Device(config-ext-nacl)# permit udp any any time-range udp-yes
Device(config-ext-nacl)# exit
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group strict in
Device(config-if)# end
```

例：ACL へのコメントの挿入

remark キーワードを使用すると、任意の IP 標準または拡張 ACL にエントリに関するコメント（注釈）を追加できます。コメントを使用すると、ACL の理解とスキャンが容易になります。1 つのコメント行の最大長は 100 文字です。

コメントは、**permit** ステートメントまたは **deny** ステートメントの前後どちらにでも配置できます。コメントがどの **permit** ステートメントまたは **deny** ステートメントの説明であるのかが明確になるように、コメントの位置に関して一貫性を保つ必要があります。たとえば、あるコメントは対応する **permit** または **deny** ステートメントの前にあり、他のコメントは対応するステートメントの後ろにあると、混乱を招く可能性があります。

番号付き IP 標準または拡張 ACL にコメントを挿入するには、**access-list access-list number remark remark** グローバルコンフィギュレーションコマンドを使用します。コメントを削除するには、このコマンドの **no** 形式を使用します。

次の例では、**user1** のワークステーションにはアクセスを許可し、**user2** のワークステーションにはアクセスを許可しません。

```
Device> enable
Device# configure terminal
Device(config)# access-list 1 remark Permit only user1 workstation through
Device(config)# access-list 1 permit 171.69.2.88
Device(config)# access-list 1 remark Do not allow user2 through
Device(config)# access-list 1 deny 171.69.3.13
Device(config)# end
```

名前付き IP ACL のエントリには、**remark** アクセスリスト コンフィギュレーション コマンドを使用します。コメントを削除するには、このコマンドの **no** 形式を使用します。

次の例では、サブネット **subnet1** にはアウトバウンド Telnet の使用が許可されません。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow subnet1 subnet to telnet out
```

```
Device(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
Device(config-ext-nacl)# end
```

例：パケットを拒否する ACL および VLAN マップの作成

ここでは、パケットを拒否する ACL および VLAN マップを作成する例を示します。最初のマップでは、*ip1* ACL (TCP パケット) に一致するすべてのパケットがドロップされます。最初に、すべての TCP パケットを許可し、それ以外のパケットをすべて拒否する *ip1* ACL を作成します。VLAN マップには IP パケットに対する *match* 句が存在するため、デフォルトのアクションでは、どの *match* 句とも一致しない IP パケットがすべてドロップされます。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended ip1
Device(config-ext-nacl)# permit tcp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map_1 10
Device(config-access-map)# match ip address ip1
Device(config-access-map)# action drop
Device(config-access-map)# end
```

例：パケットを許可する ACL および VLAN マップの作成

次に、パケットを許可する VLAN マップを作成する例を示します。ACL *ip2* は UDP パケットを許可し、*ip2* ACL と一致するすべてのパケットが転送されます。このマップでは、これ以前のどの ACL とも一致しないすべての IP パケット (TCP でも UDP でもないパケット) がドロップされます。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended ip2
Device(config-ext-nacl)# permit udp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map_1 20
Device(config-access-map)# match ip address ip2
Device(config-access-map)# action forward
Device(config-access-map)# exit
```

例：IP パケットのドロップおよび MAC パケットの転送のデフォルトアクション

次の例の VLAN マップでは、デフォルトで IP パケットがドロップされ、MAC パケットが転送されます。標準の ACL 101 および名前付き拡張アクセスリスト *igmp-match* および *tcp-match* をこのマップと組み合わせて使用すると、次のようになります。

- すべての UDP パケットが転送されます。
- すべての IGMP パケットがドロップされます。
- すべての TCP パケットが転送されます。
- その他のすべての IP パケットがドロップされます。

- すべての非 IP パケットが転送されます。

```
Device> enable
Device# configure terminal
Device(config)# access-list 101 permit udp any any
Device(config)# ip access-list extended igmp-match
Device(config-ext-nacl)# permit igmp any any
Device(config)# action forward
Device(config-ext-nacl)# permit tcp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map drop-ip-default 10
Device(config-access-map)# match ip address 101
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-ip-default 20
Device(config-access-map)# match ip address igmp-match
Device(config-access-map)# action drop
Device(config-access-map)# exit
Device(config)# vlan access-map drop-ip-default 30
Device(config-access-map)# match ip address tcp-match
Device(config-access-map)# action forward
Device(config-access-map)# end
```

例：MAC パケットのドロップおよび IP パケットの転送のデフォルトアクション

次の例の VLAN マップでは、デフォルトで MAC パケットがドロップされ、IP パケットが転送されます。MAC 拡張アクセスリスト **good-hosts** および **good-protocols** をこのマップと組み合わせると、次のようになります。

- ホスト 0000.0c00.0111 および 0000.0c00.0211 からの MAC パケットが転送されます。
- decnet-iv または vines-ip プロトコルを使用する MAC パケットが転送されます。
- その他のすべての非 IP パケットがドロップされます。
- すべての IP パケットが転送されます。

```
Device> enable
Device# configure terminal
Device(config)# mac access-list extended good-hosts
Device(config-ext-macl)# permit host 000.0c00.0111 any
Device(config-ext-macl)# permit host 000.0c00.0211 any
Device(config-ext-nacl)# exit
Device(config)# action forward
Device(config-ext-macl)# mac access-list extended good-protocols
Device(config-ext-macl)# permit any any vines-ip
Device(config-ext-nacl)# exit
Device(config)# vlan access-map drop-mac-default 10
Device(config-access-map)# match mac address good-hosts
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-mac-default 20
Device(config-access-map)# match mac address good-protocols
Device(config-access-map)# action forward
Device(config-access-map)# end
```

例：すべてのパケットをドロップするデフォルトアクション

次の例の VLAN マップでは、デフォルトですべてのパケット（IP および非 IP）がドロップされます。例2および例3のアクセスリスト **tcp-match** および **good-hosts** をこのマップと組み合わせると、次のようになります。

- すべての TCP パケットが転送されます。
- ホスト 0000.0c00.0111 および 0000.0c00.0211 からの MAC パケットが転送されます。
- その他のすべての IP パケットがドロップされます。
- その他のすべての MAC パケットがドロップされます。

```
Device> enable
Device# configure terminal
Device(config)# vlan access-map drop-all-default 10
Device(config-access-map)# match ip address tcp-match
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-all-default 20
Device(config-access-map)# match mac address good-hosts
Device(config-access-map)# action forward
Device(config-access-map)# end
```

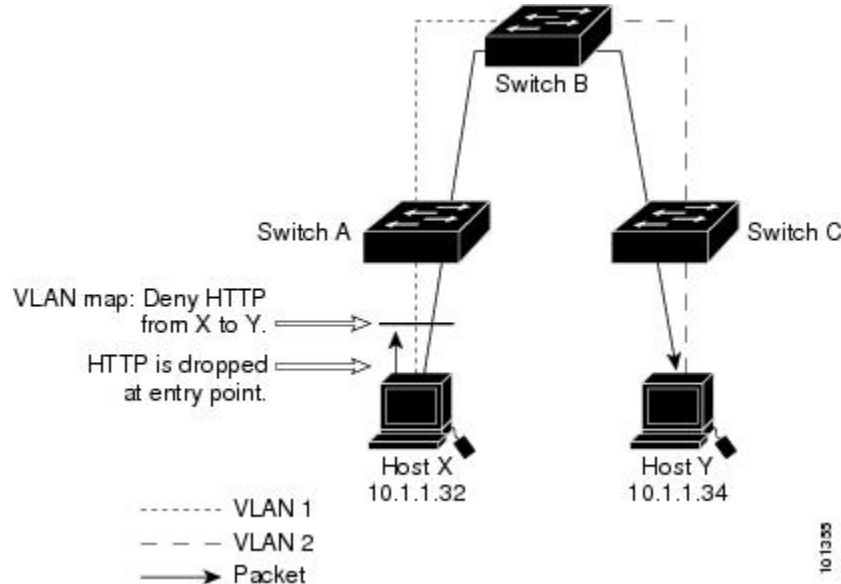
例：ネットワークでの VLAN マップの使用

例：ワイヤリングクローゼットの設定

図 18: ワイヤリングクローゼットの設定

ワイヤリングクローゼット構成では、ルーティングがスイッチ上で有効にされていない場合があります。ただし、この設定でも VLAN マップおよび QoS 分類 ACL はサポートされています。ホスト X およびホスト Y は異なる VLAN 内にあり、ワイヤリングクローゼットスイッチ A およびスイッチ C に接続されていると想定します。ホスト X からホスト Y へのトラフィックは、ルーティングが有効に設定されたレイヤ3スイッチであるスイッチ B によって最終的にルーティングされます。ホスト X からホスト Y へのトラフィックは、トラフィックのエント

リポイントであるスイッチ A でアクセスコントロールできます。



HTTP トラフィックをホスト X からホスト Y へスイッチングしない場合は、ホスト X (IP アドレス 10.1.1.32) からホスト Y (IP アドレス 10.1.1.34) に向かうすべての HTTP トラフィックがスイッチ A でドロップされ、スイッチ B にブリッジングされないように、スイッチ A の VLAN マップを設定できます。

最初に、HTTP ポート上ですべての TCP トラフィックを許可 (一致) する IP アクセスリスト *http* を定義します。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended http
Device(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Device(config-ext-nacl)# end
```

次に、*http* アクセスリストと一致するトラフィックがドロップされ、その他のすべての IP トラフィックが転送されるように、VLAN アクセスマップ *map2* を作成します。

```
Device> enable
Device# configure terminal
Device(config)# vlan access-map map2 10
Device(config-access-map)# match ip address http
Device(config-access-map)# action drop
Device(config-access-map)# exit
Device(config)# ip access-list extended match_all
Device(config-ext-nacl)# permit ip any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map2 20
Device(config-access-map)# match ip address match_all
Device(config-access-map)# action forward
Device(config-access-map)# end
```

次に、VLAN アクセスマップ *map2* を VLAN 1 に適用します。

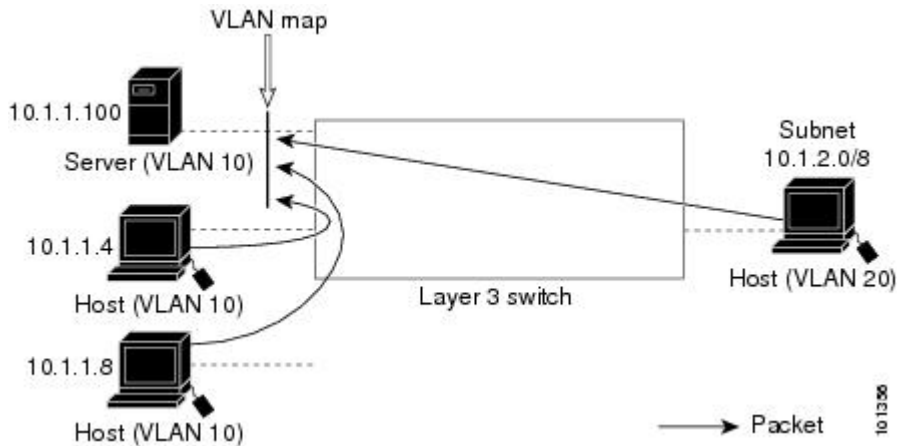
```
Device> enable
Device# configure terminal
Device(config)# vlan filter map2 vlan 1
Device(config)# end
```

例：別の VLAN にあるサーバーへのアクセスの制限

図 19:別の VLAN 上のサーバーへのアクセスの制限

別の VLAN にあるサーバーへのアクセスを制限できます。たとえば、VLAN 10 内のサーバー 10.1.1.100 では、次のホストへのアクセスを拒否する必要があります。

- VLAN 20 内のサブネット 10.1.2.0/8 にあるホストのアクセスを禁止します。
- VLAN 10 内のホスト 10.1.1.4 および 10.1.1.8 のアクセスを禁止します。



例：別の VLAN にあるサーバーへのアクセスの拒否

次に、サブネット 10.1.2.0.8 内のホスト、ホスト 10.1.1.4、およびホスト 10.1.1.8 のアクセスを拒否し、その他の IP トラフィックを許可する VLAN マップ SERVER1_ACL を作成して、別の VLAN 内のサーバーへのアクセスを拒否する例を示します。最後のステップでは、マップ SERVER1 を VLAN 10 に適用します。

正しいパケットと一致する IP ACL を定義します。

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended SERVER1_ACL
Device(config-ext-nacl)# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Device(config-ext-nacl)# permit ip host 10.1.1.4 host 10.1.1.100
Device(config-ext-nacl)# permit ip host 10.1.1.8 host 10.1.1.100
Device(config-ext-nacl)# end
```

SERVER1_ACL と一致する IP パケットをドロップして、この ACL と一致しない IP パケットを転送する ACL を使用して、VLAN マップを定義します。

```
Device> enable
Device# configure terminal
Device(config)# vlan access-map SERVER1_MAP
Device(config-access-map)# match ip address SERVER1_ACL
Device(config-access-map)# action drop
Device(config)# vlan access-map SERVER1_MAP 20
Device(config-access-map)# action forward
Device(config-access-map)# end
```

VLAN 10 に VLAN マップを適用します。

```
Device> enable
Device# configure terminal
Device(config)# vlan filter SERVER1_MAP vlan-list 10
Device(config)# end
```

IPv4 アクセスコントロールリストに関する追加情報

関連資料

関連項目	マニュアルタイトル
IPv6 ACL	『セキュリティ コンフィギュレーションガイド』の「IPv6 ACL」の章

シスコのテクニカルサポート

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

IPv4 アクセスコントロールリストの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	IPv4 アクセスコントロールリスト	この章では、ACL を使用して、スイッチのネットワークセキュリティを設定する方法について説明します。パケットフィルタリングは、ネットワークトラフィックを限定し、特定のユーザまたはデバイスによるネットワークの使用を制限するうえで役立ちます。ACL は、トラフィックをデバイスの通過時にフィルタリングし、パケットが指定されたインターフェイスを通過することを許可または拒否します。

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



第 18 章

IPv6 ACL

- [IPv6 ACL の制限 \(397 ページ\)](#)
- [IPv6 ACL の概要 \(398 ページ\)](#)
- [IPv6 ACL の設定方法 \(401 ページ\)](#)
- [IPv6 ACL のモニタリング \(409 ページ\)](#)
- [IPv6 ACL の設定例 \(409 ページ\)](#)
- [IPv6 ACL の機能履歴 \(411 ページ\)](#)

IPv6 ACL の制限

IPv6 がサポートするのは名前付き ACL だけです。IPv4 ACL では、番号制の標準 IP ACL および拡張 IP ACL、名前付き IP ACL、および MAC ACL を設定できます。

スイッチは Cisco IOS がサポートする IPv6 ACL の大部分をサポートしますが、一部例外もあります。

- スイッチは、**flowlabel**、**routing header**、および **undetermined-transport** というキーワードの照合をサポートしません。
- スイッチは再起 ACL (**reflect** キーワード) をサポートしません。
- スイッチは IPv6 フレームに MAC ベース ACL を適用しません。
- ACL を設定する場合、ACL に入力されるキーワードには、それがプラットフォームでサポートされるかどうかにかかわらず、制約事項はありません。ハードウェア転送が必要なインターフェイス (物理ポートまたは SVI) に ACL を適用する場合、スイッチはインターフェイスで ACL がサポートされるかどうか判別します。ACL がインターフェイスでサポートされていない場合、ACL は拒否されます。
- インターフェイスに適用される ACL に、サポートされないキーワードを持つアクセス コントロール エントリ (ACE) を追加しようとする場合、スイッチは現在インターフェイスに適用されている ACL に ACE が追加されるのを許可しません。
- プロトコルの TCAM をプログラムしないインターフェイスと、アンロードされた ACL にスケール ACL を適用すると、他のプロトコルのトラフィックの既存の通常移動に影響を

与える可能性があります。IPv6 および MAC アドレストラフィックにこの制限は適用されません。

- 存続可能時間 (TTL) 分類は、ACL ではサポートされていません。
- ダウンロード可能な ACL に重複するエントリが含まれている場合、エントリは自動的にマージされません。その結果、802.1Xセッション許可は失敗します。ダウンロード可能な ACL が、同じポートのポートベースのエントリや名前ベースのエントリなど、重複するエントリなしで最適化されていることを確認します。

IPv6 ACL の概要

ここでは、IPv6 ACL について説明します。

IPv6 ACL の概要

このトピックでは、IPv6 ACL の概要を示します。

アクセスコントロールリスト (ACL) とは、特定のインターフェイスへのアクセスを制限するために使用されるルールセットのことです。ACL は デバイスに設定され、管理インターフェイスおよび任意の動的インターフェイスに適用されます。

Web 認証用に事前認証 ACL を作成することもできます。このような ACL は、認証が完了するまでに特定のタイプのトラフィックを許可するために使用されます。

IPv6 ACL は、送信元、宛先、送信元ポート、宛先ポートなど、IPv4 ACL と同じオプションをサポートします。

サポートされる ACL

スイッチでは、トラフィックをフィルタリングするために、次に示す 3 種類の ACL がサポートされています。

- ポート ACL は、レイヤ 2 インターフェイスに入るトラフィックをアクセス コントロールします。IPv4 と MAC どちらのアクセスリストタイプのどの方向に対してでも、レイヤ 2 インターフェイスにポート ACL を適用できます。
- ルータ ACL は、VLAN 間でルーティングされたトラフィックのアクセスを制御し、レイヤ 3 インターフェイスで特定の方向 (インバウンドまたはアウトバウンド) に適用されません。
- VLAN ACL または VLAN マップはレイヤ 2 VLAN にのみ適用され、ブリッジされたトラフィックにのみ影響します。VLAN マップを使用すると、同じ VLAN 内のデバイス間で転送されるトラフィックをフィルタリングできます。VLAN マップは、IPv4 のレイヤ 3 アドレスに基づいてアクセスコントロールするように設定されています。イーサネット ACE を使用すると MAC アドレスにより、サポートされていないプロトコルがアクセスコントロールされます。VLAN マップを VLAN に適用すると、VLAN に入るすべてのパケット

(ルーテッドパケットまたはブリッジドパケット)がVLANマップと照合されます。パケットは、スイッチポートを介して、または、ルーティングされたパケットの場合、ルーテッドポートを介して、VLANに入ることができます。

ACLのタイプ

次のセクションではACLのタイプについて説明します。

ユーザー単位IPv6 ACL

ユーザーあたりのACLの場合、テキスト文字列としての完全なアクセスコントロールエントリ(ACE)がCisco Secure Access Control Server(Cisco Secure ACS)で設定されます。

フィルタID IPv6 ACL

filter-id ACLの場合、完全なACEおよび`acl name(filter-id)`がデバイスで設定され、`filter-id`のみが次に設定されます。Cisco Secure ACSで設定されます。

ダウンロード可能IPv6 ACL

ダウンロード可能ACL(dACL)の場合、完全なACEおよび`dacl`名はCisco Secure ACSのみで設定されます。

Cisco Secure ACSはその`ACCESS-Accept`属性で`dacl`名をデバイスに送信します。デバイスは`dacl`名を取得し、ACEのために`dACL`名を`ACCESS-request`属性を使用してCisco Secure ACSに送り返します。

スイッチスタックおよびIPv6 ACL

アクティブスイッチはIPv6 ACLをハードウェアでサポートし、IPv6 ACLをスタックメンバーに配信します。

スタンバイスイッチがアクティブスイッチを引き継ぐと、ACL設定がすべてのスタックメンバーに配信されます。メンバースイッチは、新しいアクティブスイッチによって配信された設定を同期し、不要なエントリを消去します。

ACLの修正、インターフェイスへの適用、またはインターフェイスからの解除が行われると、アクティブスイッチは変更内容をすべてのスタックメンバーに配信します。

ACL優先順位

VLANマップ、ポートACL、およびルータACLが同じスイッチに設定されている場合、入力トラフィックの場合のフィルタの優先順位は上からポートACL、VLANマップ、およびルータACLです。出力トラフィックの場合、フィルタの優先順位は、ルータACL、VLANマップ、ポートACLです。

次の例で、簡単な使用例を説明します。

- 入力ポート ACL と VLAN マップが両方とも適用されている場合に、ポート ACL が適用されたポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。その他のパケットは、VLAN マップによってフィルタリングされます。
- スイッチ仮想インターフェイス (SVI) に入力ルータ ACL および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。他のポートで受信した着信のルーティング IP パケットには、ルータ ACL のフィルタが適用されます。他のパケットはフィルタリングされません。
- SVI に出カルータ ACL および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。発信するルーティング IP パケットには、ルータ ACL のフィルタが適用されます。他のパケットはフィルタリングされません。
- SVI に VLAN マップ、入力ルータ ACL、および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL だけによってフィルタリングされます。他のポートで受信した着信のルーティング IP パケットには、VLAN マップおよびルータ ACL のフィルタが適用されます。他のパケットには、VLAN マップのフィルタだけが適用されます。
- SVI に VLAN マップ、出カルータ ACL、および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL だけによってフィルタリングされます。発信するルーティング IP パケットには、VLAN マップおよびルータ ACL のフィルタが適用されます。他のパケットには、VLAN マップのフィルタだけが適用されます。

VLAN マップ

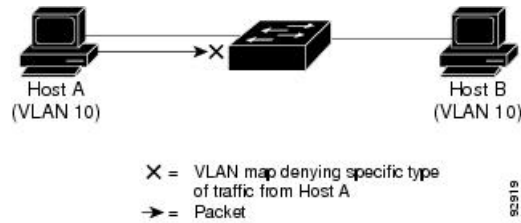
VLAN ACL または VLAN マップは、VLAN 内のネットワークトラフィックを制御するために使用されます。スイッチまたはスイッチ スタックの VLAN 内でブリッジされるすべてのパケットに VLAN マップを適用できます。VACL は、セキュリティ パケット フィルタリング および特定の物理インターフェイスへのトラフィックのリダイレクトだけを目的としたものです。VACL は方向（入力または出力）で定義されることはありません。

すべての非 IP プロトコルは、MAC VLAN マップを使用して、MAC アドレスおよび Ethertype によってアクセス コントロールされます（IP トラフィックは、MAC VLAN マップではアクセス制御されません）。VLAN マップはスイッチを通過するパケットにだけ適用できます。ハブ上またはこのスイッチに接続された別のスイッチ上のホスト間のトラフィックには、VLAN マップを適用できません。

VLAN マップを使用すると、マップに指定されたアクションに基づいてパケットの転送が許可または拒否されます。

図 20: VLAN マップによるトラフィックの制御

次の図に、VLAN マップを適用して、特定のトラフィック タイプを VLAN 10 のホスト A から転送できないように設定する例を示します。各 VLAN には、VLAN マップを 1 つだけ適用で



きます。

他の機能およびスイッチとの相互作用

- IPv6 ルータ ACL がパケットを拒否するよう設定されている場合、パケットはルーティングされません。パケットのコピーがインターネット制御メッセージプロトコル (ICMP) キューに送信され、フレームに ICMP 到達不能メッセージが生成されます。
- ブリッジドフレームがポート ACL によってドロップされる場合、このフレームはブリッジングされません。
- IPv4 ACL および IPv6 ACL の両方を 1 つのスイッチまたはスイッチ スタックに作成したり、同一インターフェイスに適用できます。各 ACL には一意の名前が必要です。設定済みの名前を使用しようとすると、エラーメッセージが表示されます。

IPv4 ACL と IPv6 ACL の作成、および同一のレイヤ 2 インターフェイスまたはレイヤ 3 インターフェイスへの IPv4 ACL または IPv6 ACL の適用には、異なるコマンドを使用します。ACL を付加するのに誤ったコマンドを使用すると（例えば、IPv6 ACL の付加に IPv4 コマンドを使用するなど）、エラーメッセージが表示されます。

- MAC ACL を使用して、IPv6 フレームをフィルタリングできません。MAC ACL は非 IP フレームだけをフィルタリングできます。
- ハードウェアメモリに空きがない場合、パケットはインターフェイスでドロップされ、アンロードのエラーメッセージが記録されます。

ハードウェアメモリが満杯の場合、設定済みの ACL を追加すると、パケットは CPU に転送され、ACL はソフトウェアで適用されます。ハードウェアが一杯になると、ACL がアンロードされたことを示すメッセージがコンソールに出力され、パケットはインターフェイスでドロップされます。

IPv6 ACL の設定方法

ここでは、IPv6 ACL の設定方法に関する情報を示します。

IPv6 ACL のデフォルト設定

デフォルトの IPv6 ACL 設定は次のとおりです。

```
Device# show access-lists preauth_ipv6_acl

IPv6 access list preauth_ipv6_acl (per-user)
permit udp any any eq domain sequence 10
permit tcp any any eq domain sequence 20
permit icmp any any nd-ns sequence 30
permit icmp any any nd-na sequence 40
permit icmp any any router-solicitation sequence 50
permit icmp any any router-advertisement sequence 60
permit icmp any any redirect sequence 70
permit udp any eq 547 any eq 546 sequence 80
permit udp any eq 546 any eq 547 sequence 90
deny ipv6 any any sequence 100
```

IPv6 ACL の設定

IPv6 トラフィックをフィルタリングするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 access-list <i>{list-name log-update threshold role-based list-name}</i> 例： Device(config)# ipv6 access-list example_acl_list	IPv6 ACL 名を定義し、IPv6 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	{deny permit} protocol <i>{source-ipv6-prefix/ prefix-length [any threshold] host source-ipv6-address}</i> [operator [port-number]] <i>{destination-ipv6-prefix/ prefix-length any host destination-ipv6-address}</i> [operator [port-number]][dscp value] [fragments] [log] [log-input][sequence value] [time-range name] 例： Device(config-ipv6-acl)# permit tcp 2001:DB8:0300:0201::/32 eq telnet any	IPv6 ACL の許可条件または拒否条件を指定します。 <ul style="list-style-type: none">protocol には、IP の名前または番号を入力します。 ahp、esp、icmp、ipv6、pcp、step、tcp、udp または IPv6 プロトコル番号を表す 0 ~ 255 の整数を使用できます。<i>source-ipv6-prefix/prefix-length</i> または <i>destination-ipv6-prefix/ prefix-length</i> は、拒否条件または許可条件を設定する送信元または宛先 IPv6 ネットワークあるいはネットワーク クラス

コマンドまたはアクション	目的
	<p>で、コロン区切りの 16 ビット値を使用した 16 進形式で指定します (RFC 2373 を参照)。</p> <ul style="list-style-type: none"> • IPv6 プレフィックス <code>::/0</code> の短縮形として、any を入力します。 • host <i>source-ipv6-address</i> または <i>destination-ipv6-address</i> には、拒否条件または許可条件を設定する送信元または宛先 IPv6 ホストアドレスを入力します。アドレスはコロン区切りの 16 ビット値を使用した 16 進形式で指定します。 • (任意) operator には、指定のプロトコルの送信元ポートまたは宛先ポートを比較するオペランドを指定します。オペランドには、lt (より小さい)、gt (より大きい)、eq (等しい)、neq (等しくない)、および range (包含範囲) があります。 <p><i>source-ipv6-prefix/prefix-length</i> 引数のあとの operator は、送信元ポートに一致する必要があります。<i>destination-ipv6-prefix/prefix-length</i> 引数のあとの operator は、宛先ポートに一致する必要があります。</p> <ul style="list-style-type: none"> • (任意) port-number は、0 ~ 65535 の 10 進数または TCP あるいは UDP ポートの名前です。TCP ポート名を使用できるのは、TCP のフィルタリング時だけです。UDP ポート名を使用できるのは、UDP のフィルタリング時だけです。 • (任意) dscp value を入力して、各 IPv6 パケットヘッダーの Traffic Class フィールド内のトラフィッククラス値と DiffServ コードポイント値を照合します。指定できる範囲は 0 ~ 63 です。 • (任意) fragments を入力して、先頭ではないフラグメントを確認します。このキーワードが表示されるのは、プロトコルが ipv6 の場合だけです。 • (任意) log を指定すると、エントリと一致するパケットに関するログメッセージがコンソールに送信されます。log-input を指定すると、ログエントリに入力インターフェイスが追加され

	コマンドまたはアクション	目的
		<p>ます。ロギングはルータ ACL でだけサポートされます。</p> <ul style="list-style-type: none"> • (任意) sequence value を入力して、アクセスリストステートメントのシーケンス番号を指定します。指定できる範囲は1～4,294,967,295です。 • (任意) time-range name を入力して、拒否または許可ステートメントに適用される時間の範囲を指定します。
ステップ 5	<pre>{deny permit} tcp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [ack] [dscp value] [established] [fin] [log] [log-input] [neq {port protocol}] [psh] [range {port protocol}] [rst] [sequence value] [syn] [time-range name] [urg]</pre> <p>例 :</p> <pre>Device(config-ipv6-acl)# deny tcp host 2001:DB8:1::1 any log-input</pre>	<p>IPv6 ACL の許可条件または拒否条件を指定します。TCP の場合は tcp を入力します。パラメータはステップ 3a で説明されているパラメータと同じですが、次に示すオプションのパラメータが追加されています。</p> <ul style="list-style-type: none"> • ack : 確認応答 (ACK) ビットセット。 • established : 確立された接続。TCP データグラムに ACK または RST ビットが設定されている場合、照合が行われます。 • fin : 終了ビットセット。送信者からのデータはそれ以上ありません。 • neq {port protocol} : 所定のポート番号上にならないパケットだけを照合します。 • psh : プッシュ機能ビットセット • range {port protocol} : ポート番号の範囲内のパケットだけを照合します。 • rst : リセットビットセット • syn : 同期ビットセット • urg : 緊急ポインタビットセット
ステップ 6	<pre>end</pre> <p>例 :</p> <pre>Device(config-ipv6-acl)# end</pre>	<p>IPv6 アクセス リスト コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。</p>
ステップ 7	<pre>show ipv6 access-list</pre> <p>例 :</p>	<p>IPv6 ACL が正しく設定されていることを確認します。</p>

	コマンドまたはアクション	目的
	Device# show ipv6 access-list	

インターフェイスへの IPv6 ACL の付加

レイヤ 3 インターフェイスで発信または着信トラフィックに ACL を、あるいはレイヤ 2 インターフェイスで着信トラフィックに を適用できます。レイヤ 3 インターフェイスで着信トラフィックにだけ ACL を適用できます。

インターフェイスへのアクセスを制御するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **no switchport**
5. **ipv6 address ipv6-address**
6. **ipv6 traffic-filter access-list-name {in | out}**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例： Device(config)# interface gigabitethernet 1/0/1	アクセスリストを適用するレイヤ 2 インターフェイス（ポート ACL 用）またはレイヤ 3 インターフェイス（ルータ ACL 用）を指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	no switchport 例： Device(config-if)# no switchport	インターフェイスをルーテッドインターフェイスの状態に戻して、レイヤ 2 の詳細設定をすべて削除します。
ステップ 5	ipv6 address ipv6-address 例： Device(config-if)# ipv6 address 2001:DB8::1	レイヤ 3 インターフェイス（ルータ ACL 用）で 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 6	ipv6 traffic-filter <i>access-list-name</i> {in out} 例： Device(config-if)# ipv6 traffic-filter acl1 in	インターフェイスの着信トラフィックまたは発信トラフィックにアクセスリストを適用します。
ステップ 7	end 例： Device(config-ipv6-acl)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

VLAN マップの設定

VLAN マップを作成して、1つまたは複数の VLAN に適用するには、次のステップを実行します。

始める前に

VLAN に適用する IPv6 ACL を作成します。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan access-map** *name* [*number*]
4. **match** {ip | ipv6 | mac} **address** {*name* | *number*} [*name* | *number*]
5. IP パケットまたは非 IP パケットを（既知の 1 MAC アドレスのみを使って）指定し、1つ以上の ACL とそのパケットを照合するには、次のコマンドのいずれかを入力します。
 - **action** { **forward** }
Device(config-access-map)# action forward
 - **action** { **drop** }
Device(config-access-map)# action drop
6. **vlan filter** *mapname* **vlan-list** *list*
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan access-map name [number] 例 : Device(config)# vlan access-map map_1 20	<p>VLAN マップを作成して、VLAN アクセスマップ コマンド モードを開始します。</p> <p>VLAN マップには、名前または（オプションで）番号を指定できます。番号は、マップ内のエントリのシーケンス番号です。</p> <p>同じ名前の VLAN マップを作成すると、10 ずつ増加する番号が順に割り当てられます。マップを変更または削除するときは、該当するマップエントリの番号を入力できます。</p> <p>VLAN マップでは、特定の permit または deny キーワードを使用しません。VLAN マップを使用してパケットを拒否するには、パケットを照合する ACL を作成して、アクションをドロップに設定します。ACL 内の permit は、一致するという意味です。ACL 内の deny は、一致しないという意味です。</p>
ステップ 4	match {ip ipv6 mac} address {name number} [name number] 例 : Device(config-access-map)# match ipv6 address ip_net	<p>パケットを 1 つまたは複数のアクセスリストと照合します。パケットの照合は、対応するプロトコルタイプのアクセスリストに対してだけ行われます。IP パケットは、IP アクセスリストに対して照合されます。非 IP パケットは、名前付き MAC アクセスリストに対してだけ照合されます。</p> <p>(注) パケットタイプ (IP または MAC) に対する match 句が VLAN マップに設定されている場合で、そのマップアクションがドロップの場合は、そのタイプに一致するすべてのパケットがドロップされます。match 句が VLAN マップになく、設定されているアクションがドロップの場合は、すべての IP およびレイヤ 2 パケットがドロップされます。</p>
ステップ 5	IP パケットまたは非 IP パケットを（既知の 1 MAC アドレスのみを使って）指定し、1 つ以上の ACL とそのパケットを照合するには、次のコマンドのいずれかを入力します。 <ul style="list-style-type: none"> • action { forward } 	マップエントリに対するアクションを設定します。

	コマンドまたはアクション	目的
	Device(config-access-map)# action forward • action { drop} Device(config-access-map)# action drop	
ステップ 6	vlan filter mapname vlan-list list 例： Device(config)# vlan filter map 1 vlan-list 20-22	VLAN マップを 1 つまたは複数の VLAN に適用します。 list には単一の VLAN ID (22)、連続した範囲 (10 ~ 22)、または VLAN ID のストリング (12、22、30) を指定できます。カンマやハイフンの前後にスペースを挿入することもできます。
ステップ 7	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

VLAN への VLAN マップの適用

VLAN マップを 1 つまたは複数の VLAN に適用するには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan filter mapname vlan-list list**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan filter mapname vlan-list list 例： Device(config)# vlan filter map 1 vlan-list 20-22	VLAN マップを 1 つまたは複数の VLAN に適用します。 list には単一の VLAN ID (22)、連続した範囲 (10 ~ 22)、または VLAN ID のストリング (12、22、

	コマンドまたはアクション	目的
		30) を指定できます。カンマやハイフンの前後にスペースを挿入することもできます。
ステップ 4	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

IPv6 ACL のモニタリング

次の表に示された 1 つまたは複数の特権 EXEC コマンドを使用して、設定済みのすべてのアクセスリスト、すべての IPv6 アクセスリスト、または特定のアクセスリストに関する情報を表示できます。

表 24: *show ACL* コマンド

コマンド	目的
show access-lists	スイッチに設定されたすべてのアクセス リストを表示します。
show ipv6 access-list [<i>access-list-name</i>]	設定済みのすべての IPv6 アクセス リストまたは名前前で指定されたアクセス リストを表示します。
show vlan access-map [<i>map-name</i>]	VLAN アクセス マップ設定を表示します。
show vlan filter [access-map <i>access-map</i> vlan <i>vlan-id</i>]	VACL と VLAN 間のマッピングを表示します。

IPv6 ACL の設定例

ここでは、IPv6 ACL の設定例を示します。

例：IPv6 ACL の作成

この例では、IPv6-ACL という名前の IPv6 アクセスリストを設定します。リスト内の最初の拒否エントリは、宛先 TCP ポート番号が 5000 より大きいパケットをすべて拒否します。2 番目の拒否エントリは、送信元 UDP ポート番号が 5000 未満のパケットを拒否します。また、この 2 番目の拒否エントリは、すべての一致をコンソールに表示します。リスト内の最初の許可エントリは、すべての ICMP パケットを許可します。リスト内の 2 番目の許可エントリは、その他のすべてのトラフィックを許可します。暗黙の全否定の条件が各 IPv6 アクセスリストの末尾にあるため、2 番目の許可エントリは必要です。



(注) ログギングは、レイヤ3 インターフェイスでのみサポートされます。

```
Device> enable
Device(config)# ipv6 access-list IPv6_ACL
Device(config-ipv6-acl)# deny tcp any any gt 5000
Device (config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Device(config-ipv6-acl)# permit icmp any any
Device(config-ipv6-acl)# permit any any
Device(config-ipv6-acl)# end
```

例：IPv6 ACL の表示

次に、**show access-lists** 特権 EXEC コマンドの出力例を示します。スイッチに設定されているすべてのアクセス リストが表示されます。

```
Device# show access-lists

Extended IP access list hello
10 permit ip any any
IPv6 access list ipv6
permit ipv6 any any sequence 10
```

次に、**show ipv6 access-lists** 特権 EXEC コマンドの出力例を示します。スイッチに設定されている IPv6 アクセス リストだけが表示されます。

```
Device# show ipv6 access-list

IPv6 access list inbound
permit tcp any any eq bgp (8 matches) sequence 10
permit tcp any any eq telnet (15 matches) sequence 20
permit udp any any sequence 30
IPv6 access list outbound
deny udp any any sequence 10
deny tcp any any eq telnet sequence 20
```

例：VLAN アクセスマップ設定の表示

次に、**show vlan access-map** 特権 EXEC コマンドの出力例を示します。

```
Device# show vlan access-map

Vlan access-map "m1" 10
  Match clauses:
    ipv6 address: ip2
  Action: drop
```

次に、**show ipv6 access-lists** 特権 EXEC コマンドの出力例を示します。スイッチに設定されている IPv6 アクセス リストだけが表示されます。

```
Device# show ipv6 access-list

IPv6 access list inbound
permit tcp any any eq bgp (8 matches) sequence 10
permit tcp any any eq telnet (15 matches) sequence 20
permit udp any any sequence 30
```



```
IPv6 access list outbound
deny udp any any sequence 10
deny tcp any any eq telnet sequence 20
```

IPv6 ACL の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	IPv6 ACL	

リリース	機能	機能情報
		IPv6 ACL を作成して、インターフェイスに適用することによって、IPv6 トラフィックをフィルタリングできます。これは、IPv4 の名前付き ACL を作成し、適用する方法と類似しています。

リリース	機能	機能情報
		レイヤ3管理トラフィックをフィルタリングするために、入力ルータ ACL を作成し、適用することもできます。
Cisco IOS XE Gibraltar 16.11.1	IPv6 ダウンロード可能 ALC	IPv6 dACL がサポートされます。

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



第 19 章

IP ソース ガードの設定

- [IP ソース ガードの概要 \(415 ページ\)](#)
- [IP ソース ガードの設定方法 \(418 ページ\)](#)
- [IP ソース ガードのモニタリング \(421 ページ\)](#)
- [IP ソース ガードの機能の履歴 \(421 ページ\)](#)

IP ソース ガードの概要

IP ソース ガード

IP ソースガード (IPSG) を使用して、ホストがネイバーの IP アドレスを使用する場合にトラフィック攻撃を防ぐことができ、また、信頼できないインターフェイスで DHCP スヌーピングが有効な場合に、IP ソースガードを有効にできます。

インターフェイス上で IPSG をイネーブルにすると、スイッチは、DHCP スヌーピングにより許可された DHCP パケットを除き、このインターフェイスで受信したすべての IP トラフィックをブロックします。

スイッチは IP アドレスをポートにバインドするためにハードウェアの発信元 IP 検索テーブルを使用します。IP および MAC のフィルタリングでは、送信元 IP 検索および送信元 MAC 検索の組み合わせが使用されます。バインディングテーブル内の送信元 IP アドレスを使用する IP トラフィックは許可され、他のすべてのトラフィックは拒否されます。

IP ソース バインディング テーブルには、DHCP スヌーピングで学習されたバインディング、または手動で設定されたバインディング (スタティック IP 送信元バインディング) があります。このテーブルのエントリには IP アドレスと、関連 MAC アドレス、および関連 VLAN 番号があります。スイッチは、IP ソース ガードがイネーブルにされている場合だけ、IP ソース バインディング テーブルを使用します。

IPSG は、アクセス ポートおよびトランク ポートを含むレイヤ 2 ポートだけでサポートされます。送信元 IP アドレスと送信元 IP および MAC アドレス フィルタリングで IPSG を設定できます。

スタティック ホスト用 IP ソース ガード



Note スタティックホストの IPSG は、アップリンクポートまたはトランクポートでは使用しないでください。

スタティック ホスト用 IPSG は、IPSG の機能を DHCP ではない、スタティックな環境に拡張するものです。これまでの IPSG は、DHCP スヌーピングにより作成されたエントリを使用して、スイッチに接続されたホストを検証していました。ホストから受信したトラフィックのうち、有効な DHCP を持たないものはすべてドロップされます。このセキュリティ機能によって、ルーティングされないレイヤ 2 インターフェイス上の IP トラフィックが制限されます。この機能は、DHCP スヌーピング バインディング データベース、および手動で設定された IP ソースバインディングに基づいてトラフィックをフィルタリングします。前バージョンの IPSG では、IPSG を動作させるために DHCP 環境が必要でした。

スタティック ホスト用 IPSG では、DHCP なしで IPSG を動作させることができます。スタティック ホスト用 IPSG は、ポート ACL をインストールするために IP デバイストラッキング テーブルエントリに依存しています。このスイッチは、指定されたポートで有効なホストのリストを維持するために、ARP リクエスト、またはその他の IP パケットに基づいてスタティック エントリを作成します。また、指定されたポートにトラフィックを送信できるホストの数を指定することもできます。これはレイヤ 3 でのポートセキュリティと同じです。

スタティック ホスト用 IPSG はダイナミック ホストもサポートしています。ダイナミック ホストが、IP DHCP スヌーピング テーブルに存在する DHCP が割り当てられた IP アドレスを受信すると、IP デバイストラッキング テーブルは同じエントリを学習します。スタック化環境では、アクティブスイッチのフェールオーバーが発生すると、メンバポートに接続されたスタティックホストの IP ソースガードエントリは、そのまま残ります。show device-tracking database EXEC コマンドを入力すると、IP デバイストラッキング テーブルには、これらのエントリが ACTIVE であると表示されます。



Note 複数のネットワーク インターフェイスを持つ IP ホストの一部は、ネットワーク インターフェイスに無効なパケットを注入することができます。この無効なパケットには、ソースアドレスとして、別のホスト ネットワーク インターフェイスの IP アドレス、または MAC アドレスが含まれます。この無効パケットは、スタティック ホスト用 IPSG がホストに接続され、無効な IP アドレス バインディングまたは MAC アドレス バインディングが学習されて、有効なバインディングが拒否される原因となります。ホストによる無効なパケットの注入を回避する方法については、対応するオペレーティングシステムとネットワーク インターフェイスのベンダーにお問い合わせください。

最初、スタティック ホスト用 IPSG は ACL ベースのスヌーピング メカニズムを通じて、動的に IP バインディング、または MAC バインディングを学習します。IP バインディング、または MAC バインディングは、ARP パケット、および IP パケットにより、スタティック ホストから学習されます。これらはデバイストラッキング データベースに保存されます。指定されたポートで動的に学習、または静的に設定された IP アドレスの数が最大値に達した場合、新

しい IP アドレスを持つパケットはすべて、ハードウェアによりドロップされます。何らかの理由で移動された、またはなくなったホストを解決するために、スタティック ホスト用 IPSG は IP デバイス トラッキングを活用して、動的に学習した IP アドレス バインディングをエーミングアウトします。この機能は、DHCP スヌーピングとともに使用できます。複数バインディングは、DHCP ホストとスタティック ホストの両方に接続されたポートに確立されます。たとえば、バインディングは、デバイス トラッキング データベースと DHCP スヌーピング バインディング データベースの両方に保存されます。

IP ソース ガードの設定時の注意事項

- スタティック IP バインディングは、非ルーテッド ポートだけで設定できます。ルーテッド インターフェイスで **ip source binding mac-address vlan vlan-id ip-address interface interface-id** グローバル コンフィギュレーション コマンドを入力すると、次のエラーメッセージが表示されます。

```
Static IP source binding can only be configured on switch port.
```

- 送信元 IP フィルタリング機能を持つ IP ソース ガードがインターフェイスでイネーブルにされている場合、このインターフェイスのアクセス VLAN で、DHCP スヌーピングをイネーブルにしておく必要があります。
- 複数の VLAN を持つトランク インターフェイス上で IP ソース ガードをイネーブルにし、これらすべての VLAN で DHCP スヌーピングをイネーブルにした場合、すべての VLAN に、送信元 IP アドレス フィルタが適用されます。



Note IP ソース ガードがイネーブルにされているときに、トランク インターフェイスの VLAN 上で DHCP スヌーピングをイネーブル、またはディセーブルにした場合、スイッチは適切にトラフィックをフィルタリングできない可能性があります。

- この機能は、802.1x ポートベースの認証がイネーブルにされている場合にイネーブルにできます。
- IP ソース ガード スマート ロギングを設定すると、指定されたアドレスや DHCP によって学習されたアドレス以外の送信元アドレスを持つパケットは拒否され、そのパケットの内容が NetFlow 収集装置に送信されます。この機能を設定する場合は、スマート ロギングがグローバルにイネーブルになっていることを確認してください。
- スイッチスタックでは、IP ソースガードがスタック メンバ インターフェイスに設定されていて、**no switch stack-member-number provision** グローバル コンフィギュレーション コマンドの入力によりそのスイッチの設定を削除した場合、インターフェイススタティック バインディングはバインディングテーブルから削除されますが、実行コンフィギュレーションからは削除されません。**switch stack-member-number provision** コマンドを入力することによって、スイッチを再度プロビジョニングした場合、バインディングは復元されます。

実行コンフィギュレーションからバインディングを削除するには、**no switch provision** コマンドを入力する前に IP ソースガードを無効化する必要があります。インターフェイスがバインディングテーブルから削除される間にスイッチがリロードされると、設定も削除されます。

IP ソース ガードの設定方法

IP ソース ガードのイネーブル化

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **ip verify source [mac-check]**
5. **exit**
6. **ip source binding mac-address vlan vlan-id ip-address interface interface-id**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id Example: Device(config)# interface gigabitethernet 1/0/1	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	ip verify source [mac-check] Example: Device(config-if)# ip verify source	送信元 IP アドレス フィルタリングによる IP ソースガードを有効にします。 (任意) mac-check : 送信元 IP アドレスによる IP ソースガードおよび MAC アドレスフィルタリングを有効にします。

	Command or Action	Purpose
ステップ 5	exit Example: Device(config-if) # exit	インターフェイス コンフィギュレーション モードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 6	ip source binding mac-address vlan vlan-id ip-address interface interface-id Example: Device(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet1/0/1	スタティック IP ソース バインディングを追加します。 スタティックバインディングごとにこのコマンドを入力します。
ステップ 7	end Example: Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

レイヤ2アクセスポートでのスタティックホスト用IPソースガードの設定

スタティックホスト用 IPSG を動作させるには、**ip device tracking maximum limit-number** インターフェイス コンフィギュレーション コマンドをグローバルに設定する必要があります。このコマンドをポートに対して実行したが、IP デバイス トラッキングをグローバルに有効にしていない、または **ip device tracking maximum** をそのインターフェイスに対して設定していない場合は、スタティックホストの IPSG によって、そのインターフェイスからの IP トラフィックはすべて拒否されます。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip device tracking**
4. **interface interface-id**
5. **switchport mode access**
6. **switchport access vlan vlan-id**
7. **ip verify source[tracking] [mac-check]**
8. **ip device tracking maximum number**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	enable Example: Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。

	Command or Action	Purpose
ステップ 2	configure terminal Example: Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip device tracking Example: Device(config)# ip device tracking	IP ホスト テーブルをオンにし、IP デバイス トラッキングをグローバルに有効にします。
ステップ 4	interface interface-id Example: Device(config)# interface gigabitethernet 1/0/1	インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	switchport mode access Example: Device(config-if)# switchport mode access	アクセスとしてポートを設定します。
ステップ 6	switchport access vlan vlan-id Example: Device(config-if)# switchport access vlan 10	このポートに VLAN を設定します。
ステップ 7	ip verify source[tracking] [mac-check] Example: Device(config-if)# ip verify source tracking mac-check	送信元 IP アドレス フィルタリングによる IP ソースガードを有効にします。 (任意) tracking : スタティックホスト用 IP ソースガードを有効にします。 (任意) mac-check : MAC アドレス フィルタリングを有効にします。 ip verify source tracking mac-check コマンドは、MAC アドレス フィルタリングのあるスタティック ホストに対して IP ソース ガードを有効にします。
ステップ 8	ip device tracking maximum number Example: Device(config-if)# ip device tracking maximum 8	そのポートで、IP デバイス トラッキング テーブルにより許可されるスタティック IP 数の上限を設定します。指定できる範囲は 1 ~ 10 です。最大値は 10 です。 Note ip device tracking maximum limit-number インターフェイス コンフィギュレーション コマンドを設定する必要があります。
ステップ 9	end Example: Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

IP ソース ガードのモニタリング

表 25: 特権 EXEC 表示コマンド

コマンド	目的
show ip verify source [interface <i>interface-id</i>]	スイッチ上または特定のインターフェイス上の IP ソース ガードの設定を表示します。
show ip device tracking { all interface <i>interface-id</i> ip <i>ip-address</i> mac <i>mac-address</i> }	IP デバイス トラッキング テーブル内のエントリに関する情報を表示します。

表 26: インターフェイス コンフィギュレーションコマンド

コマンド	目的
ip verify source tracking	データ ソースを確認します。

IP ソース ガードの機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	IP ソース ガード	ネイバーの IP アドレスを使用する場合に、トラフィック攻撃を防ぐために IP ソース ガードを使用でき、そして信頼できないインターフェイスで DHCP スヌーピングがイネーブルの場合に、IP アドレスを使用しようとすると、IP ソース ガードをイネーブルにできません。

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



第 20 章

ダイナミック ARP インспекションの設定

- [ダイナミック ARP インспекションの制約事項 \(423 ページ\)](#)
- [ダイナミック ARP インспекションに関する情報 \(425 ページ\)](#)
- [ダイナミック ARP インспекションの設定方法 \(430 ページ\)](#)
- [DAI のモニタリング \(439 ページ\)](#)
- [DAI の設定の確認 \(439 ページ\)](#)
- [ダイナミック ARP インспекションの機能履歴 \(440 ページ\)](#)

ダイナミック ARP インспекションの制約事項

ここでは、スイッチにダイナミック Address Resolution Protocol (ARP) インспекションを設定するときの制約事項および注意事項を示します。

- ダイナミック ARP インспекションは入力セキュリティ機能です。出力チェックはまったく行いません。
- ダイナミック ARP インспекションは、ダイナミック ARP インспекションをサポートしていないスイッチ、またはこの機能がイネーブルにされていないスイッチに接続されるホストに対しては有効ではありません。中間者攻撃は単一のレイヤ2ブロードキャストドメインに制限されているため、チェックされないドメインと、ダイナミック ARP インспекションによりチェックされるドメインは区別します。このアクションは、ダイナミック ARP インспекションのためにイネーブルにされているドメインでホストの ARP キャッシュを保護します。
- 着信 ARP 要求、および ARP 応答で IP/MAC アドレス バインディングを検証するために、ダイナミック ARP インспекション DHCP スヌーピング バインディング データベースのエントリに依存します。IP アドレスがダイナミックに割り当てられた ARP パケットを許可する際は、DHCP スヌーピングをイネーブルにしてください。

DHCP スヌーピングをディセーブルにしている場合、または DHCP 以外の環境では、ARP ACL を使用してパケットの許可または拒否を行います。

- ダイナミック ARP インспекションは、アクセスポート、トランクポート、および EtherChannel ポートでサポートされます。



(注) RSPAN VLAN では、ダイナミック ARP インспекションをイネーブルにしないでください。RSPAN VLAN でダイナミック ARP インспекションをイネーブルにすると、ダイナミック ARP インспекション パケットが RSPAN 宛先ポートに届かない可能性があります。

- 物理ポートを EtherChannel ポート チャンネルに結合するには、この物理ポートとチャンネルポートの信頼状態が一致する必要があります。そうでない物理ポートは、ポートチャンネル内で中断状態のままとなります。ポートチャンネルは、チャンネルと結合された最初の物理ポートの信頼状態を継承します。したがって、最初の物理ポートの信頼状態は、チャンネルの信頼状態と一致する必要はありません。

逆に、ポートチャンネルで信頼状態を変更すると、スイッチは、チャンネルを構成するすべての物理ポートで新しい信頼状態を設定します。

- レート制限は、スイッチ スタックの各スイッチで別々に算出されます。クロススタック EtherChannel の場合、これは実際のレート制限が設定値よりも高い可能性があることを意味します。たとえば、レート制限が 30 pps に設定された EtherChannel で、スイッチ 1 に 1 つのポート、およびスイッチ 2 に 1 つのポートがある場合、EtherChannel が errdisable にならずに、各ポートは 29 pps でパケットを受信できます。
- ポートチャンネルの動作レートは、チャンネル内のすべての物理ポートによる累積値です。たとえば、ポートチャンネルの ARP レート制限を 400 pps に設定すると、このチャンネルに結合されたすべてのインターフェイスは、合計で 400 pps を受信します。EtherChannel ポートで受信される ARP パケットのレートは、すべてのチャンネル メンバーからの受信パケットレートの合計となります。EtherChannel ポートのレート制限は、各チャンネル ポート メンバーが受信する ARP パケットのレートを確認してから設定してください。

物理ポートで受信されるパケットのレートは、物理ポートの設定ではなく、ポートチャンネルの設定に照合して検査されます。ポートチャンネルのレート制限設定は、物理ポートの設定には依存しません。

EtherChannel が、設定したレートより多くの ARP パケットを受信すると、このチャンネル（すべての物理ポートを含む）は errdisable ステートとなります。

- 着信トランクポートでは、ARP パケットを必ずレート制限してください。トランクポートの集約を反映し、複数のダイナミック ARP インспекションがイネーブルにされた VLAN にわたってパケットを処理するために、トランクポートのレートをより高く設定します。また、**ip arp inspection limit none** インターフェイス コンフィギュレーション コマンドを使用して、レートを無制限に設定することもできます。1 つの VLAN に高いレート制限値を設定すると、ソフトウェアによってこのポートが errdisable ステートにされた場合に、他の VLAN への DoS 攻撃を招く可能性があります。

- スイッチで、ダイナミック ARP インспекションをイネーブルにすると、ARP トラフィックをポリシングするように設定されたポリサーの有効性は失われます。この結果、すべての ARP トラフィックは CPU に送信されます。

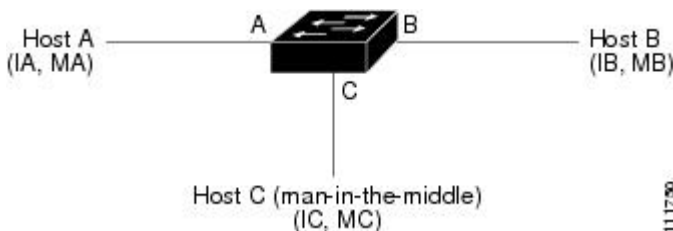
ダイナミック ARP インспекションに関する情報

ダイナミック ARP インспекションの概要

ARP では、IP アドレスを MAC アドレスにマッピングすることで、レイヤ 2 ブロードキャストドメイン内の IP 通信を実現します。たとえば、ホスト B はホスト A に情報を送信する必要がありますが、ARP キャッシュにホスト A の MAC アドレスを持っていないとします。ホスト B は、ホスト A の IP アドレスと関連付けられた MAC アドレスを取得するために、このブロードキャストドメインにあるホストすべてに対してブロードキャストメッセージを生成します。このブロードキャストドメイン内のホストはすべて ARP 要求を受信し、ホスト A は MAC アドレスで応答します。しかし、ARP は、ARP 要求を受信されなかった場合でも、ホストからの余分な応答を許可するため、ARP スプーフィング攻撃や ARP キャッシュのポイズニングが発生することがあります。攻撃が開始されると、攻撃を受けたデバイスからのすべてのトラフィックは、攻撃者のコンピュータを経由してルータ、スイッチ、またはホストに送信されるようになります。

悪意のあるユーザーは、サブネットに接続されているシステムの ARP キャッシュをポイズニングし、このサブネット上の他のホストを目的とするトラフィックを代行受信することにより、レイヤ 2 ネットワークに接続されているホスト、スイッチ、およびルータを攻撃することができます。図 26-1 に、ARP キャッシュポイズニングの例を示します。

図 21: ARP キャッシュポイズニング



ホスト A、B、および C は、インターフェイス A、B、および C 上にあるスイッチに接続されています。これらはすべて同一のサブネット上にあります。カッコ内に示されているのは、これらの IP アドレス、および MAC アドレスです。たとえば、ホスト A が使用する IP アドレスは IA、MAC アドレスは MA です。ホスト A が IP レイヤにあるホスト B と通信する必要がある場合、ホスト A は IP アドレス IB と関連付けられている MAC アドレスに ARP 要求をブロードキャストします。スイッチとホスト B は、この ARP 要求を受信すると、IP アドレスが IA で、MAC アドレスが MA のホストに対する ARP バインディングを ARP キャッシュに読み込みます。たとえば、IP アドレス IA は、MAC アドレス MA にバインドされています。ホスト B が応答すると、スイッチ、およびホスト A は、IP アドレスが IB で、MAC アドレスが MB のホストに対するバインディングを ARP に読み込みます。

ホスト C は、IP アドレスが IA (または IB) で、MAC アドレスが MC のホストに対するバインディングを持つ偽造 ARP 応答をブロードキャストすることにより、スイッチ、ホスト A、およびホスト B の ARP キャッシュをポイズニングすることができます。ARP キャッシュがポイズニングされたホストは、IA または IB 宛でのトラフィックに、宛先 MAC アドレスとして MAC アドレス MC を使用します。つまり、ホスト C がこのトラフィックを代行受信することになります。ホスト C は IA および IB に関連付けられた本物の MAC アドレスを知っているため、正しい MAC アドレスを宛先として使用することで、代行受信したトラフィックをこれらのホストに転送できます。ホスト C は自身をホスト A からホスト B へのトラフィック ストリームに挿入します。従来の中間者攻撃です。

ダイナミック ARP インспекションは、ネットワーク内の ARP パケットの正当性を確認するセキュリティ機能です。不正な IP/MAC アドレス バインディングを持つ ARP パケットを代行受信し、ログに記録して、廃棄します。この機能により、ネットワークをある種の中間者攻撃から保護することができます。

ダイナミック ARP インспекションにより、有効な ARP 要求と応答だけが確実にリレーされるようになります。スイッチが実行する機能は次のとおりです。

- 信頼できないポートを経由したすべての ARP 要求および ARP 応答を代行受信します。
- 代行受信した各パケットが、IP アドレスと MAC アドレスの有効なバインディングを持つことを確認してから、ローカル ARP キャッシュを更新するか、または適切な宛先にパケットを転送します。
- 無効な ARP パケットはドロップします。

ダイナミック ARP インспекションは、信頼できるデータベースである DHCP スヌーピング バインディング データベースに格納されている有効な IP/MAC アドレス バインディングに基づいて、ARP パケットの正当性を判断します。このデータベースは、VLAN およびスイッチ上で DHCP スヌーピングが有効になっている場合に、DHCP スヌーピングにより構築されます。信頼できるインターフェイスで ARP パケットが受信されると、スイッチは何もチェックせずに、このパケットを転送します。信頼できないインターフェイスでは、スイッチはこのパケットが有効である場合だけ、このパケットを転送します。

非 DHCP 環境では、ダイナミック ARP インспекションは、静的に設定された IP アドレスを持つホストに対するユーザー設定の ARP アクセスコントロールリスト (ACL) と照らし合わせて、ARP パケットの正当性を確認することができます。ARP ACL を定義するには、**arp access-list acl-name** グローバル コンフィギュレーション コマンドを使用します。

パケットの IP アドレスが無効である場合、または ARP パケットの本文にある MAC アドレスが、イーサネット ヘッダーで指定されたアドレスと一致しない場合、ARP パケットをドロップするようにダイナミック ARP インспекションを設定することができます。このためには、**ip arp inspection validate** {[src-mac] [dst-mac] [ip]} グローバル コンフィギュレーション コマンドを使用します。

インターフェイスの信頼状態とネットワーク セキュリティ

ダイナミック ARP インспекションは、スイッチの各インターフェイスに信頼状態を関連付けます。信頼できるインターフェイスに到着するパケットは、ダイナミック ARP インспек

ションの確認検査をすべてバイパスし、信頼できないインターフェイスに到着するパケットには、ダイナミック ARP インспекションの検証プロセスを受けます。

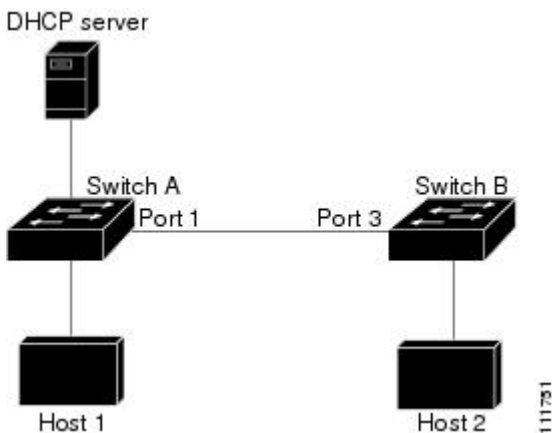
一般的なネットワーク構成では、ホスト ポートに接続されているスイッチ ポートすべてを信頼できないものに設定し、スイッチに接続されているスイッチポートすべてを信頼できるものに設定します。この構成では、指定されたスイッチからネットワークに入ってくる ARP パケットはすべて、セキュリティチェックをバイパスします。VLAN 内、またはネットワーク内のその他の場所では、他の検査を実行する必要はありません。信頼状態を設定するには、`ip arp inspection trust` インターフェイス コンフィギュレーション コマンドを使用します。



注意 信頼状態の設定は、慎重に行ってください。信頼すべきインターフェイスを信頼できないインターフェイスとして設定すると、接続が失われる場合があります。

次の図では、スイッチ A とスイッチ B の両方が、ホスト 1 とホスト 2 を含む VLAN でダイナミック ARP インспекションを実行しているとします。ホスト 1 とホスト 2 が、スイッチ A に接続している DHCP サーバーから IP アドレスを取得している場合、スイッチ A だけが、ホスト 1 の IP/MAC アドレスをバインディングします。したがって、スイッチ A とスイッチ B 間のインターフェイスが信頼できない場合は、ホスト 1 からの ARP パケットはスイッチ B ではドロップされ、ホスト 1 およびホスト 2 の間の接続は切断されます。

図 22: ダイナミック ARP インспекションのために有効にされた VLAN 上の ARP パケット検証



実際には信頼できないインターフェイスを信頼できるインターフェイスとして設定すると、ネットワーク内にセキュリティ ホールが生じます。スイッチ A でダイナミック ARP インспекションが実行されていない場合、ホスト 1 はスイッチ B の ARP キャッシュを簡単にポイズニングできます（および、これらのスイッチの間のリンクが信頼できるものとして設定されている場合はホスト 2）。この状況は、スイッチ B がダイナミック ARP インспекションを実行している場合でも発生します。

ダイナミック ARP インспекションは、ダイナミック ARP インспекションを実行しているスイッチに接続された（信頼できないインターフェイス上の）ホストが、そのネットワークにあるその他のホストの ARP キャッシュをポイズニングしていないことを保証します。しかし、ダイナミック ARP インспекションにより、ネットワークの他の部分にあるホストが、ダイ

ダイナミック ARP インспекションを実行しているスイッチに接続されているホストのキャッシュをポイズニングできないようにすることはできません。

VLAN のスイッチの一部がダイナミック ARP インспекションを実行し、残りのスイッチは実行していない場合、このようなスイッチに接続しているインターフェイスは信頼できないものとして設定します。ただし、非ダイナミック ARP インспекションスイッチからパケットのバインディングを検証するには、ARP ACL を使用して、ダイナミック ARP インспекションを実行するスイッチを設定します。このようなバインディングが判断できない場合は、レイヤ 3 で、ダイナミック ARP インспекションスイッチを実行していないスイッチから、ダイナミック ARP インспекションを実行しているスイッチを分離します。



- (注) DHCP サーバーとネットワークの設定によっては、VLAN 上のすべてのスイッチで指定された ARP パケットを検証できない可能性があります。

ARP パケットのレート制限

スイッチの CPU は、ダイナミック ARP インспекション確認検査を実行します。したがって、DoS 攻撃を阻止するために、着信 ARP パケット数はレート制限されます。デフォルトでは、信頼できないインターフェイスのレートは 15 パケット/秒 (pps) です。信頼できるインターフェイスはレート制限されません。この設定を変更するには、**ip arp inspection limit** インターフェイス コンフィギュレーション コマンドを使用します。

着信 ARP パケットのレートが設定された制限を超えると、スイッチはポートを **errdisable** ステートにします。ユーザーが介入するまで、ポートはこの状態を維持します。**errdisable recovery** グローバルコンフィギュレーションコマンドを使用すると、**errdisable** ステートの回復をイネーブルにできます。これによって、ポートは指定のタイムアウト時間が経過すると、この状態から自動的に回復するようになります。



- (注) EtherChannel のレート制限は、スタックにある各スイッチに個別に適用されます。たとえば、EtherChannel で 20 pps の制限が設定されている場合、EtherChannel にあるポートの各スイッチでは、最大 20 pps まで実行できます。スイッチが制限を超過した場合、EtherChannel 全体が **errdisable** ステートになります。

ARP ACL および DHCP スヌーピング エントリの相対的なプライオリティ

ダイナミック ARP インспекションでは、有効な IP/MAC アドレス バインディングのリストとして、DHCP スヌーピング バインディング データベースが使用されます。

DHCP スヌーピング バインディング データベース内のエントリより、ARP ACL の方が優先されます。スイッチが ACL を使用するのには、ACL が **ip arp inspection filter vlan** グローバル コンフィギュレーション コマンドを使用して作成されている場合だけです。スイッチは、まず、ARP パケットをユーザー設定の ARP ACL と比較します。DHCP スヌーピングによりデータが入力されたデータベースに有効なバインディングが存在していても、ARP ACL が ARP パケットを拒否する場合、スイッチもこのパケットを拒否します。

廃棄パケットのロギング

スイッチがパケットをドロップすると、ログバッファにエントリが記録され、その割合に応じて、システムメッセージが生成されます。メッセージの生成後、スイッチにより、ログバッファからこのエントリが消去されます。各ログエントリには、受信側の VLAN、ポート番号、送信元 IP アドレスおよび宛先 IP アドレス、送信元 MAC アドレスおよび宛先 MAC アドレスといったフロー情報が記録されます。

ip arp inspection log-buffer グローバルコンフィギュレーションコマンドを使用して、バッファ内のエントリ数や、システムメッセージ生成までの指定のインターバルに必要とされるエントリ数を設定します。記録されるパケットの種類を指定するには、**ip arp inspection vlan logging** グローバルコンフィギュレーションコマンドを使用します。

ダイナミック ARP インспекションのデフォルト設定

機能	デフォルト設定
ダイナミック ARP インспекション	すべての VLAN でディセーブル。
インターフェイスの信頼状態	すべてのインターフェイスは untrusted。
着信 ARP パケットのレート制限	1 秒間に 15 台の新規ホストに接続するホストが配置されたスイッチドネットワークの場合、信頼できないインターフェイスのレートは 15 pps に設定されます。 信頼できるすべてのインターフェイスでは、レート制限は行われません。 バースト インターバルは 1 秒です。
非 DHCP 環境に対する ARP ACL	ARP ACL は定義されません。
有効性検査	検査は実行されません。
ログ バッファ	ダイナミック ARP インспекションがイネーブル化されると、拒否またはドロップされた ARP パケットはすべてが記録されます。 ログ内のエントリ数は 32 です。 システムメッセージ数は、毎秒 5 つに制限されます。 ロギング レート インターバルは 1 秒です。
VLAN 単位のロギング	拒否または廃棄されたすべての ARP パケットが記録されます。

ARPACL および DHCP スヌーピング エントリの相対的なプライオリティ

ダイナミック ARP インスペクションでは、有効な IP/MAC アドレス バインディングのリストとして、DHCP スヌーピング バインディング データベースが使用されます。

DHCP スヌーピング バインディング データベース内のエントリより、ARP ACL の方が優先されます。スイッチが ACL を使用するのには、ACL が `ip arp inspection filter vlan` グローバル コンフィギュレーション コマンドを使用して作成されている場合だけです。スイッチは、まず、ARP パケットをユーザー設定の ARP ACL と比較します。DHCP スヌーピングによりデータが入力されたデータベースに有効なバインディングが存在していても、ARP ACL が ARP パケットを拒否する場合は、スイッチもこのパケットを拒否します。

ダイナミック ARP インスペクションの設定方法

非 DHCP 環境での ARP ACL の設定

この手順は、図 2 に示すスイッチ B がダイナミック ARP インスペクション、または DHCP スヌーピングをサポートしていないときにダイナミック ARP インスペクションを設定する方法を示しています。

スイッチ A のポート 1 を信頼できるものとして設定した場合、スイッチ A とホスト 1 は両方とも、スイッチ B またはホスト 2 により攻撃される可能性があるため、セキュリティ ホールが作り出されます。これを阻止するには、スイッチ A のポート 1 を信頼できないものとして設定する必要があります。ホスト 2 からの ARP パケットを許可するには、ARP ACL をセットアップして、これを VLAN 1 に適用する必要があります。ホスト 2 の IP アドレスがスタティックではない（スイッチ A で ACL 設定を適用することは不可能である）場合、レイヤ 3 でスイッチ A をスイッチ B から分離し、これらの間では、ルータを使用してパケットをルートする必要があります。

スイッチ A で ARP ACL を設定するには、次の手順を実行します。この手順は、非 DHCP 環境では必須です。

手順の概要

1. **enable**
2. **configure terminal**
3. **arp access-list *acl-name***
4. **permit ip host *sender-ip* mac host *sender-mac***
5. **exit**
6. **ip arp inspection filter *arp-acl-name* vlan *vlan-range* [static]**
7. **interface *interface-type* *interface-number***
8. **no ip arp inspection trust**
9. **end**
10. **show arp access-list *acl-name***
11. **show ip arp inspection vlan *vlan-range***

12. show ip arp inspection interfaces

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	arp access-list acl-name 例： Device (config)# arp access-list arpacl22	ARP ACL を定義し、ARP 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。デフォルトでは、ARP アクセス リストは定義されません。 (注) ARP アクセスリストの末尾に暗黙的な deny ip any mac any コマンドが指定されています。
ステップ 4	permit ip host sender-ip mac host sender-mac 例： Device (config-arp-nacl)# permit ip host 10.2.2.2 mac host 0018.bad8.3fbd	指定されたホスト (ホスト 2) からの ARP パケットを許可します。 <ul style="list-style-type: none"> • <i>sender-ip</i> には、ホスト 2 の IP アドレスを入力します。 • <i>sender-mac</i> には、ホスト 2 の MAC アドレスを入力します。
ステップ 5	exit 例： Device (config-arp-nacl)# exit	ARP アクセスリスト 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	ip arp inspection filter arp-acl-name vlan vlan-range [static] 例： Device (config)# ip arp inspection filter arpacl22 vlan 1-2	VLAN に ARP ACL を適用します。デフォルトでは、定義済みの ARP ACL は、どのような VLAN にも適用されません。 <ul style="list-style-type: none"> • <i>arp-acl-name</i> には、ステップ 2 で作成した ACL の名前を指定します。 • <i>vlan-range</i> では、スイッチとホストが存在する VLAN を指定します。VLAN ID 番号により識別される単一の VLAN、ハイフンで区切られた VLAN 範囲、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ~ 4094 です。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • (任意) static を指定すると、ARPACL 内の暗黙的な拒否が明示的な拒否と見なされ、それ以前に指定された ACL 句に一致しないパケットは廃棄されます。DHCP バインディングは使用されません。 <p>このキーワードを指定しない場合は、ACL 内にはパケットを拒否する明示的な拒否が存在しないこととなります。この場合は、ACL 句に一致しないパケットを許可するか拒否するかは、DHCP バインディングによって決定されます。</p> <p>IP アドレスと MAC アドレスとのバインディングしかなかった ARP パケットは、ACL に照合されず、パケットは、アクセスリストで許可された場合だけに許可されます。</p>
<p>ステップ 7</p>	<p>interface <i>interface-type interface-number</i></p> <p>例 :</p> <pre>Device(config)# interface gigabitethernt 0/1/1</pre>	<p>スイッチ B に接続されたスイッチ A の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。</p>
<p>ステップ 8</p>	<p>no ip arp inspection trust</p> <p>例 :</p> <pre>Device(config-if)# no ip arp inspection trust</pre>	<p>スイッチ B に接続されたスイッチ A のインターフェイスを untrusted として設定します。</p> <p>デフォルトでは、すべてのインターフェイスは信頼できません。</p> <p>信頼できないインターフェイスでは、スイッチはすべての ARP 要求と応答を代行受信します。ルータは、代行受信した各パケットが、IP アドレスと MAC アドレスとの有効なバインディングを持つことを確認してから、ローカル キャッシュを更新するか、適切な宛先にパケットを転送します。スイッチは、無効なパケットをドロップし、ip arp inspection vlan logging 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ロギング設定に従ってログバッファに記録します。</p>
<p>ステップ 9</p>	<p>end</p> <p>例 :</p> <pre>Device(config-if)# end</pre>	<p>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、特権 EXEC モードに戻ります。</p>
<p>ステップ 10</p>	<p>show arp access-list <i>acl-name</i></p> <p>例 :</p> <pre>Device# show arp access-list arpacl22</pre>	<p>名前付き ACL に関する情報を表示します。</p>

	コマンドまたはアクション	目的
ステップ 11	show ip arp inspection vlan <i>vlan-range</i> 例： Device# show ip arp inspection vlan 1-2	選択した範囲の VLAN の統計情報を表示します。
ステップ 12	show ip arp inspection interfaces 例： Device# show ip arp inspection interfaces	指定したインターフェイスに関して ARP パケットの信頼状態とレート制限を表示します。

DHCP 環境でのダイナミック ARP インспекションの設定

始める前に

この手順では、2つのスイッチがダイナミック ARP インспекションをサポートしているときに、この機能を設定する方法を示します。ホスト 1 はスイッチ A に、ホスト 2 はスイッチ B にそれぞれ接続されています。スイッチは両方とも、ホストが配置されている VLAN 1 でダイナミック ARP インспекションを実行しています。DHCP サーバーはスイッチ A に接続されています。両方のホストは、同一の DHCP サーバーから IP アドレスを取得します。したがって、スイッチ A はホスト 1 およびホスト 2 に対するバインディングを、スイッチ B はホスト 2 に対するバインディングを持ちます。



- (注) 着信 ARP 要求、および ARP 応答で IP/MAC アドレスバインディングを検証するために、ダイナミック ARP インспекション DHCP スヌーピングバインディングデータベースのエントリに依存します。IP アドレスがダイナミックに割り当てられた ARP パケットを許可する際は、DHCP スヌーピングをイネーブルにしてください。

ダイナミック ARP インспекションを設定するには、次の手順を実行します。この処理は、両方のスイッチで行う必要があります。この手順は必須です。

手順の概要

1. **enable**
2. **show cdp neighbors**
3. **configure terminal**
4. **ip arp inspection vlan *vlan-range***
5. **Interfacetype number**
6. **ip arp inspection trust**
7. **end**
8. **show ip arp inspection interfaces**
9. **show ip arp inspection vlan *vlan-range***
10. **show ip dhcp snooping binding**
11. **show ip arp inspection statistics vlan *vlan-range***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	show cdp neighbors 例： Device# show cdp neighbors	スイッチ間の接続を確認します。
ステップ 3	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 4	ip arp inspection vlan <i>vlan-range</i> 例： Device(config)# ip arp inspection vlan 1	VLAN 単位で、ダイナミック ARP インспекションをイネーブルにします。デフォルトでは、すべての VLAN 上でダイナミック ARP インспекションはディセーブルになっています。vlan-range には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ~ 4094 です。両方のスイッチに同じ VLAN ID を指定します。
ステップ 5	Interface <i>number</i> 例： Device(config)# interface gigabitethernet 1/0/1	他のスイッチに接続されるインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	ip arp inspection trust 例： Device(config-if)# ip arp inspection trust	スイッチ間の接続を trusted に設定します。デフォルトでは、すべてのインターフェイスは信頼できません。 スイッチは、信頼できるインターフェイスにあるもう 1 つのスイッチから受信した ARP パケットは確認しません。この場合、パケットはそのまま転送されます。 信頼できないインターフェイスでは、スイッチはすべての ARP 要求と応答を代行受信します。ルータは、代行受信した各パケットが、IP アドレスと MAC アドレスとの有効なバインディングを持つことを確認してから、ローカル キャッシュを更新するか、適切な宛先にパケットを転送します。スイッチは、無効なパケットをドロップし、ip arp

	コマンドまたはアクション	目的
		inspection vlan logging 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ロギング設定に従ってログ バッファに記録します。
ステップ 7	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 8	show ip arp inspection interfaces 例： Device# show ip arp inspection interfaces	インターフェイスでダイナミック ARP インспекションの設定を検証します。
ステップ 9	show ip arp inspection vlan vlan-range 例： Device# show ip arp inspection vlan 1	VLAN でダイナミック ARP インспекションの設定を検証します。
ステップ 10	show ip dhcp snooping binding 例： Device# show ip dhcp snooping binding	DHCP バインディングを確認します。
ステップ 11	show ip arp inspection statistics vlan vlan-range 例： Device# show ip arp inspection statistics vlan 1	VLAN でダイナミック ARP インспекションの統計情報を確認します。

着信 ARP パケットのレート制限

スイッチの CPU は、ダイナミック ARP インспекション確認検査を実行します。したがって、DoS 攻撃を阻止するために、着信 ARP パケット数はレート制限されます。

着信 ARP パケットのレートが設定された制限を超えると、スイッチはポートを `errdisable` ステートにします。 `errdisable` 回復をイネーブルにして、指定されたタイムアウト時間の後にポートがこのステートから自動的に抜け出すようにするまで、ポートはこのステートのままです。



- (注) インターフェイス上のレート制限を設定しない限り、インターフェイスの信頼状態を変更することは、レート制限をその信頼状態のデフォルト値に変更することになります。レート制限を設定すると、信頼状態が変更された場合でもインターフェイスはレート制限を保ちます。 **no ip arp inspection limit** インターフェイス コンフィギュレーション コマンドを入力すると、インターフェイスはデフォルトのレート制限に戻ります。

着信 ARP パケットのレートを制限するには、次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip arp inspection limit** {rate pps [burst interval seconds] | none}
5. **exit**
6. 次のコマンドを使用します。
 - **errdisable detect cause arp-inspection**
 - **errdisable recovery cause arp-inspection**
 - **errdisable recovery interval** *interval*
7. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>type number</i> 例： Device(config)# interface gigabitethernet 0/1/2	レート制限されるインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	ip arp inspection limit {rate pps [burst interval seconds] none}	インターフェイス上の着信 ARP 要求および ARP 応答のレートを制限します。デフォルトレートは、信頼できないインターフェイスでは 15 pps、信頼できるインターフェイスでは無制限です。バーストインターバルは 1 秒です。 キーワードの意味は次のとおりです。 <ul style="list-style-type: none"> • ratepps には、1 秒あたりに処理される着信パケット数の上限を指定します。有効な範囲は 0 ~ 2048 pps です。 • (任意) burst intervalseconds は、レートの高い ARP パケットの有無についてインターフェイスがモニタリングされる間隔 (秒) を指定します。指定できる範囲は 1 ~ 15 です。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • rate none には、処理可能な着信 ARP パケットのレートに上限を指定しません。
ステップ 5	exit 例： Device(config-if) # exit	インターフェイス コンフィギュレーション モードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 6	次のコマンドを使用します。 <ul style="list-style-type: none"> • errdisable detect cause arp-inspection • errdisable recovery cause arp-inspection • errdisable recovery interval interval 例： Device(config) # errdisable recovery cause arp-inspection	(任意) ダイナミック ARP インспекションの errdisable ステートからのエラー回復をイネーブルにし、ダイナミック ARP インспекションの回復メカニズムで使用する変数を設定します。 デフォルトでは、回復はディセーブルで、回復のインターバルは 300 秒です。 interval interval には、 error-disabled ステートから回復する時間を秒単位で指定します。指定できる範囲は 30 ~ 86400 です。
ステップ 7	exit 例： Device(config) # exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ダイナミック ARP インспекション検証チェックの実行

ダイナミック ARP インспекションは、不正な IP/MAC アドレス バインディングを持つ ARP パケットを代行受信し、ログに記録して、廃棄します。宛先 MAC アドレス、送信側および宛先の IP アドレス、および送信元 MAC アドレスで追加検証を実行するように、スイッチを設定できます。

着信 ARP パケットで特定のチェックを実行するには、次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip arp inspection validate {[src-mac] [dst-mac] [ip]}**
4. **exit**
5. **show ip arp inspection vlan vlan-range**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip arp inspection validate {[src-mac] [dst-mac] [ip]} 例： Device(config)# ip inspection validate ip	<p>着信 ARP パケットで特定の検査を実行します。デフォルトでは、検証は実行されません。</p> <p>キーワードの意味は次のとおりです。</p> <ul style="list-style-type: none"> • src-mac では、イーサネットヘッダーの送信元 MAC アドレスと ARP 本文の送信元 MAC アドレスが比較されます。この検査は、ARP 要求および ARP 応答の両方に対して実行されます。イネーブルにすると、異なる MAC アドレスを持つパケットは無効パケットとして分類され、廃棄されます。 • dst-mac では、イーサネットヘッダーの宛先 MAC アドレスと ARP 本文の宛先 MAC アドレスが比較されます。この検査は、ARP 応答に対して実行されます。イネーブルにすると、異なる MAC アドレスを持つパケットは無効パケットとして分類され、廃棄されます。 • ip では、ARP 本文から、無効な IP アドレスや予期しない IP アドレスがないかを確認します。アドレスには 0.0.0.0、255.255.255.255、およびすべての IP マルチキャスト アドレスが含まれます。送信元 IP アドレスはすべての ARP 要求および ARP 応答内で検査され、宛先 IP アドレスは ARP 応答内だけで検査されます。 <p>少なくとも 1 つのキーワードを指定する必要があります。コマンドを実行するたびに、その前のコマンドの設定は上書きされます。つまり、コマンドが src および dst mac の検証をイネーブルにし、別のコマンドが IP 検証だけをイネーブルにすると、2 番目のコマンドによって src および dst mac の検証がディセーブルになります。</p>

	コマンドまたはアクション	目的
ステップ 4	exit 例： Device(config)# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 5	show ip arp inspection vlan <i>vlan-range</i> 例： Device# show ip arp insepction vlan 1-2	選択した範囲の VLAN の統計情報を表示します。

DAI のモニタリング

DAI をモニターするには、次のコマンドを使用します。

コマンド	説明
clear ip arp inspection statistics	ダイナミック ARP インспекション統計情報をクリアします。
show ip arp inspection statistics [vlan <i>vlan-range</i>]	指定の VLAN において、転送されたパケット、廃棄されたパケット、MAC 検証に失敗したパケット、IP 検証に失敗したパケット、ACL によって許可および拒否されたパケット、DHCP によって許可および拒否されたパケットの統計情報を表示します。VLAN が指定されていない場合、または範囲が指定されている場合は、ダイナミック ARP インспекションがイネーブルにされた (アクティブ) VLAN だけの情報を表示します。
clear ip arp inspection log	ダイナミック ARP インспекションログバッファをクリアします。
show ip arp inspection log	ダイナミック ARP インспекションログバッファの設定と内容を表示します。

show ip arp inspection statistics コマンドでは、スイッチは信頼されたダイナミック ARP インспекションポート上の各 ARP 要求および応答パケットの転送済みパケット数を増加させます。スイッチは、送信元 MAC、宛先 MAC、または IP 検証チェックによって拒否された各パケットの ACL または DHCP 許可済みパケット数を増加させ、適切な失敗数を増加させます。

DAI の設定の確認

DAI の設定を表示して確認するには、次のコマンドを使用します。

コマンド	説明
show arp access-list <i>[acl-name]</i>	ARP ACL についての詳細情報を表示します。
show ip arp inspection interfaces <i>[interface-id]</i>	指定されたインターフェイスまたはすべてのインターフェイスの ARP パケットの信頼状態およびレート制限を表示します。
show ip arp inspection vlan <i>vlan-range</i>	指定された VLAN のダイナミック ARP インспекションの設定および動作ステータスを表示します。VLAN が指定されていない場合、または範囲が指定されている場合は、ダイナミック ARP インспекションがイネーブルにされた (アクティブ) VLAN だけの情報を表示します。

ダイナミック ARP インспекションの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	ダイナミック ARP インспекション	ARP では、IP アドレスを MAC アドレスにマッピングすることで、レイヤ2ブロードキャストドメイン内の IP 通信を実現します。ダイナミック ARP インспекションは、ネットワーク内の ARP パケットの正当性を確認するセキュリティ機能です。不正な IP/MAC アドレスバインディングを持つ ARP パケットを代行受信し、ログに記録して、廃棄します。この機能により、ネットワークをある種の中間者攻撃から保護することができます。

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



第 21 章

IPv6 ファースト ホップ セキュリティの設定

- IPv6 ファースト ホップ セキュリティの前提条件 (441 ページ)
- IPv6 ファースト ホップ セキュリティの制約事項 (441 ページ)
- IPv6 ファースト ホップ セキュリティに関する情報 (442 ページ)
- IPv6 ファースト ホップ セキュリティの設定方法 (444 ページ)
- IPv6 ファースト ホップ セキュリティの設定例 (475 ページ)
- IPv6 ファースト ホップ セキュリティに関する追加情報 (476 ページ)
- IPv6 ファースト ホップ セキュリティの機能履歴 (477 ページ)

IPv6 ファースト ホップ セキュリティの前提条件

- 必要な、IPv6 が有効になっている SDM テンプレートが設定されていること。
- IPv6 ネイバー探索機能についての知識が必要です。

IPv6 ファースト ホップ セキュリティの制約事項

- 次の制限は、FHS ポリシーを EtherChannel インターフェイスに適用する場合に該当します (ポート チャンネル)。
 - FHS ポリシーがアタッチされた物理ポートは EtherChannel グループに参加することができません。
 - FHS ポリシーは、EtherChannel グループのメンバーである場合に物理ポートにアタッチすることができません。
- デフォルトでは、スヌーピング ポリシーにはセキュリティ レベルのガードがあります。そのようなスヌーピングポリシーがアクセススイッチに設定されると、デバイスまたは DHCP サーバー/リレーに対応するアップリンクポートが信頼できるポートとして設定されていても、IPv6 (DHCPv6) サーバーパケットに対する外部 IPv6 ルータアドバタイズメン

ト (RA) または Dynamic Host Configuration Protocol はブロックされます。IPv6 RA または DHCPv6 サーバー メッセージを許可するには、次の手順を実行します。

- IPv6 RA ガード ポリシー (RA の場合) または IPv6 DHCP ガード ポリシー (DHCP サーバー メッセージの場合) をアップリンク ポートに適用します。
- 低いセキュリティ レベルでスヌーピング ポリシーを設定します (たとえば、`glean` や `inspect` など)。しかし、ファースト ホップ セキュリティ機能の利点が有効でないため、このようなスヌーピング ポリシーでは、低いセキュリティ レベルを設定することはお勧めしません。

IPv6 ファースト ホップ セキュリティに関する情報

IPv6 ファースト ホップ セキュリティの概要

IPv6 のファーストホップセキュリティ (FHS IPv6) は、ポリシーを物理インターフェイス、EtherChannel インターフェイス、または VLAN にアタッチできる一連の IPv6 セキュリティ機能です。IPv6 ソフトウェア ポリシー データベース サービスは、これらのポリシーを保存しアクセスします。ポリシーを設定または変更すると、ポリシー属性はソフトウェアポリシーデータベースに保存または更新され、その後指定したとおりに適用されます。次の IPv6 ポリシーが現在サポートされています。

- IPv6 スヌーピング ポリシー : IPv6 スヌーピング ポリシーは、IPv6 内の FHS で使用できるほとんどの機能をイネーブルにできるコンテナ ポリシーとして機能します。
- IPv6 FHS バインディング テーブル コンテンツ : デバイスに接続された IPv6 ネイバーのデータベーステーブルはネイバー探索 (ND) プロトコルスヌーピングなどの情報ソースから作成されます。このデータベースまたはバインディングテーブルは、リンク層アドレス (LLA)、IPv4 または IPv6 アドレス、およびスプーフィングやリダイレクト攻撃を防止するためにネイバーのプレフィックスバインディングを検証するために、さまざまな IPv6 ガード機能 (IPv6 ND インスペクションなど) によって使用されます。
- IPv6 ネイバー探索検査 : IPv6 ND 検査は、レイヤ 2 ネイバーテーブル内のステートレス自動設定アドレスのバインディングを学習し、保護します。IPv6 ND 検査は、信頼できるバインディング テーブル データベースを構築するためにネイバー探索メッセージを分析します。準拠していない IPv6 ネイバー探索メッセージは破棄されます。ND メッセージは、その IPv6 からメディアアクセスコントロール (MAC) へのマッピングが検証可能な場合に信頼できると見なされます。

この機能によって、DAD、アドレス解決、ルータ ディスカバリ、ネイバー キャッシュに対する攻撃などの、ND メカニズムに固有の脆弱性のいくつかが軽減されます。

- IPv6 ルータ アドバタイズメント ガード : IPv6 ルータ アドバタイズメント (RA) ガード機能を使用すると、ネットワーク管理者は、ネットワーク デバイス プラットフォームに到着した不要または不正な RA ガードメッセージをブロックまたは拒否できます。RA は、リンクで自身をアナウンスするためにデバイスによって使用されます。RA ガード機能は、

これらの RA を分析して、未承認のデバイスによって送信された偽の RA をフィルタリングして除外します。ホストモードでは、ポートではルータ アドバタイズメントとルータ リダイレクト メッセージはすべて許可されません。RA ガード機能は、レイヤ 2 デバイスの設定情報を、受信した RA フレームで検出された情報と比較します。レイヤ 2 デバイスは、RA フレームとルータリダイレクトフレームの内容を設定と照らし合わせて検証した後で、RA をユニキャストまたはマルチキャストの宛先に転送します。RA フレームの内容が検証されない場合は、RA は破棄されます。

- **IPv6 DHCP ガード** : IPv6 DHCP ガード機能は、承認されない DHCPv6 サーバーおよびリレー エージェントからの返信およびアドバタイズメント メッセージをブロックします。IPv6 DHCP ガードは、偽造されたメッセージがバインディングテーブルに入るのを防ぎ、DHCPv6 サーバーまたは DHCP リレーからデータを受信することが明示的に設定されていないポートで受信された DHCPv6 サーバー メッセージをブロックできます。この機能を使用するには、ポリシーを設定してインターフェイスまたは VLAN にアタッチします。DHCP ガードパケットをデバッグするには、**debug ipv6 snooping dhcp-guard** 特権 EXEC コマンドを使用します。

- **IPv6 ソース ガード** : IPv4 ソース ガードと同様、IPv6 ソース ガードは送信元アドレス スプーフィングを防ぐために、送信元アドレスまたはプレフィックスを検証します。

ソースガードでは、送信元または宛先アドレスに基づいてトラフィックを許可または拒否するようにハードウェアをプログラムします。ここでは、データパケットのトラフィックのみを処理します。

IPv6 ソース ガード機能は、ハードウェア TCAM テーブルにエントリを格納し、ホストが無効な IPv6 送信元アドレスでパケットを送信しないようにします。

ソースガードパケットをデバッグするには、**debug ipv6 snooping source-guard** 特権 EXEC コマンドを使用します。



- (注) IPv6 ソースガード機能およびプレフィックスガード機能は、入力方向でのみサポートされています。つまり、出力方向ではサポートされていません。

次の制約事項が適用されます。

- FHS ポリシーは、EtherChannel グループのメンバーである場合に物理ポートにアタッチすることができません。
- IPv6 ソース ガードがスイッチ ポートで有効になっている場合は、そのスイッチ ポートが属するインターフェイスで NDP または DHCP スヌーピングを有効にする必要があります。そうしないと、このポートからのすべてのデータトラフィックがブロックされます。
- IPv6 ソース ガードポリシーを VLAN に適用することはできません。インターフェイス レベルのみでサポートされています。

- インターフェイスで IPv4 および IPv6 のソース ガードを一緒に設定する場合は、**ip verify source** の代わりに **ip verify source mac-check** の使用を推奨します。2つの異なるフィルタリングルール (IPv4 (IP フィルタ) 用と IPv6 (IP-MAC フィルタ) 用) が設定されているため、特定のポートの IPv4 接続が切断される可能性があります。
- IPv6 ソース ガードとプレフィックス ガードは同時に使用できません。ポリシーをインターフェイスに付加する際は、「アドレスを確認」するか「プレフィックスを確認」する必要がありますが、両方を確認する必要はありません。
- PVLAN と送信元/プレフィックス ガードは同時に適用できません。
- IPv6 送信元ガードとプレフィックスガードは EtherChannel でサポートされています。
- IPv6 プレフィックス ガード: IPv6 プレフィックス ガードは、IPv6 送信元ガード機能内で動作し、デバイスがトポロジに不正なアドレスから発信されたトラフィックを拒否できるようにします。IPv6 プレフィックス ガードは、IPv6 プレフィックスが DHCP プレフィックス委任を使用してデバイス (ホーム ゲートウェイなど) に委任される場合によく使用されています。この機能は、リンクに割り当てられたアドレスの範囲を検出し、この範囲に入っていないアドレスを発信元とするトラフィックをブロックします。
- IPv6 宛先ガード: IPv6 宛先ガード機能は、IPv6 ネイバー探索で動作し、リンク上でアクティブであると認識されているアドレスについてのみ、デバイスがアドレスを解決します。アドレスグリーンニング機能に依存して、リンク上でアクティブなすべての宛先をバインディング テーブルに挿入してから、バインディング テーブルで宛先が見つからなかったときに実行される解決をブロックします。



(注) IPv6 宛先ガードは、設定された SVI のレイヤ2 VLANに適用することをお勧めします。

IPv6 ファースト ホップ セキュリティの設定方法

IPv6 スヌーピング ポリシーの設定



(注) IPv6 スヌーピングポリシー機能は廃止されました。コマンドは CLI に表示され、設定できますが、代わりにスイッチ統合セキュリティ機能 (SISF) ベースのデバイス追跡機能を使用することを推奨します。

IPv6 スヌーピング ポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 snooping policy *policy-name***
4. **{[default]|[device-role {node | switch}]|[limit address-count *value*]|[no] |[protocol {dhcp | ndp}]|[security-level {glean | guard | inspect}]|[tracking {disable [stale-lifetime [*seconds* | infinite]] enable [reachable-lifetime [*seconds* | infinite] }]|[trusted-port]}**
5. **end**
6. **show ipv6 snooping policy *policy-name***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： <pre>Device> enable</pre>	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： <pre>Device# configure terminal</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 snooping policy <i>policy-name</i> 例： <pre>Device(config)# ipv6 snooping policy example_policy</pre>	スヌーピングポリシーを作成し、IPv6 スヌーピングポリシー コンフィギュレーション モードを開始します。
ステップ 4	{[default] [device-role {node switch}] [limit address-count <i>value</i>] [no] [protocol {dhcp ndp}] [security-level {glean guard inspect}] [tracking {disable [stale-lifetime [<i>seconds</i> infinite]] enable [reachable-lifetime [<i>seconds</i> infinite] }] [trusted-port]} 例： <pre>Device(config-ipv6-snooping)# security-level inspect</pre> 例： <pre>Device(config-ipv6-snooping)# trusted-port</pre>	データアドレス グリーニングを有効にし、さまざまな条件に対してメッセージを検証し、メッセージのセキュリティ レベルを指定します。 <ul style="list-style-type: none"> • (任意) default : すべてをデフォルト オプションに設定します。 • (任意) device-role {node switch} : ポートに接続されたデバイスの役割を指定します。デフォルトは node です。 • (任意) limit address-count <i>value</i> : ターゲットごとに許可されるアドレス数を制限します。 • (任意) no : コマンドを無効にするか、またはそのデフォルトに設定します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • (任意) protocol{dhcp ndp} : 分析のために、スヌーピング機能にどのプロトコルをリダイレクトするかを指定します。デフォルトは、dhcp および ndp です。デフォルトを変更するには、no protocol コマンドを使用します。 • (任意) security-level{glean guard inspect} : この機能によって適用されるセキュリティのレベルを指定します。デフォルトは guard です。 <ul style="list-style-type: none"> glean : メッセージからアドレスを収集し、何も確認せずにバインディング テーブルに入力します。 guard : アドレスを収集し、メッセージを検査します。さらに、RA および DHCP サーバメッセージを拒否します。これがデフォルトのオプションです。 inspect : アドレスを収集し、メッセージの一貫性と準拠を検証して、アドレスの所有権を適用します。 • (任意) tracking {disable enable} : デフォルトの追跡動作を上書きし、追跡オプションを指定します。 • (任意) trusted-port : 信頼できるポートを設定します。これにより、該当するターゲットに対するガードがディセーブルになります。信頼できるポートを経由して学習されたバインディングは、他のどのポートを経由して学習されたバインディングよりも優先されます。テーブル内にエントリを作成しているときに衝突が発生した場合、信頼できるポートが優先されます。
ステップ 5	end 例 : Device(config-ipv6-snooping)# end	IPv6 スヌーピング ポリシー コンフィギュレーションモードを終了し、特権EXECモードに戻ります。
ステップ 6	show ipv6 snooping policy policy-name 例 : Device# show ipv6 snooping policy example_policy	スヌーピング ポリシー設定を表示します。

次のタスク

IPv6 スヌーピング ポリシーをインターフェイスまたは VLAN にアタッチします。

インターフェイスへの IPv6 スヌーピングポリシーの適用

インターフェイスまたはVLANにIPv6スヌーピングポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** Interface_type *stack/module/port*
4. **switchport**
5. **ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
6. **end**
7. **show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface Interface_type <i>stack/module/port</i> 例： Device(config)# interface gigabitethernet 1/1/4	インターフェイスのタイプおよび識別子を指定し、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 4	switchport 例：	switchport モードを開始します。

	コマンドまたはアクション	目的
	Device(config-if)# switchport	(注) インターフェイスがレイヤ3モードの場合に、レイヤ2パラメータを設定するには、パラメータを指定せずに switchport インターフェイスコンフィギュレーション コマンドを入力し、インターフェイスをレイヤ2モードにする必要があります。これにより、インターフェイスがいったんシャットダウンしてから再度有効になり、インターフェイスが接続しているデバイスに関するメッセージが表示されることがあります。レイヤ3モードのインターフェイスをレイヤ2モードにした場合、影響のあるインターフェイスに関連する以前の設定情報が消失する可能性があり、インターフェイスはデフォルト設定に戻ります。 switchport コンフィギュレーション モードではコマンドプロンプトは (config-if) # と表示されません。
ステップ 5	ipv6 snooping [attach-policy policy_name [vlan {vlan_id add vlan_ids exceptvlan_ids none remove vlan_ids}] vlan {vlan_id add vlan_ids exceptvlan_ids none remove vlan_ids all}] 例 : Device(config-if)# ipv6 snooping Device(config-if)# ipv6 snooping attach-policy example_policy Device(config-if)# ipv6 snooping vlan 111,112 Device(config-if)# ipv6 snooping attach-policy example_policy vlan 111,112	インターフェイスまたはそのインターフェイス上の特定の VLAN にカスタム IPv6 スヌーピングポリシーをアタッチします。デフォルトポリシーをインターフェイスにアタッチするには、 attach-policy キーワードを指定せずに ipv6 snooping コマンドを使用します。デフォルト ポリシーをインターフェイス上の VLAN にアタッチするには、 ipv6 snooping vlan コマンドを使用します。デフォルトポリシーは、セキュリティ レベル guard 、デバイス ロール node 、プロトコル ndp および dhep です。
ステップ 6	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 7	show running-config 例 : Device# show running-config	インターフェイス コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

レイヤ 2 EtherChannel インターフェイスへの IPv6 スヌーピングポリシーの適用

EtherChannel インターフェイスまたは VLAN に IPv6 スヌーピング ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface range <i>Interface_name</i> 例： Device(config)# interface range Port-channel 11	EtherChannel の作成時に割り当てられたポートチャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーションモードを開始します。 ヒント インターフェイスの名前とタイプを簡単に参照するには show interfaces summary コマンドを入力します。
ステップ 4	ipv6 snooping [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] 例： Device(config-if-range)# ipv6 snooping attach-policy example_policy Device(config-if-range)# ipv6 snooping attach-policy example_policy vlan 222,223,224 Device(config-if-range)# ipv6 snooping vlan 222,223,224	IPv6 スヌーピング ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例： Device(config-if-range)# end	インターフェイス範囲コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

VLAN への IPv6 スヌーピングポリシーのグローバル適用

	コマンドまたはアクション	目的
ステップ 6	show running-config interfaceportchannel_interface_name 例 : Device# show running-config interface portchannel 11	ポリシーが指定のインターフェイスに適用されていることを確認します。

VLAN への IPv6 スヌーピングポリシーのグローバル適用

複数のインターフェイス上の VLAN に IPv6 スヌーピング ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan configuration vlan_list**
4. **ipv6 snooping [attach-policy policy_name]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan configuration vlan_list 例 : Device(config)# vlan configuration 333	IPv6 スヌーピングポリシーを適用する VLAN を指定し、VLAN 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	ipv6 snooping [attach-policy policy_name] 例 : Device(config-vlan-config)# ipv6 snooping attach-policy example_policy	すべてのデバイスインターフェイスで、指定した VLAN に IPv6 スヌーピングポリシーを適用します。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。デフォルト ポリシーは、セキュリティ レベル guard 、デバイス ロール node 、プロトコル ndp および dhcp です。

	コマンドまたはアクション	目的
ステップ 5	end 例： Device (config-vlan-config) # end	VLAN 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、特権 EXEC モードに戻ります。

IPv6 バインディング テーブルの内容の設定

IPv6 バインディング テーブル コンテンツを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **[no] ipv6 neighbor binding [vlan vlan-id {ipv6-address interface interface_type stack/module/port hw_address [reachable-lifetimevalue [seconds | default | infinite] | [tracking { [default | disable] [reachable-lifetimevalue [seconds | default | infinite] | [enable [reachable-lifetimevalue [seconds | default | infinite] | [retry-interval {seconds} default [reachable-lifetimevalue [seconds | default | infinite] }] }**
4. **[no] ipv6 neighbor binding max-entries number [mac-limit number | port-limit number [mac-limit number] | vlan-limit number [[mac-limit number] | [port-limit number [mac-limitnumber]]]]**
5. **ipv6 neighbor binding logging**
6. **exit**
7. **show ipv6 neighbor binding**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	[no] ipv6 neighbor binding [vlan vlan-id {ipv6-address interface interface_type stack/module/port hw_address [reachable-lifetimevalue [seconds default infinite] [tracking { [default disable] [reachable-lifetimevalue [seconds default infinite] [enable [reachable-lifetimevalue [seconds default infinite] [retry-interval {seconds} default	バインディング テーブル データベースにスタティック エントリを追加します。

	コマンドまたはアクション	目的
	<pre>[reachable-lifetimevalue [seconds default infinite] }]</pre> <p>例 :</p> <pre>Device(config)# ipv6 neighbor binding</pre>	
ステップ 4	<pre>[no] ipv6 neighbor binding max-entries number [mac-limit number port-limit number [mac-limit number] vlan-limit number [[mac-limit number] [port-limit number [mac-limitnumber]]]]</pre> <p>例 :</p> <pre>Device(config)# ipv6 neighbor binding max-entries 30000</pre>	バインディング テーブル キャッシュに挿入できるエントリの最大数を指定します。
ステップ 5	<pre>ipv6 neighbor binding logging</pre> <p>例 :</p> <pre>Device(config)# ipv6 neighbor binding logging</pre>	バインディング テーブル メイン イベントのログギングを有効にします。
ステップ 6	<pre>exit</pre> <p>例 :</p> <pre>Device(config)# exit</pre>	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 7	<pre>show ipv6 neighbor binding</pre> <p>例 :</p> <pre>Device# show ipv6 neighbor binding</pre>	バインディング テーブルの内容を表示します。

IPv6 ネイバー探索インスペクションポリシーの設定

特権 EXEC モードから、IPv6 ND インスペクションポリシーを設定するには、次の手順に従ってください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 nd inspection policy *policy-name***
4. **device-role {host | switch}**
5. **limit address-count *value***
6. **tracking {enable [reachable-lifetime {*value* | infinite}] | disable [stale-lifetime {*value* | infinite}]}**
7. **trusted-port**
8. **validate source-mac**
9. **no {device-role | limit address-count | tracking | trusted-port | validate source-mac}**
10. **default {device-role | limit address-count | tracking | trusted-port | validate source-mac}**
11. **end**
12. **show ipv6 nd inspection policy *policy_name***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 nd inspection policy <i>policy-name</i> 例： Device (config)# ipv6 nd inspection policy example_policy	ND 検査ポリシー名を指定し、ND 検査ポリシー コンフィギュレーション モードを開始します。
ステップ 4	device-role {host switch} 例： Device (config-nd-inspection)# device-role switch	ポートに接続されているデバイスの役割を指定します。デフォルトは host です。
ステップ 5	limit address-count <i>value</i> 例： Device (config-nd-inspection)# limit address-count 1000	ポートで使用できる IPv6 アドレスの数を制限します。
ステップ 6	tracking {enable [reachable-lifetime {<i>value</i> infinite}] disable [stale-lifetime {<i>value</i> infinite}]} 例： Device (config-nd-inspection)# tracking disable stale-lifetime infinite	ポートのデフォルトのデバイス追跡ポリシーを上書きします。
ステップ 7	trusted-port 例： Device (config-nd-inspection)# trusted-port	信頼できるポートにするポートを設定します。
ステップ 8	validate source-mac 例： Device (config-nd-inspection)# validate source-mac	送信元 Media Access Control (MAC) アドレスをリンク層アドレスと照合します。
ステップ 9	no {device-role limit address-count tracking trusted-port validate source-mac} 例： Device (config-nd-inspection)# no validate source-mac	このコマンドの no 形式を使用してパラメータの現在の設定を削除します。

■ インターフェイスへの IPv6 ネイバー探索インスペクションポリシーの適用

	コマンドまたはアクション	目的
ステップ 10	default {device-role limit address-count tracking trusted-port validate source-mac} 例： Device(config-nd-inspection)# default limit address-count	設定をデフォルト値に戻します。
ステップ 11	end 例： Device(config-nd-inspection)# end	ND インスペクション ポリシー コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 12	show ipv6 nd inspection policy policy_name 例： Device# show ipv6 nd inspection policy example_policy	ND インスペクションの設定を確認します。

インターフェイスへの IPv6 ネイバー探索インスペクションポリシーの適用

インターフェイスまたはそのインターフェイス上の VLAN に IPv6 ND 検査ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-type interface-number**
4. **ipv6 nd inspection [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]]**
5. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	interface <i>interface-type interface-number</i> 例 : Device(config)# interface gigabitethernet 1/1/4	インターフェイスのタイプおよび ID 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 4	ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] 例 : Device(config-if)# ipv6 nd inspection attach-policy example_policy Device(config-if)# ipv6 nd inspection attach-policy example_policy vlan 222,223,2 Device(config-if)# ipv6 nd inspection vlan 222, 223,224	ネイバー探索検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

レイヤ 2 EtherChannel インターフェイスへの IPv6 ネイバー探索インスペクションポリシーの適用

EtherChannel インターフェイスまたは VLAN に IPv6 ネイバー探索検査ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface range** *interface_name*
4. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 :	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合) 。

VLAN への IPv6 ネイバー探索インスペクションポリシーのグローバル適用

	コマンドまたはアクション	目的
	Device> enable	
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface range interface_name 例： Device(config)# interface range Port-channel 11	EtherChannel の作成時に割り当てられたポートチャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーションモードを開始します。 ヒント インターフェイスの名前とタイプを簡単に参照するには show interfaces summary コマンドを入力します。
ステップ 4	ipv6 nd inspection [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}] vlan [{vlan_ids add vlan_ids exceptvlan_ids none remove vlan_ids all}] 例： Device(config-if-range)# ipv6 nd inspection attach-policy example_policy Device(config-if-range)# ipv6 nd inspection vlan 222, 223,224 Device(config-if-range)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224	ND 検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例： Device(config-if-range)# end	インターフェイス範囲コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

VLAN への IPv6 ネイバー探索インスペクションポリシーのグローバル適用

複数のインターフェイス上の VLAN に IPv6 ND 探索ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan configuration vlan_list**
4. **ipv6 nd inspection [attach-policy policy_name]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan configuration <i>vlan_list</i> 例： Device(config)# vlan configuration 334	IPv6 スヌーピングポリシーを適用する VLAN を指定し、VLAN 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	ipv6 nd inspection [attach-policy <i>policy_name</i>] 例： Device(config-vlan-config)# ipv6 nd inspection attach-policy <i>example_policy</i>	すべてのスイッチおよびスタック インターフェイスで、IPv6 ネイバー探索ポリシーを指定した VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。 デフォルトのポリシーは、 device-role host 、 no drop-unsecure 、 limit address-count disabled 、 sec-level minimum is disabled 、 tracking is disabled 、 no trusted-port 、 no validate source-mac です。
ステップ 5	end 例： Device(config-vlan-config)# end	VLAN 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、特権 EXEC モードに戻ります。

IPv6 ルータ アドバタイズメント ガード ポリシーの設定

IPv6 ルータ アドバタイズメント ポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 nd rguard policy** *policy-name*
4. [no]**device-role** {**host** | **monitor** | **router** | **switch**}
5. **hop-limit** {**maximum** | **minimum**} *value*
6. **managed-config-flag** {**off** | **on**}
7. **match** { **ipv6 access-list** *list* | **ra prefix-list** *list* }

8. `other-config-flag {on | off}`
9. `[no]router-preference maximum {high | medium | low}`
10. `trusted-port`
11. `default {device-role | hop-limit {maximum | minimum} | managed-config-flag | match {ipv6 access-list | ra prefix-list } | other-config-flag | router-preference maximum | trusted-port}`
12. `end`
13. `show ipv6 nd rguard policy policy_name`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例： Device> <code>enable</code>	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	<code>configure terminal</code> 例： Device# <code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<code>ipv6 nd rguard policy <i>policy-name</i></code> 例： Device(config)# <code>ipv6 nd rguard policy example_policy</code>	RA ガードポリシー名を指定し、RA ガード ポリシー コンフィギュレーション モードを開始します。
ステップ 4	<code>[no]device-role {host monitor router switch}</code> 例： Device(config-nd-rguard)# <code>device-role switch</code>	ポートに接続されているデバイスの役割を指定します。デフォルトは host です。 (注) ホスト側ポートとルータ側ポートの両方を備えたネットワークでは、ホスト側ポートまたは VLAN で device-role host を設定した RA ガードポリシーとともに、RA ガード機能が適切に動作できるように、ルータ側のポートで device-role router を設定した RA ガードポリシーを設定することが必須です。
ステップ 5	<code>hop-limit {maximum minimum} <i>value</i></code> 例： Device(config-nd-rguard)# <code>hop-limit maximum 33</code>	ホップ制限値によるルータアドバタイズメントメッセージのフィルタリングを有効にします。不正 RA メッセージは低いホップ制限値 (IPv4 の Time to Live と同じ) を持つ可能性があるため、ホストによって受け入れられると、ホストが不正 RA メッセージ ジェネレータを超えて宛先にトラフィックを生成することができなくなります。指定されていないホップ制限値を持つ RA メッセージはブロックされます。

	コマンドまたはアクション	目的
		(1~255) 最大および最小のホップ制限値の範囲。 設定されていない場合、このフィルタは無効になります。「 minimum 」を設定して、指定する値より低いホップ制限値を持つRAメッセージをブロックします。「 maximum 」を設定して、指定する値より高いホップ制限値を持つRAメッセージをブロックします。
ステップ 6	managed-config-flag {off on} 例： Device(config-nd-raguard)# managed-config-flag on	管理アドレス設定（「M」フラグ）フィールドに基づいてルータアドバタイズメントメッセージのフィルタリングを有効にします。「M」フィールドが1の不正 RA メッセージの結果としてホストが不正 DHCPv6 サーバーを使用する場合があります。設定されていない場合、このフィルタは無効になります。 On : 「M」値が1のRAメッセージを受け入れて転送し、0のものをブロックします。 Off : 「M」値が0のRAメッセージを受け入れて転送し、1のものをブロックします。
ステップ 7	match { ipv6 access-list list ra prefix-list list} 例： Device(config-nd-raguard)# match ipv6 access-list example_list	指定したプレフィックスリストまたはアクセスリストと照合します。
ステップ 8	other-config-flag {on off} 例： Device(config-nd-raguard)# other-config-flag on	その他の設定（「O」フラグ）フィールドに基づくルータアドバタイズメントメッセージのフィルタリングを有効にします。「O」フィールドが1の不正 RA メッセージの結果としてホストが不正 DHCPv6 サーバーを使用する場合があります。設定されていない場合、このフィルタは無効になります。 On : 「O」値が1のRAメッセージを受け入れて転送し、0のものをブロックします。 Off : 「O」値が0のRAメッセージを受け入れて転送し、1のものをブロックします。
ステップ 9	[no]router-preference maximum {high medium low} 例： Device(config-nd-raguard)# router-preference maximum high	「Router Preference」フラグを使用したルータアドバタイズメントメッセージのフィルタリングを有効にします。設定されていない場合、このフィルタはディセーブルになります。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • high : 「Router Preference」が「high」、 「medium」、または「low」に設定された RA メッセージを受け入れます。 • medium : 「Router Preference」が「high」に設定された RA メッセージをブロックします。 • low : 「Router Preference」が「medium」または「high」に設定された RA メッセージをブロックします。
ステップ 10	trusted-port 例 : Device(config-nd-raguard)# trusted-port	信頼できるポートとして設定すると、すべての接続デバイスが信頼され、より詳細なメッセージ検証は実行されません。
ステップ 11	default {device-role hop-limit {maximum minimum} managed-config-flag match {ipv6 access-list ra prefix-list } other-config-flag router-preference maximum trusted-port} 例 : Device(config-nd-raguard)# default hop-limit	コマンドをデフォルト値に戻します。
ステップ 12	end 例 : Device(config-nd-raguard)# end	RA ガードポリシー コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。
ステップ 13	show ipv6 nd raguard policy policy_name 例 : Device# show ipv6 nd raguard policy example_policy	(任意) ND ガードポリシーの設定を表示します。

インターフェイスへの IPv6 ルータ アドバタイズメント ガードポリシーの適用

インターフェイスまたはそのインターフェイス上の VLAN に IPv6 ルータ アドバタイズメント ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface type number**
4. **ipv6 nd raguard [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface type number 例： Device(config)# interface gigabitethernet 1/1/4	インターフェイスのタイプおよび ID 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 4	ipv6 nd rguard [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}] vlan [{vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}]] 例： Device(config-if)# ipv6 nd rguard attach-policy example_policy Device(config-if)# ipv6 nd rguard attach-policy example_policy vlan 222,223,224 Device(config-if)# ipv6 nd rguard vlan 222,223,224	ネイバー探索検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

レイヤ 2 EtherChannel インターフェイスへの IPv6 ルータ アドバタイズメント ガード ポリシーの適用

EtherChannel インターフェイスまたは VLAN に IPv6 ルータ アドバタイズメント ガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface range type number**
4. **ipv6 nd rguard [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}]]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface range type number 例： Device(config)# interface Port-channel 11	EtherChannel の作成時に割り当てられたポートチャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーションモードを開始します。 ヒント インターフェイス名やタイプを簡単に参照するには show interfaces summary コマンドを特権 EXEC モードで使用します。
ステップ 4	ipv6 nd rguard [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}] vlan [{vlan_ids add vlan_ids exceptvlan_ids none remove vlan_ids all}] 例： Device(config-if-range)# ipv6 nd rguard attach-policy example_policy Device(config-if-range)# ipv6 nd rguard attach-policy example_policy vlan 222,223,224 Device(config-if-range)# ipv6 nd rguard vlan 222,223,224	RA ガード ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例： Device(config-if-range)# end	インターフェイス範囲コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

VLAN への IPv6 ルータ アドバタイズメント ガード ポリシーのグローバル適用

インターフェイスに関係なく VLAN に IPv6 ルータ アドバタイズメント ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**

3. **vlan configuration** *vlan_list*
4. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan configuration <i>vlan_list</i> 例： Device(config)# vlan configuration 335	IPv6 RA ガードポリシーを適用する VLAN を指定し、VLAN 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	ipv6 dhcp guard [attach-policy <i>policy_name</i>] 例： Device(config-vlan-config)# ipv6 nd raguard attach-policy <i>example_policy</i>	すべてのスイッチおよびスタック インターフェイスで、IPv6 RA ガードポリシーを指定した VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルトポリシーがアタッチされます。
ステップ 5	end 例： Device(config-vlan-config)# end	VLAN 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、特権 EXEC モードに戻ります。

IPv6 DHCP ガードポリシーの設定

IPv6 DHCP (DHCPv6) ガードポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 dhcp guard policy** *policy-name*
4. **device-role** {*client* | *server*}
5. **match server access-list** *ipv6-access-list-name*
6. **match reply prefix-list** *ipv6-prefix-list-name*
7. **preference** { *max limit* | *min limit* }
8. **trusted-port**
9. **default** {*device-role* | *trusted-port*}

10. **end**
11. **show ipv6 dhcp guard policy *policy_name***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 dhcp guard policy <i>policy-name</i> 例： Device(config)# ipv6 dhcp guard policy example_policy	DHCPv6 ガードポリシー名を指定し、DHCPv6 ガードポリシー コンフィギュレーション モードを開始します。
ステップ 4	device-role {client server} 例： Device(config-dhcp-guard)# device-role server	(任意) 特定の役割のデバイスからのものではないポート上の DHCPv6 応答および DHCPv6 アドバタイズメントをフィルタします。デフォルトは client です。 <ul style="list-style-type: none"> • client : デフォルト値。適用されたデバイスがクライアントであることを指定します。サーバー メッセージにはこのポートで破棄されます。 • server : 適用されたデバイスが DHCPv6 サーバーであることを指定します。このポートでは、サーバー メッセージが許可されます。
ステップ 5	match server access-list <i>ipv6-access-list-name</i> 例： ;;Assume a preconfigured IPv6 Access List as follows: Device(config)# ipv6 access-list my_acls Device(config-ipv6-acl)# permit host 2001:BD8:::1 any ;;configure DHCPv6 Guard to match approved access list. Device(config-dhcp-guard)# match server access-list my_acls	(任意)。アドバタイズされた DHCPv6 サーバーまたはリレー アドレスが認証されたサーバーのアクセス リストからのものであることの確認を有効にします (アクセス リストの宛先アドレスは「any」です)。設定されていない場合、このチェックは回避されます。空のアクセス リストは、 permit all として処理されます。
ステップ 6	match reply prefix-list <i>ipv6-prefix-list-name</i> 例：	(任意) DHCPv6 応答メッセージ内のアドバタイズされたプレフィックスが設定された承認プレフィックス

	コマンドまたはアクション	目的
	<pre>;;Assume a preconfigured IPv6 prefix list as follows: Device(config)# ipv6 prefix-list my_prefix permit 2001:DB8::/64 le 128 ;; Configure DHCPv6 Guard to match prefix Device(config-dhcp-guard)# match reply prefix-list my_prefix</pre>	<p>リストからのものであることの確認を有効にします。設定されていない場合、このチェックは回避されます。空のプレフィクスリストは、permitとして処理されます。</p>
ステップ 7	<p>preference { max limit min limit }</p> <p>例 :</p> <pre>Device(config-dhcp-guard)# preference max 250 Device(config-dhcp-guard)# preference min 150</pre>	<p>device-role が server である場合に max および min を設定して、DHCPv6 サーバー アドバタイズメント値をサーバー優先度値に基づいてフィルタします。デフォルトではすべてのアドバタイズメントが許可されます。</p> <p>max limit : (0 ~ 255) (任意) アドバタイズされたプリファレンス ([preference] オプション内) が指定された制限未満であるかどうかの検証を有効にします。デフォルトは255です。設定されていない場合、このチェックは回避されます。</p> <p>min limit : (0 ~ 255) (任意) アドバタイズされたプリファレンス ([preference] オプション内) が指定された制限を超過しているかどうかの検証を有効にします。デフォルトは0です。設定されていない場合、このチェックは回避されます。</p>
ステップ 8	<p>trusted-port</p> <p>例 :</p> <pre>Device(config-dhcp-guard)# trusted-port</pre>	<p>(任意) trusted-port : ポートを信頼モードに設定します。このポートでは、これ以上のポリシングは実行されません。</p> <p>(注) 信頼できるポートを設定した場合、device-role オプションは使用できません。</p>
ステップ 9	<p>default {device-role trusted-port}</p> <p>例 :</p> <pre>Device(config-dhcp-guard)# default device-role</pre>	<p>(任意) default : コマンドをデフォルトに設定します。</p>
ステップ 10	<p>end</p> <p>例 :</p> <pre>Device(config-dhcp-guard)# end</pre>	<p>DHCPv6 ガード ポリシー コンフィギュレーションモードを終了して、特権EXECモードに戻ります。</p>
ステップ 11	<p>show ipv6 dhcp guard policy policy_name</p> <p>例 :</p> <pre>Device# show ipv6 dhcp guard policy example_policy</pre>	<p>(任意) IPv6 DHCP ガードポリシーの設定を表示します。<i>policy_name</i> 変数を省略すると、すべてのDHCPv6 ポリシーが表示されます。</p>

インターフェイスまたはインターフェイス上の VLAN への IPv6 DHCP ガードポリシーの適用

IPv6 バインディング テーブル コンテンツを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>type number</i> 例： Device(config)# interface gigabitethernet 1/1/4	インターフェイスのタイプおよび識別子を指定し、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 4	ipv6 dhcp guard [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] 例： Device(config-if)# ipv6 dhcp guard attach-policy example_policy Device(config-if)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224 Device(config-if)# ipv6 dhcp guard vlan 222,223,224	DHCP ガードポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例：	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

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

レイヤ 2 EtherChannel インターフェイスへの IPv6 DHCP ガードポリシーの適用

EtherChannel インターフェイスまたは VLAN に IPv6 DHCP ガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface range** *Interface_name*
4. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface range <i>Interface_name</i> 例： Device(config)# interface Port-channel 11	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーションモードを開始します。 ヒント インターフェイス名やタイプを簡単に参照するには show interfaces summary コマンドを特権 EXEC モードで使用します。
ステップ 4	ipv6 dhcp guard [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] 例： Device(config-if-range)# ipv6 dhcp guard attach-policy <i>example_policy</i>	DHCP ガードポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

VLAN への IPv6 DHCP ガードポリシーのグローバル適用

	コマンドまたはアクション	目的
	<pre>Device(config-if-range)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224 Device(config-if-range)# ipv6 dhcp guard vlan 222, 223,224</pre>	
ステップ 5	end 例 : <pre>Device(config-if-range)# end</pre>	インターフェイス範囲コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

VLAN への IPv6 DHCP ガードポリシーのグローバル適用

複数のインターフェイス上の VLAN に IPv6 DHCP のガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan configuration** *vlan_list*
4. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : <pre>Device> enable</pre>	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : <pre>Device# configure terminal</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan configuration <i>vlan_list</i> 例 : <pre>Device(config)# vlan configuration 334</pre>	IPv6 スヌーピングポリシーを適用する VLAN を指定し、VLAN 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	ipv6 dhcp guard [attach-policy <i>policy_name</i>] 例 : <pre>Device(config-vlan-config)# ipv6 dhcp guard attach-policy example_policy</pre>	すべてのスイッチおよびスタックインターフェイスで、IPv6 ネイバー探索ポリシーを指定した VLAN にアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。デフォルト ポリシーは、device-role client 、 no trusted-port です。

	コマンドまたはアクション	目的
ステップ 5	end 例 : Device(config-vlan-config)# end	VLAN 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、特権 EXEC モードに戻ります。

IPv6 ソース ガードの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 source-guard policy *policy_name***
4. **[deny global-autoconf] [permit link-local] [default{...}] [exit] [no{...}]**
5. **end**
6. **show ipv6 source-guard policy *policy_name***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 source-guard policy <i>policy_name</i> 例 : Device(config)# ipv6 source-guard policy example_policy	IPv6 ソース ガード ポリシー名を指定し、IPv6 ソース ガード ポリシー コンフィギュレーション モードを開始します。
ステップ 4	[deny global-autoconf] [permit link-local] [default{...}] [exit] [no{...}] 例 : Device(config-sisf-sourceguard)# deny global-autoconf	(任意) IPv6 ソース ガード ポリシーを定義します。 <ul style="list-style-type: none"> • deny global-autoconf : 自動設定されたグローバルアドレスからのデータトラフィックを拒否します。これは、リンク上のすべてのグローバルアドレスが DHCP によって割り当てられている際に、管理者が、自己設定されたアドレスを持つホストによるトラフィックの送信をブロックしたい場合に役立ちます。

■ インターフェイスへの IPv6 ソースガードポリシーの適用

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • permit link-local : リンクローカルアドレスから送信されたすべてのデータトラフィックを許可します。 <p>(注) ソースガードポリシーでは trusted オプションはサポートされません。</p>
ステップ 5	end 例 : Device(config-sisf-sourceguard)# end	IPv6 ソースガードポリシー コンフィギュレーションモードを終了して、特権 EXEC モードに戻ります。
ステップ 6	show ipv6 source-guard policy <i>policy_name</i> 例 : Device# show ipv6 source-guard policy example_policy	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

次のタスク

インターフェイスに IPv6 ソースガードポリシーを適用します。

インターフェイスへの IPv6 ソースガードポリシーの適用

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *type number***
4. **ipv6 source-guard [attach-policy <policy_name>]**
5. **end**
6. **show ipv6 source-guard policy *policy_name***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	interface <i>type number</i> 例： Device(config)# interface gigabitethernet 1/1/4	インターフェイスのタイプおよび識別子を指定し、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 4	ipv6 source-guard [attach-policy < <i>policy_name</i> >] 例： Device(config-if)# ipv6 source-guard attach-policy example_policy	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例： Device(config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show ipv6 source-guard policy <i>policy_name</i> 例： Device#(config) # show ipv6 source-guard policy example_policy	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

レイヤ 2 EtherChannel インターフェイスへの IPv6 ソースガードポリシーの適用

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number*
4. **ipv6 source-guard** [**attach-policy** <*policy_name*>]
5. **end**
6. **show ipv6 source-guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	interface port-channel <i>port-channel-number</i> 例： Device(config)# interface Port-channel 4	インターフェイスのタイプとポート番号を指定し、スイッチをポート チャネル コンフィギュレーション モードにします。
ステップ 4	ipv6 source-guard [attach-policy <i><policy_name></i>] 例： Device(config-if)# ipv6 source-guard attach-policy example_policy	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show ipv6 source-guard policy <i>policy_name</i> 例： Device# show ipv6 source-guard policy example_policy	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 プレフィックス ガードの設定



(注) プレフィックスガードが適用されている場合にリンクローカルアドレスから送信されたルーティングプロトコル制御パケットを許可するには、ソースガード ポリシー コンフィギュレーション モードで **permit link-local** コマンドを有効にします。

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 source-guard policy** *source-guard-policy*
4. **validate address**
5. **validate prefix**
6. **exit**
7. **show ipv6 source-guard policy** [*source-guard-policy*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ipv6 source-guard policy <i>source-guard-policy</i> 例： Device(config)# ipv6 source-guard policy my_snooping_policy	IPv6 ソースガードポリシー名を定義して、スイッチ統合セキュリティ機能のソースガードポリシー コンフィギュレーション モードを開始します。
ステップ 4	validate address 例： Device(config-sisf-sourceguard)# no validate address	アドレス検証機能を無効にし、IPv6 プレフィックスガード機能を設定できるようにします。
ステップ 5	validate prefix 例： Device(config-sisf-sourceguard)# validate prefix	IPv6 プレフィックスガード動作を実行するよう、IPv6 ソースガードを有効にします。
ステップ 6	exit 例： Device(config-sisf-sourceguard)# exit	スイッチ統合セキュリティ機能のソースガードポリシー コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 7	show ipv6 source-guard policy [<i>source-guard-policy</i>] 例： Device# show ipv6 source-guard policy policy1	IPv6 ソースガードポリシー設定を表示します。

インターフェイスへの IPv6 プレフィックス ガード ポリシーの適用

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *type number***
4. **ipv6 source-guard attach-policy *policy_name***
5. **end**
6. **show ipv6 source-guard policy *policy_name***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。

レイヤ 2 EtherChannel インターフェイスへの IPv6 プレフィックス ガード ポリシーの適用

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface type number 例： Device(config)# interface gigabitethernet 1/1/4	インターフェイスのタイプおよび識別子を指定し、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 4	ipv6 source-guard attach-policy policy_name 例： Device(config-if)# ipv6 source-guard attach-policy example_policy	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show ipv6 source-guard policy policy_name 例： Device(config-if)# show ipv6 source-guard policy example_policy	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

レイヤ 2 EtherChannel インターフェイスへの IPv6 プレフィックス ガード ポリシーの適用

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel port-channel-number**
4. **ipv6 source-guard [attach-policy <policy_name>]**
5. **end**
6. **show ipv6 source-guard policy policy_name**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	Device# configure terminal	
ステップ 3	interface port-channel <i>port-channel-number</i> 例： Device(config)# interface Port-channel 4	インターフェイスのタイプとポート番号を指定し、スイッチをポート チャネル コンフィギュレーション モードにします。
ステップ 4	ipv6 source-guard [attach-policy < <i>policy_name</i> >] 例： Device(config-if)# ipv6 source-guard attach-policy example_policy	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。 attach-policy オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show ipv6 source-guard policy <i>policy_name</i> 例： Device(config)# show ipv6 source-guard policy example_policy	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 ファースト ホップ セキュリティの設定例

例：IPv6 DHCP ガードポリシーの設定

DHCPv6 ガード設定の例

```

Device> enable
Device# configure terminal
Device(config)# ipv6 access-list acl1
Device(config-ipv6-acl)# permit host 2001:DB8:0000:
0000:0000:0000:0000:0001 any
Device(config-ipv6-acl)# exit
Device(config)# ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
Device(config)# ipv6 dhcp guard policy poll
Device(config-dhcp-guard)# device-role server
Device(config-dhcp-guard)# match server access-list acl1
Device(config-dhcp-guard)# match reply prefix-list abc
Device(config-dhcp-guard)# preference min 0
Device(config-dhcp-guard)# preference max 255
Device(config-dhcp-guard)# trusted-port
Device(config-dhcp-guard)# exit
Device(config)# interface GigabitEthernet 0/2/0
Device(config-if)# switchport
Device(config-if)# ipv6 dhcp guard attach-policy poll vlan add 1
Device(config-if)# exit
Device(config)# vlan 1

```

例：レイヤ 2 EtherChannel インターフェイスへの IPv6 ソースガードポリシーの適用

```
Device(config-vlan)# ipv6 dhcp guard attach-policy pol1
Device(config-vlan)# end
```

例：レイヤ 2 EtherChannel インターフェイスへの IPv6 ソースガードポリシーの適用

次の例は、IPv6 ソース ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# ipv6 source-guard policy POL
Device(config-sisf-sourceguard)# validate address
Device(config-sisf-sourceguard)# exit
Device(config)# interface Port-Channel 4
Device(config-if)# ipv6 snooping
Device(config-if)# ipv6 source-guard attach-policy POL
Device(config-if)# end
Device#
```

例：レイヤ 2 EtherChannel インターフェイスへの IPv6 プレフィックスガードポリシーの適用

次の例は、IPv6 プレフィックス ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法を示しています。

```
Device> enable
Device# configure terminal
Device(config)# ipv6 source-guard policy POL
Device (config-sisf-sourceguard)# no validate address
Device((config-sisf-sourceguard)# validate prefix
Device(config-sisf-sourceguard)# exit
Device(config)# interface Po4
Device(config-if)# ipv6 snooping
Device(config-if)# ipv6 source-guard attach-policy POL

Device(config-if)# end
```

IPv6 ファースト ホップ セキュリティに関する追加情報

関連資料

関連項目	マニュアルタイトル
SISF	『セキュリティ コンフィギュレーションガイド』の「SISF ベースのデバイスラッキングの設定」

シスコのテクニカル サポート

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

IPv6 ファースト ホップ セキュリティの機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	IPv6 ファースト ホップ セキュリティ	<p>IPv6 のファースト ホップ セキュリティは、ポリシーを物理インターフェイス、EtherChannel インターフェイス、または VLAN に適用できる一連の IPv6 セキュリティ機能です。IPv6 ソフトウェア ポリシー データベースサービスは、これらのポリシーを保存しアクセスします。ポリシーを設定または変更すると、ポリシー属性はソフトウェアポリシー データベースに保存または更新され、その後指定したとおりに適用されます。</p> <p>IPv6 スヌーピングポリシー機能は廃止されました。コマンドは CLI に表示され、設定できますが、代わりにスイッチ統合セキュリティ機能 (SISF) ベースのデバイス追跡機能を使用することを推奨します。</p>

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



第 22 章

SISF ベースのデバイストラッキングの設定

- [SISF ベースのデバイストラッキングに関する情報 \(479 ページ\)](#)
- [SISF ベースのデバイストラッキングの設定方法 \(483 ページ\)](#)
- [SISF ベースのデバイストラッキングの設定例 \(494 ページ\)](#)
- [SISF の機能履歴 \(500 ページ\)](#)

SISF ベースのデバイストラッキングに関する情報

SISF ベースのデバイストラッキングの概要

スイッチ統合セキュリティ機能ベース (SISF ベース) のデバイストラッキング機能は、一連のファーストホップセキュリティ機能の一部です。

この機能の主な役割は、ネットワーク内のエンドノードの存在、ロケーション、移動を追跡することです。SISF は、スイッチが受信したトラフィックをスヌーピングし、デバイスアイデンティティ (MAC と IP アドレス) を抽出して、バインディングテーブルに保存します。Cisco TrustSec、IEEE 802.1X、LISP、web 認証などの多くの機能は、この情報の正確性に依存して正常に動作します。

SISF ベースのデバイストラッキングは、IPv4 と IPv6 の両方をサポートします。

SISF ベースのデバイストラッキングが導入されても、レガシーデバイストラッキング CLI (IP デバイストラッキング (IPDT) および IPv6 スヌーピング CLI) は引き続き使用できます。スイッチをブートアップすると、使用可能なコマンドのセットは既存の設定によって異なり、次のいずれかのみが使用可能です。

- SISF ベースのデバイストラッキング CLI、または
- IPDT および IPv6 スヌーピング CLI



(注) IPDT および IPv6 スヌーピングコマンドは廃止されましたが、引き続き使用できます。SISF ベースのデバイストラッキングにアップグレードすることを推奨します。

IPDT および IPv6 スヌーピング CLI を使用していて、SISF ベースのデバイストラッキングに移行する場合、詳細については「レガシー IPDT と IPv6 スヌーピングから SISF ベースのデバイストラッキングへの移行」を参照してください。

SISF ベースのデバイストラッキングは、手動で (**device-tracking** コマンドを使用して)、またはプログラムで (デバイストラッキング サービスを他の機能に提供する場合に) 有効にできます。

SISF ベースのデバイストラッキングを有効にするオプション

デフォルトでは、SISF ベースのデバイストラッキングは無効になっています。

これを有効にするには、デバイストラッキングポリシーを定義し、そのポリシーを特定のターゲットに適用します。



(注) ターゲットは、インターフェイスまたは VLAN です。

SISF ベースのデバイストラッキングの手動による有効化

- **オプション 1** : **default** デバイストラッキングポリシーをターゲットに適用します。

インターフェイス コンフィギュレーション モードまたは VLAN コンフィギュレーション モードで、**device-tracking** コマンドを入力します。次に、システムは **default** ポリシーをインターフェイスまたは VLAN に適用します。



(注) **default** ポリシーは、デフォルト設定の組み込みポリシーです。**default** ポリシーの属性は変更できません。デバイストラッキングポリシーの属性を設定できるようにするには、カスタムポリシーを作成する必要があります。「オプション 2 : カスタム設定でカスタムポリシーを作成します」を参照してください。

- **オプション 2** : カスタム設定でカスタムポリシーを作成します。

グローバル コンフィギュレーション モードで **device-tracking policy** コマンドを入力し、続けてカスタムポリシー名を入力します。システムにより、指定した名前のポリシーが作成されます。その後、デバイストラッキング コンフィギュレーション モード (**config-device-tracking**) で使用可能な設定を行い、指定したターゲットにポリシーを適用できます。

プログラムによる SISF ベースのデバイストラッキングの有効化

一部の機能はデバイストラッキングに依存し、SISF ベースのデバイストラッキングが構築および維持するバインディングエントリの信頼性のあるデータベースを利用します。これらの機能

は、デバイストラッキングクライアントとも呼ばれ、プログラムによりデバイストラッキングを有効にします（デバイストラッキングポリシーを作成して適用します）。



- (注) ここでの例外は、IEEE 802.1X、web 認証、Cisco TrustSec、IP ソースガード (IPSG) です。これらもデバイストラッキングに依存しますが、有効にはしません。これらのデバイストラッキングクライアントでは、**ip dhcp snooping vlan vlan** コマンドを入力して、プログラムにより特定のターゲットでデバイストラッキングを有効にする必要があります。

プログラムによる SISF ベースのデバイストラッキングの有効化については、次の点に注意してください。

- デバイストラッキングクライアントでは、デバイストラッキングを有効にする必要があります。

複数のデバイストラッキングクライアントが存在するため、複数のプログラムポリシーを作成できます。各ポリシーの設定は、ポリシーを作成するデバイストラッキングクライアントによって異なります。

- 作成されるポリシーとその設定はシステム定義です。

設定可能なポリシー属性は、デバイストラッキングコンフィギュレーションモード (**config-device-tracking**) で使用でき、リリースごとに異なります。設定不可能な属性を変更しようとする、設定変更は拒否され、エラーメッセージが表示されます。

プログラムで作成されたポリシーの詳細については、[Cisco IOS XE Fuji 16.9.x 以降のリリースでの SISF ベースのデバイストラッキングのプログラムによる有効化 \(491 ページ\)](#) を参照してください。

レガシー IPDT と IPv6 スヌーピングから SISF ベースのデバイストラッキングへの移行

デバイスにあるレガシー設定に基づいて、**device-tracking upgrade-cli** コマンドは CLI を異なる方法でアップグレードします。既存の設定を移行する前に、次の設定シナリオ、および対応する移行結果を検討します。



- (注) 古い IPDT と IPv6 スヌーピング CLI を SISF ベースのデバイストラッキング CLI と併用することはできません。

IPDT 設定のみが存在する

デバイスに IPDT 設定のみがある場合は、**device-tracking upgrade-cli** コマンドを実行すると、設定が変換され、新しく作成されてインターフェイスで適用される SISF ポリシーが使用されます。これにより、この SISF ポリシーを更新できます。

引き続きレガシーコマンドを使用する場合、レガシーモードでの操作に制限されます。このモードでは、レガシー IPDT と IPv6 スヌーピングコマンドのみがデバイスで使用可能になります。

IPv6 スヌーピング設定のみが存在する

既存の IPv6 スヌーピング設定があるデバイスで、古い IPv6 スヌーピングコマンドを以降の設定に使用できます。次のオプションを使用できます。

- (推奨) **device-tracking upgrade-cli** コマンドを使用して、レガシー設定をすべて、新しい SISF ベースのデバイストラッキング コマンドに変換します。変換後は、新しいデバイストラッキング コマンドのみがデバイスで動作します。
- レガシー IPv6 スヌーピングコマンドを今後の設定に使用し、**device-tracking upgrade-cli** コマンドは実行しません。このオプションでは、デバイスで使用可能なのはレガシー IPv6 スヌーピングコマンドのみであり、新しい SISF ベースのデバイストラッキング CLI コマンドは使用できません。

IPDT と IPv6 スヌーピングの両方の設定が存在する

レガシー IPDT 設定と IPv6 スヌーピング設定の両方が存在するデバイスでは、レガシーコマンドを SISF ベースのデバイストラッキング CLI コマンドに変換できます。ただし、インターフェイスに適用することができるスヌーピングポリシーは 1 つだけであり、IPv6 スヌーピング ポリシー パラメータは IPDT 設定よりも優先される、ということに注意してください。



- (注) 新しい SISF ベースのコマンドに移行しておらず、レガシー IPv6 スヌーピングや IPDT コマンドを使用し続けている場合、IPv4 デバイストラッキング設定情報が IPv6 スヌーピングコマンドに表示される可能性があります。SISF ベースのデバイストラッキング機能では、IPv4 と IPv6 の両方の設定を扱うためです。これを回避するには、レガシー設定を SISF ベースのデバイストラッキング コマンドに変換することを推奨します。

IPDT または IPv6 スヌーピング設定が存在しない

デバイスにレガシー IP デバイストラッキング設定も IPv6 スヌーピング設定もない場合は、今後の設定に使用できるのは新しい SISF ベースのデバイストラッキング コマンドのみです。レガシー IPDT コマンドと IPv6 スヌーピングコマンドは使用できません。

SISF ベースのデバイストラッキングの設定方法

SISF ベースのデバイストラッキングの手動による有効化

ターゲットへのデフォルト デバイス トラッキング ポリシーの適用

デフォルトのデバイス トラッキング ポリシーをインターフェイスまたは VLAN に適用するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. インターフェイスまたは VLAN を指定します。
 - **interface** *interface*
 - **vlan configuration** *vlan_list*
4. **device-tracking**
5. **end**
6. **show device-tracking policy** *policy-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	インターフェイスまたは VLAN を指定します。 • interface <i>interface</i> • vlan configuration <i>vlan_list</i> 例： Device(config)# interface gigabitethernet 1/1/4 OR Device(config)# vlan configuration 333	interface <i>type number</i> : 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。デバイストラッキングポリシーは、指定されたインターフェイスに適用されます。 vlan configuration <i>vlan_list</i> : VLAN を指定し、VLAN 機能コンフィギュレーションモードを開始します。デバイス トラッキング ポリシーは、指定された VLAN に適用されます。

カスタム設定を使用したカスタム デバイストラッキング ポリシーの作成

	コマンドまたはアクション	目的
ステップ 4	device-tracking 例 : Device(config-if)# device-tracking OR Device(config-vlan-config)# device-tracking	SISF ベースのデバイストラッキングを有効にし、デフォルトポリシーをインターフェイスまたは VLAN に適用します。 デフォルトポリシーは、デフォルト設定の組み込みポリシーです。デフォルトポリシーの属性は変更できません。
ステップ 5	end 例 : Device(config-if)# end OR Device(config-vlan-config)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 VLAN 機能コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show device-tracking policy policy-name 例 : Device# show device-tracking policy default	デバイストラッキングポリシーの設定と、それが適用されるすべてのターゲットを表示します。

カスタム設定を使用したカスタム デバイストラッキング ポリシーの作成

デバイストラッキングポリシーを作成して設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **[no] device-tracking policy policy-name**
4. **[data-glean | default | destination-glean | device-role | distribution-switch | exit | limit | no | prefix-glean | protocol | security-level | tracking | trusted-port | vpc]**
5. **end**
6. **show device-tracking policy policy-name**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	<p>[no] device-tracking policy <i>policy-name</i></p> <p>例 :</p> <pre>Device(config)# device-tracking policy example_policy</pre>	<p>ポリシーを作成し、デバイストラッキング コンフィギュレーション モードを開始します。</p>
ステップ 4	<p>[data-glean default destination-glean device-role distribution-switch exit limit no prefix-glean protocol security-level tracking trusted-port vpc]</p> <p>例 :</p> <pre>Device(config-device-tracking)# destination-glean log-only</pre>	<p>システムプロンプトに疑問符 (?) を入力すると、このモードで使用できるオプションのリストが表示されます。IPv4 と IPv6 の両方に対して以下を設定できます。</p> <ul style="list-style-type: none"> • (任意) data-glean : ネットワーク内の送信元からスヌーピングされたデータパケットからのアドレスの学習を有効にし、データトラフィックの送信元アドレスとともにバインディングテーブルを読み込みます。次のいずれかのオプションを入力します。 <ul style="list-style-type: none"> • log-only : データパケット通知時に syslog メッセージを生成します。 • recovery : プロトコルを使用してバインディングテーブルの回復を有効にします。 NDP または DHCP の入力。 • (任意) default : ポリシー属性をデフォルト値に設定します。次のポリシー属性をデフォルト値に設定できます。 data-glean、destination-glean、device-role、limit、prefix-glean、protocol、security-level、tracking、trusted-port。 • (任意) destination-glean : データトラフィックの宛先アドレスを収集して、バインディングテーブルを読み込みます。次のいずれかのオプションを入力します。 <ul style="list-style-type: none"> • log-only : データパケット通知時に syslog メッセージを生成します。 • recovery : プロトコルを使用してバインディングテーブルの回復を有効にします。 DHCP を入力します。 • (任意) device-role : ポートに接続されているデバイスのロールを設定します。ノードまたはスイッチを指定できます。次のいずれかのオプションを入力します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • node : 接続されているデバイスをノードとして設定します。これがデフォルトのオプションです。 • switch : 接続されているデバイスをスイッチとして設定します。 • (任意) distribution-switch : このオプションは CLI には表示されますが、サポートされていません。行った設定は有効になりません。 • exit : デバイストラッキング ポリシー コンフィギュレーション モードを終了します。 • limit address-count : ポートごとのアドレスカウント制限を指定します。有効な範囲は1～32000です。 • no : コマンドを無効にするか、デフォルト値を設定します。 • (任意) prefix-glean : IPv6 ルータアダプタイズメントまたは DHCP-PD のどちらかからのプレフィックスの学習を有効にします。次のオプションがあります。 <ul style="list-style-type: none"> • (任意) only : プレフィックスのみを収集し、ホストアドレスは収集しません。 • (任意) protocol : 収集するプロトコルを設定します。デフォルトでは、すべて収集されます。次のいずれかのオプションを入力します。 <ul style="list-style-type: none"> • arp [prefix-list name] : ARP パケットのアドレスを収集します。必要に応じて、照合するプレフィックスリストの名前を入力します。 • dhcp4 [prefix-list name] : DHCPv4 パケットのアドレスを収集します。必要に応じて、照合するプレフィックスリストの名前を入力します。 • dhcp6 [prefix-list name] : DHCPv6 パケットのアドレスを収集します。必要に応じて、照合するプレフィックスリストの名前を入力します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • ndp [prefix-list name] : NDP パケットのアドレスを収集します。必要に応じて、照合するプレフィックスリストの名前を入力します。 • udp [prefix-list name] : このオプションは CLI には表示されますが、サポートされていません。行った設定は有効になりません。 • (任意) security-level : この機能によって適用されるセキュリティのレベルを指定します。次のいずれかのオプションを入力します。 <ul style="list-style-type: none"> • glean : アドレスをパッシブに収集します。 • guard : 不正なメッセージを検査してドロップします。これはデフォルトです。 • inspect : メッセージを収集して検証します。 • (任意) tracking : トラッキングオプションを指定します。次のいずれかのオプションを入力します。 <ul style="list-style-type: none"> • disable [stale-lifetime [1-86400-seconds infinite]] : デバイストラッキングをオフにします。 必要に応じて、エントリを削除するまで非アクティブにする期間を入力することも、永続的に非アクティブにすることもできます。 • enable [reachable-lifetime [1-86400-seconds infinite]] : デバイストラッキングをオンにします。 必要に応じて、エントリを到達可能にする期間を入力することも、永続的に到達可能にすることもできます。 • (任意) trusted-port : 信頼できるポートを設定します。該当するターゲットに対するガードがディセーブルになります。信頼できるポートを経由して学習されたバインディングは、他のどのポートを経由して学習されたバインディング

	コマンドまたはアクション	目的
		<p>よりも優先されます。テーブル内にエントリを作成しているときに衝突が発生した場合、信頼できるポートが優先されます。</p> <ul style="list-style-type: none"> • (任意) vpc : このオプションは CLI には表示されますが、サポートされていません。行った設定は有効になりません。
ステップ 5	end 例 : Device(config-device-tracking)# end	デバイストラッキングコンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show device-tracking policy policy-name 例 : Device# show device-tracking policy example_policy	デバイストラッキングポリシー設定を表示します。

次のタスク

ポリシーをインターフェイスまたは VLAN に適用します。

デバイストラッキングポリシーのインターフェイスへの適用

デバイストラッキングポリシーをインターフェイスにアタッチするには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface**
4. **device-tracking attach-policy policy name**
5. **end**
6. **show device-tracking policies [interface interface]**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	interface <i>interface</i> 例： Device(config-if)# interface gigabitethernet 1/1/4	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 4	device-tracking attach-policy <i>policy name</i> 例： Device(config-if)# device-tracking attach-policy example_policy	インターフェイスにデバイストラッキングポリシーを適用します。デバイストラッキングは、EtherChannel でもサポートされます。 (注) SISF ベースのデバイストラッキングポリシーは、カスタムポリシーである場合にのみ無効にできます。プログラムによって作成されたポリシーは、対応するデバイストラッキングクライアント機能の設定が削除された場合にのみ削除できます。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show device-tracking policies [interface <i>interface</i>] 例： Device# show device-tracking policies interface gigabitethernet 1/1/4	指定されたインターフェイスの種類と番号に一致するポリシーを表示します。

デバイストラッキングポリシーの VLAN への適用

複数のインターフェイスでデバイストラッキングポリシーを VLAN にアタッチするには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan configuration** *vlan_list*
4. **device-tracking attach-policy** *policy_name*
5. **do show device-tracking policies vlan** *vlan-ID*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan configuration <i>vlan_list</i> 例： Device(config)# vlan configuration 333	デバイストラッキングポリシーを適用する VLAN を指定し、その VLAN インターフェイス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 4	device-tracking attach-policy <i>policy_name</i> 例： Device(config-vlan-config)# device-tracking attach-policy example_policy	すべてのスイッチインターフェイスで、デバイストラッキングポリシーを指定された VLAN にアタッチします。 (注) SISF ベースのデバイストラッキングポリシーは、カスタムポリシーである場合にのみ無効にできます。プログラムによって作成されたポリシーは、対応するデバイストラッキングクライアント機能の設定が削除された場合にのみ削除できます。
ステップ 5	do show device-tracking policies vlan <i>vlan-ID</i> 例： Device(config-vlan-config)# do show device-tracking policies vlan 333	VLAN 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ないで、ポリシーが指定された VLAN に割り当てられていることを確認します。
ステップ 6	end 例： Device(config-vlan-config)# end	VLAN 機能コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

Cisco IOS XE Fuji 16.9.x 以降のリリースでの SISF ベースのデバイストラッキングのプログラムによる有効化

表 27: Cisco IOS XE Fuji 16.9.x 以降のリリースでの SISF ベースのデバイストラッキングのプログラムによる有効化

SISF ベースのデバイストラッキングを有効にできるデバイストラッキングクライアント機能	<p>Cisco IOS XE Fuji 16.9.x 以降のリリースでは、次の機能について SISF ベースのデバイストラッキングをプログラムで有効にできます。</p> <ul style="list-style-type: none"> • IEEE 802.1X、web 認証、Cisco TrustSec、IPSG 機能 : ip dhcp snooping vlan vlan コマンドを入力します。 • Cisco Locator/ID Separation Protocol (LISP)。 • EVPN on VLAN <p>(注) プログラムによって作成されたポリシーが複数ある場合は、優先順位が最も高いポリシーが有効になります。</p>
ポリシー名	<ul style="list-style-type: none"> • IEEE 802.1X、web 認証、Cisco TrustSec、および IPSG 機能は、ポリシー <code>DT-PROGRAMMATIC</code> を使用します。 • LISP 機能は、<code>LISP-DT-GUARD-VLAN</code> または <code>LISP-DT-GLEAN-VLAN</code> を作成します。 • EVPN on VLAN 機能は <code>evpn-sisf-policy</code> を作成します <p>設定のリストは、各プログラムポリシーによって異なります。詳細については、例を参照してください。</p>

ユーザー オプション	<ul style="list-style-type: none"> • ポリシーの優先順位がサポートされます。優先順位は、ポリシーの作成方法によって決まります。手動で作成されたポリシーが最も優先されます。これにより、プログラムで生成されたポリシーとは異なるポリシー設定を適用できます。 • 複数のポリシーを同じ VLAN に適用できます。 • 優先順位が異なる複数のポリシーが同じ VLAN に適用されている場合、優先順位が最も高いポリシーの設定が有効になります。ここでの例外は limit address-count for IPv4 per mac と limit address-count for IPv6 per mac の設定です。優先順位が最も低いポリシーの設定が有効になります。 • デバイストラッキングクライアント機能の設定が削除されない限り、ポリシーは削除できません。 • ポリシー属性は変更できません。 • MAC ごとのアドレスカウント制限は変更できません。これは limit address-count for IPv4 per mac および limit address-count for IPv6 per mac コマンドに該当します。 • VLAN のポリシー設定を変更するには、カスタマイズされたデバイストラッキング ポリシーを作成し、VLAN に適用します。 • デバイストラッキング ポリシーが VLAN のインターフェイスに適用されると、インターフェイスのポリシー設定が VLAN のポリシー設定よりも優先されます。ここでの例外は、limit address-count for IPv4 per mac と limit address-count for IPv6 per mac の値で、インターフェイスと VLAN の両方のポリシーから集約されます。
------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

トランクポートからのバインディングエントリの作成を停止するためのマルチスイッチネットワークの設定

マルチスイッチネットワークでは、SISF ベースのデバイストラッキングにより、機能を実行しているスイッチ間でバインディング テーブル エントリを分散できます。バインディング エントリは、ホストがアクセスポートに表示されるスイッチでのみ作成されます。トランクポート経由で表示されるホストのエントリは作成されません。これは、**trusted-port** および **device-role switch** オプションを使用してポリシーを設定し、トランクポートに適用することで実現されます。



重要 ポリシーで、**trusted-port** および **device-role switch** オプションの両方を設定する必要があります。

さらに、SISF ベースのデバイストラッキングが有効になっているデバイス側のポートに、このようなポリシーを適用することを推奨します。

次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **device-tracking policy** *policy-name*
4. **device-role switch**
5. **trusted-port**
6. **exit**
7. **interface** *interface*
8. **device-tracking attach-policy** *policy-name*
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	device-tracking policy <i>policy-name</i> 例： Device(config)# device-tracking policy example_trusted_policy	指定されたポリシーのデバイストラッキング ポリシー コンフィギュレーションモードを開始します。
ステップ 4	device-role switch 例： Device(config-device-tracking)# device-role switch	ポートに接続されているデバイスのロールを指定します。デフォルトは node です。ポートのバインディングエントリの作成を停止する device-role switch オプションを入力します。
ステップ 5	trusted-port 例： Device(config-device-tracking)# trusted-port	信頼できるポートを設定します。該当するターゲットに対するガードがディセーブルになります。信頼できるポートを経由して学習されたバインディング

	コマンドまたはアクション	目的
		は、他のどのポートを経由して学習されたバインディングよりも優先されます。テーブル内にエントリを作成しているときに衝突が発生した場合、信頼できるポートが優先されます。
ステップ 6	exit 例： Device(config-device-tracking)# exit	デバイストラッキング ポリシー コンフィギュレーションモードを終了して、グローバルコンフィギュレーションモードに戻ります。
ステップ 7	interface interface 例： Device(config)# interface gigabitethernet 1/0/25	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 8	device-tracking attach-policy policy-name 例： Device(config-if)# device-tracking attach-policy example_trusted_policy	デバイス トラッキング ポリシーをインターフェイスまたはそのインターフェイス上で指定されたVLAN にアタッチします。
ステップ 9	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

SISF ベースのデバイストラッキングの設定例

次の例は、デバイストラッキングの設定例と、特定の状況で推奨される、または関連するその他の設定を示しています。

例：Cisco IOS XE Everest 16.9.x 以降のリリースでの SISF ベースのデバイストラッキングのプログラムによる有効化

この出力例は、プログラムによって作成されたポリシーのさまざまな設定を示しています。

デバイストラッキングクライアント：VLAN での LISP

LISP を設定したら、特権 EXEC モードで **show device-tracking policy** コマンドを入力して、作成された LISP-DT-GUARD-VLAN ポリシーと対応する設定を表示します。

```
Device> enable
Device# show device-tracking policy LISP-DT-GUARD-VLAN
```

```
Policy LISP-DT-GUARD-VLAN configuration:
  security-level guard (*)
  device-role node
  gleaning from Neighbor Discovery
```

```

gleaning from DHCP
gleaning from ARP
gleaning from DHCP4
NOT gleaning from protocol unkn
limit address-count for IPv4 per mac 4 (*)
limit address-count for IPv6 per mac 12 (*)
tracking enable
Policy LISP-DT-GUARD-VLAN is applied on the following targets:
Target      Type      Policy      Feature      Target range
vlan 10     VLAN     LISP-DT-GUARD-VLAN     Device-tracking     vlan all
note:
Binding entry Down timer: 10 minutes (*)
Binding entry Stale timer: 30 minutes (*)

```

デバイストラッキングクライアント : VLAN での LISP

LISP を設定したら、特権 EXEC モードで **show device-tracking policy** コマンドを入力して、作成された LISP-DT-GLEAN-VLAN ポリシーと対応する設定を表示します。

```
Device# show device-tracking policy LISP-DT-GLEAN-VLAN
```

```

Policy LISP-DT-GLEAN-VLAN configuration:
security-level glean (*)
device-role node
gleaning from Neighbor Discovery
gleaning from DHCP
gleaning from ARP
gleaning from DHCP4
NOT gleaning from protocol unkn
limit address-count for IPv4 per mac 4 (*)
limit address-count for IPv6 per mac 12 (*)
tracking enable
Policy LISP-DT-GUARD-VLAN is applied on the following targets:
Target      Type      Policy      Feature      Target range
vlan 10     VLAN     LISP-DT-GLEAN-VLAN     Device-tracking     vlan all
note:
Binding entry Down timer: 10 minutes (*)
Binding entry Stale timer: 30 minutes (*)

```

デバイストラッキングクライアント : VLAN での EVPN

EVPN を設定した後、特権 EXEC モードで **show device-tracking policy** コマンドを入力して、作成された evpn-sisf-policy ポリシーとポリシーに応じて行った設定を表示します。

```
Device# show device-tracking policy evpn-sisf-policy
```

```

Policy evpn-sisf-policy configuration:
security-level glean (*)
device-role node
gleaning from Neighbor Discovery
gleaning from DHCP
gleaning from ARP
gleaning from DHCP4
NOT gleaning from protocol unkn
tracking enable
Policy evpn-sisf-policy is applied on the following targets:
Target      Type      Policy      Feature      Target range
vlan 10     VLAN     evpn-sisf-policy     Device-tracking     vlan all
note:

```

例 : Cisco IOS XE Everest 16.9.x 以降のリリースでの SISF ベースのデバイストラッキングのプログラムによる有効化

```
Binding entry Down timer: 24 hours (*)
Binding entry Stale timer: 24 hours (*)
```

デバイストラッキングクライアント : IEEE 802.1X、web 認証、Cisco TrustSec、IPSG

グローバルコンフィギュレーションモードで **ip dhcp snooping vlan** *vlan* コマンドを設定して、IEEE 802.1X、web 認証、Cisco TrustSec、IPSG 機能のデバイストラッキングを有効にします。特権 EXEC モードで **show device-tracking policy** コマンドを入力し、作成された DT-PROGRAMMATIC ポリシーとポリシーに応じて行った設定を表示します。

```
Device> enable
Device# configure terminal
Device(config)# ip dhcp snooping vlan 10
Device(config)# end
Device# show device-tracking policy DT-PROGRAMMATIC

Policy DT-PROGRAMMATIC configuration:
  security-level glean (*)
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count for IPv4 per mac 1 (*)
  tracking enable
Policy DT-PROGRAMMATIC is applied on the following targets:
Target      Type      Policy          Feature          Target range
vlan 10     VLAN     DT-PROGRAMMATIC Device-tracking  vlan all

note:
Binding entry Down timer: 24 hours (*)
Binding entry Stale timer: 24 hours (*)
```

ターゲットに複数のポリシーが適用されている場合のアクティブポリシーの識別

この例では、複数のポリシーが同じ VLAN に適用されている場合にアクティブポリシーを指定する方法を示します。

この例では、2つのポリシーが VLAN 10 に適用されており、LISP-DT-GUARD-VLAN がアクティブポリシーです。

```
Device# show device-tracking policies

Target      Type      Policy          Feature          Target range
vlan 10     VLAN     DT-PROGRAMMATIC Device-tracking  vlan all
vlan 10     VLAN     LISP-DT-GUARD-VLAN Device-tracking  vlan all

Device# show device-tracking capture-policy vlan 10

HW Target vlan 10 HW policy signature 0001DF9F policies#:2 rules 14 sig 0001DF9F
SW policy DT-PROGRAMMATIC feature Device-tracking -

SW policy LISP-DT-GUARD-VLAN feature Device-tracking - Active
```

例：ターゲットでの IPv6 デバイストラッキングの無効化

デフォルトで、SISF ベースのデバイストラッキングは IPv4 と IPv6 の両方をサポートします。次の設定例は、サポートされている場合に IPv6 デバイストラッキングを無効にする方法を示しています。

カスタムポリシーがターゲットに適用されている場合に、IPv6 のデバイストラッキングを無効にする（すべてのリリース）：

```
Device(config)# device-tracking policy example-policy
Device(config-device-tracking)# no protocol ndp
Device(config-device-tracking)# no protocol dhcp6
Device(config-device-tracking)# end
```

プログラムポリシーがターゲットに適用されている場合に、IPv6 のデバイストラッキングを無効にする（Cisco IOS XE Everest 16.6.x および Cisco IOS XE Fuji 16.8.x のみ）：

```
Device(config)# device-tracking policy DT-PROGRAMMATIC
Device(config-device-tracking)# no protocol ndp
Device(config-device-tracking)# no protocol dhcp6
Device(config-device-tracking)# end
```



- (注)
- Cisco IOS XE Everest 16.5.x リリースでは、プログラムポリシーが適用されている場合、IPv6 のデバイストラッキングを無効にすることはできません。
 - Cisco IOS XE Everest 16.6.x および Cisco IOS XE Fuji 16.8.x では、プログラムポリシーが適用されている場合、上の例に示すように、IPv6 のデバイストラッキングを無効にすることができます。
 - Cisco IOS XE Fuji 16.9.x 以降では、プログラムポリシーの設定を変更できません。

例：VLAN 上の SVI に対する IPv6 の有効化（重複アドレスの問題を軽減するため）

ネットワークで IPv6 が有効になっており、VLAN 上でスイッチ仮想インターフェイス（SVI）が設定されている場合は、SVI 設定に次の内容を追加することを推奨します。これにより、SVI はリンクローカルアドレスを自動的に取得できます。このアドレスは SISF プロローブの送信元 IP アドレスとして使用されるため、重複 IP アドレスの問題を防止できます。

```
Device> enable
Device# configure terminal
Device(config)# interface vlan 10
Device(config-if)# ipv6 enable
Device(config-if)# end
```

例：IPv4 重複アドレスの問題の緩和

次に、Microsoft Windows を実行しているクライアントによって発生した重複 IP アドレス 0.0.0.0 エラーメッセージの問題に対応する例を示します。

device-tracking tracking auto-source コマンドをグローバル コンフィギュレーション モードで設定します。このコマンドは、デバイストラッキング テーブル内のエントリを維持するために、スイッチがクライアントをプローブするよう送信するアドレス解決パケット (ARP) 要求で使用される送信元 IP および MAC アドレスを決定します。その目的は、送信元 IP アドレスとして 0.0.0.0 を使用しないようにすることです。



- (注) スイッチ仮想インターフェイス (SVI) が設定されていない場合に、**device-tracking tracking auto-source** コマンドを設定します。SVI が VLAN で IPv4 アドレスを使用して設定されている場合は、設定する必要はありません。

コマンド	アクション (デバイストラッキング ARP プローブの送信元 IP および MAC アドレスを選択するた め)	注記
device-tracking tracking auto-source	<ul style="list-style-type: none"> • 存在する場合、VLAN SVI に送信元を設定します。 • 同じサブネットからデバイストラッキング テーブルで IP および MAC バインディングを検索します。 • 0.0.0.0 を使用します 	MAC フラッピングを回避するために、すべてのトランクポートでデバイストラッキングを無効にすることを推奨します。
device-tracking tracking auto-source override	<ul style="list-style-type: none"> • 存在する場合、VLAN SVI に送信元を設定します。 • 0.0.0.0 を使用します。 	SVI がない場合は推奨しません。

コマンド	アクション (デバイストラッキング ARP プローブの送信元 IP および MAC アドレスを選択するた め)	注記
ip device tracking probe auto-source fallback 0.0.0.X 255.255.255.0	<ul style="list-style-type: none"> • 存在する場合、VLAN SVI に送信元を設定します。 • 同じサブネットからデバイストラッキング テーブルで IP および MAC バインディングを検索します。 • 提供されたホストビットとマスクを使用して、クライアント IP から送信元 IP を計算します。送信元 MAC は、クライアント側のスイッチポートの MAC アドレスから取得されます*。 	<p>MAC フラッピングを回避するために、すべてのトランクポートでデバイストラッキングを無効にすることを推奨します。</p> <p>計算された IPv4 アドレスは、クライアントまたはネットワークデバイスに割り当ててすることはできません。</p>
device-tracking tracking auto-source fallback 0.0.0.X 255.255.255.0 override	<ul style="list-style-type: none"> • 存在する場合、VLAN SVI に送信元を設定します。 <p>提供されたホストビットとマスクを使用して、クライアント IP から送信元 IP を計算します*。送信元 MAC は、クライアント側のスイッチポートの MAC アドレスから取得されます*。</p>	

* クライアント IP アドレスによっては、IPv4 アドレスを送信元 IP 用に予約する必要があります。

予約済み送信元 IPv4 アドレス = (host-ip and mask) | client-ip

- クライアント IP = 192.0.2.25
- 送信元 IP = (192.0.2.25 and 255.255.255.0) | (0.0.0.1) = 192.0.2.1

IP アドレス 192.0.2.1 をクライアントまたはネットワークデバイスに割り当てないでください。

例：短いデバイストラッキング バインディング到達可能時間の回避

以前のリリースから移行する場合、次の設定が存在している可能性があります。

```
device-tracking binding reachable-time 10
```

コマンドの **no** バージョンを入力して、これを削除します。

```
Device> enable
Device# configure terminal
Device(config)# no device-tracking binding reachable-time 10
Device(config)# end
```

SISF の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	SISF ベースのデバイストラッキング	<p>この機能が導入されました。</p> <p>SISF ベースのデバイストラッキングは、ネットワーク内のエンドノードの存在、ロケーション、移動を追跡します。この機能は、スイッチが受信したトラフィックをスヌーピングし、デバイスアイデンティティ (MAC と IP アドレス) を抽出して、バインディングテーブルに保存します。(デバイストラッキング クライアントと呼ばれる) その他の機能の適切な動作は、この情報の正確性に依存します。</p> <p>IPv4 および IPv6 のどちらもサポートされています。</p> <p>デフォルトでは、SISF ベースのデバイストラッキングは無効になっています。</p>
Cisco IOS XE Everest 16.6.1	DT_PROGRAMMATIC のパラメータを変更するオプション	<p>このリリース以降、プログラムで作成されたデバイストラッキング ポリシー (DT_PROGRAMMATIC) の特定の設定をデバイストラッキング コンフィギュレーションモード (config-device-tracking) で変更できます。</p>

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.1	ポリシーの優先順位 追加のデバイス追跡 クライアント プログラムで作成されたポリシーの変更	<p>ポリシーの優先順位のサポートが導入されました。優先順位は、ポリシーの作成方法によって決まります。手動で作成されたポリシーが最も優先されます。これにより、プログラムで生成されたポリシーとは異なるポリシー設定を適用できます。</p> <p>デバイストラッキングクライアント機能が追加されました。プログラムで作成されるポリシーは、デバイストラッキングクライアントごとに異なります。</p> <p>任意のプログラムで作成されるポリシーのパラメータを変更するオプションは廃止されました。</p>

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator にアクセスするには、<https://cfmng.cisco.com> に進みます。



第 23 章

IEEE 802.1x ポートベースの認証の設定

この章では、IEEE 802.1x ポートベース認証を設定する方法について説明します。IEEE 802.1x 認証は、不正なデバイス（クライアント）によるネットワークアクセスを防止します。別途記載のないかぎり、スイッチという用語はスタンドアロンスイッチまたはスイッチスタックを意味します。

- [IEEE 802.1x ポートベース認証の制約事項 \(503 ページ\)](#)
- [IEEE 802.1x ポートベースの認証に関する情報 \(504 ページ\)](#)
- [802.1x ポートベース認証の設定方法 \(543 ページ\)](#)
- [IEEE 802.1x ポートベースの認証の設定例 \(596 ページ\)](#)
- [IEEE 802.1x ポートベースの認証統計情報とステータスのモニタリング \(597 ページ\)](#)
- [IEEE 802.1x ポートベースの認証の機能履歴 \(598 ページ\)](#)

IEEE 802.1x ポートベース認証の制約事項

- プライベート VLAN で使用する場合、スイッチポートは常に許可されません。認証、許可、およびアカウントिंग（AAA）サーバーからプッシュされるダイナミック VLAN は、プライベート VLAN ポートではサポートされません。データ クライアントセッションは、プライベート VLAN の dot1x ポートのセカンダリ VLAN で許可されることが期待されます。
- 通常のアクセス VLAN ポートでは、インターフェイスで設定されたプライベート VLAN ベースの許可とダイナミック VLAN だけがサポートされます。
- **dot1q tag vlan native** コマンドがグローバルレベルで設定されている場合、トランクポートでの dot1x 再認証は失敗します。
- 認証の失敗を引き起こす可能性があるため、音声 VLAN とアクセス VLAN の両方に同じ VLAN ID を同時に設定しないでください。
- 管理 VRF は、RADIUS の送信元インターフェイスとして使用できません。
- ダウンロード可能な ACL に重複するエントリが含まれている場合、エントリは自動的にマージされません。その結果、802.1Xセッション許可は失敗します。ダウンロード可能な

ACLが、同じポートのポートベースのエントリや名前ベースのエントリなど、重複するエントリなしで最適化されていることを確認します。

- ポートセキュリティは、IEEE 802.1x ポートベース認証ではサポートされていません。
- インターフェイスの実行中の設定を、フラッシュにロードされた設定ファイルで上書きすると、一部のポートがエンドポイントの認証に失敗する場合があります。

IEEE 802.1x ポートベースの認証に関する情報

802.1x 規格では、一般の人がアクセス可能なポートから不正なクライアントが LAN に接続しないように規制する（適切に認証されている場合を除く）、クライアント/サーバ型のアクセスコントロールおよび認証プロトコルを定めています。認証サーバーがスイッチポートに接続する各クライアントを認証したうえで、スイッチまたは LAN が提供するサービスを利用できるようにします。



(注) TACACS は、802.1x 認証ではサポートされていません。

802.1x アクセスコントロールでは、クライアントを認証するまでの間、そのクライアントが接続しているポート経由では Extensible Authentication Protocol over LAN (EAPOL)、Cisco Discovery Protocol、およびスパンニングツリープロトコル (STP) トラフィックしか許可されません。認証に成功すると、通常のトラフィックはポートを通過できるようになります。

次の表は、各クライアントがサポートするセッションの最大数を示しています。

クライアントセッション	サポートされる最大セッション数
dot1x または MAB クライアントセッションの最大数	2000
Web ベース認証セッションの最大数	2000
クリティカル認証 VLAN を有効にしてサーバを再初期化した dot1x セッションの最大数	2000
さまざまなセッション機能が適用される MAB セッションの最大数	2000
サービス テンプレートまたはセッション機能が適用される dot1x セッションの最大数	2000

ポートベース認証プロセス

IEEE 802.1X ポートベース認証を設定するには、認証、認可、およびアカウントिंग (AAA) を有効にし、認証方式リストを指定する必要があります。方式リストは、ユーザ認証のためにクエリー送信を行う手順と認証方式を記述したものです。

AAA プロセスは認証から始まります。802.1x ポートベース認証がイネーブルであり、クライアントが 802.1x 準拠のクライアント ソフトウェアをサポートしている場合、次のイベントが発生します。

- クライアント ID が有効で 802.1x 認証に成功した場合、スイッチはクライアントにネットワークへのアクセスを許可します。
- EAPOL メッセージ交換の待機中に 802.1x 認証がタイムアウトし、MAC 認証バイパスがイネーブルの場合、スイッチはクライアント MAC アドレスを認証用に使用します。このクライアント MAC アドレスが有効で認証に成功した場合、スイッチはクライアントにネットワークへのアクセスを許可します。クライアント MAC アドレスが無効で認証に失敗した場合、ゲスト VLAN が設定されていれば、スイッチはクライアントに限定的なサービスを提供するゲスト VLAN を割り当てます。
- スイッチが 802.1x 対応クライアントから無効な ID を取得し、制限付き VLAN が指定されている場合、スイッチはクライアントに限定的なサービスを提供する制限付き VLAN を割り当てることができます。
- RADIUS 認証サーバーが使用できず (ダウンしていて) アクセスできない認証バイパスがイネーブルの場合、スイッチは、RADIUS 設定 VLAN またはユーザー指定アクセス VLAN で、ポートをクリティカル認証ステートにして、クライアントにネットワークのアクセスを許可します。

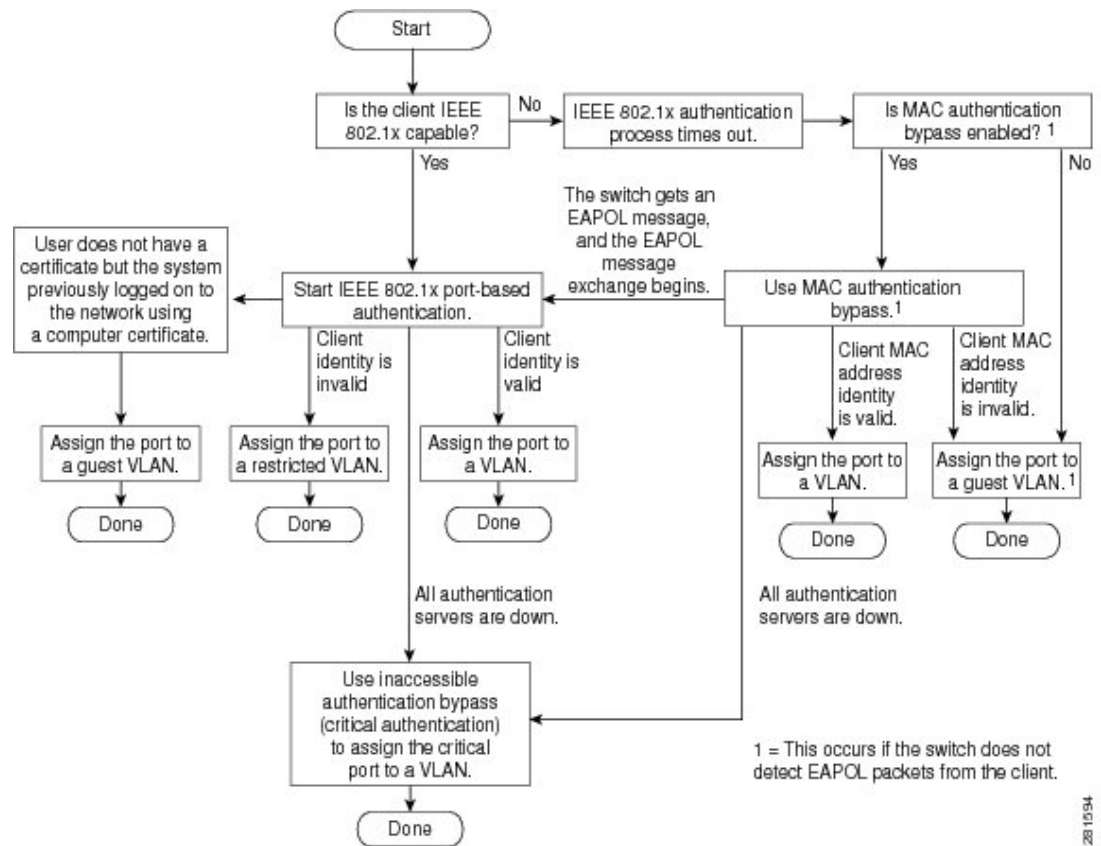


(注) アクセスできない認証バイパスは、クリティカル認証、または AAA 失敗ポリシーとも呼ばれます。

ポートで Multi Domain Authentication (MDA) が有効になっている場合、音声許可に該当する例外をいくつか伴ったフローを使用できます。

図 23: 認証フローチャート

次の図は認証プロセスを示します。



次の状況のいずれかが発生すると、スイッチはクライアントを再認証します。

- 定期的な再認証がイネーブルで、再認証タイマーの期限が切れている場合。

スイッチ固有の値を使用するか、RADIUS サーバーからの値に基づいて再認証タイマーを設定できます。

RADIUS サーバーを使用した 802.1x 認証の後で、スイッチは Session-Timeout RADIUS 属性 (Attribute[27])、および Termination-Action RADIUS 属性 (Attribute[29]) に基づいてタイマーを使用します。

Session-Timeout RADIUS 属性 (Attribute[27]) には再認証が行われるまでの時間を指定します。指定できる範囲は 1 ~ 65535 秒です。

Termination-Action RADIUS 属性 (Attribute[29]) には、再認証中に行われるアクションを指定します。アクションは *Initialize* および *ReAuthenticate* に設定できます。アクションに *Initialize* (属性値は *DEFAULT*) を設定した場合、802.1x セッションは終了し、認証中、接続は失われます。アクションに *ReAuthenticate* (属性値は *RADIUS-Request*) を設定した場合、セッションは再認証による影響を受けません。

- クライアントを手動で再認証するには、**dot1x re-authenticate interface interface-id** 特権 EXEC コマンドを入力します。

ポートベース認証の開始およびメッセージ交換

802.1x 認証中に、スイッチまたはクライアントは認証を開始できます。**authentication port-control auto** インターフェイス コンフィギュレーション コマンドを使用してポート上で認証をイネーブルにすると、スイッチは、リンクステートがダウンからアップに移行したときに認証を開始し、ポートがアップしていて認証されていない場合は定期的に認証を開始します。スイッチはクライアントに EAP-Request/Identity フレームを送信し、その ID を要求します。クライアントはフレームを受信すると、EAP-Response/Identity フレームで応答します。

ただし、クライアントが起動時にスイッチからの EAP-Request/Identity フレームを受信しなかった場合、クライアントは EAPOL-Start フレームを送信して認証を開始できます。このフレームはスイッチに対し、クライアントの識別情報を要求するように指示します。



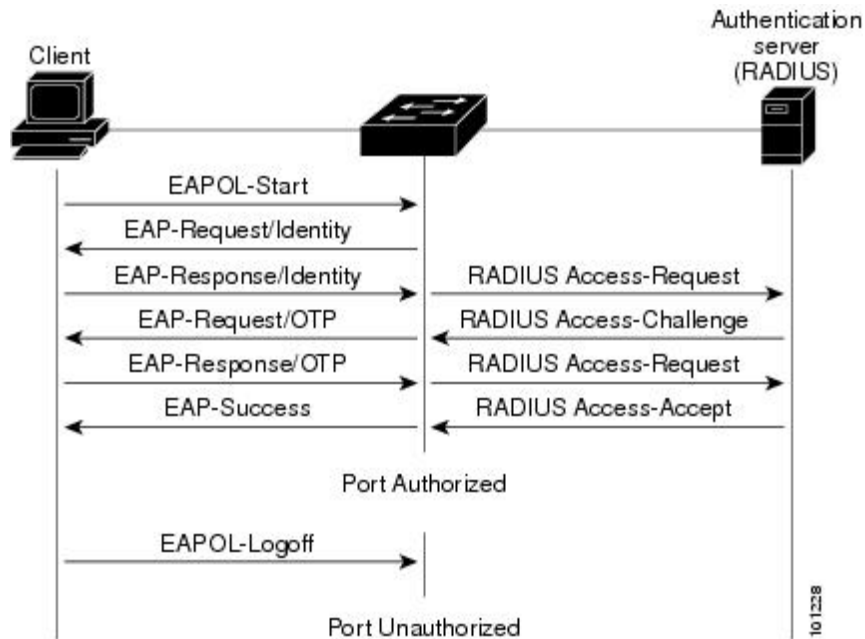
- (注) ネットワークアクセスデバイスで 802.1x 認証がイネーブルに設定されていない、またはサポートされていない場合には、クライアントからの EAPOL フレームはすべて廃棄されます。クライアントが認証の開始を 3 回試みても EAP-Request/Identity フレームを受信しなかった場合、クライアントはポートが許可ステートであるものとしてフレームを送信します。ポートが許可ステートであるということは、クライアントの認証が成功したことを実質的に意味します。

クライアントが自らの識別情報を提示すると、スイッチは仲介デバイスとしての役割を開始し、認証が成功または失敗するまで、クライアントと認証サーバーの間で EAP フレームを送受信します。認証が成功すると、スイッチポートは許可ステートになります。認証に失敗した場合、認証が再試行されるか、ポートが限定的なサービスを提供する VLAN に割り当てられるか、あるいはネットワーク アクセスが許可されないかのいずれかになります。

実際に行われる EAP フレーム交換は、使用する認証方式によって異なります。

図 24: メッセージ交換

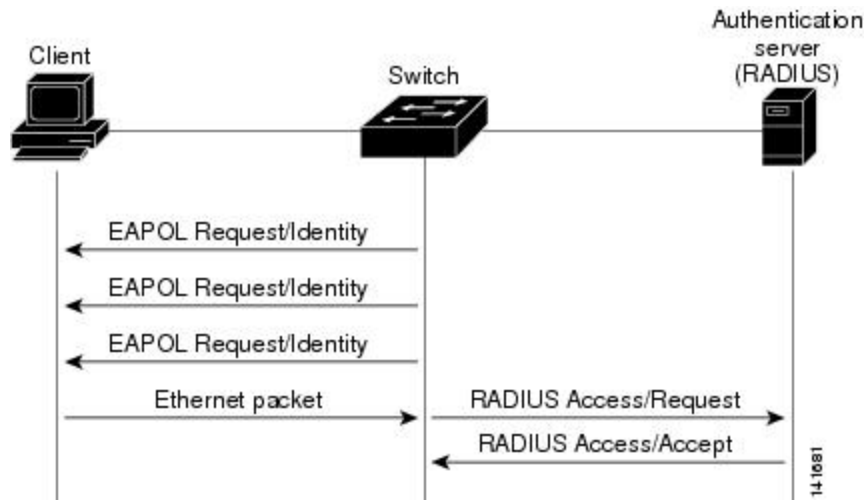
次の図に、クライアントが RADIUS サーバとの間で OTP (ワンタイムパスワード) 認証方式を使用する際に行われるメッセージ交換を示します。



EAPOL メッセージ交換の待機中に 802.1x 認証がタイムアウトし、MAC 認証バイパスがイネーブルの場合、スイッチはクライアントからイーサネットパケットを検出するとそのクライアントを認証できません。スイッチは、クライアントの MAC アドレスを ID として使用し、RADIUS サーバーに送信される RADIUS Access/Request フレームにこの情報を保存します。サーバーがスイッチに RADIUS Access/Accept フレームを送信（認証が成功）すると、ポートが許可されます。認証に失敗してゲスト VLAN が指定されている場合、スイッチはポートをゲスト VLAN に割り当てます。イーサネットパケットの待機中にスイッチが EAPOL パケットを検出すると、スイッチは MAC 認証バイパスプロセスを停止して、802.1x 認証を開始します。

図 25: MAC 認証バイパス中のメッセージ交換

次の図に、MAC 認証バイパス中のメッセージ交換を示します。



ポートベース認証方法

表 28: 802.1x 機能

認証方法	モード			
	シングルホスト	マルチホスト	MDA	複数
802.1x	VLAN 割り当て ユーザ単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	VLAN 割り当て	VLAN 割り当て ユーザ単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	VL ユー Filt ダウ AC リタ
MAC 認証バイパス	VLAN 割り当て ユーザ単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	VLAN 割り当て	VLAN 割り当て ユーザ単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	VL ユー Filt ダウ AC リタ
スタンドアロン Web 認証	プロキシ ACL、Filter-ID 属性、ダウンロード可能な ACL			

認証方法	モード			
	シングル ホスト	マルチ ホスト	MDA	複数認
NAC レイヤ 2 IP 検証	Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	Filter-ID ダウンロード可能 ACL リダイ
フォールバック方式としての Web 認証	プロキシ ACL Filter-ID 属性 ダウンロード可能 ACL	プロキシ ACL Filter-ID 属性 ダウンロード可能 ACL	プロキシ ACL Filter-ID 属性 ダウンロード可能 ACL	プロキシ Filter-ID ダウンロード可能 ACL

⁶ Cisco IOS リリース 12.2(50)SE 以降でサポートされています。

⁷ 802.1x 認証をサポートしないクライアント用。

ユーザー単位 ACL および Filter-Id



(注) Filter-Id としてロールベース ACL を使用することは推奨されません。

MDA 対応ポートおよびマルチ認証ポートでは、複数のホストを認証できます。ホストに適用される ACL ポリシーは、別のホストのトラフィックには影響を与えません。マルチ ホストポートで認証されるホストが1つだけで、他のホストが認証なしでネットワークアクセスを取得する場合、発信元アドレスに any を指定することで、最初のホストの ACL ポリシーを他の接続ホストに適用できます。

許可ステートおよび無許可ステートのポート

802.1x 認証中に、スイッチのポートステートによって、スイッチはネットワークへのクライアントアクセスを許可します。ポートは最初、無許可ステートです。このステートでは、音声 VLAN ポートとして設定されていないポートは 802.1x 認証、Cisco Discovery Protocol、および STP パケットを除くすべての入力および出力トラフィックを禁止します。クライアントの認証が成功すると、ポートは許可ステートに変更し、クライアントのトラフィック送受信を通常どおりに許可します。ポートが音声 VLAN ポートとして設定されている場合、VoIP トラフィックおよび 802.1x プロトコルパケットが許可された後クライアントが正常に認証されます。



(注) Cisco Discovery Protocol バイパスはサポートされていないため、ポートが err-disabled ステートになる場合があります。

802.1x をサポートしていないクライアントが、無許可ステートの 802.1x ポートに接続すると、スイッチはそのクライアントの識別情報を要求します。この状況では、クライアントは要求に 응답せず、ポートは引き続き無許可ステートとなり、クライアントはネットワークアクセスを許可されません。

反対に、802.1x 対応のクライアントが、802.1x 標準が稼働していないポートに接続すると、クライアントは EAPOL-Start フレームを送信して認証プロセスを開始します。応答がなければ、クライアントは同じ要求を所定の回数だけ送信します。応答がないので、クライアントはポートが許可ステートであるものとしてフレーム送信を開始します。

authentication port-control インターフェイス コンフィギュレーション コマンドおよび次のキーワードを使用して、ポートの許可ステートを制御できます。

- **force-authorized** : 802.1x 認証をディセーブルにし、認証情報の交換を必要とせず、ポートを許可ステートに変更します。ポートはクライアントとの 802.1x ベース認証を行わずに、通常のトラフィックを送受信します。これがデフォルト設定です。
- **force-unauthorized** : ポートが無許可ステートのままになり、クライアントからの認証の試みをすべて無視します。スイッチはポートを介してクライアントに認証サービスを提供できません。
- **auto** : 802.1x 認証をイネーブルにします。ポートは最初、無許可ステートであり、ポート経由で送受信できるのは EAPOL フレームだけです。ポートのリンクステートがダウンからアップに変更した際、または EAPOL-Start フレームを受信した際に、認証プロセスが開始されます。スイッチはクライアントの識別情報を要求し、クライアントと認証サーバとの間で認証メッセージのリレーを開始します。スイッチはクライアントの MAC アドレスを使用して、ネットワークアクセスを試みる各クライアントを一意に識別します。

クライアントが認証に成功すると（認証サーバから **Accept** フレームを受信すると）、ポートが許可ステートに変わり、認証されたクライアントからの全フレームがポート経由での送受信を許可されます。認証が失敗すると、ポートは無許可ステートのままですが、認証を再試行することはできません。認証サーバに到達できない場合、スイッチは要求を再送信します。所定の回数だけ試行してもサーバから応答が得られない場合には、認証が失敗し、ネットワークアクセスは許可されません。

クライアントはログオフするとき、EAPOL-Logoff メッセージを送信します。このメッセージによって、スイッチポートが無許可ステートになります。

ポートのリンクステートがアップからダウンに変更した場合、または EAPOL-Logoff フレームを受信した場合に、ポートは無許可ステートに戻ります。

802.1X のホストモード

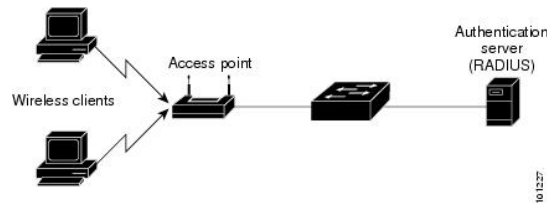
802.1x ポートは、シングルホストモードまたはマルチホストモードで設定できます。シングルホストモードでは、802.1x 対応のスイッチポートに接続できるのはクライアント 1 つだけです。スイッチは、ポートのリンクステートがアップに変化したときに、EAPOL フレームを送信することでクライアントを検出します。クライアントがログオフしたとき、または別のク

クライアントに代わったときには、スイッチはポートのリンクステートをダウンに変更し、ポートは無許可ステータスに戻ります。

マルチホストモードでは、複数のホストを単一の 802.1x 対応ポートに接続できます。このモードでは、接続されたクライアントのうち 1 つが許可されれば、クライアントすべてのネットワークアクセスが許可されます。ポートが無許可ステータスになると（再認証が失敗するか、または EAPOL-Logoff メッセージを受信した場合）、スイッチは接続しているクライアントのネットワークアクセスをすべて禁止します。

このトポロジでは、ワイヤレスアクセスポイントが接続しているクライアントの認証を処理し、スイッチに対してクライアントとしての役割を果たします。

図 26: マルチホストモードの例



- (注) すべてのホストモードで、ポートベース認証が設定されている場合、ラインプロトコルは許可の前にアップのままです。

スイッチはマルチドメイン認証 (MDA) をサポートしています。これにより、データ装置と IP Phone などの音声装置の両方を同じスイッチポートに接続できます。

MAC 移動

あるスイッチポートで MAC アドレスが認証されると、そのアドレスは同じスイッチの別の認証マネージャ対応ポートでは許可されません。スイッチが同じ MAC アドレスを別の認証マネージャ対応ポートで検出すると、そのアドレスは許可されなくなります。

場合によっては、MAC アドレスを同じスイッチ上のポート間で移動する必要があります。たとえば、認証ホストとスイッチポート間に別のデバイス（ハブまたは IP Phone など）がある場合、ホストをデバイスから接続して、同じスイッチの別のポートに直接接続する必要があります。

デバイスが新しいポートで再認証されるように、MAC 移動をグローバルにイネーブルにできます。ホストが別のポートに移動すると、最初のポートのセッションが削除され、ホストは新しいポートで再認証されます。MAC 移動はすべてのホストモードでサポートされます（認証ホストは、ポートでイネーブルにされているホストモードに関係なく、スイッチの任意のポートに移動できます）。MAC アドレスがあるポートから別のポートに移動すると、スイッチは元のポートで認証済みセッションを終了し、新しいポートで新しい認証シーケンスを開始します。MAC 移動の機能は、音声およびデータホストの両方に適用されます。



- (注) オープン認証モードでは、MACアドレスは、新しいポートでの許可を必要とせずに、元のポートから新しいポートへただちに移動します。

MAC 置換

MAC 置換機能は、ホストが、別のホストがすでに認証済みであるポートに接続しようとする
と発生する違反に対処するように設定できます。



- (注) 違反はマルチ認証モードでは発生しないため、マルチ認証モードのポートにこの機能は適用されません。マルチホストモードで認証が必要なのは最初のホストだけなので、この機能はこのモードのポートには適用されません。

replace キーワードを指定して **authentication violation** インターフェイスコンフィギュレーションコマンドを設定すると、マルチドメインモードのポートでの認証プロセスは、次のようになります。

- 既存の認証済みMACアドレスを使用するポートで新しいMACアドレスが受信されます。
- 認証マネージャは、ポート上の現在のデータホストのMACアドレスを、新しいMACアドレスで置き換えます。
- 認証マネージャは、新しいMACアドレスに対する認証プロセスを開始します。
- 認証マネージャによって新しいホストが音声ホストであると判断された場合、元の音声ホストは削除されます。

ポートがオープン認証モードになっている場合、MACアドレスはただちにMACアドレステーブルに追加されます。

802.1x アカウンティング

802.1x 標準では、ユーザの認証およびユーザのネットワークアクセスに対する許可方法を定義しています。ただし、ネットワークの使用法についてはトラッキングしません。802.1x アカウンティングは、デフォルトでディセーブルです。802.1x アカウンティングをイネーブルにすると、次の処理を 802.1x 対応のポート上でモニタできます。

- 正常にユーザを認証します。
- ユーザがログ オフします。
- リンクダウンが発生します。
- 再認証が正常に行われます。
- 再認証が失敗します。

スイッチは802.1xアカウンティング情報を記録しません。その代わりに、スイッチはこの情報をRADIUSサーバに送信します。RADIUSサーバは、アカウンティングメッセージを記録するように設定する必要があります。

802.1x アカウンティング属性値ペア

RADIUSサーバに送信された情報は、属性値（AV）ペアの形式で表示されます。これらのAVペアのデータは、各種アプリケーションによって使用されます（たとえば課金アプリケーションの場合、RADIUSパケットのAcct-Input-OctetsまたはAcct-Output-Octets属性の情報が必要です）。

AVペアは、802.1xアカウンティングが設定されているスイッチによって自動的に送信されます。次の種類のRADIUSアカウンティングパケットがスイッチによって送信されます。

- START：新規ユーザーセッションが始まると送信されます。
- INTERIM：既存のセッションが更新されると送信されます。
- STOP：セッションが終了すると送信されます。

次の表に、AVペアおよびスイッチによって送信されるAVペアの条件を示します。

表 29: アカウンティング AV ペア

属性番号	AV ペア名	START	INTERIM	STOP
属性 [1]	User-Name	送信	送信	送信
属性 [4]	NAS-IP-Address	送信	送信	送信
属性 [5]	NAS-Port	送信	送信	送信
属性 [8]	Framed-IP-Address	非送信	条件に応じて送信 ⁸	条件に応じて送信
属性 [30]	Called-Station-ID	送信	送信	送信
属性 [31]	Calling-Station-ID	送信	送信	送信
属性 [40]	Acct-Status-Type	送信	送信	送信
属性 [41]	Acct-Delay-Time	送信	送信	送信
属性 [42]	Acct-Input-Octets	非送信	送信	送信
属性 [43]	Acct-Output-Octets	非送信	送信	送信
属性 [47]	Acct-Input-Packets	非送信	送信	送信
属性 [48]	Acct-Output-Packets	非送信	送信	送信
属性 [44]	Acct-Session-ID	送信	送信	送信

属性番号	AV ペア名	START	INTERIM	STOP
属性 [45]	Acct-Authentic	送信	送信	送信
属性 [46]	Acct-Session-Time	非送信	送信	送信
属性 [49]	Acct-Terminate-Cause	非送信	非送信	送信
属性 [61]	NAS-Port-Type	送信	送信	送信

⁸ 有効な静的 IP アドレスが設定されているか、ホストに対する Dynamic Host Control Protocol (DHCP) バインディングが DHCP スヌーピング バインディング テーブルに存在している場合に、Framed-IP-Address の AV ペアが送信されます。

802.1x 準備状態チェック

802.1x 準備状態チェックは、すべてのスイッチ ポートの 802.1x アクティビティをモニタリングし、802.1x をサポートするポートに接続されているデバイスの情報を表示します。この機能を使用して、スイッチ ポートに接続されているデバイスが 802.1x に対応できるかどうかを判別できます。802.1x 機能をサポートしていないデバイスでは、MAC 認証バイパスまたは Web 認証などの代替認証を使用します。

この機能が有用なのは、クライアントのサブリカントで NOTIFY EAP 通知パケットでのクエリーがサポートされている場合だけです。クライアントは、802.1x タイムアウト値内に応答しなければなりません。

スイッチと RADIUS サーバー間の通信

RADIUS セキュリティサーバは、ホスト名または IP アドレス、ホスト名と特定の UDP ポート番号、または IP アドレスと特定の UDP ポート番号によって識別します。IP アドレスと UDP ポート番号の組み合わせによって、一意の ID が作成され、同一 IP アドレスのサーバ上にある複数の UDP ポートに RADIUS 要求を送信できるようになります。同じ RADIUS サーバ上の異なる 2 つのホスト エントリに同じサービス（たとえば認証）を設定した場合、2 番めに設定されたホスト エントリは、最初に設定されたホスト エントリのフェールオーバー バックアップとして動作します。RADIUS ホスト エントリは、設定した順序に従って試行されます。

IEEE 802.1x 認証

802.1X 認証

802.1x 認証を設定する場合の注意事項は、次のとおりです。

- 802.1x 認証を使用するには、SISF ベースのデバイストラッキングを有効にする必要があります。デフォルトでは、SISF ベースのデバイス トラッキングはスイッチで無効になっています。

- 802.1x 認証をイネーブルにすると、他のレイヤ 2 またはレイヤ 3 機能がイネーブルになる前に、ポートが認証されます。
- 802.1x 対応ポートが割り当てられている VLAN が変更された場合、この変更は透過的でスイッチには影響しません。たとえば、ポートが RADIUS サーバに割り当てられた VLAN に割り当てられ、再認証後に別の VLAN に割り当てられた場合に、この変更が発生します。

802.1x ポートが割り当てられている VLAN がシャットダウン、ディセーブル、または削除される場合、ポートは無許可になります。たとえば、ポートが割り当てられたアクセス VLAN がシャットダウンまたは削除された後、ポートは無許可になります。

- 802.1x プロトコルは、レイヤ 2 スタティックアクセスポート、音声 VLAN ポート、およびレイヤ 3 ルーテッドポートでサポートされますが、次のポートタイプではサポートされません。
 - ダイナミックポート：ダイナミックモードのポートは、ネイバーとトランクポートへの変更をネゴシエートする場合があります。ダイナミックポートで 802.1x 認証をイネーブルにしようとする、エラーメッセージが表示され、802.1x 認証はイネーブルになりません。802.1x 対応ポートのモードをダイナミックに変更しようとしても、エラーメッセージが表示され、ポートモードは変更されません。
 - EtherChannel ポート：アクティブまたはアクティブでない EtherChannel メンバを 802.1x ポートとして設定しないでください。EtherChannel ポートで 802.1x 認証をイネーブルにしようとする、エラーメッセージが表示され、802.1x 認証はイネーブルになりません。
 - スイッチドポートアナライザ (SPAN) およびリモート SPAN (RSPAN) 宛先ポート：SPAN または RSPAN 宛先ポートであるポートの 802.1x 認証をイネーブルにすることができます。ただし、ポートを SPAN または RSPAN 宛先ポートとして削除するまでは、802.1x 認証はディセーブルになります。SPAN または RSPAN 送信元ポートでは 802.1x 認証をイネーブルにすることができます。
- スイッチ上で、**dot1x system-auth-control** グローバルコンフィギュレーションコマンドを入力して 802.1x 認証をグローバルにイネーブルにする前に、802.1x 認証と EtherChannel が設定されているインターフェイスから、EtherChannel の設定を削除してください。
- 802.1x 認証に関連するシステムメッセージをフィルタリングできます。



(注) 802.1x に準拠したすべての CLI を同じインターフェイスまたは同じテンプレートで設定することを推奨します。

ポートベース認証マネージャ CLI コマンド

認証マネージャ インターフェイス コンフィギュレーション コマンドは、802.1x、MAC 認証バイパスおよび Web 認証など、すべての認証方法を制御します。認証マネージャ コマンドは、接続ホストに適用される認証方法のプライオリティと順序を決定します。

認証マネージャ コマンドは、ホスト モード、違反モードおよび認証タイマーなど、一般的な認証機能を制御します。一般的な認証コマンドには、**authentication host-mode**、**authentication violation**、および **authentication timer** インターフェイス コンフィギュレーション コマンドが含まれます。

802.1x 専用コマンドは、先頭に **dot1x** キーワードが付きます。たとえば、**authentication port-control auto** インターフェイス コンフィギュレーション コマンドは、インターフェイスでの認証をイネーブルにします。

スイッチでの **dot1x** を無効にするには、**no dot1x system-auth-control** を使用して、設定をグローバルに削除し、設定されているすべてのインターフェイスからも削除します。



- (注) 802.1x 認証がグローバルにディセーブル化されても、Web 認証など他の認証方法はそのポートでイネーブルのままです。

authentication manager コマンドは、以前の 802.1x コマンドと同じ機能を提供します。

認証マネージャが生成する冗長なシステムメッセージをフィルタリングすると、通常は、フィルタリングされた内容が認証の成功に結びつきます。802.1x 認証および MAB 認証の冗長なメッセージをフィルタリングすることもできます。認証方式ごとに異なるコマンドが用意されています。

- **no authentication logging verbose** グローバル コンフィギュレーション コマンドは、認証マネージャからの詳細メッセージをフィルタリングします。
- **no dot1x logging verbose** グローバル コンフィギュレーション コマンドは、802.1x 認証の詳細メッセージをフィルタリングします。
- **no mab logging verbose** グローバル コンフィギュレーション コマンドは、MAC 認証バイパス (MAB) の詳細メッセージをフィルタリングします。

802.1x 認証のデフォルト設定

表 30: 802.1x 認証のデフォルト設定

機能	デフォルト設定
スイッチの 802.1x イネーブル ステート	ディセーブル
ポート単位の 802.1x イネーブル ステート	ディセーブル (force-authorized) ポートはクライアントとの 802.1x ベース認証を行わ ラフィックを送受信します。

機能	デフォルト設定
AAA	ディセーブル
RADIUS サーバ <ul style="list-style-type: none"> • IP アドレス • UDP 認証ポート • デフォルトのアカウントिंग ポート • キー 	<ul style="list-style-type: none"> • 指定なし • 1645 • 1646 • 指定なし
ホスト モード	シングル ホスト モード
制御方向	双方向制御
定期的な再認証	ディセーブル。
再認証の間隔 (秒)	3600 秒
再認証回数	2 回 (ポートが無許可ステートに変わる前に、スイッチプロセスを再開する回数)
待機時間	60 秒 (スイッチがクライアントとの認証情報の交換に失敗したとき、待機状態を続ける秒数)
再送信時間	30 秒 (スイッチが EAP-Request/Identity フレームに対するクライアントからの応答を待ち、要求を再送信するまでの秒数)
最大再送信回数	2 回 (スイッチが認証プロセスを再開する前に、EAP-Request フレームを送信する回数)
クライアント タイムアウト時間	30 秒 (認証サーバーからの要求をクライアントにリクエストしたとき、スイッチが返答を待ち、クライアントに要求を再送信するまでの秒数)
認証サーバー タイムアウト時間	30 秒 (クライアントからの応答を認証サーバーにリクエストしたとき、スイッチが応答を待ち、応答をサーバーに再送信するまでの秒数) dot1x timeout server-timeout インターフェイス コンフィギュレーション コマンドを使用して、このタイムアウト時間を変更する
無活動タイムアウト	ディセーブル
ゲスト VLAN	指定なし
アクセス不能認証バイパス	ディセーブル
制限付き VLAN	指定なし

機能	デフォルト設定
オーセンティケータ（スイッチ）モード	指定なし
MAC 認証バイパス	ディセーブル
音声認識セキュリティ	ディセーブル

ポートベース認証とスイッチ スタック

スイッチが、スイッチ スタックに追加されるか、スイッチ スタックから削除される場合、RADIUS サーバーとスタックとの間の IP 接続が正常な場合、802.1x 認証は影響を受けません。これは、スタックのアクティブスイッチがスイッチスタックから削除される場合も、適用されます。アクティブスイッチに障害が発生した場合、スタック内のメンバスイッチは、選択プロセスを使用することによって新しいアクティブスイッチになり、802.1x 認証プロセスは通常どおり続行されます。

サーバーに接続されていたスイッチが削除されたか、そのスイッチに障害が発生したために、RADIUS サーバーへの IP 接続が中断された場合、これらのイベントが発生します。

- すでに認証済みで、定期的な再認証がイネーブルではないポートは、認証ステートのままです。RADIUS サーバーとの通信は、必要ではありません。
- すでに認証済みで、（**dot1x re-authentication** グローバル コンフィギュレーション コマンドを使用して）定期的な再認証が有効になっているポートは、再認証の発生時に、認証プロセスに失敗します。ポートは、再認証プロセス中に、非認証ステートに戻ります。RADIUS サーバーとの通信が必要です。

進行中の認証については、サーバー接続が行われていないため、認証はただちに失敗します。

障害が発生したスイッチが実行状態になり、スイッチスタックに再加入した場合、ブートアップの時刻と、認証の試行時までには RADIUS サーバーへの接続が再確立されたかどうかによって、認証は失敗する場合と、失敗しない場合があります。

RADIUS サーバーへの接続を失うことを避けるには、冗長接続を設定する必要があります。たとえば、アクティブスイッチへの冗長接続と、メンバスイッチへの別の接続を設定できます。アクティブスイッチに障害が発生した場合でも、スイッチスタックは、RADIUS サーバーに接続されたままです。

VLAN 割り当てを使用した 802.1x 認証

スイッチは、VLAN 割り当てを使用した 802.1x 認証をサポートしています。ポートの 802.1x 認証が成功すると、RADIUS サーバーは VLAN 割り当てを送信し、スイッチ ポートを設定します。RADIUS サーバーデータベースは、ユーザー名と VLAN のマッピングを維持し、スイッチ ポートに接続するクライアントのユーザー名に基づいて VLAN を割り当てます。この機能を使用して、特定のユーザーのネットワーク アクセスを制限できます。

音声デバイスが許可されているときに、RADIUS サーバから許可された VLAN が返された場合、このポートの音声 VLAN は、割り当てられた音声 VLAN でパケットを送受信するように設定されています。音声 VLAN 割り当ては、マルチドメイン認証 (MDA) 対応のポートでのデータ VLAN 割り当てと同じように機能します。

スイッチと RADIUS サーバ上で設定された場合、VLAN 割り当てを使用した 802.1x 認証には次の特性があります。

- RADIUS サーバから VLAN が提供されない場合、または 802.1x 認証がディセーブルの場合、認証が成功するとポートはアクセス VLAN に設定されます。アクセス VLAN とは、アクセス ポートに割り当てられた VLAN です。このポート上で送受信されるパケットはすべて、この VLAN に所属します。
- 802.1x 認証がイネーブルで、RADIUS サーバからの VLAN 情報が有効でない場合、認証に失敗して、設定済みの VLAN が引き続き使用されます。これにより、設定エラーによって不適切な VLAN に予期せぬポートが現れることを防ぎます。

設定エラーには、ルーテッドポートの VLAN、間違った VLAN ID、存在しないまたは内部 (ルーテッドポート) の VLAN ID、RSPAN VLAN、シャットダウンしている VLAN、あるいは一時停止している VLAN ID の指定などがあります。マルチドメインホストポートの場合、設定エラーには、設定済みまたは割り当て済み VLAN ID と一致するデータ VLAN の割り当て試行 (またはその逆) のために発生するものもあります。

- 802.1x 認証がイネーブルで、RADIUS サーバからのすべての情報が有効の場合、許可されたデバイスは認証後、指定した VLAN に配置されます。
- 802.1x ポートでマルチ ホスト モードがイネーブルの場合、すべてのホストは最初に認証されたホストと同じ VLAN (RADIUS サーバにより指定) に配置されます。
- ポート セキュリティをイネーブル化しても、RADIUS サーバが割り当てられた VLAN の動作には影響しません。
- 802.1x 認証がポートでディセーブルの場合、設定済みのアクセス VLAN と設定済みの音声 VLAN に戻ります。
- 802.1x ポートが認証され、RADIUS サーバによって割り当てられた VLAN に配置されると、そのポートのアクセス VLAN 設定への変更は有効になりません。マルチドメインホストの場合、ポートが完全にこれらの例外で許可されている場合、同じことが音声デバイスに適用されます。

- あるデバイスで VLAN 設定を変更したことにより、他のデバイスに設定済みまたは割り当て済みの VLAN と一致した場合、ポート上の全デバイスの認証が中断して、データおよび音声デバイスに設定済みの VLAN が一致しなくなるような有効な設定が復元されるまで、マルチドメインホストモードがディセーブルになります。
- 音声デバイスが許可されて、ダウンロードされた音声 VLAN を使用している場合、音声 VLAN 設定を削除したり設定値を *dot1p* または *untagged* に修正したりすると、音声デバイスが未許可になり、マルチドメインホストモードが無効になります。

ポートが、強制許可 (force-authorized) ステート、強制無許可 (force-unauthorized) ステート、無許可ステート、またはシャットダウンステートの場合、ポートは設定済みのアクセス VLAN に配置されます。

VLAN 割り当てを設定するには、次の作業を実行する必要があります。

- **network** キーワードを使用して AAA 認証をイネーブルにし、RADIUS サーバからのインターフェイス設定を可能にします。
- 802.1x 認証をイネーブルにします。(アクセスポートで 802.1x 認証を設定すると、VLAN 割り当て機能は自動的にイネーブルになります)。
- RADIUS サーバにベンダー固有のトンネル属性を割り当てます。RADIUS サーバーは次の属性をスイッチに返す必要があります。
 - [64] Tunnel-Type = VLAN
 - [65] Tunnel-Medium-Type = 802
 - [81] Tunnel-Private-Group-ID = VLAN 名または VLAN ID
 - [83] Tunnel-Preference

属性 [64] は、値 *VLAN* (タイプ 13) でなければなりません。属性 [65] は、値 *802* (タイプ 6) でなければなりません。属性 [81] は、IEEE 802.1x 認証ユーザーに割り当てられた *VLAN* 名または *VLAN ID* を指定します。

ユーザー単位 ACL を使用した 802.1x 認証

ユーザー単位アクセスコントロールリスト (ACL) をイネーブルにして、異なるレベルのネットワーク アクセスおよびサービスを 802.1x 認証ユーザーに提供できます。RADIUS サーバーは、802.1x ポートに接続されるユーザーを認証する場合、ユーザー ID に基づいて ACL 属性を受け取り、これらをスイッチに送信します。スイッチは、ユーザーセッションの期間中、その属性を 802.1x ポートに適用します。セッションが終了すると、認証が失敗した場合、またはリンクダウン状態の発生時に、ユーザー単位 ACL 設定が削除されます。スイッチは、RADIUS 指定の ACL を実行コンフィギュレーションには保存しません。ポートが無許可の場合、スイッチはそのポートから ACL を削除します。

ユーザーは同一のスイッチ上で、ルータ ACL および入力ポート ACL を使用できます。ただし、ポートの ACL はルータ ACL より優先されます。入力ポート ACL を VLAN に属するインターフェイスに適用する場合、ポート ACL は VLAN インターフェイスに適用する入力ルータ ACL よりも優先されます。ポート ACL が適用されたポート上で受信した着信パケットは、ポート ACL によってフィルタリングされます。その他のポートに着信したルーテッドパケットは、ルータ ACL によってフィルタリングされます。発信するルーテッドパケットには、ルータ ACL のフィルタが適用されます。コンフィギュレーションの矛盾を回避するには、RADIUS サーバーに保存するユーザー プロファイルを慎重に計画しなければなりません。

RADIUS は、ベンダー固有属性などのユーザー単位属性をサポートします。ベンダー固有属性 (VSA) は、オクテットストリング形式で、認証プロセス中にスイッチに渡されます。ユーザー単位 ACL に使用される VSA は、入力方向では `inacl#<n>` で、出力方向では `outacl#<n>` です。

MAC ACL は、入力方向に限りサポートされます。VSA は入力方向に限りサポートされます。レイヤ 2 ポートの出力方向ではポート ACL をサポートしません。

拡張 ACL 構文形式だけを使用して、RADIUS サーバに保存するユーザ単位コンフィギュレーションを定義します。RADIUS サーバから定義が渡される場合、拡張命名規則を使用して作成されます。ただし、Filter-Id 属性を使用する場合、標準 ACL を示すことができます。

Filter-Id 属性を使用して、すでにスイッチに設定されているインバウンドまたはアウトバウンド ACL を指定できます。属性には、ACL 番号と、その後ろに入力フィルタリング、出力フィルタリングを示す *.in* または *.out* が含まれています。RADIUS サーバが *.in* または *.out* 構文を許可しない場合、アクセスリストはデフォルトで発信 ACL に適用されます。RADIUS サーバから送信された Filter-Id がデバイスで設定されていない場合、ユーザーは未承認としてマークされます。Filter-Id 属性は 1 ~ 199 (IP 標準 ACL) および 1300 ~ 2699 (IP 拡張 ACL) の範囲の IP ACL に対してだけサポートされます。

ユーザ単位 ACL の最大サイズは、4000 ASCII 文字ですが、RADIUS サーバのユーザ単位 ACL の最大サイズにより制限されます。

ユーザ単位の ACL を設定するには、次の前提条件を満たす必要があります。

- AAA 認証をイネーブルにします。
- **network** キーワードを使用して AAA 認証をイネーブルにし、RADIUS サーバからのインターフェイス設定を可能にします。
- 802.1x 認証をイネーブルにします。
- RADIUS サーバにユーザ プロファイルと VSA を設定します。
- 802.1x ポートをシングル ホスト モードに設定します。



(注) ユーザー単位 ACL がサポートされるのはシングル ホスト モードだけです。

ダウンロード可能 ACL およびリダイレクト URL を使用した 802.1x 認証

ACL およびリダイレクト URL は、ホストの 802.1x 認証または MAC 認証バイパス中に、RADIUS サーバからスイッチにダウンロードできます。また、Web 認証中に ACL をダウンロードすることもできます。ダウンロード可能な ACL は *dACL* とも呼ばれます。

複数のホストが認証され、それらのホストがシングルホストモード、MDA モード、またはマルチ認証モードである場合、スイッチは ACL の送信元アドレスをホスト IP アドレスに変更します。

ACL およびリダイレクト URL は、802.1x 対応のポートに接続されるすべてのデバイスに適用できます。

ACL が 802.1x 認証中にダウンロードされない場合、スイッチは、ポートのスタティックデフォルト ACL をホストに適用します。マルチ認証モードまたは MDA モードで設定された音声 VLAN ポートでは、スイッチは ACL を認証ポリシーの一部として電話にだけ適用します。

スタック構成がある dACL の制限は、ポートベースの dACL あたり 64 ACE です。スタック構成なしの制限は、利用可能な TCAM エントリの数になり、これはアクティブな他の ACL 機能によって異なります。

同じタイプ (IPv4 または IPv6) の複数の dACL は、Cisco Identity Services Engine (ISE) ではサポートされません。一意の DACL のみが Cisco ISE から送信されるようにします。

802.1x および MAB 認証方式では、オープンおよびクローズの 2 つの認証方式がサポートされます。クローズ認証モードのポートにスタティック ACL がない場合、次のようになります。

- 最初のホスト認証では、許可ポリシーは IP アドレスを挿入せずに適用されます。
- 別のホストが検出されると、最初のホストのポリシーがリフレッシュされ、最初のセッションと後続セッションのポリシーが IP アドレスを挿入して実施されます。

オープン認証モードのポートにスタティック ACL がない場合、次のようになります。

- セキュリティ違反を防ぐために、IP アドレスを挿入してポリシーが実施されます。

許可ポリシーのないホストへのアクセスを制御するために、ディレクティブを設定することができます。サポートされているディレクティブの値は、*open* と *default* です。*open* ディレクティブを設定すると、すべてのトラフィックが許可されます。*default* ディレクティブは、ポートから提供されるアクセスにトラフィックを従わせます。ディレクティブは、AAA サーバー上のユーザープロファイル、またはスイッチ上のいずれかで設定できます。AAA サーバー上でディレクティブを設定するには、**authz-directive = <open/default>** グローバルコマンドを使用します。スイッチ上でディレクティブを設定するには、**epm access-control open** グローバルコンフィギュレーションコマンドを使用します。



(注) ディレクティブのデフォルト値は *default* です。

URL リダイレクト ACL の場合：

- 許可アクセス コントロール エントリ (ACE) ルールに一致するパケットは、AAA サーバーに転送するために CPU に送信されます。
- 拒否 ACE ルールに一致するパケットは、スイッチを介して転送されます。
- 許可 ACE ルールにも拒否 ACE ルールにも一致しないパケットは、次の dACL によって処理されます。dACL がない場合、パケットは暗黙的拒否 ACL にヒットしてドロップされます。

VLAN ID ベース MAC 認証

ダウンロード可能な VLAN ではなくスタティック VLAN ID に基づいてホストを認証する場合、VLAN ID ベース MAC 認証を使用できます。スタティック VLAN ポリシーがスイッチで設定されている場合、認証用の各ホストの MAC アドレスとともに、VLAN 情報が IAS (Microsoft) RADIUS サーバーに送信されます。接続ポートに設定されている VLAN ID は MAC 認証に使用されます。VLAN ID ベース MAC 認証を IAS サーバーで使用することで、ネットワークで一定数の VLAN を使用できます。

機能は、STP によってモニターおよび処理される VLAN の数も制限します。ネットワークは固定 VLAN として管理できます。

MAC 認証バイパスを使用した IEEE 802.1x 認証

MAC 認証バイパス機能を使用し、クライアント MAC アドレスに基づいてクライアントを許可するようにスイッチを設定できます。たとえば、プリンタなどのデバイスに接続された IEEE 802.1x ポートでこの機能をイネーブルにできます。

クライアントからの EAPOL 応答の待機中に IEEE 802.1x 認証がタイムアウトした場合、スイッチは MAC 認証バイパスを使用してクライアントを許可しようとします。

MAC 認証バイパス機能が IEEE 802.1x ポートでイネーブルの場合、スイッチはクライアント ID として MAC アドレスを使用します。認証サーバーには、ネットワーク アクセスを許可されたクライアント MAC アドレスのデータベースがあります。スイッチは、IEEE 802.1x ポート上のクライアントを検出した後で、クライアントからのイーサネットパケットを待機します。スイッチは、MAC アドレスに基づいたユーザー名およびパスワードを持つ RADIUS-access/request フレームを認証サーバーに送信します。認証に成功すると、スイッチはクライアントにネットワークへのアクセスを許可します。許可が失敗した場合、ゲスト VLAN が設定されていれば、スイッチはポートをゲスト VLAN に割り当てます。このプロセスは、ほとんどのクライアントデバイスで動作します。ただし、代替の MAC アドレス形式を使用しているクライアントでは動作しません。標準の形式とは異なる MAC アドレスを持つクライアントに対して MAB 認証をどのように実行するかや、RADIUS の設定のどこでユーザー名とパスワードが異なることが要求されるかを設定できます。

リンクのライフタイム中に EAPOL パケットがインターフェイス上で検出された場合、スイッチは、そのインターフェイスに接続されているデバイスが 802.1x 対応サブリカントであることを確認し、(MAC 認証バイパス機能ではなく) 802.1x 認証を使用してインターフェイスを認証します。インターフェイスのリンク ステータスがダウンした場合、EAPOL 履歴はクリアされます。

スイッチがすでに MAC 認証バイパスを使用してポートを許可し、IEEE 802.1x サブリカントを検出している場合、スイッチはポートに接続されているクライアントを許可します。再認証が発生するときに、Termination-Action RADIUS 属性値が DEFAULT であるために前のセッションが終了した場合、スイッチはポートに設定されている認証または再認証手法を使用します。

MAC 認証バイパスで認証されたクライアントは再認証できます。再認証プロセスは、IEEE 802.1x を使用して認証されたクライアントに対するプロセスと同じです。再認証中は、ポートは前に割り当てられた VLAN のままです。再認証に成功すると、スイッチはポートを同じ

VLAN に保持します。再認証に失敗した場合、ゲスト VLAN が設定されていれば、スイッチはポートをゲスト VLAN に割り当てます。

再認証が Session-Timeout RADIUS 属性 (Attribute[27])、および Termination-Action RADIUS 属性 (Attribute[29]) に基づいて行われるときに、Termination-Action RADIUS 属性 (Attribute[29]) のアクションが *Initialize* (属性値は *DEFAULT*) である場合、MAC 認証バイパスセッションは終了し、再認証の間の接続は失われます。MAC 認証バイパス機能が IEEE 802.1x 認証がタイムアウトした場合、スイッチは MAC 認証バイパス機能を使用して再認証を開始します。これらの AV ペアの詳細については、RFC 3580 『IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines』を参照してください。

MAC 認証バイパスは、次の機能と相互に作用します。

- IEEE 802.1x 認証：802.1x 認証がポートでイネーブルの場合にのみ MAC 認証バイパスをイネーブルにできます。
- ゲスト VLAN：クライアントの MAC アドレス ID が無効な場合、ゲスト VLAN が設定されていれば、スイッチは VLAN にクライアントを割り当てます。
- 制限付き VLAN：IEEE 802.1x ポートに接続されているクライアントが MAC 認証バイパスで認証されている場合には、この機能はサポートされません。
- ポートセキュリティ
- 音声 VLAN
- プライベート VLAN：クライアントをプライベート VLAN に割り当てられます。
- Network Edge Access Topology (NEAT)：MAB と NEAT は相互に排他的です。インターフェイス上で NEAT が有効の場合は、MAB を有効にすることはできません。また、インターフェイス上で MAB が有効の場合は、NEAT を有効にすることはできません。

MAC 認証バイパス設定の注意事項

MAC 認証バイパス設定時の注意事項は次のとおりです。

- 特に明記していないかぎり、MAC 認証バイパスの注意事項は 802.1x 認証のものと同じです。
- ポートが MAC アドレスで許可された後に、ポートから MAC 認証バイパスをディセーブルにしても、ポートステータスに影響はありません。
- ポートが未許可ステータスであり、クライアント MAC アドレスが認証サーバデータベースにない場合、ポートは未許可ステータスのままです。ただし、クライアント MAC アドレスがデータベースに追加されると、スイッチは MAC 認証バイパス機能を使用してポートを再認証できます。
- ポートが認証ステータスにない場合、再認証が行われるまでポートはこのステータスを維持します。
- MAC 認証バイパスにより接続されているが、非アクティブなホストのタイムアウト時間を設定できます。指定できる範囲は 1 ～ 65535 秒です。

802.1x マルチ認証モード

マルチ認証 (multiauth) モードでは、データ VLAN および音声 VLAN で複数のクライアントを認証できます。各ホストは個別に認証されます。マルチ認証ポートで認証できるデータデバイスまたは音声デバイスの数には制限はありません。

ハブまたはアクセス ポイントが 802.1x 対応ポートに接続されている場合、接続されている各クライアントを認証する必要があります。802.1x 以外のデバイスでは、MAC 認証バイパスまたは Web 認証をホスト単位認証フォールバック メソッドとして使用し、単一のポートで異なる方法で異なるホストを認証できます。

次の条件で、RADIUS サーバから提供された VLAN をマルチ認証モードで割り当てることができます。

- ホストがポートで最初に許可されたホストであり、RADIUS サーバが VLAN 情報を提供している。
- 後続のホストが、動作 VLAN に一致する VLAN を使用して許可される。
- ホストは VLAN が割り当てられていないポートで許可され、後続のホストでは VLAN 割り当てが設定されていないか、VLAN 情報が動作 VLAN と一致している。
- ポートで最初に許可されたホストにはグループ VLAN が割り当てられ、後続のホストでは VLAN 割り当てが設定されていないか、グループ VLAN がポート上のグループ VLAN と一致している。後続のホストが、最初のホストと同じ VLAN グループの VLAN を使用する必要がある。VLAN リストが使用されている場合、すべてのホストは VLAN リストで指定された条件に従う。
- VLAN がポート上のホストに割り当てられると、後続のホストは一致する VLAN 情報を持つ必要があり、この情報がなければポートへのアクセスを拒否される。
- クリティカル認証 VLAN の動作が、マルチ認証モード用に変更されない。ホストが認証を試みたときにサーバに到達できない場合、許可されたすべてのホストは、設定された VLAN で再初期化される。

ユーザーごとのマルチ認証 VLAN 割り当て

ユーザーごとのマルチ認証 VLAN 割り当て機能を使用すると、単一の設定済みアクセス VLAN を持つポート上のクライアントに割り当てられた VLAN に基づいて複数の運用アクセス VLAN を作成することができます。データ ドメインに関連付けられたすべての VLAN に対するトラフィックが dot1q とタグ付けされていないアクセス ポートとして設定されているポートおよびこれらの VLAN は、ネイティブ VLAN として処理されます。

マルチ認証ポート 1 つあたりのホストの数は 8 ですが、さらに多くのホストが存在する場合があります。

次のシナリオは、ユーザーごとのマルチ認証 VLAN 割り当てに関連しています。

シナリオ 1

ハブがアクセス ポートに接続されている場合、およびポートがアクセス VLAN (V0) で設定されている場合。

ホスト (H1) は、ハブを介して VLAN (V1) に割り当てられます。ポートの運用 VLAN は V1 に変更されます。この動作は、単一ホストポートまたはマルチドメイン認証ポートと同様です。

2 番目のホスト (H2) が接続され、VLAN (V2) に割り当てられる場合、ポートには 2 つの運用 VLAN があります (V1 および V2)。H1 と H2 がタグなし入力トラフィックを送信すると、H1 トラフィックは VLAN (V1) に、H2 トラフィックは VLAN (V2) にマッピングされ、VLAN (V1) および VLAN (V2) のポートからの出トラフィックはすべてタグなしになります。

両方のホスト H1 と H2 がログアウトするか、またはセッションがなんらかの理由で削除されると、VLAN (V1) と VLAN (V2) がポートから削除され、設定された VLAN (V0) がポートに復元されます。

シナリオ 2

ハブがアクセスポートに接続されている場合、およびポートがアクセス VLAN (V0) で設定されている場合。ホスト (H1) は、ハブを介して VLAN (V1) に割り当てられます。ポートの運用 VLAN は V1 に変更されます。

2 番目のホスト (H2) が接続され明示的な VLAN ポリシーなしで承認されると、H2 はポート上で復元される設定済み VLAN (V0) を使用することを予期されます。2 つの運用 VLAN、VLAN (V0) および VLAN (V1) からの出トラフィックはすべてタグなしになります。

ホスト (H2) がログアウトするか、またはセッションがなんらかの理由で削除されると、設定された VLAN (V0) がポートから削除され、VLAN (V1) がそのポートでの唯一の運用 VLAN になります。

シナリオ 3

ハブがオープンモードでアクセスポートに接続されている場合、およびポートがアクセス VLAN (V0) で設定されている場合。

ホスト (H1) は、ハブを介して VLAN (V1) に割り当てられます。ポートの運用 VLAN は V1 に変更されます。2 番目のホスト (H2) が接続され無許可のままだと、オープンモードにより、運用 VLAN (V1) に引き続きアクセスできます。

ホスト H1 がログアウトするか、またはセッションがなんらかの理由で削除されると、VLAN (V1) はポートから削除され、ホスト (H2) は VLAN (V0) に割り当てられます。



Note オープンモードと VLAN 割り当ての組み合わせは、ホスト (H2) に悪影響を与えます。そのホストは VLAN (V1) に対応するサブネット内に IP アドレスを含んでいるからです。

ユーザーごとのマルチ認証 VLAN 割り当ての制限

ユーザーごとのマルチ認証 VLAN 割り当て機能では、複数の VLAN からの出トラフィックは、ホストが自分宛てではないトラフィックを受信するポート上ではタグなしになります。これは、ブロードキャストおよびマルチキャストトラフィックで問題になる可能性があります。

- **IPv4 ARP** : ホストは他のサブネットからの ARP パケットを受信します。これは、IP アドレス範囲が重複する異なる仮想ルーティングおよび転送 (VRF) テーブルの 2 個のサブネットがポート上でアクティブな場合に問題となります。ホスト ARP キャッシュのエントリが無効になる可能性があります。
- **IPv6 制御パケット** : IPv6 の導入環境では、ルーターアドバタイズメント (RA) は、その受信を想定されていないホストによって処理されます。ある VLAN からのホストが別の VLAN からの RA を受信すると、ホストはそれ自身に間違っただけの IPv6 アドレスを割り当てます。このようなホストは、ネットワークにアクセスできません。
回避策は、IPv6 ファースト ホップ セキュリティをイネーブルにして、ブロードキャスト ICMPv6 パケットがユニキャストに変換され、マルチ認証がイネーブルのポートから送信されるようにすることです。パケットは VLAN に属するマルチ認証ポートの各クライアント用に複製され、宛先 MAC が個々のクライアントに設定されます。1 つの VLAN を持つポートで、ICMPv6 パケットは正常にブロードキャストされます。
- **IP マルチキャスト** : 送信先のマルチキャストグループへのマルチキャストトラフィックは、異なる VLAN 上のホストがそのマルチキャストグループに参加している場合それらの VLAN 用に複製されます。異なる VLAN の 2 つのホストが (同じマルチ認証ポート上の) マルチキャストグループに参加している場合、各マルチキャストパケットのコピー 2 部がそのポートから送信されます。

ゲスト VLAN を使用した 802.1x 認証

スイッチ上の各 802.1x ポートにゲスト VLAN を設定し、クライアントに対して限定的なサービスを提供できます (802.1x クライアントのダウンロードなど)。これらのクライアントは 802.1x 認証用にシステムをアップグレードできる場合がありますが、一部のホスト (Windows 98 システムなど) は IEEE 802.1x 対応ではありません。

スイッチが EAP Request/Identity フレームに対する応答を受信していない場合、または EAPOL パケットがクライアントによって送信されない場合に、802.1x ポート上でゲスト VLAN をイネーブルにすると、スイッチはクライアントにゲスト VLAN を割り当てます。

スイッチは EAPOL パケット履歴を保持します。EAPOL パケットがリンクの存続時間中にインターフェイスで検出された場合、スイッチはそのインターフェイスに接続されているデバイスが IEEE 802.1x 対応のものであると判断します。インターフェイスはゲスト VLAN ステートにはなりません。インターフェイスのリンクステータスがダウンした場合、EAPOL 履歴はクリアされます。EAPOL パケットがインターフェイスで検出されない場合、そのインターフェイスはゲスト VLAN のステートになります。

スイッチが 802.1x 対応の音声デバイスを許可しようとしたが、AAA サーバーが使用できない場合、許可は失敗します。ただし、EAPOL パケットの検出は EAPOL 履歴に保存されます。この音声デバイスは、AAA サーバーが使用可能になると許可されます。ただし、他のデバイスによるゲスト VLAN へのアクセスは許可されなくなります。この状況を防ぐには、次のいずれかのコマンドシーケンスを使用します。

- **authentication event no-response action authorize vlan *vlan-id*** インターフェイス コンフィギュレーション コマンドを入力し、ゲスト VLAN へのアクセスを許可します。

- **shutdown** インターフェイス コンフィギュレーション コマンドを入力し、さらに **no shutdown** インターフェイス コンフィギュレーション コマンドを入力してポートを再起動します。

リンクの存続時間中にデバイスが EAPOL パケットを送信した場合、スイッチはゲスト VLAN への認証アクセスに失敗したクライアントを許可しません。



- (注) インターフェイスがゲスト VLAN に変わってから EAPOL パケットが検出された場合、無許可状態に戻って 802.1x 認証を再起動します。

スイッチ ポートがゲスト VLAN に変わると、802.1x 非対応クライアントはすべてアクセスを許可されます。ゲスト VLAN が設定されているポートに 802.1x 対応クライアントが加入すると、ポートは、ユーザ設定によるアクセス VLAN で無許可状態になり、認証が再起動されます。

ゲスト VLAN は、単一のホスト、複数のホスト、複数認証、またはマルチドメイン モードにおける 802.1x ポートでサポートされています。

RSPAN VLAN、プライベート VLAN、音声 VLAN を除いて、アクティブ VLAN を 802.1x ゲスト VLAN として設定できます。ゲスト VLAN 機能は、内部 VLAN (ルーテッドポート) またはトランクポートではサポートされていません。アクセスポート上でだけサポートされます。

スイッチは MAC 認証バイパスをサポートします。MAC 認証バイパスが 802.1x ポートでイネーブルの場合、スイッチは、IEEE 802.1x 認証のタイムアウト時に EAPOL メッセージ交換を待機している間、クライアント MAC アドレスに基づいてクライアントを許可できます。スイッチは、802.1x ポート上のクライアントを検出したあとで、クライアントからのイーサネットパケットを待機します。スイッチは、MAC アドレスに基づいたユーザ名およびパスワードを持つ RADIUS-access/request フレームを認証サーバーに送信します。認証に成功すると、スイッチはクライアントにネットワークへのアクセスを許可します。認証に失敗すると、スイッチはポートにゲスト VLAN を割り当てます (指定されていない場合)。

制限付き VLAN を使用した 802.1x 認証

ゲスト VLAN にアクセスできないクライアント向けに、限定されたサービスを提供するために、スイッチ スタックまたはスイッチの各 IEEE 802.1x ポートに対して制限付き VLAN (認証失敗 VLAN と呼ばれることもあります) を設定できます。これらのクライアントは、認証プロセスに失敗したため他の VLAN にアクセスできない 802.1x 対応クライアントです。制限付き VLAN を使用すると、認証サーバーの有効なクレデンシャルを持っていないユーザ (通常、企業にアクセスするユーザ) に、サービスを制限したアクセスを提供できます。管理者は制限付き VLAN のサービスを制御できます。



- (注) 両方のタイプのユーザに同じサービスを提供する場合、ゲスト VLAN と制限付き VLAN の両方を同じに設定できます。

この機能がないと、クライアントは認証失敗を永遠に繰り返すことになるため、スイッチポートがスパンニングツリーのブロッキングステートから変わることができなくなります。制限付き VLAN の機能を使用することで、クライアントの認証試行回数を指定し（デフォルト値は 3 回）、一定回数後にスイッチポートを制限付き VLAN の状態に移行させることができます。

認証サーバはクライアントの認証試行回数をカウントします。このカウントが設定した認証試行回数を超えると、ポートが制限付き VLAN の状態に変わります。失敗した試行回数は、RADIUS サーバが *EAP failure* で応答したときや、EAP パケットなしの空の応答を返したときからカウントされます。ポートが制限付き VLAN に変わったら、このカウント数はリセットされます。

認証に失敗したユーザの VLAN は、もう一度認証を実行するまで制限された状態が続きます。VLAN 内のポートは設定された間隔に従って再認証を試みます（デフォルトは 60 秒）。再認証に失敗している間は、ポートの VLAN は制限された状態が続きます。再認証に成功した場合、ポートは設定された VLAN もしくは RADIUS サーバによって送信された VLAN に移行します。再認証はディセーブルにすることもできますが、ディセーブルにすると、*link down* または *EAP logoff* イベントを受信しない限り、ポートの認証プロセスを再起動できません。クライアントがハブを介して接続している場合、再認証機能はイネーブルにしておくことを推奨します。クライアントの接続をハブから切り離すと、ポートに *link down* や *EAP logoff* イベントが送信されない場合があります。

ポートが制限付き VLAN に移行すると、EAP 成功の疑似メッセージがクライアントに送信されます。このメッセージによって、繰り返し実行している再認証を停止させることができます。クライアントによっては（Windows XP が稼働しているデバイスなど）、EAP なしで DHCP を実装できません。

制限付き VLAN は、すべてのホストモードでの 802.1x ポート上、およびレイヤ 2 ポート上でサポートされます。

RSPAN VLAN、プライマリプライベート VLAN、音声 VLAN を除いて、アクティブ VLAN を 802.1x 制限付き VLAN として設定できます。制限付き VLAN 機能は、内部 VLAN（ルーテッドポート）またはトランクポートではサポートされていません。アクセスポート上でだけサポートされます。

ダイナミック ARP インスペクション、DHCP スヌーピング、IP 送信元ガードなどの他のセキュリティポート機能は、制限付き VLAN に対して個別に設定できます。

アクセス不能認証バイパスを使用した 802.1x 認証

スイッチが設定された RADIUS サーバに到達できず、新しいホストを認証できない場合、アクセス不能認証バイパス機能を使用します。この機能は、クリティカル認証または AAA 失敗ポリシーとも呼ばれます。これらのホストをクリティカルポートに接続するようにスイッチを設定できます。

新しいホストがクリティカルポートに接続しようとする、そのホストはユーザ指定のアクセス VLAN、クリティカル VLAN に移動されます。管理者はこれらのホストに制限付き認証を付与します。

スイッチは、クリティカルポートに接続されているホストを認証しようとする場合、設定されている RADIUS サーバのステータスをチェックします。利用可能なサーバが 1 つあれば、

スイッチはホストを認証できます。ただし、すべての RADIUS サーバーが利用不可能な場合は、スイッチはホストへのネットワークアクセスを許可して、ポートを認証ステートの特別なケースであるクリティカル認証ステートにします。



- (注) クリティカル認証をインターフェイスで設定する場合は、クリティカル承認 (クリティカル *vlan*) に使用する *vlan* をスイッチでアクティブにする必要があります。クリティカル *vlan* が非アクティブまたはダウンしていると、クリティカル認証セッションは非アクティブな *vlan* の有効化を試行し続け、繰り返し失敗します。これは大量のメモリ保持の原因となる可能性があります。

複数認証ポートのアクセス不能認証バイパスのサポート

ポートが任意のホストモードで設定されていて、AAA サーバーを使用できない場合、ポートはマルチホストモードに設定され、クリティカル VLAN に移動されます。マルチ認証 (multiauth) ポートで、このアクセス不能バイパスをサポートするには、**authentication event server dead action reinitialize vlan *vlan-id*** コマンドを使用します。新しいホストがクリティカルポートに接続しようとする、そのポートは再初期化され、接続されているすべてのホストがユーザ指定のアクセス VLAN に移動されます。

このコマンドは、すべてのホストモードでサポートされます。

アクセス不能認証バイパスの認証結果

アクセス不能認証バイパス機能の動作は、ポートの許可ステートにより異なります。

- クリティカルポートに接続されているホストが認証しようとする際にポートが無許可ですべてのサーバーが利用できない場合、スイッチは RADIUS 設定済み VLAN またはユーザー指定のアクセス VLAN にあるポートをクリティカル認証ステートにします。
- ポートが許可済みで、再認証が行われた場合、スイッチは現在の VLAN (事前に RADIUS サーバーにより割り当てられた) でクリティカルポートをクリティカル認証ステートにします。
- 認証交換中に RADIUS サーバーが利用不可能となった場合、現在の交換はタイムアウトとなり、スイッチは次の認証試行の間にクリティカルポートをクリティカル認証ステートとします。

RADIUS サーバが再び使用可能になったときにホストを再初期化し、クリティカル VLAN から移動するように、クリティカルポートを設定できます。このように設定した場合、クリティカル認証ステートのすべてのクリティカルポートは自動的に再認証されます。

アクセス不能認証バイパス機能の相互作用

アクセス不能認証バイパスは、次の機能と相互に作用します。

- ゲスト VLAN : アクセス不能認証バイパスは、ゲスト VLAN と互換性があります。ゲスト VLAN が 8021.x ポートでイネーブルの場合、この機能は次のように相互に作用します。

- スイッチが EAP Request/Identity フレームへの応答を受信しないとき、または EAPOL パケットがクライアントによって送信されないときに、少なくとも 1 つの RADIUS サーバーが使用できれば、スイッチはクライアントにゲスト VLAN を割り当てます。
- すべての RADIUS サーバーが使用できず、クライアントがクリティカル ポートに接続されている場合、スイッチはクライアントを認証して、クリティカル ポートを RADIUS 認証済み VLAN またはユーザー指定のアクセス VLAN でクリティカル認証ステートにします。
- すべての RADIUS サーバーが使用できず、クライアントがクリティカル ポートに接続されていない場合、ゲスト VLAN が設定されていても、スイッチはクライアントにゲスト VLAN を割り当てられません。
- すべての RADIUS サーバーが使用できず、クライアントがクリティカル ポートに接続されていて、すでにゲスト VLAN が割り当てられている場合、スイッチはそのポートをゲスト VLAN に保持します。
- 制限付き VLAN : ポートがすでに制限付き VLAN で許可されていて RADIUS サーバーが使用できない場合、スイッチはクリティカルポートを制限付き VLAN でクリティカル認証ステートにします。
- 802.1x アカウンティング : RADIUS サーバーが使用できない場合、アカウンティングは影響を受けません。
- プライベート VLAN : プライベート VLAN ホストポートにアクセス不能認証バイパスを設定できます。アクセス VLAN は、セカンダリ VLAN でなければなりません。
- 音声 VLAN : アクセス不能認証バイパスは音声 VLAN と互換性がありますが、RADIUS 設定済み VLAN またはユーザー指定のアクセス VLAN は、音声 VLAN と異なっていなければなりません。
- Remote Switched Port Analyzer (RSPAN) : アクセス不能認証バイパスの RADIUS 設定済み VLAN またはユーザー指定のアクセス VLAN として RSPAN VLAN を指定しないでください。

スイッチ スタックで、次の動作が発生します。

- キープアライブパケットを送信することによって、スタックのアクティブスイッチにより、RADIUS サーバーのステータスがチェックされます。RADIUS サーバーのステータスが変更されると、アクティブスイッチからメンバスイッチへ、情報が送信されます。クリティカルポートの再認証時に、メンバスイッチにより、RADIUS サーバーのステータスがチェックされます。
- 新しいアクティブスイッチが選択されると、スイッチスタックと RADIUS サーバーとの間のリンクが変更される可能性があり、新しいアクティブスイッチにより、キープアライブパケットがただちに送信され、RADIUS サーバーのステータスがアップデートされます。サーバーのステータスが *dead* から *alive* に変化すると、スイッチはクリティカル認証ステートの状態にあるすべてのスイッチ ポートを再認証します。

メンバスイッチがスタックに追加されると、アクティブスイッチからメンバスイッチへサーバステータスが送信されます。

VLAN 割り当て、ゲスト VLAN、制限付き VLAN、アクセス不能認証バイパス

VLAN 割り当て、ゲスト VLAN、制限付き VLAN、およびアクセス不能認証バイパス設定時の注意事項は、次のとおりです。

- 802.1x 認証をポート上でイネーブルにすると、音声 VLAN の機能を持つポート VLAN は設定できません。
- RSPAN VLAN または音声 VLAN を除くあらゆる VLAN を、802.1x ゲスト VLAN として設定できます。ゲスト VLAN 機能は、内部 VLAN (ルーテッドポート) またはトランクポートではサポートされていません。アクセスポート上でだけサポートされます。
- DHCP クライアントが接続されている 802.1x ポートのゲスト VLAN を設定した後、DHCP サーバからホスト IP アドレスを取得する必要があります。クライアント上の DHCP プロセスが時間切れとなり DHCP サーバからホスト IP アドレスを取得しようとする前に、スイッチ上の 802.1x 認証プロセスを再起動する設定を変更できます。802.1x 認証プロセスの設定を減らします (**authentication timer inactivity** および **authentication timer reauthentication** インターフェイス コンフィギュレーション コマンド)。設定の減少量は、接続された 802.1x クライアントのタイプによって異なります。
- アクセス不能認証バイパス機能を設定する際には、次の注意事項に従ってください。
 - この機能はシングルホストモードおよびマルチホストモードの 802.1x ポートでサポートされます。
 - Windows XP を稼働しているクライアントに接続されたポートがクリティカル認証ステータスの場合、Windows XP はインターフェイスが認証されないと報告する場合があります。
 - Windows XP クライアントで DHCP が設定され、DHCP サーバからの IP アドレスがある場合、クリティカルポートで EAP 認証成功メッセージを受信しても DHCP 設定プロセスを再初期化しません。
 - アクセス不能認証バイパス機能および制限 VLAN を 802.1x ポート上に設定できます。スイッチが制限付き VLAN 内でクリティカルポートを再認証しようとし、すべての RADIUS サーバが利用不可能な場合、スイッチはポートステータスをクリティカル認証ステータスに変更し、制限付き VLAN に残ります。
- RSPAN VLAN または音声 VLAN を除くあらゆる VLAN を、802.1x 制限付き VLAN として設定できます。制限付き VLAN 機能は、内部 VLAN (ルーテッドポート) またはトランクポートではサポートされていません。アクセスポート上でだけサポートされます。

802.1x クリティカル音声 VLAN

ポートに接続されている IP フォンが Cisco Identity Services Engine (ISE) によって認証される際、その IP フォンは音声ドメインに参加します。ISE が到達不能である場合、スイッチはデバ

イスが音声デバイスなのかどうかを判断できません。サーバーが使用できない場合、電話機は音声ネットワークにアクセスできないため、動作できません。

データトラフィックの場合、アクセス不能認証バイパス（クリティカル認証）を設定し、サーバーが使用できない場合にトラフィックがネイティブ VLAN を通過できるようにすることができます。RADIUS 認証サーバーが使用できず（ダウンして）、アクセスできない認証バイパスがイネーブルの場合、スイッチは、クライアントにネットワークのアクセスを許可し、RADIUS 設定 VLAN またはユーザー指定アクセス VLAN でポートをクリティカル認証ステートにします。設定された RADIUS サーバーにスイッチが到達できず、新しいホストを認証できない場合、スイッチはこれらのホストをクリティカルポートに接続します。クリティカルポートに接続を試行している新しいホストは、ユーザー指定のアクセス VLAN（クリティカル VLAN）に移動され、制限付き認証を許可されます。



(注) クリティカル音声 VLAN のダイナミック割り当ては、ネストされたサービステンプレートではサポートされません。そのため、デバイスはループ内で VLAN を連続的に切り替えます。

authentication event server dead action authorize voice インターフェイス コンフィギュレーション コマンドを使用して、クリティカル音声 VLAN 機能を設定できます。ISE が応答しない場合、ポートはクリティカル認証モードになります。ホストからのトラフィックが音声 VLAN でタグ付けされると、接続デバイス（電話機）は、ポートに対して設定された音声 VLAN に配置されます。IP フォンは Cisco Discovery Protocol（シスコデバイス）や LLDP または DHCP を介して音声 VLAN ID を学習します。

switchport voice vlan vlan-id インターフェイス コンフィギュレーション コマンドを入力して、ポートの音声 VLAN を設定できます。

この機能は、マルチドメイン モードおよびマルチ認証ホスト モードでサポートされます。スイッチがシングルホスト モードまたはマルチホスト モードの場合にコマンドを入力できますが、デバイスがマルチドメインまたはマルチ認証ホスト モードに変わらない限りコマンドは無効になりません。

音声 VLAN ポートを使用した IEEE 802.1x 認証

音声 VLAN ポートは特殊なアクセス ポートで、次の 2 つの VLAN ID が対応付けられています。

- IP Phone との間で音声トラフィックを伝送する VVID。VVID は、ポートに接続された IP Phone を設定するために使用されます。
- IP Phone を通じて、スイッチと接続しているワークステーションとの間でデータトラフィックを伝送する PVID。PVID は、ポートのネイティブ VLAN です。

ポートの許可ステートにかかわらず、IP Phone は音声トラフィックに対して VVID を使用します。これにより、IP Phone は IEEE 802.1x 認証とは独立して動作できます。

シングルホストモードでは、IP Phone だけが音声 VLAN で許可されます。マルチホストモードでは、サブリカントが PVID で認証された後、追加のクライアントがトラフィックを音声

VLAN 上で送信できます。マルチ ホスト モードがイネーブルの場合、サブリカント認証は PVID と VVID の両方に影響します。

リンクがあるとき、音声 VLAN ポートはアクティブになり、IP Phone からの最初の Cisco Discovery Protocol メッセージを受け取るとデバイスの MAC アドレスが表示されます。Cisco IP Phone は、他のデバイスから受け取った Cisco Discovery Protocol メッセージをリレーしません。その結果、複数の IP Phone が直列に接続されている場合、スイッチは直接接続されている 1 台の IP Phone のみを認識します。音声 VLAN ポートで IEEE 802.1x 認証がイネーブルの場合、スイッチは 2 ホップ以上離れた認識されない IP Phone からのパケットをドロップします。

IEEE 802.1x 認証をスイッチ ポート上でイネーブルにすると、音声 VLAN でもあるアクセスポート VLAN を設定できます。

IP 電話がシングル ホスト モードで 802.1x 対応のスイッチ ポートに接続されている場合、スイッチは認証を行わずに電話ネットワーク アクセスを承認します。ポートで Multidomain Authentication (MDA) を使用して、データ デバイスと IP フォンなどの音声デバイスの両方を認証することを推奨します。



-
- (注) 音声 VLAN が設定され、Cisco IP Phone が接続されているアクセスポートで IEEE 802.1x 認証をイネーブルにした場合、Cisco IP Phone のスイッチへの接続が最大 30 秒間失われます。
-

WoL 機能を使用した IEEE 802.1x 認証

IEEE 802.1x 認証の Wake-on-LAN (WoL) 機能を使用すると、スイッチにマジックパケットと呼ばれる特定のイーサネットフレームを受信させて、休止状態の PC を起動させることができます。この機能は、管理者が休止状態のシステムへ接続しなければならない場合に役立ちます。

WoL を使用するホストが IEEE 802.1x ポートを通じて接続され、ホストの電源がオフになると、IEEE 802.1x ポートは無許可になります。無許可になったポートは EAPOL パケットしか送信できないため、WoL マジックパケットはホストに届きません。さらに PC が休止状態になると、PC が認証されなくなるため、スイッチポートは閉じたままになります。

スイッチが WoL 機能を有効にした IEEE 802.1x 認証を使用している場合、スイッチはマジックパケットを含むトラフィックを無許可の IEEE 802.1x ポートに転送します。ポートが無許可の間、スイッチは EAPOL パケット以外の入力トラフィックをブロックし続けます。ホストはパケットを受信できますが、パケットをネットワーク内にある他のデバイスに送信できません。



-
- (注) PortFast がポートでイネーブルになっていないと、そのポートは強制的に双方向ステートになります。
-

authentication control-direction in インターフェイス コンフィギュレーション コマンドを使用してポートを単一方向として設定すると、ポートはスパンニングツリーフォワーディングステートに変更されます。ポートは、ホストにパケットを送信できますが、受信はできません。

authentication control-direction both インターフェイス コンフィギュレーション コマンドを使用してポートを双方向として設定すると、ポートは、両方向でアクセスコントロールされます。ポートは、ホストとの間でパケットを送受信しません。

柔軟な認証の順序設定

柔軟な認証の順序設定を使用して、ポートが新しいホストを認証するときを使用する方法の順序を設定できます。The IEEE 802.1X の柔軟な認証機能では、以下の3つの認証方法をサポートしています。

- dot1X : IEEE 802.1X 認証はレイヤ 2 の認証方式です。
- mab : MAC 認証バイパスはレイヤ 2 の認証方式です。
- webauth : Web 認証はレイヤ 3 の認証方式です。

この機能を使用すると、各ポートでどの認証方式を使用するかを制御できます。また、そのポートの方式についてフェールオーバー順も制御できます。たとえば、MAC 認証バイパスおよび 802.1x は、プライマリまたはセカンダリ認証方法として使用し、Web 認証は、これらの認証のいずれか、または両方が失敗した場合のフォールバック方法として使用できます。

The IEEE 802.1X の柔軟な認証機能では、以下のホストモードをサポートしています。

- multi-auth : マルチ認証では、音声 VLAN に 1 つの認証、データ VLAN に複数の認証を使用できます。
- multi-domain : マルチドメイン認証では、音声 VLAN に 1 つ、データ VLAN に 1 つ、計 2 つの認証を使用できます。

Open1x 認証

Open1x 認証によって、デバイスが認証される前に、そのデバイスがポートにアクセスできるようになります。オープン認証が設定されている場合、新しいホストはポートに定義されているアクセスコントロールリスト (ACL) に基づいてトラフィックを渡します。ホストが認証されると、RADIUS サーバに設定されているポリシーがそのホストに適用されます。

オープン認証を次の状況で設定できます。

- シングルホストモードでのオープン認証 : 1 人のユーザーだけが認証の前後にネットワークにアクセスできます。
- MDA モードでのオープン認証 : 音声ドメインの 1 人のユーザーだけ、およびデータドメインの 1 人のユーザーだけが許可されます。
- マルチホストモードでのオープン認証 : 任意のホストがネットワークにアクセスできます。
- 複数認証モードでのオープン認証 : MDA の場合と似ていますが、複数のホストを認証できます。



- (注) オープン認証が設定されている場合は、他の認証制御よりも優先されます。これは、**authentication open** インターフェイス コンフィギュレーション コマンドを使用した場合、**authentication port-control** インターフェイス コンフィギュレーション コマンドに関係なく、ポートがホストにアクセス権を付与することを意味します。

マルチドメイン認証

スイッチはマルチドメイン認証 (MDA) をサポートしています。これにより、データ装置と IP Phone などの音声装置の両方を同じスイッチポート上で認証できます。ポートはデータドメインと音声ドメインに分割されます。



- (注) すべてのホスト モードで、ポートベース認証が設定されている場合、ラインプロトコルは許可の前にアップのままです。

MDA では、デバイス認証の順序が指定されません。ただし、最適な結果を得るには、MDA 対応のポート上のデータ デバイスよりも前に音声デバイスを認証することを推奨します。

MDA を設定するときには、次の注意事項に従ってください。

- MDA のスイッチ ポートを設定する必要があります。
- ホスト モードがマルチドメインに設定されている場合、IP Phone の音声 VLAN を設定する必要があります。
- MDA 対応ポートでの音声 VLAN の割り当てはサポートされています。
- 音声デバイスを認可するには、値を *device-traffic-class=voice* に設定した Cisco 属性値 (AV) ペア属性を送信するように AAA サーバーを設定する必要があります。この値を使用しない場合、音声デバイスはデータ デバイスとして扱われます。



- (注) *traffic-class=voice* が AAA サーバーから *service-template* としてダウンロードされると、音声ドメインではなくデータドメインでセッションが作成されます。

- ゲスト VLAN および制限付き VLAN 機能は、MDA 対応のポートのデータ デバイスだけに適用されます。許可に失敗した音声デバイスは、データ デバイスとして扱われます。
- 複数のデバイスでポートの音声またはデータ ドメインの許可を行おうとすると、*errordisable* になります。

- デバイスが許可されるまで、ポートはそのトラフィックをドロップします。他社製 IP Phone または音声デバイスはデータおよび音声 VLAN の両方に許可されます。データ VLAN では、音声デバイスを DHCP サーバーに接続して IP アドレスおよび音声 VLAN 情報を取得することができます。音声デバイスが音声 VLAN で送信を開始すると、データ VLAN へのアクセスはブロックされます。
- データ VLAN とバインドしている音声デバイス MAC アドレスは、ポート セキュリティ MAC アドレス制限にカウントされません。
- MDA では、IEEE 802.1x 認証をサポートしていないデバイスへのスイッチポートの接続を許可するフォールバック メカニズムとして、MAC 認証バイパスを使用できます。
- データまたは音声デバイスがポートで検出されると、認証に成功するまでその MAC アドレスがブロックされます。許可に失敗した場合、MAC アドレスが 5 分間ブロックされたままになります。
- ポートが未認証中に 6 つ以上のデバイスがデータ VLAN で検出された場合や、複数の音声デバイスが音声 VLAN で検出された場合、ポートは `errdisable` になります。
- ポートのホスト モードをシングルホスト モードまたはマルチホスト モードからマルチドメイン モードに変更すると、ポートでは許可されたデータ デバイスは許可されたままになります。ただし、ポートの音声 VLAN で許可されている Cisco IP Phone は自動的に削除されるので、そのポートでは再認証を行う必要があります。
- ゲスト VLAN や制限付き VLAN などのアクティブ フォールバック メカニズムは、ポートをシングル モードまたはマルチホスト モードからマルチドメイン モードに変更したあとでも設定されたままになります。
- ポートのホスト モードをマルチドメイン モードからシングル モードまたはマルチホスト モードに変更すると、許可されているすべてのデバイスがポートから削除されます。
- まずデータ ドメインを許可してゲスト VLAN に参加させる場合、IEEE 802.1x 非対応の音声デバイスは、音声 VLAN のパケットをタグ付けして、認証を開始する必要があります。
- MDA 対応ポートでは、ユーザー単位 ACL を推奨しません。ユーザー単位 ACL ポリシーを備えた、許可されたデバイスは、ポートの音声 VLAN とデータ VLAN の両方のトラフィックに影響を与えることがあります。このようなデバイスを使用する場合は、ポートでユーザー単位 ACL を適用するデバイスは 1 台だけにしてください。

Network Edge Access Topology を使用した 802.1x サプリカントおよびオーセンティケータスイッチ

Network Edge Access Topology (NEAT) 機能は、ワイヤリング クローゼット (会議室など) 外の領域まで識別を拡張します。これにより、任意のタイプのデバイスをポートで認証できます。

- 802.1x スイッチ サプリカント : 802.1x サプリカント機能を使用することで、別のスイッチの サプリカントとして機能するようにスイッチを設定できます。この設定は、たとえば、スイッチがワイヤリング クローゼット外にあり、トランクポートを介してアップストリー

ムスイッチに接続される場合に役に立ちます。802.1x スイッチ サプリカント機能を使用して設定されたスイッチは、セキュアな接続のためにアップストリームスイッチで認証します。サプリカント スイッチが認証に成功すると、オーセンティケータ スイッチでポートモードがアクセスからトランクに変更されます。サプリカント スイッチでは、CISP を有効にするときに手動でトランクを設定する必要があります。

- アクセス VLAN は、オーセンティケータ スイッチで設定されている場合、認証が成功した後にトランク ポートのネイティブ VLAN になります。

デフォルトでは、BPDU ガードが有効にされたオーセンティケータ スイッチにサプリカントのスイッチを接続する場合、オーセンティケータのポートはサプリカントスイッチが認証する前にスパンニングツリープロトコル (STP) のブリッジプロトコルデータユニット (BPDU) を受信した場合、errdisable 状態になる可能性があります。認証中にサプリカントのポートから送信されるトラフィックを制御できます。**dot1x supplicant controlled transient** グローバル コンフィギュレーション コマンドを入力すると、認証が完了する前にオーセンティケータ ポートがシャットダウンすることがないように、認証中に一時的にサプリカントのポートをブロックします。認証に失敗すると、サプリカントのポートが開きます。**no dot1x supplicant controlled transient** グローバル コンフィギュレーション コマンドを入力すると、認証期間中にサプリカント ポートが開きます。これはデフォルトの動作です。

BPDU ガードが **spanning-tree bpduguard enable** インターフェイス コンフィギュレーション コマンドによりオーセンティケータのスイッチ ポートで有効になっている場合、サプリカント スイッチで **dot1x supplicant controlled transient** コマンドを使用することを強く推奨します。



- (注) **spanning-tree portfast bpduguard default** グローバル コンフィギュレーション コマンドを使用して、グローバルにオーセンティケータ スイッチで BPDU ガードを有効にした場合、**dot1x supplicant controlled transient** コマンドを入力すると、BPDU の違反が避けられなくなります。

1 つ以上のサプリカント スイッチに接続するオーセンティケータ スイッチ インターフェイスで MDA または multiauth モードをイネーブルにできます。マルチホストモードはオーセンティケータ スイッチ インターフェイスではサポートされていません。

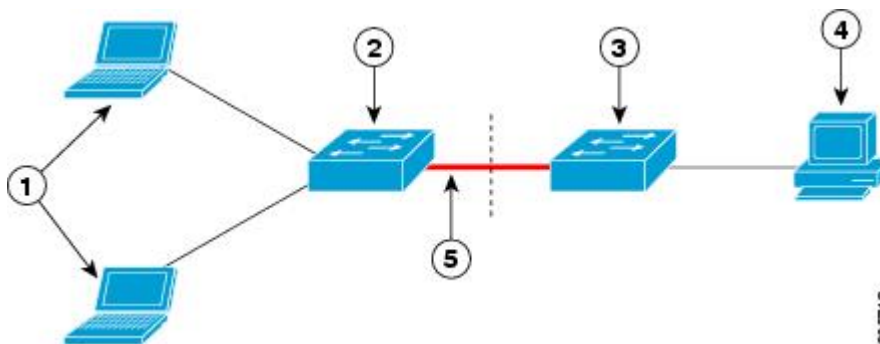
インターフェイスで有効になっているシングルホスト モードでオーセンティケータ スイッチをリブートすると、インターフェイスが認証前に err-disabled 状態に移行する場合があります。err-disabled 状態から回復するには、オーセンティケータ ポートをフラップしてインターフェイスを再度アクティブにし、認証を開始します。

すべてのホストモードで機能するように **dot1x supplicant force-multicast** グローバル コンフィギュレーション コマンドを Network Edge Access Topology (NEAT) のサプリカントスイッチで使用します。

- ホスト許可：許可済み (サプリカントでスイッチに接続する) ホストからのトラフィックだけがネットワークで許可されます。これらのスイッチは、Client Information Signalling Protocol (CISP) を使用して、サプリカント スイッチに接続する MAC アドレスをオーセンティケータ スイッチに送信します。

- 自動有効化：オーセンティケータ スイッチでのトランク コンフィギュレーションを自動的に有効化します。これにより、サブリカントスイッチから着信する複数の VLAN のユーザー トラフィックが許可されます。ISE で `cisco-av-pair` を `device-traffic-class=switch` として設定します（この設定は `group` または `user` 設定で行うことができます）。

図 27: CISP を使用したオーセンティケータまたはサブリカントスイッチ



1	ワークステーション (クライアント)	2	サブリカント スイッチ (ワイヤリング クローゼット外)
3	オーセンティケータ スイッチ	4	Cisco ISE
5	トランク ポート		



- (注) **switchport nonegotiate** コマンドは、NEAT を使用したサブリカントおよびオーセンティケータ スイッチではサポートされません。このコマンドは、トポロジのサブリカント側で設定しないでください。オーセンティケータサーバ側で設定した場合は、内部マクロによってポートからこのコマンドが自動的に削除されます。

802.1x ユーザ ディストリビューション

802.1x ユーザ ディストリビューションを設定すると、複数の異なる VLAN で同じグループ名のユーザのロード バランシングを行うことができます。

VLAN は、RADIUS サーバにより提供されるか、VLAN グループ名でスイッチ CLI を介して設定します。

- RADIUS サーバを設定して、ユーザの複数の VLAN 名を送信します。複数の VLAN 名は、ユーザへの応答の一部として送信できます。802.1x ユーザ ディストリビューションは、特定の VLAN のすべてのユーザを追跡し、許可されたユーザをユーザ数が最も少ない VLAN に移動することでロード バランシングを行います。
- RADIUS サーバを設定してユーザの VLAN グループ名を送信します。VLAN グループ名は、ユーザへの応答の一部として送信できます。スイッチ CLI を使用して設定した VLAN

グループ名で、選択された VLAN グループ名を検索できます。VLAN グループ名が検出されると、この VLAN グループ名で対応する VLAN を検索して、ユーザ数が最も少ない VLAN が検出されます。ロード バランシングは、対応する許可済みユーザをその VLAN に移動することで行われます。

RADIUS サーバーが、承認の結果として属性の VLAN グループ名を送信するたびに、グループの中で最もユーザー数の少ない VLAN がエンドユーザーに割り当てられます。再認証の場合（認証セッションが存在する）、および CoA の場合（セッションアライブ）、グループ内で最もユーザー数の少ない VLAN でなくても、同じ VLAN が維持されます。



(注) RADIUS サーバーは、VLAN-ID、VLAN 名または VLAN グループを任意に組み合わせて VLAN 情報を送信できます。

802.1x ユーザ ディストリビューションの設定時の注意事項

- 少なくとも1つの VLAN が VLAN グループにマッピングされることを確認してください。
- 複数の VLAN を VLAN グループにマッピングできます。
- VLAN を追加または削除することで、VLAN グループを変更できます。
- 既存の VLAN を VLAN グループ名からクリアする場合、VLAN の認証済みポートはクリアされませんが、既存の VLAN グループからマッピングが削除されます。
- 最後の VLAN を VLAN グループ名からクリアすると、VLAN グループがクリアされます。
- アクティブ VLAN がグループにマッピングされても VLAN グループをクリアできます。VLAN グループをクリアすると、グループ内で任意の VLAN の認証状態であるポートまたはユーザはクリアされませんが、VLAN の VLAN グループへのマッピングはクリアされます。

Network Admission Control レイヤ 2 IEEE 802.1x 検証

スイッチは、デバイスのネットワーク アクセスを許可する前にエンドポイント システムやクライアントのウイルス対策の状態またはポスチャを調べる Network Admission Control (NAC) レイヤ 2 IEEE 802.1x 検証をサポートしています。NAC レイヤ 2 IEEE 802.1x 検証を使用すると、以下の作業を実行できます。

- Session-Timeout RADIUS 属性（属性 [27]）と Termination-Action RADIUS 属性（属性 [29]）を認証サーバーからダウンロードします。
- Session-Timeout RADIUS 属性（属性 [27]）の値として再認証試行間の秒数を指定し、RADIUS サーバーからクライアントのアクセス ポリシーを取得します。

- スイッチが Termination-Action RADIUS 属性（属性[29]）を使用してクライアントを再認証する際のアクションを設定します。値が *DEFAULT* であるか、値が設定されていない場合、セッションは終了します。値が RADIUS 要求の場合、再認証プロセスが開始します。
- VLAN の番号や名前、または VLAN グループ名のリストを Tunnel Group Private ID（属性 [81]）の値として設定し、VLAN の番号や名前、または VLAN グループ名のプリファレンスを Tunnel Preference（属性 [83]）の値として設定します。Tunnel Preference を設定しない場合、最初の Tunnel Group Private ID（属性 [81]）属性がリストから選択されます。
- NAC ポスチャトークンを表示します。これは、**show authentication** 特権 EXEC コマンドを使用して、クライアントのポスチャを示します。
- ゲスト VLAN としてセカンダリ プライベート VLAN を設定します。

NAC レイヤ 2 IEEE 802.1x 検証の設定は、RADIUS サーバーにポスチャ トークンを設定する必要があることを除いて、IEEE 802.1x ポートベース認証と似ています。

音声認識 802.1x セキュリティ



(注) 音声認識 IEEE 802.1x 認証を使用するには、スイッチが LAN Base イメージを実行している必要があります。

音声認識 802.1x セキュリティ機能を使用して、セキュリティ違反が発生した場合にデータまたは音声 VLAN に関係なく VLAN だけをディセーブルにするようにスイッチを設定します。以前のリリースでは、セキュリティ違反の原因であるデータ クライアントを認証しようとする、ポート全体がシャットダウンし、接続が完全に切断されます。

この機能は、PC が IP Phone に接続されている IP Phone 環境で使用できます。データ VLAN でセキュリティ違反が検出されると、データ VLAN だけがシャットダウンされます。音声 VLAN のトラフィックは中断することなくスイッチで送受信されます。

コモンセッション ID

認証マネージャは、使用された認証方式が何であれ、クライアントの単一のセッション ID（共通セッション ID）を使用します。この ID は、表示コマンドや MIB などのすべてのレポートに使用されます。セッション ID は、セッション単位のすべての Syslog メッセージに表示されません。

セッション ID には、次の情報が含まれます。

- ネットワーク アクセス デバイス (NAD) の IP アドレス
- 一意の 32 ビット整数（機械的に増加します）
- セッション開始タイム スタンプ（32 ビット整数）

次に、`show authentication` コマンドの出力に表示されたセッション ID の例を示します。この例では、セッション ID は `160000050000000B288508E5` です。

```
Device# show authentication sessions
Interface  MAC Address      Method  Domain  Status      Session ID
Fa4/0/4    0000.0000.0203    mab     DATA   Authz Success 160000050000000B288508E5
```

次に、Syslog 出力にセッション ID が表示される例を示します。この例でも、セッション ID は `160000050000000B288508E5` です。

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on Interface
Fa4/0/4 AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client
(0000.0000.0203) on Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

セッション ID は、NAD、AAA サーバー、その他のレポート分析アプリケーションでクライアントを識別するために使用されます。ID は自動的に表示されます。設定は必要ありません。

ポートあたりのデバイスの最大数

802.1x 対応のポートに接続できるデバイスの最大数です。

- シングル ホスト モードの場合、アクセス VLAN で接続できるデバイスは 1 台だけです。ポートが音声 VLAN でも設定されている場合、音声 VLAN を介して送受信できる Cisco IP Phone の数には制限はありません。
- マルチドメイン認証 (MDA) モードの場合、アクセス VLAN で 1 台のデバイス、音声 VLAN で 1 台の IP Phone が許可されます。
- マルチホストモードでは、1 つの 802.1x サプリカントだけがポートで許可されますが、非 802.1x ホストは数に制限なく、アクセス VLAN で許可されます。音声 VLAN で許可されるデバイスの数には制限はありません。

802.1x ポートベース認証の設定方法

802.1X 認証の設定

ユーザー単位 ACL または VLAN 割り当てを可能にするには、AAA 許可をイネーブルにしてネットワーク関連のすべてのサービス要求に対してスイッチを設定する必要があります。

次に、802.1x の AAA プロセスを示します。

始める前に

802.1x ポートベース認証を設定するには、認証、許可、アカウントिंग（AAA）をイネーブルにして認証方式リストを指定する必要があります。方式リストは、ユーザ認証のためにクエリー送信を行う手順と認証方式を記述したものです。

手順の概要

1. ユーザーがスイッチのポートに接続します。
2. 認証が実行されます。
3. RADIUS サーバ設定に基づいて、VLAN 割り当てが適宜イネーブルになります。
4. スイッチが開始メッセージをアカウントिंग サーバーに送信します。
5. 必要に応じて、再認証が実行されます。
6. スイッチが仮のアカウントングアップデートを、再認証結果に基づいたアカウントングサーバーに送信します。
7. ユーザーがポートから切断します。
8. スイッチが停止メッセージをアカウントングサーバーに送信します。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	ユーザーがスイッチのポートに接続します。	
ステップ 2	認証が実行されます。	
ステップ 3	RADIUS サーバ設定に基づいて、VLAN 割り当てが適宜イネーブルになります。	
ステップ 4	スイッチが開始メッセージをアカウントングサーバーに送信します。	
ステップ 5	必要に応じて、再認証が実行されます。	
ステップ 6	スイッチが仮のアカウントングアップデートを、再認証結果に基づいたアカウントングサーバーに送信します。	
ステップ 7	ユーザーがポートから切断します。	
ステップ 8	スイッチが停止メッセージをアカウントングサーバーに送信します。	

802.1x ポートベース認証の設定

802.1x ポートベースの認証を設定するには、次の手順を実行します。

手順の概要

1. **enable**

2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication dot1x{ default } method1**
5. **dot1x system-auth-control**
6. **aaa authorization network {default} group radius**
7. **radius server server name**
8. **address {ipv4 | ipv6} ip address**
9. **key string**
10. **exit**
11. **interface interface-id**
12. **switchport mode access**
13. **authentication port-control auto**
14. **dot1x pae authenticator**
15. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例 : Device (config)# aaa new-model	AAA をイネーブルにします。
ステップ 4	aaa authentication dot1x{ default } method1 例 : Device (config)# aaa authentication dot1x default group radius	802.1x 認証方式リストを作成します。 authentication コマンドにリストが指定されていない場合に使用するデフォルトのリストを作成するには、 default キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。 method1 には、 group radius キーワードを入力して、認証用のすべての RADIUS サーバリストを使用できるようにします。

	コマンドまたはアクション	目的
		(注) コマンドラインのヘルプストリングには他のキーワードも表示されますが、サポートされるのは group radius キーワードのみです。
ステップ 5	dot1x system-auth-control 例： Device(config)# dot1x system-auth-control	スイッチで 802.1x 認証をグローバルに有効にします。
ステップ 6	aaa authorization network {default} group radius 例： Device(config)# aaa authorization network default group radius	(任意) ユーザー単位 ACL や VLAN 割り当てなど、ネットワーク関連のすべてのサービス要求に対するユーザー RADIUS 許可をスイッチに設定します。
ステップ 7	radius server server name 例： Device(config)# radius server rsim address ipv4 124.2.2.12	(任意) RADIUS サーバーの IP アドレスを指定します。
ステップ 8	address {ipv4 ipv6} ip address 例： Device(config-radius-server)# address ipv4 10.0.1.12	RADIUS サーバーの IP アドレスを設定します。
ステップ 9	key string 例： Device(config-radius-server)# key rad123	(任意) RADIUS サーバー上で動作する RADIUS デーモンとスイッチの間で使用する認証および暗号キーを指定します。
ステップ 10	exit 例： Device(config-radius-server)# exit	RADIUS サーバーモードを終了して、グローバルコンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 11	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 1/0/2	IEEE 802.1x 認証をイネーブルにするクライアントに接続しているポートを指定し、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 12	switchport mode access 例： Device(config-if)# switchport mode access	(任意) ステップ 6 および 7 で RADIUS サーバを設定した場合のみ、ポートをアクセス モードに設定します。
ステップ 13	authentication port-control auto 例： Device(config-if)# authentication port-control auto	ポートでの 802.1x 認証を有効にします。
ステップ 14	dot1x pae authenticator 例： Device(config-if)# dot1x pae authenticator	インターフェイスのポート アクセス エンティティを、オーセンティケータとしてのみ動作し、サブリカント用のメッセージは無視するように設定します。
ステップ 15	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

定期的な再認証の設定

802.1x クライアントの定期的な再認証をイネーブルにし、再認証の間隔を指定できます。再認証を行う間隔を指定しない場合、3600 秒おきに再認証が試みられます。

クライアントの定期的な再認証を有効にし、再認証が行われるまでの間隔 (秒) を設定するには、次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **authentication periodic**
5. **authentication timer** {{{[inactivity | reauthenticate | restart | unauthorized]}} {value}}

6. end

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例 : Device(config)# interface gigabitethernet 2/0/1	設定するポートを指定し、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 4	authentication periodic 例 : Device(config-if)# authentication periodic	クライアントの定期的な再認証 (デフォルトではディセーブル) をイネーブルにします。 (注) デフォルト値は 3600 秒です。再認証タイマーの値を変更するか、スイッチに RADIUS-provided セッションタイムアウトを使用させるには、 authentication timer reauthenticate コマンドを入力します。
ステップ 5	authentication timer {[inactivity reauthenticate restart unauthorized]} {value}} 例 : Device(config-if)# authentication timer reauthenticate 180	再認証の試行の間隔 (秒) を設定します。 authentication timer キーワードの意味は次のとおりです。 <ul style="list-style-type: none"> inactivity : クライアントからのアクティビティがなくなり無許可になるまでの間隔 (秒) reauthenticate : 自動再認証試行が開始されるまでの時間 (秒) restart value : 無許可ポートの認証の試行が行われるまでの間隔 (秒) unauthorized value : 不正セッションが削除されるまでの間隔 (秒)

	コマンドまたはアクション	目的
		このコマンドがスイッチの動作に影響するのは、定期的な再認証をイネーブルに設定した場合だけです。
ステップ 6	end 例： Device(config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

802.1x 違反モードの設定

次に示す状況で、シャットダウン、Syslog エラーを生成、または新しいデバイスからのパケットを廃棄するように 802.1x ポートを設定できます。

- デバイスが 802.1x 対応のポートに接続した
- ポートで認証されるデバイスの最大数に達した

スイッチ上にセキュリティ違反アクションを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication dot1x { default } method1**
5. **interface interface-type interface-number**
6. **switchport mode access**
7. **authentication violation {shutdown | restrict | protect | replace}**
8. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	Device# <code>configure terminal</code>	
ステップ 3	aaa new-model 例： Device(config)# <code>aaa new-model</code>	AAA をイネーブルにします。
ステップ 4	aaa authentication dot1x{ default } method1 例： Device(config)# <code>aaa authentication dot1x default group radius</code>	802.1x 認証方式リストを作成します。 authentication コマンドにリストが指定されていない場合に使用するデフォルトのリストを作成するには、 default キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。 <i>method1</i> には、 group radius キーワードを入力して、認証用のすべての RADIUS サーバリストを使用できるようにします。
ステップ 5	interface interface-type interface-number 例： Device(config)# <code>interface gigabitethernet 1/0/4</code>	IEEE 802.1x 認証をイネーブルにするクライアントに接続しているポートを指定し、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	switchport mode access 例： Device(config-if)# <code>switchport mode access</code>	ポートをアクセス モードに設定します。
ステップ 7	authentication violation {shutdown restrict protect replace} 例： Device(config-if)# <code>authentication violation restrict</code>	違反モードを設定します。キーワードの意味は次のとおりです。 <ul style="list-style-type: none"> • shutdown : エラーによってポートがディセーブルになります。 • restrict : Syslog エラーを生成します。 • protect : トラフィックをポートに送信するすべての新しいデバイスからパケットをドロップします。 • replace : 現在のセッションを削除し、新しいホストで認証します。

	コマンドまたはアクション	目的
ステップ 8	end 例 : Device(config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

待機時間の変更

スイッチはクライアントを認証できなかった場合に、所定の時間だけアイドル状態を続け、その後再び認証を試みます。**authentication timer restart** インターフェイス コンフィギュレーション コマンドは、アイドル状態の期間を制御します。認証が失敗する理由としては、クライアントが無効なパスワードを提示した場合などが考えられます。デフォルトよりも小さい値を入力することによって、ユーザへの応答時間を短縮できます。

待機時間を変更するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **authentication timer restart seconds**
5. **end**
6. **show authentication sessions interface interface-id**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例 :	設定するポートを指定し、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。

	コマンドまたはアクション	目的
	Device(config)# interface gigabitethernet 2/0/1	
ステップ 4	authentication timer restart seconds 例： Device(config-if)# authentication timer restart 30	クライアントとの認証のやり取りに失敗した場合に、スイッチが待機状態のままの秒数を設定します。 指定できる範囲は 1 ~ 65535 秒です。デフォルトは 60 秒です。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show authentication sessions interface interface-id 例： Device# show authentication sessions interface gigabitethernet2/0/1	入力を確認します。
ステップ 7	copy running-config startup-config 例： Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

スイッチからクライアントへの再送信時間の変更

クライアントはスイッチからの EAP-Request/Identity フレームに対し、EAP-Response/Identity フレームで応答します。スイッチがこの応答を受信できなかった場合、所定の時間（再送信時間）だけ待機し、その後フレームを再送信します。



(注) このコマンドのデフォルト値は、リンクの信頼性が低下した場合や、特定のクライアントおよび認証サーバーの動作に問題がある場合など、異常な状況に対する調整を行う必要があるときに限って変更してください。

スイッチがクライアントからの通知を待機する時間を変更するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. enable

2. **configure terminal**
3. **interface *interface-id***
4. **authentication timer reauthenticate *seconds***
5. **end**
6. **show authentication sessions interface *interface-id***
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet2/0/1	設定するポートを指定し、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 4	authentication timer reauthenticate <i>seconds</i> 例： Device(config-if)# authentication timer reauthenticate 60	スイッチが EAP-Request/Identity フレームに対するクライアントからの応答を待ち、要求を再送信するまでの秒数を設定します。 指定できる範囲は 1 ～ 65535 秒です。デフォルトは 5 秒です。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show authentication sessions interface <i>interface-id</i> 例： Device# show authentication sessions interface gigabitethernet 2/0/1	入力を確認します。

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

スイッチからクライアントへのフレーム再送信回数の設定

スイッチからクライアントへの再送信時間を変更できるだけでなく、(クライアントから応答が得られなかった場合に) スイッチが認証プロセスを再起動する前に、クライアントに EAP-Request/Identity フレームを送信する回数を変更できます。



(注) このコマンドのデフォルト値は、リンクの信頼性が低下した場合や、特定のクライアントおよび認証サーバーの動作に問題がある場合など、異常な状況に対する調整を行う必要があるときに限って変更してください。

スイッチからクライアントへのフレーム再送信回数を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **dot1x max-reauth-req count**
5. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	interface interface-id 例 : Device(config)# interface gigabitethernet2/0/1	設定するポートを指定し、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 4	dot1x max-reauth-req count 例 : Device(config-if)# dot1x max-reauth-req 5	スイッチが認証処理を再開するまでに、クライアントへ EAP 要求/アイデンティティ フレームを送信する回数を変更できます。指定できる範囲は 1 ~ 10 です。デフォルトは 2 です。
ステップ 5	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ホストモードの設定

authentication port-control インターフェイス コンフィギュレーション コマンドが **auto** に設定されている IEEE 802.1x 許可ポート上で、複数のホスト（クライアント）を許可するには、特権 EXEC モードで次の手順を実行します。マルチドメイン認証（MDA）を設定してイネーブルにするには、**multi-domain** キーワードを使用します。これにより、ホストデバイス、および IP Phone（シスコ製または他社製）など音声デバイスの両方が同じスイッチポートで許可されます。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **authentication host-mode[multi-auth |multi-domain |multi-host |single-host]**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例 : Device(config)# interface gigabitethernet 2/0/1	複数ホストが間接的に接続されているポートを指定し、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 4	authentication host-mode[multi-auth multi-domain multi-host single-host] 例 : Device(config-if)# authentication host-mode multi-host	<p>単一の 802.1x 許可ポートで複数のホスト（クライアント）を許可することができます。</p> <p>キーワードの意味は次のとおりです。</p> <ul style="list-style-type: none"> • multi-auth : 音声 VLAN とデータ VLAN の両方で複数の認証クライアントを許可します。 (注) multi-auth キーワードは、authentication host-mode コマンドでのみ使用できます。 • multi-host : シングル ホストの認証後に 802.1x 許可ポートで複数のホスト（クライアント）の接続を許可します。 • multi-domain : ホスト デバイスと IP Phone（シスコ製または他社製）など音声デバイスの両方が、IEEE 802.1x 許可ポートで認証されるようにします。 (注) ホストモードが multi-domain に設定されている場合、IP Phone の音声 VLAN を設定する必要があります。 <p>指定のインターフェイスに対し authentication port-control インターフェイス コンフィギュレーション コマンドが auto に設定されていることを確認してください。</p>
ステップ 5	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

MAC 移動のイネーブル化

MAC 移動を使用すると、認証されたホストをデバイスのポート間で移動できます。

デバイスで MAC 移動をグローバルに有効にするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **authentication mac-move permit**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	authentication mac-move permit 例： Device(config)# authentication mac-move permit	デバイスで MAC 移動を有効にします。デフォルトは deny です。 セッション認識型ネットワークモードでは、デフォルト CLI は access-session mac-move deny です。セッション認識型ネットワークで MAC 移動をイネーブルにするには、 no access-session mac-move グローバルコンフィギュレーションコマンドを使用します。 mac-move のデフォルト値は、レガシーモード (IBNS 1.0) の場合は deny で、C3PL モード (IBNS 2.0) の場合は permit です。
ステップ 4	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

MAC 置換のイネーブル化

MAC 置換を使用すると、ホストはポート上の認証ホストを置換できます。

インターフェイス上で MAC 置換をイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **authentication violation** {**protect** | **replace** | **restrict** | **shutdown**}
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Devic (config)# interface gigabitethernet2/0/2	設定するポートを指定し、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 4	authentication violation { protect replace restrict shutdown } 例： Device (config-if)# authentication violation replace	インターフェイス上で MAC 置換をイネーブルにするには、 replace キーワードを使用します。ポートが現在のセッションを削除し、新しいホストを使用して認証を開始します。 他のキーワードは、次のような機能があります。 <ul style="list-style-type: none"> • protect : ポートは、システムメッセージを生成せずに、予期しない MAC を使用するパケットをドロップします。 • restrict : 違反パケットが CPU によってドロップされ、システムメッセージが生成されます。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • shutdown : ポートは、予期しない MAC アドレスを受信すると error disabled になります。
ステップ 5	end 例 : Device(config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

802.1x アカウンティングの設定

802.1x アカウンティングを使用して、AAA システム アカウンティングをイネーブルにすると、ロギングのためにシステム リロード イベントをアカウンティング RADIUS サーバに送信できます。サーバは、アクティブな 802.1x セッションすべてが終了したものと判断します。



- (注) Cisco IOS XE Everest 16.6.x では、定期的な AAA アカウンティングのアップデートはサポートされていません。スイッチは、定期的中間アカウンティングレコードをアカウンティングサーバに送信しません。定期的な AAA アカウンティングのアップデートは、Cisco IOS XE Fuji 16.9.x 以降のリリースで利用できます。

RADIUS は信頼性の低い UDP トランスポート プロトコルを使用するため、ネットワーク状態が良好でないと、アカウンティングメッセージが失われることがあります。設定した回数のアカウンティング要求の再送信後、スイッチが RADIUS サーバからアカウンティング応答メッセージを受信しない場合、次のメッセージが表示されます。

```
Accounting message %s for session %s failed to receive Accounting Response.
```

このストップ メッセージが正常に送信されない場合、次のメッセージが表示されます。

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```



- (注) ロギングの開始、停止、仮のアップデート メッセージ、タイム スタンプなどのアカウンティングタスクを実行するように、RADIUS サーバを設定する必要があります。これらの機能をオンにするには、RADIUS サーバの [Network Configuration] タブの [Update/Watchdog packets from this AAA client] のロギングをイネーブルにします。次に、RADIUS サーバの [System Configuration] タブの [CVS RADIUS Accounting] をイネーブルにします。

AAA がスイッチでイネーブルになった後、802.1x アカウンティングを設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **aaa accounting dot1x default start-stop group radius**
5. **aaa accounting system default start-stop group radius**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 1/0/3	設定するポートを指定し、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 4	aaa accounting dot1x default start-stop group radius 例： Device(config-if)# aaa accounting dot1x default start-stop group radius	すべての RADIUS サーバのリストを使用して 802.1x アカウンティングをイネーブルにします。
ステップ 5	aaa accounting system default start-stop group radius 例： Device(config-if)# aaa accounting system default start-stop group radius	(任意) システムアカウンティングをイネーブルにし (すべての RADIUS サーバのリストを使用)、スイッチがリロードするときにシステム アカウンティング リロード イベント メッセージを生成します。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

802.1x 準備状態チェックの設定

802.1x 準備状態チェックは、すべてのスイッチ ポートの 802.1x アクティビティをモニタリングし、802.1x をサポートするポートに接続されているデバイスの情報を表示します。この機能を使用して、スイッチ ポートに接続されているデバイスが 802.1x に対応できるかどうかを判別できます。

802.1x 準備状態チェックは、802.1x で設定できるすべてのポートで使用できます。準備状態チェックは、**dot1x force-unauthorized** として設定されるポートでは使用できません。

802.1x 準備状態チェックをスイッチでイネーブルにする場合には、次の手順に従ってください。

始める前に

準備状態チェックをスイッチでイネーブルにする場合、次の注意事項に従ってください。

- 準備状態チェックは通常、802.1x がスイッチでイネーブルにされる前に使用されます。
- インターフェイスを指定せずに **dot1x test eapol-capable** 特権 EXEC コマンドを使用すると、スイッチ スタックのすべてのポートがテストされます。
- **dot1x test eapol-capable** コマンドを 802.1x 対応のポートで設定し、リンクがアップになると、ポートは、802.1x に対応するかどうか、接続クライアントでクエリーを実行します。クライアントが通知パケットに応答すると、802.1x 対応です。クライアントがタイムアウト時間内に応答すると Syslog メッセージが生成されます。クライアントがクエリーに応答しない場合、クライアントは 802.1x 対応ではありません。Syslog メッセージは生成されません。
- 準備状態チェックは、複数のホスト（たとえば、IP Phone に接続される PC）を扱うポートに送信できます。Syslog メッセージは、タイマー時間内に準備状態チェックに応答する各クライアントに生成されます。

手順の概要

1. **enable**
2. **dot1x test eapol-capable [interface interface-id]**
3. **configure terminal**
4. **dot1x test timeout timeout**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	dot1x test eapol-capable [interface <i>interface-id</i>] 例 : <pre>Device# dot1x test eapol-capable interface gigabitethernet1/0/13 DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL capable</pre>	スイッチ上で 802.1x 準備状態チェックをイネーブルにします。 (任意) <i>interface-id</i> では、IEEE 802.1x の準備状態をチェックするポートを指定します。 (注) オプションの interface キーワードを省略した場合、スイッチのすべてのインターフェイスがテストされます。
ステップ 3	configure terminal 例 : <pre>Device# configure terminal</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 4	dot1x test timeout <i>timeout</i> 例 : <pre>Device(config)# dot1x test timeout 54</pre>	(任意) EAPOL 応答の待機に使用するタイムアウトを設定します。範囲は 1 ~ 65535 秒です。デフォルトは 10 秒です。
ステップ 5	end 例 : <pre>Device(config)# end</pre>	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

スイッチ/RADIUS サーバー間通信の設定

RADIUS サーバーのパラメータを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip radius source-interface vlan *vlan interface number***
4. **radius server *server name***
5. **address {ipv4 | ipv6} *ip address***
6. **key *string***
7. **exit**
8. **radius-server dead-criteria tries *num-tries***
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip radius source-interface vlan <i>vlan interface number</i> 例： Device(config)# ip radius source-interface vlan 80	RADIUS パケットが、指定されたインターフェイスの IP アドレスを含むように指定します。
ステップ 4	radius server <i>server name</i> 例： Device(config)# radius server rsim address ipv4 172.16.0.1	（任意）RADIUS サーバーの IP アドレスを指定します。
ステップ 5	address {ipv4 ipv6} <i>ip address</i> 例： Device(config-radius-server)# address ipv4 10.0.1.2 auth-port 1550 acct-port 1560	RADIUS サーバーの IP アドレスを設定します。
ステップ 6	key <i>string</i> 例： Device(config-radius-server)# key rad123	（任意）RADIUS サーバー上で動作する RADIUS デーモンとスイッチの間で使用する認証および暗号キーを指定します。
ステップ 7	exit 例： Device(config-radius-server)# exit	RADIUS サーバーモードを終了して、グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 8	radius-server dead-criteria tries <i>num-tries</i> 例： Device(config)# radius-server dead-criteria tries 30	RADIUS サーバーに送信されたメッセージへの応答がない場合に、このサーバーが非アクティブであると見なすまでの送信回数を指定します。指定できる <i>num-tries</i> の範囲は 1 ~ 100 です。
ステップ 9	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

再認証回数の設定

ポートが無許可ステートに変わる前に、デバイスが認証プロセスを再開する回数を変更することもできます。



(注) このコマンドのデフォルト値は、リンクの信頼性が低下した場合や、特定のクライアントおよび認証サーバの動作に問題がある場合など、異常な状況に対する調整を行う必要がある際に限って変更してください。

再認証回数を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport mode access**
5. **dot1x max-req *count***
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例： Device# interface gigabitethernet2/0/1	設定するポートを指定し、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 4	switchport mode access 例： Device(config-if)# switchport mode access	RADIUS サーバを事前に設定した場合に限り、ポートをアクセス モードに設定します。
ステップ 5	dot1x max-req count 例： Device(config-if)# dot1x max-req 4	ポートが無許可ステートになる前に、デバイスが認証プロセスを再開する回数を設定します。指定できる範囲は 0 ~ 10 です。デフォルトは 2 です。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ゲスト VLAN の設定

サーバが EAP Request/Identity フレームに対する応答を受信しない場合、ゲスト VLAN を設定すると、802.1x 対応でないクライアントはゲスト VLAN に配置されます。802.1x 対応であっても、認証に失敗したクライアントは、ネットワークへのアクセスが許可されません。スイッチは、シングルホストモードまたはマルチホストモードでゲスト VLAN をサポートします。

ゲスト VLAN を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **dot1x port-control auto**

5. **authentication event no-response action authorize vlan *vlan-id***
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 2/0/2	設定するポートを指定し、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 4	dot1x port-control auto 例： Device(config-if)# dot1x port-control auto	ポートでの 802.1x 認証をイネーブルにします。
ステップ 5	authentication event no-response action authorize vlan <i>vlan-id</i> 例： Device(config-if)# authentication event no-response action authorize vlan 2	アクティブ VLAN を 802.1x ゲスト VLAN として指定します。指定できる範囲は 1 ~ 4094 です。 内部 VLAN（ルーテッドポート）、RSPAN VLAN、音声 VLAN を除くあらゆるアクティブ VLAN を 802.1x ゲスト VLAN として設定できます。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

制限付き VLAN の設定

デバイスに制限付き VLAN を設定すると、認証サーバが有効なユーザ名とパスワードを受信しなかった場合、IEEE 802.1x 準拠のクライアントが制限付き VLAN に移動します。デバイスは、シングルホストモードでのみ制限付き VLAN をサポートします。

制限付き VLAN を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan *vlan-id***
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 2/0/2	設定するポートを指定し、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 4	authentication port-control auto 例： Device(config-if)# authentication port-control auto	ポートでの 802.1x 認証をイネーブルにします。
ステップ 5	authentication event fail action authorize vlan <i>vlan-id</i> 例：	アクティブ VLAN を 802.1x 制限付き VLAN として指定します。指定できる範囲は 1 ~ 4094 です。

制限付き VLAN の認証試行回数の設定

	コマンドまたはアクション	目的
	<code>Device(config-if)# authentication event fail action authorize vlan 2</code>	内部 VLAN (ルーテッドポート)、RSPAN VLAN または音声 VLAN を除き、任意のアクティブ VLAN を 802.1x 制限 VLAN として設定できます。
ステップ 6	end 例： <code>Device(config-if)# end</code>	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、特権 EXEC モードに戻ります。

制限付き VLAN の認証試行回数の設定

ユーザーに制限付き VLAN を割り当てる前に、**authentication event fail retry retry count** インターフェイス コンフィギュレーション コマンドを使用して、認証試行回数を最大に設定できます。指定できる試行回数は 1～3 です。デフォルトは 3 回に設定されています。

認証試行回数を最大に設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan vlan-id**
6. **authentication event fail retry retry count**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： <code>Device> enable</code>	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例： <code>Device# configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例：	設定するポートを指定し、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。

	コマンドまたはアクション	目的
	Device(config)# interface gigabitethernet 2/0/3	
ステップ 4	authentication port-control auto 例： Device(config-if) # authentication port-control auto	ポートでの 802.1x 認証をイネーブルにします。
ステップ 5	authentication event fail action authorize vlan vlan-id 例： Device(config-if) # authentication event fail action authorize vlan 8	アクティブ VLAN を 802.1x 制限付き VLAN として指定します。指定できる範囲は 1 ~ 4094 です。 内部 VLAN (ルーテッドポート)、RSPAN VLAN または音声 VLAN を除き、任意のアクティブ VLAN を 802.1x 制限 VLAN として設定できます。
ステップ 6	authentication event fail retry retry count 例： Device(config-if) # authentication event fail retry 2	
ステップ 7	end 例： Device(config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

クリティカル音声 VLAN を使用した 802.1x アクセス不能認証バイパスの設定

ポートにクリティカル音声 VLAN を設定し、アクセス不能認証バイパス機能をイネーブルにするには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **radius-server dead-criteria {time seconds } [tries number]**
5. **radius-server deadtime** 分
6. **radius server server name**
7. **address {ipv4 | ipv6} ip address auth-port port_number acct-port port_number**

8. **key string**
9. **exit**
10. **dot1x critical {eapol | recovery delay milliseconds}**
11. **interface interface-id**
12. **authentication event server dead action {authorize | reinitialize} vlan vlan-id]**
13. **switchport voice vlan vlan-id**
14. **authentication event server dead action authorize voice**
15. **end**
16. **show authentication interface interface-id**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA をイネーブルにします。
ステップ 4	radius-server dead-criteria {time seconds} [tries number] 例： Device(config)# radius-server dead-criteria time 20 tries 10	RADIUS サーバーが使用不可またはダウン（切断）と見なされる条件を設定します。 • time : 1 ~ 120 秒。スイッチは、デフォルトの <i>seconds</i> 値を 10 ~ 60 の間で動的に決定します。 • number : 1 ~ 100 の試行回数。スイッチは、デフォルトの <i>triesnumber</i> を 10 ~ 100 の間で動的に決定します。
ステップ 5	radius-server deadtime 分 例： Device(config)# radius-server deadtime 60	（任意）RADIUS サーバに要求が送信されない分数を設定します。指定できる範囲は 0 ~ 1440 分（24 時間）です。デフォルト値は 0 分です。

	コマンドまたはアクション	目的
ステップ 6	radius server <i>server name</i> 例 : Device (config) # radius server rsim address ipv4 124.2.2.12	(任意) RADIUS サーバーの IP アドレスを指定します。
ステップ 7	address {ipv4 ipv6} <i>ip address auth-port port_number acct-port port_number</i> 例 : Device (config-radius-server) # address ipv4 10.0.1.2 auth-port 1550 acct-port 1560	RADIUS サーバーの IP アドレスを設定します。
ステップ 8	key string 例 : Device (config-radius-server) # key rad123	(任意) RADIUS サーバー上で動作する RADIUS デーモンとスイッチの間で使用する認証および暗号キーを指定します。
ステップ 9	exit 例 : Device (config-radius-server) # exit	RADIUS サーバーモードを終了して、グローバル コンフィギュレーションモードを開始します。
ステップ 10	dot1x critical {eapol recovery delay milliseconds} 例 : Device (config) # dot1x critical eapol Device (config) # dot1x critical recovery delay 2000	(任意) アクセス不能認証バイパスのパラメータを設定します。 <ul style="list-style-type: none"> • eapol : スイッチがクリティカルポートを正常に認証すると、スイッチが EAPOL 成功メッセージを送信するように指定します。 • recovery delay milliseconds : 使用できない RADIUS サーバーが使用できるようになったときに、スイッチがクリティカルポートを再初期化するために待機する回復遅延期間を設定します。指定できる範囲は 1 ~ 10000 ミリ秒です。デフォルトは 1000 ミリ秒です (ポートは毎秒再初期化できます)。
ステップ 11	interface interface-id 例 : Device (config) # interface gigabitethernet 1/0/1	設定するポートを指定し、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。

	コマンドまたはアクション	目的
ステップ 12	authentication event server dead action {authorize reinitialize} vlan <i>vlan-id</i> 例： <pre>Device(config-if)# authentication event server dead action reinitialicze vlan 20</pre>	これらのキーワードを使用して、RADIUS サーバが到達不能な場合にポートでホストを移動します。 <ul style="list-style-type: none"> • authorize : 認証しようとする新しいホストをユーザー指定のクリティカル VLAN に移動します。 • reinitialize : ポートのすべての許可済みホストをユーザー指定のクリティカル VLAN に移動します。
ステップ 13	switchport voice vlan <i>vlan-id</i> 例： <pre>Device(config-if)# switchport voice vlan</pre>	ポートの音声 VLAN を指定します。音声 VLAN はステップ 6 で設定されたクリティカルデータ VLAN と同じにはできません。
ステップ 14	authentication event server dead action authorize voice 例： <pre>Device(config-if)# authentication event server dead action authorize voice</pre>	RADIUS サーバが到達不能な場合、ポートのデータトラフィックを音声 VLAN に移動するために、クリティカル音声 VLAN を設定します。
ステップ 15	end 例： <pre>Device(config-if)# end</pre>	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、特権 EXEC モードに戻ります。
ステップ 16	show authentication interface <i>interface-id</i> 例： <pre>Device(config-if)# show authentication interface gigabitethernet 1/0/1</pre>	(任意) 設定を確認します。

例

RADIUS サーバのデフォルト設定に戻すには、**no radius-server dead-criteria**、**no radius-server deadtime**、および **no radius-server** グローバル コンフィギュレーション コマンドを使用します。アクセス不能な認証バイパスをディセーブルにするには、**no authentication event server dead action** インターフェイス コンフィギュレーション コマンドを使用します。クリティカル音声 VLAN をディセーブルにするには、**authentication**

event server dead action authorize voice インターフェイス コンフィギュレーション コマンドを使用します。

WoL を使用した 802.1x 認証の設定

WoL を使用した 802.1x 認証をイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **authentication control-direction {both | in}**
5. **end**
6. **show authentication sessions interface *interface-id***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet2/0/3	設定するポートを指定し、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 4	authentication control-direction {both in} 例： Device(config-if)# authentication control-direction both	ポートで WoL を使用して 802.1x 認証をイネーブルにし、次のキーワードを使用してポートを双方向または単方向に設定します。 <ul style="list-style-type: none"> • both : ポートを双方向に設定します。ポートは、ホストにパケットを送受信できません。デフォルトでは、ポートは双方向です。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • in : ポートを単方向に設定します。ポートは、ホストにパケットを送信できますが、受信はできません。
ステップ 5	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	show authentication sessions interface interface-id 例 : Device# show authentication sessions interface gigabitethernet2/0/3	インターフェイスの現在の認証マネージャセッションに関する情報を表示します。

MAC 認証バイパスの設定

MAC 認証バイパスをイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **authentication port-control auto**
5. **mab [eap]**
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 2/0/1	設定するポートを指定し、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 4	authentication port-control auto 例： Device(config-if)# authentication port-control auto	ポートでの 802.1x 認証をイネーブルにします。
ステップ 5	mab [eap] 例： Device(config-if)# mab	MAC 認証バイパスをイネーブルにします。 (任意) eap キーワードを使用して、許可用に EAP を使用できるようにデバイスを設定します。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

802.1x ユーザー ディストリビューションの設定

VLAN グループを設定して、VLAN をそのグループにマッピングするには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan group** *vlan-group-name* **vlan-list** *vlan-list*
4. **no vlan group** *vlan-group-name* **vlan-list** *vlan-list*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例：	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
	Device> enable	
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	vlan group <i>vlan-group-name</i> vlan-list <i>vlan-list</i> 例： Device(config)# vlan group eng-dept vlan-list 10	VLAN グループを設定し、単一の VLAN または VLAN の範囲をそのグループにマッピングします。
ステップ 4	no vlan group <i>vlan-group-name</i> vlan-list <i>vlan-list</i> 例： Device(config)# no vlan group eng-dept vlan-list 10	VLAN グループ コンフィギュレーションまたは VLAN グループ コンフィギュレーションの要素をクリアします。
ステップ 5	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

NAC レイヤ 2 802.1x 検証の設定

NAC レイヤ 2 802.1x 検証を設定できます。これは、RADIUS サーバーを使用した 802.1x 認証とも呼ばれます。

NAC レイヤ 2 802.1x 検証を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport mode access**
5. **authentication event no-response action authorize vlan *vlan-id***
6. **authentication periodic**
7. **authentication timer reauthenticate**
8. **end**

9. show authentication sessions interface *interface-id*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例： Device(config)# interface gigabitethernet2/0/3	設定するポートを指定し、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 4	switchport mode access 例： Device(config-if)# switchport mode access	RADIUS サーバを設定した場合に限り、ポートをアクセス モードに設定します。
ステップ 5	authentication event no-response action authorize vlan vlan-id 例： Device(config-if)# authentication event no-response action authorize vlan 8	アクティブ VLAN を 802.1x ゲスト VLAN として指定します。指定できる範囲は 1 ~ 4094 です。 内部 VLAN（ルーテッドポート）、RSPAN VLAN、音声 VLAN を除くあらゆるアクティブ VLAN を 802.1x ゲスト VLAN として設定できます。
ステップ 6	authentication periodic 例： Device(config-if)# authentication periodic	クライアントの定期的な再認証（デフォルトではディセーブル）をイネーブルにします。
ステップ 7	authentication timer reauthenticate 例： Device(config-if)# authentication timer reauthenticate	クライアントに対する再認証試行を設定します（1 時間に設定）。 このコマンドがスイッチの動作に影響するのは、定期的な再認証をイネーブルに設定した場合だけです。

	コマンドまたはアクション	目的
ステップ 8	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 9	show authentication sessions interface interface-id 例： Device# show authentication sessions interface gigabitethernet2/0/3	インターフェイスの現在の認証マネージャセッションに関する情報を表示します。

NEAT を使用したオーセンティケータ スイッチの設定

この機能を設定するには、ワイヤリングクローゼット外の1つのスイッチがサブリカントとして設定され、オーセンティケータ スイッチに接続されている必要があります。



- (注)
- CISP または NEAT セッションがアクティブなときにラインカードを取り外してシャーシに挿入する場合は、オーセンティケータ スイッチ インターフェイスの設定を明示的にフラッシングすることによって、アクセスモードに復元する必要があります。
 - *cisco-av-pairs* は、ISE で *device-traffic-class=switch* として設定されている必要があります。これにより、サブリカントが正常に認証された後でトランクとしてインターフェイスが設定されます。

スイッチをオーセンティケータに設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **cisp enable**
4. **interface interface-id**
5. **switchport mode access**
6. **authentication port-control auto**
7. **dot1x pae authenticator**
8. **spanning-tree portfast**
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	cisp enable 例： Device(config)# cisp enable	CISP をイネーブルにします。
ステップ 4	interface interface-id 例： Device(config)# interface gigabitethernet 2/0/1	設定するポートを指定し、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 5	switchport mode access 例： Device(config-if)# switchport mode access	ポートモードを access に設定します。
ステップ 6	authentication port-control auto 例： Device(config-if)# authentication port-control auto	ポート認証モードを auto に設定します。
ステップ 7	dot1x pae authenticator 例： Device(config-if)# dot1x pae authenticator	インターフェイスをポートアクセス エンティティ (PAE) オーセンティケータとして設定します。

	コマンドまたはアクション	目的
ステップ 8	spanning-tree portfast 例： Device(config-if)# spanning-tree portfast trunk	単一ワークステーションまたはサーバに接続されたアクセスポート上で Port Fast をイネーブルにします。
ステップ 9	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

NEAT を使用したサブリカントスイッチの設定

スイッチをサブリカントに設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **cisp enable**
4. **dot1x credentials profile**
5. **username suppswitch**
6. **password password**
7. **dot1x supplicant force-multicast**
8. **interface interface-id**
9. **switchport trunk encapsulation dot1q**
10. **switchport mode trunk**
11. **dot1x pae supplicant**
12. **dot1x credentials profile-name**
13. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	Device# configure terminal	
ステップ 3	cisp enable 例 : Device (config) # cisp enable	CISP をイネーブルにします。
ステップ 4	dot1x credentials profile 例 : Device (config) # dot1x credentials test	802.1x クレデンシャル プロファイルを作成します。 これは、サブリカントとして設定されるポートに接続する必要があります。
ステップ 5	username suppswitch 例 : Device (config) # username suppswitch	ユーザ名を作成します。
ステップ 6	password password 例 : Device (config) # password myswitch	新しいユーザ名のパスワードを作成します。
ステップ 7	dot1x supplicant force-multicast 例 : Device (config) # dot1x supplicant force-multicast	ユニキャストまたはマルチキャストパケットのいずれかを受信した場合にスイッチに強制的にマルチキャスト EAPOL だけを送信させます。 これにより、NEAT がすべてのホストモードでのサブリカントスイッチで機能できるようになります。
ステップ 8	interface interface-id 例 : Device (config) # interface gigabitethernet1/0/1	設定するポートを指定し、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 9	switchport trunk encapsulation dot1q 例 : Device (config-if) # switchport trunk encapsulation dot1q	ポートをトランク モードに設定します。

	コマンドまたはアクション	目的
ステップ 10	switchport mode trunk 例： Device(config-if)# switchport mode trunk	インターフェイスを VLAN トランク ポートとして設定します。
ステップ 11	dot1x pae supplicant 例： Device(config-if)# dot1x pae supplicant	インターフェイスをポート アクセス エンティティ (PAE) サブリカントとして設定します。
ステップ 12	dot1x credentials profile-name 例： Device(config-if)# dot1x credentials test	802.1x クレデンシャルプロファイルをインターフェイスに対応付けます。
ステップ 13	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ダウンロード可能 ACL およびリダイレクト URL を使用した 802.1x 認証の設定



(注) スイッチにダウンロードする前に、ダウンロード可能な ACL を ACS で設定する必要があります。

ポートでの認証後、**show ip access-list** 特権 EXEC コマンドを使用して、ポートにダウンロードした ACL を表示できます。



(注) **show ip access-lists interface** コマンドの出力には、dACL フィルタ ID や ACL フィルタ ID は表示されません。これは、物理インターフェイスではなく、各認証セッションのマルチドメイン認証によって作成された仮想ポートに ACL が接続されるためです。dACL フィルタ ID や ACL フィルタ ID を表示するには、**show ip access-lists access-list-name** コマンドを使用します。**access-list-name** は、**show access-session interface interface-name detail** コマンドの出力から取得する必要があります。**access-list-name** では大文字と小文字が区別されます。

ダウンロード可能な ACL の設定

これらのポリシーは、クライアントが認証され、クライアント IP アドレスが IP デバイストラッキングテーブルに追加された後で有効になります。その後スイッチがダウンロード可能な ACL をポートに適用します。

特権 EXEC モードで次の手順を実行します。

始める前に

SISF ベースのデバイストラッキングは、802.1x 認証を設定するための前提条件です。デバイストラッキングをプログラムまたは手動で有効にしていることを確認します。詳細については、「SISF ベースのトラッキングの設定」の章を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authorization network default local group radius**
5. **radius-server vsa send authentication**
6. **interface *interface-id***
7. **ip access-group *acl-id* in**
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA 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 4	aaa authorization network default local group radius 例 : <pre>Device(config)# aaa authorization network default local group radius</pre>	許可の方法をローカルに設定します。認可方式を削除するには、 no aaa authorization network default local group radius コマンドを使用します。
ステップ 5	radius-server vsa send authentication 例 : <pre>Device(config)# radius-server vsa send authentication</pre>	RADIUS VSA 送信認証を設定します。
ステップ 6	interface interface-id 例 : <pre>Device(config)# interface gigabitethernet2/0/4</pre>	設定するポートを指定し、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 7	ip access-group acl-id in 例 : <pre>Device(config-if)# ip access-group default_acl in</pre>	ポートの入力方向のデフォルト ACL を設定します。 (注) <i>acl-id</i> はアクセスリストの名前または番号です。
ステップ 8	end 例 : <pre>Device(config-if)# end</pre>	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、特権 EXEC モードに戻ります。

ダウンロードポリシーの設定

始める前に

SISF ベースのデバイストラッキングは、802.1x 認証を設定するための前提条件です。デバイストラッキングをプログラムまたは手動で有効にしていることを確認します。

詳細については、「*SISF* ベースの追跡の設定」を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **access-list access-list-number { deny | permit } { hostname | any | host } log**
4. **interface interface-id**

5. **ip access-group *acl-id* in**
6. **exit**
7. **aaa new-model**
8. **aaa authorization network default group radius**
9. **radius-server vsa send authentication**
10. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	access-list <i>access-list-number</i> { deny permit } { hostname any host } log 例 : Device(config)# access-list 1 deny any log	デフォルト ポート ACL を定義します。 access-list-number には、1 ~ 99 または 1300 ~ 1999 の 10 進数を指定します。 条件が一致した場合にアクセスを拒否する場合は deny を指定し、許可する場合は permit を指定します。 source は、次のようなパケットを送信するネットワークまたはホストの送信元アドレスです。 <ul style="list-style-type: none"> • hostname : ドット付き 10 進表記による 32 ビット長の値。 • any : source および source-wildcard の値 0.0.0.0 255.255.255.255 の省略形を意味するキーワード any。source-wildcard 値を入力する必要はありません。 • host : source および source-wildcard の値 source 0.0.0.0 の省略形を意味するキーワード host。 (任意) source-wildcard ビットを送信元アドレスに適用します。

	コマンドまたはアクション	目的
		(任意) ログを入力して、エントリと一致するパケットに関する情報ロギングメッセージをコンソールに送信します。
ステップ 4	interface interface-id 例 : Device(config)# interface gigabitethernet 2/0/2	インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	ip access-group acl-id in 例 : Device(config-if)# ip access-group default_acl in	ポートの入力方向のデフォルトACLを設定します。 (注) <i>acl-id</i> はアクセスリストの名前または番号です。
ステップ 6	exit 例 : Device(config-if)# exit	インターフェイス コンフィギュレーション モードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 7	aaa new-model 例 : Device(config)# aaa new-model	AAA をイネーブルにします。
ステップ 8	aaa authorization network default group radius 例 : Device(config)# aaa authorization network default group radius	許可の方法をローカルに設定します。認可方式を削除するには、 no aaa authorization network default group radius コマンドを使用します。
ステップ 9	radius-server vsa send authentication 例 : Device(config)# radius-server vsa send authentication	ベンダー固有属性を認識し使用するために、ネットワーク アクセス サーバーを設定します。 (注) ダウンロード可能なACLが機能する必要があります。
ステップ 10	end 例 : Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

VLAN ID ベース MAC 認証の設定

特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **mab request format attribute 32 vlan access-vlan**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mab request format attribute 32 vlan access-vlan 例： Device(config)# mab request format attribute 32 vlan access-vlan	VLAN ID ベース MAC 認証をイネーブルにします。
ステップ 4	end 例： Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

柔軟な認証順序の設定

下の手順で使用される例は、MAB が IEEE 802.1x 認証 (dot1x) の前に試行されるように柔軟な認証の順序設定の順序を変更します。MAB は最初の認証方式として設定されているため、MAB は他のすべての認証方式よりも優先されます。

特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode access**
5. **authentication order** [**dot1x** | **mab**] | {**webauth**}
6. **authentication priority** [**dot1x** | **mab**] | {**webauth**}
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 1/0/1	設定するポートを指定し、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 4	switchport mode access 例： Device(config-if)# switchport mode access	RADIUS サーバを事前に設定した場合に限り、ポートをアクセス モードに設定します。
ステップ 5	authentication order [dot1x mab] { webauth } 例： Device(config-if)# authentication order mab dot1x	（任意）ポート上で使用される認証方式の順序を設定します。
ステップ 6	authentication priority [dot1x mab] { webauth } 例： Device(config-if)# authentication priority mab	（任意）認証方式をポートプライオリティリストに追加します。

	コマンドまたはアクション	目的
	<code>dot1x</code>	
ステップ 7	end 例 : Device (config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Open1x の設定

ポートの許可ステータスの手動制御をイネーブルにするには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport mode access**
5. **authentication control-direction {both | in}**
6. **authentication fallback *name***
7. **authentication host-mode[*multi-auth* | *multi-domain* | *multi-host* | *single-host*]**
8. **authentication open**
9. **authentication order [*dot1x* | *mab*] | {*webauth*}**
10. **authentication periodic**
11. **authentication port-control {*auto* | *force-authorized* | *force-un authorized*}**
12. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 1/0/1	設定するポートを指定し、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 4	switchport mode access 例： Device(config-if)# switchport mode access	RADIUS サーバを設定した場合に限り、ポートをアクセスモードに設定します。
ステップ 5	authentication control-direction {both in} 例： Device(config-if)# authentication control-direction both	(任意) ポート制御を単一方方向モードまたは双方向モードに設定します。
ステップ 6	authentication fallback name 例： Device(config-if)# authentication fallback profile1	(任意) 802.1x 認証をサポートしないクライアント用のフォールバック方法として Web 認証を使用するようポートを設定します。
ステップ 7	authentication host-mode[multi-auth multi-domain multi-host single-host] 例： Device(config-if)# authentication host-mode multi-auth	(任意) ポート上で認証マネージャモードを設定します。
ステップ 8	authentication open 例： Device(config-if)# authentication open	(任意) ポート上でオープンアクセスをイネーブルまたはディセーブルにします。
ステップ 9	authentication order [dot1x mab] {webauth} 例： Device(config-if)# authentication order dot1x webauth	(任意) ポート上で使用される認証方式の順序を設定します。

	コマンドまたはアクション	目的
ステップ 10	authentication periodic 例： Device (config-if) # authentication periodic	(任意) ポート上で再認証をイネーブまたはディセーブルにします。
ステップ 11	authentication port-control {auto force-authorized force-un authorized} 例： Device (config-if) # authentication port-control auto	(任意) ポートの許可ステータスの手動制御をイネーブにします。
ステップ 12	end 例： Device (config-if) # end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ポート上での 802.1x 認証のディセーブル化

802.1x 認証をポートでディセーブルにするには、**no dot1x pae** インターフェイス コンフィギュレーション コマンドを使用します。

ポートで 802.1x 認証をディセーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **switchport mode access**
5. **no dot1x pae authenticator**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。

802.1x 認証設定のデフォルト値へのリセット

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例： Device(config)# interface gigabitethernet 2/0/1	設定するポートを指定し、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 4	switchport mode access 例： Device(config-if)# switchport mode access	(任意) RADIUS サーバを設定した場合に限り、ポートをアクセス モードに設定します。
ステップ 5	no dot1x pae authenticator 例： Device(config-if)# no dot1x pae authenticator	ポートでの 802.1x 認証をディセーブルにします。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

802.1x 認証設定のデフォルト値へのリセット

802.1x 認証設定をデフォルト値に戻すには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **dot1x default**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例： Device(config)# interface gigabitethernet 1/0/2	インターフェイス コンフィギュレーション モードを開始し、設定するポートを指定します。
ステップ 4	dot1x default 例： Device(config-if)# dot1x default	設定可能な 802.1x のパラメータをデフォルト値へ戻します。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

音声認識 802.1x セキュリティの設定

音声認識 802.1x セキュリティ機能をデバイスで使用して、セキュリティ違反が発生した場合にデータまたは音声 VLAN に関係なく VLAN だけをディセーブルにします。この機能は、PC が IP Phone に接続されている IP Phone 環境で使用できます。データ VLAN でセキュリティ違反が検出されると、データ VLAN だけがシャットダウンされます。音声 VLAN のトラフィックは中断することなくデバイスで送受信されます。

デバイスで音声認識 802.1x 音声セキュリティを設定する場合、次の注意事項に従ってください。

- 音声認識 802.1x セキュリティをイネーブルにするには、**errdisable detect cause security-violation shutdown vlan** グローバル コンフィギュレーション コマンドを入力します。音声認識 802.1x セキュリティをディセーブルにするには、このコマンドの **no** バー

ジョンを入力します。このコマンドは、デバイスの 802.1x 設定ポートのすべてに適用されます。



(注) **shutdown vlan** キーワードを指定しない場合、**error-disabled** ステータスになった際にポート全体がシャットダウンされます。

- **errdisable recovery cause security-violation** グローバル コンフィギュレーション コマンドを使用して、**error-disabled** リカバリを設定すると、ポートは自動的に再びイネーブルにされます。**error-disabled** リカバリがポートで設定されていない場合、**shutdown** および **no shutdown** インターフェイス コンフィギュレーション コマンドを使用してポートを再びイネーブルにします。
- 個々の VLAN を再びイネーブルにするには、**clear errdisable interface interface-id vlan [vlan-list]** 特権 EXEC コマンドを使用します。範囲を指定しない場合、ポートのすべての VLAN がイネーブルにされます。

音声認識 802.1x セキュリティを有効にするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **errdisable detect cause security-violation shutdown vlan**
4. **errdisable recovery cause security-violation**
5. 次を入力します。
 - **shutdown**
 - **no shutdown**
6. **exit**
7. **clear errdisable interface interface-id vlan [vlan-list]**
8. **show errdisable detect**

手順の詳細

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

	コマンドまたはアクション	目的
	Device# <code>configure terminal</code>	
ステップ 3	errdisable detect cause security-violation shutdown vlan 例： Device(config)# <code>errdisable detect cause security-violation shutdown vlan</code>	セキュリティ違反エラーが発生したすべての VLAN をシャットダウンします。 (注) shutdown vlan キーワードを指定しない場合、すべてのポートが <code>errdisable</code> ステータスになり、シャットダウンされます。
ステップ 4	errdisable recovery cause security-violation 例： Device(config)# <code>errdisable recovery cause security-violation</code>	802.1X セキュリティ違反により無効になったポートの自動回復を有効にします。
ステップ 5	次を入力します。 • shutdown • no shutdown 例： Device(config)# <code>no shutdown</code>	(任意) <code>errdisable</code> の VLAN を再びイネーブルにして、すべての <code>errdisable</code> 指示をクリアします。
ステップ 6	exit 例： Device(config)# <code>exit</code>	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 7	clear errdisable interface interface-id vlan [vlan-list] 例： Device# <code>clear errdisable interface gigabitethernet 0/1/1 vlan vlan_list</code>	(任意) <code>errdisable</code> になっている個々の VLAN を再びイネーブルにします。 • <i>interface-id</i> 引数の場合、個々の VLAN を再び有効にするポートを指定します。 • (任意) <i>vlan-list</i> 引数の場合、再び有効にする VLAN のリストを指定します。 <i>vlan-list</i> を指定しない場合は、すべての VLAN が再びイネーブルになります。
ステップ 8	show errdisable detect 例：	error-disabled 検出ステータスを表示します。

コマンドまたはアクション	目的
Device# <code>show errdisable detect</code>	

IEEE 802.1x ポートベースの認証の設定例

例：アクセス不能認証バイパスの設定

次に、アクセス不能認証バイパス機能を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# radius-server dead-criteria time 30 tries 20
Device(config)# radius-server deadtime 60
Device(config)# radius server server1
Device(config-radius-server)# address ipv4 172.29.36.49 acct-port 1618 auth-port 1612
Device(config-radius-server)# key abc1234
Device(config-radius-server)# exit
Device(config)# dot1x critical eapol
Device(config)# dot1x critical recovery delay 2000
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# dot1x critical
Device(config-if)# dot1x critical recovery action reinitialize
Device(config-if)# dot1x critical vlan 20
Device(config-if)# end
```

例：VLAN グループの設定

次に、VLAN グループを設定し、そのグループに VLAN をマッピングし、VLAN グループ コンフィギュレーションおよび指定 VLAN とのマッピングを確認する例を示します。

```
Device> enable
Device(config)# vlan group eng-dept vlan-list 10
Device(config)# exit
Device# show vlan group group-name eng-dept

Group Name                Vlans Mapped
-----                -
eng-dept                   10

Device# show dot1x vlan-group all

Group Name                Vlans Mapped
-----                -
eng-dept                   10
hr-dept                     20
```

次に、VLAN を既存の VLAN グループに追加し、VLAN が追加されたことを確認する例を示します。

```
Device> enable
Device(config)# vlan group eng-dept vlan-list 30
Device(config)# exit
Device(config)# show vlan group eng-dept
```

```
Group Name                Vlans Mapped
-----
eng-dept                  10, 30
```

次に、VLAN を VLAN グループから削除する例を示します。

```
Device> enable
Device# no vlan group eng-dept vlan-list 10
```

次に、すべての VLAN が VLAN グループからクリアされたときに、その VLAN グループもクリアされる例を示します。

```
Device> enable
Device(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.
Device(config)# exit
Device# show vlan group group-name eng-dept
```

次の例では、すべての VLAN グループをクリアする方法を示します。

```
Device> enable
Device(config)# no vlan group end-dept vlan-list all
Device(config)# exit
Device# show vlan-group all
```

IEEE 802.1x ポートベースの認証統計情報とステータスのモニタリング

表 31: 特権 EXEC 表示コマンド

コマンド	目的
<code>show dot1x all statistics</code>	すべてのポートの 802.1x 統計情報を表示します。
<code>show dot1x interface interface-id statistics</code>	指定されたポートの 802.1x 統計情報を表示します。
<code>show dot1x all[count details statistics summary]</code>	スイッチの 802.1x 管理ステータスおよび動作ステータスを表示します。
<code>show dot1x interface interface-id</code>	指定されたポートの 802.1x 管理ステータスおよび動作ステータスを表示します。

表 32: グローバルコンフィギュレーションコマンド

コマンド	目的
<code>no dot1x logging verbose</code>	詳細な 802.1x 認証メッセージをフィルタリングします。

IEEE 802.1x ポートベースの認証の機能履歴

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	IEEE 802.1x ポートベースの認証	IEEE 802.1x 認証は、不正なデバイス（クライアント）によるネットワークアクセスを防止します。

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



第 24 章

Web ベース認証

この章では、デバイスで Web ベース認証を設定する方法について説明します。この章の内容は、次のとおりです。

- [Web ベース認証の制約事項 \(599 ページ\)](#)
- [Web ベース認証について \(599 ページ\)](#)
- [Web ベース認証の設定方法 \(610 ページ\)](#)
- [Web ベース認証の確認 \(624 ページ\)](#)
- [Web ベース認証の機能履歴 \(624 ページ\)](#)

Web ベース認証の制約事項

ホストスイッチ仮想インターフェイス (SVI) のないデバイスは、Cisco Identity Services Engine (ISE) ポスチャリダイレクションの TCP SYN パケットを傍受しません。

Web ベース認証について

Web ベース認証の概要

IEEE 802.1x サプリカントが実行されていないホストシステムでエンドユーザーを認証するには、Web 認証プロキシとして知られている Web ベース認証機能を使用します。

HTTP セッションを開始すると、Web ベース認証は、ホストからの受信 HTTP パケットを横取りし、ユーザーに HTML ログイン ページを送信します。ユーザーはクレデンシャルを入力します。このクレデンシャルは、Web ベース認証機能により、認証のために認証、許可、アカウントティング (AAA) サーバーに送信されます。

認証が成功すると、Web ベース認証はログイン成功 HTML ページをホストに送信し、AAA サーバーから返されたアクセス ポリシーを適用します。

認証に失敗した場合、Web ベース認証は、ログインの失敗を示す HTML ページをユーザーに転送し、ログインを再試行するように、ユーザーにプロンプトを表示します。最大試行回数を

超過した場合、Web ベース認証は、ログインの期限切れを示す HTML ページをホストに転送し、このユーザーは待機期間中、ウォッチ リストに載せられます。



(注) 中央 Web 認証リダイレクト用の HTTPS トラフィック インターセプションはサポートされていません。



(注) グローバルパラメータ マップ (method-type、custom、redirect) は、すべてのクライアントおよび SSID で同じ Web 認証方式 (consent、web consent、webauth など) を使用するときのみ使用する必要があります。これにより、すべてのクライアントが同じ Web 認証方式になります。

要件により、1つの SSID に consent、別の SSID に webauth を使用する場合、名前付きパラメータ マップを 2 つ使用する必要があります。1 番目のパラメータ マップには consent を設定し、2 番目のパラメータ マップには webauth を設定する必要があります。



(注) Webauth クライアントの認証試行時に受信する traceback には、パフォーマンスや行動への影響はありません。これは、ACL アプリケーションの EPM に FFM が返信したコンテキストがすでにキュー解除済み (タイマーの有効期限切れの可能性あり) で、セッションが「未承認」になった場合にまれに発生します。

Web ページがホストされている場所に基づいて、ローカル Web 認証は次のように分類できます。

- 内部：ローカル Web 認証時に、コントローラの内部デフォルト HTML ページ (ログイン、成功、失敗、および期限切れ) が使用されます。
- カスタマイズ：ローカル Web 認証時に、カスタマイズされた Web ページ (ログイン、成功、失敗、および期限切れ) がコントローラにダウンロードされ、使用されます。
- 外部：組み込みまたはカスタム Web ページを使用する代わりに、外部 Web サーバー上でカスタマイズされた Web ページがホストされます。

さまざまな Web 認証ページに基づき、Web 認証のタイプは次のように分類できます。

- *Webauth*：これが基本的な Web 認証です。この場合、コントローラはユーザー名とパスワードの入力が必要なポリシーページを提示します。ネットワークにアクセスするには、ユーザーは正しいクレデンシャルを入力する必要があります。
- *Consent* または *web-passthrough*：この場合、コントローラは [Accept] ボタンまたは [Deny] ボタンが表示されたポリシー ページを提示します。ネットワークにアクセスするには、ユーザーは [Accept] ボタンをクリックする必要があります。

- **Webconsent** : これは webauth と consent の Web 認証タイプの組み合わせです。この場合、コントローラは [Accept] ボタンまたは [Deny] ボタンがあり、ユーザー名とパスワードの入力が必要なポリシーページを提示します。ネットワークにアクセスするには、ユーザーは正しいクレデンシャルを入力して [Accept] ボタンをクリックする必要があります。

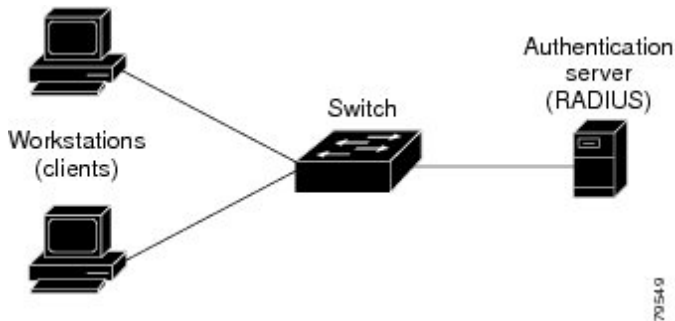
デバイスのロール

Web ベース認証では、ネットワーク上のデバイスに次のような固有の役割があります。

- **クライアント** : LAN およびサービスへのアクセスを要求し、スイッチからの要求に応答するデバイス (ワークステーション)。このワークステーションでは、**JavaScript** が有効な HTML ブラウザが実行されている必要があります。
- **認証サーバー** : クライアントを認証します。認証サーバーはクライアントの識別情報を確認し、そのクライアントが LAN およびスイッチのサービスへのアクセスを許可されたか、あるいはクライアントが拒否されたのかをスイッチに通知します。
- **スイッチ** : クライアントの認証ステータスに基づいて、ネットワークへの物理アクセスを制御します。スイッチはクライアントと認証サーバーとの仲介デバイス (プロキシ) として動作し、クライアントに識別情報を要求し、その情報を認証サーバーで確認し、クライアントに応答をリレーします。

図 28: Web ベース認証デバイスの役割

次の図は、ネットワーク上でのこれらのデバイスの役割を示します。



ホストの検出

スイッチは、検出されたホストに関する情報を格納するために、IP デバイス追跡 テーブルを維持します。

レイヤ 2 インターフェイスでは、Web ベース認証は、これらのメカニズムを使用して、IP ホストを検出します。

- **ARP ベースのトリガー** : ARP リダイレクト ACL により、Web ベース認証は、スタティック IP アドレス、またはダイナミック IP アドレスを持つホストを検出できます。
- **ダイナミック ARP 検査**

- DHCP スヌーピング：スイッチがホストの DHCP バインディング エントリを作成するときに Web ベース認証が通知されます。

セッションの作成

Web ベース認証により、新しいホストが検出されると、次のようにセッションが作成されます。

- 例外リストをレビューします。
ホスト IP が例外リストに含まれている場合、この例外リスト エントリからポリシーが適用され、セッションが確立されます。
- 認証バイパスをレビューします。
ホスト IP が例外リストに含まれていない場合、Web ベース認証は応答しないホスト (NRH) 要求をサーバーに送信します。
サーバーの応答が **access accepted** であった場合、認証はこのホストにバイパスされます。セッションが確立されます。
- HTTP インターセプト ACL を設定します。
NRH 要求に対するサーバーの応答が **access rejected** であった場合、HTTP インターセプト ACL がアクティブ化され、セッションはホストからの HTTP トラフィックを待機します。

認証プロセス

Web ベース認証を有効にすると、次のイベントが発生します。

- ユーザーが HTTP セッションを開始します。
- HTTP トラフィックが横取りされ、認証が開始されます。スイッチは、ユーザーにログインページを送信します。ユーザーはユーザー名とパスワードを入力します。スイッチはこのエントリを認証サーバーに送信します。
- 認証に成功した場合、スイッチは認証サーバーからこのユーザーのアクセスポリシーをダウンロードし、アクティブ化します。ログインの成功ページがユーザーに送信されます。
- 認証に失敗した場合は、スイッチはログインの失敗ページを送信します。ユーザーはログインを再試行します。失敗の回数が試行回数の最大値に達した場合、スイッチはログイン期限切れページを送信します。このホストはウォッチリストに入れられます。ウォッチリストのタイムアウト後、ユーザーは認証プロセスを再試行することができます。
- 認証サーバーがスイッチに応答せず、AAA 失敗ポリシーが設定されている場合、スイッチはホストに失敗アクセスポリシーを適用します。ログインの成功ページがユーザーに送信されます。
- ホストがレイヤ 2 インターフェイス上の ARP プロブに応答しなかった場合、またはホストがレイヤ 3 インターフェイスでアイドルタイムアウト内にトラフィックを送信しなかった場合、スイッチはクライアントを再認証します。

- ホストがレイヤ 2 インターフェイス上の ARP プロブに応答しない場合、スイッチはクライアントを再認証します。
- この機能は、ダウンロードされたタイムアウト、またはローカルに設定されたセッションタイムアウトを適用します。
- Termination-Action が RADIUS である場合、この機能は、サーバーに NRH 要求を送信します。Termination-Action は、サーバーからの応答に含まれます。
- Termination-Action がデフォルトである場合、セッションは廃棄され、適用されたポリシーは削除されます。

ローカル Web 認証バナー

Web 認証を使用して、デフォルトのカスタマイズ済み Web ブラウザ バナーを作成して、スイッチにログインしたときに表示するようにできます。

このバナーは、ログインページと認証結果ポップアップページの両方に表示されます。デフォルトのバナーメッセージは次のとおりです。

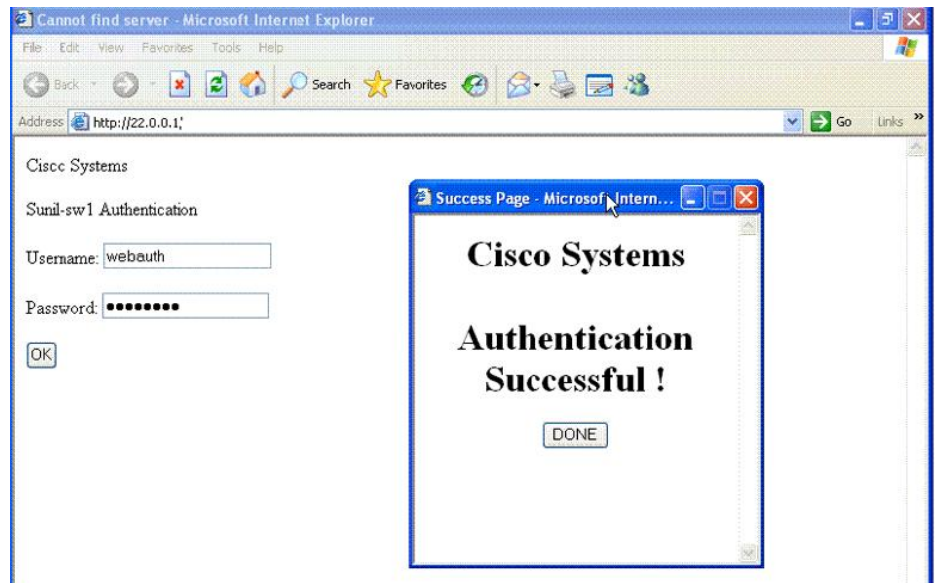
- 認証成功
- 認証失敗
- 認証期限切れ

ローカル Web 認証バナーは、次のように設定できます。

- レガシーモード : **ip admission auth-proxy-banner http** グローバル コンフィギュレーション コマンドを使用します。
- 新スタイルモード : **parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。

ログインページには、デフォルトのバナー、*Cisco Systems*、および *Switch host-name Authentication* が表示されます。*Cisco Systems* は認証結果ポップアップページに表示されます。

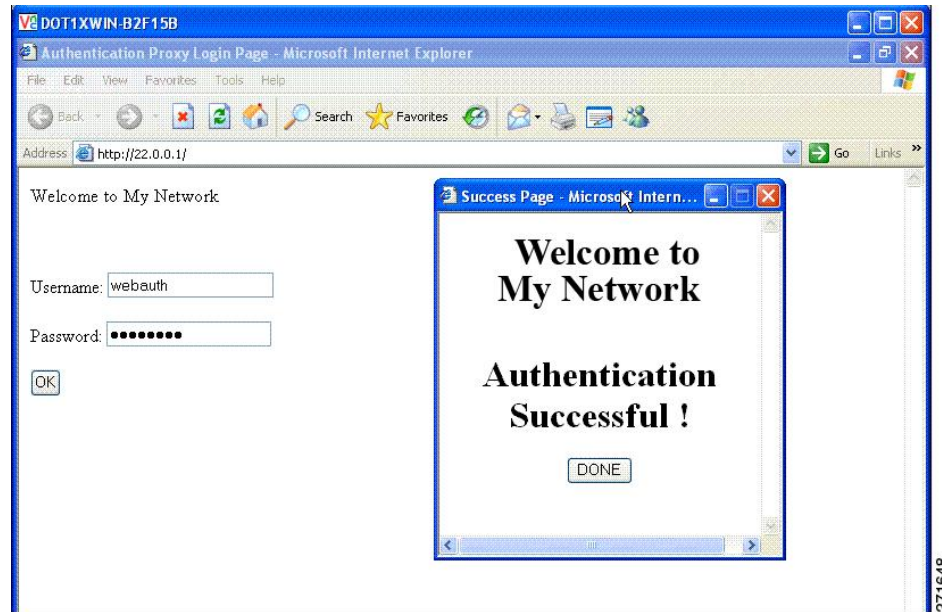
図 29: 認証成功バナー



バナーは次のようにカスタマイズ可能です。

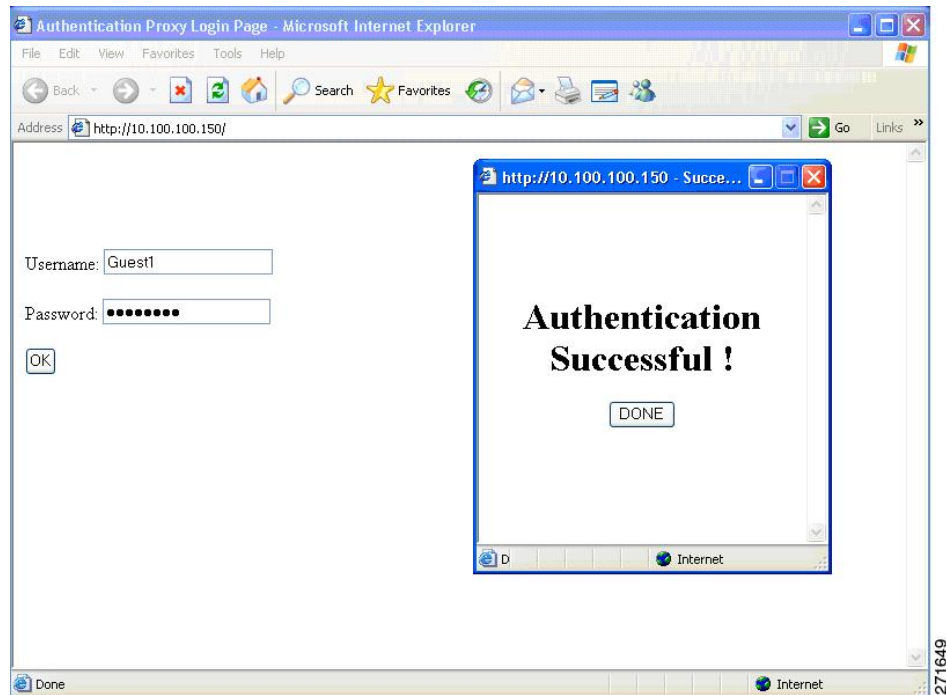
- スイッチ名、ルータ名、または会社名などのメッセージをバナーに追加する。
 - レガシーモード : **ip admission auth-proxy-banner http banner-text** グローバル コンフィギュレーション コマンドを使用します。
 - 新スタイル モード : **parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。
- ログまたはテキスト ファイルをバナーに追加する。
 - レガシーモード : **ip admission auth-proxy-banner http file-path** グローバル コンフィギュレーション コマンドを使用します。
 - 新スタイル モード : **parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。

図 30: カスタマイズされた Web バナー



バナーが有効にされていない場合、Web 認証ログイン画面にはユーザー名とパスワードのダイアログボックスだけが表示され、スイッチにログインしたときにはバナーは表示されません。

図 31: バナーが表示されていないログイン画面



Web 認証カスタマイズ可能な Web ページ

Web ベース認証プロセスでは、スイッチ内部の HTTP サーバーは、認証中のクライアントに配信される 4 種類の HTML ページをホストします。サーバーはこれらのページを使用して、ユーザーに次の 4 種類の認証プロセス ステートを通知します。

- ログイン：資格情報が要求されています。
- 成功：ログインに成功しました。
- 失敗：ログインに失敗しました。
- 期限切れ：ログインの失敗回数が多すぎて、ログインセッションが期限切れになりました。

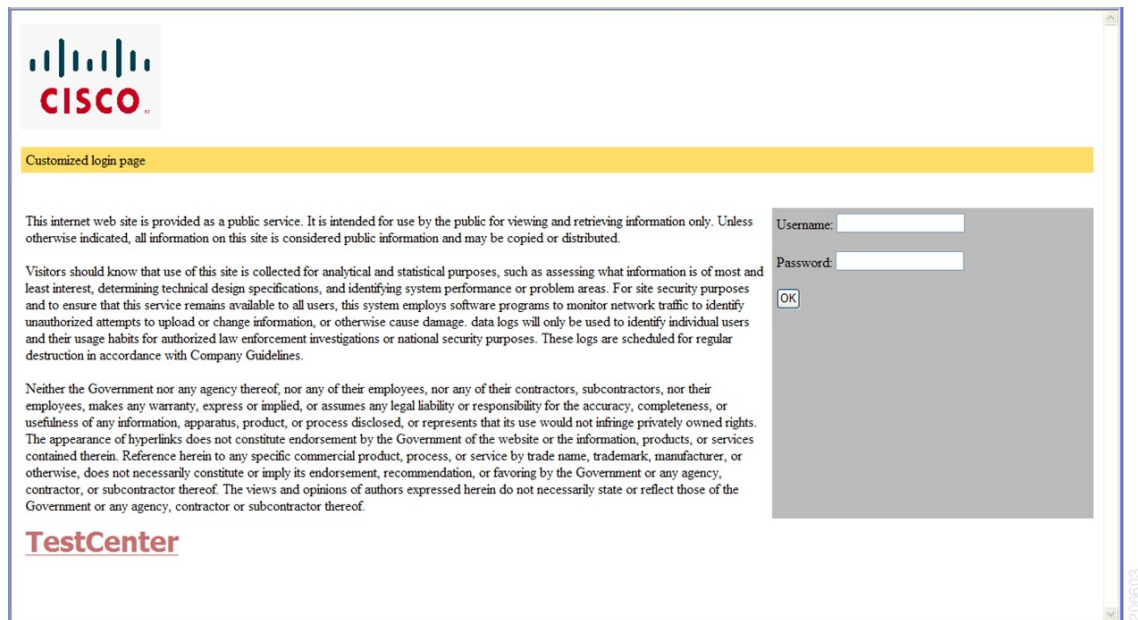
ガイドライン

- デフォルトの内部 HTML ページの代わりに、独自の HTML ページを使用することができます。
- ロゴを使用することもできますし、ログイン、成功、失敗、および期限切れ Web ページでテキストを指定することもできます。
- バナー ページで、ログイン ページのテキストを指定できます。
- これらのページは、HTML で記述されています。
- 成功ページには、特定の URL にアクセスするための HTML リダイレクト コマンドを記入する必要があります。
- この URL 文字列は有効な URL（例：<http://www.cisco.com>）でなければなりません。不完全な URL は、Web ブラウザで、「ページが見つかりません」またはこれに類似するエラーの原因となる可能性があります。
- HTTP 認証で使用される Web ページを設定する場合、これらのページには適切な HTML コマンド（例：ページのタイムアウトを設定、暗号化されたパスワードの設定、同じページが 2 回送信されていないことの確認など）を記入する必要があります。
- 設定されたログイン フォームが有効な場合、特定の URL にユーザーをリダイレクトする CLI コマンドは使用できません。管理者は、Web ページにリダイレクトが設定されていることを保証する必要があります。
- 認証後、特定の URL にユーザーをリダイレクトする CLI コマンドを入力してから、Web ページを設定するコマンドを入力した場合、特定の URL にユーザーをリダイレクトする CLI コマンドは効力を持ちません。
- 設定された Web ページは、スイッチのブート フラッシュ、またはフラッシュにコピーできます。
- スタック可能なスイッチでは、アクティブスイッチまたはメンバスイッチのフラッシュから設定済みのページにアクセスできます。

- ログインページを任意のフラッシュ上に、成功ページと失敗ページを別のフラッシュ（たとえば、アクティブスイッチ、またはメンバスイッチのフラッシュ）に配置できます。
- 4 ページすべてを設定する必要があります。
- Web ページを使ってバナー ページを設定した場合、このバナー ページには効果はありません。
- システムディレクトリ（たとえば、flash、disk0、disk）に保存されていて、ログインページに表示する必要があるロゴファイル（イメージ、フラッシュ、オーディオ、ビデオなど）すべてには、必ず、`web_auth_<filename>` の形式で名前をつけてください。
- 設定された認証プロキシ機能は、HTTP と SSL の両方をサポートしています。

デフォルトの内部 HTML ページの代わりに、自分の HTML ページを使用することができます。認証後のユーザーのリダイレクト先で、内部成功ページの代わりとなる URL を指定することもできます。

図 32: カスタマイズ可能な認証ページ



認証プロキシ Web ページの注意事項

カスタマイズされた認証プロキシ Web ページを設定する際には、次の注意事項に従ってください。

- カスタム Web ページ機能を有効にするには、カスタム HTML ファイルを 4 個すべて指定します。指定したファイルの数が 4 個未満の場合、内部デフォルト HTML ページが使用されます。
- これら 4 個のカスタム HTML ファイルは、スイッチのフラッシュ メモリ内に存在しなければなりません。各 HTML ファイルの最大サイズは 8 KB です。

- カスタム ページ上のイメージはすべて、アクセス可能はHTTPサーバー上に存在しなければなりません。インターセプト ACL は、管理ルール内で設定します。
- カスタム ページからの外部リンクはすべて、管理ルール内でのインターセプト ACL の設定を必要とします。
- 有効な DNS サーバーにアクセスするには、外部リンクまたはイメージに必要な名前解決で、管理ルール内にインターセプト ACL を設定する必要があります。
- カスタム Web ページ機能が有効に設定されている場合、設定された auth-proxy-banner は使用されません。
- カスタム Web ページ機能が有効に設定されている場合、ログインの成功に対するリダイレクション URL は使用できません。
- カスタム ファイルの指定を解除するには、このコマンドの **no** 形式を使用します。

カスタム ログイン ページはパブリック Web フォームであるため、このページについては、次の注意事項に従ってください。

- ログインフォームは、ユーザーによるユーザー名とパスワードの入力を受け付け、これらを **uname** および **pwd** として示す必要があります。
- カスタム ログイン ページは、ページタイムアウト、暗号化されたパスワード、冗長送信の防止など、Web フォームに対するベストプラクティスに従う必要があります。

成功ログインに対するリダイレクト URL の注意事項

成功ログインに対するリダイレクション URL を設定する場合、次の注意事項に従ってください。

- カスタム認証プロキシ Web ページ機能がイネーブルに設定されている場合、リダイレクション URL 機能はディセーブルにされ、CLI では使用できません。リダイレクションは、カスタム ログイン成功ページで実行できます。
- リダイレクション URL 機能が有効に設定されている場合、設定された auth-proxy-banner は使用されません。
- リダイレクション URL の指定を解除するには、このコマンドの **no** 形式を使用します。
- Web ベースの認証クライアントが正常に認証された後にリダイレクション URL が必要な場合、URL 文字列は有効な URL (たとえば `http://`) で開始し、その後に URL 情報が続く必要があります。 `http://` を含まない URL が指定されると、正常に認証が行われても、そのリダイレクション URL によって Web ブラウザでページが見つからないまたは同様のエラーが生じる場合があります。

その他の機能と Web ベース認証の相互作用

ポートセキュリティ

Web ベース認証とポートセキュリティは、同じポートに設定できます。Web ベース認証はポートを認証し、ポートセキュリティは、クライアントの MAC アドレスを含むすべての MAC アドレスに対するネットワーク アクセスを管理します。この場合、このポートを介してネットワークへアクセスできるクライアントの数とグループを制限できます。

LAN ポート IP

LAN ポート IP (LPIP) とレイヤ 2 Web ベース認証は、同じポートに設定できます。ホストは、まず Web ベース認証、次に LPIP ポスチャ検証を使用して認証されます。LPIP ホストポリシーは、Web ベース認証のホストポリシーに優先されます。

Web ベース認証のアイドル時間が満了すると、NAC ポリシーは削除されます。ホストが認証され、ポスチャが再度検証されます。

ゲートウェイ IP

VLAN のいずれかのスイッチポートで Web ベース認証が設定されている場合、レイヤ 3 VLAN インターフェイス上にゲートウェイ IP (GWIP) を設定することはできません。

Web ベース認証はゲートウェイ IP と同じレイヤ 3 インターフェイスに設定できます。ソフトウェアで、両方の機能のホストポリシーが適用されます。GWIP ホストポリシーは、Web ベース認証のホストポリシーに優先されます。

ACL

インターフェイスで VLAN ACL、または Cisco IOS ACL を設定した場合、ACL は、Web ベース認証のホストポリシーが適用された後だけ、ホストトラフィックに適用されます。

レイヤ 2 Web ベース認証では、ポートに接続されたホストからの受信トラフィックについて、ポート ACL (PACL) をデフォルトのアクセスポリシーとして設定することが必須ではないものの、より安全です。認証後、Web ベース認証のホストポリシーは、PACL に優先されます。ポートに設定された ACL がなくても、ポリシー ACL はセッションに適用されます。

MAC ACL と Web ベース認証を同じインターフェイスに設定することはできません。

アクセス VLAN が VACL キャプチャ用に設定されているポートには Web ベース認証は設定できません。

コンテキストベース アクセス コントロール

コンテキストベース アクセス コントロール (CBAC) が、ポート VLAN のレイヤ 3 VLAN インターフェイスで設定されている場合、レイヤ 2 ポートで Web ベース認証は設定できません。

EtherChannel

Web ベース認証は、レイヤ 2 EtherChannel インターフェイス上に設定できます。Web ベース認証設定は、すべてのメンバチャンネルに適用されます。

Web ベース認証の設定方法

デフォルトの Web ベース認証の設定

次の表に、デフォルトの Web ベース認証の設定を示しています。

表 33: デフォルトの Web ベース認証の設定

機能	デフォルト設定
AAA	無効
RADIUS サーバ <ul style="list-style-type: none"> • IP アドレス • UDP 認証ポート • キー 	<ul style="list-style-type: none"> • 指定なし • 指定なし
無活動タイムアウトのデフォルト値	3600 秒
無活動タイムアウト	有効

Web ベース認証の設定に関する注意事項と制約事項

- Web ベース認証は受信時だけの機能です。
- Web ベース認証は、アクセス ポートだけで設定できます。Web ベース認証は、トランクポート、EtherChannel メンバポート、またはダイナミック トランクポートではサポートされていません。
- スイッチが特定のホストまたは Web サーバーにクライアントをリダイレクトしてログインメッセージを表示する場合、外部 Web 認証はサポートされません。
- スタティックな ARP キャッシュが割り当てられているレイヤ 2 インターフェイス上のホストは認証できません。これらのホストは ARP メッセージを送信しないため、Web ベース認証機能では検出されません。
- デフォルトでは、スイッチの IP デバイス追跡機能は無効にされています。Web ベース認証を使用するには、IP デバイス追跡機能を有効にする必要があります。

- Web ベース認証を使用するには、SISF ベースのデバイス追跡を有効にする必要があります。デフォルトでは、SISF ベースのデバイス追跡はスイッチで無効になっています。
- スイッチ HTTP サーバーを実行するには、IP アドレスを少なくとも 1 つ設定する必要があります。また、各ホスト IP アドレスに到達するようにルートを設定する必要があります。HTTP サーバーは、ホストに HTTP ログイン ページを送信します。
- 2 ホップ以上離れたところにあるホストでは、STP トポロジの変更により、ホスト トラフィックの到着するポートが変わってしまった場合、トラフィックが停止する可能性があります。これは、レイヤ 2 (STP) トポロジの変更後に、ARP および DHCP の更新が送信されていない場合に発生します。
- Web ベース認証は、ダウンロード可能なホスト ポリシーとして、VLAN 割り当てをサポートしていません。
- Web ベース認証はセッション認識型ポリシー モードで IPv6 をサポートします。IPv6 Web 認証には、スイッチで設定された少なくとも 1 つの IPv6 アドレスおよびスイッチ ポートに設定された IPv6 スヌーピングが必要です。
- Web ベース認証および Network Edge Access Topology (NEAT) は、相互に排他的です。インターフェイス上で NEAT が有効の場合、Web ベース認証を使用できず、インターフェイス上で Web ベース認証が実行されている場合は、NEAT を使用できません。
- スイッチから RADIUS サーバーへの通信の設定に使用される次の RADIUS セキュリティ サーバー設定を確認します。
 - ホスト名
 - ホスト IP アドレス
 - ホスト名と特定の UDP ポート番号
 - IP アドレスと特定の UDP ポート番号

IP アドレスと UDP ポート番号の組み合わせによって、一意の ID が作成され、サーバーの同一 IP アドレス上にある複数の UDP ポートに RADIUS 要求を送信できるようになります。同じ RADIUS サーバー上の異なる 2 つのホスト エントリに同じサービス (たとえば認証) を設定した場合、2 番めに設定されたホスト エントリは、最初に設定されたホスト エントリのフェールオーバー バックアップとして動作します。RADIUS ホスト エントリは、設定した順序に従って選択されます。

- RADIUS サーバー パラメータを設定する場合は、次の点に注意してください。
 - 別のコマンドラインに、**key string** を指定します。
 - **key string** には、スイッチと、RADIUS サーバー上で動作する RADIUS デーモンとの間で使用する、認証および暗号キーを指定します。キーは、RADIUS サーバーで使用する暗号化キーに一致するテキスト スtring でなければなりません。
 - **key string** を指定する場合、キーの中間、および末尾にスペースを使用します。キーにスペースを使用する場合は、引用符がキーの一部である場合を除き、引用符で

キーを囲まないでください。キーはRADIUSデーモンで使用する暗号に一致している必要があります。

- すべてのRADIUSサーバーについて、タイムアウト、再送信回数、および暗号キー値をグローバルに設定するには、**radius-server host** グローバルコンフィギュレーションコマンドを使用します。これらのオプションをサーバー単位で設定するには、**radius-server timeout**、**radius-server transmit**、および **radius-server key** グローバルコンフィギュレーションコマンドを使用します。



(注) RADIUS サーバーでは、スイッチの IP アドレス、サーバーとスイッチで共有される **key string**、およびダウンロード可能な ACL (DACL) などの設定を行う必要があります。詳細については、RADIUS サーバーのマニュアルを参照してください。

- URL リダイレクト ACL の場合：
 - 許可アクセス コントロール エントリ (ACE) ルールに一致するパケットは、AAA サーバーに転送するために CPU に送信されます。
 - 拒否 ACE ルールに一致するパケットは、スイッチを介して転送されます。
 - 許可 ACE ルールにも拒否 ACE ルールにも一致しないパケットは、次の dACL によって処理されます。dACL がない場合、パケットは暗黙的拒否 ACL にヒットしてドロップされます。

認証ルールとインターフェイスの設定

認証ルールおよびインターフェイスを設定するには、次の手順を実行します。

始める前に

SISF ベースのデバイス追跡は、web 認証の前提条件です。デバイストラッキングをプログラムまたは手動で有効にしていることを確認します。

詳細については、「*SISF* ベースの追跡の設定」を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission name name proxy http**
4. **interface type slot/port**
5. **ip access-group name**
6. **ip admission name**
7. **exit**

8. show ip admission

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip admission name name proxy http 例： Device(config)# ip admission name webauth1 proxy http	Web ベース許可の認証ルールを設定します。
ステップ 4	interface type slot/port 例： Device(config)# interface gigabitethernet 1/0/1	インターフェイス コンフィギュレーション モードを開始し、Web ベース認証を有効にする受信レイヤ 2 またはレイヤ 3 インターフェイスを指定します。 <i>type</i> には、FastEthernet、GigabitEthernet、または TenGigabitEthernet を指定できます。
ステップ 5	ip access-group name 例： Device(config-if)# ip access-group webauthag	デフォルト ACL を適用します。
ステップ 6	ip admission name 例： Device(config)# ip admission name	インターフェイスの Web ベース認可の認証ルールを設定します。
ステップ 7	exit 例： Device# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	show ip admission 例 : Device# show ip admission	ネットワークアドミSSIONのキャッシュエントリと Web 認証セッションに関する情報を表示します。

AAA 認証の設定

VTY 回線で方式リストを設定する場合、対応する方式リストを AAA 設定に追加する必要があります。

```
Device(config)# line vty 0 4
Device(config-line)# authorization commands 15 list1
Device(config-line)# exit
Device(config)# aaa authorization commands 15 list1 group tacacs+
```

VTY 回線で方式リストを設定しない場合、デフォルトの方式リストを AAA 設定に追加する必要があります。

```
Device(config)# line vty 0 4
Device(config-line)# exit
Device(config)# aaa authorization commands 15 default group tacacs+
```

AAA 認証を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default group {tacacs+ | radius}**
5. **aaa authorization auth-proxy default group {tacacs+ | radius}**
6. **tacacs server server-name**
7. **address {ipv4 | ipv6} ip address**
8. **key** 文字列
9. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA 機能を有効にします。
ステップ 4	aaa authentication login default group {tacacs+ radius} 例： Device(config)# aaa authentication login default group tacacs+	ログイン時の認証方法のリストを定義します。 named_authentication_list は、31 文字未満の名前を示します。 AAA_group_name はサーバー グループ名を示します。サーバーグループ server_name をその先頭で定義する必要があります。
ステップ 5	aaa authorization auth-proxy default group {tacacs+ radius} 例： Device(config)# aaa authorization auth-proxy default group tacacs+	Web ベース許可の許可方式リストを作成します。
ステップ 6	tacacs server server-name 例： Device(config)# tacacs server yourserver	AAA サーバーを指定します。
ステップ 7	address {ipv4 ipv6} ip address 例： Device(config-server-tacacs)# address ipv4 10.0.1.12	TACACS サーバーの IP アドレスを設定します。
ステップ 8	key 文字列 例： Device(config-server-tacacs)# key cisco123	スイッチと TACACS サーバーとの間で使用される許可および暗号キーを設定します。

	コマンドまたはアクション	目的
ステップ 9	end 例 : Device(config-server-tacacs) # end	TACACS サーバモードを終了し、特権 EXEC モードに戻ります。

スイッチ/RADIUS サーバー間通信の設定

RADIUS サーバーのパラメータを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip radius source-interface vlan *vlan interface number***
4. **radius server *server name***
5. **address {ipv4 | ipv6} *ip address***
6. **key *string***
7. **exit**
8. **radius-server vsa send authentication *string***
9. **radius-server dead-criteria [*time seconds*] [*tries num-tries*]**
10. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip radius source-interface vlan <i>vlan interface number</i> 例 : Device(config)# ip radius source-interface vlan 80	RADIUS パケットが、指定されたインターフェイスの IP アドレスを含むように指定します。

	コマンドまたはアクション	目的
ステップ 4	radius server <i>server name</i> 例 : <pre>Device(config)# radius server rsim address ipv4 124.2.2.12</pre>	(任意) RADIUS サーバーの IP アドレスを指定します。
ステップ 5	address {ipv4 ipv6} <i>ip address</i> 例 : <pre>Device(config-radius-server)# address ipv4 10.0.1.2 auth-port 1550 acct-port 1560</pre>	RADIUS サーバーの IP アドレスを設定します。
ステップ 6	key string 例 : <pre>Device(config-radius-server)# key rad123</pre>	(任意) RADIUS サーバー上で動作する RADIUS デーモンとスイッチの間で使用する認証および暗号キーを指定します。
ステップ 7	exit 例 : <pre>Device(config-radius-server)# exit</pre>	RADIUS サーバーモードを終了して、グローバル コンフィギュレーション モードを開始します。
ステップ 8	radius-server vsa send authentication string 例 : <pre>Device(config)# radius-server vsa send authentication</pre>	RADIUS サーバーからの ACL のダウンロードをイネーブルにします。
ステップ 9	radius-server dead-criteria [time seconds] [tries num-tries] 例 : <pre>Device(config)# radius-server dead-criteria tries 45</pre>	<p>RADIUS サーバーが使用不可または切断と見なされる条件を設定します。</p> <p>RADIUS サーバーからデバイスへの応答がない時間 time を秒単位で入力します。</p> <p>RADIUS サーバーからデバイスへの有効な応答がない試行回数 tries を入力します。指定できる <i>num-tries</i> の範囲は 1 ~ 100 です。</p>

	コマンドまたはアクション	目的
		(注) デバイスがスタックの一部であるときに、試行回数の合計が45以下に設定されている場合、デバイスはシャットダウンします。より長い期間を入力することをお勧めします。試行回数の値が大きいほど、ブート中にデバイスがシャットダウンすることを回避できません。
ステップ 10	end 例： Device# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

HTTP サーバーの設定

Web ベース認証を使用するには、**device** で HTTP サーバを有効にする必要があります。このサーバーは HTTP または HTTPS のいずれかについて有効にできます。



(注) Apple の疑似ブラウザは、**ip http secure-server** コマンドを設定するだけでは開きません。**ip http server** コマンドも設定する必要があります。

HTTP または HTTPS のいずれかについてサーバーを有効にするには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **ip http secure-server**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip http server 例： Device(config)# ip http server	HTTP サーバーを有効にします。Web ベース認証機能は、HTTP サーバーを使用してホストと通信し、ユーザー認証を行います。
ステップ 4	ip http secure-server 例： Device(config)# ip http secure-server	HTTPS を有効にします。 カスタム認証プロキシ Web ページを設定するか、成功ログインのリダイレクション URL を指定します。 (注) ip http secure-server コマンドを入力したときに、セキュア認証が確実に行われるようにするには、ユーザーが HTTP 要求を送信した場合でも、ログイン ページは必ず HTTPS (セキュア HTTP) 形式になるようにします。
ステップ 5	end 例： Device# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

認証プロキシ Web ページのカスタマイズ

Web ベースの認証中、のデフォルト HTML ページではなく、代替の HTML ページがユーザーに表示されるように、Web 認証を設定できます。

カスタム認証プロキシ Web ページの使用を指定するには、次の手順を実行してください。

始める前に

device のフラッシュ メモリにカスタム HTML ファイルを保存します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission proxy http login page file** *device:login-filename*
4. **ip admission proxy http success page file** *device:success-filename*

5. `ip admission proxy http failure page file device:fail-filename`
6. `ip admission proxy http login expired page file device:expired-filename`
7. `end`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip admission proxy http login page file device:login-filename 例 : Device(config)# ip admission proxy http login page file disk1:login.htm	device のメモリ ファイルシステム内で、デフォルトのログイン ページの代わりに使用するカスタム HTML ファイルの場所を指定します。device: はフラッシュ メモリです。
ステップ 4	ip admission proxy http success page file device:success-filename 例 : Device(config)# ip admission proxy http success page file disk1:success.htm	デフォルトのログイン成功ページの代わりに使用するカスタム HTML ファイルの場所を指定します。
ステップ 5	ip admission proxy http failure page file device:fail-filename 例 : Device(config)# ip admission proxy http fail page file disk1:fail.htm	デフォルトのログイン失敗ページの代わりに使用するカスタム HTML ファイルの場所を指定します。
ステップ 6	ip admission proxy http login expired page file device:expired-filename 例 : Device(config)# ip admission proxy http login expired page file disk1:expired.htm	デフォルトのログイン失効ページの代わりに使用するカスタム HTML ファイルの場所を指定します。

	コマンドまたはアクション	目的
ステップ 7	end 例 : Device# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

成功ログインに対するリダイレクション URL の指定

認証後に内部成功 HTML ページを効果的に置き換えユーザーのリダイレクト先となる URL を指定するためには、次の手順を実行してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission proxy http success redirect *url-string***
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip admission proxy http success redirect <i>url-string</i> 例 : Device(config)# ip admission proxy http success redirect www.example.com	デフォルトのログイン成功ページの代わりにユーザーをリダイレクトする URL を指定します。
ステップ 4	end 例 : Device# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Web ベース認証パラメータの設定

クライアントが待機時間中にウォッチリストに掲載されるまで許容される失敗ログイン試行の最大回数を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission max-login-attempts *number***
4. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip admission max-login-attempts <i>number</i> 例： Device(config)# ip admission max-login-attempts 10	失敗ログイン試行の最大回数を設定します。指定できる範囲は 1 ～ 2147483647 回です。デフォルトは 5 分です。
ステップ 4	exit 例： Device# exit	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Web ベース認証ローカルバナーの設定

Web 認証が設定されているスイッチにローカルバナーを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission auth-proxy-banner http** [*banner-text* | *file-path*]
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	ip admission auth-proxy-banner http [<i>banner-text</i> <i>file-path</i>] 例： Device(config)# ip admission auth-proxy-banner http C My Switch C	ローカル バナーを有効にします。 (任意) <i>C banner-text C</i> (<i>C</i> は区切り文字)、またはバナーに表示されるファイル（たとえばロゴまたはテキストファイル）のファイルパスを入力して、カスタムバナーを作成します。
ステップ 4	end 例： Device# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Web ベース認証キャッシュ エントリの削除

Web ベース認証キャッシュ エントリを削除するには、次の手順を実行します。

手順の概要

1. **enable**
2. **clear ip auth-proxy cache** {*} | *host ip address*}
3. **clear ip admission cache** {*} | *host ip address*}

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	clear ip auth-proxy cache { <i>*</i> <i>host ip address</i> } 例： Device# clear ip auth-proxy cache 192.168.4.5	Delete 認証プロキシエントリを削除します。キャッシュエントリすべてを削除するには、アスタリスクを使用します。シングルホストのエントリを削除するには、具体的な IP アドレスを入力します。
ステップ 3	clear ip admission cache { <i>*</i> <i>host ip address</i> } 例： # clear ip admission cache 192.168.4.5	Delete 認証プロキシエントリを削除します。キャッシュエントリすべてを削除するには、アスタリスクを使用します。シングルホストのエントリを削除するには、具体的な IP アドレスを入力します。

Web ベース認証の確認

すべてのインターフェイスまたは特定のポートに対する Web ベース認証設定を表示するには、このトピックのコマンドを使用します。

表 34: 特権 EXEC 表示コマンド

コマンド	目的
show authentication sessions method webauth	FastEthernet、ギガビットイーサネット、または 10 ギガビットイーサネットのすべてのインターフェイスに対する Web ベースの認証設定を表示します。
show authentication sessions interface type slot/port[details]	FastEthernet、ギガビットイーサネット、または 10 ギガビットイーサネットの特定のインターフェイスに対する Web ベースの認証設定を表示します。 セッション認識型ネットワーク モードでは、 show access-session interface コマンドを使用します。

Web ベース認証の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	Web ベース認証	Web ベースの認証機能を使用して、IEEE 802.1x サプリカントを実行していないホストシステムでエンドユーザーを認証できます。

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



第 25 章

ポート単位のトラフィック制御の設定

- [ポートベースのトラフィック制御 \(627 ページ\)](#)

ポートベースのトラフィック制御

ポートベースのトラフィック制御は、特定トラフィック状態に応じてポートレベルでパケットをフィルタまたはブロックするために使用するシスコデバイス上のレイヤ2機能の組み合わせです。次のポートベースのトラフィック制御機能がサポートされています。

- ストーム制御
- 保護ポート
- ポートブロッキング

ポートベースのトラフィック制御に関する情報

ストーム制御

ストーム制御は、物理インターフェイスの1つで発生したブロードキャスト、マルチキャスト、またはユニキャストストームによってLAN上のトラフィックが混乱することを防ぎます。LANストームは、LANにパケットがフラッディングした場合に発生します。その結果、トラフィックが極端に増えてネットワークパフォーマンスが低下します。プロトコルスタックの実装エラー、ネットワーク構成の間違い、またはユーザによって引き起こされるDoS攻撃もストームの原因になります。

ストームコントロール（またはトラフィック抑制）は、インターフェイスからスイッチングバスを通過するパケットをモニタし、パケットがユニキャスト、マルチキャスト、またはブロードキャストのいずれであるかを判別します。スイッチは、1秒間に受け取った特定のタイプのパケットの数をカウントして、事前に定義された抑制レベルのしきい値とその測定結果を比較します。

ストーム制御ポリサのハードウェアレートリミッタ機能では、1000パケット/秒未満のすべてのブロードキャスト、マルチキャスト、およびユニキャストパケットがブロックされます。

測定されたトラフィックアクティビティ

この動作は、ASIC ベースのプラットフォームのハードウェア制限によるものです。これらのプラットフォームでは、ストーム制御は1秒あたり 1K 以上のパケット、および1秒あたり 8K 以上のビットで動作します。

測定されたトラフィックアクティビティ

ストーム コントロールは、次のうちのいずれかをトラフィック アクティビティの測定方法に使用します。

- 帯域幅（ブロードキャスト、マルチキャスト、またはユニキャストトラフィックが使用できるポートの総帯域幅の割合）。
- 秒単位で受信するパケット（ブロードキャスト、マルチキャスト、またはユニキャスト）のトラフィック レート
- 秒単位で受信するビット（ブロードキャスト、マルチキャスト、またはユニキャスト）のトラフィック レート
- 小さいフレームのトラフィック レートの秒単位のパケット数。この機能は、グローバルにイネーブルです。小さいフレームのしきい値は、各インターフェイスで設定されます。

上記の方法のいずれを使用しても、しきい値に到達すると、ポートはトラフィックをブロックします。トラフィック レートが下限しきい値（指定されている場合）を下回らない限り、ポートはブロックされたままになり、その後、通常の転送が再開されます。下限抑制レベルが指定されていない場合、トラフィック レートが上限抑制レベルを下回るまで、デバイスはすべてのトラフィックをブロックします。一般に、そのレベルが高ければ高いほど、ブロードキャストストームに対する保護効果は薄くなります。



- (注) マルチキャストトラフィックのストーム制御しきい値に達した場合、ブリッジプロトコルデータ ユニット (BPDU) および Cisco Discovery Protocol (CDP) フレームなどの制御トラフィック以外のマルチキャストトラフィックはすべてブロックされます。ただし、デバイスでは Open Shortest Path First (OSPF) などのルーティングアップデートと、正規のマルチキャストデータトラフィックは区別されないため、両方のトラフィックタイプがブロックされます。

トラフィック パターン

T1 から T2、T4 から T5 のタイムインターバルで、転送するブロードキャストトラフィックが設定されたしきい値を上回っています。指定のトラフィック量がしきい値を上回ると、次のインターバルで、そのタイプのトラフィックがすべてドロップされます。したがって、T2 と T5 の後のインターバルの間、ブロードキャストトラフィックがブロックされます。その次のインターバル（たとえば、T3）では、しきい値を上回らない限り、ブロードキャストトラフィックが再び転送されます。

ストーム制御抑制レベルと1秒間のインターバルを組み合わせると、ストーム制御アルゴリズムの動作を制御します。しきい値が高いほど、通過できるパケット数が多くなります。しきい値が 100% であれば、トラフィックに対する制限はありません。値を 0.0 にすると、そのポート

上ではすべてのブロードキャスト、マルチキャスト、またはユニキャストトラフィックがブロックされます。



- (注) パケットは一定の間隔で届くわけではないので、トラフィックアクティビティを測定する1秒間のインターバルがストーム制御の動作を左右する可能性があります。

各トラフィックタイプのしきい値を設定するには、**storm-control** インターフェイス コンフィギュレーション コマンドを使用します。

保護ポート

アプリケーションによっては、あるネイバーが生成したトラフィックが別のネイバーにわからないように、同一デバイス上のポート間でレイヤ2トラフィックが転送されないように設定する必要があります。このような環境では、保護ポートを使用すると、デバイス上のポート間でユニキャスト、ブロードキャスト、またはマルチキャストトラフィックの交換が確実になくなります。

保護ポートには、次の機能があります。

- 保護ポートは、同様に保護ポートになっている他のポートに対して、ユニキャスト、マルチキャスト、またはブロードキャストトラフィックを転送しません。データトラフィックはレイヤ2の保護ポート間で転送されません。PIM パケットなどは CPU で処理されてソフトウェアで転送されるため、このような制御トラフィックだけが転送されます。保護ポート間を通過するすべてのデータトラフィックは、レイヤ3デバイスを介して転送されなければなりません。
- 保護ポートと非保護ポート間の転送動作は、通常どおりに進みます。

デバイススタックは論理的には1つのデバイスを表しているため、レイヤ2トラフィックは、スタック内の同一デバイスか異なるデバイスかにかかわらず、デバイススタックの保護ポート間では転送されません。

保護ポートのガイドライン

保護ポートは、物理インターフェイス (GigabitEthernet ポート 1 など) または EtherChannel グループ (port-channel 5 など) に設定できます。ポート チャンネルで保護ポートをイネーブルにした場合は、そのポート チャンネル グループ内のすべてのポートでイネーブルになります。

デフォルトでは、保護ポートは定義されていません。

ポート ブロッキング

デフォルトでは、デバイスは未知の宛先 MAC アドレスが指定されたパケットをすべてのポートからフラグディングします。未知のユニキャストおよびマルチキャストトラフィックが保護ポートに転送されると、セキュリティ上、問題になる可能性があります。未知のユニキャストおよびマルチキャストトラフィックがあるポートから別のポートに転送されないようにするた

めに、（保護または非保護）ポートをブロックし、未知のユニキャストまたはマルチキャストパケットが他のポートにフラディングされないようにします。



- (注) マルチキャストトラフィックでは、ポートブロッキング機能は純粋なレイヤ2パケットだけをブロックします。ヘッダーにIPv4またはIPv6の情報を含むマルチキャストパケットはブロックされません。

ポートベースのトラフィック制御の設定方法

ストーム制御およびしきい値レベルの設定

ポートにストーム制御を設定し、特定のトラフィックタイプで使用するしきい値レベルを入力します。

ただし、ハードウェアの制約とともに、さまざまなサイズの packets をどのように数えるかという問題があるので、しきい値の割合はあくまでも近似値です。着信トラフィックを形成するパケットのサイズによって、実際に適用されるしきい値は設定されたレベルに対して、数%の差異が生じる可能性があります。



- (注) ストーム制御は、物理インターフェイスでサポートされています。また、EtherChannelでもストーム制御を設定できます。ストーム制御をEtherChannelで設定する場合、ストーム制御設定はEtherChannel物理インターフェイスに伝播します。

ストーム制御としきい値レベルを設定するには、次の手順を実行します。

始める前に

ストーム制御は、物理インターフェイスでサポートされています。また、EtherChannelでもストーム制御を設定できます。ストーム制御をEtherChannelで設定する場合、ストーム制御設定はEtherChannel物理インターフェイスに伝播します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **storm-control {broadcast | multicast | unicast} level {*level* [*level-low*] | bps *bps* [*bps-low*] | pps *pps* [*pps-low*]}**
5. **storm-control action {shutdown | trap}**
6. **storm-control unknown-unicast**
7. **end**
8. **show storm-control [*interface-id*] [broadcast | multicast | unicast]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例： Device(config)# interface gigabitethernet1/0/1	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	storm-control {broadcast multicast unicast} level {level [level-low] bps bps [bps-low] pps pps [pps-low]} 例： Device(config-if)# storm-control unicast level 87 65	ブロードキャスト、マルチキャスト、またはユニキャストストーム制御を設定します。デフォルトでは、ストーム制御はディセーブルに設定されています。 <ul style="list-style-type: none"> level には、ブロードキャスト、マルチキャスト、またはユニキャストトラフィックの上限しきい値レベルを帯域幅のパーセンテージで指定します（小数点第2位まで）。上限しきい値に到達すると、ポートはトラフィックをブロックします。指定できる範囲は0.00～100.00です。 （任意）level-low には、下限しきい値レベルを帯域幅のパーセンテージで指定します（小数点第2位まで）。この値は上限抑制値より小さいか、または等しくなければなりません。トラフィックがこのレベルを下回っていれば、ポートはトラフィックを転送します。下限抑制レベルを設定しない場合、上限抑制レベルの値に設定されます。指定できる範囲は0.00～100.00です。 しきい値に最大値（100%）を指定した場合、トラフィックの制限はなくなります。しきい値に0.0を設定すると、そのポート上のすべてのブロードキャスト、マルチキャスト、またはユニキャストトラフィックがブロックされます。 <ul style="list-style-type: none"> bps bps には、ブロードキャスト、マルチキャスト、またはユニキャストトラフィックの上限し

	コマンドまたはアクション	目的
		<p>きい値レベルをビット/秒で指定します（小数点第1位まで）。上限しきい値に到達すると、ポートはトラフィックをブロックします。指定できる範囲は0.0～10000000000.0です。</p> <ul style="list-style-type: none"> （任意）<i>bps-low</i>には、下限しきい値レベルをビット/秒で指定します（小数点第1位まで）。この値は上限しきい値レベル以下の値である必要があります。トラフィックがこのレベルを下回っていれば、ポートはトラフィックを転送します。指定できる範囲は0.0～10000000000.0です。 <i>pps pps</i>には、ブロードキャスト、マルチキャスト、またはユニキャストトラフィックの上限しきい値レベルをパケット/秒で指定します（小数点第1位まで）。上限しきい値に到達すると、ポートはトラフィックをブロックします。指定できる範囲は0.0～10000000000.0です。 （任意）<i>pps-low</i>には、下限しきい値レベルをパケット/秒で指定します（小数点第1位まで）。この値は上限しきい値レベル以下の値である必要があります。トラフィックがこのレベルを下回っていれば、ポートはトラフィックを転送します。指定できる範囲は0.0～10000000000.0です。 <p>BPSおよびPPSの設定には、しきい値の数値を大きく設定できるように、サフィックスに測定記号（k、m、gなど）を使用できます。</p>
<p>ステップ5</p>	<p>storm-control action {shutdown trap}</p> <p>例：</p> <pre>Device(config-if)# storm-control action trap</pre>	<p>ストーム検出時に実行するアクションを指定します。ストームが検出されると、shutdownまたはtrapアクションがすべてのトラフィックに適用されます。デフォルトではトラフィックにフィルタリングを実行し、トラップは送信しない設定です。</p> <ul style="list-style-type: none"> ストーム中、ポートをerrdisableの状態にするには、shutdownキーワードを選択します。 ストームが検出された場合、SNMPトラップを生成するには、trapキーワードを選択します。
<p>ステップ6</p>	<p>storm-control unknown-unicast</p> <p>例：</p>	<p>ストーム制御用の不明なユニキャストアドレスを指定します。</p>

	コマンドまたはアクション	目的
	Device(config-if)# storm-control unknown-unicast	
ステップ 7	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 8	show storm-control [interface-id] [broadcast multicast unicast] 例： Device# show storm-control gigabitethernet1/0/1 unicast	指定したトラフィック タイプについて、インターフェイスで設定したストーム制御抑制レベルを確認します。トラフィックタイプを入力しない場合は、すべてのトラフィックタイプ（ブロードキャスト、マルチキャスト、ユニキャスト）の詳細が表示されます。

保護ポートの設定

始める前に

保護ポートは事前定義されていません。これは設定する必要があるタスクです。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **switchport protected**
5. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	interface <i>interface-id</i> 例： Device(config)# interface gigabitethernet 1/0/1	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	switchport protected 例： Device(config-if)# switchport protected	インターフェイスを保護ポートとして設定します。
ステップ 5	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

保護ポートの監視

表 35: 保護ポートの設定を表示するコマンド

コマンド	目的
show interfaces [<i>interface-id</i>] switchport	すべてのスイッチング（非ルーティング）ポートまたはポートの管理ステータスまたは動作ステータスを、ロックングおよびポート保護の設定を含めて表示します。

インターフェイスでのフラディング トラフィックのブロッキング

始める前に

インターフェイスは物理インターフェイスまたは EtherChannel グループのいずれも可能です。ポート チャネルのマルチキャストまたはユニキャスト トラフィックをブロックすると、ポート チャネル グループのすべてのポートでブロックされます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport block multicast**
5. **switchport block unicast**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例： Device(config)# interface gigabitethernet 1/0/1	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 4	switchport block multicast 例： Device(config-if)# switchport block multicast	ポートからの未知のマルチキャストの転送をブロックします。
ステップ 5	switchport block unicast 例： Device(config-if)# switchport block unicast	ポートからの未知のユニキャストの転送をブロックします。
ステップ 6	end 例： Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ポートブロッキングの監視

表 36: ポートブロッキングの設定を表示するコマンド

コマンド	目的
show interfaces [interface-id] switchport	すべてのスイッチング（非ルーティング）ポートまたはポートの管理ステータスまたは動作ステータス、ブロッキングおよびポート保護の設定を含めて表示

ポートベースのトラフィック制御に関するその他の関連資料

関連資料

関連項目	マニュアルタイトル
ポートセキュリティ	『セキュリティ コンフィギュレーション ガイド』の「ポートセキュリティ」

シスコのテクニカル サポート

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

ポートベースのトラフィック制御の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	ポートベースのトラフィック制御	ポートベースのトラフィック制御は、特定トラフィック状態に応じてポート レベルでパケットをフィルタまたはブロックするために使用する Cisco Catalyst スイッチ上のレイヤ 2 機能の組み合わせです。

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



第 26 章

ポート セキュリティ

- [ポートセキュリティの前提条件 \(637 ページ\)](#)
- [ポートセキュリティの制約事項 \(637 ページ\)](#)
- [ポートセキュリティの概要 \(638 ページ\)](#)
- [ポートセキュリティの設定方法 \(644 ページ\)](#)
- [ポートセキュリティの設定例 \(652 ページ\)](#)
- [ポートセキュリティの機能の履歴 \(653 ページ\)](#)

ポート セキュリティの前提条件

最大値をインターフェイス上ですでに設定されているセキュアアドレスの数より小さい値に設定しようとする、コマンドが拒否されます。

ポート セキュリティの制約事項

- スイッチに設定できるセキュア MAC アドレスの最大数は、システムで許可されている MAC アドレスの最大数によって決まります。この値は、使用可能な MAC アドレス（その他のレイヤ 2 機能やインターフェイスに設定されたその他のセキュア MAC アドレスで使用する MAC アドレスを含む）の総数を表します。
- ポートセキュリティは、EtherChannel インターフェイスではサポートされていません。
- ポートセキュリティは、プライベート VLAN ポートではサポートされていません。

ポートセキュリティの概要

ポートセキュリティ

ポートセキュリティ機能を使用すると、ポートへのアクセスを許可するステーションの MAC アドレスを制限および識別して、インターフェイスへの入力を制限できます。セキュアポートにセキュア MAC アドレスを割り当てると、ポートは定義されたアドレスグループ以外の送信元アドレスを持つパケットを転送しません。セキュア MAC アドレス数を 1 つに制限し、単一のセキュア MAC アドレスを割り当てると、そのポートに接続されたワークステーションに、ポートの帯域幅全体が保証されます。

セキュアポートとしてポートを設定し、セキュア MAC アドレスが最大数に達した場合、ポートにアクセスを試みるステーションの MAC アドレスが識別されたセキュア MAC アドレスのいずれとも一致しないので、セキュリティ違反が発生します。また、あるセキュアポート上でセキュア MAC アドレスが設定または学習されているステーションが、別のセキュアポートにアクセスしようとしたときにも、違反のフラグが立てられます。

セキュア MAC アドレスのタイプ

スイッチは、次のセキュア MAC アドレス タイプをサポートします。

- **スタティックセキュア MAC アドレス** : `switchport port-security mac-address mac-address` インターフェイス コンフィギュレーション コマンドを使用して手動で設定され、アドレステーブルに保存された後、スイッチの実行コンフィギュレーションに追加されます。
- **ダイナミックセキュア MAC アドレス** : 動的に設定されてアドレステーブルにのみ保存され、スイッチの再起動時に削除されます。
- **スティッキーセキュア MAC アドレス** : 動的に学習することも、手動で設定することもできます。アドレステーブルに保存され、実行コンフィギュレーションに追加されます。このアドレスがコンフィギュレーションファイルに保存されていると、スイッチの再起動時にインターフェイスはこれらを動的に再設定する必要がありません。

MAC アドレス テーブルのデフォルト設定

次の表に、MAC アドレス テーブルのデフォルト設定を示します。

表 37: MAC アドレスのデフォルト設定

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

機能	デフォルト設定
スタティック アドレス	未設定

MAC アドレス テーブルの作成

すべてのポートでサポートされる複数の MAC アドレスを使用して、他のネットワークデバイスにデバイス上のすべてのポートを接続できます。デバイスは、各ポートで受信するパケットの送信元アドレスを取得し、アドレステーブルにアドレスとそれに関連付けられたポート番号を追加することによって、動的なアドレス指定を行います。ネットワークでデバイスの追加または削除が行われると、デバイスによってアドレステーブルが更新され、新しいダイナミックアドレスが追加され、使用されていないアドレスは期限切れになります。

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

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

スティッキ セキュア MAC アドレス

スティッキー ラーニングをイネーブルにすると、ダイナミック MAC アドレスをスティッキー セキュア MAC アドレスに変換して実行コンフィギュレーションに追加するようにインターフェイスを設定できます。インターフェイスはスティッキー ラーニングがイネーブルになる前に学習したものを含め、すべてのダイナミック セキュア MAC アドレスをスティッキー セキュア MAC アドレスに変換します。すべてのスティッキー セキュア MAC アドレスは実行コンフィギュレーションに追加されます。

スティッキー セキュア MAC アドレスは、コンフィギュレーション ファイル（スイッチが再起動されるたびに使用されるスタートアップコンフィギュレーション）に、自動的に反映されません。スティッキー セキュア MAC アドレスをコンフィギュレーション ファイルに保存すると、スイッチの再起動時にインターフェイスはこれらを再び学習する必要がありません。スティッキ セキュア アドレスを保存しない場合、アドレスは失われます。

スティッキ ラーニングがディセーブルの場合、スティッキ セキュア MAC アドレスはダイナミック セキュア アドレスに変換され、実行コンフィギュレーションから削除されます。

セキュリティ違反

次のいずれかの状況が発生すると、セキュリティ違反になります。

- 最大数のセキュア MAC アドレスがアドレステーブルに追加されている状態で、アドレステーブルに未登録の MAC アドレスを持つステーションがインターフェイスにアクセスしようとした場合。
- あるセキュア インターフェイスで学習または設定されたアドレスが、同一 VLAN 内の別のセキュア インターフェイスで使用された場合。
- ポートセキュリティが有効な状態で診断テストを実行しています。

違反が発生した場合の対処に基づいて、次の3種類の違反モードのいずれかにインターフェイスを設定できます。

- **protect (保護)** : セキュア MAC アドレスの数がポートで許可されている最大限度に達すると、最大値を下回るまで十分な数のセキュア MAC アドレスを削除するか、許可アドレス数を増やさないかぎり、未知の送信元アドレスを持つパケットはドロップされます。セキュリティ違反が起こっても、ユーザには通知されません。



(注) トランク ポートに **protect** 違反モードを設定することは推奨しません。保護モードでは、ポートが最大数に達していても VLAN が保護モードの最大数に達すると、ラーニングがディセーブルになります。

- **restrict (制限)** : セキュア MAC アドレスの数がポートで許可されている最大限度に達すると、最大値を下回るまで十分な数のセキュア MAC アドレスを削除するか、許可アドレス数を増やさないかぎり、未知の送信元アドレスを持つパケットはドロップされます。このモードでは、セキュリティ違反が発生したことが通知されます。SNMP トラップが送信されます。Syslog メッセージがロギングされ、違反カウンタが増加します。
- **shutdown (シャットダウン)** : ポートセキュリティ違反により、インターフェイスが **error-disabled** になり、ただちにシャットダウンされます。そのあと、ポートの LED が消灯します。セキュアポートが **error-disabled** 状態の場合は、**errdisable recovery cause psecure-violation** グローバル コンフィギュレーション コマンドを入力してこの状態を解消するか、**shutdown** および **no shut down** インターフェイス コンフィギュレーション コマンドを入力して手動で再度有効にできます。これは、デフォルトのモードです。
- **shutdown vlan (VLAN シャットダウン)** : VLAN 単位でセキュリティ違反モードを設定するために使用します。このモードで違反が発生すると、ポート全体ではなく、VLAN が **errdisable** になります。

次の表に、ポートセキュリティをインターフェイスに設定した場合の違反モードおよび対処について示します。

表 38: セキュリティ違反モードの処置

違反モード	トラフィックの転送 9	SNMP トラップの送信	Syslog メッセージの送信	エラーメッセージの表示 10	違反カウンタの増加
protect	非対応	非対応	非対応	非対応	非対応
restrict	非対応	対応	対応	非対応	対応
shutdown	非対応	非対応	非対応	非対応	対応
shutdown vlan	非対応	非対応	対応	非対応	対応

⁹ 十分な数のセキュア MAC アドレスを削除するまで未知の送信元アドレスを持つパケットがドロップされます。

¹⁰ セキュリティ違反を引き起こすアドレスを手動で設定した場合、スイッチがエラーメッセージを返します。

¹¹ 違反が発生した VLAN のみシャットダウンします。

ポートセキュリティ エージング

ポート上のすべてのセキュアアドレスにエージングタイムを設定するには、ポートセキュリティエージングを使用します。ポートごとに2つのタイプのエージングがサポートされています。

- **absolute** : 指定されたエージングタイムの経過後に、ポート上のセキュアアドレスが削除されます。
- **inactivity** : 指定されたエージングタイムの間、セキュアアドレスが非アクティブであった場合に限り、ポート上のセキュアアドレスが削除されます。

ポートセキュリティとスイッチスタック

スタックに新規に加入したスイッチは、設定済みのセキュアアドレスを取得します。他のスタックメンバーから新しいスタックメンバーに、ダイナミックセキュアアドレスがすべてダウンロードされます。

スイッチ（アクティブスイッチまたはスタックメンバのいずれか）がスタックから離れると、その他のスタックメンバに通知が行き、そのスイッチが設定または学習したセキュア MAC アドレスがセキュア MAC アドレステーブルから削除されます。

デフォルトのポートセキュリティ設定

表 39: デフォルトのポートセキュリティ設定

機能	デフォルト設定
ポートセキュリティ	ポート上でディセーブル
スティッキーアドレスラーニング	ディセーブル
ポートあたりのセキュア MAC アドレスの最大数	1つのアドレス
違反モード	shutdown。セキュア MAC アドレスが最大数を上回ると、ポートがシャットダウンします。
ポートセキュリティエージング	ディセーブルエージング タイムは 0 スタティック エージングはディセーブル タイプは absolute

ポートセキュリティの設定時の注意事項

- ポートセキュリティを設定できるのは、スタティック アクセス ポートまたはトランク ポートに限られます。セキュア ポートをダイナミック アクセス ポートにすることはできません。
- セキュア ポートをスイッチド ポート アナライザ (SPAN) の宛先ポートにすることはできません。
- 音声 VLAN はアクセス ポートでのみサポートされており、設定可能であってもトランク ポートではサポートされていません。
- 音声 VLAN が設定されたインターフェイス上でポートセキュリティをイネーブルにする場合は、ポートの最大セキュアアドレス許容数を 2 に設定します。ポートを Cisco IP Phone に接続する場合は、IP Phone に MAC アドレスが 1 つ必要です。Cisco IP Phone のアドレスは音声 VLAN 上で学習されますが、アクセス VLAN 上では学習されません。1 台の PC を Cisco IP Phone に接続する場合は、MAC アドレスの追加は必要ありません。複数の PC を Cisco IP Phone に接続する場合は、各 PC と IP Phone に 1 つずつ使用できるように、十分な数のセキュアアドレスを設定する必要があります。
- トランクポートがポートセキュリティで設定され、データトラフィック用のアクセス VLAN と音声トラフィック用の音声 VLAN に割り当てられている場合、**switchport voice** およびインターフェイス コンフィギュレーション コマンドを入力して **switchport priority extend** も効果はありません。

接続装置が同じ MAC アドレスを使用してアクセス VLAN の IP アドレス、音声 VLAN の IP アドレスの順に要求すると、アクセス VLAN だけが IP アドレスに割り当てられます。

- インターフェイスの最大セキュアアドレス値を入力したときに、新しい値がそれまでの値より大きいと、それまで設定されていた値が新しい値によって上書きされます。新しい値が前回の値より小さく、インターフェイスで設定されているセキュアアドレス数が新しい値より大きい場合、コマンドは拒否されます。
- スイッチはスティックセキュア MAC アドレスのポートセキュリティエージングをサポートしていません。

次の表に、他のポートベース機能と互換性のあるポートセキュリティについてまとめます。

表 40: ポートセキュリティと他のポートベース機能との互換性

ポートタイプまたはポートの機能	ポートセキュリティとの互換性
DTP ¹² ポート ¹³	なし
トランク ポート	あり
ダイナミックアクセスポート ¹⁴	なし
ルーテッド ポート	なし
SPAN 送信元ポート	あり
SPAN 宛先ポート	なし
EtherChannel	非対応
トンネリング ポート	あり
保護ポート	あり
IEEE 802.1x ポート	あり
音声 VLAN ポート ¹⁵	あり
IP ソース ガード	あり
ダイナミック アドレス解決プロトコル (ARP) インспекション	あり
Flex Link	対応

¹² DTP = Dynamic Trunking Protocol

¹³ **switchport mode dynamic** インターフェイス コンフィギュレーション コマンドで設定されたポート A。

¹⁴ **switchport access vlan dynamic** インターフェイス コンフィギュレーション コマンドで設定される VLAN Query Protocol (VQP) ポート。

- ¹⁵ ポートに最大限可能なセキュアなアドレスを設定します（アクセスVLANで可能なセキュアなアドレスの最大数に2を加えた数）。

ポートセキュリティの設定方法

ポートセキュリティのイネーブル化および設定

始める前に

このタスクは、ポートにアクセスできるステーションのMACアドレスを制限および識別して、インターフェイスへの入力を制約します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport mode {access | trunk}**
5. **switchport voice vlan *vlan-id***
6. **switchport port-security**
7. **switchport port-security [maximum *value* [vlan {*vlan-list* | {access | voice}}]]**
8. **switchport port-security violation {protect | restrict | shutdown | shutdown vlan}**
9. **switchport port-security [mac-address *mac-address* [vlan {*vlan-id* | {access | voice}}]]**
10. **switchport port-security mac-address sticky**
11. **switchport port-security mac-address sticky [*mac-address* | vlan {*vlan-id* | {access | voice}}]]**
12. **end**
13. **show port-security**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface <i>interface-id</i> 例：	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。

	コマンドまたはアクション	目的
	Device(config)# interface gigabitethernet 1/0/1	
ステップ 4	switchport mode {access trunk} 例： Device(config-if)# switchport mode access	インターフェイス スイッチポート モードを access または trunk に設定します。デフォルト モード (dynamic auto) のインターフェイスは、セキュア ポートとして設定できません。
ステップ 5	switchport voice vlan vlan-id 例： Device(config-if)# switchport voice vlan 22	ポート上で音声 VLAN をイネーブルにします。 <i>vlan-id</i> : 音声トラフィックに使用する VLAN を指定します。
ステップ 6	switchport port-security 例： Device(config-if)# switchport port-security	インターフェイス上でポートセキュリティをイネーブルにします。 (注) 特定の条件下では、スイッチ スタックのメンバー ポートでポートセキュリティが有効になっていると、DHCP および ARP パケットがドロップされます。回避策として、インターフェイスをシャットダウンした後に no shutdown コマンドを設定します。
ステップ 7	switchport port-security [maximum value [vlan {vlan-list {access voice}}]] 例： Device(config-if)# switchport port-security maximum 20	(任意) インターフェイスの最大セキュア MAC アドレス数を設定します。スイッチまたはスイッチ スタックに設定できるセキュア MAC アドレスの最大数は、システムで許可されている MAC アドレスの最大数によって決まります。この値は、使用可能な MAC アドレス (その他のレイヤ 2 機能やインターフェイスに設定されたその他のセキュア MAC アドレスで使用される MAC アドレスを含む) の総数を表します。 (任意) vlan : VLAN 当たりの最大値を設定します。 vlan キーワードを入力後、次のいずれかのオプションを入力します。 • <i>vlan-list</i> : トランク ポート上で、ハイフンで区切った範囲の VLAN、またはカンマで区切った一連の VLAN における、VLAN 単位の最大値を設定できます。VLAN を指定しない場合、VLAN ごとの最大値が使用されます。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • access : アクセスポートで、VLANをアクセスVLANとして指定します。 • voice : アクセスポートで、VLANを音声VLANとして指定します。 <p>(注) voice キーワードは、音声VLANがポートに設定されていて、さらにそのポートがアクセスVLANでない場合のみ有効です。インターフェイスに音声VLANが設定されている場合、セキュアMACアドレスの最大数を2に設定します。</p>
ステップ 8	<p>switchport port-security violation {protect restrict shutdown shutdown vlan}</p> <p>例 :</p> <pre>Device(config-if)# switchport port-security violation restrict</pre>	<p>(任意) 違反モードを設定します。セキュリティ違反が発生した場合に、次のいずれかのアクションを実行します。</p> <ul style="list-style-type: none"> • protect : ポートセキュアMACアドレスの数がポートで許可されている最大限度に達すると、最大値を下回るまで十分な数のセキュアMACアドレスを削除するか、許可アドレス数を増やさない限り、未知の送信元アドレスを持つパケットはドロップされます。セキュリティ違反が起こっても、ユーザには通知されません。 (注) トランクポート上に保護モードを設定することは推奨できません。保護モードでは、ポートが最大数に達していなくてもVLANが保護モードの最大数に達すると、ラーニングがディセーブルになります。 • restrict : セキュアMACアドレス数がポートで許可されている最大数に到達した場合、不明な送信元アドレスのパケットはドロップされます。セキュアMACアドレス数を上限よりも少なくするか、許容できるアドレスの最大数を増やさない限り、この状態が続きます。SNMPトラップが送信されます。Syslogメッセージがロギングされ、違反カウンタが増加します。 • shutdown : 違反が発生すると、インターフェイスがerror-disabledになり、ポートのLEDが消灯します。SNMPトラップが送信されます。

	コマンドまたはアクション	目的
		<p>Syslog メッセージがロギングされ、違反カウンタが増加します。</p> <ul style="list-style-type: none"> • shutdown vlan : VLAN 単位でセキュリティ違反モードを設定するために使用します。このモードで違反が発生すると、ポート全体ではなく、VLAN が errdisable になります。 <p>(注) セキュア ポートが error-disabled ステータスの場合は、errdisable recovery cause psecure-violation グローバル コンフィギュレーション コマンドを入力して、このステータスから回復させることができます。手動で再びイネーブルにするには、shutdown および no shutdown インターフェイス コンフィギュレーション コマンドを入力するか、clear errdisable interface vlan 特権 EXEC コマンドを入力します。</p>
<p>ステップ 9</p>	<p>switchport port-security [mac-address mac-address [vlan {vlan-id {access voice}}]</p> <p>例 :</p> <pre>DEvice (config-if) # switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice</pre>	<p>(任意) インターフェイスのセキュア MAC アドレスを入力します。このコマンドを使用すると、最大数のセキュア MAC アドレスを入力できます。設定したセキュア MAC アドレスが最大数より少ない場合、残りの MAC アドレスは動的に学習されます。</p> <p>(注) このコマンドの入力後にスティッキーラーニングをイネーブルにすると、動的に学習されたセキュアアドレスがスティッキーセキュア MAC アドレスに変換されて実行コンフィギュレーションに追加されます。</p> <p>(任意) vlan : VLAN 当たりの最大値を設定します。</p> <p>vlan キーワードを入力後、次のいずれかのオプションを入力します。</p> <ul style="list-style-type: none"> • vlan-id : トランク ポートで、VLAN ID および MAC アドレスを指定できます。VLAN ID を指定しない場合、ネイティブ VLAN が使用されます。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • access : アクセスポートで、VLANをアクセス VLANとして指定します。 • voice : アクセスポートで、VLANを音声VLANとして指定します。 <p>(注) voice キーワードは、音声VLANがポートに設定されていて、さらにそのポートがアクセスVLANでない場合のみ有効です。インターフェイスに音声VLANが設定されている場合、セキュアMACアドレスの最大数を2に設定します。</p>
ステップ 10	switchport port-security mac-address sticky 例 : <pre>Device(config-if)# switchport port-security mac-address sticky</pre>	(任意) インターフェイス上でスティッキーラーニングをイネーブルにします。
ステップ 11	switchport port-security mac-address sticky <code>[mac-address vlan {vlan-id} {access voice}]</code> 例 : <pre>Device(config-if)# switchport port-security mac-address sticky 00:A0:C7:12:C9:25 vlan voice</pre>	<p>(任意) スティックキーセキュアMACアドレスを入力し、必要な回数だけコマンドを繰り返します。設定したセキュアMACアドレスの数が最大数より少ない場合、残りのMACアドレスは動的に学習されてスティッキーセキュアMACアドレスに変換され、実行コンフィギュレーションに追加されます。</p> <p>(注) このコマンドの入力前にスティッキーラーニングをイネーブルにしないと、エラーメッセージが表示されてスティッキーセキュアMACアドレスを入力できません。</p> <p>(任意) vlan : VLAN当たりの最大値を設定します。</p> <p>vlan キーワードを入力後、次のいずれかのオプションを入力します。</p> <ul style="list-style-type: none"> • vlan-id : トランクポートで、VLAN IDおよびMACアドレスを指定できます。VLAN IDを指定しない場合、ネイティブVLANが使用されます。 • access : アクセスポートで、VLANをアクセスVLANとして指定します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • voice : アクセスポートで、VLANを音声VLANとして指定します。 <p>(注) voice キーワードは、音声VLANがポートに設定されていて、さらにそのポートがアクセスVLANでない場合のみ有効です。</p>
ステップ 12	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 13	show port-security 例 : Device# show port-security	ポートセキュリティ設定に関する情報を表示します。

ポートセキュリティ エージングのイネーブル化および設定

この機能を使用すると、既存のセキュア MAC アドレスを手動で削除しなくても、セキュアポート上のデバイスを削除および追加し、なおかつポート上のセキュアアドレス数を制限できます。セキュアアドレスのエージングは、ポート単位でイネーブルまたはディセーブルにできます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **switchport port-security aging {static | time time | type {absolute | inactivity}}**
5. **end**
6. **show port-security [interface interface-id] [address]**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	interface interface-id 例 : Device(config)# interface gigabitethernet1/0/1	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 4	switchport port-security aging {static time time type {absolute inactivity}} 例 : Device(config-if)# switchport port-security aging time 120	<p>セキュアポートのスタティックエージングをイネーブルまたはディセーブルにします。またはエージング タイムやタイプを設定します。</p> <p>(注) スイッチは、スティッキーセキュアアドレスのポートセキュリティ エージングをサポートしていません。</p> <p>このポートに、スタティックに設定されたセキュアアドレスのエージングをイネーブルにする場合は、staticを入力します。</p> <p>timeには、このポートのエージングタイムを指定します。指定できる範囲は、0 ~ 1440 分です。</p> <p>typeには、次のキーワードのいずれか1つを選択します。</p> <ul style="list-style-type: none"> • absolute : (任意) エージングタイプを絶対エージングとして設定します。このポートのセキュアアドレスはすべて、指定した時間 (分単位) が経過すると期限切れになり、セキュアアドレス リストから削除されます。 • inactivity : (任意) エージングタイプを非アクティブエージングとして設定します。指定された time 期間中にセキュア送信元アドレスからのデータ トラフィックがない場合に限り、このポートのセキュアアドレスが期限切れになります。
ステップ 5	end 例 : Device(config-if)# end	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 6	show port-security [interface <i>interface-id</i>] [address] 例 : Device# show port-security interface gigabitethernet1/0/1	指定したインターフェイスでのポートセキュリティ設定に関する情報を表示します。

アドレス エージング タイムの変更

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

手順の概要

1. **enable**
2. **configure terminal**
3. **mac address-table aging-time [0 | 10-1000000] [routed-mac | vlan *vlan-id*]**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none"> • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	mac address-table aging-time [0 10-1000000] [routed-mac vlan <i>vlan-id</i>] 例 : Device(config)# mac address-table aging-time 500 vlan 2	ダイナミック エントリが使用または更新された後、MAC アドレス テーブル内に保持される時間を設定します。 指定できる範囲は 10 ~ 1000000 秒です。デフォルトは 300 です。0 を入力して期限切れをディセーブにすることもできます。スタティック アドレスは、期限切れになることもテーブルから削除されることもありません。 <i>vlan-id</i> : 有効な ID は 1 ~ 4094 です。
ステップ 4	end 例 :	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

コマンドまたはアクション	目的
Device(config)# end	

ポートセキュリティの監視

次の表に、ポートセキュリティ情報を表示します。

表 41: ポートセキュリティのステータスおよび設定を表示するコマンド

コマンド	目的
show port-security [interface <i>interface-id</i>]	デバイスまたは指定されたインターフェイスのポートセキュリティ設定を、各インターフェイスで許容されるセキュア MAC アドレスの最大数、インターフェイスのセキュア MAC アドレス、発生したセキュリティ違反の数、違反モードを含めて表示します。
show port-security [interface <i>interface-id</i>] address	すべてのデバイスインターフェイスまたは指定されたインターフェイスに設定されたすべてのセキュア MAC アドレス、アドレスのエージング情報を表示します。
show port-security interface <i>interface-id</i> vlan	指定されたインターフェイスに VLAN 単位で設定されたセキュア MAC アドレスの数を表示します。

ポートセキュリティの設定例

次に、ポート上でポートセキュリティをイネーブルにし、セキュアアドレスの最大数を 50 に設定する例を示します。違反モードはデフォルトです。スタティックセキュア MAC アドレスは設定せず、スティッキー ラーニングはイネーブルです。

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# switchport mode access
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security maximum 50
Device(config-if)# switchport port-security mac-address sticky
Device(config-if)# end
```

次に、ポートの VLAN 3 上にスタティックセキュア MAC アドレスを設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# switchport mode trunk
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security mac-address 0000.0200.0004 vlan 3
Device(config-if)# end
```

次に、ポートのスティッキー ポート セキュリティをイネーブルにする例を示します。データ VLAN および音声 VLAN の MAC アドレスを手動で設定し、セキュアアドレスの総数を 20 に設定します（データ VLAN に 10、音声 VLAN に 10 を割り当てます）。

```
Device> enable
Device# configure terminal
Device(config)# interface tengigabitethernet1/0/1
Device(config-if)# switchport access vlan 21
Device(config-if)# switchport mode access
Device(config-if)# switchport voice vlan 22
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security maximum 20
Device(config-if)# switchport port-security violation restrict
Device(config-if)# switchport port-security mac-address sticky
Device(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Device(config-if)# switchport port-security mac-address 0000.0000.0003
Device(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Device(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Device(config-if)# switchport port-security maximum 10 vlan access
Device(config-if)# switchport port-security maximum 10 vlan voice
Device(config-if)# end
```

ポートセキュリティの機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	ポートセキュリティ	ポートセキュリティ機能で、ポートへのアクセスを許可するステーションの MAC アドレスを制限および識別して、インターフェイスへの入力を制限します。
Cisco IOS XE Everest 16.5.1a	ポートセキュリティ MAC エージング	ネットワークでデバイスの追加または削除が行われると、デバイスによってアドレステーブルが更新され、新しいダイナミックアドレスが追加され、使用されていないアドレスは期限切れになります。

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



第 27 章

コントロールプレーンポリシングの設定

- [CoPP の制約事項 \(655 ページ\)](#)
- [CoPP の概要 \(656 ページ\)](#)
- [CoPP の設定方法 \(667 ページ\)](#)
- [CoPP の設定例 \(672 ページ\)](#)
- [CoPP のモニタリング \(677 ページ\)](#)
- [CoPP の機能の履歴 \(677 ページ\)](#)

CoPP の制約事項

コントロールプレーンポリシング (CoPP) の制約事項は、次のとおりです。

- 入力 CoPP だけがサポートされます。 **system-cpp-policy** ポリシーマップは、入力方向でのみ、コントロールプレーンインターフェイスで使用可能です。
- コントロールプレーンインターフェイスにインストールできるのは、 **system-cpp-policy** ポリシーマップのみです。
- **system-cpp-policy** ポリシーマップおよびシステム定義のクラスは、変更または削除することはできません。
- **system-cpp-policy** ポリシーマップの下で許可されるのは、 **police** アクションのみです。システム定義クラスのポリシングレートは、1秒あたりのパケット数 (pps) でのみ設定する必要があります。
- 1つ以上の CPU キューがそれぞれのクラスマップの一部となります。複数の CPU キューが1つのクラスマップに属している場合、クラスマップのポリサーレートを変更すると、そのクラスマップに属しているすべての CPU キューに影響します。同様に、クラスマップでポリサーを無効にすると、そのクラスマップに属するすべてのキューが無効になります。各クラスマップに属する CPU キューの詳細については、「*CoPP* のシステム定義値」の表を参照してください。
- システム定義のクラスマップのポリサーを無効にしないこと、つまり **no police rate rate pps** コマンドを設定しないことを推奨します。これを行うと、CPU へのトラフィックが多い場合に、システム全体の正常性に影響します。さらに、システム定義のクラスマップの

ポリサーレートを無効にした場合でも、システム起動プロセスを保護するために、システムはシステムのブートアップ後にデフォルトのポリサーレートに自動的に戻ります。

- `system-cpp` ポリシーの下で設定されたクラスがデフォルト値のままの場合、それらのクラスに関する情報は `show run` コマンドで表示されません。代わりに `show policy-map system-cpp-policy` または `show policy-map control-plane` コマンドを使用します。
引き続き `show run` コマンドを使用して、カスタムポリシーに関する情報を表示できます。
- 大量の CPU バウンドパケットを使用するプロトコルは、同じクラスの他のプロトコルに影響を与える可能性があります。これらのプロトコルの一部は同じポリサーを共有するためです。たとえば、Address Resolution Protocol (ARP) は、`system-cpp-police-forus` クラスの Telnet、Internet Control Message Protocol (ICMP)、SSH、FTP、SNMP などのホストプロトコルの配列と 4000 個のハードウェアポリサーを共有します。ARP ポイズニングまたは ICMP 攻撃が発生すると、ハードウェアポリサーは、4000 パケット/秒を超える着信トラフィックのロットリングを開始し、CPU とシステムの全体的な完全性を保護します。その結果、ARP および ICMP ホストプロトコルは、同じクラスを共有する他のホストプロトコルとともにドロップされます。
- Cisco IOS XE Fuji 16.8.1a 以降、ユーザー定義のクラスマップの作成はサポートされていません。

CoPP の概要

この章では、コントロールプレーンポリシング (CoPP) がデバイスで機能する仕組みと、その設定方法について説明します。

CoPP の概要

CoPP 機能は、不要なトラフィックおよび Denial of Service (DoS) 攻撃から CPU を保護することでデバイスのセキュリティを向上させます。また、他の優先順位の低い大量のトラフィックによって発生するトラフィックのドロップから、制御トラフィックおよび管理トラフィックを保護することもできます。

デバイスは通常、3つの操作プレーンにセグメント化され、それぞれに独自の目的があります。

- データパケットを転送するための、データプレーン。
- データを適切にルーティングするための、コントロールプレーン。
- ネットワーク要素を管理するための、管理プレーン。

CoPP を使用することで、大半の CPU 行きトラフィックを保護し、ルーティングの安定性と信頼性を確保し、パケットを確実に配信することができます。特に重要なのは、DoS 攻撃から CPU を保護するために CoPP を使用できることです。

CoPP は、モジュラ QoS コマンドラインインターフェイス (MQC) および CPU キューを使用して、これらの目的を達成します。さまざまなタイプのコントロールプレーントラフィック

が特定の条件に基づいてグループ化され、CPU キューに割り当てられます。ハードウェアに専用のポリサーを設定することで、これらの CPU キューを管理できます。たとえば、特定の CPU キュー（トラフィック タイプ）のポリサー レートを変更したり、特定のタイプのトラフィックに対するポリサーを無効にしたりできます。

ポリサーはハードウェアに設定されていますが、CoPP は CPU のパフォーマンスやデータプレーンのパフォーマンスには影響しません。しかし、CPU に着信するパケット数は制限されるため、CPU 負荷が制御されます。これは、ハードウェアからのパケットを待っているサービスが、より制御された着信パケットのレート（ユーザー設定可能なレート）を確認する可能性があることを意味します。

システム定義の CoPP の特徴

デバイスの初回の電源投入時は、システムによって次のタスクが自動的に実行されます。

- ポリシーマップ **system-cpp-policy** を検索します。見つからない場合、システムはそれを作成してコントロールプレーンにインストールします。
- **system-cpp-policy** の下に 18 個のクラスマップを作成します。
次回デバイスの電源を入れたときに、すでに作成済みのポリシーマップとクラスマップがシステムによって検出されます。
- デフォルトで、すべての CPU キューをそれぞれのデフォルトレートで有効にします。デフォルトのレートを「CoPP のシステム定義値」の表に示します。

system-cpp-policy ポリシーマップはシステム デフォルト ポリシー マップであり、通常はデバイスのスタートアップ コンフィギュレーションに明示的に保存する必要はありません。ただし、スタンバイデバイスとのバルク同期に失敗すると、コンフィギュレーションがスタートアップ コンフィギュレーションから消去される可能性があります。この場合、手動で **system-cpp-policy** ポリシーマップをスタートアップ コンフィギュレーションに保存する必要があります。 **show running-config** 特権 EXEC コマンドを使用して、保存されていることを確認します。

```
policy-map system-cpp-policy
```

次の表（CoPP のシステム定義値）に、デバイスのロード時にシステムから作成されるクラスマップを示します。各クラス マップに対応するポリサーと、各クラス マップの下にグループ化された 1 つ以上の CPU キューを示します。クラス マップとポリサーには 1 対 1 のマッピングがあり、1 つ以上の CPU キューがクラス マップにマッピングします。この後には、各 CPU キューに関連付けられている機能をリストする別のテーブル（CPU キューと関連機能）が続きます。

表 42: CoPP のシステム定義された値

クラス マップ名	ポリサー インデックス (ポリサー No.)	CPU キュー (キュー No.)
system-cpp-police-data	WK_CPP_POLICE_DATA(0)	WK_CPU_Q_ICMP_GEN(3) WK_CPU_Q_BROADCAST(12) WK_CPU_Q_ICMP_REDIRECT(6)
system-cpp-police-l2-control	WK_CPP_POLICE_L2_CONTROL(1)	WK_CPU_Q_L2_CONTROL(1)
system-cpp-police-routing-control	WK_CPP_POLICE_ROUTING_CONTROL(2)	WK_CPU_Q_ROUTING_CONTROL(4) WK_CPU_Q_LOW_LATENCY (27)
system-cpp-police-punt-webauth	WK_CPP_POLICE_PUNT_WEBAUTH(7)	WK_CPU_Q_PUNT_WEBAUTH(22)
system-cpp-police-topology-control	WK_CPP_POLICE_TOPOLOGY_CONTROL(8)	WK_CPU_Q_TOPOLOGY_CONTROL(15)
system-cpp-police-multicast	WK_CPP_POLICE_MULTICAST(9)	WK_CPU_Q_TRANSIT_TRAFFIC(18) WK_CPU_Q_MCAST_DATA(30)
system-cpp-police-sys-data	WK_CPP_POLICE_SYS_DATA(10)	WK_CPU_Q_OPENFLOW (13) WK_CPU_Q_CRYPTOPROTECT(23) WK_CPU_Q_EXCEPTION(24) WK_CPU_Q_EGR_EXCEPTION(28) WK_CPU_Q_NFL_SAMPLED_DATA(26) WK_CPU_Q_GOLD_PKT(31) WK_CPU_Q_RPF_FAILED(19)
system-cpp-police-dot1x-auth	WK_CPP_POLICE_DOT1X(11)	WK_CPU_Q_DOT1X_AUTH(0)
system-cpp-police-protocol-snooping	WK_CPP_POLICE_PR(12)	WK_CPU_Q_PROTO_SNOOPING(16)
system-cpp-police-dhcp-snooping	WK_CPP_DHCP_SNOOPING(6)	WK_CPU_Q_DHCP_SNOOPING(17)
system-cpp-police-sw-forward	WK_CPP_POLICE_SW_FWD(13)	WK_CPU_Q_SW_FORWARDING_Q(14) WK_CPU_Q_LOGGING(21) WK_CPU_Q_L2_LVX_DATA_PACK (11)
system-cpp-police-forus	WK_CPP_POLICE_FORUS(14)	WK_CPU_Q_FORUS_ADDR_RESOLUTION(5) WK_CPU_Q_FORUS_TRAFFIC(2)
system-cpp-police-multicast-end-station	WK_CPP_POLICE_MULTICAST_SNOOPING(15)	WK_CPU_Q_MCAST_END_STATION_SERVICE(20)

クラス マップ名	ポリサー インデックス (ポリサー No.)	CPU キュー (キュー No.)
system-cpp-default	WK_CPP_POLICE_DEFAULT_POLICER(16)	WK_CPU_Q_INTER_FED_TRAFFIC(7) WK_CPU_Q_EWLC_CONTROL(9) WK_CPU_Q_EWLC_DATA(10)
system-cpp-police-stackwise-virt-control	WK_CPP_STACKWISE_VIRTUAL_CONTROL(6)	WK_CPU_Q_STACKWISE_VIRTUAL_CONTROL(29)
system-cpp-police-l2lvx-control	WK_CPP_L2_LVX_CONT_PACK(4)	WK_CPU_Q_L2_LVX_CONT_PACK(8)
system-cpp-police-high-rate-app	WK_CPP_HIGH_RATE_APP(18)	WK_CPU_Q_HIGH_RATE_APP(23)
system-cpp-police-system-critical	WK_CPP_SYSTEM_CRITICAL(3)	WK_CPU_Q_SYSTEM_CRITICAL(25)

次の表に、CPU キューと、各 CPU キューに関連付けられた機能を示します。

表 43: CPU キューと関連機能

CPU キュー (キュー No.)	機能
WK_CPU_Q_DOT1X_AUTH(0)	IEEE 802.1x ポートベースの認証
WK_CPU_Q_L2_CONTROL(1)	ダイナミック トランキンング プロトコル (DTP) VLAN トランキンング プロトコル (VTP) ポート集約プロトコル (PAgP) Client Information Signalling Protocol (CISP) メッセージセッション リレー プロトコル マルチ VLAN 登録プロトコル (MVRP) Metropolitan Mobile Network (MMN) リンクレベル検出プロトコル (LLDP) 単一方向リンク検出 (UDLD) リンク集約制御プロトコル (LACP) Cisco Discovery Protocol (CDP) スパニング ツリー プロトコル (STP)

CPU キュー (キュー No.)	機能
WK_CPU_Q_FORUS_TRAFFIC(2)	Telnet、Pingv4 および Pingv6、SNMP などのホスト キープアライブ/ループバック検出 開始 - インターネット キー エクスチェンジ (IKE) プロトコル (IPSec)
WK_CPU_Q_ICMP_GEN(3)	ICMP - 接続先到達不能 ICMP - TTL 期限切れ

CPU キュー (キュー No.)	機能
WK_CPU_Q_ROUTING_CONTROL(4)	

CPU キュー (キュー No.)	機能
	Routing Information Protocol バージョン 1 (RIPv1) RIPv2 Interior Gateway Routing Protocol (IGRP) Border Gateway Protocol (BGP) PIM-UDP 仮想ルータ冗長プロトコル (VRRP) Hot Standby Router Protocol バージョン 1 (HSRPv1) HSRPv2 ゲートウェイ ロード バランシング プロトコル (GLBP) ラベル配布プロトコル (LDP) Web Cache Communication Protocol (WCCP) 次世代 Routing Information Protocol (RIPng) Open Shortest Path First (OSPF) Open Shortest Path First バージョン 3 (OSPFv3) Enhanced Interior Gateway Routing Protocol (EIGRP) Enhanced Interior Gateway Routing Protocol バージョン 6 (EIGRPv6) DHCPv6 プロトコルに依存しないマルチキャスト (PIM) Protocol Independent Multicast バージョン 6 (PIMv6) 次世代 Hot Standby Router Protocol (HSRPng) IPv6 制御 Generic Routing Encapsulation (GRE) キーペアライブ

CPU キュー (キュー No.)	機能
	ネットワークアドレス変換 (NAT) パン ト Intermediate System-to-Intermediate System (IS-IS)
WK_CPU_Q_FORUS_ADDR_RESOLUTION(5)	アドレス解決プロトコル (ARP) IPv6 ネイバーアドバタイズメントおよび ネイバー勧誘
WK_CPU_Q_ICMP_REDIRECT(6)	インターネット制御メッセージプロトコ ル (ICMP) リダイレクト
WK_CPU_Q_INTER_FED_TRAFFIC(7)	内部通信用のレイヤ2ブリッジドメイン 注入。
WK_CPU_Q_L2_LVX_CONT_PACK(8)	Exchange ID (XID) パケット
WK_CPU_Q_EWLC_CONTROL(9)	Embedded Wirelss Controller (eWLC) [ワ イヤレスアクセスポイントの制御とプロ ビジョニング (CAPWAP) (UDP 5246)]
WK_CPU_Q_EWLC_DATA(10)	eWLC データパケット (CAPWAP DATA、UDP 5247)
WK_CPU_Q_L2_LVX_DATA_PACK(11)	不明なユニキャストパケットがマップ要 求のためにパントされました。
WK_CPU_Q_BROADCAST(12)	すべてのタイプのブロードキャスト
WK_CPU_Q_OPENFLOW(13)	学習キャッシュオーバーフロー (レイヤ 2+レイヤ3)
WK_CPU_Q_CONTROLLER_PUNT(14)	データ - アクセスコントロールリスト (ACL) フル データ - IPv4 オプション データ - IPv6 ホップバイホップ データ - リソース不足/すべてをキャッチ データ - リバースパス フォワーディン グ (RPF) が不完全 収集パケット

CPU キュー (キュー No.)	機能
WK_CPU_Q_TOPOLOGY_CONTROL(15)	スパニング ツリー プロトコル (STP) Resilient Ethernet Protocol (REP) Shared Spanning Tree Protocol (SSTP)
WK_CPU_Q_PROTO_SNOOPING(16)	ダイナミック ARP インスペクション (DAI) の Address Resolution Protocol (ARP) スヌーピング
WK_CPU_Q_DHCP_SNOOPING(17)	DHCP スヌーピング
WK_CPU_Q_TRANSIT_TRAFFIC(18)	これは、ソフトウェアパスで処理する必要がある NAT によってパントされたパケットに使用されます。
WK_CPU_Q_RPF_FAILED(19)	データ - mRPF (マルチキャスト RPF) が失敗しました
WK_CPU_Q_MCAST_END_STATION_SERVICE(20)	Internet Group Management Protocol (IGMP) /Multicast Listener Discovery (MLD) 制御
WK_CPU_Q_LOGGING(21)	アクセスコントロールリスト (ACL) ロギング
WK_CPU_Q_PUNT_WEBAUTH(22)	Web 認証
WK_CPU_Q_HIGH_RATE_APP(23)	有線アプリケーションの可視性と制御 (WDAVC) トラフィック ネットワークベースのアプリケーション認識 (NBAR) トラフィック トラフィック分析および分類のための暗号化トラフィック分析 (ETA)
WK_CPU_Q_EXCEPTION(24)	IKE の表示 IP ラーニング違反 IP ポートのセキュリティ違反 IP スタティックアドレス違反 IPv6 スコープチェック リモートコピープロトコル (RCP) 例外 ユニキャスト RPF 失敗

CPU キュー (キュー No.)	機能
WK_CPU_Q_SYSTEM_CRITICAL(25)	メディアシグナリング/ワイヤレスプロキシ ARP
WK_CPU_Q_NFL_SAMPLED_DATA(26)	Netflow サンプルデータと Media Services Proxy (MSP)
WK_CPU_Q_LOW_LATENCY(27)	双方向フォワーディング検出 (BFD)、Precision Time Protocol (PTP)
WK_CPU_Q_EGR_EXCEPTION(28)	出力解決例外
WK_CPU_Q_STACKWISE_VIRTUAL_CONTROL(29)	前面スタッキングプロトコル、つまり SVL
WK_CPU_Q_MCAST_DATA(30)	データ - (S、G) の作成 データ - ローカル結合 データ - PIM 登録 データ - SPT スイッチオーバー データ - マルチキャスト
WK_CPU_Q_GOLD_PKT(31)	Gold

ユーザー設定可能な CoPP の特徴

次のタスクを実行して、コントロールプレーントラフィックを管理できます。



- (注) すべての `system-cpp-policy` コンフィギュレーションは、再起動後も保持されるように保存する必要があります。

CPU キューのポリサーの有効化と無効化

CPU キューのポリサーを有効にするには、`system-cpp-policy` ポリシーマップ内で、対応するクラスマップの下にポリサーアクション (パケット/秒) を設定します。

CPU キューのポリサーを無効にするには、`system-cpp-policy` ポリシーマップ内で、対応するクラスマップの下にポリサーアクションを削除します。



- (注) デフォルトのポリサーがすでに存在する場合は、その削除を慎重に考慮して制御します。そうしないと、システムが CPU 占有や制御パケットドロップなどのその他の異常を検出する場合があります。

ポリサーレートの変更

これは、`system-cpp-policy` ポリシーマップ内で、対応するクラスマップの下にポリサーレートアクション（パケット/秒単位）を設定することで実行できます。

ポリサーレートを設定する場合、設定したレートは最も近い200の倍数に自動的に変換されることに注意してください。たとえば、CPU キューのポリサーレートを 100 pps に設定すると、システムは 200 に変更します。または、ポリサーレートを 650 に設定すると、システムは 600 に変更します。この動作を示す出力例については、この章の「例：すべての CPU キューに対するデフォルトのポリサーレートの設定」を参照してください。

ポリサーレートをデフォルトに設定

グローバル コンフィギュレーション モードで `cpp system-default` コマンドを入力することによって、CPU キューのポリサーをデフォルト値に設定します。

ソフトウェアバージョンのアップグレードまたはダウングレード

ソフトウェアバージョンのアップグレードと CoPP

デバイスのソフトウェアバージョンをアップグレードすると、システムは CoPP に必要な更新を確認して実行します（たとえば、`system-cpp-policy` ポリシーマップを確認し、欠落している場合は作成します）。また、アップグレードアクティビティの前後に特定のタスクを完了する必要があります。これにより、設定の更新が正しく反映され、CoPP が期待どおりに動作し続けることが保証されます。ソフトウェアのアップグレードに使用する方法に応じて、アップグレード関連のタスクはオプションのシナリオまたは推奨されるシナリオもあれば、必須のシナリオもあります。

ここでは、アップグレードのシステムアクションとユーザーアクションについて説明します。また、リリース固有の警告も含まれます。

アップグレードのシステムアクション

デバイスのソフトウェアバージョンをアップグレードすると、システムは以下のアクションを実行します。これはすべてのアップグレード方法で共通です。

- アップグレード前のデバイスに `system-cpp-policy` ポリシーマップがなかった場合、アップグレード時にシステムはデフォルトポリシーを作成します。
- アップグレード前のデバイスに `system-cpp-policy` ポリシーマップがあった場合、アップグレード時にシステムはポリシーを再生成しません。

アップグレードのユーザーアクション

アップグレードのユーザーアクション（アップグレード方法に応じて）：

アップグレード方法	条件	アクション時間とアクション	目的
標準 ¹⁶	なし	アップグレード後（必須） グローバル コンフィギュレーション モードで cpp system-default コマンドを入力します。	最新のデフォルトのポリサーレートを取得します。

¹⁶ スイッチのリロードを伴うソフトウェアアップグレードの方法を指します。インストールモードまたはバンドルモードにすることができます。

ソフトウェアバージョンのダウングレードと CoPP

ダウングレードのシステムアクションとユーザーアクションについて、ここで説明します。

ダウングレードのシステムアクション

デバイスのソフトウェアバージョンをダウングレードすると、これらのアクションが実行されます。これはすべてのダウングレード方法に適用されます。

- システムは `system-cpp-policy` ポリシーマップをデバイスに保持し、コントロールプレーンにインストールします。

ダウングレードのユーザーアクション

ダウングレードのユーザーアクション：

アップグレード方法	条件	アクション時間とアクション	目的
標準 ¹⁷	なし	操作は不要です。	N/A

¹⁷ スイッチのリロードを伴うソフトウェアアップグレードの方法を指します。インストールモードまたはバンドルモードにすることができます。

ソフトウェアバージョンをダウングレードしてからアップグレードする場合、適用されるシステムアクションとユーザーアクションは、アップグレードについて説明したものと同じです。

CoPP の設定方法

CPU キューの有効化またはポリサー レートの変更

CPU キューを有効にし、CPU キューのポリサー レートを変更する手順は、同じです。手順は次のとおりです。

手順の概要

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-name*
5. **police rate** *rate* **pps**
6. **exit**
7. **control-plane**
8. **service-policy input** *policy-name*
9. **end**
10. **show policy-map control-plane**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	policy-map <i>policy-map-name</i> 例： Device(config)# policy-map system-cpp-policy Device(config-pmap)#	ポリシーマップ コンフィギュレーション モードを開始します。
ステップ 4	class <i>class-name</i> 例： Device(config-pmap)# class system-cpp-police-protocol-snooping Device(config-pmap-c)#	クラスアクション コンフィギュレーション モードを開始します。有効にする CPU キューに対応するクラスの名前を入力します。「CoPP のシステム定義値」の表を参照してください。
ステップ 5	police rate <i>rate</i> pps 例： Device(config-pmap-c)# police rate 100 pps Device(config-pmap-c-police)#	指定したトラフィッククラスに対し、1秒間に処理される着信パケット数の上限を指定します。 (注) 指定するレートは、指定したクラスマップに属するすべてのCPUキューに適用されます。
ステップ 6	exit 例：	グローバル コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
	<pre>Device (config-pmap-c-police) # exit Device (config-pmap-c) # exit Device (config-pmap) # exit Device (config) #</pre>	
ステップ 7	<p>control-plane</p> <p>例 :</p> <pre>Device (config) # control-plane Device (config-cp) #</pre>	制御プレーン (config-cp) コンフィギュレーションモードを開始します。
ステップ 8	<p>service-policy input <i>policy-name</i></p> <p>例 :</p> <pre>Device (config) # control-plane Device (config-cp) #service-policy input system-cpp-policy Device (config-cp) #</pre>	system-cpp-policy を FED にインストールします。このコマンドは、FED ポリシーを表示するために必要です。このコマンドを設定しないと、エラーになります。
ステップ 9	<p>end</p> <p>例 :</p> <pre>Device (config-cp) # end</pre>	特権 EXEC モードに戻ります。
ステップ 10	<p>show policy-map control-plane</p> <p>例 :</p> <pre>Device# show policy-map control-plane</pre>	system-cpp ポリシーの下で設定されたすべてのクラス、さまざまなトラフィックタイプに設定されたレート、および統計情報を表示します。

CPU キューの無効化

CPU キューを無効にするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **policy-map *policy-map-name***
4. **class *class-name***
5. **no police rate *rate* pps**
6. **end**
7. **show policy-map control-plane**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	policy-map policy-map-name 例： Device(config)# policy-map system-cpp-policy Device(config-pmap)#	ポリシー マップ コンフィギュレーション モードを開始します。
ステップ 4	class class-name 例： Device(config-pmap)# class system-cpp-police-protocol-snooping Device(config-pmap-c)#	クラス アクション コンフィギュレーション モードを開始します。無効にする CPU キューに対応するクラスの名前を入力します。「 <i>CoPP</i> のシステム定義値」の表を参照してください。
ステップ 5	no police rate rate pps 例： Device(config-pmap-c)# no police rate 100 pps	指定したトラフィッククラスの着信パケットの処理を無効にします。 (注) これにより、指定したクラス マップに属するすべての CPU キューが無効になります。
ステップ 6	end 例： Device(config-pmap-c)# end	特権 EXEC モードに戻ります。
ステップ 7	show policy-map control-plane 例： Device# show policy-map control-plane	system-cpp ポリシーの下で設定されたすべてのクラス、およびさまざまなトラフィックタイプと統計情報に設定されたレートを表示します。

すべての CPU キューに対するデフォルトのポリサー レートの設定

すべての CPU キューのポリサー レートをデフォルトのレートに設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **cpp system-default**
4. **end**
5. **show platform hardware fed switch {switch-number} qos que stats internal cpu policer**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	cpp system-default 例： Device(config)# cpp system-default Defaulting CPP : Policer rate for all classes will be set to their defaults	すべてのクラスのポリサー レートをデフォルトのレートに設定します。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードに戻ります。
ステップ 5	show platform hardware fed switch {switch-number} qos que stats internal cpu policer 例： Device# show platform hardware fed switch 1 qos que stat internal cpu policer	さまざまなトラフィック タイプに設定されたレートを表示します。

CoPP の設定例

例：CPU キューの有効化または CPU キューのポリサー レートの変更

次の例に、CPU キューを有効にする方法、または CPU キューのポリサー レートを変更する方法を示します。ここでは、**class system-cpp-police-protocol-snooping** CPU キューが有効になり、ポリサー レートは **2000 pps** です。

```
Device> enable
Device# configure terminal
Device(config)# policy-map system-cpp-policy
Device(config-pmap)# class system-cpp-police-protocol-snooping
Device(config-pmap-c)# police rate 2000 pps
Device(config-pmap-c-police)# end
```

```
Device# show policy-map control-plane
```

```
Control Plane
```

```
Service-policy input: system-cpp-policy
```

```
<output truncated>
```

```
Class-map: system-cpp-police-dot1x-auth (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match: none
  police:
    rate 1000 pps, burst 244 packets
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      drop
```

```
Class-map: system-cpp-police-protocol-snooping (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match: none
  police:
    rate 2000 pps, burst 488 packets
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      drop
```

```
<output truncated>
```

```
Class-map: class-default (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match: any
```

例：CPU キューの無効化

次に、CPU キューをディセーブルにする例を示します。ここでは、**class system-cpp-police-protocol-snooping** CPU キューが無効になります。

```
Device> enable
Device# configure terminal
Device(config)# policy-map system-cpp-policy
Device(config-pmap)# class system-cpp-police-protocol-snooping
Device(config-pmap-c)# no police rate 100 pps
Device(config-pmap-c)# end
```

```
Device# show running-config | begin system-cpp-policy
```

```
policy-map system-cpp-policy
class system-cpp-police-data
  police rate 200 pps
class system-cpp-police-sys-data
  police rate 100 pps
class system-cpp-police-sw-forward
  police rate 1000 pps
class system-cpp-police-multicast
  police rate 500 pps
class system-cpp-police-multicast-end-station
  police rate 2000 pps
class system-cpp-police-punt-webauth
class system-cpp-police-l2-control
class system-cpp-police-routing-control
  police rate 500 pps
class system-cpp-police-control-low-priority
class system-cpp-police-wireless-priority1
class system-cpp-police-wireless-priority2
class system-cpp-police-wireless-priority3-4-5
class system-cpp-police-topology-control
class system-cpp-police-dot1x-auth
class system-cpp-police-protocol-snooping
class system-cpp-police-forus
class system-cpp-default
```

<output truncated>

例：すべてのCPUキューに対するデフォルトのポリサーレートの設定

次に、すべてのCPUキューのポリサーレートをデフォルトに設定し、その後に設定を確認する例を示します。



- (注) 一部のCPUキューでは、すべてのクラスにデフォルトレートを設定しても、デフォルトレートと設定レートの値は同じにはなりません。これは、設定レートが最も近い200の倍数に丸められるためです。この動作は、デバイスのクロック速度によって制御されます。下の出力例では、DHCP スヌーピングと NFL SAMPLED DATA のデフォルトレートと設定レートの値にこの違いが示されています。

例：すべてのCPUキューに対するデフォルトのポリサーレートの設定

```
Device> enable
Device# configure terminal
Device(config)# cpp system-default
Defaulting CPP : Policer rate for all classes will be set to their defaults
Device(config)# end

Device# show platform hardware fed switch 1 qos queue stats internal cpu policer
```

CPU Queue Statistics

QId	PlcIdx	Queue Name	Enabled	(default) Rate	(set) Rate	Queue Drop(Bytes)	Queue
Drop(Frames)							
0	11	DOT1X Auth	Yes	1000	1000	0	0
1	1	L2 Control	Yes	2000	2000	0	0
2	14	Forus traffic	Yes	4000	4000	0	0
3	0	ICMP GEN	Yes	600	600	0	0
4	2	Routing Control	Yes	5400	5400	0	0
5	14	Forus Address resolution	Yes	4000	4000	0	0
6	0	ICMP Redirect	Yes	600	600	0	0
7	16	Inter FED Traffic	Yes	2000	2000	0	0
8	4	L2 LVX Cont Pack	Yes	1000	1000	0	0
9	16	EWLC Control	Yes	2000	2000	0	0
10	16	EWLC Data	Yes	2000	2000	0	0
11	13	L2 LVX Data Pack	Yes	1000	1000	0	0
12	0	BROADCAST	Yes	600	600	0	0
13	10	Openflow	Yes	100	200	0	0
14	13	Sw forwarding	Yes	1000	1000	0	0
15	8	Topology Control	Yes	13000	13000	0	0
16	12	Proto Snooping	Yes	2000	2000	0	0
17	6	DHCP Snooping	Yes	500	400	0	0
18	9	Transit Traffic	Yes	500	400	0	0
19	10	RPF Failed	Yes	100	200	0	0
20	15	MCAST END STATION	Yes	2000	2000	0	0
21	13	LOGGING	Yes	1000	1000	0	0
22	7	Punt Webauth	Yes	1000	1000	0	0
23	18	High Rate App	Yes	13000	13000	0	0
24	10	Exception	Yes	100	200	0	0

25	3	System Critical	Yes	1000	1000	0	0
26	10	NFL SAMPLED DATA	Yes	100	200	0	0
27	2	Low Latency	Yes	5400	5400	0	0
28	10	EGR Exception	Yes	100	200	0	0
29	5	Stackwise Virtual OOB	Yes	8000	8000	0	0
30	9	MCAST Data	Yes	500	400	0	0
31	10	Gold Pkt	Yes	100	200	0	0

* NOTE: CPU queue policer rates are configured to the closest hardware supported value

CPU Queue Policer Statistics

```

=====
Policer      Policer Accept  Policer Accept  Policer Drop  Policer Drop
Index        Bytes          Frames          Bytes          Frames
-----
0            0              0              0              0
1            0              0              0              0
2            0              0              0              0
3            0              0              0              0
4            0              0              0              0
5            0              0              0              0
6            0              0              0              0
7            0              0              0              0
8            0              0              0              0
9            0              0              0              0
10           0              0              0              0
11           0              0              0              0
12           0              0              0              0
13           0              0              0              0
14           0              0              0              0
15           0              0              0              0
16           0              0              0              0
17           0              0              0              0
18           0              0              0              0
    
```

Second Level Policer Statistics

```

=====
20           52772252      688073          0              0
21           0              0              0              0
    
```

Policer Index Mapping and Settings

```

-----
level-2      :   level-1              (default)  (set)
PlcIndex     :   PlcIndex              rate      rate
-----
20           :   1  2  8              13000     13000
21           :   0  4  7  9 10 11 12 13 14 15  6000     6000
    
```

Second Level Policer Config

```

=====
level-1 level-2              level-2
QId PlcIdx PlcIdx Queue Name          Enabled
-----
0    11     21     DOT1X Auth              Yes
1    1      20     L2 Control              Yes
    
```

例 : すべての CPU キューに対するデフォルトのポリサー レートの設定

2	14	21	Forus traffic	Yes
3	0	21	ICMP GEN	Yes
4	2	20	Routing Control	Yes
5	14	21	Forus Address resolution	Yes
6	0	21	ICMP Redirect	Yes
7	16	-	Inter FED Traffic	No
8	4	21	L2 LVX Cont Pack	Yes
9	19	-	EWLC Control	No
10	16	-	EWLC Data	No
11	13	21	L2 LVX Data Pack	Yes
12	0	21	BROADCAST	Yes
13	10	21	Openflow	Yes
14	13	21	Sw forwarding	Yes
15	8	20	Topology Control	Yes
16	12	21	Proto Snooping	Yes
17	6	-	DHCP Snooping	No
18	13	21	Transit Traffic	Yes
19	10	21	RPF Failed	Yes
20	15	21	MCAST END STATION	Yes
21	13	21	LOGGING	Yes
22	7	21	Punt Webauth	Yes
23	18	-	High Rate App	No
24	10	21	Exception	Yes
25	3	-	System Critical	No
26	10	21	NFL SAMPLED DATA	Yes
27	2	20	Low Latency	Yes
28	10	21	EGR Exception	Yes
29	5	-	Stackwise Virtual OOB	No
30	9	21	MCAST Data	Yes
31	3	-	Gold Pkt	No

CPP Classes to queue map

```

=====
PlcIdx CPP Class                               : Queues
=====
0      system-cpp-police-data                   : ICMP GEN/BROADCAST/ICMP Redirect/
10     system-cpp-police-sys-data               : Openflow/Exception/EGR Exception/NFL
      SAMPLED DATA/Gold Pkt/RPF Failed/
13     system-cpp-police-sw-forward            : Sw forwarding/LOGGING/L2 LVX Data
Pack/
9      system-cpp-police-multicast              : Transit Traffic/MCAST Data/
15     system-cpp-police-multicast-end-station : MCAST END STATION /
7      system-cpp-police-punt-webauth          : Punt Webauth/
1      system-cpp-police-l2-control            : L2 Control/
2      system-cpp-police-routing-control       : Routing Control/Low Latency/
3      system-cpp-police-system-critical       : System Critical/
4      system-cpp-police-l2lvx-control         : L2 LVX Cont Pack/
8      system-cpp-police-topology-control      : Topology Control/
11     system-cpp-police-dot1x-auth            : DOT1X Auth/
12     system-cpp-police-protocol-snooping    : Proto Snooping/
6      system-cpp-police-dhcp-snooping        : DHCP Snooping/
14     system-cpp-police-forus                 : Forus Address resolution/Forus traffic/
5      system-cpp-police-stackwise-virt-control : Stackwise Virtual OOB/
16     system-cpp-default                     : Inter FED Traffic/ EWLC Data/
18     system-cpp-police-high-rate-app        : High Rate App/
19     system-cpp-police-ewlc-control         : EWLC Control/
20     system-cpp-police-ios-routing          : L2 Control/ Topology Control/ Routing
Control/ Low Latency/
21     system-cpp-police-ios-feature           : ICMP GEN/ BROADCAST/ ICMP Redirect/
L2 LVX Cont Pack/ Proto Snooping/ Punt Webauth/ MCAST Data/ Transit Traffic/ DOT1X Auth/
Sw forwarding/ LOGGING/ L2 LVX Data Pack/ Forus traffic/ Forus Address resolution/ MCAST
END STATION / Openflow/ Exception/ EGR Exception/ NFL SAMPLED DATA/ RPF Failed/

```

CoPP のモニタリング

CPUキューのトラフィックタイプやポリサーレート（ユーザーが設定したレートやデフォルトのレート）などのポリサー設定を表示するには、次のコマンドを使用します。

コマンド	目的
show policy-map control-plane	さまざまなトラフィックタイプに設定されたレートを表示します。
show policy-map system-cpp-policy	system-cpp ポリシーの下で設定されたすべてのクラスとポリサーレートを表示します。
show platform hardware fed switch {switch-number} qos que stats internal cpu policer	さまざまなトラフィックタイプに設定されたレートを表示します。
show platform software fed {switch-number} qos policy target status	ポリシーステータスとターゲットポートタイプに関する情報を表示します。

CoPP の機能の履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
Cisco IOS XE Everest 16.5.1a	コントロールプレーンポリシー (CoPP) または CPP	<p>CoPP 機能によって、不要なトラフィックまたは DoS トラフィックから CPU を保護し、コントロールプレーンおよび管理トラフィックを優先させることにより、デバイスのセキュリティが向上します。</p> <p>この機能は、CPU キューの有効化および無効化、ポリサーレートの変更、ポリサーレートのデフォルトへの設定、およびユーザー定義のクラスマップ（フィルタ付き）を作成してポリシーマップ <code>system-cpp-policy</code> への追加を行う CLI 設定オプションを提供します。</p>

リリース	機能	機能情報
Cisco IOS XE Everest 16.6.1	CoPPのシステム定義値の変更	<p>次の新しいシステム定義のクラスが導入されました。</p> <ul style="list-style-type: none"> • system-cpp-police-stackwise-virt-control • system-cpp-police-l2lvx-control <p>次の新しいCPUキューが既存の system-cpp-default クラスに追加されました。</p> <ul style="list-style-type: none"> • WK_CPU_Q_UNUSED (7) • WK_CPU_Q_EWLC_CONTROL(9) • WK_CPU_Q_EWLC_DATA(10) <p>CPU キュー WK_CPU_Q_L2_LVX_DATA_PACK (11) がクラス system-cpp-police-sw-forward に追加されました。</p> <p>CPU キュー WK_CPU_Q_SGT_CACHE_FULL(27)は使用できなくなりました。</p>
Cisco IOS XE Everest 16.6.4	設定されているポリサーレートのシステム動作の変更。	一部のCPUキューでは、すべてのクラスにデフォルトレートを設定しても、デフォルトレートと設定レートの値は同じにはなりません。これは、設定レートが最も近い200の倍数に丸められるためです。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.8.1a	ユーザー定義のクラスマップのサポート停止、およびCoPPのシステム定義値の変更	<ul style="list-style-type: none"> • このリリース以降、ユーザー定義のクラスマップの作成はサポートされません。 • 新しいシステム定義クラス <code>system-cpp-police-dhcp-snooping</code> が導入されました。 • 新しいCPUキュー <code>WK_CPU_Q_INTER_FED_TRAFFIC</code> が既存の <code>system-cpp-default</code> クラスに追加されました。 • 次のCPUキューは使用できなくなりました。 <ul style="list-style-type: none"> • <code>WK_CPU_Q_SHOW_FORWARD</code> • <code>WK_CPU_Q_UNUSED</code> • 一部のCPUキューのデフォルトポリサーレート (pps) が変更されました。 <ul style="list-style-type: none"> • <code>WK_CPU_Q_EXCEPTION(24)</code> のデフォルトレートが 100 に変更されました。 • <code>system-cpp-default</code> の下のすべてのCPUキューのデフォルトレートが 2000 に増えました。 • <code>system-cpp-police-forus</code> の下のすべてのCPUキューのデフォルトレートが 4000 に増えました。

リリース	機能	機能情報
Cisco IOS XE Fuji 16.9.1	CoPPのシステム定義値の変更	<p>このリリース以降、18個のシステム定義クラスが <code>system-cpp-policy</code> の下に作成されます。</p> <p>次の新しいシステム定義のクラスが導入されました。</p> <ul style="list-style-type: none"> • <code>system-cpp-police-high-rate-app</code> • <code>system-cpp-police-system-critical</code> <p>CPU キュー <code>WK_CPU_Q_OPENFLOW (13)</code> がクラス <code>system-cpp-police-sys-data</code> に追加されました。</p> <p>CPU キュー <code>WK_CPU_Q_LEARNING_CACHE_OVFL(13)</code> は使用できなくなりました。</p>
Cisco IOS XE Fuji 16.9.4	システム定義のクラスマップの廃止	<p>システム定義のクラスマップ <code>system-cpp-police-control-low-priority</code> は廃止されました。</p>
Cisco IOS XE Gibraltar 16.11.1c	コントロールプレーンポリシング (CoPP) または CPP	<p>この機能は、シリーズの C9300L モデルで導入されました。</p>

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator にアクセスするには、<https://cfng.cisco.com> に進みます。



第 28 章

PKI での証明書の許可および失効の設定

- [PKI での証明書の許可および失効の設定 \(681 ページ\)](#)

PKI での証明書の許可および失効の設定

この章では、公開キーインフラストラクチャ (PKI) での証明書の許可および失効について説明します。

証明書の許可および失効に関する前提条件

PKI ストラテジの計画



ヒント 実際の証明書の展開を開始する前に、全体の PKI ストラテジを計画することを強く推奨します。

ユーザーまたはネットワーク管理者が次の作業を完了した後に、許可および失効が発生します。

- 認証局 (CA) の設定。
- ピア デバイスの CA への登録。
- ピアツーピア通信に使用される (IPsec またはセキュアソケットレイヤ (SSL) などの) プロトコルの確認および設定。

許可および失効に固有の情報をピアデバイス証明書に含めなければならない場合があるため、ピア デバイスを登録する前に、設定する許可および失効ストラテジを決定する必要があります。

高可用性

ハイアベイラビリティのため、IPsec 保護された Stream Control Transmission Protocol (SCTP) はアクティブデバイスとスタンバイデバイスの両方で設定する必要があります。同期を機能さ

せるには、SCTPを設定した後に、証明書サーバーの冗長性モードをACTIVE/STANDBYに設定する必要があります。

証明書の許可および失効に関する制約事項

- Cisco IOS XE リリースに応じて、Lightweight Directory Access Protocol (LDAP) がサポートされます。

証明書の許可および失効に関する情報

PKIの許可

PKI認証では、許可を行いません。多くの場合、一元的に管理されるソリューションが必要ですが、現在の許可用のソリューションは、設定対象のルータに固有です。

それによって証明書を特定の作業に対して許可し、その他の作業に対しては許可しない、と定義できる標準的なメカニズムはありません。アプリケーションが証明書ベースの許可情報を認識する場合、この許可情報を証明書自体に取り込みます。このソリューションでは、許可情報をリアルタイムで更新するための簡単なメカニズムを提供していないため、証明書に組み込まれた固有の許可情報を認識するように各アプリケーションに強制します。

証明書ベースのアクセスコントロールリスト (ACL) メカニズムがトラストポイント認証の一部として設定される場合、該当アプリケーションは、この許可情報を判別する役割を担うのではなく、どのアプリケーションに対して証明書を許可するのか指定できません。ルータ上の証明書ベースのACLは、大きくなりすぎて管理できないことがあります。また、外部サーバーから証明書ベースのACL指示を取得する方が有利です。

許可の問題にリアルタイムで対処する現在のソリューションでは、新しいプロトコルの指定や新しいサーバーの構築（それとともに管理およびデータ配布などの関連作業）が必要になります。

証明書ステータスのためのPKIとAAAサーバーの統合

PKIを認証、許可、アカウントिंग (AAA) サーバーと統合することにより、既存のAAAインフラストラクチャを活用する代替オンライン証明書ステータスソリューションを実現します。証明書を適切な許可レベルでAAAデータベースに一覧表示できます。PKI-AAAを明示的にサポートしないコンポーネントでは、デフォルトラベルの「all」を指定すると、AAAサーバーからの許可が可能になります。また、AAAデータベースのラベルが「none」の場合、指定された証明書が有効でないことを示します（アプリケーションラベルが欠如していることと同じですが、「none」は完全性および明確性のために含まれます）。アプリケーションコンポーネントがPKI-AAAをサポートしている場合、コンポーネントを直接指定できる場合があります。たとえば、アプリケーションコンポーネントを「ipsec」、「ssl」、または「osp」に指定できます（ipsec = IPセキュリティ、ssl = セキュアソケットレイヤ、および osp = Open Settlement Protocol）。



(注) 現在、アプリケーション ラベルの指定をサポートするアプリケーション コンポーネントはありません。

- AAA サーバーにアクセスしたときに、時間遅延が生じる場合があります。AAA サーバーを利用できない場合、許可は失敗します。

RADIUS または TACACS+ : AAA サーバー プロトコルの選択

AAA サーバーは、RADIUS または TACACS+ プロトコルと連動するように設定できます。PKI 統合用に AAA サーバーを設定する場合、許可に必要な RADIUS または TACACS 属性を設定する必要があります。

RADIUS プロトコルが使われている場合は、AAA サーバーのユーザー名に設定するパスワードを「cisco」に設定する必要があります。証明書の検証が認証を行い、AAA データベースは許可の目的だけに使用されているので、このパスワードは受け入れ可能です。TACACS プロトコルを使用する場合、TACACS では認証が不要な許可をサポートする（認証にパスワードを使用）ので、AAA サーバーのユーザー名に対して設定されるパスワードとは無関係です。

さらに、TACACS を使用する場合は、AAA サーバーに PKI サービスを追加する必要があります。カスタム属性「cert-application=all」が、PKI サービスの特定のユーザーまたはユーザーグループに追加され、特定のユーザー名が許可されます。

PKI と AAA サーバー統合用の属性値ペア

次の表に、AAA サーバーと PKI との統合を設定する場合に使用される属性値 (AV) ペアを示します (表に示す値は、可能な値であることに注意してください)。AV ペアはクライアント設定と一致する必要があります。AV ペアが一致しない場合、ピア証明書は許可されません。



(注) 場合によっては、ユーザーは、他のすべてのユーザーの AV ペアとは異なる AV ペアを持つことができます。その場合、ユーザーごとに一意のユーザー名が必要になります。(authorization username コマンド内に) all パラメータを設定すると、証明書のサブジェクト名全体を許可ユーザー名として使用するよう指定できます。

表 44: 一致する必要がある AV ペア

AV ペア	値
cisco-avpair=pki:cert-application=all	有効な値は、[all] および [none] です。

AV ペア	値
cisco-avpair=pki:cert-trustpoint=msca	<p>この値は、Cisco IOS XE コマンドライン インターフェイス (CLI) 設定のトラストポイントラベルです。</p> <p>(注) cert-trustpoint AV ペアの指定は、通常任意です。このペアが指定されている場合、デバイスクエリは、一致するラベルを持つ証明書トラストポイントから受信する必要があり、認証された証明書は、指定された証明書シリアル番号を持っている必要があります。</p>
cisco-avpair=pki:cert-serial=16318DB7000100001671	<p>この値は証明書のシリアル番号です。</p> <p>(注) cert-serial AV ペアの指定は、通常任意です。このペアが指定されている場合、シスコデバイスクエリは、一致するラベルを持つ証明書トラストポイントから受信する必要があり、認証された証明書は、指定された証明書シリアル番号を持っている必要があります。</p>
cisco-avpair=pki:cert-lifetime-end=1:00 jan 1, 2003	<p>cert-lifetime-end AV ペアは、証明書で指示された期間を越えた証明書のライフタイムを人為的に延長する場合に使用できます。cert-lifetime-end AV ペアを使用する場合は、cert-trustpoint および cert-serial AV ペアも指定する必要があります。この値は、時/分/月/日/年の形式と一致する必要があります。</p> <p>(注) 月を表す最初の 3 文字 (Jan、Feb、Mar、Apr、May、Jun、Jul、Aug、Sep、Oct、Nov、Dec) だけが使用されます。月を表す文字として 4 文字以上入力すると、残りの文字は無視されます (たとえば、Janxxxx)。</p>

CRL または OCSP サーバー：証明書失効メカニズムの選択

証明書が適切に署名された証明書として有効になった後、証明書失効方法を実行して、証明書が発行元 CA によって無効にされていないことを確認します。Cisco IOS XE ソフトウェアは、2つの失効メカニズムとして証明書失効リスト（CRL）と Online Certificate Status Protocol（OCSP）をサポートします。Cisco IOS XE ソフトウェアも、証明書のチェックのために AAA 統合をサポートしますが、これには追加の許可機能が含まれます。PKI と AAA 証明書の許可とステータス確認に関する詳細については、「証明書ステータスのための PKI と AAA サーバーの統合」を参照してください。

次の項では、各失効メカニズムの機能方法について説明します。

CRL とは

CRL とは、失効した証明書のリストです。CRL は、証明書を発行した CA によって作成され、デジタル署名されます。CRL には、各証明書の発行日と失効日が含まれています。

CA は、新しい CRL を定期的に、あるいは CA が責任を負う証明書が失効したときに公開します。デフォルトでは、現在キャッシュされている CRL が失効すると、新しい CRL がダウンロードされます。管理者は、CRL がルータのメモリにキャッシュされる時間を設定したり、CRL キャッシングを完全にディセーブルにしたりできます。CRL キャッシング設定は、トラストポイントに関連付けられたすべての CRL に適用されます。

CRL が失効すると、ルータはキャッシュから CRL を削除します。証明書が検証用に表示されると、新しい CRL がダウンロードされます。ただし、検証中の証明書を記載した新しいバージョンの CRL がサーバー上にあるにもかかわらず、ルータがキャッシュ内の CRL を使用し続ける場合、ルータは証明書が失効したことを認識しません。証明書は拒否されるはずのもので、失効チェックに合格します。

CA は、証明書を発行すると、証明書にその CRL 配布ポイント（CDP）を含めることができます。Cisco IOS クライアントデバイスは、CDP を使用して適切な CRL を見つけ、ロードします。Cisco IOS クライアントは複数の CDP をサポートしますが、Cisco IOS CA は現在 1 つの CDP しかサポートしません。ただし、サードパーティベンダー製の CA には、証明書ごとに複数の CDP または異なる CDP をサポートするものがあります。CDP が証明書に指定されていない場合、クライアントデバイスは、デフォルトの Simple Certificate Enrollment Protocol（SCEP）方式を使用して CRL を取得します（CDP の場所は、**cdp-url** コマンドを使用して指定できます）。

CRL を実装する際は、次の設計上の注意事項を考慮する必要があります。

- CRL ライフタイムとセキュリティアソシエーション（SA）およびインターネットキー交換（IKE）ライフタイム
- CRL ライフタイムにより、CA が CRL の更新を発行する時間間隔が決まります。デフォルト CRL ライフタイム値は 168 時間（1 週間）です。これは、**lifetime crl** コマンドで変更できます。
- CDP のこの方式により、CRL の取得方法が決まり、この方式として、HTTP、Lightweight Directory Access Protocol（LDAP）、SCEP、または TFTP を選択できます。最も一般的に使用されている方式は、HTTP、TFTP、および LDAP です。Cisco IOS ソフトウェアでは、

SCEP にデフォルト設定されていますが、CRL を使用して大容量のインストールを実行する場合、HTTP CDP を推奨します。HTTP では高いスケーラビリティを実現できるからです。

- CDP のこの場所は、CRL の取得先を決定します。たとえば、サーバーおよび CRL の取得先となるファイルパスを指定できます。

失効チェック中にすべての CDP を照会

CDP サーバーが要求に応答しない場合、Cisco IOS XE ソフトウェアはエラーを報告し、その結果、ピアの証明書が拒否されることがあります。証明書に複数の CDP がある場合、証明書が拒否されないようにするために、Cisco IOS XE ソフトウェアは、証明書に表示されている順序で CDP を使用しようと試みます。デバイスは、それぞれの CDP URL またはディレクトリ指定を使用して CRL を取得しようと試みます。ある CDP を使用してエラーが発生すると、次の CDP を使用して試行します。



ヒント Cisco IOS XE ソフトウェアは、指示された CDP のいずれかから CRL を取得するためにあらゆる試行を行いますが、CDP 応答の遅延によるアプリケーションのタイムアウトを避けるために、HTTP CDP サーバーを高速の冗長 HTTP サーバーと併用することを推奨します。

OCSP とは

OCSP は、証明書の有効性を判別するために使用されるオンラインのメカニズムであり、失効メカニズムとして次のような柔軟性を備えています。

- OCSP では、証明書ステータスをリアルタイムでチェックできます。
- OCSP を使用すると、ネットワーク管理者は、中央 OCSP サーバーを指定でき、これにより、ネットワーク内のすべてのデバイスにサービスを提供できます。
- また、OCSP により、ネットワーク管理者は、クライアント証明書ごと、またはクライアント証明書のグループごとに複数の OCSP サーバーを柔軟に指定できます。
- OCSP サーバーの検証は通常、ルート CA 証明書または有効な下位 CA 証明書に基づいて実行されますが、外部の CA 証明書または自己署名証明書を使用できるように設定することもできます。外部の CA 証明書または自己署名証明書を使用すると、代替の PKI 階層から OCSP サーバー証明書を発行し、有効にできます。

ネットワーク管理者は、さまざまな CA サーバーから CRL を収集し、更新するように OCSP サーバーを設定できます。ネットワーク内のデバイスは、OCSP サーバーに依存して、ピアごとに CRL を取得してキャッシュすることなく証明書ステータスをチェックできます。ピアは、証明書の失効ステータスをチェックする必要がある場合、OCSP 要求に関して疑わしい証明書のシリアル番号およびオプションの固有識別情報（ナンズ）を含む OCSP サーバーにクエリーを送信します。OCSP サーバーは、CRL のコピーを保持して、CA がその証明書を無効として記載しているかどうか判別します。次に、サーバーは、ナンズを含むピアに応答します。応答のナンズが OCSP サーバーからピアによって送信された元のナンズと一致しない場合、応答は

無効と見なされ、証明書の検証が失敗します。OCSP サーバーとピア間の対話での帯域幅の消費量は、ほとんどの場合、CRL ダウンロードより少なくなります。

OCSP サーバーが CRL を使用する場合は、CRL 時間の制約事項が適用されます。つまり、追加の証明書失効情報を含む CRL によって新しい CRL が発行されていても、まだ有効な CRL が OCSP サーバーで使用されることがあります。CRL 情報を定期的にダウンロードするデバイスが少なくなっているため、CRL ライフタイム値を小さくするか、CRL をキャッシュしないように OCSP サーバーを設定できます。詳細は、OCSP サーバーのマニュアルを参照してください。



- (注) OCSP の複数応答処理：応答パケットの OCSP レスポンダからの複数の OCSP 単一応答の処理はサポートされています。このデバッグログメッセージに加えて、次のデバッグログメッセージが表示されます。
- CRYPTO_PKI : OCSP 応答の単一応答の数 : 1 (この値は応答の数に応じて変化します)。

OCSP サーバーを使用する場合

PKI に次のいずれかの特性がある場合、CRL よりも OCSP の方が適している場合があります。

- リアルタイムの証明書失効ステータスが必要。CRL が定期的にしか更新されず、必ずしも最新の CRL がクライアント デバイスでキャッシュされていない場合があります。たとえば、最新の CRL がまだクライアントにキャッシュされておらず、また、新たに無効にされた証明書がチェック中の場合は、無効にされた証明書が失効チェックに合格します。
- 無効にされた大量の証明書または複数の CRL があります。大きな CRL をキャッシュすると、Cisco IOS メモリの大部分が消費されてしまい、他のプロセスに使用できるリソースが減少することがあります。
- CRL が頻繁に失効するため、CDP は大量の CRL を処理します。

許可または失効用に証明書ベースの ACL を使用する場合

証明書には、指定された処理の実行をデバイスまたはユーザーが許可されているかどうかの判別に使用されるフィールドがいくつか含まれています。

証明書ベース ACL はデバイス上に設定されるため、大量の ACL を十分にスケーリングしません。ただし、証明書ベースの ACL では、特定のデバイスの動作を非常に細かく制御できます。また、証明書ベース ACL は追加機能で活用され、失効、許可、またはトラストポイントなどの PKI コンポーネントを使用するタイミングを判別するのに役立ちます。証明書ベース ACL は全般的なメカニズムを提供しており、このメカニズムによりユーザーは、許可または追加処理に対して有効になっている特定の証明書または証明書のグループを選択できます。

証明書ベース ACL では、証明書内の 1 つ以上のフィールドおよび指定された各フィールドで許可される値を指定します。証明書内でチェックする必要があるフィールドと、それらのフィールドで認められる値または認められない値を指定できます。

フィールドと値との比較には、6つの論理テスト（Equal（等しい）、Notequal（等しくない）、Contains（含む）、Less than（未満）、Does not contain（含まない）、Greater than or equal（以上））を使用できます。1つの証明書ベース ACL で複数のフィールドを指定した場合、その ACL と一致するには、ACL 内のすべてのフィールド条件に合致しなければなりません。同じ ACL 内で、同じフィールドを複数回指定できます。複数の ACL を指定できます。一致するものが見つかるか、または ACL の処理がすべて完了するまで、各 ACL が順に処理されます。

証明書ベース ACL を使用した失効チェックの無視

証明書ベース ACL を設定して、有効なピアの失効チェックおよび失効した証明書を無視するようルータに指示できます。したがって、指定基準を満たす証明書は、証明書の有効期間にかかわらず受け入れることができます。また、証明書が指定基準を満たしている場合は失効チェックを実行する必要がなくなります。AAA サーバーとの通信が証明書で保護される場合にも、証明書ベース ACL を使用して失効チェックを無視できます。

失効リストの無視

トラストポイントが特定の証明書を除いて CRL を適用できるようにするには、**skip revocation-check** キーワードを指定して **match certificate** コマンドを入力します。このような適用は、スポークツースポークの直接接続も可能なハブアンドスポーク設定に最も便利です。純粋なハブアンドスポーク設定では、すべてのスポークはハブだけに接続するので、CRL チェックはハブ上だけで済みます。スポークが別のスポークと直接通信する場合、ネイバーピア証明書に対して、各スポーク上で CRL を要求する代わりに、**skip revocation-check** キーワードを指定して **match certificate** コマンドを使用できます。

失効した証明書の無視

失効した証明書を無視するようにルータを設定するには、**allow expired-certificate** キーワードを指定して **match certificate** コマンドを入力します。このコマンドには、次のような目的があります。

- このコマンドは、ピアの証明書が失効した場合にピアが新しい証明書を取得するまで、失効した証明書を「許可する」ために使用できます。
- ルータクロックがまだ正しい時間に設定されていない場合、クロックが設定されるまで、ピアの証明書はまだ有効ではないものとして表示されます。このコマンドは、ルータクロックが未設定であっても、ピアの証明書を許可する場合に使用できます。



- (注) ネットワークタイムプロトコル (NTP) が IPSec 接続だけで（通常、ハブアンドスポーク設定のハブによって）利用可能な場合は、ルータクロックを絶対に設定できません。ハブの証明書がまだ有効でないため、ハブへのトンネルを「アップ」状態にできません。
- 「失効」とは、失効している証明書またはまだ有効ではない証明書の総称です。証明書には、開始時刻と終了時刻が指定されます。ACL を目的とした、失効証明書は、ルータの現在時刻が証明書で指定された開始および終了時刻の範囲外の証明書です。

証明書の AAA チェックのスキップ

AAA サーバーとの通信が証明書で保護され、証明書の AAA チェックをスキップする場合は、**skip authorization-check** キーワードを指定して **match certificate** コマンドを使用します。たとえば、すべての AAA トラフィックがバーチャルプライベートネットワーク (VPN) トンネルを通過するように設定され、このトンネルが証明書で保護されている場合は、**skip authorization-check** キーワードを指定して **match certificate** コマンドを使用すると、証明書チェックをスキップしてトンネルを確立できます。

AAA サーバーとの PKI 統合が設定されると、**match certificate** コマンドと **skip authorization-check** キーワードを設定する必要があります。



- (注) AAA サーバーが IPSec 接続によってのみ使用可能な場合は、IPSec 接続が確立されるまで AAA サーバーとは通信できません。AAA サーバーの証明書がまだ有効でないため、IPSec 接続を「アップ」状態にできません。

PKI 証明書チェーンの検証

証明書チェーンにより、ピア証明書からルート CA 証明書までの、一連の信頼できる証明書を確立します。階層型 PKI 内では、登録されているすべてのピアが信頼できるルート CA 証明書または共通の下位 CA を共有している場合、証明書を相互に検証できます。各 CA が 1 つのラストポイントに対応します。

証明書チェーンをピアから受信すると、最初の信頼できる証明書またはラストポイントに到達するまで、証明書チェーンパスのデフォルト処理が続けられます。管理者は証明書チェーンが、すべての証明書 (下位 CA 証明書を含む) で処理されるレベルを設定できます。

証明書チェーンの処理レベルを設定すると、信頼できる証明書の再認証、信頼できる証明書チェーンの延長、および欠落のある証明書チェーンの補完が可能になります。

信頼できる証明書の再認証

このデフォルト動作でデバイスは、チェーンを検証する前に、ピアによって送信された証明書チェーンから任意の信頼できる証明書を削除します。管理者は証明書チェーンパス処理を設定して、チェーン検証の前にすでに信頼されている CA 証明書をデバイスが削除しないようにできます。そのため、チェーン内のすべての証明書は現在のセッションに対して再度認証されません。

信頼できる証明書チェーンの延長

このデフォルト動作でデバイスは、ピアによって送信された証明書チェーンに欠落している証明書がある場合、その信頼できる証明書を使用して証明書チェーンを延長します。デバイスが検証するのは、ピアによって送信されたチェーンの証明書だけです。管理者は証明書チェーンパス処理を設定して、ピアの証明書チェーンの証明書およびデバイスの信頼できる証明書を、指定したポイントに対して有効にできます。

証明書チェーンの欠落の補完

管理者は証明書チェーン処理を設定して、設定済みのトラストポイント階層に欠落がある場合、ピアによって送信された証明書を使用して証明書のセットを有効にできます。



(注) 親検証を要求するようにトラストポイントが設定され、ピアが完全な証明書チェーンを提示しない場合、欠落を補完できないため証明書チェーンは拒否され、無効になります。



(注) 親検証を要求するようにトラストポイントが設定されていて、設定済みの親トラストポイントがない場合は、設定エラーです。発生する証明書チェーンの欠落を補完できず、下位 CA 証明書を有効にできません。この証明書チェーンは無効です。

PKI で証明書の許可および失効を設定する方法

AAA サーバーとの PKI 統合の設定

ピアによって提出された証明書から AAA ユーザー名を生成し、証明書内で AAA データベース ユーザー名の作成に使用するフィールドを指定するには、次の作業を実行します。



(注) **authorization username** コマンドでサブジェクト名として **all** キーワードを使用する際に、次の制約事項を考慮する必要があります。

- 一部の AAA サーバーでは、ユーザー名の長さが制限されます (たとえば、64 文字まで)。その結果、証明書の全体のサブジェクト名は、サーバーの制約条件より長くできません。
- 一部の AAA サーバーでは、ユーザー名に使用できる文字セットが制限されます (たとえば、スペース () および等号 (=) を使用できない場合があります)。このような文字セットの制限がある AAA サーバーでは、**all** キーワードを使用できません。
- トラストポイント設定の **subject-name** コマンドは、必ずしも最終の AAA サブジェクト名とは限りません。証明書要求に完全修飾ドメイン名 (FQDN)、シリアル番号、またはルータの IP アドレスが含まれている場合は、発行された証明書のサブジェクト名フィールドにもこれらのコンポーネントが含まれます。コンポーネントをオフにするには、**fqdn**、**serial-number**、および **ip-address** の各コマンドに **none** キーワードを使用します。
- CA サーバーが証明書を発行すると、CA サーバーは、要求したサブジェクト名フィールドを変更することがあります。たとえば、一部のベンダーの CA サーバーが要求したサブジェクト名の相対識別名 (RDN) を CN、OU、O、L、ST、および C に切り替えます。ただし、別の CA サーバーは、設定した LDAP ディレクトリルート (O=cisco.com など) を要求したサブジェクト名の最後に追加する場合があります。
- 証明書の表示用に選択するツールによっては、サブジェクト名の RDN の印刷順序が異なることがあります。Cisco IOS ソフトウェアでは、重要度が最低の RDN を先頭に表示しますが、Open Source Secure Socket Layer (OpenSSL) などの、他のソフトウェアでは、重要度が最高の RDN を先頭に表示します。したがって、完全な識別名 (DN) (サブジェクト名) を持つ AAA サーバーを対応するユーザー名として設定する場合は、Cisco IOS ソフトウェアスタイル (つまり、重要度が最低の RDN を先頭に表示) が使用されていることを確認してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authorization network listname [method]**
5. **crypto pki trustpoint name**
6. **enrollment [mode] [retry period minutes] [retry count number] url url [pem]**
7. **revocation-check method**
8. **exit**
9. **authorization username subjectname subjectname**
10. **authorization list listname**
11. **tacacs server server-name**
12. **address {ipv4 | ipv6} ip-address**
13. **key string**

14. end

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	aaa new-model 例： Device(config)# aaa new-model	AAA アクセスコントロールモデルをイネーブルにします。
ステップ 4	aaa authorization network listname [method] 例： Device(config)# aaa authorization network maxaaa group tacacs+	ネットワークへのユーザー アクセスを制限するパラメータを設定します。 • method : group radius 、 group tacacs+ 、または group group-name のいずれか。
ステップ 5	crypto pki trustpoint name 例： Device(config)# crypto pki trustpoint msca	トラストポイントおよび設定された名前を宣言して、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 6	enrollment [mode] [retry period minutes] [retry count number] url url [pem] 例： Device(ca-trustpoint)# enrollment url http://caserver.myexample.com または Device(ca-trustpoint)# enrollment url http://[2001:DB8:1:1::1]:80	CA の次の登録パラメータを指定します。 • (任意) CA システムが登録局 (RA) を提供する場合、 mode キーワードとして RA モードを指定します。デフォルトでは、RA モードは無効です。 • (任意) retry period キーワードおよび <i>minutes</i> 引数は、CA に別の証明書要求を送信するまでルータが待機する期間を分単位で指定します。有効値は 1 ~ 60 です。デフォルトは 1 です。 • (任意) retry count キーワードおよび <i>number</i> 引数は、直前の要求に対する応答をルータが受信しない場合、ルータが証明書要求を再送信する回数を指定します。有効な値は、1 ~ 100 です。デフォルトは 10 です。 • <i>url</i> 引数は、ルータが証明書要求を送信する CA の URL です。

	コマンドまたはアクション	目的
		<p>(注) IPv6 アドレスは http: 登録方式に追加できます。たとえば、 http://[ipv6-address]:80 です。URL 内の IPv6 アドレスは括弧で囲む必要があります。</p> <ul style="list-style-type: none"> (任意) pem キーワードは、証明書要求にプライベート強化メール (PEM) の境界を追加します。
ステップ 7	revocation-check method 例 : Device(ca-trustpoint)# revocation-check crl	(任意) 証明書の失効ステータスをチェックします。
ステップ 8	exit 例 : Device(ca-trustpoint)# exit	CA トラストポイントコンフィギュレーションモードを終了し、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 9	authorization username subjectname subjectname 例 : Device(config)# authorization username subjectname serialnumber	AAA ユーザー名の構築に使用する異なる証明書フィールドのパラメータを設定します。 <i>subjectname</i> 引数には、次のいずれかを指定できます。 <ul style="list-style-type: none"> all : 証明書の識別名 (サブジェクト名) 全体。 commonname : 証明書の共通名。 country : 証明書の国。 email : 証明書の電子メール。 ipaddress : 証明書の IP アドレス。 locality : 証明書の地域。 organization : 証明書の組織。 organizationalunit : 証明書の組織単位。 postalcode : 証明書の郵便番号。 serialnumber : 証明書のシリアル番号。 state : 証明書の州フィールド。 streetaddress : 証明書の住所。 title : 証明書のタイトル。

■ トラブルシューティングのヒント

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • unstructuredname : 証明書の非公式名。
ステップ 10	authorization list <i>listname</i> 例 : Device(config)# authorization list maxaaa	AAA 認可リストを指定します。
ステップ 11	tacacs server <i>server-name</i> 例 : Device(config)# tacacs server yourserver	TACACS+ サーバーを指定します。
ステップ 12	address {ipv4 ipv6} <i>ip-address</i> 例 : Device(config-server-tacacs)# address ipv4 192.0.2.2	TACACS サーバーの IP アドレスを設定します。
ステップ 13	key string 例 : Device(config-server-tacacs)# key a_secret_key	スイッチと TACACS サーバーとの間で使用される許可および暗号キーを設定します。
ステップ 14	end 例 : Device(config-server-tacacs)# end 例 :	特権 EXEC モードに戻ります。

■ トラブルシューティングのヒント

CA とルータ間のインタラクションのトレース (メッセージタイプ) に関するデバッグメッセージを表示するには、**debug crypto pki transactions** コマンドを使用します (サンプル出力を参照してください。ここでは、AAA サーバー交換との成功した PKI 統合、および AAA サーバー交換との失敗した PKI 統合を示します)。

成功した交換

```
Device# debug crypto pki transactions
```

```
Apr 22 23:15:03.695: CRYPTO_PKI: Found a issuer match
Apr 22 23:15:03.955: CRYPTO_PKI: cert revocation status unknown.
Apr 22 23:15:03.955: CRYPTO_PKI: Certificate validated without revocation check
```

「CRYPTO_PKI_AAA」と表示されている各行は、AAA 認可チェックの状態を示します。各 AAA AV ペアが示され、認可チェックの結果が表示されます。

```
Apr 22 23:15:04.019: CRYPTO_PKI_AAA: checking AAA authorization (ipsecca_script_aalist,
PKIAAA-L, <all>)
Apr 22 23:15:04.503: CRYPTO_PKI_AAA: reply attribute ("cert-application" = "all")
Apr 22 23:15:04.503: CRYPTO_PKI_AAA: reply attribute ("cert-trustpoint" = "CA1")
```

```
Apr 22 23:15:04.503: CRYPTO_PKI_AAA: reply attribute ("cert-serial" = "15DE")
Apr 22 23:15:04.503: CRYPTO_PKI_AAA: authorization passed
Apr 22 23:12:30.327: CRYPTO_PKI: Found a issuer match
```

失敗した交換

```
Device# debug crypto pki transactions
```

```
Apr 22 23:11:13.703: CRYPTO_PKI_AAA: checking AAA authorization =
Apr 22 23:11:14.203: CRYPTO_PKI_AAA: reply attribute ("cert-application" = "all")
Apr 22 23:11:14.203: CRYPTO_PKI_AAA: reply attribute ("cert-trustpoint" = "CA1")
Apr 22 23:11:14.203: CRYPTO_PKI_AAA: reply attribute ("cert-serial" = "233D")
Apr 22 23:11:14.203: CRYPTO_PKI_AAA: parsed cert-lifetime-end as: 21:30:00
Apr 22 23:11:14.203: CRYPTO_PKI_AAA: timezone specific extended
Apr 22 23:11:14.203: CRYPTO_PKI_AAA: cert-lifetime-end is expired
Apr 22 23:11:14.203: CRYPTO_PKI_AAA: cert-lifetime-end check failed.
Apr 22 23:11:14.203: CRYPTO_PKI_AAA: authorization failed
```

上記の失敗した交換では、証明書が失効しています。

PKI 証明書ステータス チェックの失効メカニズムの設定

証明書失効メカニズム（CRL または OCSP）として CRL を設定し、PKI の証明書のステータスをチェックするには、次の作業を実行します。

revocation-check コマンド

revocation-check コマンドを使用し、ピアの証明書が無効にされていないことを確認するための方式（OCSP、CRL、または失効チェックのスキップ）を少なくとも 1 つ指定します。複数の方式を指定する場合、方式を適用する順序は、このコマンドで指定した順序になります。

デバイスに適用可能な CRL がなく、いずれの CRL も取得できない場合、または OCSP サーバーがエラーを返す場合、設定に **none** キーワードを含めない限り、デバイスはピアの証明書を拒否します。**none** キーワードを設定した場合、失効チェックは実行されず、証明書は常に受け入れられます。

OCSP サーバーとのナンスおよびピア通信

OCSP を使用すると、OCSP サーバーとのピア通信時に、OCSP 要求に関するナンス（固有識別情報）がデフォルトで送信されます。ナンスを使用することにより、ピアと OCSP サーバー間にセキュアで信頼性の高い通信チャンネルが確立されます。

OCSP サーバーがナンスをサポートしていない場合は、ナンスの送信をディセーブルにできます。詳細については、OCSP サーバーのマニュアルを参照してください。

始める前に

- クライアント証明書を発行する前に、サーバーで適切な設定（CDP の設定など）を行う必要があります。
- OCSP サーバーから CA サーバーの失効ステータスを返すように設定するときは、CA サーバーが発行した OCSP 応答署名証明書を OCSP サーバーに設定する必要があります。署名証明書が正しいフォーマットであることを確認してください。署名証明書のフォーマット

が正しくない場合、ルータは、OCSP 応答を受理しません。詳細については、OCSP のマニュアルを参照してください。



- (注)
- OCSP は、HTTP を使用してメッセージを転送するので、OCSP サーバーにアクセスする際に遅延が発生する場合があります。
 - OCSP サーバーが、失効ステータスのチェックを通常の CRL 処理に依存している場合、CRL の遅延は OCSP にも適用されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint *name***
4. **ocsp url *url***
5. **revocation-check *method1* [*method2 method3*]**
6. **ocsp disable-nonce**
7. **end**
8. **show crypto pki certificates**
9. **show crypto pki trustpoints [*status* | *label* [*status*]]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例 : Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します (要求された場合)。
ステップ 2	configure terminal 例 : Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	crypto pki trustpoint <i>name</i> 例 : Device(config)# crypto pki trustpoint hazel	トラストポイントおよび設定された名前を宣言して、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 4	ocsp url <i>url</i> 例 : Device(ca-trustpoint)# ocsp url http://ocsp-server	<i>url</i> 引数は、トラストポイントが証明書ステータスをチェックできるように OCSP サーバーの URL を指定します。この URL は、証明書の AIA 拡張部に指定されている OCSP サーバーの URL (存在する場合) を上書きします。設定したトラストポイントに

	コマンドまたはアクション	目的
	または <pre>Device(ca-trustpoint)# oosp url http://10.10.10.1:80</pre> または <pre>Device(ca-trustpoint)# oosp url http://[2001DB8:1:1::2]:80</pre>	関連するすべての証明書は、OCSP サーバーによって確認されます。使用可能な URL は、ホスト名、IPv4 アドレス、または IPv6 アドレスです。
ステップ 5	revocation-check method1 [method2 method3] 例： <pre>Device(ca-trustpoint)# revocation-check oosp none</pre>	証明書の失効ステータスをチェックします。 <ul style="list-style-type: none"> • crl : CRL によって証明書をチェックします。これがデフォルトのオプションです。 • none : 証明書のチェックを無視します。 • oosp : OCSP サーバーによって証明書をチェックします。 2 番目と 3 番目の方法を指定した場合、各方法はその直前の方法でエラーが返された場合（サーバーがダウンしている場合など）にだけ使用されます。
ステップ 6	oosp disable-nonce 例： <pre>Device(ca-trustpoint)# oosp disable-nonce</pre>	（任意）OCSP サーバーとピアが通信するときに、ナンズ（OCSP 要求に関する固有識別情報）が送信されないように指定します。
ステップ 7	end 例： <pre>Device(ca-trustpoint)# end</pre>	CA トラストポイント コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 8	show crypto pki certificates 例： <pre>Device# show crypto pki certificates</pre>	（任意）証明書に関する情報を表示します。
ステップ 9	show crypto pki trustpoints [status label [status]] 例： <pre>Device# show crypto pki trustpoints</pre>	ルータに設定されているトラストポイントに関する情報を表示します。

証明書の許可および失効の設定

証明書ベース ACL の指定、失効チェックまたは失効した証明書の無視、手動によるデフォルトの CDP の場所の上書き、手動による OCSP サーバー設定の上書き、CRL キャッシングの設

失効チェックを無視するように証明書ベース ACL を設定

定、あるいは証明書シリアル番号に基づくセッションの受理/拒否の設定を行うには、必要に応じて次の作業を実行します。

失効チェックを無視するように証明書ベース ACL を設定

証明書ベース ACL を使用して、失効チェックおよび失効証明書を無視するようにルータを設定するには、次の手順を実行します。

- 既存のトラストポイントの識別またはピアの証明書の検証に使用される新しいトラストポイントを作成します。トラストポイントがまだ認証されていない場合は、認証してください。必要に応じて、ルータをこのトラストポイントに登録できます。**match certificate** コマンドと **skip revocation-check** キーワードを使用する場合は、トラストポイントにオプションの CRL を設定しないでください。
- 証明書自体の CRL をチェックする必要がない証明書の固有の特性と、許可する必要がある失効証明書の固有の特性を判別します。
- 前のステップで確認した特性と一致する証明書マップを定義します。
- 最初の手順で作成または指定したトラストポイントに、**match certificate** コマンドと **skip revocation-check** キーワード、**match certificate command** と **allow expired-certificate** キーワードを追加できます。



(注) 証明書マップは、ピアの公開キーがキャッシュされている場合でも確認されます。たとえば、ピアによって公開キーがキャッシュされており、証明書マップがトラストポイントに追加されて証明書が禁止されると、証明書マップが有効になります。これにより、過去に一度接続され、現在は禁止されている証明書を持つクライアントが再接続することを防ぎます。

証明書内の CDP の手動による上書き

ユーザーは、手動で設定した CDP で証明書内の CDP を上書きできます。証明書の CDP の手動による上書きは、特定のサーバーが長時間利用できない場合に便利です。元の CDP を含む証明書のすべてを再発行しなくても、証明書の CDP を URL またはディレクトリ指定に置き換えることができます。

手動による証明書の OCSP サーバー設定の上書き

管理者はクライアント証明書の Authority Information Access (AIA) フィールドに指定された、または **ocsp url** コマンドを発行して設定された OCSP サーバーの設定値を上書きできます。**match certificate override ocsp** コマンドを使用すると、1つまたは複数の OCSP サーバーをクライアント証明書ごとに、またはクライアント証明書のグループごとに手動で指定できます。失効チェック時にクライアント証明書が証明書マップに正常に照合された場合、**match certificate override ocsp** コマンドを発行すると、クライアント証明書 AIA フィールドまたは **ocsp url** コマンド設定が上書きされます。



(注) 1つのクライアント証明書には、OCSP サーバーを1つだけ指定できます。

CRL キャッシュコントロールの設定

デフォルトでは、現在キャッシュされている CRL が失効すると、新しい CRL がダウンロードされます。管理者は、**crl cache delete-after** コマンドを発行して、CRL がキャッシュに保持される最大時間（分単位）を設定するか、**crl cache none** コマンドを発行して CRL キャッシュを無効にできます。**crl-cache delete-after** コマンドまたは **crl-cache none** コマンドのみを指定できます。トラストポイントに両方のコマンドを入力した場合は、後に実行されたコマンドが有効になり、メッセージが表示されます。

crl-cache none コマンドまたは **crl-cache delete-after** コマンドのいずれを実行しても現在キャッシュされている CRL に影響はありません。**crl-cache none** コマンドを設定した場合、このコマンドを発行すると、ダウンロードされたすべての CRL はキャッシュされません。**crl-cache delete-after** コマンドを設定した場合、このコマンドの発行後に設定されたライフタイムだけがダウンロードされた CRL に影響します。

この機能は、CA が失効日を指定せずに CRL を発行する場合、あるいは失効日が数日後または数週間後に迫っている場合に役立ちます。

証明書のシリアル番号セッションコントロールの設定

証明書検証要求がセッションのトラストポイントによって受け入れられる、または拒否されるように証明書シリアル番号を指定できます。証明書のシリアル番号セッションコントロールによっては、証明書がまだ有効であっても、セッションが拒否される場合があります。証明書のシリアル番号セッションコントロールは、**serial-number** フィールドを持つ証明書マップまたは AAA 属性のいずれかを使用して **cert-serial-not** コマンドで設定できます。

セッションコントロールに証明書マップを使用すると、管理者は、1つの証明書シリアル番号を指定できます。AAA 属性を使用すると、管理者は、セッションコントロールに証明書シリアル番号を指定できます。

始める前に

- 証明書マップをトラストポイントに関連付ける前に、トラストポイントを定義し、認証する必要があります。
- CDP オーバライド機能を有効にする、または **serial-number** コマンドを発行する前に、証明書マップを設定する必要があります。
- PKI と AAA サーバーとの統合は、「証明書ステータスのための PKI と AAA サーバーの統合」の説明のとおり AAA 属性を使用して正常に完了する必要があります。

手順の概要

1. **enable**
2. **configure terminal**

3. **crypto pki certificate map** *label sequence-number*
4. *field-name match-criteria match-value*
5. **exit**
6. **crypto pki trustpoint** *name*
7. 次のいずれかを実行します。
 - **crl-cache** **none**
 - **crl-cache delete-after** *time*
8. **match certificate** *certificate-map-label* [**allow expired-certificate** | **skip revocation-check** | **skip authorization-check**]
9. **match certificate** *certificate-map-label* **override cdp** {*url* | **directory**} *string*
10. **match certificate** *certificate-map-label* **override oosp** [**trustpoint** *trustpoint-label*] *sequence-number url oosp-url*
11. **exit**
12. **aaa new-model**
13. **aaa attribute list** *list-name*
14. **attribute type** {*name*} {*value*}
15. **exit**
16. **show crypto pki certificates**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	crypto pki certificate map <i>label sequence-number</i> 例： Device(config)# crypto pki certificate map Group 10	証明書において、一致する必要がある値または一致する必要がない値を定義し、CA 証明書マップ 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	<i>field-name match-criteria match-value</i> 例： Device(ca-certificate-map)# subject-name co MyExample	1 つまたは複数の証明書フィールドと、これらのフィールドの一致基準および照合する値を指定します。 <i>field-name</i> には、次のいずれかの名前文字列（大文字と小文字を区別しない）または日付を指定します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • alt-subject-name • expires-on • issuer-name • name • serial-number • subject-name • unstructured-subject-name • valid-start <p>(注) 日付フィールドのフォーマットは、dd mm yyyy hh:mm:ss または mmm dd yyyy hh:mm:ss です。</p> <p><i>match-criteria</i> には、次の論理演算子のいずれかを指定します。</p> <ul style="list-style-type: none"> • co : 含む (名前およびシリアル番号フィールドでのみ有効) • eq : 等しい (名前、シリアル番号、および日付フィールドで有効) • ge : 以上 (日付フィールドでのみ有効) • lt : 未満 (日付フィールドでのみ有効) • nc : 含まない (名前およびシリアル番号フィールドでのみ有効) • ne : 等しくない (名前、シリアル番号、および日付フィールドで有効) <p><i>match-value</i> は、<i>match-criteria</i> で割り当てられた論理演算子を使用してテストする名前または日付です。</p> <p>(注) このコマンドは、証明書ベース ACL を設定する場合にだけ使用し、失効チェックまたは失効した証明書を無視するように証明書ベース ACL を設定する場合には使用しないでください。</p>

	コマンドまたはアクション	目的
ステップ 5	exit 例 : <pre>Device(ca-certificate-map)# exit</pre>	ca-certificate-map コンフィギュレーションモードを終了して、グローバルコンフィギュレーションモードに戻ります。
ステップ 6	crypto pki trustpoint name 例 : <pre>Device(config)# crypto pki trustpoint Access2</pre>	トラストポイントおよび設定された名前を宣言して、CA トラストポイントコンフィギュレーションモードを開始します。
ステップ 7	次のいずれかを実行します。 <ul style="list-style-type: none"> • crl-cache none • crl-cache delete-after time 例 : <pre>Device(ca-trustpoint)# crl-cache none</pre> 例 : <pre>Device(ca-trustpoint)# crl-cache delete-after 20</pre>	(任意) トラストポイントに関連付けられたすべての CRL の CRL キャッシングを完全にディセーブルにします。 crl-cache none コマンドを実行しても、現在キャッシュされている CRL に影響はありません。このコマンドが設定された後にダウンロードされるすべての CRL は、キャッシュされません。 (任意) トラストポイントに関連付けられたすべての CRL に関して、CRL がキャッシュに保持される最大時間を指定します。 <ul style="list-style-type: none"> • time : CRL が削除されるまでの時間 (分単位)。 crl-cache delete-after コマンドを実行しても、現在キャッシュされている CRL に影響はありません。設定されたライフタイムは、このコマンドが設定された後にダウンロードされた CRL だけに影響します。
ステップ 8	match certificate certificate-map-label [allow expired-certificate skip revocation-check skip authorization-check] 例 : <pre>Device(ca-trustpoint)# match certificate Group skip revocation-check</pre>	(任意) 証明書ベース ACL (crypto pki certificate map コマンドによって定義されている) をトラストポイントに関連付けます。 <ul style="list-style-type: none"> • certificate-map-label : crypto pki certificate map コマンドを使用して指定した label 引数と一致する必要があります。 • allowexpired-certificate : 失効した証明書を無視します。 • skip revocation-check : トラストポイントが、特定の証明書を除く CRL を適用できるようにします。

	コマンドまたはアクション	目的
ステップ 9	<p>match certificate <i>certificate-map-label</i> override cdp {url directory} <i>string</i></p> <p>例 :</p> <pre>Device(ca-trustpoint)# match certificate Group1 override cdp url http://server.cisco.com</pre>	<ul style="list-style-type: none"> • skip authorization-check : AAA サーバーとの PKI 統合を設定すると、証明書の AAA チェックをスキップします。 <p>(任意) URL またはディレクトリが指定された証明書の、既存の CDP エントリを手動で上書きします。</p> <ul style="list-style-type: none"> • <i>certificate-map-label</i> : ユーザー指定のラベル。事前に定義された crypto pki certificate map コマンドに指定した <i>label</i> 引数と一致する必要があります。 • url : 証明書の CDP が HTTP または LDAP URL で上書きされるように指定します。 • directory : 証明書の CDP が LDAP ディレクトリ指定で上書きされるように指定します。 • <i>string</i> : URL またはディレクトリ指定。 <p>(注) 一部のアプリケーションは、すべての CDP が試行される前にタイムアウトすることがあり、エラーメッセージで報告します。エラーメッセージはルータに影響を及ぼしません。また、Cisco IOS ソフトウェアは、すべての CDP が試行されるまで CRL の取得を続行します。</p>
ステップ 10	<p>match certificate <i>certificate-map-label</i> override ocsp [trustpoint <i>trustpoint-label</i>] <i>sequence-number</i> url <i>ocsp-url</i></p> <p>例 :</p> <pre>Device(ca-trustpoint)# match certificate mycertmapname override ocsp trustpoint mytp 15 url http://192.0.2.2</pre>	<p>(任意) OCSP サーバーをクライアント証明書ごとに、またはクライアント証明書のグループごとに指定し、複数回発行して、追加の OCSP サーバーおよびクライアント証明書の設定（代替の PKI 階層を含む）を指定できます。</p> <ul style="list-style-type: none"> • <i>certificate-map-label</i> : 既存の証明書マップ名。 • trustpoint : OCSP サーバー証明書を検証するときに使用されるトラストポイント。 • <i>sequence-number</i> : match certificate override ocsp コマンド文を検証対象の証明書に適用する順序。照合が最低のシーケンス番号から最高のシーケンス番号に実行されます。同じシーケンス番号で複数のコマンドを発行すると、前の

	コマンドまたはアクション	目的
		<p>OCSPサーバーオーバーライド設定が上書きされます。</p> <ul style="list-style-type: none"> • url : OCSPサーバーのURL。 <p>証明書が設定された証明書マップと一致すると、クライアント証明書のAIAフィールドおよび以前に発行されたocsp urlコマンド設定値は、指定されたOCSPサーバーで上書きされます。</p> <p>マップベースの一致が発生しない場合、引き続き次の2つのケースがクライアント証明書に適用されます。</p> <ul style="list-style-type: none"> • OCSPを失効方法として指定すると、AIAフィールド値がクライアント証明書に引き続き適用されます。 • ocsp url設定が存在する場合は、ocsp url設定が引き続きクライアント証明書に適用されます。
ステップ 11	<p>exit</p> <p>例 :</p> <pre>Device(ca-trustpoint)# exit</pre>	<p>グローバル コンフィギュレーション モードに戻ります。</p>
ステップ 12	<p>aaa new-model</p> <p>例 :</p> <pre>Device(config)# aaa new-model</pre>	<p>(任意) AAA アクセス コントロール モデルをイネーブルにします。</p>
ステップ 13	<p>aaa attribute list <i>list-name</i></p> <p>例 :</p> <pre>Device(config)# aaa attribute list crl</pre>	<p>(任意) ルータにローカルでAAA属性リストを定義し、config-attr-list コンフィギュレーション モードを開始します。</p>
ステップ 14	<p>attribute type <i>{name}</i> <i>{value}</i></p> <p>例 :</p> <pre>Device(config-attr-list)# attribute type cert-serial-not 6C4A</pre>	<p>(任意) ルータのAAA属性リストにローカルに追加されるAAA属性タイプを定義します。</p> <p>証明書のシリアル番号セッションコントロールを設定するために、管理者は、<i>value</i> フィールドの特定の証明書を、<i>name</i> が cert-serial-not に設定されているシリアル番号に基づき受け入れるか、拒否するか指定できます。証明書のシリアル番号が属性タイプ設定で指定されたシリアル番号と一致した場合、証明書は拒否されます。</p>

	コマンドまたはアクション	目的
		使用可能な AAA 属性タイプのリストを表示するには、 show aaa attributes コマンドを実行してください。
ステップ 15	exit 例 : Device(ca-trustpoint)# end 例 : Device(config-attr-list)# end	特権 EXEC モードに戻ります。
ステップ 16	show crypto pki certificates 例 : Device# show crypto pki certificates	(任意) CA 証明書が認証されたら、ルータにインストールされた証明書のコンポーネントを表示します。

例

次に、サンプル証明書を示します。OCSP 関連の拡張子は感嘆符を使用して示されません。

```

Certificate:
  Data:
    Version: v3
    Serial Number:0x14
    Signature Algorithm:SHAwithRSA - 1.2.840.113549.1.1.4
    Issuer:CN=CA server,OU=PKI,O=Cisco Systems
    Validity:
      Not Before:Thursday, August 8, 2002 4:38:05 PM PST
      Not After:Tuesday, August 7, 2003 4:38:05 PM PST
    Subject:CN=OCSP server,OU=PKI,O=Cisco Systems
    Subject Public Key Info:
      Algorithm:RSA - 1.2.840.113549.1.1.1
      Public Key:
        Exponent:65537
        Public Key Modulus:(2048 bits) :
          <snip>
    Extensions:
      Identifier:Subject Key Identifier - 2.5.29.14
        Critical:no
        Key Identifier:
          <snip>
      Identifier:Authority Key Identifier - 2.5.29.35
        Critical:no
        Key Identifier:
          <snip>
      Identifier:OCSP NoCheck:- 1.3.6.1.5.5.7.48.1.5
        Critical:no
      Identifier:Extended Key Usage:- 2.5.29.37
        Critical:no
        Extended Key Usage:

```

トラブルシューティングのヒント

```

                                OCSPSigning
!
                                Identifier:CRL Distribution Points - 2.5.29.31
                                Critical:no
                                Number of Points:1
                                Point 0
                                Distribution Point:
[URIName:ldap://CA-server/CN=CA server,OU=PKI,O=Cisco Systems]
Signature:
    Algorithm:SHAwithRSA - 1.2.840.113549.1.1.4
Signature:
<snip>

```

次の例は、既存のシーケンスの先頭に **match certificate override ocsp** コマンドを追加したときの実行コンフィギュレーション出力の抜粋を示します。

```

match certificate map3 override ocsp 5 url http://192.0.2.3/
show running-configuration
.
.
.
                                match certificate map3 override ocsp 5 url http://192.0.2.3/
                                match certificate map1 override ocsp 10 url http://192.0.2.1/
                                match certificate map2 override ocsp 15 url http://192.0.2.2/

```

次の例は、既存の **match certificate override ocsp** コマンドが置き換えられ、トラストポイントが代替の PKI 階層を使用するように指定された場合の、実行コンフィギュレーション出力の抜粋を示します。

```

match certificate map4 override ocsp trustpoint tp4 10 url http://192.0.2.4/newvalue
show running-configuration
.
.
.
                                match certificate map3 override ocsp trustpoint tp3 5 url http://192.0.2.3/
                                match certificate map1 override ocsp trustpoint tp1 10 url http://192.0.2.1/
                                match certificate map4 override ocsp trustpoint tp4 10 url
http://192.0.2.4/newvalue
                                match certificate map2 override ocsp trustpoint tp2 15 url http://192.0.2.2/

```

トラブルシューティングのヒント

失効チェックまたは失効した証明書を無視した場合は、慎重に設定を確認する必要があります。証明書マップが、当該の証明書または許可する証明書、あるいはスキップする AAA チェックのいずれかと適切に一致していることを確認してください。管理された環境で、証明書マップを変更して想定どおりに機能していないものを判別します。

証明書チェーンの設定

ピア証明書の証明書チェーンパスに処理レベルを設定するには、次の作業を実行します。

始める前に

- デバイスを PKI 階層に登録する必要があります。
- 適切なキー ペアを証明書に関連付ける必要があります。



- (注)
 - ルート CA に関連付けられたトラストポイントは、次のレベルに対して有効になるように設定できません。

chain-validation コマンドは、ルート CA に関連付けられたトラストポイント用に **continue** キーワードを指定して設定します。エラーメッセージが表示され、チェーン検証はデフォルトの **chain-validation** コマンド設定に戻ります。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto pki trustpointname**
4. **chain-validation** [{stop | continue} [parent-trustpoint]]
5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードを有効にします。 <ul style="list-style-type: none">パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	crypto pki trustpointname 例： Device(config)# crypto pki trustpoint ca-sub1	トラストポイントおよび設定された名前を宣言して、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 4	chain-validation [{stop continue} [parent-trustpoint]] 例： Device(ca-trustpoint)# chain-validation continue ca-sub1	証明書チェーンが、すべての証明書（下位 CA 証明書を含む）で処理されるレベルを設定します。 <ul style="list-style-type: none">stop キーワードを使用して、証明書がすでに信頼できることを明示します。これがデフォルトの設定です。continue キーワードを使用して、トラストポイントに関連付けられた下位 CA 証明書を有効にする必要があることを明示します。

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • <i>parent-trustpoint</i> 引数は、証明書を照合する必要がある親トラストポイント名を指定します。
ステップ 5	exit 例： Device(ca-trustpoint)# exit	CA トラストポイント コンフィギュレーションモードを終了し、グローバル コンフィギュレーションモードに戻ります。

PKIにおける証明書の許可および失効の設定例

PKI AAA 許可の設定および確認の例

ここでは、PKI AAA 認可の設定例を示します。

例：デバイス設定

次の **show running-config** コマンド出力は、AAA サーバー機能との PKI 統合を使用して、VPN 接続を許可するように設定されたデバイスの動作設定を示します。

```
Device#show running-config

Building configuration...
!
version 16.8
!
hostname catxxxx
!
aaa new-model
!
!
aaa authentication login default group tacacs+
aaa authentication login no_tacacs enable
aaa authentication ppp default group tacacs+
aaa authorization exec ACSLab group tacacs+
aaa authorization network ACSLab group tacacs+
aaa accounting exec ACSLab start-stop group tacacs+
aaa accounting network default start-stop group ACSLab
aaa session-id common
!
ip domain name example.com
!
crypto pki trustpoint EM-CERT-SERV
  enrollment url http://192.0.2.33:80
  serial-number
  crl optional
  rsakeypair STOREVPN 2048
  auto-enroll
  authorization list ACSLab
!
crypto pki certificate chain EM-CERT-SERV
  certificate 04
    30820214 3082017D A0030201 02020104 300D0609 2A864886 F70D0101 04050030
    17311530 13060355 0403130C 454D2D43 4552542D 53455256 301E170D 30343031
```

```

31393232 30323535 5A170D30 35303131 38323230 3235355A 3030312E 300E0603
55040513 07314437 45424434 301C0609 2A864886 F70D0109 02160F37 3230302D
312E6772 696C2E63 6F6D3081 9F300D06 092A8648 86F70D01 01010500 03818D00
30818902 818100BD F3B837AA D925F391 2B64DA14 9C2EA031 5A7203C4 92F8D6A8
7D2357A6 BCC8596F A38A9B10 47435626 D59A8F2A 123195BB BE5A1E74 B1AA5AE0
5CA162FF 8C3ACA4F B3EE9F27 8B031642 B618AE1B 40F2E3B4 F996BEFE 382C7283
3792A369 236F8561 8748AA3F BC41F012 B859BD9C DB4F75EE 3CEE2829 704BD68F
FD904043 0F555702 03010001 A3573055 30250603 551D1F04 1E301C30 1AA018A0
16861468 7474703A 2F2F3633 2E323437 2E313037 2E393330 0B060355 1D0F0404
030205A0 301F0603 551D2304 18301680 1420FC4B CF0B1C56 F5BD4C06 0AFD4E67
341AE612 D1300D06 092A8648 86F70D01 01040500 03818100 79E97018 FB955108
12F42A56 2A6384BC AC8E22FE F1D6187F DA5D6737 C0E241AC AAAEC75D 3C743F59
08DEEFF2 0E813A73 D79E0FA9 D62DC20D 8E2798CD 2C1DC3EC 3B2505A1 3897330C
15A60D5A 8A13F06D 51043D37 E56E45DF A65F43D7 4E836093 9689784D C45FD61D
EC1F160C 1ABC8D03 49FB11B1 DA0BED6C 463E1090 F34C59E4
quit
certificate ca 01
30820207 30820170 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
17311530 13060355 0403130C 454D2D43 4552542D 53455256 301E170D 30333132
31363231 34373432 5A170D30 36313231 35323134 3734325A 30173115 30130603
55040313 0C454D2D 43455254 2D534552 5630819F 300D0609 2A864886 F70D0101
01050003 818D0030 81890281 8100C14D 833641CF D784F516 DA6B50C0 7B3CB3C9
589223AB 99A7DC14 04F74EF2 AAEEE8F5 E3BFAE97 F2F980F7 D889E6A1 2C726C69
54A29870 7E7363FF 3CD1F991 F5A37CFF 3FFDD3D0 9E486C44 A2E34595 C2D078BB
E9DE981E B733B868 AA8916C0 A8048607 D34B83C0 64BDC101 161FC103 13C06500
22D6EE75 7D6CF133 7F1B515F 32830203 010001A3 63306130 0F060355 1D130101
FF040530 030101FF 300E0603 551D0F01 01FF0404 03020186 301D0603 551D0E04
16041420 FC4BCF0B 1C56F5BD 4C060AFD 4E67341A E612D130 1F060355 1D230418
30168014 20FC4BCF 0B1C56F5 BD4C060A FD4E6734 1AE612D1 300D0609 2A864886
F70D0101 04050003 81810085 D2E386F5 4107116B AD3AC990 CBE84063 5FB2A6B5
BD572026 528E92ED 02F3A0AE 1803F2AE AA4C0ED2 0F59F18D 7B50264F 30442C41
0AF19C4E 70BD3CB5 0ADD8DE8 8EF636BD 24410DF4 DB62DAFC 67DA6E58 3879AA3E
12AFB1C3 2E27CB27 EC74E1FC AEE2F5CF AA80B439 615AA8D5 6D6DEDC3 7F9C2C79
3963E363 F2989FB9 795BA8
quit
!
!
crypto isakmp policy 10
  encr aes
  group 14
!
!
crypto ipsec transform-set ISC_TS_1 esp-aes esp-sha-hmac
!
crypto ipsec profile ISC_IPSEC_PROFILE_2
  set security-association lifetime kilobytes 53000000
  set security-association lifetime seconds 14400
  set transform-set ISC_TS_1
!
!
controller ISA 1/1
!
!
interface Tunnel0
  description MGRE Interface provisioned by ISC
  bandwidth 10000
  ip address 192.0.2.172 255.255.255.0
  no ip redirects
  ip mtu 1408
  ip nhrp map multicast dynamic
  ip nhrp network-id 101
  ip nhrp holdtime 500
  ip nhrp server-only
  no ip split-horizon eigrp 101

```

例：成功した PKI AAA 許可のデバッグ

```

tunnel source FastEthernet2/1
tunnel mode gre multipoint
tunnel key 101
tunnel protection ipsec profile ISC_IPSEC_PROFILE_2
!
interface FastEthernet2/0
 ip address 192.0.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet2/1
 ip address 192.0.2.2 255.255.255.0
 duplex auto
 speed auto
!
!
end

```

例：成功した PKI AAA 許可のデバッグ

次の **show debugging** コマンド出力は、AAA サーバー機能との PKI 統合を使用して、成功した許可を示します。

```

Device#show debugging

General OS:
  TACACS access control debugging is on
  AAA Authentication debugging is on
  AAA Authorization debugging is on
Cryptographic Subsystem:
Crypto PKI Trans debugging is on
Device#
May 28 19:36:11.117: CRYPTO_PKI: Trust-Point EM-CERT-SERV picked up
May 28 19:36:12.789: CRYPTO_PKI: Found a issuer match
May 28 19:36:12.805: CRYPTO_PKI: cert revocation status unknown.
May 28 19:36:12.805: CRYPTO_PKI: Certificate validated without revocation check
May 28 19:36:12.813: CRYPTO_PKI_AAA: checking AAA authorization (ACSLab, POD5.example.com,
<all>)
May 28 19:36:12.813: AAA/BIND(00000042): Bind i/f
May 28 19:36:12.813: AAA/AUTHOR (0x42): Pick method list 'ACSLab'
May 28 19:36:12.813: TPLUS: Queuing AAA Authorization request 66 for processing
May 28 19:36:12.813: TPLUS: processing authorization request id 66
May 28 19:36:12.813: TPLUS: Protocol set to None .....Skipping
May 28 19:36:12.813: TPLUS: Sending AV service=pki
May 28 19:36:12.813: TPLUS: Authorization request created for 66(POD5.example.com)
May 28 19:36:12.813: TPLUS: Using server 192.0.2.55
May 28 19:36:12.813: TPLUS(00000042)/0/NB_WAIT/203A4628: Started 5 sec timeout
May 28 19:36:12.813: TPLUS(00000042)/0/NB_WAIT: wrote entire 46 bytes request
May 28 19:36:12.813: TPLUS: Would block while reading pak header
May 28 19:36:12.817: TPLUS(00000042)/0/READ: read entire 12 header bytes (expect 27
bytes)
May 28 19:36:12.817: TPLUS(00000042)/0/READ: read entire 39 bytes response
May 28 19:36:12.817: TPLUS(00000042)/0/203A4628: Processing the reply packet
May 28 19:36:12.817: TPLUS: Processed AV cert-application=all
May 28 19:36:12.817: TPLUS: received authorization response for 66: PASS
May 28 19:36:12.817: CRYPTO_PKI_AAA: reply attribute ("cert-application" = "all")
May 28 19:36:12.817: CRYPTO_PKI_AAA: authorization passed
Device#
May 28 19:36:18.681: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 101: Neighbor 192.0.2.171 (Tunnel0)
is up: new adjacency
Device#
Device# show crypto isakmp sa

```

```
dst          src          state          conn-id slot
192.0.2.22  192.0.2.102  QM_IDLE       84      0
```

例：失敗した PKI AAA 許可のデバッグ

次の **show debugging** コマンド出力は、デバイスが、VPN を使用しての接続を許可されていないことを示します。このメッセージは、このような状況で表示される典型的なメッセージです。

この例においてピアユーザー名は、Cisco Secure ACS の VPN_Disabled と呼ばれる Cisco Secure ACS グループに移動することにより、許可されていないものとして設定されました。デバイス (device9.example.com) は、任意のピアに VPN 接続を確立する前に、Cisco Secure ACS AAA サーバーを確認するように設定されています。

```
Device#show debugging
```

```
General OS:
```

```
TACACS access control debugging is on
AAA Authentication debugging is on
AAA Authorization debugging is on
```

```
Cryptographic Subsystem:
```

```
Crypto PKI Trans debugging is on
```

```
Device#
```

```
May 28 19:48:29.837: CRYPTO_PKI: Trust-Point EM-CERT-SERV picked up
May 28 19:48:31.509: CRYPTO_PKI: Found a issuer match
May 28 19:48:31.525: CRYPTO_PKI: cert revocation status unknown.
May 28 19:48:31.525: CRYPTO_PKI: Certificate validated without revocation check
May 28 19:48:31.533: CRYPTO_PKI_AAA: checking AAA authorization (ACSLab, POD5.example.com,
<all>)
May 28 19:48:31.533: AAA/BIND(00000044): Bind i/f
May 28 19:48:31.533: AAA/AUTHOR (0x44): Pick method list 'ACSLab'
May 28 19:48:31.533: TPLUS: Queuing AAA Authorization request 68 for processing
May 28 19:48:31.533: TPLUS: processing authorization request id 68
May 28 19:48:31.533: TPLUS: Protocol set to None .....Skipping
May 28 19:48:31.533: TPLUS: Sending AV service=pki
May 28 19:48:31.533: TPLUS: Authorization request created for 68(POD5.example.com)
May 28 19:48:31.533: TPLUS: Using server 192.0.2.55
May 28 19:48:31.533: TPLUS(00000044)/0/NE_WAIT/203A4C50: Started 5 sec timeout
May 28 19:48:31.533: TPLUS(00000044)/0/NE_WAIT: wrote entire 46 bytes request
May 28 19:48:31.533: TPLUS: Would block while reading pak header
May 28 19:48:31.537: TPLUS(00000044)/0/READ: read entire 12 header bytes (expect 6 bytes)
May 28 19:48:31.537: TPLUS(00000044)/0/READ: read entire 18 bytes response
May 28 19:48:31.537: TPLUS(00000044)/0/203A4C50: Processing the reply packet
May 28 19:48:31.537: TPLUS: received authorization response for 68: FAIL
May 28 19:48:31.537: CRYPTO_PKI_AAA: authorization declined by AAA, or AAA server not
found.
May 28 19:48:31.537: CRYPTO_PKI_AAA: No cert-application attribute found. Failing.
May 28 19:48:31.537: CRYPTO_PKI_AAA: authorization failed
May 28 19:48:31.537: CRYPTO_PKI: AAA authorization for list 'ACSLab', and user
'POD5.example.com' failed.
May 28 19:48:31.537: %CRYPTO-5-IKMP_INVALID_CERT: Certificate received from 192.0.2.162
is bad: certificate invalid
May 28 19:48:39.821: CRYPTO_PKI: Trust-Point EM-CERT-SERV picked up
May 28 19:48:41.481: CRYPTO_PKI: Found a issuer match
May 28 19:48:41.501: CRYPTO_PKI: cert revocation status unknown.
May 28 19:48:41.501: CRYPTO_PKI: Certificate validated without revocation check
May 28 19:48:41.505: CRYPTO_PKI_AAA: checking AAA authorization (ACSLab, POD5.example.com,
<all>)
```

例：失効メカニズムの設定

```

May 28 19:48:41.505: AAA/BIND(00000045): Bind i/f
May 28 19:48:41.505: AAA/AUTHOR (0x45): Pick method list 'ACSLab'
May 28 19:48:41.505: TPLUS: Queuing AAA Authorization request 69 for processing
May 28 19:48:41.505: TPLUS: processing authorization request id 69
May 28 19:48:41.505: TPLUS: Protocol set to None .....Skipping
May 28 19:48:41.505: TPLUS: Sending AV service=pki
May 28 19:48:41.505: TPLUS: Authorization request created for 69(POD5.example.com)
May 28 19:48:41.505: TPLUS: Using server 198.168.244.55
May 28 19:48:41.509: TPLUS(00000045)/0/IDLE/63B22834: got immediate connect on new 0
May 28 19:48:41.509: TPLUS(00000045)/0/WRITE/63B22834: Started 5 sec timeout
May 28 19:48:41.509: TPLUS(00000045)/0/WRITE: wrote entire 46 bytes request
May 28 19:48:41.509: TPLUS(00000045)/0/READ: read entire 12 header bytes (expect 6 bytes)
May 28 19:48:41.509: TPLUS(00000045)/0/READ: read entire 18 bytes response
May 28 19:48:41.509: TPLUS(00000045)/0/63B22834: Processing the reply packet
May 28 19:48:41.509: TPLUS: received authorization response for 69: FAIL
May 28 19:48:41.509: CRYPTO_PKI_AAA: authorization declined by AAA, or AAA server not found.
May 28 19:48:41.509: CRYPTO_PKI_AAA: No cert-application attribute found. Failing.
May 28 19:48:41.509: CRYPTO_PKI_AAA: authorization failed
May 28 19:48:41.509: CRYPTO_PKI: AAA authorization for list 'ACSLab', and user 'POD5.example.com' failed.
May 28 19:48:41.509: %CRYPTO-5-IKMP_INVAL_CERT: Certificate received from 192.0.2.162 is bad: certificate invalid
Device#
Device# show crypto iskmp sa

```

dst	src	state	conn-id	slot
192.0.2.2	192.0.2.102	MM_KEY_EXCH	95	0

例：失効メカニズムの設定

ここでは、PKIの失効メカニズムを指定する際に使用できる設定例を示します。

例：OCSPサーバーの設定

次の例では、証明書のAIA拡張部で指定されたOCSPサーバーを使用するようにルータを設定する方法を示します。

```

Device> enable
Device# configure terminal
Device(config)#crypto pki trustpoint mytp
Device(ca-trustpoint)# revocation-check ocsp
Device(ca-trustpoint)# end

```

例：CRLの指定後のOCSPサーバーの指定

次の例では、CRLをCDPからダウンロードするようにルータを設定する方法を示します。CRLを利用できない場合は、証明書のAIA拡張部で指定されるOCSPサーバーが使用されます。両方のオプションが失敗した場合、証明書の検証も失敗します。

```

Device> enable
Device# configure terminal
Device(config)#crypto pki trustpoint mytp
Device(ca-trustpoint)#revocation-check crl ocsp
Device(ca-trustpoint)# end

```

例：OCSP サーバーの指定

以下に、HTTP URL 「http://myocspserver:81」にある OCSP サーバーを使用するようにルータを設定する例を示します。このサーバーがダウンしている場合は、失効チェックは行われません。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint mytp
Device(ca-trustpoint)# ocsf url http://myocspserver:81
Device(ca-trustpoint)# revocation-check ocsf none
Device(ca-trustpoint)# end
```

例：OCSP サーバーとの通信でのナンスの無効化

次の例は、OCSP 要求に関するナンス（固有識別情報）が、OCSP サーバーとの通信でディセーブルになっている場合の通信を示します。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint mytp
Device(ca-trustpoint)# ocsf url http://myocspserver:81
Device(ca-trustpoint)# revocation-check ocsf none
Device(ca-trustpoint)# ocsf disable-nonce
Device(ca-trustpoint)# end
```

例：セントラルサイトにあるハブデバイスを証明書失効チェック用に設定

次の例では、複数のブランチオフィスにセントラルサイトへの接続を提供しているセントラルサイトにあるハブデバイスを示します。

ブランチ オフィスも追加の IPSec トンネルを使用して、ブランチ オフィス間で直接相互に通信できます。

CA は、セントラルサイトにある HTTP サーバーの CRL を公開します。セントラルサイトは、各ピアと IPSec トンネルを設定する場合、そのピアの CRL をチェックします。

次の例では、IPSec 設定を示しません。PKI 関連の設定だけを示します。

ホーム オフィスのハブ設定

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint VPN-GW
Device(ca-trustpoint)# enrollment url http://ca.home-office.com:80/certsrv/mscep/mscep.dll
Device(ca-trustpoint)# serial-number none
Device(ca-trustpoint)# fqdn none
Device(ca-trustpoint)# ip-address none
Device(ca-trustpoint)# subject-name o=Home Office Inc,cn=Central VPN Gateway
Device(ca-trustpoint)# revocation-check crl
Device(ca-trustpoint)# end
```

例：セントラルサイトにあるハブデバイスを証明書失効チェック用に設定

セントラルサイトのハブデバイス

```
Device# show crypto ca certificate

Certificate
  Status: Available
  Certificate Serial Number: 2F62BE14000000000CA0
  Certificate Usage: General Purpose
  Issuer:
    cn=Central Certificate Authority
    o=Home Office Inc
  Subject:
    Name: Central VPN Gateway
    cn=Central VPN Gateway
    o=Home Office Inc
  CRL Distribution Points:
    http://ca.home-office.com/CertEnroll/home-office.crl
  Validity Date:
    start date: 00:43:26 GMT Sep 26 2003
    end   date: 00:53:26 GMT Sep 26 2004
    renew date: 00:00:00 GMT Jan 1 1970
  Associated Trustpoints: VPN-GW
CA Certificate
  Status: Available
  Certificate Serial Number: 1244325DE0369880465F977A18F61CA8
  Certificate Usage: Signature
  Issuer:
    cn=Central Certificate Authority
    o=Home Office Inc
  Subject:
    cn=Central Certificate Authority
    o=Home Office Inc
  CRL Distribution Points:
    http://ca.home-office.com/CertEnroll/home-office.crl
  Validity Date:
    start date: 22:19:29 GMT Oct 31 2002
    end   date: 22:27:27 GMT Oct 31 2017
  Associated Trustpoints: VPN-GW
```

ブランチオフィスデバイスのトラストポイント

```
Device> enable
Device# configure terminal
Device(ca-trustpoint)# crypto pki trustpoint home-office
Device(ca-trustpoint)# enrollment url http://ca.home-office.com:80/certsrv/mscep/mscep.dll
Device(ca-trustpoint)# serial-number none
Device(ca-trustpoint)# fqdn none
Device(ca-trustpoint)# ip-address none
Device(ca-trustpoint)# subject-name o=Home Office Inc,cn=Branch 1
Device(ca-trustpoint)# revocation-check crl
Device(ca-trustpoint)# end
```

証明書マップがブランチオフィスデバイスに入力されます。

```
branch1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
branch1(config)# crypto pki certificate map central-site 10
branch1(ca-certificate-map)# end
```


セントラルサイトのハブデバイス上で発行された **show certificate** コマンドの出力では、証明書が以下によって発行されたことを示しています。

```
cn=Central Certificate Authority
o=Home Office Inc
```

この2行は、行を区切るためのカンマ (,) を使用して1行に結合され、元の2行が最初の一致基準として追加されています。

```
Device(ca-certificate-map)# issuer-name co cn=Central Certificate Authority, ou=Home
Office Inc
!The above line wrapped but should be shown on one line with the line above it.
```

セントラルサイトデバイスの証明書のサブジェクト名についても、同じように組み合わせられています (「Name:」で始まる行は、サブジェクト名の一部ではなく、証明書マップ基準を作成する際に無視する必要があることに注意してください)。これが証明書マップで使用されるサブジェクト名です。

```
cn=Central VPN Gateway
```

```
o=Home Office Inc
```

```
Device(ca-certificate-map)# subject-name eq cn=central vpn gateway, o=home office inc
```

これで、以前に設定された証明書マップがトラストポイントに追加されます。

```
Device> enable
Device# configure terminal
Device(ca-certificate-map)# crypto pki trustpoint home-office
Device(ca-trustpoint)# match certificate central-site skip revocation-check
Device(ca-trustpoint)# end
```

設定がチェックされます (大部分の設定は示されていません)。

```
Device# write term

!Many lines left out
.
.
.
crypto pki trustpoint home-office
  enrollment url http://ca.home-office.com:80/certsrv/mscep/mscep.dll
  serial-number none
  fqdn none
  ip-address none
  subject-name o=Home Office Inc,cn=Branch 1
  revocation-check crl
  match certificate central-site skip revocation-check
!
!
crypto pki certificate map central-site 10
  issuer-name co cn = Central Certificate Authority, ou = Home Office Inc
  subject-name eq cn = central vpn gateway, o = home office inc
!many lines left out
```

今後のピアの証明書との照合のために、発行者名の行とサブジェクト名の行が矛盾しないように再フォーマットされていることに注意してください。

例：セントラルサイトにあるハブデバイスを証明書失効チェック用に設定

ブランチオフィスが AAA をチェックする場合は、トラストポイントには次のような行があります。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint home-office
Device(ca-trustpoint)# authorization list allow_list
Device(ca-trustpoint)# authorization username subjectname commonname
Device(ca-trustpoint)# end
```

証明書マップが上記のように定義されると、次のコマンドがトラストポイントに追加され、セントラルサイトハブの AAA チェックがスキップされます。

```
Device(ca-trustpoint)# match certificate central-site skip authorization-check
```

両方のケースにおいてブランチサイトデバイスは、CRL のチェックまたは AAA サーバーと通信するために、セントラルサイトに IPSec トンネルを確立する必要があります。ただし、**match certificate** コマンドと **central-site skip authorization-check (argument and keyword)** を使用しないと、ブランチオフィスが CRL または AAA サーバーを確認するまで、トンネルを確立することはできません (**match certificate** コマンドと **central-site skip authorization-check** 引数およびキーワードを使用しない限り、トンネルは確立されません)。

ブランチサイトにあるデバイスの証明書が失効していて、その証明書を更新するためにセントラルサイトにトンネルを確立する必要がある場合、セントラルサイトで **match certificate** コマンドと **allow expired-certificate** キーワードを使用できます。

セントラルサイトデバイスのトラストポイント

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint VPN-GW
Device(ca-trustpoint)# enrollment url http://ca.home-office.com:80/certsrv/mscep/mscep.dll
Device(ca-trustpoint)# serial-number none
Device(ca-trustpoint)# fqdn none
Device(ca-trustpoint)# ip-address none
Device(ca-trustpoint)# subject-name o=Home Office Inc,cn=Central VPN Gateway
Device(ca-trustpoint)# revocation-check crl
Device(ca-trustpoint)# end
```

ブランチ 1 サイトデバイスのトラストポイント

```
Device# show crypto ca certificate

Certificate
  Status: Available
  Certificate Serial Number: 2F62BE14000000000CA0
  Certificate Usage: General Purpose
  Issuer:
    cn=Central Certificate Authority
    o=Home Office Inc
  Subject:
    Name: Branch 1 Site
    cn=Branch 1 Site
    o=Home Office Inc
  CRL Distribution Points:
```

```

    http://ca.home-office.com/CertEnroll/home-office.crl
Validity Date:
    start date: 00:43:26 GMT Sep 26 2003
    end   date: 00:53:26 GMT Oct 3 2003
    renew date: 00:00:00 GMT Jan 1 1970
Associated Trustpoints: home-office
CA Certificate
Status: Available
Certificate Serial Number: 1244325DE0369880465F977A18F61CA8
Certificate Usage: Signature
Issuer:
    cn=Central Certificate Authority
    o=Home Office Inc
Subject:
    cn=Central Certificate Authority
    o=Home Office Inc
CRL Distribution Points:
    http://ca.home-office.com/CertEnroll/home-office.crl
Validity Date:
    start date: 22:19:29 GMT Oct 31 2002
    end   date: 22:27:27 GMT Oct 31 2017
Associated Trustpoints: home-office

```

証明書マップがセントラルサイトデバイスに入力されます。

```

Device> enable
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# crypto pki certificate map branch1 10
Device(ca-certificate-map)# issuer-name co cn=Central Certificate Authority, ou=Home
Office Inc
!The above line wrapped but should be part of the line above it.
Device(ca-certificate-map)# subject-name eq cn=Brahcn 1 Site,o=home office inc
Device(ca-certificate-map)# end

```

証明書マップがトラストポイントに追加されます。

```

Device> enable
Device# configure terminal
Device(ca-certificate-map)# crypto pki trustpoint VPN-GW
Device(ca-trustpoint)# match certificate branch1 allow expired-certificate
Device(ca-trustpoint)# exit
Device (config) #exit

```

設定がチェックされます (設定の大部分は示されていません)。

```

Device# write term

!many lines left out
crypto pki trustpoint VPN-GW
enrollment url http://ca.home-office.com:80/certsrv/mscep/mscep.dll
serial-number none
fqdn none
ip-address none
subject-name o=Home Office Inc,cn=Central VPN Gateway
revocation-check crl
match certificate branch1 allow expired-certificate
!
!
crypto pki certificate map central-site 10
issuer-name co cn = Central Certificate Authority, ou = Home Office Inc
subject-name eq cn = central vpn gateway, o = home office inc
! many lines left out

```

match certificate コマンド、**branch1 allow expired-certificate** (引数とキーワード) および証明書マップは、ブランチデバイスが新しい証明書を取得した後すぐに削除する必要があります。

例：証明書の許可および失効の設定

この項では、CRL キャッシュ コントロールの設定または証明書のシリアル番号セッション コントロールを指定する場合に使用する設定例を示します。

例：CRL キャッシュコントロールの設定

次の例では、CA1 トラストポイントに関連付けられたすべての CRL の CRL キャッシングをディセーブルにする方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint CA1
Device(ca-trustpoint)# enrollment url http://CA1:80
Device(ca-trustpoint)# ip-address FastEthernet0/0
Device(ca-trustpoint)# crl query ldap://ldap_CA1
Device(ca-trustpoint)# revocation-check crl
Device(ca-trustpoint)# crl cache none
Device(ca-trustpoint)# end
```

上記の例の設定を実行した直後は、まだ現在の CRL がキャッシュされています。

Device# show crypto pki crls

```
CRL Issuer Name:
  cn=name Cert Manager,ou=pki,o=example.com,c=US
  LastUpdate: 18:57:42 GMT Nov 26 2005
  NextUpdate: 22:57:42 GMT Nov 26 2005
  Retrieved from CRL Distribution Point:
    ldap://ldap.example.com/CN=name Cert Manager,O=example.com
```

現在の CRL が失効すると、次の更新時に新しい CRL がルータにダウンロードされます。**crl-cache none** コマンドが有効になり、トラストポイントの CRL はすべてキャッシュされなくなります。また、キャッシュは無効になります。**show crypto pki crls** コマンドを実行して、CRL がキャッシュされていないことを確認できます。キャッシュされている CRL がないため、出力は表示されません。

次の例では、CA1 トラストポイントに関連付けられたすべての CRL に 2 分の最大ライフタイムを設定する方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint CA1
Device(ca-trustpoint)# enrollment url http://CA1:80
Device(ca-trustpoint)# ip-address FastEthernet 0/0
Device(ca-trustpoint)# crl query ldap://ldap_CA1
Device(ca-trustpoint)# revocation-check crl
Device(ca-trustpoint)# crl cache delete-after 2
Device(ca-trustpoint)# end
```

CRL の最大ライフタイムを設定するために上記例の設定を実行した直後でも、依然現在の CRL がキャッシュされます。

Device# show crypto pki crls

```
CRL Issuer Name:
cn=name Cert Manager,ou=pki,o=example.com,c=US
LastUpdate: 18:57:42 GMT Nov 26 2005
NextUpdate: 22:57:42 GMT Nov 26 2005
Retrieved from CRL Distribution Point:
ldap://ldap.example.com/CN=name Cert Manager,O=example.com
```

When the current CRL expires, a new CRL is downloaded to the router at the next update and the **crl-cache delete-after** command takes effect. This newly cached CRL and all subsequent CRLs will be deleted after a maximum lifetime of 2 minutes.

You can verify that the CRL will be cached for 2 minutes by executing the **show crypto pki crls** command. Note that the NextUpdate time is 2 minutes after the LastUpdate time.

Device# show crypto pki crls

```
CRL Issuer Name:
cn=name Cert Manager,ou=pki,o=example.com,c=US
LastUpdate: 22:57:42 GMT Nov 26 2005

NextUpdate: 22:59:42 GMT Nov 26 2005
Retrieved from CRL Distribution Point:

ldap://ldap.example.com/CN=name Cert Manager,O=example.com
```

例：証明書のシリアル番号セッションコントロールの設定

次の例では、CA1 トラストポイントの証明書マップを使用した証明書のシリアル番号セッションコントロールの設定を示します。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint CA1
Device(ca-trustpoint)# enrollment url http://CA1
Device(ca-trustpoint)# chain-validation stop
Device(ca-trustpoint)# crl query ldap://ldap_server
Device(ca-trustpoint)# revocation-check crl
Device(ca-trustpoint)# match certificate crl
Device(ca-trustpoint)# exit
Device(config)# crypto pki certificate map crl 10
Device(ca-certificate-map)# serial-number co 279d
Device(ca-certificate-map)# end
```



- (注) *match-criteria* 値が **co** (含む) ではなく **eq** (等しい) に設定されている場合、シリアル番号はスペースを含めて、証明書マップのシリアル番号に正確に一致する必要があります。

次の例では、AAA 属性を使用した証明書のシリアル番号セッションコントロールの設定を示します。この場合、証明書にシリアル番号「4ACA」がなければ、有効な証明書はすべて受け入れられません。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint CA1
Device(ca-trustpoint)# enrollment url http://CA1
Device(ca-trustpoint)# ip-address FastEthernet0/0
Device(ca-trustpoint)# crl query ldap://ldap_CA1
```

例：証明書チェーン検証の設定

```

Device(ca-trustpoint)# revocation-check crl
Device(ca-trustpoint)# exit
Device(config)# aaa new-model
Device(config)# aaa attribute list crl
Device(config-attr-list)# attribute-type aaa-cert-serial-not 4ACA
Device(config-attr-list)# end

```

サーバーログは、シリアル番号「4ACA」を持つ証明書が拒否されたことを示しています。証明書の拒否は、感嘆符で表示されます。

```

.
.
.
Dec 3 04:24:39.051: CRYPTO_PKI: Trust-Point CA1 picked up
Dec 3 04:24:39.051: CRYPTO_PKI: locked trustpoint CA1, refcount is 1
Dec 3 04:24:39.051: CRYPTO_PKI: unlocked trustpoint CA1, refcount is 0
Dec 3 04:24:39.051: CRYPTO_PKI: locked trustpoint CA1, refcount is 1
Dec 3 04:24:39.135: CRYPTO_PKI: validation path has 1 certs
Dec 3 04:24:39.135: CRYPTO_PKI: Found a issuer match
Dec 3 04:24:39.135: CRYPTO_PKI: Using CA1 to validate certificate
Dec 3 04:24:39.135: CRYPTO_PKI: Certificate validated without revocation check
Dec 3 04:24:39.135: CRYPTO_PKI: Selected AAA username: 'PKIAAA'
Dec 3 04:24:39.135: CRYPTO_PKI: Anticipate checking AAA list:'CRL'
Dec 3 04:24:39.135: CRYPTO_PKI_AAA: checking AAA authorization (CRL, PKIAAA-L1, <all>)
Dec 3 04:24:39.135: CRYPTO_PKI_AAA: pre-authorization chain validation status (0x4)
Dec 3 04:24:39.135: AAA/BIND(00000021): Bind i/f
Dec 3 04:24:39.135: AAA/AUTHOR (0x21): Pick method list 'CRL'
.
.
.
Dec 3 04:24:39.175: CRYPTO_PKI_AAA: reply attribute ("cert-application" = "all")
Dec 3 04:24:39.175: CRYPTO_PKI_AAA: reply attribute ("cert-trustpoint" = "CA1")
!
Dec 3 04:24:39.175: CRYPTO_PKI_AAA: reply attribute ("cert-serial-not" = "4ACA")
Dec 3 04:24:39.175: CRYPTO_PKI_AAA: cert-serial doesn't match ("4ACA" != "4ACA")
!
Dec 3 04:24:39.175: CRYPTO_PKI_AAA: post-authorization chain validation status (0x7)
!
Dec 3 04:24:39.175: CRYPTO_PKI: AAA authorization for list 'CRL', and user 'PKIAAA'
failed.
Dec 3 04:24:39.175: CRYPTO_PKI: chain cert was anchored to trustpoint CA1, and chain
validation result was:
  CRYPTO_PKI_CERT_NOT_AUTHORIZED
!
Dec 3 04:24:39.175: %CRYPTO-5-IKMP_INVALID_CERT: Certificate received from 192.0.2.43 is
bad: certificate invalid
Dec 3 04:24:39.175: %CRYPTO-6-IKMP_MODE_FAILURE: Processing of Main mode failed with
peer at 192.0.2.43
.
.
.

```

例：証明書チェーン検証の設定

この項では、デバイス証明書の証明書チェーン処理レベルを指定する場合に使用する設定例を示します。

ピアからルート CA への証明書チェーン検証の設定

次の設定例では、ピア、SubCA11、SubCA1、および RootCA のすべての証明書が検証されます。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint RootCA
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# chain-validation stop
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# rsakeypair RootCA
Device(ca-trustpoint)# exit
Device(config)# crypto pki trustpoint SubCA1
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# chain-validation continue RootCA
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# rsakeypair SubCA1
Device(ca-trustpoint)# exit
Device(config)# crypto pki trustpoint SubCA11
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# chain-validation continue SubCA1
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# rsakeypair SubCA11
Device(ca-trustpoint)# end
```

ピアから下位 CA への証明書チェーン検証の設定

次の設定例では、ピア証明書および SubCA1 証明書が有効にされます。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint RootCA
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# chain-validation stop
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# rsakeypair RootCA
Device(ca-trustpoint)# exit
Device(config)# crypto pki trustpoint SubCA1
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# chain-validation continue RootCA
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# rsakeypair SubCA1
Device(ca-trustpoint)# exit
Device(config)# crypto pki trustpoint SubCA11
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# chain-validation continue SubCA1
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# rsakeypair SubCA11
Device(ca-trustpoint)# end
```

証明書チェーンの欠落確認の設定

次の設定例では、SubCA1 が、設定済みの Cisco IOS 階層にはないが、提出された証明書チェーンでピアによって提示されたと想定しています。

ピアが、提出された証明書チェーンで SubCA1 証明書を提示した場合、ピア、SubCA11、および SubCA1 の各証明書が有効になります。

ピアが、提出された証明書チェーンで SubCA1 証明書を提示しない場合、チェーンの検証は失敗します。

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpoint RootCA
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# chain-validation stop
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# rsakeypair RootCA
Device(ca-trustpoint)# exit
Device(config)# crypto pki trustpoint SubCA11
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# chain-validation continue RootCA
Device(ca-trustpoint)# revocation-check none
Device(ca-trustpoint)# rsakeypair SubCA11
Device(ca-trustpoint)# end
```

PKI での証明書の許可および失効の追加リファレンス

シスコのテクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカルサポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	https://www.cisco.com/cisco/web/support/index.html

PKI での証明書の許可および失効の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

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

リリース	機能	機能情報
	PKI での証明書の許可および失効	証明書には、指定された処理の実行をデバイスまたはユーザーが許可されているかどうかの判別に使用されるフィールドがいくつか含まれています。また、証明書ベース ACL は失効、許可、またはトラストポイントなどの PKI コンポーネントを使用するタイミングを判別するのに役立ちます。

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

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。