Cisco IOS リリース 15.2(7)Ex（Catalyst 3560-CX および 2960-CX スイッチ）統合プラットフォームコンフィギュレーションガイド

初版：2019 年 3 月 27 日

最終更新：2022 年 9 月 29 日

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

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

目 次

はじめに

ここでは、このマニュアルの表記法、および他資料の入手方法について説明します。また、シスコ製品のマニュアルの最新情報についても説明します。

表記法

このマニュアルでは、以下の表記法を使用しています。

表記法	説明
^ または Ctrl	^ 記号と Ctrl は両方ともキーボードの Control（Ctrl）キーを表します。たとえば、^D または Ctrl+D というキーの組み合わせは、Ctrl キーを押しながら D キーを押すことを意味します（ここではキーを大文字で表記していますが、小文字で入力してもかまいません）。
太字	コマンド、キーワード、およびユーザーが入力するテキストは**太字**で記載されます。
italic フォント	文書のタイトル、新規用語、強調する用語、およびユーザが値を指定する引数は、イタリック体で示しています。
`Courier` フォント	システムが表示する端末セッションおよび情報は、`courier` フォントで示しています。
`太字の courier` フォント	**`太字の Courier`** フォントは、ユーザが入力しなければならないテキストを示します。
[x]	角カッコの中の要素は、省略可能です。
...	構文要素の後の省略記号（3 つの連続する太字ではないピリオドでスペースを含まない）は、その要素を繰り返すことができることを示します。

表記法	説明
|	パイプと呼ばれる縦棒は、一連のキーワードまたは引数の選択肢であることを示します。
[x | y]	どれか1つを選択できる省略可能なキーワードは、角カッコで囲み、縦棒で区切って示しています。
{x | y}	どれか1つを選択しなければならない必須キーワードは、波カッコで囲み、縦棒で区切って示しています。
[x {y | z}]	角かっこまたは波かっこが入れ子になっている箇所は、任意または必須の要素内の任意または必須の選択肢であることを表します。角かっこ内の波かっこと縦棒は、省略可能な要素内で選択すべき必須の要素を示しています。
string	引用符を付けない一組の文字。string の前後には引用符を使用しません。引用符を使用すると、その引用符も含めてstringとみなされます。
< >	パスワードのように出力されない文字は、山カッコで囲んで示しています。
[ ]	システムプロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

読者への警告の表記法

このマニュアルでは、読者への警告に次の表記法を使用しています。

（注） 「注釈」です。役立つ情報やこのマニュアルに記載されていない参照資料を紹介しています。

ヒント 「問題解決に役立つ情報」です。

注意 「要注意」の意味です。機器の損傷またはデータ損失を予防するための注意事項が記述されています。

ワンポイント アドバイス

時間を節約する方法です。ここに紹介している方法で作業を行うと、時間を短縮できます。

警告

安全上の重要な注意事項

この警告マークは「危険」の意味です。人身事故を予防するための注意事項が記述されています。機器の取り扱い作業を行うときは、電気回路の危険性に注意し、一般的な事故防止対策に留意してください。各警告の最後に記載されているステートメント番号を基に、装置に付属の安全についての警告を参照してください。ステートメント 1071

SAVE THESE INSTRUCTIONS

関連資料

マニュアルの入手方法およびテクニカル サポート

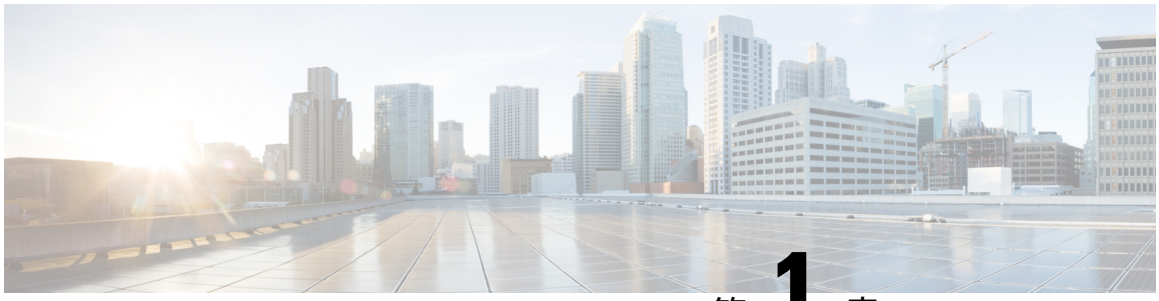
マニュアルの入手方法、テクニカルサポート、その他の有用な情報について、毎月更新される『更新情報』を参照してください。シスコの新規および改訂版の技術マニュアルの一覧も示されています。

http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html

『更新情報』は RSS フィードとして購読できます。また、リーダー アプリケーションを使用してコンテンツがデスクトップに直接配信されるように設定することもできます。RSS フィードは無料のサービスです。シスコは現在、RSS バージョン 2.0 をサポートしています。

第 部

インターフェイスおよびハードウェア



第 1 章

インターフェイス特性の設定

インターフェイス特性の設定について

インターフェイス タイプ

ここでは、deviceでサポートされているインターフェイスの異なるタイプについて説明します。また、インターフェイスの物理特性に応じた設定手順についても説明します。

ポートベースの VLAN

VLANは、ユーザの物理的な位置に関係なく、機能、チーム、またはアプリケーションなどで論理的に分割された、スイッチによるネットワークです。ポートで受信したパケットが転送されるのは、その受信ポートと同じVLANに属するポートに限られます。異なるVLAN上のネットワーク デバイスは、VLAN 間でトラフィックをルーティングするレイヤ 3 デバイスがなければ、互いに通信できません。

VLANに分割することにより、VLAN内でトラフィック用の堅固なファイアウォールを実現します。また、各 VLAN には固有の MAC アドレス テーブルがあります。VLAN が認識されるのは、ローカル ポートが VLAN に対応するように設定されたとき、VLAN Trunking Protocol（VTP）トランク上のネイバーからその存在を学習したとき、またはユーザがVLANを作成したときです。

VLANを設定するには、**vlan** *vlan-id* グローバル コンフィギュレーション コマンドを使用して、VLAN コンフィギュレーション モードを開始します。標準範囲 VLAN（VLAN ID 1 ～ 1005）の VLAN 設定は、VLAN データベースに保存されます。VTP がバージョン 1 または 2 の場合に、拡張範囲 VLAN（VLAN ID が 1006 ～ 4094）を設定するには、最初に VTP モードをトランスペアレントに設定する必要があります。トランスペアレント モードで作成された拡張範囲 VLAN は、VLAN データベースには追加されませんが、deviceの実行コンフィギュレーション

に保存されます。VTPバージョン3では、クライアントまたはサーバモードで拡張範囲VLANを作成できます。これらの VLAN は VLAN データベースに格納されます。

switchport インターフェイス コンフィギュレーション コマンドを使用すると、VLAN にポートが追加されます。

- インターフェイスを特定します。
- トランク ポートには、トランク特性を設定し、必要に応じて所属できる VLAN を定義します。
- アクセス ポートには、所属する VLAN を設定して定義します。

スイッチ ポート

スイッチ ポートは、物理ポートに対応付けられたレイヤ2専用インターフェイスです。スイッチ ポートは1つまたは複数の VLAN に所属します。スイッチ ポートは、アクセス ポートまたはトランク ポートにも使用できます。ポートは、アクセス ポートまたはトランク ポートに設定できます。また、ポート単位で Dynamic Trunking Protocol（DTP）を稼働させ、リンクのもう一端のポートとネゴシエートすることで、スイッチ ポート モードも設定できます。スイッチ ポートは、物理インターフェイスおよび関連付けられているレイヤ2プロトコルの管理に使用され、ルーティングやブリッジングは処理しません。

スイッチポートの設定には、**switchport** インターフェイス コンフィギュレーション コマンドを使用します。

アクセス ポート

アクセス ポートは（音声 VLAN ポートとして設定されている場合を除き）1 つの VLAN だけに所属し、その VLAN のトラフィックだけを伝送します。トラフィックは、VLAN タグが付いていないネイティブ形式で送受信されます。アクセス ポートに着信したトラフィックは、ポートに割り当てられている VLAN に所属すると見なされます。アクセス ポートがタグ付きパケット（スイッチ間リンク（ISL）またはタグ付き IEEE 802.1Q）を受信した場合、そのパケットはドロップされ、送信元アドレスは学習されません。

サポートされているアクセス ポートのタイプは、次のとおりです。

- スタティック アクセス ポート。このポートは、手動でVLANに割り当てます（IEEE 802.1xで使用する場合は RADIUS サーバを使用します）。
- ダイナミック アクセス ポートの VLAN メンバーシップは、着信パケットを通じて学習されます。デフォルトでは、ダイナミック アクセス ポートはどの VLAN のメンバーでもなく、ポートとの伝送はポートの VLAN メンバーシップが検出されたときにだけイネーブルになります。device上のダイナミック アクセス ポートは、VLAN メンバーシップ ポリシー サーバー（VMPS）によって VLAN に割り当てられます。Catalyst 6500 シリーズ スイッチを VMPS にできます。このdeviceを VMPS サーバーにすることはできません。

また、Cisco IP Phone と接続するアクセス ポートを、1 つの VLAN は音声トラフィック用に、もう 1 つの VLAN は Cisco IP Phone に接続しているデバイスからのデータ トラフィック用に使用するように設定できます。

トランク ポート

トランク ポートは複数の VLAN のトラフィックを伝送し、デフォルトで VLAN データベース内のすべての VLAN のメンバとなります。

deviceは IEEE 802.1Q トランク ポートだけをサポートします。IEEE 802.1Q トランク ポートは、タグ付きとタグなしの両方のトラフィックを同時にサポートします。IEEE 802.1Q トランク ポートは、デフォルトのポート VLAN ID（PVID）に割り当てられ、すべてのタグなしトラフィックはポートのデフォルト PVID 上を流れます。NULL VLAN ID を備えたすべてのタグなしおよびタグ付きトラフィックは、ポートのデフォルト PVID に所属するものと見なされます。発信ポートのデフォルト PVID と等しい VLAN ID を持つパケットは、タグなしで送信されます。残りのトラフィックはすべて、VLAN タグ付きで送信されます。

デフォルトでは、トランク ポートは、VTP に認識されているすべての VLAN のメンバですが、トランク ポートごとに VLAN の許可リストを設定して、VLAN メンバーシップを制限できます。許可 VLAN のリストは、その他のポートには影響を与えませんが、対応トランク ポートには影響を与えます。デフォルトでは、使用可能なすべての VLAN（VLAN ID 1 ～ 4094）が許可リストに含まれます。トランクポートは、VTP が VLAN を認識し、VLAN が有効な状態にある場合に限り、VLAN のメンバーになることができます。VTP が新しい有効になっている VLAN を認識し、その VLAN がトランクポートの許可リストに登録されている場合、トランクポートは自動的にその VLAN のメンバになり、トラフィックはその VLAN のトランク ポート間で転送されます。VTP が、VLAN のトランクポートの許可リストに登録されていない、新しい有効な VLAN を認識した場合、ポートはその VLAN のメンバーにはならず、その VLAN のトラフィックはそのポート間で転送されません。

スイッチ仮想インターフェイス

スイッチ仮想インターフェイス（SVI）は、スイッチポートの VLAN を、システムのルーティング機能またはブリッジング機能に対する 1 つのインターフェイスとして表します。1 つの VLAN に関連付けることができる SVI は 1 つだけです。VLAN に対して SVI を設定するのは、VLAN 間でルーティングするため、またはdeviceに IP ホスト接続を提供するためだけです。デフォルトでは、SVI はデフォルト VLAN（VLAN 1）用に作成され、リモート deviceの管理を可能にします。追加の SVI は明示的に設定する必要があります。

（注） インターフェイス VLAN 1 は削除できません。

SVI はシステムにしか IP ホスト接続を行いません。SVI は、VLAN インターフェイスに対して **vlan** インターフェイス コンフィギュレーション コマンドを実行した際に初めて作成されます。VLAN は、ISL または IEEE 802.1Q カプセル化トランク上のデータ フレームに関連付けられた VLAN タグ、あるいはアクセス ポート用に設定された VLAN ID に対応します。トラフィックをルーティングするそれぞれの VLAN に対して VLAN インターフェイスを設定し、IP アドレスを割り当ててください。

interface range コマンドを使用して、範囲内の既存の VLAN SVI を設定できます。interface range コマンド下で入力したコマンドは、範囲内の既存の VLAN SVI すべてに適用されます。コマンド **interface range create vlan** *x-y*を入力すると、まだ存在しない指定された範囲内のすべての

vlan を作成できます。VLAN インターフェイスが作成されると、**interface range vlan** *id*を使用して VLAN インターフェイスを設定できます。

物理ポートと関連付けられていない場合、SVI を作成してもアクティブにはなりません。

SVI 自動ステート除外

VLAN 上の複数のポートを装備した SVI のラインステートは、次の条件を満たしたときにはアップ状態になります。

- VLAN が存在し、deviceの VLAN データベースでアクティブです。
- VLAN インターフェイスが存在し、管理上のダウン状態ではありません。
- 少なくとも 1 つのレイヤ 2（アクセスまたはトランク）ポートが存在し、この VLAN のリンクがアップ状態であり、ポートが VLAN でスパニングツリー フォワーディング ステートです。

（注） 対応する VLAN リンクに属する最初のスイッチポートが起動し、STP フォワーディング ステートになると、VLAN インターフェイスのプロトコル リンク ステートがアップ状態になります。

VLANに複数のポートがある場合のデフォルトのアクションでは、VLAN内のすべてのポートがダウンすると SVI もダウン状態になります。SVI 自動ステート除外機能を使用して、SVI ラインステート アップオアダウン計算に含まれないようにポートを設定できます。たとえば、VLAN 上で 1 つのアクティブ ポートだけがモニターリング ポートである場合、他のすべてのポートがダウンすると VLAN もダウンするよう自動ステート除外機能をポートに設定できます。ポートでイネーブルである場合、**autostate exclude** はポート上でイネーブルであるすべての VLAN に適用されます。

VLAN 内の 1 つのレイヤ 2 ポートに収束時間がある場合（STP リスニング/ラーニング ステートからフォワーディング ステートへの移行）、VLAN インターフェイスが起動します。これにより、ルーティングプロトコルなどの機能は、完全に動作した場合と同様にVLANインターフェイスを使用せず、他の問題を最小限にします。

EtherChannel ポートグループ

EtherChannel ポートグループは、複数のスイッチポートを 1 つのスイッチポートとして扱います。このようなポート グループは、devices 間、または devices およびサーバー間で高帯域接続を行う単一論理ポートとして動作します。EtherChannelは、チャネルのリンク全体でトラフィックの負荷を分散させます。EtherChannel 内のリンクで障害が発生すると、それまでその障害リンクで伝送されていたトラフィックが残りのリンクに切り替えられます。複数のトランク ポートを 1 つの論理トランク ポートに、または複数のアクセス ポートを 1 つの論理アクセス ポートにまとめることができます。ほとんどのプロトコルは単一のまたは集約スイッチ ポートで動作し、ポート グループ内の物理ポートを認識しません。例外は、DTP、Cisco Discovery Protocol（CDP）、およびポート集約プロトコル（PAgP）で、物理ポート上でしか動作しません。

EtherChannel を設定するとき、ポートチャネル論理インターフェイスを作成し、EtherChannel にインターフェイスを割り当てます。レイヤ 2 インターフェイスの場合は、**channel-group** インターフェイス コンフィギュレーション コマンドを使用して、ポートチャネル論理インターフェイスを動的に作成します。このコマンドは物理および論理ポートをバインドします。

（注） Cisco Catalyst 2960-CX および 3560-CX は最大で 6 個のイーサチャネル ポート グループをサポートします。

Power over Ethernet（PoE）ポート

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

- シスコ先行標準受電装置（Cisco IP Phone および Cisco Aironet アクセス ポイントなど）
- IEEE 802.3af 準拠の受電装置

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

スイッチの USB ポートの使用

USB ミニタイプ B コンソール ポート

には、 device 次のコンソールポートがあります。

- USB ミニタイプ B コンソール接続
- RJ-45 コンソール ポート

コンソール出力は両方のポートに接続されたデバイスに表示されますが、コンソール入力は一度に 1 つのポートしかアクティブになりません。デフォルトでは、USB コネクタは RJ-45 コネクタよりも優先されます。

（注） Windows PC には、USB ポートのドライバが必要です。ドライバ インストレーションの手順については、ハードウェア インストレーション ガイドを参照してください。

付属の USB Type A-to-USB mini-Type B ケーブルを使用して、PC またはその他のデバイスを deviceに接続します。接続されたデバイスには、ターミナル エミュレーション アプリケーションが必要です。device が、ホスト機能をサポートする電源の入っているデバイス（PC など）への有効な USB 接続を検出すると、RJ-45 コンソールからの入力がただちに無効になり、USB コンソールからの入力が有効になります。USB 接続が削除されると、RJ-45 コンソールからの

入力はただちに再度イネーブルになります。device の LED は、どのコンソール接続が使用中であるかを示します。

コンソール ポート変更ログ

ソフトウェア起動時に、ログに USB または RJ-45 コンソールのいずれがアクティブであるかが示されます。すべてのdeviceは常にまず RJ-45 メディア タイプを表示します。

USB ケーブルが取り外されるか、PC が USB 接続を非アクティブ化すると、ハードウェアは自動的に RJ-45 コンソール インターフェイスに変わります。

コンソール タイプが常に RJ-45 であるように設定でき、さらに USB コネクタの無活動タイムアウトを設定できます。

USB タイプ A ポート

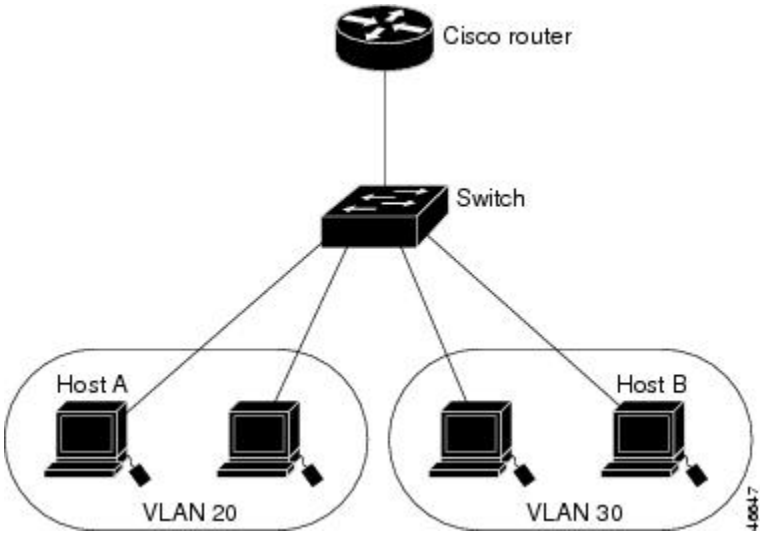
USB タイプ A ポートは、外部 USB フラッシュ デバイス（サム ドライブまたは USB キーとも呼ばれる）へのアクセスを提供します。スイッチで Cisco 64 MB、256 MB、512 MB、1 GB、4 GB、および 8 GB のフラッシュ ドライブがサポートされます。標準 Cisco IOS コマンドライン インターフェイス（CLI）コマンドを使用して、フラッシュ デバイスの読み取り、書き込み、および、コピー元やコピー先として使用できます。スイッチを USB フラッシュ ドライブから起動するようにも設定できます。

インターフェイスの接続

単一 VLAN 内のデバイスは、スイッチを通じて直接通信できます。異なる VLAN に属すポート間では、ルーティングデバイスを介さなければデータを交換できません。

次の設定例では、VLAN 20 のホスト A が VLAN 30 のホスト B にデータを送信する場合は、データはホスト A からデバイスを経由してルータへ送られた後、再びデバイスに戻ってからホスト B へ送信される必要があります。

図 *1*：スイッチと *VLAN* との接続



標準のレイヤ 2 スイッチを使用すると、異なる VLAN のポートは、ルータを通じて情報を交換する必要があります。

（注） Catalyst 3560-CX スイッチおよび 2960-CX スイッチは、スタッキングをサポートしません。このドキュメント全体を通じて、すべてのスタッキングへの参照を無視します。

インターフェイス コンフィギュレーション モード

deviceは、次のインターフェイス タイプをサポートします。

- 物理ポート：device ポートおよびルーテッド ポート
- VLAN：スイッチ仮想インターフェイス
- ポート チャネル：EtherChannel インターフェイス

インターフェイス範囲も設定できます。

物理インターフェイス（ポート）を設定するには、インターフェイス タイプ、モジュール番号、およびdeviceポート番号を指定して、インターフェイスコンフィギュレーションモードを開始します。

- タイプ：10/100/1000 Mbps イーサネット ポートの場合はギガビット イーサネット（gigabitethernet または gi）、または Small Form-Factor Pluggable（SFP）モジュール ギガビット イーサネット インターフェイス（gigabitethernet または gi）。
- モジュール番号：スイッチのモジュールまたはスロット番号（常に 0）。
- ポート番号：スイッチ上のインターフェイス番号。10/100/1000 ポート番号は常に 1 から始まり、スイッチに向かって左のポートから順番に付けられています。たとえば、gigabitethernet1/0/1 またはr gigabitethernet1/0/8 のようになります。10/100/1000 ポートと SFP モジュール ポートのあるスイッチの場合、SFP モジュール ポートの番号は 10/100/1000 ポートの後に連続して付けられます。

スイッチ上のインターフェイスの位置を物理的に確認することで、物理インターフェイスを識別できます。**show** 特権 EXEC コマンドを使用して、スイッチ上の特定のインターフェイスまたはすべてのインターフェイスに関する情報を表示することもできます。以降、この章では、主に物理インターフェイスの設定手順について説明します。

イーサネット インターフェイスのデフォルト設定

次の表は、レイヤ2インターフェイスにのみ適用される一部の機能を含む、イーサネットインターフェイスのデフォルト設定を示しています。

表 *1:* レイヤ *2* イーサネット インターフェイスのデフォルト設定

機能	デフォルト設定
動作モード	レイヤ2またはスイッチングモード（**switchport** コマンド）。
VLAN 許容範囲	VLAN 1 ～ 4094。
デフォルト VLAN（アクセス ポート用）	VLAN 1。
ネイティブ VLAN（IEEE 802.1Q トランク用）	VLAN 1。
802.1p プライオリティ タグ付きトラフィック	VLAN 0 のタグが付いたパケットをすべてドロップ。
VLAN トランキング	Switchport mode dynamic auto（DTP をサポート）。
ポート イネーブル ステート	すべてのポートが有効。
ポート記述	未定義。
速度	自動ネゴシエーション（10 ギガビット インターフェイス上では未サポート）。
デュプレックス モード	自動ネゴシエーション（10 ギガビット インターフェイス上では未サポート）。
フロー制御	フロー制御は **receive: off** に設定されます。送信パケットでは常にオフです。
EtherChannel（PAgP）	すべてのイーサネット ポートで無効。
ポート ブロッキング（不明マルチキャストおよび不明ユニキャスト トラフィック）	ディセーブル（ブロッキングされない）。
ブロードキャスト、マルチキャスト、およびユニキャスト ストーム制御	無効。
保護ポート	ディセーブル。
ポート セキュリティ	ディセーブル。

機能	デフォルト設定
PortFast	無効。
Auto-MDIX	有効。 （注） 受電デバイスがクロスケーブルでdeviceに接続されている場合、deviceは、IEEE 802.3afに完全には準拠していない、Cisco IP Phoneやアクセスポイントなどの準規格の受電をサポートしていない場合があります。これは、スイッチ ポート上で Automatic Medium-Dependent Interface Crossover（Auto-MIDX）が有効かどうかは関係ありません。
Power over Ethernet（PoE）	有効（auto）。
キープアライブ メッセージ	SFP モジュールでディセーブル。他のすべてのポートでイネーブル。

インターフェイス速度およびデュプレックス モード

スイッチのイーサネットインターフェイスは、全二重または半二重モードのいずれかで、10、100、または 1000 Mb/s で動作します。全二重モードの場合、2 つのステーションが同時にトラフィックを送受信できます。通常、10 Mbps ポートは半二重モードで動作します。これは、各ステーションがトラフィックを受信するか、送信するかのどちらか一方しかできないことを意味します。

スイッチモジュールには、、ギガビットイーサネット（10/100/1000 Mbps）ポート、、および SFP モジュールをサポートする Small Form-Factor Pluggable（SFP）モジュール スロットが含まれます。

速度とデュプレックス モードの設定時の注意事項

インターフェイス速度とデュプレックスモードを設定する際には、次のガイドラインに注意してください。

- PoE スイッチでは自動ネゴシエーションをディセーブルにしないでください。
- ギガビット イーサネット（10/100/1000 Mbps）ポートは、すべての速度オプションとデュプレックス オプション（自動、半二重、全二重）をサポートします。ただし、1000 Mbps で稼働させているギガビットイーサネットポートは、半二重モードをサポートしません。
- SFP モジュール ポートの場合、次の SFP モジュール タイプによって速度とデュプレックスの CLI（コマンドライン インターフェイス）オプションが変わります。

- 1000BASE-*x*（-*x* は -BX、-CWDM、-LX、-SX、-ZX）SFP モジュールポートは、**speed** インターフェイス コンフィギュレーション コマンドで **nonegotiate** キーワードをサポートします。デュプレックス オプションはサポートされません。
- 1000BASE-T SFP モジュール ポートは、10/100/1000 Mbps ポートと同一の速度とデュプレックス オプションをサポートします。
-

- 回線の両側で自動ネゴシエーションがサポートされる場合は、デフォルト設定の **auto** ネゴシエーションの使用を強くお勧めします。
- 一方のインターフェイスが自動ネゴシエーションをサポートし、もう一方がサポートしない場合は、両方のインターフェイス上でデュプレックスと速度を設定します。サポートする側で **auto** 設定を使用しないでください。
- STP が有効な場合にポートを再設定すると、device がループの有無を調べるために最大で 30 秒かかる可能性があります。STP の再設定が行われている間、ポート LED はオレンジに点灯します。
- ベストプラクティスとして、速度とデュプレックスのオプションをリンク上で自動に設定するか、リンク終端の両側で固定に設定することを推奨します。リンクのいずれかの終端が自動に設定され、もう一方が固定に設定されていると、正常な動作として、リンクはアップしません。

注意 インターフェイス速度とデュプレックスモードの設定を変更すると、再設定中にインターフェイスがシャットダウンし、再び有効になる場合があります。

IEEE 802.3x フロー制御

フロー制御により、接続しているイーサネットポートは、輻輳しているノードがリンク動作をもう一方の端で一時停止できるようにすることによって、輻輳時のトラフィック レートを制御できます。あるポートで輻輳が生じ、それ以上はトラフィックを受信できなくなった場合、ポーズフレームを送信することによって、その状態が解消されるまで送信を中止するように、そのポートから相手ポートに通知します。ポーズ フレームを受信すると、送信側デバイスはデータ パケットの送信を中止するので、輻輳時のデータ パケット損失が防止されます。

（注） スイッチ ポートは、ポーズ フレームを受信できますが、送信はできません。

flowcontrol インターフェイス コンフィギュレーション コマンドを使用して、インターフェイスのポーズフレームを **receive** する機能を **on**、**off**、または **desired** に設定します。

desired に設定した場合、インターフェイスはフロー制御パケットの送信を必要とする接続デバイス、または必要ではないがフロー制御パケットを送信できる接続デバイスに対して動作できます。

デバイスのフロー制御設定には、次のルールが適用されます。

- **receive on**（または **desired**）：ポートはポーズフレームを送信できませんが、ポーズフレームを送信する必要のある、または送信できる接続デバイスと組み合わせて使用できます。ポーズフレームの受信は可能です。
- **receive off**：フロー制御はどちらの方向にも動作しません。輻輳が生じても、リンクの相手側に通知はなく、どちら側の装置も休止フレームの送受信を行いません。

（注） コマンドの設定と、その結果生じるローカルおよびリモートポートでのフロー制御解決の詳細については、このリリースのコマンドリファレンスに記載された **flowcontrol** インターフェイス コンフィギュレーション コマンドを参照してください。

インターフェイス特性の設定方法

インターフェイスの設定

次の一般的な手順は、すべてのインターフェイス設定プロセスに当てはまります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** 例： `スイッチ(config)# interface gigabitethernet 1/0/1` `スイッチ(config-if)#`	インターフェイスタイプおよびコネクタの数を識別します。 （注） インターフェイス タイプとインターフェイス番号の間にスペースを入れる必要はありません。たとえば、前の行では、**gigabitethernet 1/0/1**、**gigabitethernet1/0/1**、**gi 1/0/1**、または **gi1/0/1** のいずれかを指定できます。

	コマンドまたはアクション	目的
ステップ 4	各 **interface** コマンドの後ろに、インターフェイスに必要なインターフェイス コンフィギュレーション コマンドを続けて入力します。	インターフェイス上で実行するプロトコルとアプリケーションを定義します。別のインターフェイス コマンドまたは **end** を入力して特権 EXEC モードに戻ると、コマンドが収集されてインターフェイスに適用されます。
ステップ 5	**interface range** または **interface range macro**	（任意）インターフェイスの範囲を設定します。 （注） ある範囲内で設定したインターフェイスは、同じタイプである必要があります。また、同じ機能オプションを指定して設定しなければなりません。
ステップ 6	**show interfaces**	スイッチ上のまたはスイッチに対して設定されたすべてのインターフェイスのリストを表示します。デバイスがサポートする各インターフェイスまたは指定したインターフェイスのレポートが出力されます。

インターフェイスに関する記述の追加

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **description** *string*
5. **end**
6. **show interfaces** *interface-id* **description**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/2**	記述を追加するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**description** *string* 例： スイッチ(config-if)# **description Connects to Marketing**	インターフェイスに関する説明を追加します（最大 240 文字）。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show interfaces** *interface-id* **description**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

インターフェイス範囲の設定

同じ設定パラメータを持つ複数のインターフェイスを設定するには、**interface range** グローバル コンフィギュレーション コマンドを使用します。インターフェイス レンジ コンフィギュレーション モードを開始すると、このモードを終了するまで、入力されたすべてのコマンド パラメータはその範囲内のすべてのインターフェイスに対するものと見なされます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface range** {*port-range* | **macro** *macro_name*}
4. **end**
5. **show interfaces** [*interface-id*]
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface range** {*port-range* | **macro** *macro_name*} 例： スイッチ(config)# **interface range macro**	設定するインターフェイス範囲（VLAN または物理ポート）を指定し、インターフェイス コンフィギュレーション モードを開始します。 • **interface range** コマンドを使用すると、最大 5 つのポート範囲または定義済みマクロを 1 つ設定できます。 • **macro** 変数は、「インターフェイス レンジ マクロの設定および使用方法」の項で説明しています。 • カンマで区切った *port-range* では、各エントリに対応するインターフェイス タイプを入力し、カンマの前後にスペースを含めます。 • ハイフンで区切った *port-range* では、インターフェイス タイプの再入力は不要ですが、ハイフンの前後にスペースを入力する必要があります。 （注） この時点で、通常のコンフィギュレーション コマンドを使用して、範囲内のすべてのインターフェイスにコンフィギュレーション パラメータを適用します。各コマンドは、入力されたとおりに実行されます。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 5	**show interfaces** [*interface-id*] 例： スイッチ# **show interfaces**	指定した範囲内のインターフェイスの設定を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

インターフェイス レンジ マクロの設定および使用方法

インターフェイス レンジ マクロを作成すると、設定するインターフェイスの範囲を自動的に選択できます。**interface range macro** グローバル コンフィギュレーション コマンド文字列で **macro** キーワードを使用する前に、**define interface-range** グローバル コンフィギュレーション コマンドを使用してマクロを定義する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **define interface-range** *macro_name interface-range*
4. **interface range macro** *macro_name*
5. **end**
6. **show running-config | include define**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**define interface-range** *macro_name interface-range* 例： スイッチ(config)# **define interface-range enet_list gigabitethernet 1/0/1 - 2**	インターフェイス範囲マクロを定義して、NVRAM に保存します。 • *macro_name* は、最大 32 文字の文字列です。 • マクロには、カンマで区切ったインターフェイスを 5 つまで指定できます。 • それぞれの *interface-range* は、同じポート タイプで構成されていなければなりません。 （注） **interface range macro** グローバル コンフィギュレーション コマンド文字列で **macro** キーワードを使用する前に、**define interface-range** グローバル コンフィギュレーション コマンドを使用してマクロを定義する必要があります。
ステップ 4	**interface range macro** *macro_name* 例： スイッチ(config)# **interface range macro enet_list**	*macro_name* の名前でインターフェイス範囲マクロに保存された値を使用することによって、設定するインターフェイスの範囲を選択します。 ここで、通常のコンフィギュレーション コマンドを使用して、定義したマクロ内のすべてのインターフェイスに設定を適用できます。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config | include define** 例： スイッチ# **show running-config | include define**	定義済みのインターフェイス範囲マクロの設定を表示します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

イーサネット インターフェイスの設定

インターフェイス速度およびデュプレックス パラメータの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **duplex** {**auto** | **full** | **half**}
5. **end**
6. **show interfaces** *interface-id*
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/3`	設定する物理インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**duplex** {**auto**	**full**	**half**} 例： `スイッチ(config-if)# duplex half`	このコマンドは、10 ギガビット イーサネット インターフェイスでは使用できません。 インターフェイスのデュプレックス パラメータを入力します。 半二重モードをイネーブルにします（10 または 100Mbpsのみで動作するインターフェイスの場合）。1000 Mbps で動作するインターフェイスには半二重モードを設定できません。

	コマンドまたはアクション	目的
		デュプレックス設定を行うことができるのは、速度が **auto** に設定されている場合です。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show interfaces** *interface-id* 例： スイッチ# **show interfaces gigabitethernet 1/0/3**	インターフェイス速度およびデュプレックス モードの設定を表示します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IEEE 802.3x フロー制御の設定

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **flowcontrol** {**receive**} {**on** | **off** | **desired**}
4. **end**
5. **show interfaces** *interface-id*
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例：	設定する物理インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**flowcontrol** {**receive**} {**on** | **off** | **desired**} 例： スイッチ(config-if)# **flowcontrol receive on**	ポートのフロー制御モードを設定します。
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show interfaces** *interface-id* 例：	インターフェイス フロー制御の設定を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SVI 自動ステート除外の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport autostate exclude**
5. **end**
6. **show running config interface** *interface-id*
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/2**	レイヤ 2 インターフェイス（物理ポートまたはポートチャネル）を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport autostate exclude** 例： スイッチ(config-if)# **switchport autostate exclude**	SVI ライン ステート（アップまたはダウン）のステータスを定義する際、アクセスまたはトランクポートを除外します。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running config interface** *interface-id*	（任意）実行コンフィギュレーションを表示します。 設定を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

インターフェイスのシャットダウンおよび再起動

インターフェイスをシャットダウンすると、指定されたインターフェイスのすべての機能が無効になり、使用不可能であることがすべてのモニタ コマンドの出力に表示されます。この情報は、すべてのダイナミック ルーティング プロトコルを通じて、他のネットワーク サーバに伝達されます。ルーティング アップデートには、インターフェイス情報は含まれません。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** { **vlan** *vlan-id*} | { **gigabitethernet***interface-id*} | { **port-channel** *port-channel-number*}
4. **shutdown**
5. **no shutdown**

6. **end**
7. **show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** { **vlan** *vlan-id*}	{ **gigabitethernet***interface-id*}	{ **port-channel** *port-channel-number*} 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	設定するインターフェイスを選択します。
ステップ 4	**shutdown** 例： `スイッチ(config-if)# shutdown`	インターフェイスをシャットダウンします。
ステップ 5	**no shutdown** 例： `スイッチ(config-if)# no shutdown`	インターフェイスを再起動します。
ステップ 6	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。

コンソール メディア タイプの設定

コンソール メディア タイプを RJ-45 に設定するには、次の手順を実行します。RJ-45 としてコンソールを設定すると、USB コンソールの動作は無効になり、入力は RJ-45 コネクタからのみ供給されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **media-type rj45**
5. **end**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**line console 0** 例： `スイッチ(config)# line console 0`	コンソールを設定し、ライン コンフィギュレーション モードを開始します。
ステップ 4	**media-type rj45** 例： `スイッチ(config-line)# media-type rj45`	コンソール メディア タイプが RJ-45 ポート以外に設定されないようにします。このコマンドを入力せず、両方のタイプが接続された場合は、デフォルトで USB ポートが使用されます。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

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

USB 無活動タイムアウトの設定

無活動タイムアウトを設定している場合、USB コンソール ポートがアクティブ化されているものの、指定された時間内にポートで入力アクティビティがないときに、RJ-45 コンソール ポートが再度アクティブになります。タイムアウトのために USB コンソール ポートは非アクティブ化された場合、USB ポートを切断し、再接続すると、動作を回復できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **usb-inactivity-timeout** *timeout-minutes*
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**line console 0** 例： スイッチ(config)# **line console 0**	コンソールを設定し、ライン コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**usb-inactivity-timeout** *timeout-minutes* 例： `スイッチ(config-line)# usb-inactivity-timeout 30`	コンソールポートの無活動タイムアウトを指定します。指定できる範囲は 1 ～ 240 分です。デフォルトでは、タイムアウトが設定されていません。
ステップ 5	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

インターフェイス特性のモニタ

インターフェイス ステータスの監視

特権EXECプロンプトにコマンドを入力することによって、ソフトウェアおよびハードウェアのバージョン、コンフィギュレーション、インターフェイスに関する統計情報などのインターフェイス情報を表示できます。

表 *2:* インターフェイス用の ***show*** コマンド

コマンド	目的
show interfaces *interface-number* **downshift module***module-number*	指定したインターフェイスとモジュールのダウンシフトステータスの詳細を表示します。
show interfaces *interface-id* **status** [**err-disabled**]	インターフェイスのステータスまたはerror-disabledステートにあるインターフェイスのリストを表示します。
show interfaces [*interface-id*] **switchport**	スイッチング（非ルーティング）ポートの管理上および動作上のステータスを表示します。このコマンドを使用すると、ポートがルーティングまたはスイッチングのどちらのモードにあるかが判別できます。
show interfaces [*interface-id*] **description**	1 つのインターフェイスまたはすべてのインターフェイスに関する記述とインターフェイスのステータスを表示します。
show ip interface [*interface-id*]	IP ルーティング用に設定されたすべてのインターフェイスまたは特定のインターフェイスについて、使用できるかどうかを表示します。

コマンド	目的
show interface [*interface-id*] **stats**	インターフェイスのパスごとに入出力パケットを表示します。
show interfaces *interface-id*	（任意）インターフェイスの速度およびデュプレックスを表示します。
show interfaces transceiver dom-supported-list	（任意）接続 SFP モジュールの Digital Optical Monitoring（DOM）ステータスを表示します。
show interfaces transceiver properties	（任意）インターフェイスの温度、電圧、電流量を表示します。
show interfaces [*interface-id*] [{**transceiver properties** | **detail**}] *module number*]	SFP モジュールに関する物理および動作ステータスを表示します。
show running-config interface [*interface-id*]	インターフェイスに対応する RAM 上の実行コンフィギュレーションを表示します。
show version	ハードウェア設定、ソフトウェア バージョン、コンフィギュレーション ファイルの名前と送信元、およびブートイメージを表示します。
show controllers ethernet-controller *interface-id* **phy**	インターフェイスの Auto-MDIX 動作ステートを表示します。

インターフェイスおよびカウンタのクリアとリセット

表 *3:* インターフェイス用の ***clear*** コマンド

コマンド	目的
clear counters [*interface-id*]	インターフェイス カウンタをクリアします。
clear interface *interface-id*	インターフェイスのハードウェア ロジックをリセットします。
clear line [*number* | **console 0** | **vty** *number*]	非同期シリアル回線に関するハードウェア ロジックをリセットします。

（注） **clear counters** 特権 EXEC コマンドは、簡易ネットワーク管理プロトコル（SNMP）を使用して取得されたカウンタをクリアしません。**show interface** 特権 EXEC コマンドで表示されるカウンタのみをクリアします。

インターフェイス特性の設定例

インターフェイス範囲の設定：例

この例では、**interface range** グローバル コンフィギュレーション コマンドを使用して、スイッチ 1 上のポート 1 ～ 4 で速度を 100 Mb/s に設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# interface range gigabitethernet 1/0/1 - 4
スイッチ(config-if-range)# speed 100
```

この例では、カンマを使用して範囲に異なるインターフェイス タイプ ストリングを追加して、ギガビット イーサネット ポート 1 ～ 3 と、10 ギガビット イーサネット ポート 1 および 2 の両方をイネーブルにし、フロー制御ポーズ フレームを受信できるようにします。

```
スイッチ# configure terminal
スイッチ(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/1/1 - 2
スイッチ(config-if-range)# flowcontrol receive on
```

インターフェイス レンジ モードで複数のコンフィギュレーション コマンドを入力した場合、各コマンドは入力した時点で実行されます。インターフェイス レンジモードを終了した後で、コマンドがバッチ処理されるわけではありません。コマンドの実行中にインターフェイス レンジ コンフィギュレーション モードを終了すると、一部のコマンドが範囲内のすべてのインターフェイスに対して実行されない場合もあります。コマンドプロンプトが再表示されるのを待ってから、インターフェイス範囲コンフィギュレーション モードを終了してください。

インターフェイス レンジ マクロの設定および使用方法：例

次に、*enet_list* という名前のインターフェイス範囲マクロを定義してスイッチ 1 上のポート 1 および 2 を含め、マクロ設定を確認する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# define interface-range enet_list gigabitethernet 1/1/1 - 2
スイッチ(config)# end
スイッチ# show running-config | include define
define interface-range enet_list gigabitethernet 1/1/1 - 2
```

次に、複数のタイプのインターフェイスを含む マクロ *macro1* を作成する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# define interface-range macro1 gigabitethernet1/1/1 - 2,
gigabitethernet1/1/5 - 7, tengigabitethernet1/1/1 -2
スイッチ(config)# end
```

次に、インターフェイス レンジ マクロ *enet_list* に対するインターフェイス レンジ コンフィギュレーション モードを開始する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# interface range macro enet_list
スイッチ(config-if-range)#
```

次に、インターフェイス レンジ マクロ *enet_list* を削除し、処理を確認する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# no define interface-range enet_list
スイッチ(config)# end
スイッチ# show run | include define
スイッチ#
```

インターフェイス速度およびデュプレックス モードの設定：例

次に、インターフェイス速度を 100 Mb/s に、10/100/1000 Mbps ポートのデュプレックス モードを半二重に設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# interface gigabitethernet 1/0/3
スイッチ(config-if)# speed 10
スイッチ(config-if)# duplex half
```

次に、10/100/1000 Mbps ポートで、インターフェイスの速度を 100 Mbps に設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# interface gigabitethernet 1/0/2
スイッチ(config-if)# speed 100
```

コンソール メディア タイプの設定：例

次に、USB コンソール メディア タイプをディセーブルにし、RJ-45 コンソール メディア タイプをイネーブルにする例を示します。

```
スイッチ# configure terminal
スイッチ(config)# line console 0
スイッチ(config-line)# media-type rj45
```

次に、前の設定を逆にして、ただちにすべての接続された USB コンソールをアクティブにする例を示します。

```
スイッチ# configure terminal
```

```
スイッチ(config)# line console 0
スイッチ(config-line)# no media-type rj45
```

USB 無活動タイムアウトの設定：例

次に、無活動タイムアウトを 30 分に設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# line console 0
スイッチ(config-line)# usb-inactivity-timeout 30
```

設定をディセーブルにするには、次のコマンドを使用します。

```
スイッチ# configure terminal
スイッチ(config)# line console 0
スイッチ(config-line)# no usb-inactivity-timeout
```

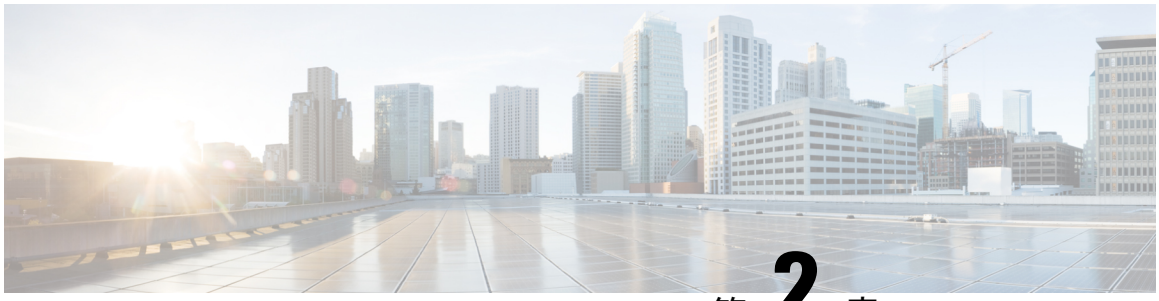
設定された分数の間に USB コンソール ポートで（入力）アクティビティがなかった場合、無活動タイムアウト設定が RJ-45 ポートに適用され、ログにこの発生が示されます。

```
*Mar  1 00:47:25.625: %USB_CONSOLE-6-INACTIVITY_DISABLE: Console media-type USB disabled
 due to inactivity, media-type reverted to RJ45.
```

この時点で、USB コンソール ポートを再度アクティブ化する唯一の方法は、ケーブルを取り外し、再接続することです。

スイッチのUSBケーブルが取り外され再接続された場合、ログは次のような表示になります。

```
*Mar  1 00:48:28.640: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```



第 2 章

Auto-MDIX の設定

Auto-MDIX の前提条件

デフォルトで Automatic Medium-Dependent Interface Crossover（Auto-MDIX）機能が有効に設定されます。

Auto-MDIX は、すべての 10/100/1000 Mbps インターフェイスと、10/100/1000BASE-TX Small Form-Factor Pluggable（SFP）モジュール インターフェイスでサポートされています。1000BASE-SX または 1000BASE-LX SFP モジュール インターフェイスではサポートされません。

Auto-MDIX の制約事項

受電デバイスがクロスケーブルでdeviceに接続されている場合、deviceは、IEEE 802.3afに完全には準拠していない、Cisco IP Phone やアクセスポイントなどの準規格の受電をサポートしていない場合があります。これは、スイッチ ポート上で Automatic Medium-Dependent Interface Crossover（Auto-MIDX）が有効かどうかは関係ありません。

Auto-MDIX の設定について

インターフェイスでの Auto-MDIX

自動メディア依存型インターフェイスクロスオーバー（MDIX）が有効になっているインターフェイスでは、必要なケーブル接続タイプ（ストレートまたはクロス）が自動的に検出され、

接続が適切に設定されます。Auto-MDIX機能を使用せずにdevicesを接続する場合、サーバー、ワークステーション、またはルータなどのデバイスの接続にはストレートケーブルを使用し、他のdevicesやリピータの接続にはクロスケーブルを使用する必要があります。Auto-MDIXが有効になっている場合、他のデバイスとの接続にはどちらのケーブルでも使用でき、ケーブルが正しくない場合はインターフェイスが自動的に修正を行います。ケーブル接続の詳細については、ハードウェア インストレーション ガイドを参照してください。

次の表に、Auto-MDIX の設定およびケーブル接続ごとのリンク ステートを示します。

表 *4:* リンク状態と *Auto-MDIX* の設定

ローカル側の Auto-MDIX	リモート側の Auto-MDIX	ケーブル接続が正しい場合	ケーブル接続が正しくない場合
オン	点灯	リンク アップ	リンク アップ
点灯	消灯	リンク アップ	リンク アップ
消灯	点灯	リンク アップ	リンク アップ
消灯	消灯	リンク アップ	リンク ダウン

Auto-MDIX の設定方法

インターフェイスでの Auto-MDIX の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **speed auto**
5. **duplex auto**
6. **end**
7. **copy running-config startup-config**

手順の詳細

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

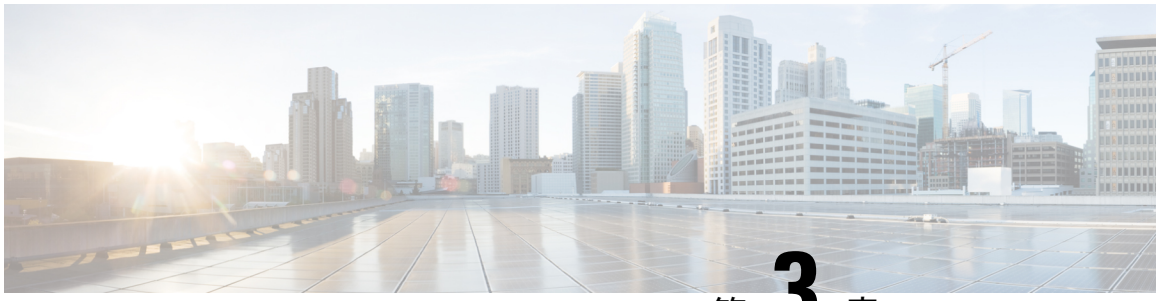
	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定する物理インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**speed auto** 例： スイッチ(config-if)# **speed auto**	接続されたデバイスと速度の自動ネゴシエーションを行うようにインターフェイスを設定します。
ステップ 5	**duplex auto** 例： スイッチ(config-if)# **duplex auto**	接続されたデバイスとデュプレックス モードの自動ネゴシエーションを行うようにインターフェイスを設定します。
ステップ 6	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

Auto-MDIX の設定例

次の例では、ポートの Auto MDIX を有効にする方法を示します。

```
スイッチ# configure terminal
スイッチ(config)# interface gigabitethernet 1/0/1
スイッチ(config-if)# speed auto
スイッチ(config-if)# duplex auto
スイッチ(config-if)# mdix auto
```

```
スイッチ(config-if)# end
```



第 3 章

LLDP、LLDP-MED、およびワイヤード ロケーション サービスの設定

LLDP、LLDP-MED、およびワイヤード ロケーション サービスについて

LLDP

Cisco Discovery Protocol（CDP）は、すべてのシスコ製デバイス（ルータ、ブリッジ、アクセス サーバ、スイッチ、およびコントローラ）のレイヤ 2（データ リンク層）上で動作するデバイス検出プロトコルです。ネットワーク管理アプリケーションは CDP を使用することにより、ネットワーク接続されている他のシスコ デバイスを自動的に検出し、識別できます。

device では他社製のデバイスをサポートし他のデバイス間の相互運用性を確保するために、IEEE 802.1AB リンク層検出プロトコル（LLDP）をサポートしています。LLDP は、ネットワーク デバイスがネットワーク上の他のデバイスに自分の情報をアドバタイズするために使用するネイバー探索プロトコルです。このプロトコルはデータリンク層で動作するため、異なるネットワーク層プロトコルが稼働する 2 つのシステムで互いの情報を学習できます。

LLDP でサポートされる TLV

LLDP は一連の属性をサポートし、これらを使用してネイバーデバイスを検出します。属性には、Type、Length、および Value の説明が含まれていて、これらを TLV と呼びます。LLDP をサポートするデバイスは、ネイバーとの情報の送受信に TLV を使用できます。このプロトコルは、設定情報、デバイス機能、およびデバイス ID などの詳細情報をアドバタイズできます。

スイッチは、次の基本管理 TLV をサポートします。これらは必須の LLDP TLV です。

- ポート記述 TLV
- システム名 TLV
- システム記述 TLV
- システム機能 TLV
- 管理アドレス TLV

次の IEEE 固有の LLDP TLV もアドバタイズに使用されて LLDP-MED をサポートします。

- ポート VLAN ID TLV（IEEE 802.1 に固有の TLV）
- MAC/PHY コンフィギュレーション/ステータス TLV（IEEE 802.3 に固有の TLV）

LLDP および Cisco Medianet

LLDP または CDP のロケーション情報をポート単位で設定すると、リモートデバイスから device に Cisco Medianet のロケーション情報を送信できます。

LLDP-MED

LLDP for Media Endpoint Devices（LLDP-MED）は LLDP の拡張版で、IP 電話などのエンドポイント デバイスとネットワーク デバイスの間で動作します。特に VoIP アプリケーションをサポートし、検出機能、ネットワーク ポリシー、Power over Ethernet（PoE）、インベントリ管理、およびロケーション情報に関する TLV を提供します。デフォルトで、すべての LLDP-MED TLV が有効になります。

LLDP-MED でサポートされる TLV

LLDP-MED では、次の TLV がサポートされます。

- LLDP-MED 機能 TLV

  LLDP-MED エンドポイントは、接続装置がサポートする機能と現在有効になっている機能を識別できます。

- ネットワーク ポリシー TLV

  ネットワーク接続デバイスとエンドポイントはともに、VLAN 設定、および関連するレイヤ 2 とレイヤ 3 属性をポート上の特定アプリケーションにアドバタイズできます。たとえば、スイッチは使用する VLAN 番号を IP 電話に通知できます。IP 電話は任意の device に接続し、VLAN 番号を取得してから、コール制御の通信を開始できます。

  ネットワーク ポリシー プロファイル TLV を定義することによって、VLAN、サービス クラス（CoS）、Diffserv コード ポイント（DSCP）、およびタギング モードの値を指定して、音声と音声信号のプロファイルを作成できます。その後、これらのプロファイル属性は、スイッチで中央集約的に保守され、IP 電話に伝播されます。

- 電源管理 TLV

  LLDP-MED エンドポイントとネットワーク接続デバイスの間で拡張電源管理を可能にします。devices および IP 電話は、デバイスの受電方法、電源プライオリティ、デバイスの消費電力などの電源情報を通知することができます。

  LLDP-MED は拡張電源 TLV もサポートして、きめ細かな電力要件、エンドポイント電源プライオリティ、およびエンドポイントとネットワークの接続デバイスの電源ステータスをアドバタイズします。LLDP が有効でポートに電力が供給されているときは、電力 TLV によってエンドポイントデバイスの実際の電力要件が決定するので、それに応じてシステムの電力バジェットを調整することができます。device は要求を処理し、現在の電力バジェットに基づいて電力を許可または拒否します。要求が許可されると、スイッチは電力バジェットを更新します。要求が拒否されると、device はポートへの電力供給をオフにし、Syslog メッセージを生成し、電力バジェットを更新します。LLDP-MED が無効になっている場合や、エンドポイントが LLDP-MED 電力 TLV をサポートしていない場合は、初期割り当て値が接続終了まで使用されます。

  power inline {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} インターフェイス コンフィギュレーション コマンドを入力して、電力設定を変更できます。PoE インターフェイスはデフォルトで **auto** モードに設定されています。値を指定しない場合は、最大電力（30 W）が許可されます。

- インベントリ管理 TLV

  エンドポイントは、device スイッチにエンドポイントの詳細なインベントリ情報を送信することが可能です。インベントリ情報には、ハードウェアリビジョン、ファームウェアバージョン、ソフトウェアバージョン、シリアル番号、メーカー名、モデル名、Asset ID TLV などがあります。

- ロケーション TLV

  deviceからのロケーション情報をエンドポイントデバイスに提供します。ロケーション TLV はこの情報を送信することができます。

  - 都市ロケーション情報

    都市アドレス情報および郵便番号情報を提供します。都市ロケーション情報の例には、地名、番地、郵便番号などがあります。

  - ELIN ロケーション情報

    発信側のロケーション情報を提供します。ロケーションは、緊急ロケーション識別番号（ELIN）によって決定されます。これは、緊急通報を Public Safety Answering Point（PSAP）にルーティングする電話番号で、PSAP はこれを使用して緊急通報者にコールバックすることができます。

ワイヤード ロケーション サービス

device は、接続されているデバイスのロケーション情報およびアタッチメント追跡情報を Cisco Mobility Services Engine（MSE）に送信するのにロケーション サービス機能を使用します。ト

ラッキングされたデバイスは、ワイヤレス エンドポイント、ワイヤード エンドポイント、またはワイヤード device やワイヤード コントローラになります。device は、MSE にネットワーク モビリティ サービス プロトコル（NMSP）のロケーション通知および接続通知を介して、デバイスのリンク アップ イベントおよびリンク ダウン イベントを通知します。

MSE が device に対して NMSP 接続を開始すると、サーバー ポートが開きます。MSE が device に接続する場合は、バージョンの互換性を確保する 1 組のメッセージ交換およびサービス交換情報があり、その後にロケーション情報の同期が続きます。接続後、 device は定期的にロケーション通知および接続通知を MSE に送信します。インターバル中に検出されたリンク アップ イベントまたはリンク ダウン イベントは、集約されてインターバルの最後に送信されます。

device がリンク アップ イベントまたはリンク ダウン イベントでデバイスの有無を確認した場合は、スイッチは、MAC アドレス、IP アドレス、およびユーザー名のようなクライアント固有情報を取得します。クライアントが LLDP-MED または CDP に対応している場合は、 device は LLDP-MED ロケーション TLV または CDP でシリアル番号および UDI を取得します。

デバイス機能に応じて、device は次のクライアント情報をリンク アップ時に取得します。

- ポート接続で指定されたスロットおよびポート。
- クライアント MAC アドレスで指定された MAC アドレス。
- ポート接続で指定された IP アドレス。
- 802.1X ユーザー名（該当する場合）。
- デバイス カテゴリは、*wired station* として指定されます。
- ステートは *new* として指定されます。
- シリアル番号、UDI。
- モデル番号。
- device による関連付け検出後の時間（秒）。

デバイス機能に応じて、 device は次のクライアント情報をリンク ダウン時に取得します。

- 切断されたスロットおよびポート。
- MAC アドレス
- IP アドレス
- 802.1X ユーザー名（該当する場合）。
- デバイス カテゴリは、*wired station* として指定されます。
- ステートは *delete* として指定されます。
- シリアル番号、UDI。
- device による関連付け検出後の時間（秒）。

device がシャットダウンする場合は、スイッチは、MSE との NMSP 接続を終了する前に、ステート *delete*および IP アドレスとともに接続情報通知を送信します。MSE は、この通知を、deviceに関連付けられているすべてのワイヤードクライアントに対する関連付け解除として解釈します。

device上のロケーション アドレスを変更すると、deviceは、影響を受けるポートを識別する NMSP ロケーション通知メッセージ、および変更されたアドレス情報を送信します。

デフォルトの LLDP 設定

表 *5:* デフォルトの *LLDP* 設定

機能	デフォルト設定
LLDP グローバル ステート	無効
LLDP ホールドタイム（廃棄までの時間）	120 秒
LLDP タイマー（パケット更新頻度）	30 秒
LLDP 再初期化遅延	2 秒
LLDP tlv-select	無効（すべての TLV との送受信）
LLDP インターフェイス ステート	無効
LLDP 受信	無効
LLDP 転送	無効
LLDP med-tlv-select	無効（すべてのLLDP-MEDTLVへの送信）。LLDP がグローバルに有効になると、LLDP-MED-TLV も有効になります。

LLDP に関する制約事項

- インターフェイスがトンネル ポートに設定されていると、LLDPは自動的に無効になります。
- 最初にインターフェイス上にネットワーク ポリシー プロファイルを設定した場合、インターフェイス上に **switchport voice vlan** コマンドを適用できません。**switchport voice vlan** *vlan-id* がすでに設定されているインターフェイスには、ネットワーク ポリシー プロファイルを適用できます。このように、そのインターフェイスには、音声または音声シグナリング VLAN ネットワーク ポリシー プロファイルが適用されます。
- ネットワークポリシープロファイルを持つインターフェイス上では、スタティック セキュア MAC アドレスを設定できません。

- Cisco Discovery Protocol と LLDP が両方とも同じスイッチ内で使用されている場合、Cisco Discovery Protocol が電源ネゴシエーションに使用されているインターフェイスで LLDP を無効にする必要があります。LLDP は、コマンド **no lldp tlv-select power-management** または **no lldp transmit / no lldp receive** を使用してインターフェイスレベルで無効にすることができます。

LLDP、LLDP-MED、およびワイヤード ロケーション サービスの設定方法

LLDP の有効化

手順の概要

1. **enable**
2. **configure terminal**
3. **lldp run**
4. **interface** *interface-id*
5. **lldp transmit**
6. **lldp receive**
7. **end**
8. **show lldp**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**lldp run** 例：	deviceでLLDPをグローバルにイネーブルにします。

	コマンドまたはアクション	目的
	スイッチ (config)# **lldp run**	
ステップ 4	**interface** *interface-id* 例： スイッチ (config)# **interface gigabitethernet 2/0/1**	LLDP を有効にするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**lldp transmit** 例： スイッチ(config-if)# **lldp transmit**	LLDP パケットを送信するようにインターフェイスを有効にします。
ステップ 6	**lldp receive** 例： スイッチ(config-if)# **lldp receive**	LLDP パケットを受信するようにインターフェイスを有効にします。
ステップ 7	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show lldp** 例： スイッチ# **show lldp**	設定を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

LLDP 特性の設定

LLDP 更新の頻度、情報を廃棄するまでの保持期間、および初期化遅延時間を設定できます。送受信する LLDP および LLDP-MED TLV も選択できます。

（注） ステップ 3 ～ 6 は任意であり、どの順番で実行してもかまいません。

手順の概要

1. **enable**
2. **configure terminal**
3. **lldp holdtime** *seconds*
4. **lldp reinit** *delay*
5. **lldp timer** *rate*
6. **lldp tlv-select**
7. **interface** *interface-id*
8. **lldp med-tlv-select**
9. **end**
10. **show lldp**
11. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**lldp holdtime** *seconds* 例： `スイッチ(config)# lldp holdtime 120`	（任意）デバイスから送信された情報を受信側デバイスが廃棄するまで保持する必要がある期間を指定します。 指定できる範囲は0～65535秒です。デフォルトは120 秒です。
ステップ 4	**lldp reinit** *delay* 例： `スイッチ(config)# lldp reinit 2`	（任意）任意のインターフェイス上でLLDPの初期化の遅延時間（秒）を指定します。 指定できる範囲は2～5秒です。デフォルトは2秒です。

	コマンドまたはアクション	目的
ステップ 5	**lldp timer** *rate* 例： `スイッチ(config)# lldp timer 30`	（任意）インターフェイス上でLLDPの更新の遅延時間（秒）を指定します。 指定できる範囲は5～65534秒です。デフォルトは30秒です。
ステップ 6	**lldp tlv-select** 例： `スイッチ(config)# tlv-select`	（任意）送受信する LLDP TLV を指定します。
ステップ 7	**interface** *interface-id* 例： `スイッチ (config)# interface gigabitethernet 2/0/1`	LLDPを有効にするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8	**lldp med-tlv-select** 例： `スイッチ (config-if)# lldp med-tlv-select inventory management`	（任意）送受信する LLDP-MED TLV を指定します。
ステップ 9	**end** 例： `スイッチ (config-if)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show lldp** 例： `スイッチ# show lldp`	設定を確認します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

LLDP-MED TLV の設定

デフォルトでは、device はエンドデバイスから LLDP-MED パケットを受信するまで、LLDP パケットだけを送信します。スイッチは、MED TLV を持つ LLDP も送信します。LLDP-MED エントリが期限切れになった場合は、スイッチは再び LLDP パケットだけを送信します。

lldp インターフェイス コンフィギュレーション コマンドを使用して、インターフェイスが次の表にリストされている TLV を送信しないように設定できます。

表 *6 : LLDP-MED TLV*

LLDP-MED TLV	説明
inventory-management	LLDP-MED インベントリ管理 TLV
location	LLDP-MED ロケーション TLV
network-policy	LLDP-MED ネットワーク ポリシー TLV
power-management	LLDP-MED 電源管理 TLV

インターフェイスで TLV を有効にするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lldp med-tlv-select**
5. **end**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： `スイッチ (config)# interface gigabitethernet 2/0/1`	LLDP を有効にするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**lldp med-tlv-select** 例： `スイッチ(config-if)# lldp med-tlv-select inventory management`	有効にする TLV を指定します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

Network-Policy TLV の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **network-policy profile** *profile number*
4. {**voice** | **voice-signaling**} **vlan** [*vlan-id* { **cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** { **cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**]
5. **exit**
6. **interface** *interface-id*
7. **network-policy** *profile number*
8. **lldp med-tlv-select network-policy**
9. **end**
10. **show network-policy profile**
11. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**network-policy profile** *profile number* 例： スイッチ(config)# **network-policy profile 1**	ネットワーク ポリシープロファイル番号を指定し、ネットワーク ポリシー コンフィギュレーション モードを開始します。指定できる範囲は 1 ～ 4294967295 です。
ステップ 4	{**voice** | **voice-signaling**} **vlan** [*vlan-id* { **cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** { **cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**] 例： スイッチ(config-network-policy)# **voice vlan 100 cos 4**	ポリシー属性の設定: • **voice**：音声アプリケーション タイプを指定します。 • **voice-signaling**：音声シグナリング アプリケーション タイプを指定します。 • **vlan**：音声トラフィックのネイティブ VLAN を指定します。 • *vlan-id*：（任意）音声トラフィックの VLAN を指定します。指定できる範囲は 1 ～ 4094 です。 • **cos** *cvalue*：（任意）設定された VLAN に対するレイヤ 2 プライオリティ サービス クラス（CoS）を指定します。指定できる範囲は 0 ～ 7 です。デフォルト値は 5 です。 • **dscp** *dvalue*：（任意）設定された VLAN に対する DiffServ コードポイント（DSCP）値を指定します。指定できる範囲は 0 ～ 63 です。デフォルト値は 46 です。 • **dot1p**：（任意）IEEE 802.1p プライオリティ タギングおよび VLAN 0（ネイティブ VLAN）を使用するように電話を設定します。

	コマンドまたはアクション	目的
		• **none**：（任意）音声 VLAN に関して IP Phone に指示しません。IP Phone のキー パッドから入力された設定を使用します。 • **untagged**：（任意）IP Phone を、タグなしの音声トラフィックを送信するよう設定します。これが IP Phone のデフォルト設定になります。 • **untagged**：（任意）IP Phone を、タグなしの音声トラフィックを送信するよう設定します。これが IP Phone のデフォルト設定になります。
ステップ 5	**exit** 例： `スイッチ(config)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**interface** *interface-id* 例： `スイッチ (config)# interface gigabitethernet 2/0/1`	ネットワーク ポリシー プロファイルを設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 7	**network-policy** *profile number* 例： `スイッチ(config-if)# network-policy 1`	ネットワーク ポリシー プロファイル番号を指定します。
ステップ 8	**lldp med-tlv-select network-policy** 例： `スイッチ(config-if)# lldp med-tlv-select network-policy`	ネットワーク ポリシー TLV を指定します。
ステップ 9	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show network-policy profile** 例：	設定を確認します。

	コマンドまたはアクション	目的
	スイッチ# **show network-policy profile**	
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ロケーション TLV およびワイヤード ロケーション サービスの設定

エンドポイントのロケーション情報を設定し、その設定をインターフェイスに適用するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **location** { **admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string* **identifier** *id* | **custom-location identifier** {*id* | **host**} | **geo-location identifier** {*id* | **host**}}
3. **exit**
4. **interface** *interface-id*
5. **location** { **additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} }
6. **end**
7. 次のいずれかを使用します。
   - **show location admin-tag** *string*
   - **show location civic-location identifier** *id*
   - **show location elin-location identifier** *id*
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**location** { **admin-tag** *string*	**civic-location identifier** {*id*	**host**}	**elin-location** *string* **identifier** *id*	**custom-location identifier** {*id*	**host**}	**geo-location identifier** {*id*	**host**}} 例：	エンドポイントにロケーション情報を指定します。 • **admin-tag**：管理タグまたはサイト情報を指定します。

	コマンドまたはアクション	目的
	スイッチ(config)# **location civic-location identifier 1** スイッチ(config-civic)# **number 3550** スイッチ(config-civic)# **primary-road-name "Cisco Way"** スイッチ(config-civic)# **city "San Jose"** スイッチ(config-civic)# **state CA** スイッチ(config-civic)# **building 19** スイッチ(config-civic)# **room C6** スイッチ(config-civic)# **county "Santa Clara"** スイッチ(config-civic)# **country US**	• **civic-location**：都市ロケーション情報を指定します。 • **elin-location**：緊急ロケーション情報（ELIN）を指定します。 • **custom-location**：カスタム ロケーション情報を指定します。 • **geo-location**：地理空間のロケーション情報を指定します。 • **identifier** *id*：都市、ELIN、カスタム、または地理ロケーションの ID を指定します。 • **host**：ホストの都市、カスタム、または地理ロケーションを指定します。 • *string*：サイト情報またはロケーション情報を英数字形式で指定します。
ステップ 3	**exit** 例： スイッチ(config-civic)# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 4	**interface** *interface-id* 例： スイッチ (config)# **interface gigabitethernet2/0/1**	ロケーション情報を設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**location** { **additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} } 例： スイッチ(config-if)# **location elin-location-id 1**	インターフェイスのロケーション情報を入力します。 • **additional-location-information**：ロケーションまたは場所に関する追加情報を指定します。 • **civic-location-id**：インターフェイスにグローバル都市ロケーション情報を指定します。 • **elin-location-id**：インターフェイスに緊急ロケーション情報を指定します。 • **custom-location-id**：インターフェイスにカスタム ロケーション情報を指定します。

	コマンドまたはアクション	目的
		• **geo-location-id**：インターフェイスに地理空間のロケーション情報を指定します。 • **host**：ホストのロケーション ID を指定します。 • *word*：追加のロケーション情報を指定する語またはフレーズを指定します。 • *id*：都市、ELIN、カスタム、または地理ロケーションの ID を指定します。指定できる ID 範囲は 1 ～ 4095 です。
ステップ 6	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 7	次のいずれかを使用します。 • **show location admin-tag** *string* • **show location civic-location identifier** *id* • **show location elin-location identifier** *id* 例： `スイッチ# show location admin-tag` または `スイッチ# show location civic-location identifier` または `スイッチ# show location elin-location identifier`	設定を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

での有線ロケーション サービスのイネーブル化 デバイス

始める前に

ワイヤードロケーションが機能するためには、まず、**ip device tracking** グローバル コンフィギュレーション コマンドを入力する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **nmsp notification interval** {**attachment** | **location**} *interval-seconds*
4. **end**
5. **show network-policy profile**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**nmsp notification interval** {**attachment** | **location**} *interval-seconds* 例： `スイッチ(config)# nmsp notification interval location 10`	NMSP 通知間隔を指定します。 **attachment**：接続通知間隔を指定します。 **location**：ロケーション通知間隔を指定します。 *interval-seconds*：deviceから MSE にロケーション更新または接続更新が送信されるまでの期間（秒）。指定できる範囲は 1 ～ 30 です。デフォルト値は 30 です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

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

LLDP、LLDP-MED、およびワイヤード ロケーション サービスの設定例

Network-Policy TLV の設定：例

次に、CoS を持つ音声アプリケーションの VLAN 100 を設定して、インターフェイス上のネットワーク ポリシー プロファイルおよびネットワーク ポリシー TLV を有効にする例を示します。

```
Switch# configure terminal
Switch(config)# network-policy 1
Switch(config-network-policy)# voice vlan 100 cos 4
Switch(config-network-policy)# exit
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# network-policy profile 1
Switch(config-if)# lldp med-tlv-select network-policy
```

次の例では、プライオリティ タギングを持つネイティブ VLAN 用の音声アプリケーション タイプを設定する方法を示します。

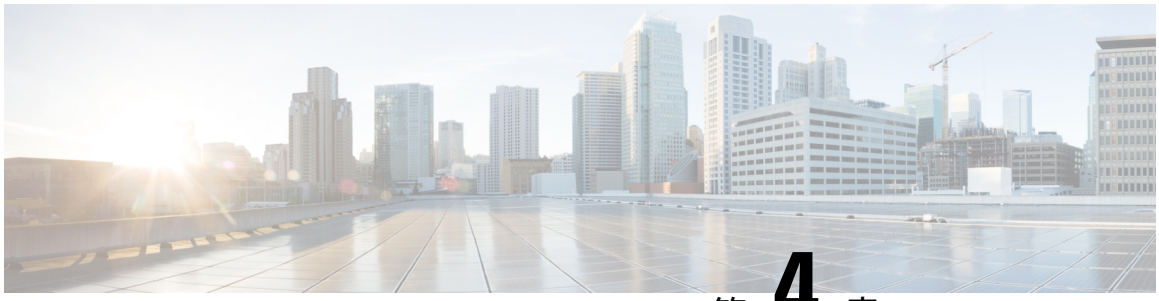
```
Switchconfig-network-policy)# voice vlan dot1p cos 4
Switchconfig-network-policy)# voice vlan dot1p dscp 34
```

LLDP、LLDP-MED、ワイヤード ロケーション サービスのモニタリングとメンテナンス

以下は、LLDP、LLDP-MED、ワイヤードロケーションサービスのモニタリングとメンテナンスのコマンドです。

コマンド	説明
clear lldp counters	トラフィック カウンタを0にリセットします。
clear lldp table	LLDP ネイバー情報テーブルを削除します。
clear nmsp statistics	NMSP 統計カウンタをクリアします。
show lldp	送信頻度、送信するパケットのホールドタイム、LLDP 初期化の遅延時間のような、インターフェイス上のグローバル情報を表示します。
show lldp entry *entry-name*	特定のネイバーに関する情報を表示します。 アスタリスク（*）を入力すると、すべてのネイバーの表示、またはネイバーの名前の入力が可能です。
show lldp interface [*interface-id*]	LLDPが有効になっているインターフェイスに関する情報を表示します。 表示対象を特定のインターフェイスに限定できます。
show lldp neighbors [*interface-id*] [**detail**]	デバイス タイプ、インターフェイスのタイプや番号、ホールドタイム設定、機能、ポート ID など、ネイバーに関する情報を表示します。 特定のインターフェイスに関するネイバー情報だけを表示したり、詳細表示にするため表示内容を拡張したりできます。
show lldp traffic	送受信パケットの数、廃棄したパケットの数、認識できない TLV の数など、LLDP カウンタを表示します。
show location admin-tag *string*	指定した管理タグまたはサイトのロケーション情報を表示します。

コマンド	説明
show location civic-location identifier *id*	特定のグローバル都市ロケーションのロケーション情報を表示します。
show location elin-location identifier *id*	緊急ロケーションのロケーション情報を表示します。
show network-policy profile	設定されたネットワークポリシー プロファイルを表示します。
show nmsp	NMSP 情報を表示します。



第 4 章

WS-C3560CX-8PD-S でのマルチギガビットポートの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

マルチギガビットポートの概要

シスコのマルチギガビット イーサネット テクノロジーにより、デバイスで 802.11ac Wave 2 の速度を活用できます。Cisco IOS XE 3.7.E1 および IOS 15.2(3)E1 以降、WS-C3560CX-8XPD-S モジュールをスイッチポートで複数の速度を自動ネゴシエートするように設定できます。サポートされる速度は、カテゴリ 5e ケーブルでは 100 Mbps、1 Gbps、2.5 Gbps、および 5 Gbps、カテゴリ 6 およびカテゴリ 6a ケーブルでは最大 10 Gbps です。

Cisco IOS XE 3.9.E1 および IOS 15.2(5)E1 以降、デフォルトでは、マルチギガビットポートのインターフェイス速度がダウンシフトされます。インターフェイスが高速リンクを確立できない

場合、ラインレートは自動的にダウンシフトされるか、速度が下げられます。インターフェイスは、次に利用可能な低速にダウンシフトする前に、現在の速度を使用してリンクの再確立を最大 4 回試行します。マルチギガビット インターフェイスでダウンシフトをサポートするには、リンクの両側でインターフェイス速度を **auto** に設定する必要があります。

WS-C3560CX-8XPD-S モジュールには 8 つのポートがあり、そのうち 6 つのポートは 1 ギガビットイーサネットポートで、2つのポートはマルチギガビットポートです。このモジュールには、2 つの SFP+ ポートもあります。

マルチギガビットポートの制約事項

次の制約事項が適用されます。

- マルチギガビットポートは、10Mbps の速度をサポートしていません。
- マルチギガビットポートは、半二重モードをサポートしていません。
- マルチギガビットポートは、EEE をサポートしていません。
- マルチギガビットポートは、リンクの両側でインターフェイス速度が **auto** に設定されている場合にのみダウンシフトをサポートします。

サポートされるケーブルタイプと最大長

次の表に、マルチギガビットポートでサポートされるケーブルのタイプと最大長を示します。

ケーブル タイプ	100M	1G	2.5G	5G	10G
カテゴリ 5e	対応	対応	対応	対応	使用不可（Not Available）
カテゴリ 6	対応	対応	対応	対応	対応（55 m）
カテゴリ 6a	対応	対応	対応	対応	対応

インターフェイス速度の設定

マルチギガビット イーサネット インターフェイス（1000Base-T ポート）でポート速度を 100Mbps/1000Mbps/2500Mbps/5000Mbps/10000Mbps に設定するには、次の作業を行います。

（注） WS-C3560CX-8XPD-S モジュールの 2 つのポートのみがマルチギガビット イーサネットをサポートします。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface tengigabitethernet***slot/interface*
4. **speed** [*100* | *1000* | *2500* | *5000* | *10000* | **auto** [*100* | *1000* | *2500* | *5000* | *10000*]]
5. **[no] downshift disable**
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface tengigabitethernet***slot/interface* 例： `スイッチ(config)# interface tengigabitethernet 1/0/2`	設定するインターフェイスを指定します。
ステップ 4	**speed** [*100* | *1000* | *2500* | *5000* | *10000* | **auto** [*100* | *1000* | *2500* | *5000* | *10000*]] 例： `スイッチ (config-if)# speed 5000`	インターフェイスの速度を設定します。 （注） 10G の速度は、カテゴリ 6 およびカテゴリ 6a ケーブルでのみサポートされます。
ステップ 5	**[no] downshift disable** 例： `スイッチ(config-if)# no downshift disable`	デフォルトでは、ダウンシフトはマルチギガビットポートでイネーブルになります。**downshift disable** コマンドにより、指定したインターフェイス上でダウンシフトがディセーブルになります。**no downshift disable** コマンドにより、インターフェイス上でダウンシフトがイネーブルになります。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

次のタスク

自動ネゴシエーション（デフォルト設定）に戻すには、インターフェイス コンフィギュレーション モードで **no speed** コマンドを入力します。

例：インターフェイス速度の設定

次に、マルチギガビット イーサネット インターフェイス 1/0/2 のインターフェイス速度を 5G に設定する例を示します。

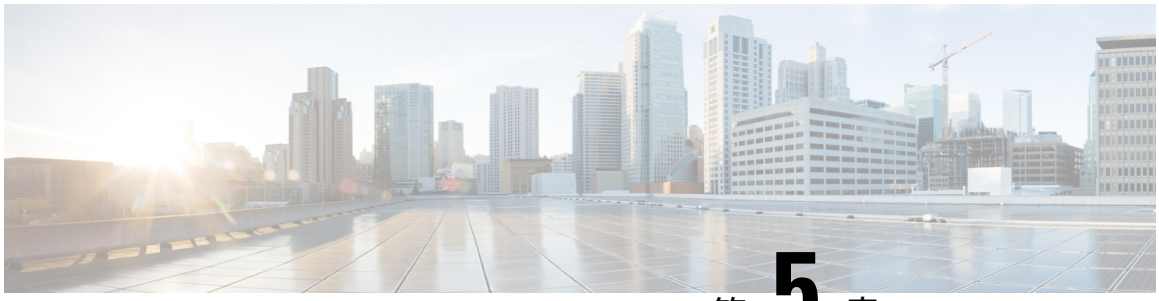
```
Switch(config)# interface tengigabitethernet 1/0/2
Switch (config-if)# speed 5000
```

次に、マルチギガビット イーサネット インターフェイス 1/0/2 が速度とデュプレックスモードを自動ネゴシエーションする例を示します。

```
Switch(config)# interface tengigabitethernet 1/0/2
Switch(config-if)# speed auto
```

次に、マルチギガビット イーサネット インターフェイス 1/0/1 の速度ネゴシエーションを 2.5G に制限する例を示します。

```
Switch(config)# interface tengigabitethernet 1/0/1
Switch(config-if)# speed auto 2500
```



第 5 章

システム MTU の設定

MTU について

システム MTU のガイドライン

システム MTU 値を設定する場合、次の注意事項に留意してください。

- すべてのインターフェイスで送受信されるフレームのデフォルト最大伝送単位（MTU）サイズは、1500 バイトです。10 または 100 Mbps で動作するすべてのインターフェイスで MTU サイズを増やすには、**system mtu** グローバル コンフィギュレーション コマンドを使用します。また、**system mtu jumbo** グローバル コンフィギュレーション コマンドを使用すると、すべてのギガビット イーサネット インターフェイス上でジャンボ フレームをサポートするように MTU サイズを増やすことができます。
- **system mtu** コマンドはギガビット イーサネット ポートには影響せず、**system mtu jumbo** コマンドは 10/100 ポートには影響しません。**system mtu jumbo** コマンドを設定していない場合、**system mtu** コマンドの設定はすべてのギガビット イーサネット インターフェイスに適用されます。

MTU の設定方法

システム MTU の設定

10/100 インターフェイスまたはギガビット イーサネット インターフェイスすべての MTU サイズを変更するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **system mtu** *bytes*
3. **system mtu jumbo** *bytes*
4. **end**
5. **copy running-config startup-config**
6. **show system mtu**

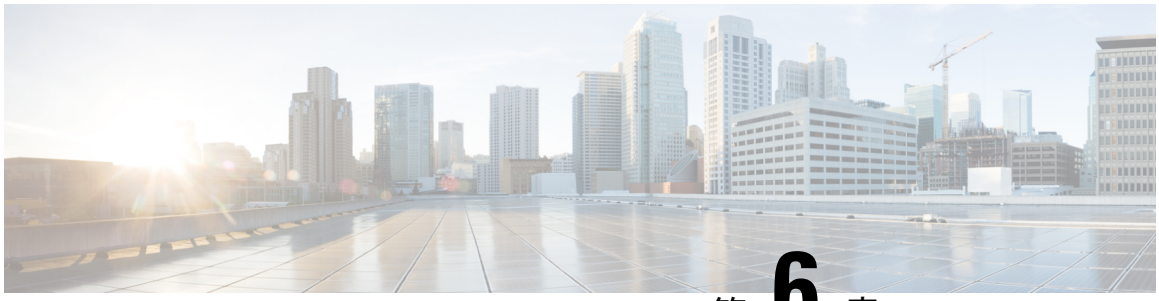
手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**system mtu** *bytes* 例： `スイッチ(config)# system mtu 2500`	指定できる範囲は、1500～9198バイトです。デフォルトは 1500 バイトです。
ステップ 3	**system mtu jumbo** *bytes* 例： `スイッチ(config)# system mtu jumbo7500`	指定できる範囲は、1500～9198バイトです。デフォルトは 1500 バイトです。
ステップ 4	**end** 例： `スイッチ(config)#  end`	特権 EXEC モードに戻ります。
ステップ 5	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	コンフィギュレーション ファイルに設定を保存します。
ステップ 6	**show system mtu** 例： `スイッチ# show system mtu`	設定を確認します。

システム MTU の設定例

次に、ギガビット イーサネット ポートの最大パケットサイズを バイトに設定する例を示します。

```
スイッチ(config)#
スイッチ(config)# exit
```



第 6 章

ブート ファストの設定

スイッチでのブート ファストの設定

この機能は有効になっている場合、スイッチを迅速に起動するために役立ちます。限定された範囲でメモリ テストが実行され、スイッチはファイル システム チェック（FSCK）と POST テストをスキップします。

（注） 高速ブートが有効になっている場合も、スイッチの起動後に、コマンド ライン インターフェイスから **diagnostic start** コマンドを使用することで、POST テストを実行できます。

ブート ファストの有効化

ブート ファスト機能を有効にするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **boot fast**
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**boot fast** 例： スイッチ(config)# **boot fast**	ブート ファスト機能を有効にします。 限定された範囲でメモリ テストが実行され、ファイル システム チェック（FSCK）と POST テストをスキップします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

ブート ファストの無効化

ブート ファスト機能を無効にするには、次の手順を実行します。

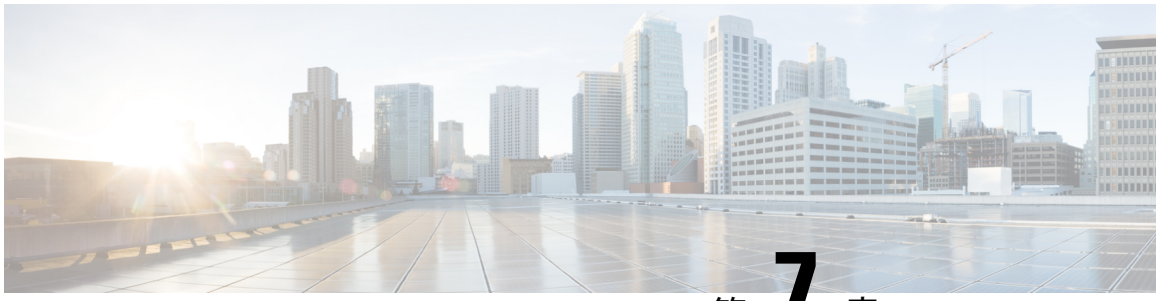
手順の概要

1. **enable**
2. **configure terminal**
3. **no boot fast**
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**no boot fast** 例： スイッチ(config)# **no boot fast**	ブート ファスト機能を無効にします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。



第 7 章

Power over Ethernet の設定

PoE について

Power over Ethernet（PoE）ポート

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

- シスコ先行標準受電装置（Cisco IP Phone および Cisco Aironet アクセス ポイントなど）
- IEEE 802.3af 準拠の受電装置

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

Catalyst WS-C3560CX-8PT-S の PoE および PoE パススルー ポート

Catalyst WS-C3560CX-8PT-S は PD/PSE 製品です。つまり、スイッチは電源デバイス（PD）としても、また電源装置（PSE）としても動作できます。このスイッチは、アップリンク ポート（PD1 または PD2）から供給される PoE 電圧、あるいは外部予備電源（AUX）から供給される電圧によって電源が投入されます。このスイッチでは、AC および DC 入力に加え、PoE、PoE+、UPOE で電源を供給できます。

アップリンクから供給される電力と電源アダプタから供給される電力は、より高いPoEバジェットに変換されて、入力電源に追加されます。この電力の一部は、システム電源に使用され、残りは IP 電話、IP カメラなどのその他の PoE 周辺機器に電力を供給できるパススルー電力としてダウンリンク POE+ ポートに供給されます。

- Catalyst WS-C3560CX-8PT-S は 2xUPOE アップリンクからの電源をサポートします。
- これは、スイッチで 24V DC 入力による電源投入を可能にする DC 電源アダプタをサポートします。
- AUX はシステムに 78W を供給します。
- 電源（AC または DC）および PoE は追加できます。次の表に、PoE バジェットのいくつかの電力値を示します。

表 *7: PoE* バジェット

PoE バジェット（ワット）	アップリンク 1	アップリンク 2	コメント
0	PoE	PoE	通常の動作、PoE バジェットなし
0	0	PoE+	通常の動作、PoE バジェットなし
20	PoE+	PoE+	PoE バジェット使用可能
22	0	UPoE	PoE バジェット使用可能
33	UPoE	PoE	PoE バジェット使用可能
44	PoE+	UPoE	PoE バジェット使用可能
68	UPoE	UPoE	PoE バジェット使用可能

このスイッチでは、T1 電力で起動し、T2 電力にネゴシエートすることが予想されます（これは低電力起動とも呼ばれます）。低電力起動は次のような場合に起こります。

- アップリンク ポートの 1 つが PSE に接続されている。
- 予備電源アダプタが接続されていない。

この場合、スイッチは低電力モードで電源が投入され、ASIC の電源は切られます。また、CDP/LLDP を使用して電源をネゴシエートします。電源がネゴシエートされると、システムに電源が投入され、ASIC が初期化されます。また、ソフトウェアをリロードせずに起動し続けます。

例：WS-C3560CX-8PT-S での PoE および PoE パススルー ポートの設定

show env power 特権 EXEC コマンドは、スイッチの電源オプションに関する情報を提供します。

```
スイッチ# show env power

Power Source   Type           Power(w)  Mode
-------------- -------------- --------- ---------
A.C. Input     Auxilliary     80(w)     Available
Gi0/9          Type2          30(w)     Available
Gi0/10         Type2          30(w)     Available

Available : The PoE received on this link is used for powering this switch and
            providing PoE pass-through if applicable.
```

（注） これらの電源装置はすべて、PoEバジェットまで追加されます。システム消費量は約24Wです。

サポート対象のプロトコルおよび標準規格

device は PoE のサポートに次のプロトコルと規格を使用します。

- 電力の消費について CDP を使用：受電デバイスは、device に消費している電力量を通知します。device はこの電力消費に関するメッセージに応答しません。device は、PoE ポートに電力を供給するか、このポートへの電力を取り除くだけです。
- シスコ インテリジェント電力管理：受電装置およびdeviceは、電力ネゴシエーション CDP メッセージによって電力消費レベルについてネゴシエーションを行います。このネゴシエーションにより、7 W より多くを消費する高電力のシスコ受電デバイスは、最も高い電力モードで動作できるようになります。受電デバイスは、最初に低電力モードでブートして 7 W 未満の電力を消費し、ネゴシエーションを行って高電力モードで動作するための十分な電力を取得します。受電装置が高電力モードに切り替わるのは、device から確認を受信した場合に限られます。

  高電力装置は、電力ネゴシエーション CDP をサポートしない devices で低電力モードで動作できます。

  シスコのインテリジェントな電力管理の機能には、電力消費に関して CDP との下位互換性があるため、device は、受信する CDP メッセージに従って応答します。CDP はサードパーティの受電デバイスをサポートしません。このため、device は、IEEE 分類を使用して装置の消費電力を判断します。
- IEEE 802.3a：この規格の主な機能は、受電装置の検出、電力の管理、切断の検出です。オプションとして受電装置の電力分類があります。詳細については、この規格を参照してください。

受電デバイスの検出と初期電力割り当て

deviceは、PoE対応ポートがシャットダウンの状態でなく、PoEはイネーブルになっていて（デフォルト）、接続した装置はACアダプタから電力供給されていない場合、シスコの先行標準受電デバイスまたはIEEE準拠の受電デバイスを検出します。

装置の検出後、deviceは、次のように装置のタイプに応じて電力要件を判断します。

- 初期電力割り当ては、受電デバイスが要求する最大電力量です。deviceは、受電デバイスを検出および電力供給する場合、この電力を最初に割り当てます。deviceが受電デバイスからCDPメッセージを受信し、受電デバイスがCDP電力ネゴシエーションメッセージを通じてdeviceと電力レベルをネゴシエートしたときに、初期電力割り当てが調整される場合があります。
- deviceは検出したIEEE装置を消費電力クラス内で分類します。deviceは、電力バジェットに使用可能な電力量に基づいて、ポートに通電できるかどうかを決定します。表8:IEEE電力分類 （66ページ） に、各種レベルの一覧を示します。

表 *8: IEEE* 電力分類

クラス	から要求される最大電力レベルデバイス
0（クラス ステータスは不明）	15.4 W
1	4 W
2	7 W
3	15.4 W

deviceは電力要求をモニタリングおよび追跡して必要な場合にだけ電力供給を許可します。deviceは自身の電力バジェット（PoEのdeviceで使用可能な電力量）を追跡します。電力の供給許可または拒否がポートで行われると、deviceはパワーアカウンティング計算を実行し、電力バジェットを最新に保ちます。

電力がポートに適用されたあとで、deviceはCDPを使用して、接続されたシスコ受電デバイスの *CDP* 固有の電力消費要件を調べます。この要件は、CDPメッセージに基づいて割り当てられる電力量です。これに従って、deviceは電力バジェットを調整します。これは、サードパーティのPoE装置には適用されません。deviceは要件を処理して電力の供給または拒否を行います。要求が許可されると、deviceは電力バジェットを更新します。要求が拒否された場合は、deviceはポートの電力がオフに切り替わっていることを確認し、syslogメッセージを生成してLEDを更新します。受電デバイスはより多くの電力について、deviceとのネゴシエーションを行うこともできます。

不足電圧、過電圧、オシレータ障害、または短絡状態による障害をdeviceが検出した場合、ポートへの電源をオフにし、syslogメッセージを生成し、電力バジェットとLEDを更新します。

電力管理モード

deviceでは、次の PoE モードがサポートされます。

- **auto**：接続されている装置で電力が必要であるかどうか、device が自動的に検出します。ポートに接続されている受電デバイスを device が検出し、device に十分な電力がある場合は、電力を供給して電力バジェットを更新し、先着順でポートの電力をオンに切り替えて LED を更新します。LED の詳細については、ハードウェア インストレーション ガイドを参照してください。

  すべての受電デバイス用として device に十分な電力がある場合は、すべての受電デバイスが起動します。device に接続された受電デバイスすべてに対し十分な電力が利用できる場合、すべての装置に電力を供給します。使用可能な PoE がない場合、または他の装置が電力供給を待機している間に装置の接続が切断されて再接続した場合、どの装置へ電力を供給または拒否されるかが判断できなくなります。

  許可された電力がシステムの電力バジェットを超えている場合、device は電力を拒否し、ポートへの電力がオフになっていることを確認したうえで syslog メッセージを生成し、LED を更新します。電力供給が拒否された後、device は定期的に電力バジェットを再確認し、継続して電力要求の許可を試みます。

  device により電力を供給されている装置が、さらに壁面コンセントに接続している場合、device は装置に電力を供給し続ける場合があります。このとき、装置が device から受電しているか、AC 電源から受電しているかにかかわらず、device は引き続き装置へ電力を供給していることを報告し続ける場合があります。

  受電デバイスが取り外された場合、device は切断を自動的に検出し、ポートから電力を取り除きます。非受電デバイスを接続しても、そのデバイスに障害は発生しません。

  ポートで許可される最大ワット数を指定できます。受電デバイスの IEEE クラス最大ワット数が設定されている最大値より大きい場合、device はそのポートに電力を供給しません。device が受電デバイスに電力供給したが、受電デバイスが設定の最大値より多くの電力を CDP メッセージによって後で要求した場合、device はポートの電力を取り除きます。その受電デバイスに割り当てられていた電力は、グローバル電力バジェットに送られます。ワット数を指定しない場合、device は最大値の電力を供給します。任意の PoE ポートで **auto** 設定を使用してください。auto モードがデフォルト設定です。

- **static**：device は、受電装置が接続されていなくてもポートに電力をあらかじめ割り当て、そのポートで電力が使用できるようにします。device は、設定された最大ワット数をポートに割り当てます。その値は、IEEE クラスまたは受電デバイスからの CDP メッセージによって調節されることはありません。これは、電力があらかじめ割り当てられていることから、最大ワット数以下の電力を使用するすべての受電デバイスが固定ポートに接続されている場合に電力が保証されるためです。ポートはもう先着順方式ではなくなります。

  ただし、受電装置の IEEE クラスが最大ワット数を超えると、device は装置に電力を供給しません。受電 device が最大ワット数を超える電力を消費していることを CDP メッセージによって知ると、device は受電デバイスをシャットダウンします。

ワット数を指定しない場合、device は最大数をあらかじめ割り当てます。device は、受電デバイスを検出した場合に限り、ポートに電力を供給します。優先順位が高いインターフェイスには、**static** 設定を使用してください。

- **never**：device は受電装置の検出をディセーブルにして、電力が供給されていない装置が接続されても、PoE ポートに電力を供給しません。PoE 対応ポートに電力を絶対に適用せず、そのポートをデータ専用ポートにする場合に限り、このモードを使用してください。

ほとんどの場合、デフォルトの設定（自動モード）の動作は適切に行われ、プラグアンドプレイ動作が提供されます。それ以上の設定は必要ありません。ただし、優先順位の高いPoEポートを設定したり、PoEポートをデータ専用にしたり、最大ワット数を指定して高電力受電デバイスをポートで禁止したりする場合は、このタスクを実行します。

電力モニタリングおよび電力ポリシング

リアルタイム電力消費のポリシングをイネーブルにした場合、受電デバイスが最大割り当て量（カットオフ電力値）を超えて電力を消費すると、device はアクションを開始します。

PoEがイネーブルである場合、deviceは受電デバイスのリアルタイムの電力消費を検知します。接続されている受電デバイスのリアルタイム電力消費を device が監視することを、電力モニターリングまたは電力検知といいます。また、deviceはパワーポリシング機能を使用して消費電力をポリシングします。

電力モニタリングは、シスコのインテリジェントな電力管理および CDP ベースの消費電力に対して下位互換性があります。電力モニタリングはこれらの機能とともに動作して、PoEポートが受電デバイスに電力を供給できるようにします。

device は次のようにして、接続されている装置のリアルタイム電力消費を検知します。

1. device は、個々のポートでリアルタイム消費電力をモニターリングします。
2. device は、ピーク時の電力消費を含め、電力消費を記録します。device は CISCO-POWER-ETHERNET-EXT-MIB を介して情報を報告します。
3. 電力ポリシングがイネーブルの場合、device はリアルタイムの消費電力を装置に割り当てられた最大電力と比較して、消費電力をポリシングします。最大消費電力は、PoE ポートでカットオフ電力とも呼ばれます。

   装置がポートで最大電力割り当てを超える電力を使用すると、device はポートへの電力をオフにしたり、または device コンフィギュレーションに基づいて受電装置に電力を供給しながら device が syslog メッセージを生成して LED（ポート LED はオレンジ色で点滅）を更新したりすることができます。デフォルトでは、すべてのPoEポートで消費電力のポリシングはディセーブルになっています。

   PoE の error-disabled ステートからのエラー回復がイネーブルの場合、指定の時間の経過後、device は PoE ポートを error-disabled ステートから自動的に回復させます。

   エラー回復が無効な場合、**shutdown** および **no shutdown** インターフェイス コンフィギュレーション コマンドを使用して、手動で PoE ポートをイネーブルにできます。
4. ポリシングが無効である場合、受電デバイスがPoEポートに割り当てられた最大電力より多くの量を消費しても対処されないため、device に悪影響を与える場合があります。

PoE ポートでの最大電力割り当て（カットオフ電力）

電力ポリシングがイネーブルの場合、deviceは次の順序でいずれかの値をPoEポートでのカットオフ電力とします。

1. deviceがポートに対して予定しているユーザー定義電力レベルを設定している場合は、**power inline consumption default** *wattage* グローバル コンフィギュレーション コマンドまたはインターフェイス コンフィギュレーション コマンドを使用して手動で行う。
2. ポート上で許可される電力を制限するユーザー定義の電力レベルを設定している場合は、**power inline auto max** *max-wattage* インターフェイス コンフィギュレーション コマンドまたは **power inline static max** *max-wattage* インターフェイス コンフィギュレーション コマンドを使用して手動で行う。
3. deviceにおいて受電装置の電力消費が設定されている場合は、CDP 電力ネゴシエーションまたは IEEE 分類と LLDP 電力ネゴシエーションを使用して自動的に行われる。

power inline consumption default *wattage* または **power inline** [**auto** | **static max**] *max-wattage* コマンドを入力することにより、カットオフ電力値を手動で設定するには、リストの1番めまたは2番めの方法を使用します。

CDP/LLDP電力ネゴシエーションがサポートされていない状況でのみ、ポートの電力レベルを手動で設定するには、**power inline consumption default** *wattage* コマンドを使用する必要があります。

カットオフ電力量の値を手動で設定しない場合、deviceは、CDP電力ネゴシエーションまたはデバイスのIEEE分類とLLDP電力ネゴシエーションを使用して自動的に値を決定します。CDPまたはLLDPがイネーブルでない場合は、デフォルト値の30Wが適用されます。ただし、CDPまたはLLDPがない場合は、15400 ～ 30000 mW の値が CDP 要求または LLDP 要求だけに基づいて割り当てられるため、装置で 15.4 W を超える電力の消費がdevice から許可されません。受電デバイスが CDP または LLDP のネゴシエーションなしに 15.4 W を超える電力を消費する場合、デバイスは最大電流（*Imax*）の制限に違反し、最大値を超える電流が供給されるという *Icut* 障害が発生する可能性があります。再び電源を入れるまで、ポートは障害状態のままになります。ポートで継続的に 15.4 W を超える電力が給電される場合、このサイクルが繰り返されます。

（注） PoE+ ポートに接続されている受電デバイスが再起動し、電力 TLV で CDP パケットまたは LLDP パケットが送信される場合、device は最初のパケットの電力ネゴシエーション プロトコルをロックし、その他のプロトコルからの電力要求に応答しません。たとえば、device が CDP にロックされている場合、LLDP 要求を送信する装置に電力を供給しません。deviceがCDPにロックされた後でCDPがディセーブルになった場合、deviceはLLDP電源要求に応答せず、アクセサリの電源がオンにならなくなります。この場合、受電デバイスを再起動する必要があります。

電力消費値

ポートの初期電力割り当ておよび最大電力割り当てを設定することができます。ただし、これらの値は、deviceがPoEポートの電力をオンまたはオフにするときを指定するために設定する値です。最大電力割り当ては、受電デバイスの実際の電力消費と同じではありません。deviceによって電力ポリシングに使用される実際のカットオフ電力値は、設定済みの電力値と同等ではありません。

電力ポリシングがイネーブルの場合、deviceは、スイッチポートで、受電装置の消費電力を超える消費電力ポリシングを行います。最大電力割り当てを手動で設定する場合、スイッチポートと受電デバイス間のケーブルでの電力損失を考慮する必要があります。カットオフ電力とは、受電デバイスの定格消費電力とケーブル上での最悪時の電力損失を合計したものです。

deviceのPoEがイネーブルの場合、電力ポリシングをイネーブルにすることを推奨します。たとえば、ポリシングがディセーブルで、**power inline auto max 6300**インターフェイス コンフィギュレーション コマンドを使用してカットオフ値を設定すると、PoEポートに設定される最大電力割り当ては6.3 W（6300 mW）です。装置が最大で6.3 Wの電力を必要とする場合、deviceはポートに接続されている装置に電力を供給します。CDPによるパワー ネゴシエーション実施後の値またはIEEE分類値が設定済みカットオフ値を超えると、deviceは接続されている装置に電力を供給しなくなります。deviceがPoEポートで電力をオンにしたあと、deviceは受電装置のリアルタイム電力消費のポリシングを行わないので、受電装置は最大割り当て量を超えて電力を消費できることになり、deviceと、他のPoEポートに接続されている受電装置に悪影響を及ぼすことがあります。

deviceは内部電源装置およびCisco Redundant Power System 2300（RPS 2300）をサポートしており、受電デバイスが利用できる総電力量は電源装置の設定によって異なります。

PoEの設定方法

PoEポートの電力管理モードの設定

（注） PoE設定を変更するとき、設定中のポートでは電力が低下します。新しい設定、その他のPoEポートの状態、電力バジェットの状態により、そのポートの電力は再びアップしない場合があります。たとえば、ポート1が自動でオンの状態になっていて、そのポートを固定モードに設定するとします。deviceはポート1から電力が取り除き、受電デバイスを検出してポートに電力を再び供給します。ポート1が自動でオンの状態になっていて、最大ワット数を10 Wに設定した場合、deviceはポートから電力を取り除き、受電デバイスを再び検出します。deviceは、受電デバイスがクラス1、クラス2、またはシスコ専用受電デバイスのいずれかの場合に、ポートに電力を再び供給します。

手順の概要

1. **enable**

2. **configure terminal**
3. **interface** *interface-id*
4. **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]}
5. **end**
6. **show power inline** [*interface-id* | **module** *switch-number*]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/1`	設定する物理ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} 例： `スイッチ(config-if)# power inline auto`	ポートの PoE モードを設定します。キーワードの意味は次のとおりです。 • **auto**：受電デバイスの検出をイネーブルにします。十分な電力がある場合は、装置の検出後に PoE ポートに電力を自動的に割り当てます。これがデフォルト設定です。 • **max** *max-wattage*：ポートで許可されている電力を制限します。値を指定しない場合は、最大電力が供給されます。 • **never** ：デバイスの検出とポートへの電力供給をディセーブルにします。 （注） ポートにシスコの受電デバイスが接続されている場合は、**power inline never** コマンドでポートを設定しないでください。問題のあるリンクアップが発生し、ポートが error-disabled ステートになることがあります。

	コマンドまたはアクション	目的
		• **static**：受電デバイスの検出をイネーブルにします。deviceが受電デバイスを検出する前に、ポートへの電力を事前に割り当てます（確保します）。device は、装置が接続されていなくてもこのポートに電力を予約し、装置の検出時に電力が供給されることを保証します。 deviceは、自動モードに設定されたポートに電力を割り当てる前に、固定モードに設定されたポートに PoE を割り当てます。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show power inline** [*interface-id* | **module** *switch-number*] 例： `スイッチ# show power inline`	device、指定したインターフェイス.
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

Catalyst WS-C3560CX-8PT-S での PoE および PoE パススルー ポートの設定

他の PoE スイッチと同様に、Catalyst WS-C3560CX-8PT-S コンパクト スイッチ PoE ポートで電源管理、バジェット、およびポリシングを設定できます。

show env power 特権 EXEC コマンドは、スイッチの電源オプションに関する情報を提供します。

無停止型 POE

無停止型 POE は、PSE スイッチが起動している場合でも、接続された PD デバイスへの連続電源を提供します。

（注） ポートへの電源供給はMCUファームウェアのアップグレード時には中断され、ポートはアップグレード直後にバックアップされます。

（注） この機能は、Catalyst 3560-CX および Catalyst 2960-CX スイッチの次のモデルでのみ利用できます。

- WS-3560CX-8PC-S
- WS-3560CX-12PC-S
- WS-C3560CX-8XPD-S
- WS-C2960CX-8PC-L

高速 POE

この機能は、IOS が起動するのを待機することなく、AC 電源が接続された瞬間（電源投入の 15 ～ 20 秒以内）に特定の PSE ポートから引き出された最後の電力を記憶し、電源をオンにします。**poe-ha** が特定のポートで有効な場合、電源障害後の復旧時に、IOS 転送が開始されるまでの短期間、スイッチが接続されているエンドポイントデバイスに電源を供給します。

この機能は、すでに実装されている **poe-ha** と同じコマンドで設定できます。スイッチの電源がオフになったときにポートに接続されている電源デバイスをユーザが交換した場合、この新しいデバイスは、以前のデバイスが利用していた電力を取得します。

（注） 高速 POE は、Catalyst 3850 でのみサポートされています。

（注） UPOE の場合、高速 POE はスイッチ側で使用可能ですが、UPOE 電力の可用性の信号伝達を LLDP に依存するため、PD エンドポイントは同様の機能を利用できない可能性があります。LLDP に依存する場合、IOS が起動して LLDP パケット交換が可能になり、UPOE 電力の可用性を信号で伝達できるようになるまで、PD エンドポイントはそのまま待機する必要があります。

持続性および高速 POE の設定

持続性 POE および POE を設定するには、次の手順を実行します。

（注） PD を接続する前に **poe-ha** コマンドを設定する、または、**poe-ha** を設定した後にポートを手動で閉じる/開く必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline port poe-ha**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/1`	設定する物理ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**power inline port poe-ha** 例： `スイッチ(config-if)# power inline port poe-ha`	PoE の高可用性を設定します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

PoE ポートに接続された受電デバイスの電力バジェット

Cisco 受電装置が PoE ポートに接続されている場合、deviceは Cisco Discovery Protocol（CDP）を使用してデバイスのプロトコル固有の電力消費を判断し、それに応じてdeviceは電力バジェットを調整します。この機能は、IEEE サードパーティの受電デバイスには適用されません。この装置の場合、deviceが電力要求を許可したときに、受電装置の IEEE 分類に応じてdeviceが電力バジェットを調整します。受電デバイスがクラス 0（クラスステータス不明）またはクラス

3 の場合、deviceは CDP 固有の電力所要量に関係なく、受電デバイスに 15,400 mW を計上します。受電デバイスがCDP固有の消費よりも高いクラスを報告してきたり、または電力分類（デフォルトはクラス 0）をサポートしていない場合、deviceはIEEE クラス情報を使用してグローバル電力バジェットを追跡するため、電力供給できるデバイスが少なくなります。

power inline consumption *wattage* インターフェイス コンフィギュレーション コマンドまたは **power inline consumption default** *wattage* グローバル コンフィギュレーション コマンドを使用すれば、IEEE 分類で指定されたデフォルトの電力要件を上書きできます。IEEE 分類で指定された電力と実際にデバイスが必要とする電力の差は、追加のデバイスが使用するためグローバル電力バジェットに入れられます。したがって、deviceの電力バジェットを拡張してもっと効率的に使用できます。

注意 deviceの電力バジェットは慎重に計画し、電力モニターリング機能をイネーブルにし、電源装置に対してオーバーサブスクライブにならないようにする必要があります。

（注） 手動で電力バジェットを設定する場合、deviceと受電デバイスの間のケーブルでの電力消失を考慮する必要があります。

すべての PoE ポートのパワー バジェット

手順の概要

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **power inline consumption default** *wattage*
5. **end**
6. **show power inline consumption default**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**no cdp run** 例： `スイッチ(config)# no cdp run`	（任意）CDP をディセーブルにします。
ステップ 4	**power inline consumption default** *wattage* 例： `スイッチ(config)# power inline consumption default 5000`	各 PoE ポートに接続された受電デバイスの消費電力を設定します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show power inline consumption default** 例： `スイッチ# show power inline consumption default`	消費電力のステータスを表示します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

特定の PoE ポートのパワー バジェット

手順の概要

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **interface** *interface-id*
5. **power inline consumption** *wattage*
6. **end**
7. **show power inline consumption**
8. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no cdp run** 例： スイッチ(config)# **no cdp run**	（任意）CDP をディセーブルにします。
ステップ 4	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定する物理ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**power inline consumption** *wattage* 例： スイッチ(config-if)# **power inline consumption 5000**	device の PoE ポートに接続された受電装置の消費電力を設定します。 各受電装置に指定できる範囲は4000～です。デフォルトは です。
ステップ 6	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show power inline consumption** 例： スイッチ# **show power inline consumption**	電力消費データを表示します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

電力ポリシングの設定

デフォルトでは、device は接続されている受電装置の消費電力をリアルタイムでモニターリングします。消費電力に対するポリシングを行うように device を設定できます。デフォルトではポリシングは無効になります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline police** [**action**{**log** | **errdisable**}]
5. **exit**
6. 次のいずれかを使用します。
   - **errdisable detect cause inline-power**
   - **errdisable recovery cause inline-power**
   - **errdisable recovery interval** *interval*
7. **exit**
8. 次のいずれかを使用します。
   - **show power inline police**
   - **show errdisable recovery**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/1`	設定する物理ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**power inline police** [**action**{**log**	**errdisable**}] 例： `スイッチ(config-if)# power inline police`	ポートでリアルタイム消費電力が最大電力割り当てを超えるときに、次のいずれかのアクションを実行するようにdeviceを設定します。 • **power inline police**：PoE ポートをシャット ダウンし、ポートへの電力供給をオフにし、PoE ポートを error-disabled ステートに移行します。

	コマンドまたはアクション	目的
		（注） **errdisable detect cause inline-power** グローバル コンフィギュレーション コマンドを使用すると、PoE error-disabled の原因についてエラー検出を有効にできます。**errdisable recovery cause inline-power interval** *interval* グローバル コンフィギュレーション コマンドを使用すると、PoE error-disabled ステートから回復するためのタイマーを有効にすることもできます。 • **power inline police action errdisable**：リアルタイムの電力消費がポートの最大電力割り当てを超過した場合、ポートへの電力をオフにします。 • **power inline police action log**：ポートへの電源供給を継続し、syslog メッセージを生成します。 **action log** キーワードを入力しない場合、デフォルトのアクションによってポートがシャットダウンされ、error-disabled ステートになります。
ステップ 5	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	次のいずれかを使用します。 • **errdisable detect cause inline-power** • **errdisable recovery cause inline-power** • **errdisable recovery interval** *interval* 例： `スイッチ(config)# errdisable detect cause inline-power` `スイッチ(config)# errdisable recovery cause inline-power` `スイッチ(config)# errdisable recovery interval 100`	（任意）PoE error-disabled ステートからのエラー回復を有効にし、PoE 回復メカニズム変数を設定します。 デフォルトでは、回復間隔は 300 秒です。 **interval** *interval* には、error-disabled ステートから回復する時間を秒単位で指定します。指定できる範囲は 30 ～ 86400 です。
ステップ 7	**exit** 例： `スイッチ(config)# exit`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	次のいずれかを使用します。 • **show power inline police** • **show errdisable recovery** 例： スイッチ# **show power inline police** スイッチ# **show errdisable recovery**	電力モニタリング ステータスを表示し、エラー回復設定を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

電力ステータスのモニタ

表 *9:* 電力ステータスの ***show*** コマンド

コマンド	目的
show env power switch	（任意）指定したスイッチの内部電源装置のステータスを表示します。
show power inline [*interface-id*	スイッチ、インターフェイス、の PoE ステータスを表示します。
show power inline police	電力ポリシングのデータを表示します。
show env power	指定したスイッチの電源モジュールのステータスを表示します。

PoE の設定例

パワー バジェット：例

次のいずれかのコマンドを入力すると、

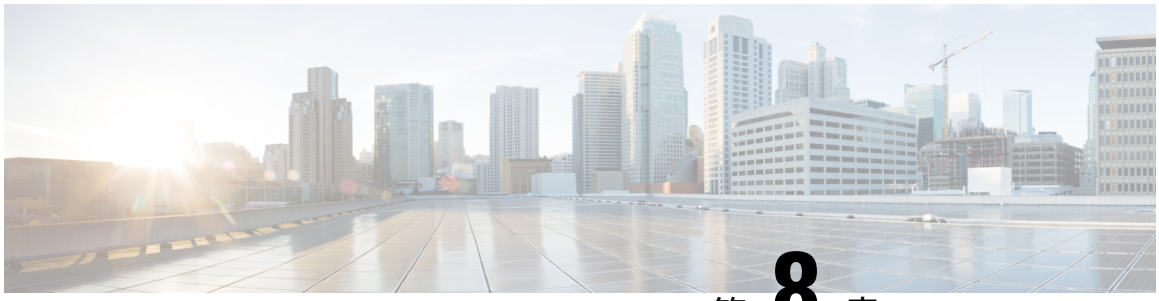
- [**no**] **power inline consumption default** *wattage* グローバル コンフィギュレーション コマンド

- [**no**] **power inline consumption** *wattage*

  インターフェイス コンフィギュレーション コマンド

次の注意メッセージが表示されます。

```
%CAUTION: Interface Gi1/0/1: Misconfiguring the 'power inline  consumption/allocation'
command may cause damage to the
switch and void your warranty. Take precaution not to oversubscribe the power supply.
It is recommended to enable power
policing if the switch supports it. Refer to documentation.
```



第 8 章

2 イベント分類の設定

2 イベント分類について

クラス 4 デバイスが検出されると、IOS は、CDP または LLDP のネゴシエーションを行うことなく 30W を割り当てます。これは、リンクがアップする前であっても、クラス 4 の電源デバイスは 30W を得ることを意味します。

また、ハードウェア レベルで、PSE は 2 イベント分類を行い、これにより、クラス 4 PD はハードウェアから 30W を供給する PSE の能力を検出し、それ自体を登録することができます。また、CDP/LLDP パケット交換を待つことなく最大 PoE+ レベルまで移動できます。

2 イベントがポートで有効になったら、ポートの遮断または開放を手動で行うか、または PD を再度接続して IEEE 検出を再度開始する必要があります。2 イベント分類がポートで有効になっている場合、クラス 4 デバイスの電力バジェット割り当ては 30W です。その他の場合は 15.4W です。

2 イベント分類の設定

2 イベント分類についてスイッチを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline port 2-event**
5. **end**

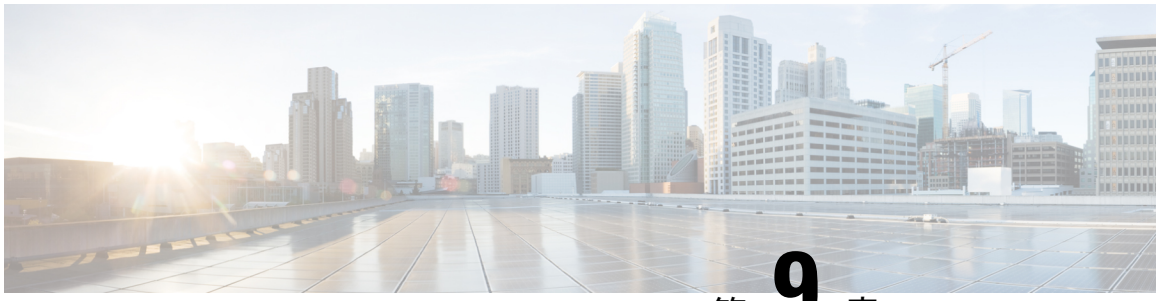
手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet2/0/1**	設定する物理ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**power inline port 2-event** 例： スイッチ(config-if)# **power inline port 2-event**	スイッチで 2 イベント分類を設定します。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

例：2 イベント分類の設定

次に、2 イベント分類を設定する例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)#  interface gigabitethernet2/0/1
スイッチ(config-if)# power inline port 2-event
スイッチ(config-if)# end
```



第 9 章

EEE の設定

EEE の制約事項

Energy Efficient Ethernet（EEE）には次の制限があります。

- EEE の設定を変更すると、デバイスがレイヤ 1 の自動ネゴシエーションを再起動しなければならないため、インターフェイスがリセットされます。
- 受信パスでデータを受け入れる前により長いウェイク アップ時間を必要とするデバイスのリンク層検出プロトコル（LLDP）を有効にする必要がある場合があります。これにより、デバイスは送信リンク パートナーから拡張システムのウェイク アップ時間についてネゴシエーションできます。

EEE について

EEE の概要

Energy Efficient Ethernet (EEE) は、アイドル時間にイーサネット ネットワークの消費電力を減らすように設計された IEEE 802.3az の標準です。

デフォルトの EEE 設定

EEE の設定方法

EEE 対応リンク パートナーに接続されているインターフェイスの EEE を有効または無効にできます。

EEE の有効化または無効化

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **power efficient-ethernet auto**
4. **no power efficient-ethernet auto**
5. **end**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Device(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**power efficient-ethernet auto** 例： `Device(config-if)# power efficient-ethernet auto`	特定のインターフェイスで EEE を有効にします。EEE が有効になっている場合、デバイスはリンク パートナーに EEE をアドバタイズし、自動ネゴシエートします。
ステップ 4	**no power efficient-ethernet auto** 例： `Device(config-if)# no power efficient-ethernet auto`	指定したインターフェイス上で EEE を無効にします。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

EEE の監視

表 *10 : EEE* 設定を表示するコマンド

コマンド	目的
show eee capabilities interface *interface-id*	指定インターフェイスの EEE 機能を表示します。
show eee status interface *interface-id*	指定したインターフェイスの EEE ステータス情報を表示します。
show eee counters interface *interface-id*	指定したインターフェイスの EEE 機能を表示します。

次に、**show eee** コマンドの例を示します。

```
Switch#show eee capabilities interface gigabitEthernet2/0/1
Gi2/0/1
EEE(efficient-ethernet): yes (100-Tx and 1000T auto)
Link Partner : yes (100-Tx and 1000T auto)

ASIC/Interface : EEE Capable/EEE Enabled

Switch#show eee status interface gigabitEthernet2/0/1
Gi2/0/1 is up
EEE(efficient-ethernet): Operational
Rx LPI Status : Low Power
Tx LPI Status : Low Power
Wake Error Count : 0

ASIC EEE STATUS
Rx LPI Status : Receiving LPI
Tx LPI Status : Transmitting LPI
Link Fault Status : Link Up
Sync Status : Code group synchronization with data stream intact

Switch#show eee counters interface gigabitEthernet2/0/1
```

```
LP Active Tx Time (10us) : 66649648
LP Transitioning Tx : 462
LP Active Rx Time (10us) : 64911682
LP Transitioning Rx : 153
```

EEE の設定例

次に、インターフェイスで EEE を有効にする例を示します。

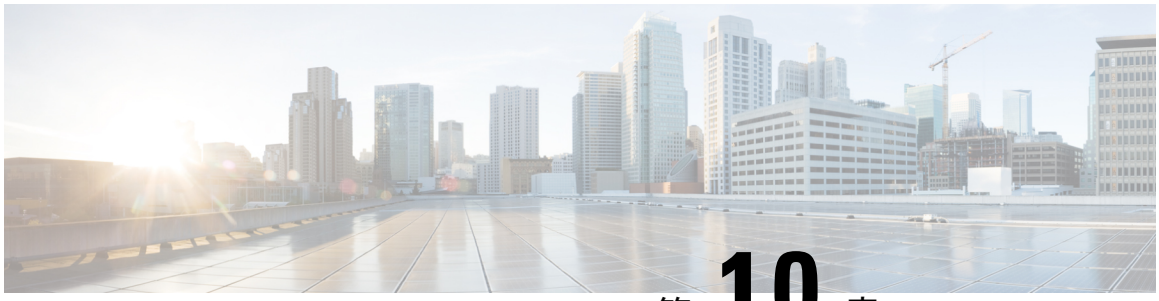
```
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# power efficient-ethernet auto
```

次に、インターフェイスで EEE を無効にする例を示します。

```
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# no power efficient-ethernet auto
```

第 II 部

IP マルチキャスト ルーティング



第 10 章

IP マルチキャスト ルーティング テクノロジーの概要

IP マルチキャスト テクノロジーに関する情報

情報配信における IP マルチキャストの役割

IP マルチキャストは、単一の情報ストリームを何千もの潜在的な企業および家庭に同時に配信することによってトラフィックを削減する帯域幅節約テクノロジーです。マルチキャストを利用するアプリケーションには、ビデオ会議、企業コミュニケーション、通信教育、およびソフトウェア、株価情報、ニュースの配信などが含まれます。

IP マルチキャスト ルーティングにより、ホスト（ソース）は、IP マルチキャスト グループ アドレスと呼ばれる特別な形式の IP アドレスを使用して、IP ネットワーク内の任意の場所にあるホスト（レシーバ）にパケットを送信できます。ソースのホストは、マルチキャスト グループ アドレスをパケットの宛先 IP アドレス フィールドに挿入します。IP マルチキャスト ルータおよびマルチレイヤ スイッチは、受信した IP マルチキャスト パケットを、マルチキャスト グループのメンバにつながるすべてのインターフェイスから転送します。どのホストも、グループのメンバであるかどうかにかかわらず、グループに送信できます。ただし、グループのメンバだけがメッセージを受信します。

IP マルチキャスト ルーティング プロトコル

ソフトウェアでは、IP マルチキャスト ルーティングを実装するため、次のプロトコルがサポートされています。

- IGMP を LAN 上のホストとその LAN 上のルータ間で使用して、ホストがメンバーになっているマルチキャスト グループを追跡します。

- PIM（Protocol Independent Multicast）は、相互に転送されるマルチキャスト パケット、および直接接続されている LAN に転送されるマルチキャスト パケットを追跡するためにルータ間で使用されます。

次の図に、これらのプロトコルが IP マルチキャスト環境内のどの部分で動作するかを示します。

マルチキャスト グループ伝送方式

IP通信は、最初の図に示すように、トラフィックの送信者として機能するホストと、レシーバとして機能するホストで構成されます。送信者はソースと呼ばれます。従来の IP 通信は、単一のホスト ソースがパケットを別の単一ホスト（ユニキャスト伝送）またはすべてのホスト（ブロードキャスト伝送）に送信することによって行われます。IPマルチキャストは第三の方式を提供するものであり、ホストはすべてのホストのサブセットにパケットを送信できます（マルチキャスト伝送）。受信側のホストのこのサブセットをマルチキャスト グループと呼びます。マルチキャスト グループに属するホストは、グループ メンバと呼ばれます。

マルチキャストは、このグループの概念に基づいています。マルチキャスト グループは、特定のデータ ストリームを受信するためにグループに加入する任意の数のレシーバです。このマルチキャスト グループには、物理的境界または地理的境界はありません。ホストは、インターネット上または任意のプライベート ネットワーク上のどこにでも配置できます。ソースから特定のグループに対するデータを受信する必要があるホストはそのグループに加入する必要があります。グループに加入するには、ホスト レシーバで Internet Group Management Protocol（IGMP）を使用します。

マルチキャスト環境では、どのホストも、グループのメンバであるかどうかにかかわらず、グループに送信できます。ただし、そのグループに送信されたパケットはグループのメンバだけが受信できます。IP ユニキャスト パケットと同様、マルチキャスト パケットは、ベストエフォート型の信頼性を使用してグループに配信されます。

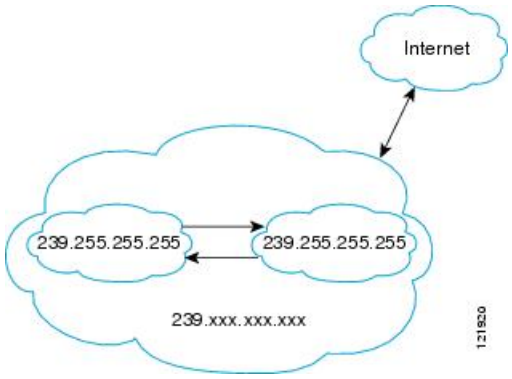
次の図では、レシーバ（指定したマルチキャスト グループ）がソースからのビデオ データ ストリームを受信する必要があります。これらのレシーバは、ネットワーク内のルータに IGMP ホスト レポートを送信することによってその意思を示します。この場合、ルータがソースからレシーバへのデータの配信を担います。ルータは、Protocol Independent Multicast（PIM）を使用して、マルチキャスト配信ツリーを動的に作成します。その後、ソースとレシーバ間のパスにあるネットワーク セグメントにのみ、ビデオ データ ストリームが配信されます。

IP マルチキャスト境界

図に示すように、アドレス スコーピングは、同じ IP アドレスを持つ RP が含まれるドメインが相互にデータを漏出させることのないように、ドメイン境界を定義します。スコーピング

は、大きなドメイン内のサブネット境界や、ドメインとインターネットの間の境界で実行されます。

図 *2*：境界でのアドレス スコーピング



マルチキャスト グループ アドレッシングのインターフェイスに管理スコープの境界を設定するには、**ip multicast boundary** コマンドと *access-list* 引数を使用します。影響を受けるアドレス範囲は、標準アクセス リストによって定義されます。境界が設定されると、マルチキャスト データ パケットは境界を越えて出入りできなくなります。境界を定めることで、同じマルチキャスト グループ アドレスをさまざまな管理ドメイン内で使用できます。

Internet Assigned Numbers Authority（IANA）は、マルチキャスト アドレス範囲 239.0.0.0 ～ 239.255.255.255 を管理スコープ アドレスとして指定しています。この範囲のアドレスは、さまざまな組織で管理されるドメイン内で再使用されます。これらは、グローバルに一意ではなくローカルとみなされます。

filter-autorp キーワードを設定して、管理用スコープの境界で Auto-RP 検出と通知メッセージを検査し、フィルタできます。境界のアクセス コントロール リスト（ACL）に拒否された Auto-RP パケットからの Auto-RP グループ範囲通知は削除されます。Auto-RP グループ範囲通知は、Auto-RP グループ範囲のすべてのアドレスが境界 ACL によって許可される場合に限り境界を通過できます。許可されないアドレスがある場合は、グループ範囲全体がフィルタリングされ、Auto-RP メッセージが転送される前に Auto-RP メッセージから削除されます。

IP マルチキャスト グループ アドレッシング

マルチキャスト グループは、マルチキャスト グループ アドレスによって識別されます。マルチキャスト パケットは、そのマルチキャスト グループ アドレスに配信されます。単一のホストを独自に識別するユニキャスト アドレスとは異なり、マルチキャスト IP アドレスは特定のホストを識別しません。マルチキャストアドレスに送信されるデータを受信するには、アドレスが識別するグループにホストが参加する必要があります。データは、マルチキャストアドレスに送信され、そのグループに送信されたトラフィックを受信する意思を示してグループに加入しているすべてのホストによって受信されます。マルチキャスト グループ アドレスは、送信元でグループに割り当てられます。マルチキャスト グループ アドレスを割り当てるネットワーク管理者は、Internet Assigned Numbers Authority（IANA）で予約されるマルチキャスト アドレスの範囲にアドレスが準拠していることを確認する必要があります。

IP クラス D アドレス

IP マルチキャスト アドレスは、IANA によって IPv4 クラス D アドレス空間に割り当てられました。クラス D アドレスの上位 4 ビットは 1110 です。したがって、ホスト グループ アドレスの範囲は 224.0.0.0 ～ 239.255.255.255 であると考えられます。マルチキャスト アドレスは送信元（送信者）でマルチキャスト グループの受信先として選択されます。

（注） クラス D アドレスの範囲は、IP マルチキャスト トラフィックのグループ アドレスまたは宛先アドレスにだけ使用されます。マルチキャスト データグラムの送信元アドレスは常にユニキャスト送信元アドレスになります。

IP マルチキャスト アドレスのスコーピング

さまざまなアドレス範囲の予測可能な動作を提供したり、より小規模なドメイン内でアドレスを再利用したりできるよう、マルチキャストアドレスの範囲はさらに分割されます。表に、マルチキャスト アドレスの範囲を要約します。それに続いて、各範囲について簡単に説明します。

表 *11* : マルチキャスト アドレス範囲の割り当て

名前	範囲	説明
予約済みリンクローカル アドレス	224.0.0.0 ～ 224.0.0.255	ローカル ネットワーク セグメントのネットワーク プロトコルで使用するために予約されています。
グローバル スコープ アドレス	224.0.1.0 ～ 238.255.255.255	組織間およびインターネット上でマルチキャスト データを送信するために予約されています。
Source Specific Multicast	232.0.0.0 ～ 232.255.255.255	明示的にグループに参加している受信者だけにデータを転送する SSM データグラム配信モデル用に予約されています。
GLOP アドレス	233.0.0.0 ～ 233.255.255.255	割り当て済みの自律システム（AS）ドメイン番号をすでに持つ組織によって静的に定義されるアドレス用に予約されています。
限定スコープ アドレス	239.0.0.0 ～ 239.255.255.255	管理スコープ アドレスまたはプライベート マルチキャスト ドメインで使用するための限定スコープ アドレスとして予約されています。

予約済みリンクローカル アドレス

IANA では、ローカル ネットワーク セグメントのネットワーク プロトコルで使用するために 224.0.0.0 ～ 224.0.0.255 の範囲を予約しています。この範囲のアドレスを持つパケットはスコープ内ローカルであり、IP ルータによって転送されません。通常、リンク ローカル宛先アドレ

スを持つパケットは存続可能時間（TTL）値 1 を使用して送信されるため、ルータによって転送されません。

この範囲内の予約済みリンクローカル アドレスは、それぞれに予約されたネットワーク プロトコル機能を提供します。ネットワーク プロトコルは、これらのアドレスをルータの自動検出および重要なルーティング情報の伝達用に使用します。たとえば、Open Shortest Path First（OSPF）は、IP アドレスの 224.0.0.5 と 224.0.0.6 を使用してリンクステート情報を交換します。

IANA では、ネットワーク プロトコルやネットワーク アプリケーションに対する単一マルチキャスト アドレス要求を 224.0.1.xxx のアドレス範囲外に割り当てています。マルチキャスト ルータはこれらのマルチキャスト アドレスを転送します。

（注） ASR 903 RSP2 モジュールでは、デフォルトにより、予約済みのリンクローカル アドレスを持つすべてのパケットが CPU にパントされます。

グローバル スコープ アドレス

224.0.1.0 ～ 238.255.255.255 の範囲のアドレスは、グローバル スコープ アドレスと呼ばれます。これらのアドレスは、組織間およびインターネット上でのマルチキャスト データの送信に使用します。これらのアドレスの一部はマルチキャスト アプリケーションで使用するよう IANA によって予約されています。たとえば、IP アドレス 224.0.1.1 は、Network Time Protocol（NTP）用に予約されています。

Source Specific Multicast アドレス

232.0.0.0/8 のアドレス範囲は、Source Specific Multicast（SSM）用に予約されています。Cisco IOS ソフトウェアでは、**ip pim ssm** コマンドを使用して任意の IP マルチキャスト アドレス用の SSM も設定できます。SSM は、1 対多通信での効率的なデータ配信メカニズムを可能にする Protocol Independent Multicast（PIM）の拡張版です。SSM については、IP マルチキャスト配信モード （96 ページ）の項を参照してください。

GLOP アドレス

GLOP アドレッシングでは（233/8 の RFC 2770、GLOP アドレッシングで提案されているように）、AS 番号をすでに予約している組織による静的に定義されたアドレス用に 233.0.0.0/8 の範囲を予約することを提案しています。これは、GLOP アドレッシングと呼ばれます。ドメインの AS 番号は 233.0.0.0/8 アドレス範囲の 2 番目と 3 番目のオクテットに組み込まれます。たとえば、AS 62010 は 16 進数形式で F23A と表されます。この 2 つのオクテット F2 および 3A を分割すると、結果は 10 進数でそれぞれ 242 および 58 となります。これらの値は、AS 62010 に使用するようにグローバルに予約される 233.242.58.0/24 のサブネットとなります。

限定スコープ アドレス

239.0.0.0 ～ 239.255.255.255 の範囲は、管理スコープ アドレス、またはプライベート マルチキャスト ドメインで使用する限定スコープ アドレスとして予約されています。これらのアドレス

は、ローカル グループまたは組織に使用するように制限されています。会社、大学および他の組織は、限定スコープアドレスを使用すると、ドメイン外に転送されないローカルマルチキャストアプリケーションを使用できます。通常、ルータは、このアドレス範囲のマルチキャストトラフィックが自律システム（AS）またはユーザー定義のドメイン外にフローしないようにするフィルタを使用して設定されます。AS またはドメイン内では、ローカル マルチキャスト境界を定義できるように、限定スコープ アドレス範囲を細分化することもできます。

（注） ネットワーク管理者はこの範囲内のマルチキャストアドレスを使用できます。これによって、インターネット内の他の場所と競合することはありません。

レイヤ 2 マルチキャスト アドレス

従来、LAN セグメントのネットワーク インターフェイス カード（NIC）が受信できるのは、Burned-In MAC Address またはブロードキャスト MAC アドレスに指定されたパケットだけでした。IP マルチキャストでは、複数のホストが共通の宛先 MAC アドレスを使用した単一のデータ ストリームを受信する必要があります。複数のホストが同じパケットを受信する場合、複数のマルチキャスト グループを区別できるように、何らかの方法を考案する必要があります。そのための 1 つの方法は、IP マルチキャスト クラス D アドレスを MAC アドレスに直接マッピングすることです。この方法を使用すると、NIC は多くの異なる MAC アドレスを宛先とするパケットを受信できます。

Cisco グループ管理プロトコル（CGMP）は、IGMP によって実行される作業と同様の作業を実行するために、Catalyst スイッチに接続されたルータ上で使用されます。IP マルチキャスト データ パケットと IGMP レポート メッセージ（いずれも MAC レベルで同じグループ アドレスにアドレス指定されます）を区別できない Catalyst スイッチの場合、CGMP が必要になります。

IP マルチキャスト配信モード

IP マルチキャスト配信のモードは、送信元ホストではなく、受信側ホストのみによって異なります。送信元ホストは、パケットの IP 送信元アドレスとしての固有の IP アドレスと、パケットの IP 宛先アドレスとしてのグループ アドレスを使用して、IP マルチキャスト パケットを送信します。

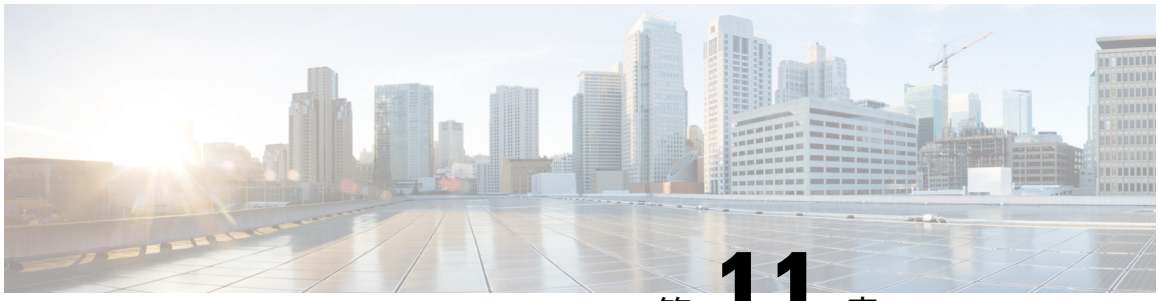
Source Specific Multicast

Source Specific Multicast（SSM）は、ブロードキャスト アプリケーションとしても知られる 1 対多アプリケーションをサポートする最善のデータグラム配信モデルです。SSM は、オーディオおよびビデオのブロードキャストアプリケーション環境を対象としたシスコの IP マルチキャストのコア ネットワーク テクノロジーです。

SSM 配信モードの場合、IP マルチキャスト レシーバ ホストは IGMP バージョン3（IGMPv3）を使用してチャネル（S, G）を登録する必要があります。このチャネルに登録することによって、ソース ホストがグループ G に送信した IP マルチキャスト トラフィックの受信をレシーバ

ホストが要求していることを示します。ネットワークは、ソース ホスト S からグループ G に送信された IP マルチキャスト パケットを、チャネル (S, G) に登録したネットワーク内のすべてのホストに配信します。

SSM では、ネットワーク内でグループ アドレスを割り当てる必要はありません。各ソース ホスト内で割り当てるだけです。同じソース ホストで実行している各アプリケーションはそれぞれ異なる SSM グループを使用する必要があります。異なるソース ホストで実行しているアプリケーションは、SSM グループ アドレスを再利用できます。ネットワークに大量のトラフィックを発生させることはありません。



第 11 章

基本的な IP マルチキャスト ルーティングの設定

基本的な IP マルチキャスト ルーティングの前提条件

次に、基本的な IP マルチキャスト ルーティングを設定するための前提条件を示します。

- IP マルチキャスト ルーティングを実行するには、PIM バージョンおよび PIM モードを設定する必要があります。スイッチはモード設定に従って、マルチキャスト ルーティング テーブルを読み込み、直接接続された LAN から受信したマルチキャスト パケットを転送します。インターフェイスは PIM デンス モード、スパース モード、または SM-DM スパース - デンス モードのいずれかに設定できます。
- インターフェイスで PIM をイネーブルにすると、同じインターフェイス上で IGMP 処理もイネーブルになります。（IP マルチキャスティングに加入するには、マルチキャスト ホスト、ルータ、およびマルチレイヤデバイスで IGMP が動作している必要があります）

  複数のインターフェイスで PIM をイネーブルにした場合に、そのほとんどのインターフェイスが発信インターフェイス リストに含まれておらず、IGMP スヌーピングがディセーブルになっている場合は、レプリケーションが増加することにより、発信インターフェイスが回線レートを維持できないこともあります。

基本的な IP マルチキャスト ルーティングの制約事項

次に、IP マルチキャスト ルーティングの制約事項を示します。

- マルチキャスト ルーティングは Catalyst 3560-CX スイッチでのみサポートされます。

基本的な IP マルチキャスト ルーティングに関する情報

IP マルチキャストは、ネットワーク リソース（特に、音声やビデオなどの帯域幅集約型サービス）を効率的に使用する方法です。IP マルチキャスト ルーティングにより、ホスト（ソース）は、IP マルチキャスト グループ アドレスと呼ばれる特別な形式の IP アドレスを使用して、IP ネットワーク内の任意の場所にあるホスト（レシーバ）にパケットを送信できます。

送信側ホストは、マルチキャスト グループ アドレスをパケットの IP 宛先アドレスフィールドに挿入します。IP マルチキャスト ルータおよびマルチレイヤ devices は、マルチキャスト グループのメンバに接続されたすべてのインターフェイスから着信した IP マルチキャスト パケットを転送します。どのホストも、グループのメンバであるかどうかにかかわらず、グループに送信できます。ただし、グループのメンバだけがメッセージを受信します。

IP マルチキャスト ルーティングのデフォルト設定

次の表に、IP マルチキャスト ルーティングのデフォルト設定を示します。

表 *12 : IP* マルチキャスト ルーティングのデフォルト設定

機能	デフォルト設定
マルチキャスト ルーティング	すべてのインターフェイスでディセーブル
PIM のバージョン	バージョン 2
PIM モード	モードは未定義
PIM スタブ ルーティング	未設定
PIM RP アドレス	未設定
PIM ドメイン境界	ディセーブル。
PIM マルチキャスト境界	なし
候補 BSR	ディセーブル。
候補 RP	ディセーブル。
SPT しきい値レート	0 kb/s
PIM ルータ クエリー メッセージ インターバル	30 秒

sdr リスナー サポートの

MBONEは、相互接続された、IPマルチキャストトラフィックの転送が可能なインターネットルータおよびホストの小さなサブセットです。その他のマルチメディアコンテンツも、通常はMBONEを通してブロードキャストされます。マルチメディアセッションに加入する前に、このセッションで使用されているマルチメディア グループ アドレス、ポート、セッションがアクティブになる時期、およびワークステーションで必要となるアプリケーションの種類（音声、ビデオなど）を把握する必要があります。この情報は、MBONE Session Directory バージョン 2（sdr）ツールによって提供されます。このフリーウェア アプリケーションは WWW 上の複数のサイト（http://www.video.ja.net/mice/index.html など）からダウンロードできます。

SDR は、Session Announcement Protocol（SAP）マルチキャスト パケット用の Well-known マルチキャスト グループ アドレスおよびポートを、SAP クライアントから傍受するマルチキャスト アプリケーションです（SAP クライアントは、会議セッションをアナウンスします）。これらのSAPパケットには、セッションの説明、セッションがアクティブな期間、IPマルチキャスト グループ アドレス、メディア形式、担当者、およびアドバタイズされたマルチメディアセッションに関するその他の情報が格納されます。SAP パケットの情報は、[SDR Session Announcement] ウィンドウに表示されます。

基本的な IP マルチキャスト ルーティングの設定方法

基本的な IP マルチキャスト ルーティングの設定

デフォルトでは、マルチキャストルーティングはディセーブルとなっており、モードは設定されていません。

この手順は必須です。

始める前に

PIM バージョンと PIM モードを設定する必要があります。スイッチはモード設定に従って、マルチキャスト ルーティング テーブルを読み込み、直接接続されたLANから受信したマルチキャスト パケットを転送します。

マルチキャスト ルーティング テーブルへのパケット読み込みでは、DM インターフェイスは常にテーブルに追加されます。SM インターフェイスがテーブルに追加されるのは、ダウンストリーム デバイスから定期的な Join メッセージを受信した場合、またはインターフェイスに直接接続されたメンバーが存在する場合に限ります。LAN から転送する場合、グループが認識している RP があれば、SM 動作が行われます。その場合、パケットはカプセル化され、その RP に送信されます。認識している RP がなければ、パケットは DM 方式でフラッディングされます。マルチキャスト送信元アドレスは、PIMデンスモードとPIM Any Sourceマルチキャストモードの両方で、直接接続された着信インターフェイス（同じサブネットの一部）に存在する必要があります。特定の送信元からのマルチキャスト トラフィックが十分であれば、レシーバの先頭ホップ ルータからその送信元に Join メッセージが送信され、送信元を基点とする配信ツリーが構築されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	マルチキャスト ルーティングをイネーブルにするレイヤ 3 インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。 次のいずれかのインターフェイスを指定する必要があります。 • ルーテッドポート：レイヤ 3 ポートとして **no switchport** インターフェイス コンフィギュレーション コマンドを入力して設定された物理ポートです。また、インターフェイスの IP PIM スパース - デンス モードをイネーブルにして、静的に接続されたメンバーとしてインターフェイスを IGMP スタティック グループに加入させる必要があります。 • SVI：**interface vlan** *vlan-id* グローバル コンフィギュレーション コマンドを使用して作成された VLAN インターフェイスです。また、VLAN 上で IP PIM スパース - デンス モードをイネーブルにして、静的に接続されたメンバーとして VLAN を IGMP スタティック グループに加入させ、VLAN、IGMP スタティック グループ、お

	コマンドまたはアクション	目的
		よび物理インターフェイスで IGMP スヌーピングをイネーブルにする必要があります。 これらのインターフェイスには、IP アドレスを割り当てる必要があります。
ステップ 4	**ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**} 例： `スイッチ(config-if)# ip pim sparse-dense-mode`	インターフェイスで PIM モードをイネーブルにします。 デフォルトで、モードは設定されていません。 キーワードの意味は次のとおりです。 • **dense-mode**：デンス動作モードをイネーブルにします。 • **sparse-mode**：スパース動作モードをイネーブルにします。SM を設定する場合は、RP も設定する必要があります。 • **sparse-dense-mode**：グループが属するモードでインターフェイスが処理されるようにします。DM-SM 設定を推奨します。 （注） インターフェイスで PIM を無効化するには、**no ip pim** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

オプションの IP マルチキャスト ルーティングの設定

IP マルチキャスト境界の定義

自動 RP メッセージが PIM ドメインに入らないようにする場合は、マルチキャスト境界を定義します。自動 RP 情報を伝達する 224.0.1.39 および 224.0.1.40 宛てのパケットを拒否するアクセス リストを作成します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **deny** *source* [*source-wildcard*]
4. **interface** *interface-id*
5. **ip multicast boundary** *access-list-number*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**access-list** *access-list-number* **deny** *source* [*source-wildcard*] 例： スイッチ(config)# **access-list 12 deny 224.0.1.39** **access-list 12 deny 224.0.1.40**	標準アクセス リストを作成し、コマンドを必要な回数だけ実行します。 • *access-list-number* の範囲は 1 ～ 99 です。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。 • *source* には、自動 RP 情報を伝達するマルチキャスト アドレス 224.0.1.39 および 224.0.1.40 を入力します。

	コマンドまたはアクション	目的
		• （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在します。
ステップ 4	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**ip multicast boundary** *access-list-number* 例： `スイッチ(config-if)# ip multicast boundary 12`	ステップ 2 で作成したアクセス リストを指定し、境界を設定します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

マルチキャスト VRF の設定

手順

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**ip routing** 例： スイッチ(config)# ip routing	IP ルーティング モードをイネーブルにします
ステップ 3	**ip vrf** *vrf-name* 例： スイッチ(config)# ip vrf vpn1	VRF 名を指定し、VRF コンフィギュレーション モードを開始します。
ステップ 4	**rd** *route-distinguisher* 例： スイッチ(config-vrf)# rd 100:2	ルート識別子を指定して VRF テーブルを作成します。AS 番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。
ステップ 5	**route-target** {**export** | **import** | **both**} *route-target-ext-community* 例： スイッチ(config-vrf)# route-target import 100:2	指定された VRF のインポート、エクスポート、またはインポートおよびエクスポート ルート ターゲット コミュニティのリストを作成します。AS システム番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。*route-target-ext-community* は、ステップ 4 で入力した *route-distinguisher* と同一にする必要があります。
ステップ 6	**import map** *route-map* 例： スイッチ(config-vrf)# import map importmap1	（任意）VRF にルート マップを対応付けます。
ステップ 7	**ip multicast-routing vrf** *vrf-name* **distributed** 例： スイッチ(config-vrf)# ip multicast-routing vrf vpn1 distributed	（任意）VRF テーブルでグローバル マルチキャスト ルーティングをイネーブルにします。
ステップ 8	**interface** *interface-id* 例： スイッチ(config-vrf)# interface gigabitethernet 1/0/2	VRF に関連付けるレイヤ 3 インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスはルーテッド ポートまたは SVI に設定できます。
ステップ 9	**ip vrf forwarding** *vrf-name* 例：	VRF をレイヤ 3 インターフェイスに対応付けます。

	コマンドまたはアクション	目的
	スイッチ(config-if)# ip vrf forwarding vpn1	
ステップ 10	**ip address** *ip-address* **mask** 例： スイッチ(config-if)# ip address 10.1.5.1 255.255.255.0	レイヤ 3 インターフェイスの IP アドレスを設定します。
ステップ 11	**ip pim sparse-dense mode** 例： スイッチ(config-if)# ip pim sparse-dense mode	VRF に関連付けられているレイヤ 3 インターフェイス上で、PIM をイネーブルにします。
ステップ 12	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 13	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] 例： スイッチ# show ip vrf detail vpn1	設定を確認します。設定した VRF に関する情報を表示します。
ステップ 14	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SAP リスナーを使用したマルチキャスト マルチメディア セッションのアドバタイジング

マルチキャスト メディア会議やその他のマルチキャスト セッションを支援したり、参加予定者に関連セッションの設定情報を通知したりするために Session Description Protocol と Session Anouncement Protocol、およびアプリケーションを使用する場合は、SAP リスナー サポートをイネーブルにします。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sap cache-timeout** *minutes*
4. **interface** *type number*
5. **ip sap listen**
6. **end**

7. **clear ip sap** [*group-address* | **"** *session-name* **"**]
8. **show ip sap** [*group-address* | **"** *session-name* **"**| **detail**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sap cache-timeout** *minutes* 例： `Router(config)# ip sap cache-timeout 600`	（任意）SAP キャッシュ エントリがキャッシュ内でアクティブである期間を制限します。 • デフォルトでは、SAP キャッシュ エントリはネットワークから受信された 24 時間後に削除されます。
ステップ 4	**interface** *type number* 例： `Router(config)# interface ethernet 1`	IGMPv3 をイネーブルにできるホストに接続されているインターフェイスを選択します。
ステップ 5	**ip sap listen** 例： `Router(config-if)# ip sap listen`	セッション ディレクトリ アナウンスメントをリッスンするソフトウェアをイネーブルにします。
ステップ 6	**end** 例： `Router(config-if)# end`	セッションを終了し、EXEC モードに戻ります。
ステップ 7	**clear ip sap** [*group-address*	**"** *session-name* **"**] 例： `Router# clear ip sap "Sample Session"`	SAP キャッシュ エントリまたは SAP キャッシュ全体を削除します。
ステップ 8	**show ip sap** [*group-address*	**"** *session-name* **"**| **detail**] 例： `Router# show ip sap 224.2.197.250 detail`	（任意）SAP キャッシュを表示します。

基本的なIPマルチキャストルーティングのモニタリングおよびメンテナンス

キャッシュ、テーブル、およびデータベースのクリア

特定のキャッシュ、テーブル、またはデータベースのすべての内容を削除できます。特定のキャッシュ、テーブル、またはデータベースの内容が無効である場合、または無効である可能性がある場合は、これらをクリアする必要があります。

次の表に示す特権EXECコマンドのいずれかを使用すると、IPマルチキャストのキャッシュ、テーブル、データベースをクリアできます。

表 *13*：キャッシュ、テーブル、およびデータベースをクリアするコマンド

コマンド	目的
clear ip igmp group {**group** [*hostname* | *IP address*] | **vrf** *name* **group** [*hostname* | *IP address*] }	IGMP キャッシュのエントリを削除します。
clear ip mroute { ***** | [*hostname* | *IP address*] | **vrf** *name* **group** [*hostname* | *IP address*] }	IPマルチキャストルーティングテーブルからエントリを削除します。
clear ip sap [*group-address* | **"***session-name***"**]	Session Directory Protocol バージョン 2 キャッシュ（sdr キャッシュ）エントリを削除します。

システムおよびネットワーク統計情報の表示

IP ルーティング テーブル、キャッシュ、データベースの内容など、特定の統計情報を表示できます。

（注） このリリースでは、ルート単位の統計情報がサポートされていません。

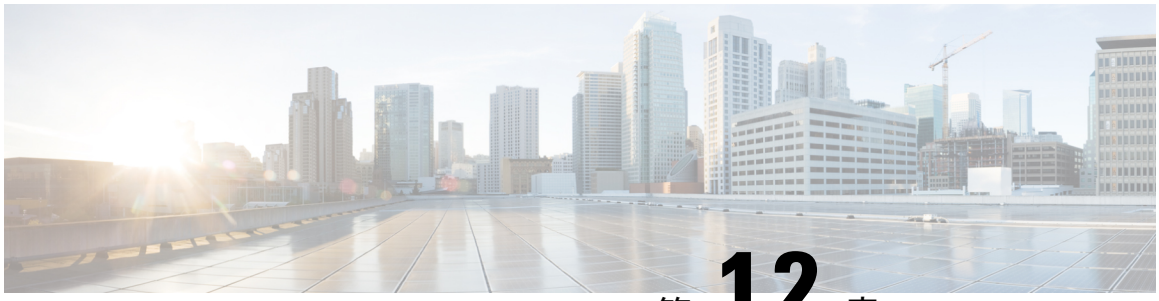
また、リソースの使用状況を学習し、ネットワーク問題を解決するための情報を表示することもできます。さらに、ノードの到達可能性に関する情報を表示し、そのパケットが経由するネットワーク内のパスを検出することもできます。

次の表に示す特権EXECコマンドのいずれかを使用すると、さまざまなルーティング統計情報を表示できます。

表 *14 :* システムおよびネットワーク統計情報を表示するコマンド

コマンド	目的
ping [*group-name* | *group-address*]	マルチキャスト グループ アドレスにインターネット制御メッセージ プロトコル（ICMP）エコー要求を送信します。
show ip igmp groups [*group-name*|*group-address*|*type-number*]	deviceに直接接続され、IGMP によって取得されたマルチキャスト グループを表示します。
show ip igmp interface [*type number*]	インターフェイスのマルチキャスト関連情報を表示します。
show ip mroute [*group-name* | *group-address*] [*source*] [**count** | **interface** | **proxy** | **pruned** | **summary** | **verbose**]	IP マルチキャスト ルーティング テーブルの内容を表示します。
show ip pim interface [*type number*] [**count** | **detail** | **df** | **stats**]	PIM に対して設定されたインターフェイスに関する情報を表示します。このコマンドは、すべてのソフトウェア イメージで使用できます。
show ip pim neighbor [*type number*]	deviceによって検出された PIM ネイバーのリストを表示します。このコマンドは、すべてのソフトウェア イメージで使用できます。
show ip pim rp [*group-name* | *group-address*]	スパース モードのマルチキャスト グループに関連付けられた RP ルータを表示します。このコマンドは、すべてのソフトウェア イメージで使用できます。
show ip rpf {*source-address* | *name*}	deviceの RPF の実行方法（ユニキャスト ルーティング テーブル、DVMRP ルーティング テーブル、静的マルチキャスト ルーティングのいずれかを使用）を表示します。 コマンド パラメータは次のとおりです。 • *Host name* または *IP address*：IP 名またはグループ アドレス。 • **Select**：グループ ベースの VRF 選択情報。 • **vrf**：VPN ルーティング/転送インスタンスを選択します。

コマンド	目的
show ip sap [*group* | **"***session-name***"** | **detail**]	Session Announcement Protocol（SAP）バージョン 2 キャッシュを表示します。 コマンド パラメータは次のとおりです。 • *A.B.C.D*：IP グループ アドレス。 • *WORD*：セッション名（二重引用符で囲む）。 • **detail**：セッションの詳細。



第 12 章

IGMP の設定

IGMP のの前提条件

- このモジュールの作業を実行する前に、『IP Multicast Routing Technology Overview』モジュールで説明している概念をよく理解しておく必要があります。
- このモジュールの作業では、IP マルチキャストがイネーブルに設定され、「Configuring Multicast Routing」モジュールで説明されている作業を使用して、Protocol Independent Multicast（PIM）インターフェイスが設定されていることを前提とします。

IGMP 設定の制約事項

次に、IGMP を設定する際の制約事項を示します。

- deviceは IGMP バージョン 1、2、3 をサポートしています。

（注） IGMP バージョン 3 の場合、IGMP バージョン 3 BISS（基本的な IGMPv3 スヌーピング サポート）のみがサポートされます。

- IGMP バージョン 3 では新しいメンバーシップ レポート メッセージを使用しますが、これらは以前の IGMP スヌーピングdevicesで正しく認識されない可能性があります。

- IGMPv3 は、ISM および SSM と同時に動作可能です。ISM では、exclude と include の両方のモードのレポートを適用できます。SSM では、ラストホップ ルータは include モードのレポートだけを受け入れます。exclude モードのレポートは無視されます。

IGMP に関する情報

Internet Group Management Protocol の役割

IGMP は、マルチキャスト グループの個々のホストを特定の LAN にダイナミックに登録するために使用します。インターフェイスで PIM をイネーブルにすると、IGMP もイネーブルになります。IGMP は、特別なマルチキャスト クエリアおよびホストを使用して、ネットワーク全体でマルチキャスト トラフィックのフローを自動的に制御および制限する手段を提供します。

- クエリアは、クエリー メッセージを送信して、特定のマルチキャスト グループのメンバであるネットワーク デバイスを検出するネットワーク デバイス（ルータなど）です。
- ホストは、クエリアにホスト メンバーシップを通知するためのレポート メッセージ（クエリー メッセージに応答するメッセージ）を送信するレシーバで、ルータも含まれます。ホストでは、IGMP メッセージを使用して、マルチキャスト グループに加入し、マルチキャスト グループを脱退します。

ホストは、そのローカル マルチキャスト デバイスに IGMP メッセージを送信することで、グループ メンバーシップを識別します。IGMP では、デバイスは IGMP メッセージを受信し、定期的にクエリーを送信して、特定のサブネットでアクティブなグループと非アクティブなグループを検出します。

IGMP マルチキャスト アドレス

IP マルチキャスト トラフィックには、グループ アドレス（クラス D IP アドレス）が使用されます。クラス D アドレスの上位 4 ビットは 1110 です。したがって、ホスト グループ アドレスの範囲は 224.0.0.0 ～ 239.255.255.255 であると考えられます。

224.0.0.0 ～ 224.0.0.255 のマルチキャスト アドレスは、ルーティング プロトコルおよびその他のネットワーク制御トラフィックが使用するために予約されています。アドレス 224.0.0.0 は、どのグループにも割り当てられません。

IGMP パケットは IP マルチキャスト グループ アドレスを使用して次のように送信されます。

- IGMP 汎用クエリーは、アドレス 224.0.0.1（サブネット上のすべてのシステム）を宛先とします。
- IGMP グループ固有のクエリーは、クエリー対象デバイスのグループ IP アドレスを宛先とします。
- IGMP グループ メンバーシップ レポートは、レポート対象デバイスのグループ IP アドレスを宛先とします。

- IGMPv2 グループ脱退メッセージは、アドレス 224.0.0.2（サブネット上のすべてのデバイス）を宛先とします。
- IGMPv3 メンバーシップ レポートはアドレス 224.0.0.22 を宛先とします。すべての IGMPv3 対応マルチキャスト デバイスはこのアドレスをリッスンする必要があります。

IGMP のバージョン

device は、IGMP バージョン 1、IGMP バージョン 2、および IGMP バージョン 3 をサポートしています。これらバージョンは、 device 上でそれぞれ相互運用できます。たとえば、IGMP スヌーピングがイネーブルになっており、クエリアのバージョンが IGMPv2 で、device がホストから IGMPv3 レポートを受信している場合、device は IGMPv3 レポートをマルチキャストルータに転送できます。

IGMPv3 device は、Source Specific Multicast（SSM）機能を実行しているデバイスとメッセージの送受信を行うことができます。

IGMP バージョン 1

IGMP バージョン 1（IGMPv1）にはクエリー応答モデルが使用されているため、マルチキャスト ルータおよびマルチレイヤ deviceは、ローカル サブネット上のどのマルチキャスト グループがアクティブであるか（マルチキャスト グループに関係するホストが 1 台または複数存在するか）を判別できます。IGMPv1 では別のプロセスを使用して、ホストをマルチキャスト グループに加入および脱退させることができます。詳細については、RFC 1112 を参照してください。

IGMPv2

IGMP バージョン 2 は IGMP 機能の拡張版です。IGMP 脱退処理などの機能を提供して、脱退遅延を短縮し、グループ固有のクエリー数を削減し、明示的な最大クエリー応答時間を短縮します。また、この作業を実行するために、マルチキャストプロトコルに依存することなく IGMP クエリアを選択する機能もルータに追加されます。詳細については、RFC 2236 を参照してください。

（注） IGMP バージョン 2 は device のデフォルトバージョンです。

IGMP バージョン 3

deviceは IGMP バージョン 3 をサポートしています。

IGMPv3 deviceは、Basic IGMPv3 Snooping Support（BISS）をサポートしています。BISS は、IGMPv1 および IGMPv2 スイッチでのスヌーピング機能と、IGMPv3 メンバーシップ レポート メッセージをサポートしています。ネットワークに IGMPv3 ホストがある場合、BISS によりマルチキャスト トラフィックのフラッディングは抑制されます。トラフィックは、IGMPv2 ま

たは IGMPv1 ホストの IGMP スヌーピング機能の場合とほぼ同じポート セットに抑制されます。

IGMPv3 device は、Source Specific Multicast（SSM）機能を実行しているデバイスとメッセージの送受信を行うことができます。

IGMPv3 ホスト シグナリング

IGMPv3 では、ホストはマルチキャスト グループのラストホップ ルータにメンバーシップ シグナルを送信します。ホストは、グループ メンバーシップ シグナルの送信に、送信元に関するフィルタリング機能を使用できます。ホストは、いくつかの特定の送信元を除くすべての送信元からグループへのトラフィックを受信する（exclude モード）というシグナルか、または、いくつかの特定の送信元からグループへのトラフィックだけを受信する（include モード）というシグナルを送信できます。

IGMPv3 は、インターネット標準マルチキャスト（ISM）でも、Source Specific Multicast（SSM）でも動作できます。ISM では、exclude と include の両方のモードのレポートを適用できます。SSM では、ラストホップ ルータは include モードのレポートだけを受け入れます。exclude モードのレポートは無視されます。

IGMP のバージョンの違い

Internet Engineering Task Force（IETF）の Request for Comments（RFC）ドキュメントで定義されているように、IGMP には 3 種類のバージョンがあります。IGMPv2 は IGMPv1 の強化版で、ホストがマルチキャスト グループからの脱退を通知する機能が追加されていいます。IGMPv3 は IGMPv2 の強化版で、あるソース IP アドレスのセットから送信されたマルチキャストだけをリッスンする機能が追加されています。

表 *15 : IGMP* のバージョン

IGMP のバージョン	説明
IGMPv1	どのマルチキャスト グループがアクティブであるかをマルチキャストデバイスが判断できる基本的なクエリー応答メカニズムと、ホストがマルチキャスト グループに加入および脱退できるようにするためのその他のプロセスを提供します。RFC 1112 で、IP マルチキャスト用の IGMPv1 ホスト拡張が定義されています。
IGMPv2	IGMP の拡張で、IGMP の脱退処理、グループ固有のクエリーおよび明示的な最大応答時間フィールドなどの機能が可能になっています。また、IGMPv2 ではこの作業を実行するために、マルチキャスト プロトコルに依存することなく IGMP クエリアを選択する機能もデバイスに追加されます。IGMPv2 は RFC 2236 で定義されています。

（注） デフォルトでは、インターフェイスでPIMをイネーブルにすると、そのデバイスでIGMPv2がイネーブルになります。IGMPv2 は、可能な限り IGMPv1 と下位互換性を保つよう設計されました。この下位互換性を実現するために、RFC 2236 は特別な相互運用性ルールを定義しています。ネットワークにレガシー IGMPv1 ホストが含まれている場合は、これらの運用性ルールをよく知っておく必要があります。IGMPv1 と IGMPv2 の相互運用性の詳細については、RFC 2236『Internet Group Management Protocol, Version 2』を参照してください。

IGMPv1 を実行するデバイス

IGMPv1 デバイスは、「全ホスト」へのマルチキャスト アドレスである 224.0.0.1 に IGMP クエリーを送信して、アクティブ マルチキャスト レシーバが存在するマルチキャスト グループを求めます。マルチキャスト レシーバも、デバイスに IGMP レポートを送信して、特定のマルチキャスト ストリームの受信を待機していることを通知できます。ホストは非同期に、またはデバイスによって送信される IGMP クエリーに対応して、レポートを送信できます。同じマルチキャスト グループに複数のマルチキャスト レシーバが存在する場合、これらのホストの 1 つのみで、IGMP レポート メッセージが送信されます。他のホストでは、レポート メッセージが抑制されます。

IGMPv1 では、IGMP クエリア選択はありません。セグメント内に複数のデバイスがある場合、すべてのデバイスが定期的に IGMP クエリーを送信します。IGMPv1 には、ホストがグループから脱退できる特別なメカニズムはありません。ホストで、特定のグループに対するマルチキャスト パケットを受信する必要がなくなった場合は、デバイスから送信される IGMP クエリー パケットに対する応答を行わないだけです。デバイスはクエリー パケットを送信し続けます。デバイスが 3 回 IGMP クエリーの応答を受信しないと、グループはタイムアウトし、デバイスはグループのセグメントへのマルチキャスト パケットの送信を停止します。ホストがタイムアウト期間後にマルチキャスト パケットを受信する場合、そのホストは新しい IGMP join をデバイスに送信するだけです。これにより、デバイスはマルチキャスト パケットの転送を再開します。

LAN 上に複数のデバイスが存在する場合は、指定ルータ（DR）を選択して、接続されているホストに対するマルチキャスト トラフィックの重複を回避する必要があります。PIM デバイスは DR を選択する選定プロセスに従います。最も大きい IP アドレスを持つ PIM デバイスが DR になります。

DR は、次のタスクを担当します。

- PIM 登録メッセージ、PIM 加入メッセージ、および PIM プルーニング メッセージをランデブー ポイント（RP）に送信し、ホスト グループ メンバーシップに関する情報を通知する。
- IGMP ホスト クエリー メッセージを送信する。
- IGMP オーバーヘッドをホストおよびネットワークでできるだけ低く維持するために、ホスト クエリー メッセージをデフォルトで 60 秒ごとに送信する。

IGMPv2 を実行するデバイス

IGMPv2 では、IGMPv1 のクエリー メッセージング機能が改善されました。

IGMPv2 のクエリーおよびメンバーシップ レポート メッセージは、次の 2 つの例外を除き、IGMPv1 メッセージと同じです。

- IGMPv2 クエリー メッセージは、一般クエリー（IGMPv1 クエリーと同じ）とグループ固有クエリーの 2 つのカテゴリに分かれる。
- IGMPv1 メンバーシップ レポートと IGMPv2 メンバーシップ レポートの IGMP タイプ コードが異なる。

IGMPv2 では、次の機能に対するサポートを追加することにより、IGMP の機能の強化も行われました。

- クエリア選択プロセス：IGMPv2 デバイスが、プロセスを実行するマルチキャスト ルーティング プロトコルに依存せずに、IGMP クエリアを選択できる機能を提供します。
- [Maximum Response Time] フィールド：IGMP クエリアを使用して最大クエリー応答時間を指定できる、クエリー メッセージの新しいフィールド。このフィールドで、応答のバースト性を制御し、脱退遅延を調整するクエリー応答プロセスの調整ができます。
- グループ固有クエリー メッセージ：すべてのグループではなく特定の 1 つのグループでクエリー操作を実行する目的で、IGMP クエリアを使用することができます。
- グループ脱退メッセージ：グループから脱退することをネットワーク上のデバイスに通知する手段をホストに提供します。

DR と IGMP クエリアが通常同じデバイスである IGMPv1 とは異なり、IGMPv2 では 2 つの機能は分離されます。DR と IGMP クエリアは異なる基準で選択され、同じサブネット上の異なるデバイスである場合があります。DR はサブネットで IP アドレスが最大のデバイスで、IGMP クエリアは最小の IP アドレスを持つデバイスです。

次のように、クエリー メッセージは IGMP クエリアの選択に使用されます。

1. 各 IGMPv2 デバイスは起動時に、そのインターフェイス アドレスを一般クエリー メッセージのソース IP アドレス フィールドに使用して、当該メッセージを全システムのグループ アドレス 224.0.0.1 にマルチキャスト送信します。
2. IGMPv2 デバイスが一般クエリー メッセージを受信すると、デバイスは自分のインターフェイス アドレスとメッセージのソース IP アドレスを比較します。サブネット上の最下位 IP アドレスが使用されているデバイスにより、IGMP クエリアが選択されます。
3. すべてのデバイス（クエリアは除く）でクエリー タイマーが開始されます。IGMP クエリアから一般クエリー メッセージを受信するたびに、タイマーはリセットされます。クエリー タイマーが切れると、IGMP クエリアがダウンしたと見なされ、新しい IGMP クエリアを選択するために選択プロセスが再度実行されます。

デフォルトでは、タイマーはクエリー インターバルの 2 倍です。

IGMP の加入および脱退処理

IGMP の加入処理

ホストがマルチキャスト グループに加入するとき、ホストは、加入するマルチキャスト グループに 1 つ以上の送信要求されていないメンバーシップ レポートを送信します。IGMP 加入処理は、IGMPv1 ホストと IGMPv2 ホストで同じです。

IGMPv3 では、ホストの加入処理は次のように処理されます。

- ホストがグループに加入する場合は、空の EXCLUDE リストを使用して、224.0.0.22 に IGMPv3 メンバーシップ レポートを送信します。
- ホストが特定のチャネルに加入する場合は、特定のソース アドレスを含む INCLUDE リストを使用して、224.0.0.22 に IGMPv3 メンバーシップ レポートを送信します。
- ホストが特定のソースを除くグループに加入する場合は、これらのソースを EXCLUDE リストで除外して、224.0.0.22 に IGMPv3 メンバーシップ レポートを送信します。

（注） LAN 上にある一部の IGMPv3 ホストでソースが除外され、その他のホストで同じソースが含まれている場合、デバイスは LAN 上でそのソースのトラフィックを送信します（つまり、この場合、包含が除外より優先されます）。

IGMP の脱退処理

ホストがグループから脱退するために使用する方法は、動作中の IGMP のバージョンによって異なります。

IGMPv1 の脱退処理

IGMPv1 には、ホストがあるグループからのマルチキャスト トラフィックを受信しないことをそのサブネットのデバイスに通知するグループ脱退メッセージはありません。ホストでは、マルチキャスト グループに対するトラフィックの処理が停止するだけで、そのグループに対する IGMP メンバーシップ レポートを使用した IGMP クエリーへの応答が終了します。その結果、IGMPv1 デバイスがサブネットの特定のマルチキャスト グループにアクティブなレシーバがなくなったことを認識する唯一の方法は、デバイスがメンバーシップ レポートを受信しなくなったときになります。このプロセスを容易にするために、IGMPv1 デバイスは、サブネットの IGMP グループとカウント ダウン タイマーを関連付けます。サブネットのグループがメンバーシップ レポートを受信すると、タイマーがリセットされます。IGMPv1 デバイスでは、このタイム アウト間隔は通常クエリー間隔の 3 倍（3 分）です。このタイムアウト間隔は、すべてのホストがマルチキャスト グループから脱退した後最大 3 分間、デバイスがサブネットにマルチキャスト トラフィックを転送し続ける可能性があることを意味します。

IGMPv2 の脱退処理

IGMPv2には、特定のグループのマルチキャストトラフィックの受信を停止することをホストが提示する手段を提供するグループ脱退メッセージが組み込まれています。IGMPv2 ホストがマルチキャスト グループから脱退するとき、そのホストがそのグループのメンバーシップ レポートでクエリーに応答する最後のホストである場合、デバイス全体のマルチキャストグループ（224.0.0.2）にグループ脱退メッセージを送信します。

IGMPv3 の脱退処理

IGMPv3 は、IGMPv3 メンバーシップ レポートにソース、グループ、またはチャネルを含めるか除外することによって、ホストが特定のグループ、ソース、またはチャネルからのトラフィックの受信を停止できる機能を導入することで、脱退処理を拡張しています。

IGMP のデフォルト設定

次の表に、deviceの IGMP デフォルト設定を示します。

表 *16 : IGMP* のデフォルト設定

機能	デフォルト設定
マルチキャストグループのメンバーとしてのマルチレイヤdevice	グループ メンバーシップは未定義
マルチキャスト グループへのアクセス	インターフェイスのすべてのグループを許可
IGMP のバージョン	すべてのインターフェイスでバージョン 2
IGMP ホストクエリー メッセージ インターバル	すべてのインターフェイスで 60 秒
IGMP クエリー タイムアウト	すべてのインターフェイスで 60 秒
IGMP 最大クエリー応答時間	すべてのインターフェイスで 10 秒
静的に接続されたメンバーとしてのマルチレイヤdevice	ディセーブル

IGMP の設定方法

グループのメンバーとしてデバイスを設定

deviceをマルチキャストグループのメンバーとして設定し、マルチキャストがネットワークに到達可能かどうかを検出できます。管理対象のすべてのマルチキャスト対応ルータおよびマルチレイヤdevicesがマルチキャストグループのメンバーである場合、グループに ping を送信す

ると、これらのすべてのデバイスが応答します。デバイスは、所属グループにアドレス指定された ICMP エコー要求パケットに応答します。もう 1 つの例は、ソフトウェア付属のマルチキャスト トレースルート ツールです。

⚠

注意 この手順を実行すると、グループ アドレス用のデータ トラフィックがすべて CPU に送られるため、CPU のパフォーマンスが低下する場合があります。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp join-group** *group-address*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	マルチキャスト ルーティングをイネーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip igmp join-group** *group-address* 例： `スイッチ(config-if)# ip igmp`	deviceをマルチキャストグループに加入するように設定します。 デフォルトで、グループのメンバーシップは定義されていません。

	コマンドまたはアクション	目的
	`join-group 225.2.2.2`	*group-address* には、マルチキャスト IP アドレスをドット付き 10 進表記で指定します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show ip igmp interface** [*interface-id*] 例： `スイッチ# show ip igmp interface`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IP マルチキャスト グループへのアクセスの制御

スイッチは IGMP ホストクエリー メッセージを送信し、接続されたローカル ネットワーク上のメンバーが属しているマルチキャスト グループを判別します。次に、スイッチは、マルチキャスト グループにアドレス指定されたすべてのパケットをこれらのグループ メンバーに転送します。インターフェイスごとにフィルタを適用し、インターフェイスで処理されるサブネット上のホストが加入可能なマルチキャスト グループを制限できます。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp access-group** *access-list-number*
5. **exit**
6. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
7. **end**
8. **show ip igmp interface** [*interface-id*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface GigabitEthernet 1/0/12`	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip igmp access-group** *access-list-number* 例： `スイッチ(config-if)# ip igmp access-group 10`	インターフェイスで処理されるサブネット上のホストが加入できるマルチキャスト グループを指定します。 デフォルトでは、インターフェイスのすべてのグループが許可されています。 *access-list-number* には、IP 標準アドレス アクセス リスト番号を指定します。 指定できる範囲は 1 ～ 199 です。 （注） インターフェイスでグループをディセーブルにするには、 **no ip igmp access-group** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： `スイッチ(config)# access-list 10 permit`	標準アクセス リストを作成します。 • *access-list-number* には、ステップ 3 で作成したアクセス リストを指定します。

	コマンドまたはアクション	目的
		• **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *source* には、サブネット上のホストが加入できるマルチキャスト グループを指定します。 • （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。
ステップ 7	**end** 例： `スイッチ(config-igmp-profile)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show ip igmp interface** [*interface-id*] 例： `スイッチ# show ip igmp interface`	入力を確認します。

IGMP バージョンの変更

スイッチでは、IGMP クエリー タイムアウトや最大クエリー応答時間などの機能を使用できる IGMP バージョン 2 がデフォルトで使用されます。

サブネット上のすべてのシステムで、同じバージョンをサポートする必要があります。スイッチは自動的にバージョン 1 のシステムを検出せず、バージョン 1 へのスイッチングも行いません。バージョン 2 のルータまたはスイッチは、常に IGMPv1 ホストと正しく連動しているため、バージョン 1 とバージョン 2 のホストはサブネット上で混在できます。

使用しているホストでバージョン 2 がサポートされていない場合は、スイッチをバージョン 1 に設定してください。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**

3. **interface** *interface-id*
4. **ip igmp version** {**1** | **2** | **3** }
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip igmp version** {**1**	**2**	**3** } 例： `スイッチ(config-if)# ip igmp version 2`	スイッチで使用する IGMP バージョンを指定します。 （注） バージョン 1 に変更すると、 **ip igmp query-interval** および **ip igmp query-max-response-time** インターフェイス コンフィギュレーション コマンドを設定できません。 デフォルトの設定に戻す場合は、 **no ip igmp version** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 6	**show ip igmp interface** [*interface-id*] 例： スイッチ# **show ip igmp interface**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP ホストクエリー メッセージ インターバルの変更

deviceは、IGMP ホストクエリメッセージを定期的に送信し、接続されたネットワーク上にあるマルチキャストグループを検出します。これらのメッセージは、TTL が 1 の全ホスト マルチキャスト グループ（224.0.0.1）に送信されます。deviceはホストクエリメッセージを送信し、ネットワーク上に存在するメンバーシップに関する情報をリフレッシュします。クエリーをいくつか実行したあとで、マルチキャスト グループのメンバーであるローカルホストが存在しないことをソフトウェアが検出した場合、そのグループのリモート送信元からローカル ネットワークへのマルチキャストパケット転送が停止され、プルーニング メッセージが送信元のアップストリーム方向へ送信されます。

deviceは LAN（サブネット）用の PIM DR を選択します。DR は、LAN 上のすべてのホストに IGMP ホストクエリー メッセージを送信します。SM の場合、DR は PIM 登録メッセージおよび PIM Join メッセージも RP ルータに向けて送信します。IGMPv2 では、DR は IP アドレスが最大である、ルータまたはマルチレイヤdeviceです。IGMPv1 では、DR は LAN 上で動作するマルチキャスト ルーティング プロトコルに従って選択されます。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp query-interval** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	マルチキャスト ルーティングをイネーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip igmp query-interval** *seconds* 例： `スイッチ(config-if)# ip igmp query-interval 75`	DR が IGMP ホストクエリー メッセージを送信する頻度を設定します。 デフォルトでは、DR は IGMP ホストクエリー メッセージを 60 秒ごとに送信し、ホストおよびネットワークでの IGMP オーバーヘッドを抑制します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show ip igmp interface** [*interface-id*] 例： `スイッチ# show ip igmp interface`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMPv2 の IGMP クエリー タイムアウトの変更

IGMPv2 を使用している場合、deviceがインターフェイスのクエリアとして引き継ぐまでの時間を指定できます。デフォルトでは、deviceは **ip igmp query-interval** インターフェイス コンフィギュレーション コマンドによって制御されるクエリインターバルの2倍の時間だけ待機します。この時間を経過しても、deviceがクエリを受信しない場合は、スイッチがクエリアになります。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp querier-timeout** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	マルチキャスト ルーティングをイネーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip igmp querier-timeout** *seconds* 例： `スイッチ(config-if)# ip igmp querier-timeout 120`	IGMP クエリー タイムアウトを指定します。 デフォルトは 60 秒です（クエリー インターバルの 2 倍）。指定できる範囲は 60 ～ 300 です。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ip igmp interface** [*interface-id*] 例： スイッチ# **show ip igmp interface**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMPv2 の最大クエリー応答時間の変更

IGMPv2 を使用している場合は、IGMP クエリーでアドバタイズされる最大クエリー応答時間を変更できます。deviceは最大クエリ応答時間を使用し、LAN 上に直接接続されたグループメンバーが存在しないことを短時間で検出します。値を小さくすると、deviceによるグループのプルーニング速度が向上します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp query-max-response-time** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	マルチキャスト ルーティングをイネーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip igmp query-max-response-time** *seconds* 例： スイッチ(config-if)# **ip igmp query-max-response-time 15**	IGMP クエリーでアドバタイズされる最大クエリー応答時間を変更します。 デフォルトは 10 秒です。指定できる範囲は 1 ～ 25 です。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ip igmp interface** [*interface-id*] 例： スイッチ# **show ip igmp interface**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

静的に接続されたメンバーとしてデバイスを設定

ネットワーク セグメント上にグループ メンバが存在しなかったり、ホストで IGMP を使用してグループ メンバーシップを報告できないことがあります。しかし、そのネットワーク セグ

メントに対して、マルチキャスト トラフィックの送信が必要な場合もあります。マルチキャスト トラフィックをネットワーク セグメントに送り込むには、次のコマンドを使用します。

- **ip igmp join-group**：deviceはマルチキャストパケットの転送だけでなく、マルチキャストパケットを受け入れます。マルチキャスト パケットを受信すると、deviceは高速スイッチングを実行できません。
- **ip igmp static-group**：deviceはパケットを転送するだけで、パケット自体は受け入れません。この方法を使用すると、高速スイッチングが可能です。発信インターフェイスがIGMPキャッシュに格納されますが、マルチキャスト ルート エントリに「L」（ローカル）フラグが付かないことからも明らかなように、device自体はメンバーではありません。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp static-group** *group-address*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	マルチキャスト ルーティングをイネーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip igmp static-group** *group-address* 例：	deviceを静的に接続されたグループのメンバーとして設定します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **ip igmp static-group 239.100.100.101**	デフォルトでは、この機能はディセーブルになっています。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ip igmp interface** [*interface-id*] 例： スイッチ# **show ip igmp interface gigabitethernet 1/0/1**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP のモニタリング

IP ルーティング テーブル、キャッシュ、データベースの内容など、特定の統計情報を表示できます。

（注） このリリースでは、ルート単位の統計情報がサポートされていません。

また、リソースの使用状況を学習し、ネットワーク問題を解決するための情報を表示することもできます。さらに、ノードの到達可能性に関する情報を表示し、そのパケットが経由するネットワーク内のパスを検出することもできます。

次の表に示す特権EXECコマンドのいずれかを使用すると、さまざまなルーティング統計情報を表示できます。

表 *17:* システムおよびネットワーク統計情報を表示するコマンド

コマンド	目的
show ip igmp groups [*type-number* | *detail*]	deviceに直接接続され、IGMP によって取得されたマルチキャスト グループを表示します。

コマンド	目的
show ip igmp interface [*type number*]	インターフェイスのマルチキャスト関連情報を表示します。
show ip igmp profile [*profile_number*]	IGMP プロファイル情報を表示します。
show ip igmp ssm-mapping [*hostname/IP address*]	IGMP SSM マッピング情報を表示します。
show ip igmp static-group {**class-map** [**interface** [*type*]]	スタティック グループ情報を表示します。
show ip igmp vrf	選択したVPNルーティング/転送インスタンスを名前別に表示します。

IGMP の設定例

例：マルチキャストグループのメンバーとしてデバイスを設定

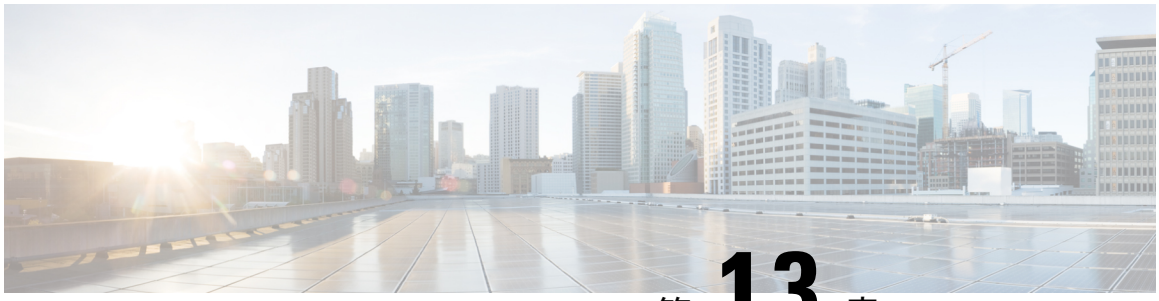
次に、マルチキャストグループ 255.2.2.2 へのdevice加入を許可する例を示します。

```
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# ip igmp join-group 255.2.2.2
スイッチ(config-if)#
```

例：IP マルチキャスト グループへのアクセスの制御

次に、ポートに接続されたホストが、グループ255.2.2.2にだけ加入できるように設定する例を示します。

```
Switch(config)# access-list 1 255.2.2.2 0.0.0.0
Switch(config-if)# interface gigabitethernet1/0/1
Switch(config-if)# ip igmp access-group 1
```



第 13 章

IGMP スヌーピングおよびマルチキャスト VLAN レジストレーションの設定

IGMP スヌーピングおよび MVR の設定の前提条件

IGMP スヌーピングの前提条件

IGMP スヌーピング クエリアを設定するときには、次の注意事項を順守します。

- VLAN をグローバル コンフィギュレーション モードに設定してください。
- IP アドレスおよび VLAN インターフェイスを設定してください。IGMP スヌーピング クエリアは、イネーブルの場合この IP アドレスをクエリーの送信元アドレスとして使用します。
- VLAN インターフェイス上で IP アドレスが設定されていない場合、IGMP スヌーピング クエリアはIGMP クエリア用に設定されたグローバル IP アドレスを使用しようとします。グローバル IP アドレスが指定されていない場合、IGMP クエリアは VLAN device仮想インターフェイス（SVI）IP アドレス（存在する場合）を使用しようとします。SVI IP アドレスが存在しない場合、deviceはdevice上で設定された利用可能な最初の IP アドレスを使用します。利用可能な最初の IP アドレスは、**show ip interface** 特権 EXEC コマンドの出力に表示されます。IGMP スヌーピングクエリアはdevice上で利用可能な IP アドレスを検出できない場合、IGMP 一般クエリーを生成しません。
- IGMP スヌーピング クエリアは IGMP バージョン 1 および 2 をサポートします。

- 管理上イネーブルである場合、IGMP スヌーピング クエリアはネットワークにマルチキャスト ルータの存在を検出すると、非クエリア ステートになります。
- 管理上イネーブルである場合、IGMP スヌーピング クエリアは操作上、次の状況でディセーブル ステートになります。
  - IGMP スヌーピングが VLAN でディセーブルの場合
  - PIM が、VLAN に対応する SVI でイネーブルの場合
-
-

MVR の前提条件

マルチキャスト VLAN レジストレーション（MVR）の前提条件は次のとおりです。

- MVR を使用するには、deviceが LAN Base イメージを実行している必要があります。

IGMP スヌーピングおよび MVR の設定の制約事項

IGMP スヌーピングの制約事項

次に、IGMP スヌーピングの制約事項を示します。

- スイッチは同種スタックおよび混合スタック構成をサポートします。混合スタック構成は、Catalyst 2960-S スイッチだけでサポートされます。同種スタックは 8 つまで、混合スタックは4つまでのスタックメンバを持つことができます。スイッチスタック内のすべてのスイッチが LAN Base イメージを実行している必要があります。
- IGMP フィルタリングまたはマルチキャスト VLAN レジストレーション（MVR）が実行されている devices は、IGMPv3 Join および Leave メッセージをサポートしません。
- IGMP レポート抑制は、マルチキャストクエリに IGMPv1 レポートと IGMPv2 レポートがある場合にだけサポートされます。この機能は、クエリに IGMPv3 レポートが含まれている場合はサポートされません。
- IGMP の脱退時間の設定は、IGMP バージョン 2 が稼働しているホストでのみサポートされます。IGMP バージョン 2 は device のデフォルトバージョンです。

  ネットワークで実際の脱退にかかる待ち時間は、通常、設定した脱退時間どおりになります。ただし、脱退時間は、リアルタイムの CPU の負荷の状態、およびネットワークの遅延状態、インターフェイスから送信されたトラフィック量によって、設定された時間を前後することがあります。
- IGMP スロットリング アクションの制約事項は、レイヤ 2 ポートにだけ適用されます。**ip igmp max-groups action replace** インターフェイス コンフィギュレーション コマンドは論

理 EtherChannel インターフェイスで使用できますが、EtherChannel ポート グループに属するポートでは使用できません。

グループの最大数に関する制限がデフォルト（制限なし）に設定されている場合、**ip igmp max-groups action** {**deny** | **replace**} コマンドを入力しても効果はありません。

インターフェイスによりマルチキャスト エントリが転送テーブルに追加されてから、スロットリング アクションを設定し、グループの最大数の制限を設定すると、転送テーブルのエントリは、スロットリング アクションに応じて期限切れになるか削除されます。

MVR の制約事項

次に、MVR の制約事項を示します。

- MVR に参加するのは、レイヤ 2 ポートだけです。ポートを MVR 受信ポートとして設定する必要があります。
- 各 device または device スタックでサポートされる MVR マルチキャスト VLAN は、1 つのみです。
- 受信ポートはアクセス ポートでなければなりません。トランク ポートにはできません。device のレシーバポートは異なる VLAN に属していてもかまいませんが、マルチキャスト VLAN に属することはできません。
- device上で設定可能なマルチキャスト エントリ（MVR グループ アドレス）の最大数（つまり、受信可能な TV チャネルの最大数）は、256 です。
- 送信元 VLAN で受信され、レシーバポートから脱退する MVR マルチキャストデータは、deviceで存続可能時間（TTL）が 1 だけ少なくなります。
- device 上の MVR は、MAC マルチキャストアドレスではなく IP マルチキャストアドレスを使用するので、device 上でエイリアス IP マルチキャストアドレスを使用できます。ただし、deviceが Catalyst 3550 または Catalyst 3500 XL devicesと連携動作している場合は、それらの間でエイリアスとして使用される IP アドレスや予約済みの IP マルチキャストアドレス（224.0.0.xxx 範囲内）を設定する必要はありません。
- プライベート VLAN ポートに MVR を設定しないでください。
- device 上でマルチキャストルーティングがイネーブルの場合、MVR はサポートされません。MVR がイネーブルの場合に、マルチキャスト ルーティングおよびマルチキャスト ルーティング プロトコルをイネーブルにすると、MVR がディセーブルになり、警告メッセージが表示されます。マルチキャスト ルーティングおよびマルチキャスト ルーティング プロトコルがイネーブルの場合に、MVR をイネーブルにしようとすると、MVR をイネーブルにする操作が取り消され、エラー メッセージが表示されます。
- MVR 受信ポートで受信した MVR データは、MVR 送信元ポートに転送されません。
- MVR は IGMPv3 メッセージをサポートしていません。

- スイッチは同種スタックおよび混合スタック構成をサポートします。混合スタック構成は、Catalyst 2960-S スイッチだけでサポートされます。同種スタックは 8 つまで、混合スタックは4つまでのスタックメンバを持つことができます。スイッチスタック内のすべてのスイッチが LAN Base イメージを実行している必要があります。

IGMP スヌーピングおよび MVR に関する情報

IGMP スヌーピング

レイヤ 2 devices は IGMP スヌーピングを使用して、レイヤ 2 インターフェイスを動的に設定し、マルチキャストトラフィックが IP マルチキャストデバイスと対応付けられたインターフェイスにのみ転送されるようにすることによって、マルチキャストトラフィックのフラッディングを制限できます。名称が示すとおり、IGMP スヌーピングの場合は、LAN device でホストとルータ間の IGMP 伝送をスヌーピングし、マルチキャストグループとメンバポートを追跡する必要があります。device が特定のマルチキャストグループについて、ホストから IGMP レポートを受信した場合、device はホストのポート番号を転送テーブルエントリに追加します。ホストから IGMP Leave Group メッセージを受信した場合は、テーブルエントリからホストポートを削除します。マルチキャスト クライアントから IGMP メンバーシップ レポートを受信しなかった場合にも、スイッチはエントリを定期的に削除します。

（注） IP マルチキャストおよび IGMP の詳細については、RFC 1112 および RFC 2236 を参照してください。

上に設定されたマルチキャストルータ は、すべての VLAN に一般的なクエリを定期的に送信しますこのマルチキャスト トラフィックに関心のあるホストはすべて Join 要求を送信し、転送テーブルのエントリに追加されます。device は、IGMP Join 要求の送信元となる各グループの IGMP スヌーピング IP マルチキャスト転送テーブルで、VLAN ごとに 1 つずつエントリを作成します。

device は、MAC アドレスに基づくグループではなく、IP マルチキャストグループに基づくブリッジングをサポートしています。マルチキャスト MAC アドレスに基づくグループの場合、設定されている IP アドレスを設定済みの MAC アドレス（エイリアス）または予約済みのマルチキャスト MAC アドレス（224.0.0.xxx の範囲内）に変換すると、コマンドがエラーになります。device では IP マルチキャスト グループを使用するので、アドレス エイリアスの問題は発生しません。

IGMP スヌーピングによって、IP マルチキャスト グループは動的に学習されます。ただし、**ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id* グローバル コンフィギュレーション コマンドを使用すると、マルチキャストグループを静的に設定できます。グループ メンバーシップをマルチキャスト グループ アドレスに静的に指定すると、その設定値は IGMP スヌーピングによる自動操作より優先されます。マルチキャスト グループ メンバーシップの

リストは、ユーザが定義した設定値およびIGMP スヌーピングによって学習された設定値の両方で構成できます。

マルチキャスト トラフィックはルーティングする必要がないのでマルチキャスト インターフェイスを使用せずに、サブネットの IGMP スヌーピングをサポートするよう IGMP スヌーピング クエリーを設定できます。

ポート スパニングツリー、ポート グループ、または VLAN ID が変更された場合、VLAN 上のこのポートから IGMP スヌーピングで学習されたマルチキャスト グループは削除されます。

ここでは、IGMP スヌーピングの特性について説明します。

IGMP のバージョン

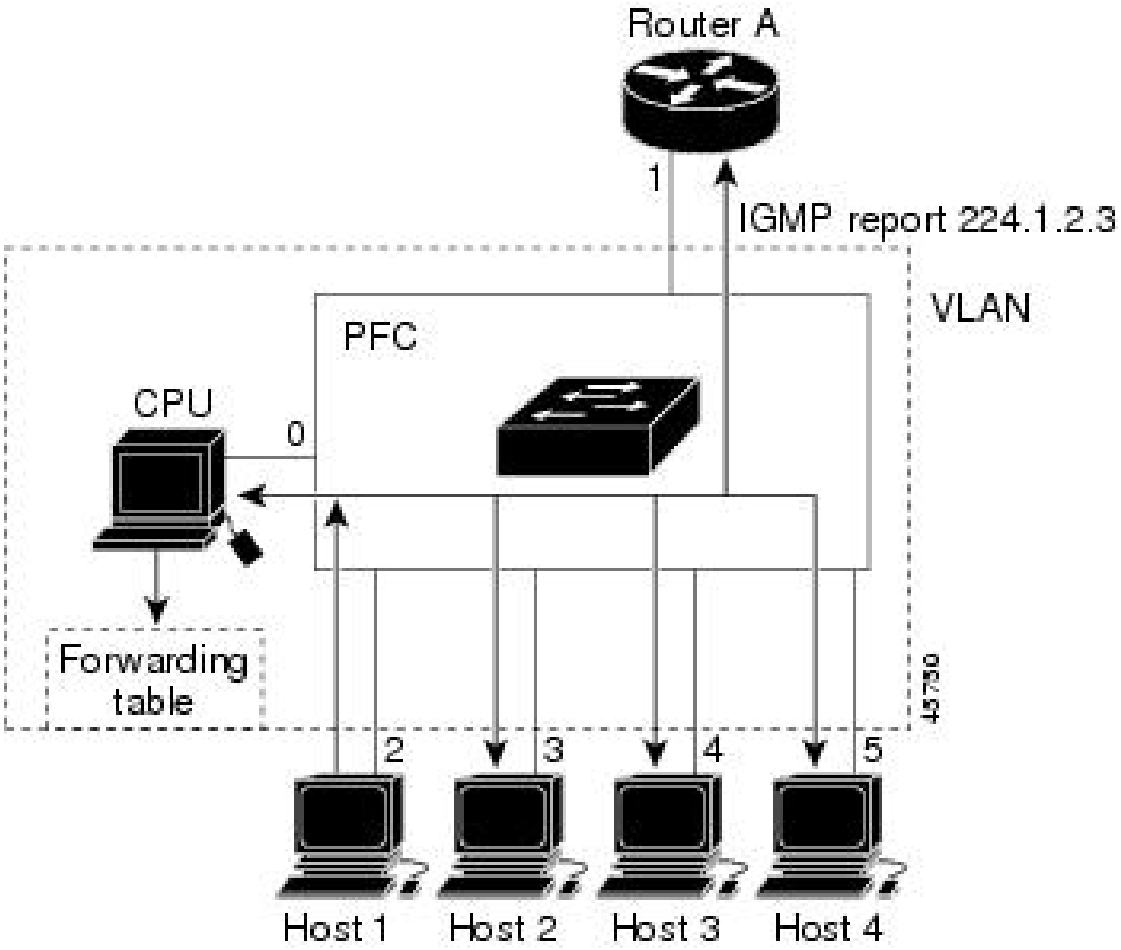
device は、IGMP バージョン 1、IGMP バージョン 2、および IGMP バージョン 3 をサポートしています。これらバージョンは、 device 上でそれぞれ相互運用できます。たとえば、IGMP スヌーピングがイネーブルになっており、クエリアのバージョンが IGMPv2 で、device がホストから IGMPv3 レポートを受信している場合、device は IGMPv3 レポートをマルチキャスト ルータに転送できます。

IGMPv3 device は、Source Specific Multicast（SSM）機能を実行しているデバイスとメッセージの送受信を行うことができます。

マルチキャスト グループへの加入

図 *3 :* 最初の *IGMP Join* メッセージ

device に接続したホストが IP マルチキャスト グループに加入し、なおかつそのホストが IGMP バージョン 2 クライアントの場合、ホストは加入する IP マルチキャスト グループを指定した非送信請求 IGMP Join メッセージを送信します。別の方法として、ルータから一般クエリを受信した device は、そのクエリーを VLAN 内のすべてのポートに転送します。IGMP バージョン 1 またはバージョン 2 のホストがマルチキャスト グループに加入する場合、ホストは device に Join メッセージを送信することによって応答します。device の CPU は、そのグループのマルチキャスト転送テーブル エントリがまだ存在していないのであれば、エントリを作成します。CPU はさらに、Join メッセージを受信したインターフェイスを転送テーブル エントリに追加します。そのインターフェイスと対応付けられたホストが、そのマルチキャスト グループ用のマルチキャスト トラフィックを受信します。



ルータ A が device に一般クエリを送信し、そこでそのクエリは同じ VLAN のすべてのメンバであるポート 2 ～ 5 に転送されます。ホスト 1 はマルチキャスト グループ 224.1.2.3 に加入するために、グループに IGMP メンバーシップ レポート（IGMP Join メッセージ）をマルチキャストします。device の CPU は IGMP レポートの情報を使用して、転送テーブルのエントリを設定します。転送テーブルにはホスト 1 およびルータに接続しているポート番号が含まれます。

表 *18 : IGMP* スヌーピング転送テーブル

宛先アドレス	パケットのタイプ	ポート
224.1.2.3	IGMP	1、2

deviceのハードウェアは、IGMP 情報パケットをマルチキャスト グループの他のパケットと区別できます。テーブルの情報は、224.1.2.3 マルチキャスト IP アドレス宛ての、IGMP パケットではないフレームを、ルータおよびグループに加入したホストに対して送信するように、スイッチング エンジンに指示します。

図 *4 : 2* 番目のホストのマルチキャスト グループへの加入

別のホスト（たとえば、ホスト 4）が、同じグループ用に非送信請求 IGMP Join メッセージを送信する場合、CPUがそのメッセージを受け取り、ホスト4のポート番号を転送テーブルに追加します。転送テーブルはCPU宛てだけにIGMPメッセージを送るので、メッセージはdeviceの他のポートにフラッディングされません。認識されているマルチキャストトラフィックは、CPU 宛てではなくグループ宛てに転送されます。

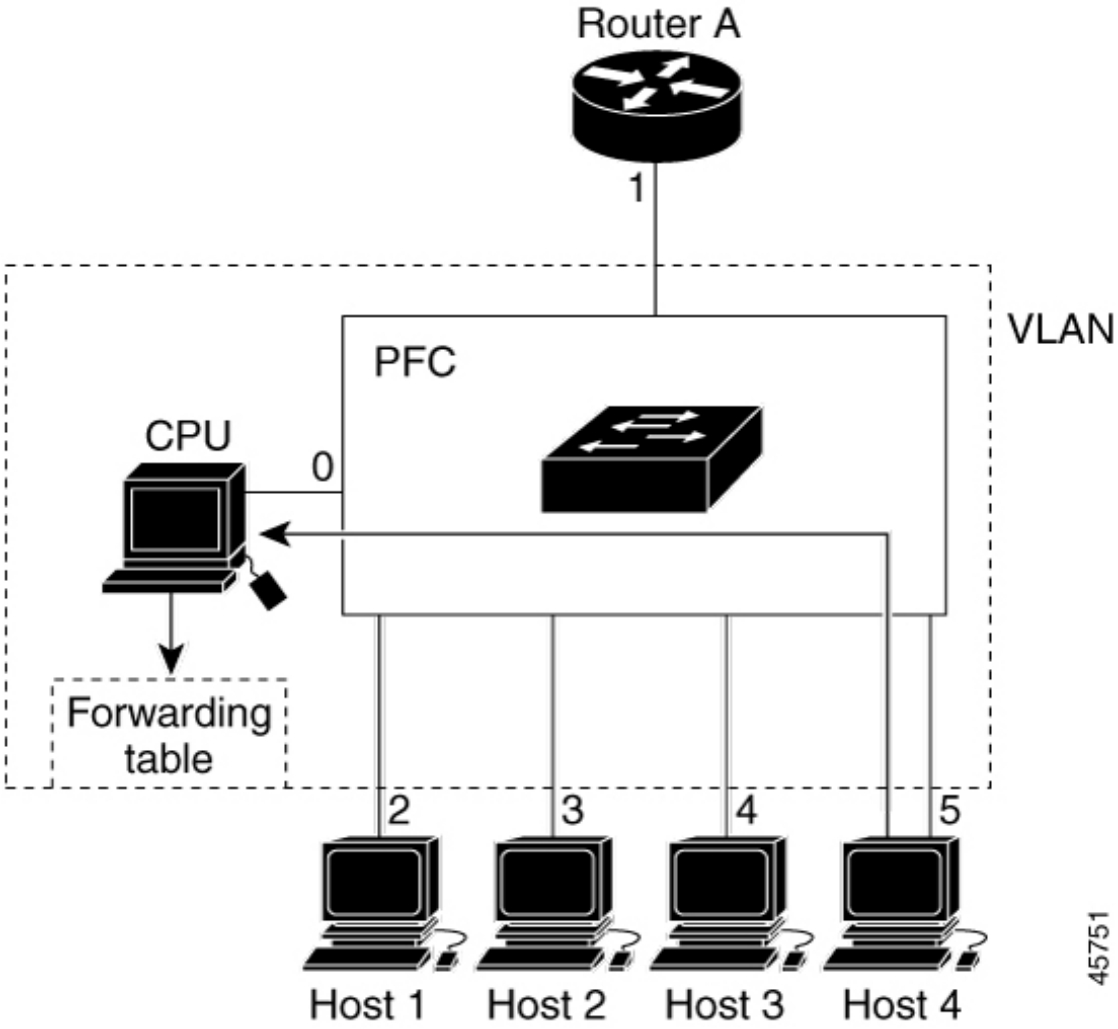


表 *19 :* 更新された *IGMP* スヌーピング転送テーブル

宛先アドレス	パケットのタイプ	ポート
224.1.2.3	IGMP	1, 2, 5

マルチキャスト グループからの脱退

ルータはマルチキャスト一般クエリを定期的に送信し、deviceはそれらのクエリをVLANのすべてのポートを通じて転送します。関心のあるホストがクエリーに応答します。VLAN内の少

なくとも 1 つのホストがマルチキャスト トラフィックを受信するようなら、ルータは、その VLAN へのマルチキャスト トラフィックの転送を続行します。device は、その IGMP スヌーピングによって維持された IP マルチキャストグループの転送テーブルで指定されたホストに対してだけ、マルチキャスト グループ トラフィックを転送します。

ホストがマルチキャスト グループから脱退する場合、何も通知せずに脱退することも、Leave メッセージを送信することもできます。ホストから Leave メッセージを受信した device は、グループ固有のクエリーを送信して、そのインターフェイスに接続された他のデバイスが所定のマルチキャストグループのトラフィックに関与しているかどうかを学習します。device はさらに、転送テーブルでその MAC グループの情報を更新し、そのグループのマルチキャストトラフィックの受信に関心のあるホストだけが、転送テーブルに指定されるようにします。ルータが VLAN からレポートを受信しなかった場合、その VLAN 用のグループは IGMP キャッシュから削除されます。

即時脱退

device は IGMP スヌーピングの即時脱退を使用して、先に device からインターフェイスにグループ固有のクエリーを送信しなくても、Leave メッセージを送信するインターフェイスを転送テーブルから削除できるようにします。VLAN インターフェイスは、最初の Leave メッセージで指定されたマルチキャストグループのマルチキャストツリーからプルーニングされます。即時脱退によって、複数のマルチキャストグループが同時に使用されている場合でも、スイッチド ネットワークのすべてのホストに最適な帯域幅管理が保証されます。

即時脱退機能をサポートするのは、IGMP バージョン 2 が稼働しているホストだけです。IGMP バージョン 2 は device のデフォルトバージョンです。

（注） 即時脱退機能を使用するのは、各ポートに接続されているホストが 1 つだけの VLAN に限定してください。ポートに複数のホストが接続されている VLAN 上で即時脱退をイネーブルにすると、一部のホストが誤ってドロップされる可能性があります。

IGMP 脱退タイマーの設定

まだ指定のマルチキャストグループに関心があるかどうかを確認するために、グループ固有のクエリーを送信した後の device の待機時間を設定できます。IGMP 脱退応答時間は、100 ～ 32767 ミリ秒の間で設定できます。

IGMP レポート抑制

（注） IGMP レポート抑制は、マルチキャストクエリに IGMPv1 レポートと IGMPv2 レポートがある場合にだけサポートされます。この機能は、クエリに IGMPv3 レポートが含まれている場合はサポートされません。

device は IGMP レポート抑制を使用して、マルチキャストルータクエリごとに 1 つの IGMP レポートのみをマルチキャストデバイスに転送します。IGMP レポート抑制がイネーブル（デフォルト）である場合、device は最初の IGMP レポートをグループのすべてのホストからすべてのマルチキャストルータに送信します。deviceは、グループの残りの IGMP レポートをマルチキャストルータに送信しません。この機能により、マルチキャストデバイスにレポートが重複して送信されることを防ぎます。

マルチキャストルータクエリに IGMPv1 および IGMPv2 レポートに対する要求のみが含まれている場合、device は最初の IGMPv1 レポートまたは IGMPv2 レポートのみを、グループのすべてのホストからすべてのマルチキャストルータに転送します。

マルチキャストルータクエリに IGMPv3 レポートに対する要求も含まれる場合、deviceはグループのすべての IGMPv1、IGMPv2、および IGMPv3 レポートをマルチキャストデバイスに転送します。

IGMP レポート抑制をディセーブルにすると、すべての IGMP レポートはマルチキャスト ルータに転送されます。

IGMP スヌーピングのデフォルト設定

次の表に、device の IGMP スヌーピングのデフォルト設定を示します。

表 *20 : IGMP* スヌーピングのデフォルト設定

機能	デフォルト設定
IGMP スヌーピング	グローバルおよび VLAN 単位でイネーブル
マルチキャスト ルータ	未設定
IGMP スヌーピング即時脱退	ディセーブル
スタティック グループ	未設定
TCN[1]フラッド クエリ カウント	2
TCN クエリー送信要求	ディセーブル
IGMP スヌーピング クエリア	ディセーブル
IGMP レポート抑制	有効

[1] (1) TCN = トポロジ変更通知

マルチキャスト VLAN レジストレーション

マルチキャスト VLAN レジストレーション（MVR）は、イーサネット リング ベースのサービス プロバイダ ネットワーク上でマルチキャスト トラフィックの広範囲展開を使用するアプリケーション（サービス プロバイダ ネットワーク上の複数の TV チャネルのブロードキャストなど）用に設計されています。MVR によってポート上の加入者は、ネットワークワイドなマ

ルチキャスト VLAN 上のマルチキャスト ストリームに加入し、脱退できます。また、ネットワーク上で 1 つのマルチキャスト VLAN を共有しながら、加入者が別の VLAN に接続できます。MVR によって、マルチキャスト VLAN でマルチキャスト ストリームを連続送信する能力が得られますが、ストリームと加入者の VLAN は、帯域幅およびセキュリティ上の理由で分離されます。

ここでは、MVR について説明します。

MVR と IGMP

（注） device上で、MVR は IGMP スヌーピングと共存できます。

MVR では、加入者ポートが IGMP Join および Leave メッセージを送信することによって、マルチキャスト ストリームへの加入および脱退（Join および Leave）を行うことが前提です。これらのメッセージは、イーサネットで接続され、IGMP バージョン 2 に準拠しているホストから発信できます。MVR は IGMP スヌーピングの基本メソッドで動作しますが、この 2 つの機能はそれぞれ単独で動作します。それぞれ他方の機能の動作に影響を与えずに、イネーブルまたはディセーブルにできます。ただし、IGMP スヌーピングと MVR が両方ともイネーブルの場合、MVR は MVR 環境で設定されたマルチキャスト グループが送信した Join および Leave メッセージだけに反応します。他のマルチキャスト グループから送信された Join および Leave メッセージはすべて、IGMP スヌーピングが管理します。

device の CPU は、MVR IP マルチキャストストリームとそれに対応する device 転送テーブル内の IP マルチキャストグループを識別し、IGMP メッセージを代行受信し、転送テーブルを変更して、マルチキャストストリームの受信側としての加入者を追加または削除します。受信側が送信元と異なる VLAN 上に存在している場合でも同じです。この転送動作により、異なる VLAN の間でトラフィックを選択して伝送できます。

動作モード

device の MVR 動作は、互換モードまたはダイナミックモードに設定できます。

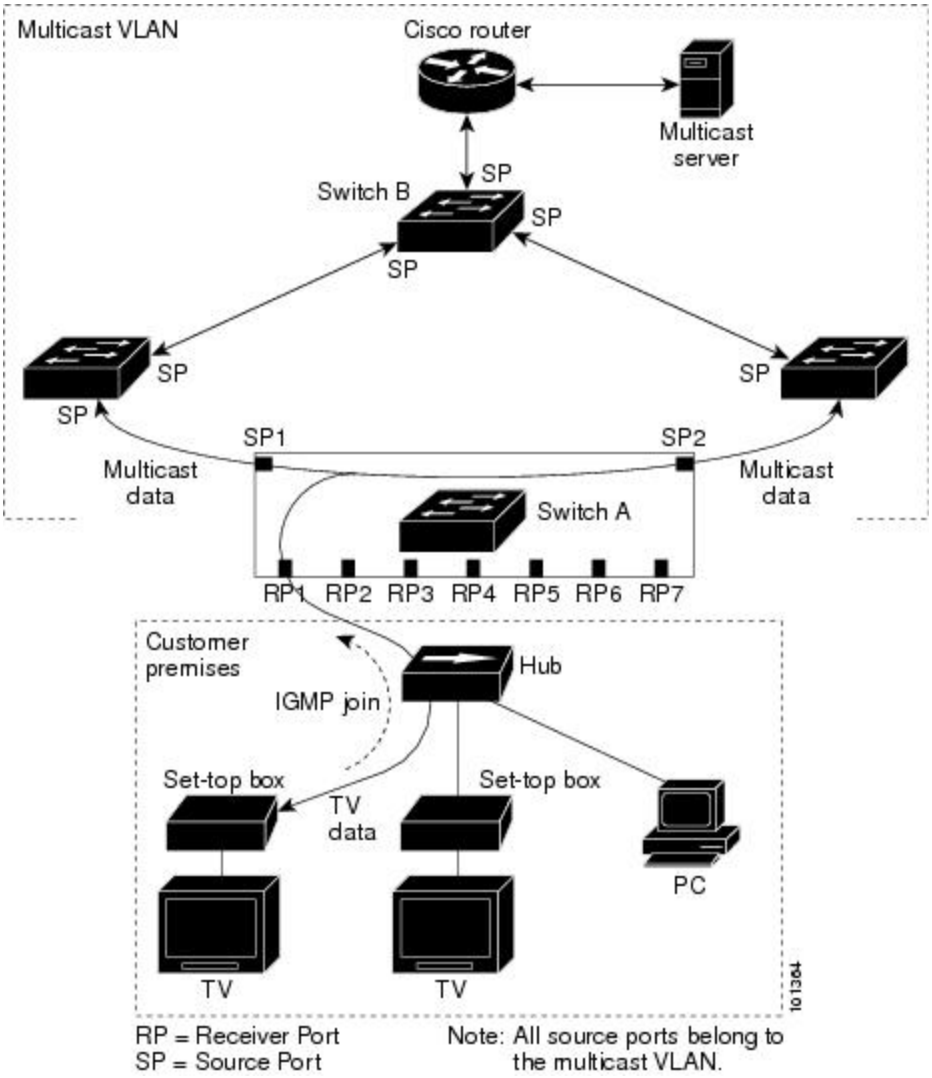
- 互換モードの場合、MVR ホストが受信したマルチキャスト データはすべての MVR データ ポートに転送されます。MVR データ ポートの MVR ホスト メンバーシップは無関係です。マルチキャスト データは、IGMP レポートまたは静的な MVR 設定のどちらかによって、MVR ホストが加入しているレシーバ ポートだけに転送されます。MVR ホストから受信した IGMP レポートが、device に設定された MVR データポートから転送されることはありません。
- ダイナミックモードの場合、device 上の MVR ホストが受信したマルチキャストデータは、IGMP レポートまたは静的な MVR 設定のどちらかによって、MVR ホストが加入している MVR データおよびクライアントポートから転送されます。それ以外のポートからは転送されません。MVR ホストから受信した IGMP レポートも、ホストのすべての MVR データ ポートから転送されます。したがって、互換モードで device を稼働させた場合と異なり、MVR データポートリンクで不要な帯域幅を使用しなくてすみます。

マルチキャスト TV アプリケーションでの MVR

マルチキャスト TV アプリケーションでは、PC またはセットトップ ボックスを装備したテレビでマルチキャスト ストリームを受信できます。1 つのサブスクライバポートに複数のセットトップボックスまたは PC を接続できます。サブスクライバポートは、MVR 受信ポートとして設定された device ポートです。

図 *5 :* マルチキャスト *VLAN* レジストレーションの例

次に、設定例を示します。



この設定例では、Dynamic Host Configuration Protocol（DHCP）によって、セットトップ ボックスまたは PC に IP アドレスが割り当てられます。加入者がチャネルを選択すると、適切なマルチキャストに加入するために、セットトップ ボックスまたは PC からスイッチ A に IGMP レポートが送信されます。IGMP レポートが設定済みの IP マルチキャスト グループ アドレスのいずれかに一致する場合、device CPU は、この受信ポートおよび VLAN が指定のマルチキャスト ストリームの転送先として含まれるように、マルチキャスト VLAN からマルチキャスト

ストリームを受信した際に、ハードウェア アドレス テーブルを変更します。マルチキャスト VLAN との間でマルチキャスト データを送受信するアップリンク ポートを、MVR 送信元ポートと呼びます。

加入者がチャネルを切り替えた場合、またはテレビのスイッチを切った場合には、セットトップ ボックスからマルチキャスト ストリームに対する IGMP Leave メッセージが送信されます。device CPU は、受信ポートの VLAN 経由で MAC ベースの一般クエリを送信します。VLAN に、このグループに加入している別のセットトップ ボックスがある場合、そのセットトップ ボックスはクエリーに指定された最大応答時間内に応答しなければなりません。応答を受信しなかった場合、CPU はこのグループの転送先としての受信ポートを除外します。

即時脱退機能を使用しない場合、受信ポートのサブスクライバから IGMP Leave メッセージを受信した device は、そのポートに IGMP クエリを送信し、IGMP グループ メンバーシップ レポートを待ちます。設定された時間内にレポートを受信しなかった場合は、受信ポートがマルチキャスト グループ メンバーシップから削除されます。即時脱退機能がイネーブルの場合、IGMP Leave を受信したレシーバ ポートから IGMP クエリーが送信されません。Leave メッセージの受信後ただちに、受信ポートがマルチキャスト グループ メンバーシップから削除されるので、脱退遅延時間が短縮されます。即時脱退機能をイネーブルにするのは、接続されているレシーバ デバイスが 1 つだけのレシーバ ポートに限定してください。

MVR を使用すると、各 VLAN の加入者に対してテレビ チャネルのマルチキャスト トラフィックを重複して送信する必要がなくなります。すべてのチャネル用のマルチキャスト トラフィックは、マルチキャスト VLAN 上でのみ、VLAN トランクに 1 回だけ送信されます。IGMP Leave および Join メッセージは、加入者ポートが割り当てられている VLAN で送信されます。これらのメッセージは、レイヤ 3 デバイス上のマルチキャスト VLAN のマルチキャスト トラフィック ストリームに対し、動的に登録します。アクセス レイヤ device（スイッチ A）は、マルチキャスト VLAN から別の VLAN 内の加入者ポートにトラフィックが転送されるよう転送動作を変更し、2 つの VLAN 間で選択的にトラフィックが送信されるようにします。

IGMP レポートは、マルチキャスト データと同じ IP マルチキャスト グループ アドレスに送信されます。スイッチ A の CPU は、レシーバ ポートから送られたすべての IGMP Join および Leave メッセージを取り込み、MVR モードに基づいて、送信元（アップリンク）ポートのマルチキャスト VLAN に転送しなければなりません。

MVR のデフォルト設定

表 *21 : MVR* のデフォルト設定

機能	デフォルト設定
MVR	グローバルおよびインターフェイス単位でディセーブル
マルチキャスト アドレス	未設定
クエリーの応答時間	0.5 秒
マルチキャスト VLAN	VLAN 1

機能	デフォルト設定
モード	compatible
インターフェイスのデフォルト（ポート単位）	受信ポートでも送信元ポートでもない
即時脱退	すべてのポートでディセーブル

IGMP フィルタリングおよびスロットリング

都市部や Multiple-Dwelling Unit（MDU）などの環境では、device ポート上のユーザーが属する一連のマルチキャストグループを制御する必要があります。この機能を使用することにより、IP/TV などのマルチキャスト サービスの配信を、特定タイプの契約またはサービス計画に基づいて制御できます。また、マルチキャストグループの数を、device ポート上でユーザーが所属できる数に制限することもできます。

IGMP フィルタリング機能を使用すると、IP マルチキャストプロファイルを設定し、それらを各 device ポートに関連付けて、ポート単位でマルチキャスト加入をフィルタリングできます。IGMP プロファイルにはマルチキャスト グループを 1 つまたは複数格納して、グループへのアクセスを許可するか拒否するかを指定できます。マルチキャストグループへのアクセスを拒否する IGMP プロファイルが device ポートに適用されると、IP マルチキャストトラフィックのストリームを要求する IGMP Join レポートが廃棄され、ポートはそのグループからの IP マルチキャストトラフィックを受信できなくなります。マルチキャスト グループへのアクセスがフィルタリング アクションで許可されている場合は、ポートからの IGMP レポートが転送されて、通常の処理が行われます。レイヤ 2 インターフェイスが加入できる IGMP グループの最大数も設定できます。

IGMP フィルタリングで制御されるのは、グループ固有のクエリーおよびメンバーシップ レポート（Join および Leave レポートを含む）だけです。一般 IGMP クエリーは制御されません。IGMP フィルタリングは、IP マルチキャストトラフィックの転送を指示する機能とは無関係です。フィルタリング機能は、マルチキャスト トラフィックの転送に CGMP が使用されているか、または MVR が使用されているかに関係なく、同じように動作します。

IGMP フィルタリングが適用されるのは、IP マルチキャスト グループ アドレスを動的に学習する場合だけです。静的な設定には適用されません。

IGMP スロットリング機能を使用すると、レイヤ 2 インターフェイスが加入できる IGMP グループの最大数を設定できます。IGMP グループの最大数が設定され、IGMP スヌーピング転送テーブルに最大数のエントリが登録されていて、インターフェイスで IGMP Join レポートを受信する場合、インターフェイスを設定することにより、IGMP レポートを廃棄するか、あるいは受信した IGMP レポートでランダムに選択されたマルチキャスト エントリを上書きします。

（注） IGMP フィルタリングが実行されているdevicesは、IGMPv3 Join および Leave メッセージをサポートしていません。

IGMP フィルタリングおよび IGMP スロットリングのデフォルト設定

次の表に、device の IGMP フィルタリングおよびスロットリングのデフォルト設定を示します。

表 *22 : IGMP* フィルタリングのデフォルト設定

機能	デフォルト設定
IGMP フィルタ	適用なし
IGMP グループの最大数	最大数の設定なし （注） 転送テーブルに登録されているグループが最大数に達していると、デフォルトの IGMP スロットリング アクションは IGMP レポートを拒否します。
IGMP プロファイル	未定義
IGMP プロファイル アクション	範囲で示されたアドレスを拒否

IGMP スヌーピングおよび MVR の設定方法

デバイス

IGMP スヌーピングがグローバルにイネーブルまたはディセーブルに設定されている場合は、既存のすべての VLAN インターフェイスでもイネーブルまたはディセーブルになります。デフォルトでは IGMP スヌーピングはすべての VLAN でイネーブルになっていますが、VLAN 単位でイネーブルまたはディセーブルにすることができます。

グローバル IGMP スヌーピングは、VLAN IGMP スヌーピングより優先されます。グローバル スヌーピングがディセーブルの場合、VLAN スヌーピングをイネーブルに設定することはできません。グローバル スヌーピングがイネーブルの場合、VLAN スヌーピングをイネーブルまたはディセーブルに設定できます。

device で IGMP スヌーピングをグローバルにイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping**
4. **end**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp snooping** 例： `スイッチ(config)# ip igmp snooping`	既存のすべての VLAN インターフェイスでグローバルに IGMP スヌーピングを有効にします。 （注） すべての VLAN インターフェイス上で IGMP スヌーピングをグローバルにディセーブルにするには、**no ip igmp snooping** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VLAN インターフェイスでの IGMP スヌーピングのイネーブル化またはディセーブル化

VLAN インターフェイス上で IGMP スヌーピングを有効にするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan** *vlan-id*

4. **end**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp snooping vlan** *vlan-id* 例： `スイッチ(config)# ip igmp snooping vlan 7`	VLAN インターフェイス上で IGMP スヌーピングをイネーブルにします。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。 VLAN スヌーピングをイネーブルにするには、IGMP スヌーピングをグローバルにイネーブルに設定しておく必要があります。 （注） 特定の VLAN インターフェイス上で IGMP スヌーピングをディセーブルにするには、**no ip igmp snooping vlan** *vlan-id* グローバル コンフィギュレーション コマンドを、指定した VLAN 番号に対して使用します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

スヌーピング方法の設定

マルチキャスト対応のルータ ポートは、レイヤ 2 マルチキャスト エントリごとに転送テーブルに追加されます。スイッチは、次のいずれかの方法でポートを学習します。

- IGMP クエリー、Protocol-Independent Multicast（PIM）パケット、およびディスタンスベクトル マルチキャスト ルーティング プロトコル（DVMRP）パケットのスヌーピング
- 他のルータからの Cisco Group Management Protocol（CGMP）パケットのリスニング
- **ip igmp snooping mrouter** グローバル コンフィギュレーション コマンドによるマルチキャストルータポートへの静的な接続

IGMP クエリーおよび PIM パケットと DVMRP パケットのスヌーピング、または CGMP self-join パケットまたは proxy-join パケットのいずれかの待ち受けを行うように、スイッチを設定できます。デフォルトでは、スイッチはすべての VLAN 上の PIM パケットと DVMRP パケットをスヌーピングします。CGMP パケットだけでマルチキャストルータポートを学習するには、**ip igmp snooping vlan vlan-id mrouter learn cgmp** グローバル コンフィギュレーション コマンドを使用します。このコマンドを入力すると、ルータは CGMP self-join パケットおよび CGMP proxy-join パケットだけを待ち受け、その他の CGMP パケットは待ち受けません。PIM-DVMRP パケットだけでマルチキャストルータポートを学習するには、 **ip igmp snooping vlan vlan-id mrouter learn pim-dvmrp** グローバル コンフィギュレーション コマンドを使用します。

学習方法として CGMP を使用する場合で、なおかつ VLAN に CGMP プロキシ対応のマルチキャストルータがない場合は、 **ip cgmp router-only** コマンドを入力し、ルータに動的にアクセスする必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan** *vlan-id* **mrouter learn** {**cgmp** | **pim-dvmrp** }
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**ip igmp snooping vlan** *vlan-id* **mrouter learn** {**cgmp** | **pim-dvmrp** } 例： スイッチ(config)# **ip igmp snooping vlan 1 mrouter learn cgmp**	マルチキャスト ルータの学習方式を指定します。 • **cgmp**：CGMPパケットを待ち受けます。この方法は、制御トラフィックを減らす場合に有用です。 • **pim-dvmrp**：IGMP クエリーおよびPIM/DVMRPパケットをスヌーピングします。これはデフォルトです。 （注） デフォルトの学習方式に戻すには、**no ip igmp snooping vlan** *vlan-id* **mrouter learn cgmp** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show ip igmp snooping** 例： スイッチ# **show ip igmp snooping**	設定を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

マルチキャスト ルータ ポートの設定

deviceにマルチキャスト ルータ ポートを追加する（マルチキャスト ルータへのスタティック接続を有効にする）には、次の手順を実行します。

（注） マルチキャストルータへのスタティック接続は、device ポートに限りサポートされます。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id*
4. **end**
5. **show ip igmp snooping mrouter** [**vlan** *vlan-id*]
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id* 例： `スイッチ(config)# ip igmp snooping vlan 5 mrouter interface gigabitethernet1/0/1`	マルチキャスト ルータの VLAN ID およびマルチキャスト ルータに対するインターフェイスを指定します。 • 指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。 • このインターフェイスには物理インターフェイスまたはポート チャネルを指定できます。ポート チャネル範囲は 1 ～ 128 です。 （注） VLAN からマルチキャスト ルータ ポートを削除するには、**no ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id* グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 5	**show ip igmp snooping mrouter** [**vlan** *vlan-id*] 例： スイッチ# **show ip igmp snooping mrouter vlan 5**	VLAN インターフェイス上で IGMP スヌーピングが有効になっていることを確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

グループに加入するホストの静的な設定

ホストまたはレイヤ 2 ポートは通常、マルチキャスト グループに動的に加入しますが、インターフェイス上にホストを静的に設定することもできます。

マルチキャスト グループのメンバーとしてレイヤ2ポートを追加するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id*
4. **end**
5. **show ip igmp snooping groups**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id* 例： `スイッチ(config)# ip igmp snooping vlan 105 static 230.0.0.1 interface gigabitethernet1/0/1`	マルチキャスト グループのメンバとしてレイヤ 2 ポートを静的に設定します。 • *vlan-id* は、マルチキャスト グループの VLAN ID です。指定できる範囲は 1 ～ 1001 または 1006 ～ 4094 です。 • *ip-address* は、グループの IP アドレスです。 • *interface-id* は、メンバ ポートです。物理インターフェイスまたはポート チャネル（1～128）に設定できます。 （注） マルチキャストグループからレイヤ2ポートを削除するには、 **no ip igmp snooping vlan** *vlan-id* **static** *mac-address* **interface** *interface-id* グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show ip igmp snooping groups** 例： `スイッチ# show ip igmp snooping groups`	メンバ ポートおよび IP アドレスを確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP 即時脱退のイネーブル化

IGMP 即時脱退をイネーブルに設定すると、deviceはポート上で IGMP バージョン 2 の Leave メッセージを検出した場合、ただちにそのポートを削除します。 即時脱退機能は、VLAN の各ポートにレシーバが 1 つ存在する場合にだけ使用してください。

（注） 即時脱退機能をサポートするのは、IGMP バージョン 2 が稼働しているホストだけです。IGMP バージョン 2 は、deviceのデフォルトバージョンです。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan** *vlan-id* **immediate-leave**
4. **end**
5. **show ip igmp snooping vlan** *vlan-id*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp snooping vlan** *vlan-id* **immediate-leave** 例： `スイッチ(config)# ip igmp snooping vlan 21 immediate-leave`	VLAN インターフェイス上で、IGMP 即時脱退をイネーブルにします。 （注） VLAN 上で IGMP 即時脱退をディセーブルにするには、 **no ip igmp snooping vlan** *vlan-id* **immediate-leave** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show ip igmp snooping vlan** *vlan-id* 例：	VLAN インターフェイス上で即時脱退がイネーブルになっていることを確認します。

	コマンドまたはアクション	目的
	スイッチ# **show ip igmp snooping vlan 21**	
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

IGMP 脱退タイマーの設定

脱退時間はグローバルまたはVLAN単位で設定できます。IGMP脱退タイマーの設定をイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping last-member-query-interval** *time*
4. **ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time*
5. **end**
6. **show ip igmp snooping**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp snooping last-member-query-interval** *time* 例： スイッチ(config)# **ip igmp snooping**	IGMP 脱退タイマーをグローバルに設定します。指定できる範囲は 100 ～ 32767 ミリ秒です。 デフォルトの脱退時間は 1000 ミリ秒です。

	コマンドまたはアクション	目的
	last-member-query-interval 1000	（注） IGMP 脱退タイマーをグローバルにリセットしてデフォルト設定に戻すには、 **no ip igmp snooping last-member-query-interval** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time* 例： スイッチ(config)# **ip igmp snooping vlan 210 last-member-query-interval 1000**	（任意）VLAN インターフェイス上で IGMP 脱退時間を設定します。有効値は 100 ～ 32767 ミリ秒です。 （注） VLAN 上に脱退時間を設定すると、グローバルに設定された内容は上書きされます。 （注） 特定の VLAN から IGMP 脱退タイマーの設定を削除するには、**no ip igmp snooping vlan** *vlan-id* **last-member-query-interval** グローバル コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ip igmp snooping** 例： スイッチ# **show ip igmp snooping**	（任意）設定された IGMP 脱退時間を表示します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

TCN 関連コマンドの設定

TCN イベント後のマルチキャスト フラッディング時間の制御

トポロジ変更通知（TCN）イベント後にフラッディングするマルチキャストデータのトラフィックに対し、一般クエリー数を設定できます。TCN フラッド クエリ カウントを 1 に設定した場合は、1 つの一般クエリーを受信した後にフラッディングが停止します。カウントを 7 に設定

した場合、一般クエリーを7つ受信するまでフラッディングが続きます。グループは、TCNイベント中に受信した一般的クエリーに基づいて学習されます。

クライアントロケーションが変更され、ブロックされていた後に現在は転送中の受信者が同じポートに存在する場合や、ポートが脱退メッセージを送信せずにダウンした場合などに TCN イベントが発生します。

TCN フラッド クエリー カウントを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping tcn flood query count** *count*
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp snooping tcn flood query count** *count* 例： `スイッチ(config)# ip igmp snooping tcn flood query count 3`	マルチキャスト トラフィックがフラッディングする IGMP の一般クエリー数を指定します。 指定できる範囲は1～10です。デフォルトのフラッディング クエリー カウントは 2 です。 （注） デフォルトのフラッディング クエリ カウントに戻すには、**no ip igmp snooping tcn flood query count** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例：	特権 EXEC モードに戻ります。

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

フラッディング モードからの回復

トポロジの変更が発生した場合、スパニングツリーのルートは特別な IGMP Leave メッセージ（グローバル Leave メッセージ）をグループ マルチキャスト アドレス 0.0.0.0. に送信します。ただし、スパニングツリーのルートであるかどうかにかかわらず、グローバルな Leave メッセージを送信するようにdeviceを設定できます。ルータはこの特別な Leave メッセージを受信した場合、即座に一般クエリーを送信して、TCN 中のフラッディング モードからできるだけ早く回復するようにします。deviceがスパニングツリーのルートであれば、このコンフィギュレーションに関係なく、Leave メッセージが常に送信されます。

Leave メッセージを送信できるようにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping tcn query solicit**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp snooping tcn query solicit** 例： `スイッチ(config)# ip igmp snooping tcn query solicit`	TCN イベント中に発生したフラッド モードから回復するプロセスの速度を上げるために、IGMP Leave メッセージ（グローバル脱退）を送信します。デフォルトでは、クエリー送信要求はディセーブルに設定されています。 （注） デフォルトのクエリソリューションに戻すには、 **no ip igmp snooping tcn query solicit** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show ip igmp snooping** 例： `スイッチ# show ip igmp snooping`	TCN の設定を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

TCN イベント中のマルチキャスト フラッディングのディセーブル化

deviceはTCNを受信すると、一般クエリを2つ受信するまで、すべてのポートに対してマルチキャストトラフィックをフラッディングします。異なるマルチキャストグループのホストに接続されているポートが複数ある場合、リンク範囲を超えてdeviceによるフラッディングが行われ、パケット損失が発生する可能性があります。TCN フラッディングを制御するには、次の手順を実行します。

手順の概要

1. **enable**

2. **configure terminal**
3. **interface** *interface-id*
4. **no ip igmp snooping tcn flood**
5. **end**
6. **show ip igmp snooping**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**no ip igmp snooping tcn flood** 例： `スイッチ(config-if)# no ip igmp snooping tcn flood`	スパニングツリーの TCN イベント中に発生するマルチキャスト トラフィックのフラッディングをディセーブルにします。 デフォルトでは、インターフェイス上のマルチキャスト フラッディングはイネーブルです。 （注） インターフェイス上でマルチキャスト フラッディングを再度イネーブルにするには、 **ip igmp snooping tcn flood** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 6	**show ip igmp snooping** 例： `スイッチ# show ip igmp snooping`	TCN の設定を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP スヌーピング クエリアの設定

特定の VLAN で IGMP スヌーピング クエリア機能をイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp snooping querier**
4. **ip igmp snooping querier address** *ip_address*
5. **ip igmp snooping querier query-interval** *interval-count*
6. **ip igmp snooping querier tcn query** [**count** *count* | **interval** *interval*]
7. **ip igmp snooping querier timer expiry** *timeout*
8. **ip igmp snooping querier version** *version*
9. **end**
10. **show ip igmp snooping vlan** *vlan-id*
11. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**ip igmp snooping querier** 例： スイッチ(config)# **ip igmp snooping querier**	IGMP スヌーピング クエリアをイネーブルにします。
ステップ 4	**ip igmp snooping querier address** *ip_address* 例： スイッチ(config)# **ip igmp snooping querier address 172.16.24.1**	（任意）IGMP スヌーピング クエリアの IP アドレスを指定します。IP アドレスを指定しない場合、クエリアはIGMPクエリアに設定されたグローバル IP アドレスを使用します。 （注） IGMPスヌーピングクエリがdevice上でIP アドレスを検出できない場合、IGMP 一般クエリを生成しません。
ステップ 5	**ip igmp snooping querier query-interval** *interval-count* 例： スイッチ(config)# **ip igmp snooping querier query-interval 30**	（任意）IGMPクエリアの間隔を設定します。指定できる範囲は 1 ～ 18000 秒です。
ステップ 6	**ip igmp snooping querier tcn query** [**count** *count* | **interval** *interval*] 例： スイッチ(config)# **ip igmp snooping querier tcn query interval 20**	（任意）トポロジ変更通知（TCN）クエリーの間隔を設定します。指定できる count の範囲は 1 ～ 10 です。指定できる interval の範囲は 1 ～ 255 秒です。
ステップ 7	**ip igmp snooping querier timer expiry** *timeout* 例： スイッチ(config)# **ip igmp snooping querier timer expiry 180**	（任意）IGMPクエリアが期限切れになる時間を設定します。指定できる範囲は 60 ～ 300 秒です。
ステップ 8	**ip igmp snooping querier version** *version* 例： スイッチ(config)# **ip igmp snooping querier version 2**	（任意）クエリア機能が使用するIGMPバージョン番号を選択します。選択できる番号は1または2です。

	コマンドまたはアクション	目的
ステップ 9	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 10	**show ip igmp snooping vlan** *vlan-id* 例： スイッチ# **show ip igmp snooping vlan 30**	（任意）VLANインターフェイス上でIGMPスヌーピング クエリアがイネーブルになっていることを確認します。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP レポート抑制のディセーブル化

IGMP レポート抑制をディセーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no ip igmp snooping report-suppression**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**no ip igmp snooping report-suppression** 例： スイッチ(config)# **no ip igmp snooping report-suppression**	IGMP レポート抑制をディセーブルにします。IGMP レポート抑制がディセーブルの場合、すべての IGMP レポートがマルチキャスト ルータに転送されます。 IGMP レポート抑制はデフォルトでイネーブルです。 IGMP レポート抑制がイネーブルの場合、deviceはマルチキャスト ルータ クエリごとに IGMP レポートを 1 つだけ転送します。 （注） IGMP レポート抑制を再びイネーブルにするには、 **ip igmp snooping report-suppression** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show ip igmp snooping** 例： スイッチ# **show ip igmp snooping**	IGMP レポート抑制がディセーブルになっていることを確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MVR グローバル パラメータの設定

デフォルト値を使用する場合は、オプションの MVR パラメータを設定する必要はありません。デフォルトのパラメータを変更する場合には（MVR VLAN以外）、最初にMVRをイネーブルにする必要があります。

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

手順の概要

1. **enable**
2. **configure terminal**
3. **mvr**
4. **mvr group** *ip-address* [*count*]
5. **mvr querytime** *value*
6. **mvr vlan** *vlan-id*
7. **mvr mode** {**dynamic** | **compatible**}
8. **end**
9. 次のいずれかを使用します。
   - **show mvr**
   - **show mvr members**
10. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mvr** 例： スイッチ (config)# **mvr**	device上で MVR をイネーブルにします。
ステップ 4	**mvr group** *ip-address* [*count*] 例： スイッチ(config)# **mvr group 228.1.23.4**	device 上で IP マルチキャストアドレスを設定するか、または *count* パラメータを使用して（*count* の範囲は 1 ～ 256 で、デフォルトは 1）連続する MVR グループアドレスを設定します。このアドレスに送信されるすべてのマルチキャストデータは、device 上の送信元ポートおよびこのマルチキャスト アドレス上のデータを受信するよう選択されたすべての受信ポートに送信されます。マルチキャスト アドレスとテレビ チャネルは 1 対 1 の対応です。

	コマンドまたはアクション	目的
		（注） スイッチをデフォルト設定に戻すには、**no mvr** [**mode** | **group** *ip-address* | **querytime** | **vlan**] グローバル コンフィギュレーション コマンドを使用します。
ステップ 5	**mvr querytime** *value* 例： `スイッチ(config)# mvr querytime 10`	（任意）マルチキャスト グループ メンバーシップからポートを削除する前に、受信ポート上でIGMP レポート メンバーシップを待機する最大時間を定義します。この値は10分の1秒単位で設定します。範囲は 1 ～ 100、デフォルトは 10 分の 5 秒、つまり 0.5 秒です。
ステップ 6	**mvr vlan** *vlan-id* 例： `スイッチ(config)# mvr vlan 22`	（任意）マルチキャスト データを受信する VLAN を指定します。すべての送信元ポートはこのVLAN に属する必要があります。VLANの範囲は1～1001 および 1006 ～ 4094 です。デフォルトは VLAN 1 です。
ステップ 7	**mvr mode** {**dynamic** | **compatible**} 例： `スイッチ(config)# mvr mode dynamic`	（任意）次の MVR の動作モードを指定します。 • **dynamic**：送信元ポートでダイナミック MVR メンバーシップを使用できます。 • **compatible**：Catalyst 3500 XL および Catalyst 2900 XL devicesとの互換性が得られます。送信元ポートでのダイナミック IGMP Join はサポートされません。 デフォルトは **compatible** モードです。 （注） スイッチをデフォルト設定に戻すには、**no mvr** [**mode** | **group** *ip-address* | **querytime** | **vlan**] グローバル コンフィギュレーション コマンドを使用します。
ステップ 8	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 9	次のいずれかを使用します。 • **show mvr** • **show mvr members** 例：	設定を確認します。

	コマンドまたはアクション	目的
	スイッチ# **show mvr** OR スイッチ# **show mvr members**	
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MVR インターフェイスの設定

レイヤ 2 MVR インターフェイスを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **mvr**
4. **interface** *interface-id*
5. **mvr type** {**source** | **receiver**}
6. **mvr vlan** *vlan-id* **group** [*ip-address*]
7. **mvr immediate**
8. **end**
9. 次のいずれかを使用します。
   - **show mvr**
   - **show mvr interface**
   - **show mvr members**
10. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**mvr** 例： スイッチ (config)# **mvr**	device上で MVR をイネーブルにします。
ステップ 4	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/2**	設定するレイヤ2ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**mvr type** {**source** | **receiver**} 例： スイッチ(config-if)# **mvr type receiver**	MVR ポートを次のいずれかに設定します。 • **source**：マルチキャスト データを送受信するアップリンク ポートを送信元ポートとして設定します。加入者が送信元ポートに直接接続することはできません。device上の送信元ポートはいずれも、1 つのマルチキャスト VLAN に属する必要があります。 • **receiver**：加入者ポートであり、マルチキャストデータを受信するだけの場合、レシーバポートとしてポートを設定します。受信ポートは、スタティックな設定、または IGMP Leave および Join メッセージによってマルチキャスト グループのメンバーになるまでは、データを受信しません。受信ポートをマルチキャストVLANに所属させることはできません。 デフォルトでは、非MVRポートとして設定されます。非 MVR ポートに MVR 特性を設定しようとしても、エラーになります。 （注） インターフェイスをデフォルト設定に戻すには、 **no mvr** [**type** | **immediate** | **vlan** *vlan-id* | **group**] インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 6	**mvr vlan** *vlan-id* **group** [*ip-address*] 例：	（任意）マルチキャスト VLAN および IP マルチキャスト アドレスに送信されたマルチキャスト トラフィックを受信するポートを静的に設定します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **mvr vlan 22 group 228.1.23.4**	グループ メンバとして静的に設定されたポートは、静的に削除されない限り、グループ メンバのままです。 （注） 互換モードでは、このコマンドが適用されるのはレシーバ ポートだけです。ダイナミック モードでは、レシーバ ポートおよび送信元ポートに適用されます。 レシーバ ポートは、IGMP Join および Leave メッセージを使用することによって、マルチキャスト グループに動的に加入することもできます。
ステップ 7	**mvr immediate** 例： スイッチ(config-if)# **mvr immediate**	（任意）ポート上でMVRの即時脱退機能をイネーブルにします。 （注） このコマンドが適用されるのは、受信ポートだけです。また、イネーブルにするのは、単一の受信デバイスが接続されている受信ポートに限定してください。
ステップ 8	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 9	次のいずれかを使用します。 • **show mvr** • **show mvr interface** • **show mvr members** 例： スイッチ# **show mvr interface** Port Type Status Immediate Leave ---- ---- ------- --------------- Gi1/0/2 RECEIVER ACTIVE/DOWN ENABLED	設定を確認します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP プロファイルの設定

IGMP プロファイルを作成するには、次の手順を実行します。

このタスクはオプションです。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp profile** *profile number*
4. **permit** | **deny**
5. **range** *ip multicast address*
6. **end**
7. **show ip igmp profile** *profile number*
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp profile** *profile number* 例： `スイッチ(config)# ip igmp profile 3`	設定するプロファイルに番号を割り当て、IGMP プロファイル コンフィギュレーション モードを開始します。指定できるプロファイル番号の範囲は 1 ～ 4294967295 です。IGMP プロファイル コンフィギュレーション モードでは、次のコマンドを使用することでプロファイルを作成できます。 • **deny**：一致するアドレスを拒否します。デフォルトで設定されています。 • **exit**：IGMP プロファイル コンフィギュレーション モードを終了します。 • **no**：コマンドを否定するか、または設定をデフォルトに戻します。

	コマンドまたはアクション	目的
		• **permit**：一致するアドレスを許可するように指定します。 • **range**：プロファイルの IP アドレスの範囲を指定します。単一の IP アドレス、または開始アドレスと終了アドレスで指定された IP アドレス範囲を入力できます。 デフォルトでは、device には IGMP プロファイルが設定されていません。 （注） プロファイルを削除するには、**no ip igmp profile** *profile number* グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**permit** | **deny** 例： `スイッチ(config-igmp-profile)# permit`	（任意）IP マルチキャスト アドレスへのアクセスを許可または拒否するアクションを設定します。アクションを設定しないと、プロファイルのデフォルト設定はアクセス拒否になります。
ステップ 5	**range** *ip multicast address* 例： `スイッチ(config-igmp-profile)# range 229.9.9.0`	アクセスを制御する IP マルチキャスト アドレスまたは IP マルチキャスト アドレスの範囲を入力します。範囲を入力する場合は、IP マルチキャスト アドレスの下限値、スペースを 1 つ、IP マルチキャスト アドレスの上限値を入力します。 **range** コマンドを複数回入力し、複数のアドレスまたはアドレス範囲を入力できます。 （注） IP マルチキャスト アドレスまたは IP マルチキャスト アドレス範囲を削除するには、**no range** *ip multicast address* IGMP プロファイル コンフィギュレーション コマンドを使用します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show ip igmp profile** *profile number* 例： `スイッチ# show ip igmp profile 3`	プロファイルの設定を確認します。

	コマンドまたはアクション	目的
ステップ 8	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP プロファイルの適用

IGMP プロファイルで定義されているとおりにアクセスを制御するには、プロファイルを該当するインターフェイスに適用する必要があります。IGMP プロファイルを適用できるのは、レイヤ 2 アクセス ポートだけです。ルーテッド ポートや SVI には適用できません。EtherChannel ポート グループに所属するポートに、プロファイルを適用することはできません。1 つのプロファイルを複数のインターフェイスに適用できますが、1 つのインターフェイスに適用できるプロファイルは 1 つだけです。

スイッチ ポートに IGMP プロファイルを適用するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp filter** *profile number*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	物理インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは、EtherChannel ポート グループに所属していないレイヤ 2 ポートでなければなりません。
ステップ 4	**ip igmp filter** *profile number* 例： スイッチ(config-if)# **ip igmp filter 321**	インターフェイスに指定された IGMP プロファイルを適用します。指定できる範囲は 1 ～ 4294967295 です。 （注） インターフェイスからプロファイルを削除するには、 **no ip igmp filter***profile number* インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP グループの最大数の設定

レイヤ 2 インターフェイスが加入できる IGMP グループの最大数を設定するには、次の手順を実行します。

始める前に

この制限が適用されるのはレイヤ 2 ポートだけです。ルーテッド ポートや SVI には IGMP グループの最大数を設定できません。このコマンドは、論理 EtherChannel インターフェイスでも使用できますが、EtherChannel ポート グループに属するポートでは使用できません。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp max-groups** *number*
5. **end**
6. **show running-config interface** *interface-id*
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/2`	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは、EtherChannel ポート グループに所属しないレイヤ 2 ポート、または EtherChannel インターフェイスのいずれかにできます。
ステップ 4	**ip igmp max-groups** *number* 例： `スイッチ(config-if)# ip igmp max-groups 20`	インターフェイスが加入できる IGMP グループの最大数を設定します。指定できる範囲は 0 ～ 4294967294 です。デフォルトでは最大数は設定されません。 （注） グループの最大数に関する制限を削除し、デフォルト設定（制限なし）に戻すには、**no ip igmp max-groups** インターフェイス コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config interface** *interface-id* 例： スイッチ# **interface gigabitethernet1/0/1**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP スロットリング アクションの設定

レイヤ2インターフェイスが加入できるIGMPグループの最大数を設定した後、受信したIGMPレポートの新しいグループで、既存のグループを上書きするようにインターフェイスを設定できます。

転送テーブルに最大数のエントリが登録されているときにスロットリングアクションを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp max-groups action** {**deny** | **replace**}
5. **end**
6. **show running-config interface** *interface-id*
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	設定する物理インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは、EtherChannel ポート グループに所属しないレイヤ 2 ポート、または EtherChannel インターフェイスのいずれかにできます。トランク ポートをインターフェイスにすることはできません。
ステップ 4	**ip igmp max-groups action** {**deny** | **replace**} 例： スイッチ(config-if)# **ip igmp max-groups action replace**	インターフェイスが IGMP レポートを受信したときに、転送テーブルに最大数のエントリが登録されている場合は、次のいずれかのアクションをインターフェイスに指定します。 • **deny**：レポートを破棄します。このスロットリング アクションを設定すると、すでに転送テーブルに登録されていたエントリは、削除されることはありませんが期限切れになります。エントリが期限切れになり、最大数のエントリが転送テーブルに登録されていると、deviceは、インターフェイスで受信した次の IGMP レポートを廃棄します。 • **replace**：既存のグループを、IGMP レポートを受信した新しいグループで上書きします。このスロットリング アクションを設定すると、すでに転送テーブルに登録されていたエントリは削除されます。転送テーブルのエントリが最大数まで達したら、deviceはランダムに選択したエントリを受信した IGMP レポートで上書きします。 deviceが転送テーブルのエントリを削除しないようにするには、インターフェイスにより転送テーブルにエントリが追加される前に、IGMP スロットリング アクションを設定します。

	コマンドまたはアクション	目的
		（注） レポートの廃棄というデフォルトのアクションに戻すには、**no ip igmp max-groups action** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config interface** *interface-id* 例： スイッチ# **show running-config interface gigabitethernet1/0/1**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IGMP スヌーピングおよび MVR のモニターリング

IGMP スヌーピング情報の監視

ダイナミックに学習された、あるいはスタティックに設定されたルータ ポートおよび VLAN インターフェイスの IGMP スヌーピング情報を表示できます。また、IGMP スヌーピング用に設定された VLAN の IP アドレス マルチキャスト エントリを表示することもできます。

表 *23 : IGMP* スヌーピング情報を表示するためのコマンド

コマンド	目的
show ip igmp snooping [**vlan** *vlan-id* [detail]]	device上のすべての VLAN または特定の VLAN のスヌーピング設定情報を表示します。 （任意）個々のVLANに関する情報を表示するには、**vlan** *vlan-id* を入力します。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。

コマンド	目的
show ip igmp snooping groups [**count** |**dynamic** [**count**] | **user** [**count**]]	deviceまたは特定のパラメータに関して、マルチキャスト テーブル情報を表示します。 • **count**：実エントリの代わりに、指定のコマンド オプションのエントリ総数を表示します。 • **dynamic**：IGMP スヌーピングによって学習されたエントリを表示します。 • **user**：ユーザーによって設定されたマルチキャスト エントリだけを表示します。
show ip igmp snooping groups vlan *vlan-id* [*ip_address* | **count** | **dynamic** [**count**] | **user**[**count**]]	マルチキャスト VLAN またはその VLAN の特定のパラメータについて、マルチキャストテーブル情報を表示します。 • *vlan-id*：VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。 • **count**：実エントリの代わりに、指定のコマンド オプションのエントリ総数を表示します。 • **dynamic**：IGMP スヌーピングによって学習されたエントリを表示します。 • *ip_address*：指定したグループ IP アドレスのマルチキャスト グループの特性を表示します。 • **user**：ユーザーによって設定されたマルチキャスト エントリだけを表示します。
show ip igmp snooping mrouter [**vlan** *vlan-id*]	ダイナミックに学習され、手動で設定されたマルチキャスト ルータ インターフェイスの情報を表示します。 （注） IGMP スヌーピングを有効にすると、device はマルチキャスト ルータの接続先インターフェイスを自動的に学習します。これらのインターフェイスは動的に学習されます。 （オプション）**vlan** *vlan-id* を入力すると、特定の VLAN に関する情報が表示されます。
show ip igmp snooping querier [**vlan** *vlan-id*] **detail**	IP アドレスおよび VLAN で受信した最新の IGMP クエリー メッセージの受信ポートに関する情報、VLAN の IGMP スヌーピング クエリアの設定および動作ステートに関する情報を表示します。

MVR のモニターリング

スイッチまたは指定されたインターフェイスの MVR をモニターするには、次の MVR 情報を表示します。

表 *24 : MVR* 情報を表示するためのコマンド

コマンド	目的
show mvr	スイッチの MVR ステータスおよび値を表示します。これは、MVR のイネーブルまたはディセーブルの判別、マルチキャスト VLAN、マルチキャスト グループの最大数（256）および現在の数（0 ～ 256）、クエリーの応答時間、および MVR モードです。
show mvr interface [*interface-id*] [**members** [**vlan** *vlan-id*]]	すべての MVR インターフェイスおよびその MVR 設定を表示します。 特定のインターフェイスを指定すると、次の情報が表示されます。 • Type：Receiver または Source • Status：次のいずれか • ACTIVE は、ポートが VLAN に含まれていることを意味します。 • UP/DOWN は、ポートが転送中または転送中ではないことを示します。 • INACTIVE は、ポートが VLAN に含まれていないことを意味します。 • Immediate Leave：Enabled または Disabled **members** キーワードを入力すると、そのポート上のすべてのマルチキャスト グループ メンバが表示されます。VLAN ID を入力した場合は、その VLAN 上のすべてのマルチキャスト グループ メンバが表示されます。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。
show mvr members [*ip-address*]	すべての IP マルチキャスト グループまたは指定した IP マルチキャスト グループ IP アドレスに含まれているレシーバ ポートおよび送信元ポートがすべて表示されます。

IGMP のフィルタリングおよびスロットリング設定のモニターリング

IGMP プロファイルの特性を表示したり、device 上のすべてのインターフェイスまたは指定されたインターフェイスの IGMP プロファイルや最大グループ設定を表示したりできます。また、device 上のすべてのインターフェイスまたは指定したインターフェイスに関する IGMP スロットリング設定を表示することもできます。

表 *25 : IGMP* のフィルタリングおよびスロットリング設定を表示するためのコマンド

コマンド	目的
show ip igmp profile [*profile number*]	特定の IGMP プロファイルまたはdevice上で定義されているすべての IGMP プロファイルを表示します。
show running-config [**interface** *interface-id*]	インターフェイスが所属できる IGMP グループの最大数（設定されている場合）や、インターフェイスに適用される IGMP プロファイルを含む、特定のインターフェイスまたはdevice上のすべてのインターフェイスの設定を表示します。

IGMP スヌーピングおよび MVR の設定例

例：CGMP パケットを使用した IGMP スヌーピングの設定

次に、CGMP パケットを学習方式として使用するように IGMP スヌーピングを設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ip igmp snooping vlan 1 mrouter learn cgmp
スイッチ(config)# end
```

例：マルチキャスト ルータへの静的な接続のイネーブル化

次に、マルチキャスト ルータへの静的な接続をイネーブルにする例を示します。

```
スイッチ configure terminal
スイッチ ip igmp snooping vlan 200 mrouter interface gigabitethernet1/0/2
スイッチ end
```

例：グループに加入するホストの静的な設定

次に、ポート上のホストを静的に設定する例を示します。

```
スイッチ#  configure terminal
スイッチ#  ip igmp snooping vlan 105 static 0100.1212.0000 interface gigabitethernet1/0/1

スイッチ#  end
```

例：IGMP 即時脱退のイネーブル化

次に、VLAN 130 上で IGMP 即時脱退をイネーブルにする例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ip igmp snooping vlan 130 immediate-leave
スイッチ(config)# end
```

例：IGMP スヌーピング クエリアの送信元アドレスの設定

次に、IGMP スヌーピング クエリアの送信元アドレスを 10.0.0.64 に設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ip igmp snooping querier 10.0.0.64
スイッチ(config)# end
```

例：IGMP スヌーピング クエリアの最大応答時間の設定

次の例では、IGMP スヌーピング クエリアの最大応答時間を 25 秒に設定する方法を示します。

```
スイッチ# configure terminal
スイッチ(config)# ip igmp snooping querier query-interval 25
スイッチ(config)# end
```

例：IGMP スヌーピング クエリア タイムアウトの設定

次の例では、IGMP スヌーピング クエリアのタイムアウトを 60 秒に設定する方法を示します。

```
スイッチ# configure terminal
スイッチ(config)# ip igmp snooping querier timeout expiry 60
スイッチ(config)# end
```

例：IGMP スヌーピング クエリア機能の設定

次に、IGMP スヌーピング クエリア機能をバージョン 2 に設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# no ip igmp snooping querier version 2
スイッチ(config)# end
```

例：IGMP プロファイルの設定

次に、単一の IP マルチキャスト アドレスへのアクセスを許可する IGMP プロファイル 4 を作成して、設定を確認する例を示します。アクションが拒否（デフォルト）である場合は、**show ip igmp profile** の出力には表示されません。

```
スイッチ(config)# ip igmp profile 4
スイッチ(config-igmp-profile)# permit
スイッチ(config-igmp-profile)# range 229.9.9.0
スイッチ(config-igmp-profile)# end
スイッチ# show ip igmp profile 4
IGMP Profile 4
    permit
    range 229.9.9.0 229.9.9.0
```

例：IGMP プロファイルの適用

次に、ポートに IGMP プロファイル 4 を適用する例を示します。

```
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# ip igmp filter 4
スイッチ(config-if)# end
```

例：IGMP グループの最大数の設定

次の例では、ポートが加入できる IGMP グループ数を 25 に制限する方法を示します。

```
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# ip igmp max-groups 25
スイッチ(config-if)# end
```

例：MVR グローバル パラメータの設定

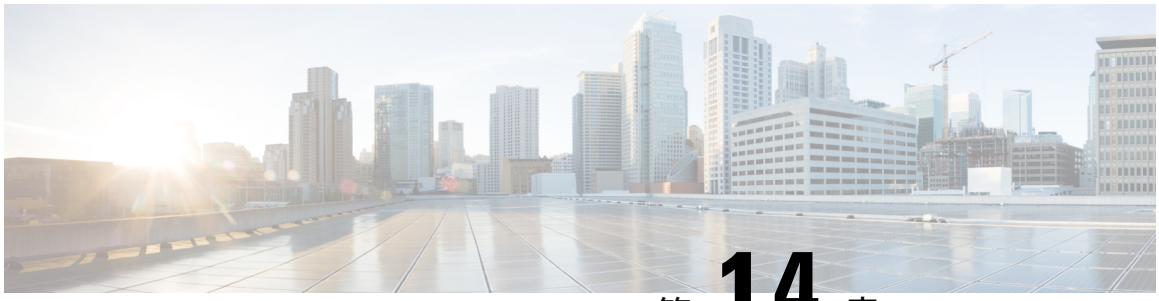
次に、MVR をイネーブルにして、MVR グループ アドレスを設定し、クエリー タイムを 1 秒（10 分の 10 秒）に設定し、MVR マルチキャスト VLAN を VLAN 22 として指定し、MVR モードをダイナミックに設定する例を示します。

```
スイッチ(config)# mvr
スイッチ(config)# mvr group 228.1.23.4
スイッチ(config)# mvr querytime 10
スイッチ(config)# mvr vlan 22
スイッチ(config)# mvr mode dynamic
スイッチ(config)# end
```

例：MVR インターフェイスの設定

次に、ポートをレシーバ ポートとして設定し、マルチキャスト グループ アドレスに送信されたマルチキャスト トラフィックを受信するようにポートを静的に設定し、ポートに即時脱退機能を設定し、結果を確認する例を示します。

```
スイッチ(config)# mvr
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# mvr type receiver
スイッチ(config-if)# mvr vlan 22 group 228.1.23.4
スイッチ(config-if)# mvr immediate
スイッチ(config)# end
スイッチ# show mvr interface

Port Type Status Immediate Leave
---- ---- ------- ---------------
Gi1/0/2 RECEIVER ACTIVE/DOWN ENABLED
```



第 14 章

CGMP の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

CGMP の設定の前提条件

CGMP を設定する際の前提条件は次のとおりです。

- 複数のシスコ CGMP 対応デバイスがスイッチド ネットワークに接続されていて、**ip cgmp proxy** コマンドを使用する必要がある場合は、すべてのデバイスを同じ CGMP オプションを使用して設定し、他社製のルータよりも IGMP クエリアになる優先順位を上げてください。
- CGMP を使用するには、3560-CX スイッチで IP Services フィーチャ セットがイネーブルになっている必要があります。

CGMP の制約事項

次に、CGMP の制約事項を示します。

- CGMP と HSRPv1 は両立できません。CGMP 脱退処理と HSRPv1 を同時にイネーブルにできません。ただし、CGMP と HSRPv2 は同時にイネーブルにできます。

CGMP に関する情報

Cisco Group Management Protocol、またはCGMP サーバーサポートはdeviceで提供されます。クライアント側機能は提供されません。deviceは、IGMP スヌーピングをサポートしない、CGMP クライアント機能が組み込まれているデバイス用の CGMP サーバーとして機能します。

CGMP はレイヤ 2 Catalyst devicesに接続された Cisco ルータおよびマルチレイヤdevicesで使用されるプロトコルであり、IGMP で実行される作業と同様の作業を実行します。CGMP を使用すると、レイヤ 2 グループ メンバーシップ情報を CGMP サーバからdeviceに通信できます。これにより、deviceはすべてのdeviceインターフェイスにマルチキャストトラフィックをフラッディングしないで、マルチキャストメンバーが存在する場所（インターフェイス）を取得できるようになります。（IGMP スヌーピングは、マルチキャストパケットのフラッディングを抑制するためのもう 1 つの方法です）。

CGMP が必要となるのは、レイヤ 2 deviceで IP マルチキャスト データ パケットと IGMP レポートメッセージを区別できないためです。これらはともに MAC レベルで、同じグループアドレスにアドレス指定されます。

CGMP サーバ サポートのイネーブル化

複数のシスコ CGMP 対応デバイスがスイッチドネットワークに接続されていて、**ip cgmp proxy** コマンドを設定する場合は、すべてのデバイスを同じ CGMP オプションを使用して設定し、他社製のルータよりも IGMP クエリアになる優先順位を上げてください。device インターフェイスで CGMP サーバーをイネーブルにするには、次の手順を実行します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip cgmp** [**proxy** | **router-only**]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	レイヤ 2 Catalyst deviceに接続されたインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip cgmp** [**proxy** | **router-only**] 例： スイッチ(config-if)# **ip cgmp proxy**	インターフェイスでCGMPをイネーブルにします。 デフォルトでは、CGMP はすべてのインターフェイス上でディセーブルです。 CGMP をイネーブルにすると、CGMP Join メッセージが送信されます。レイヤ 2 Catalyst devicesに接続されたレイヤ 3 インターフェイスでだけ、CGMP をイネーブルにします。 （任意）**proxy** キーワードを入力すると、CGMP プロキシ機能がイネーブルになります。プロキシルータは、CGMP 非対応ルータの MAC アドレス、およびグループ アドレス 0000.0000.0000 が格納された CGMP Join メッセージを送信し、CGMP 非対応ルータが存在することをアドバタイズします。

	コマンドまたはアクション	目的
		（注） CGMP プロキシを実行するには、deviceをIGMP クエリアに設定する必要があります。**ip cgmp proxy** コマンドを設定する場合は、ネットワークで動作中の IGMP のバージョンに応じて、IPアドレスが最大または最小のdeviceが IGMP クエリアになるように IP アドレスを手動で操作する必要があります。IGMP バージョン 2 クエリアは、インターフェイスの最小の IP アドレスに基づいて選択されます。IGMP バージョン 1 クエリアは、インターフェイスで使用されるマルチキャスト ルーティング プロトコルに基づいて選択されます。 （注） インターフェイス上で CGMP をディセーブルにするには、**no ip cgmp** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

レイヤ 2 Catalyst device CGMP クライアントの設定を確認します。詳細については、製品に付属のマニュアルを参照してください。

CGMP のモニタリング

IP ルーティング テーブル、キャッシュ、データベースの内容など、特定の統計情報を表示できます。

（注） このリリースでは、ルート単位の統計情報がサポートされていません。

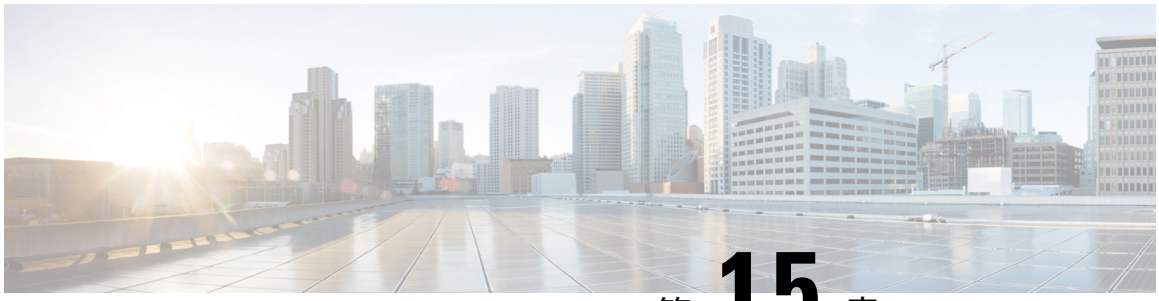
また、リソースの使用状況を学習し、ネットワーク問題を解決するための情報を表示することもできます。さらに、ノードの到達可能性に関する情報を表示し、そのパケットが経由するネットワーク内のパスを検出することもできます。

次の表に示す特権 EXEC コマンドのいずれかを使用すると、さまざまなルーティング統計情報を表示できます。

表 *26 :* システムおよびネットワーク統計情報を表示するコマンド

コマンド	目的
ping [*group-name* | *group-address*]	マルチキャスト グループ アドレスにインターネット制御メッセージプロトコル（ICMP）エコー要求を送信します。
show ip igmp groups [*group-name* | *group-address* | *type number*]	スイッチに直接接続されており、IGMP を介して学習したマルチキャスト グループを表示します。
show ip igmp interface [*type number*]	インターフェイスのマルチキャスト関連情報を表示します。
show ip mcache [*group* [*source*]]	IP 高速スイッチング キャッシュの内容を表示します。
show ip mpacket [*source-address* | *name*] [*group-address* | *name*] [**detail**]	循環キャッシュヘッダー バッファの内容を表示します。
show ip mroute [*group-name* | *group-address*] [*source*] [**summary**] [**count**] [**active** *kbps*]	IP マルチキャスト ルーティング テーブルの内容を表示します。
show ip pim interface [*type number*] [**count**] [**detail**]	PIM に対して設定されたインターフェイスに関する情報を表示します。このコマンドは、すべてのソフトウェア イメージで使用できます。
show ip pim neighbor [*type number*]	スイッチによって検出された PIM ネイバーのリストを示します。このコマンドは、すべてのソフトウェア イメージで使用できます。

コマンド	目的
show ip pim rp [*group-name* | *group-address*]	スパース モードのマルチキャスト グループに関連付けられた RP ルータを表示します。このコマンドは、すべてのソフトウェア イメージで使用できます。
show ip rpf {*source-address* | *name*}	スイッチの RPF の実行方法（ユニキャスト ルーティング テーブル、DVMRP ルーティング テーブル、またはスタティック マルチキャスト ルーティングのいずれか）を表示します。
show ip sap [*group* | *session-name* | **detail**]	Session Announcement Protocol（SAP）バージョン 2 キャッシュを表示します。



第 15 章

PIM（Protocol Independent Multicast）の設定

PIM の前提条件

- PIM設定プロセスを開始する前に、使用するPIMモードを決定します。この決定は、ネットワーク上でサポートするアプリケーションに基づきます。次の注意事項に従ってください。
  - 一般に、本質的に1対多または多対多アプリケーションではPIM-SMを正常に使用できます。
  - 1対多アプリケーションで最適なパフォーマンスを得るには、SSMが適しています。ただし、IGMPバージョン3サポートが必要です。
- PIM スタブ ルーティングを設定する前に、次の条件を満たしていることを確認します。
  - スタブ ルータと中央のルータの両方にIP マルチキャスト ルーティングが設定されている必要があります。さらに、スタブルータのアップリンク インターフェイスにPIM モード（デンス モード、スパース モード、またはスパース - デンス モード）が設定されている必要があります。
  - また、deviceに Enhanced Interior Gateway Routing Protocol（EIGRP）スタブ ルーティングが設定されている必要があります。
  - PIM スタブ ルータは、ディストリビューション ルータ間の伝送トラフィックのルーティングは行いません。ユニキャスト（EIGRP）スタブルーティングではこの動作が

強制されます。PIM スタブ ルータの動作を支援するためにユニキャスト スタブ ルーティングを設定する必要があります。

PIM に関する制約事項

PIMv1 および PIMv2 の相互運用性

device 上でのマルチキャストルーティングの設定ミスを回避するには、ここに記載する情報を確認してください。

シスコの PIMv2 実装を使用すると、バージョン 1 とバージョン 2 間での相互運用性および変換が可能となります。ただし、若干の問題が発生する場合もあります。

PIMv2 に差分的にアップグレードできます。PIM バージョン 1 および 2 を、1 つのネットワーク内の異なるルータおよびマルチレイヤ devices に設定できます。内部的には、共有メディアネットワーク上のすべてのルータおよびマルチレイヤ devicesで同じ PIM バージョンを実行する必要があります。したがって、PIMv2 デバイスが PIMv1 デバイスを検出した場合は、バージョン 1 デバイスがシャットダウンするかアップグレードされるまで、バージョン 2 デバイスはバージョン 1 にダウングレードされます。

PIMv2 は BSR を使用して各グループプレフィックスの RP 設定情報を検出し、PIM ドメイン内のすべてのルータおよびマルチレイヤ devices にアナウンスします。自動 RP 機能を組み合わせることにより、PIMv2 BSR と同じ作業を PIMv1 で実行できます。ただし、自動 RP は PIMv1 から独立している、スタンドアロンのシスコ独自のプロトコルで、PIMv2 は IETF 標準の追跡プロトコルです。

（注） したがって、PIMv2 の使用を推奨します。BSR 機能は、Cisco ルータおよびマルチレイヤ devices 上の Auto-RP と相互運用します。

PIMv2 デバイスを PIMv1 デバイスと相互運用させる場合は、自動 RP を事前に導入しておく必要があります。自動 RP マッピング エージェントでもある PIMv2 BSR は、自動 RP で選択された RP を自動的にアドバタイズします。つまり、自動 RP によって、グループ内のルータまたはマルチレイヤ deviceごとに 1 つの RP が設定されます。ドメイン内のルータおよび devices の中には、複数の RP を選択するために PIMv2 ハッシュ機能を使用しないものもあります。

PIMv1 の自動 RP 機能は PIMv2 RP 機能と相互運用するため、PIMv1 と PIMv2 が混在する領域内に SM グループを設定できます。すべての PIMv2 デバイスで PIMv1 を使用できますが、RP を PIMv2 にアップグレードすることを推奨します。PIMv2 への移行を簡単に行うには、以下を推奨します。

- 領域全体で Auto-RP を使用します。
- 領域全体でスパース - デンス モードを設定します。

自動 RP がまだ PIMv1 領域に設定されていない場合は、自動 RP を設定してください。

PIM スタブ ルーティングの設定に関する制約事項

- IP サービス イメージには、完全なマルチキャスト ルーティングが含まれます。
- 直接接続されたマルチキャスト（IGMP）レシーバおよび送信元だけが、レイヤ 2 アクセス ドメインで許可されます。アクセス ドメインでは、PIM プロトコルはサポートされません。
- PIM スタブルーティングを使用するネットワークでは、ユーザに対する IP トラフィックの唯一の許容ルートは、PIM スタブルーティングを設定している device 経由です。
- 冗長 PIM スタブ ルータ トポロジーはサポートされません。PIM スタブ機能では、非冗長アクセス ルータ トポロジーだけがサポートされます。

Auto-RP および BSR の設定に関する制約事項

Auto-RP および BSR を設定する場合は、ネットワーク設定と次の制約事項を考慮してください。

Auto-RP の制約事項

次に、Auto-RP の設定に関する制約事項を示します（ネットワーク設定で使用する場合）。

- PIM をスパース モードまたはデンス モードに設定し、Auto-RP を設定しない場合は、RP を手動で設定する必要があります。
- ルーテッド インターフェイスが SM に設定されていると、すべてのデバイスが自動 RP グループの手動 RP アドレスによって設定されている場合も、自動 RP を使用できます。
- ルーテッドインターフェイスが SM で設定され、**ip pim autorp listener** グローバル コンフィギュレーション コマンドを入力する場合、すべてのデバイスが Auto-RP グループの手動 RP アドレスを使用して設定されていなくても、Auto-RP は引き続き使用できます。

BSR 設定の制約事項

次に、BSR の設定に関する制約事項を示します（ネットワーク設定で使用する場合）。

- 候補 BSR を自動 RP 用の RP マッピング エージェントとして設定します。
- グループ プレフィックスが自動 RP によってアドバタイズされた場合は、異なる RP セットによって処理されたこれらのグループ プレフィックスのサブ範囲が、PIMv2 BSR メカニズムによってアドバタイズされないようにする必要があります。PIMv1 および PIMv2 ドメインが混在する環境では、バックアップ RP で同じグループ プレフィックスが処理されるように設定します。このようにすると、RP マッピング データベースの最長一致検索によって、PIMv2 DR はこれらの PIMv1 DR から異なる RP を選択できなくなります。

Auto-RP および BSR の注意事項と制限事項

次に、Auto-RPおよびBSRの設定に関する制約事項を示します（ネットワーク設定で使用する場合）。

- 使用しているネットワークがすべて Cisco ルータおよびマルチレイヤ devices である場合は、Auto-RP または BSR のいずれかを使用できます。
- ネットワークに他社製のルータがある場合は、BSR を使用する必要があります。
- Cisco PIMv1 および PIMv2 ルータとマルチレイヤ devices、および他社製のルータがある場合は、自動 RP と BSR の両方を使用する必要があります。ネットワークに他のベンダー製のルータが含まれる場合には、シスコの PIMv2 デバイス上に自動 RP マッピング エージェントと BSR を設定します。BSR と他社製の PIMv2 デバイス間のパス上に、PIMv1 デバイスが配置されていないことを確認してください。

（注） PIMv2 は 2 つの方法で使用できます。1 つはバージョン 2 をネットワーク内で排他的に使用する方法、もう 1 つは PIM バージョンの混在環境を採用してバージョン2に移行する方法です。

- ブートストラップメッセージはホップ単位で送信されるため、PIMv1 デバイスの場合、これらのメッセージはネットワーク内の一部のルータおよびマルチレイヤ devices に到達しません。このため、ネットワーク内に PIMv1 デバイスがあり、Cisco ルータおよびマルチレイヤ devices だけが存在する場合は、Auto-RP を使用してください。
- ネットワーク内に他社製のルータが存在する場合は、Cisco PIMv2 ルータまたはマルチレイヤ device に Auto-RP マッピングエージェントおよび BSR を設定します。BSR と他社製の PIMv2 ルータ間のパス上に、PIMv1 デバイスが配置されていないことを確認してください。
- シスコ PIMv1 ルータおよびマルチレイヤ devices と他社製の PIMv2 ルータを相互運用させる場合は、Auto-RP と BSR の両方が必要です。シスコ PIMv2 デバイスを、自動 RP マッピング エージェントと BSR の両方に設定してください。

PIM に関する情報

Protocol Independent Multicast

PIM（Protocol Independent Multicast）プロトコルは、受信側が開始したメンバーシップの現在の IP マルチキャスト サービス モードを維持します。PIM は、特定のユニキャスト ルーティング プロトコルに依存しません。つまり、IP ルーティング プロトコルに依存せず、ユニキャスト ルーティング テーブルへの入力に使用されるユニキャスト ルーティング プロトコル（Enhanced Interior Gateway Routing Protocol（EIGRP）、Open Shortest Path First（OSPF）、Border

Gateway Protocol（BGP）、および スタティック ルート）のいずれも利用できます。PIM は、ユニキャスト ルーティング情報を使用してマルチキャスト転送機能を実行します。

PIM はマルチキャスト ルーティング テーブルと呼ばれていますが、実際には完全に独立したマルチキャスト ルーティング テーブルを作成する代わりに、ユニキャスト ルーティング テーブルを使用してリバース パス フォワーディング（RPF）チェック機能を実行します。他のルーティング プロトコルとは異なり、PIM はルータ間のルーティング アップデートを送受信しません。

PIM は、デンス モードまたはスパース モードで動作します。ルータは、スパース グループとデンス グループの両方を同時に処理できます。これらのモードは、ルータによるマルチキャスト ルーティング テーブルの書き込み方法と、ルータが直接接続された LAN から受信したマルチキャスト パケットの転送方法を決定します。

PIM は 3560 CX スイッチでのみサポートされます。

PIM 転送（インターフェイス）モードについては、次の項を参照してください。

PIM デンス モード（PIM-DM）

PIM デンス モード（PIM-DM）は、プッシュ モデルを使用してマルチキャスト トラフィックをネットワークの隅々にまでフラッディングします。このプッシュ モデルは、データを要求するレシーバを使用せずにデータをレシーバに配信するための方式です。この方式は、ネットワークのあらゆるサブネットにアクティブなレシーバが存在する特定の配置には効率的です。

デンス モードでは、ルータは、他のすべてのルータが特定のグループのマルチキャスト パケットの転送を求めていると想定します。あるルータがマルチキャスト パケットを受信した場合、直接接続されたメンバまたは PIM ネイバーが存在しないときは、ソースにプルーニング メッセージが返送されます。後続のマルチキャスト パケットは、このプルーニング済みのブランチのこのルータにはフラッディングされません。PIM は、ソース ベースのマルチキャスト配信ツリーを構築します。

PIM-DM は最初に、ネットワーク全体にマルチキャスト トラフィックをフラッディングします。ダウンストリーム ネイバーを持たないルータは、不要なトラフィックをプルーニングします。このプロセスは 3 分ごとに繰り返されます。

ルータは、フラッディングとプルーニングのメカニズムを介してデータ ストリームを受信することでステート情報を累積します。これらのデータ ストリームには送信元およびグループの情報が含まれているため、ダウンストリーム ルータがマルチキャスト転送テーブルを構築できます。PIM-DM ではソース ツリー、つまり（S, G）エントリしかサポートしていないため、共有配信ツリーの構築に使用できません。

（注） デンス モードはほとんど使用されておらず、また、その使用もお勧めしません。このため、関連モジュールの設定作業では指定しません。

PIM スパース モード（PIM-SM）

PIM スパース モード（PIM-SM）は、プル モデルを使用してマルチキャスト トラフィックを配信します。明示的にデータを要求したアクティブなレシーバを含むネットワーク セグメントだけがトラフィックを受信します。

スパース モードのインターフェイスは、ダウンストリームのルータから定期的に加入メッセージを受信する場合またはインターフェイスに直接接続のメンバがある場合のみマルチキャスト ルーティング テーブルに追加されます。LAN から転送する場合、グループが認識している RP があれば、SM 動作が行われます。その場合、パケットはカプセル化され、その RP に送信されます。認識している RP がなければ、パケットは DM 方式でフラッディングされます。特定のソースからのマルチキャスト トラフィックが十分である場合、レシーバのファースト ホップ ルータは、ソース ベースのマルチキャスト配信ツリーを構築するために加入メッセージをソースに向けて送信できます。

PIM-SMは、共有ツリー上のデータ パケットを転送することによって、アクティブな送信元に関する情報を配布します。PIM-SM は少なくとも最初は共有ツリーを使用するので、ランデブー ポイント（RP）を使用する必要があります。RP は管理上メットワークで設定されている必要があります。詳細については、ランデブー ポイント（202 ページ）を参照してください。

スパース モードでは、ルータは、トラフィックに対する明示的な要求がない限り、他のルータはグループのマルチキャスト パケットを転送しないと見なします。ホストがマルチキャスト グループに加入すると、直接接続されたルータはRP にPIM 加入メッセージを送信します。RP はマルチキャスト グループを追跡します。マルチキャスト パケットを送信するホストは、そのホストのファースト ホップ ルータによって RP に登録されます。その後、RP は、ソースに加入メッセージを送信します。この時点で、パケットが共有配信ツリー上で転送されます。特定のソースからのマルチキャスト トラフィックが十分である場合、ホストのファースト ホップ ルータは、ソース ベースのマルチキャスト配信ツリーを構築するために加入メッセージをソースに向けて送信できます。

送信元が RP に登録され、データは共有ツリーを下ってレシーバに転送されます。エッジ ルータは、RP を介してソースから共有ツリーでデータ パケットを受信するときに、そのソースについて学習します。次に、エッジ ルータは、そのソースに向けて PIM (S, G)加入メッセージを送信します。リバース パスに沿った各ルータは、RP アドレスのユニキャスト ルーティング メトリックをソース アドレスのメトリックと比較します。送信元アドレスのメトリックの方が良い場合は、ソースに向けて PIM（S, G）加入メッセージを転送します。RP のメトリックと同じ、または RP のメトリックの方が良い場合は、RP と同じ方向に PIM (S, G)加入メッセージが送信されます。この場合、共有ツリーとソース ツリーは一致すると見なされます。

共有ツリーがソースとレシーバの間の最適なパスでない場合、ルータは動的にソース ツリーを作成し、共有ツリーの下方向へのトラフィック フローを停止します。この動作は、ソフトウェアのデフォルトの動作です。ネットワーク管理者は、**ip pim spt-threshold infinity** コマンドを使用して、トラフィックを強制的に共有ツリー上で保持することができます。

PIM-SM は、WAN リンク付きのネットワークを含む、任意のサイズのネットワークに合わせて拡大または縮小します。明示的な加入メカニズムによって、不要なトラフィックが WAN リンクでフラッディングするのを防ぎます。

スパース-デンス モード

インターフェイス上でスパース モードまたはデンス モードを設定すると、そのインターフェイス全体にスパース性またはデンス性が適用されます。ただし、環境によっては、単一リージョン内の一部のグループについては PIM をスパース モードで実行し、残りのグループについてはデンス モードで実行しなければならない場合があります。

デンス モードだけ、またはスパース モードだけをイネーブルにする代わりに、スパース-デンス モードをイネーブルにできます。この場合、グループがデンス モードであればインターフェイスはデンス モードとして処理され、グループがスパース モードであればインターフェイスはスパース モードとして処理されます。インターフェイスがスパース-デンス モードである場合にグループをスパース グループとして処理するには、RP が必要です。

スパース-デンス モードを設定すると、ルータがメンバになっているグループにスパース性またはデンス性の概念が適用されます。

スパース - デンス モードのもう 1 つの利点は、Auto-RP 情報をデンス モードで配信しながら、ユーザー グループのマルチキャスト グループをスパース モード方式で使用できることです。したがって、リーフ ルータ上にデフォルト RP を設定する必要はありません。

インターフェイスがデンス モードで処理される場合、次のいずれかの条件が満たされると、そのインターフェイスはマルチキャスト ルーティング テーブルの発信インターフェイス リストに追加されます。

- インターフェイス上にメンバまたは DVMRP ネイバーが存在する。
- PIM ネイバーが存在し、グループがプルーニングされていない。

インターフェイスがスパース モードで処理される場合、次のいずれかの条件が満たされると、そのインターフェイスはマルチキャスト ルーティング テーブルの発信インターフェイス リストに追加されます。

- インターフェイス上にメンバまたは DVMRP ネイバーが存在する。
- インターフェイス上の PIM ネイバーが明示的な加入メッセージを受信した。

PIM のバージョン

PIMv2 は、PIMv1 と比べて次の点が改善されています。

- マルチキャスト グループごとに、複数のバックアップ ランデブー ポイント（RP）を持つアクティブな RP が 1 つ存在します。この単一の RP で、PIMv1 内の同じグループにアクティブな RP が複数ある場合と同様の処理を行います。
- ブートストラップルータ（BSP）は耐障害性のある、自動化された RP ディスカバリメカニズム、および配信機能を提供します。これらの機能により、ルータおよびマルチレイヤ devices はグループ/RP マッピングを動的に取得できます。
- スパース モード（SM）およびデンス モード（DM）は、インターフェイスではなく、グループに関するプロパティです。

（注） SM または DM のいずれか一方だけでなく、SM-DM（スパース/デンス モード）を使用してください。

- PIM の Join メッセージおよびプルーニング メッセージを使用すると、複数のアドレス ファミリを柔軟に符号化できます。
- 現在以降の機能オプションを符号化するため、クエリー パケットではなく、より柔軟な hello パケット形式が使用されています。
- RP に送信される登録メッセージが、境界ルータによって送信されるか、あるいは指定ルータによって送信されるかを指定します。
- PIM パケットは IGMP パケット内に格納されず、独立したパケットとして処理されます。

PIM スタブ ルーティング

PIM スタブルーティング機能は、すべての device ソフトウェアイメージで使用でき、エンドユーザーの近くにルーテッドトラフィックを移動することでリソースの使用状況を低減させます。

PIM スタブ ルーティング機能は、ディストリビューション レイヤとアクセス レイヤの間のマルチキャストルーティングをサポートします。サポート対象の PIM インターフェイスは、アップリンク PIM インターフェイスと PIM パッシブ インターフェイスの 2 種類です。PIM パッシブ モードに設定されているルーテッド インターフェイスは、PIM 制御トラフィックの通過も転送も行いません。通過させたり転送したりするのは IGMP トラフィックだけです。

PIM スタブルーティングを使用するネットワークでは、ユーザーに対する IP トラフィックの唯一の許容ルートは、PIM スタブルーティングを設定している device 経由です。PIM 受動インターフェイスは、VLAN などのレイヤ 2 アクセス ドメイン、または他のレイヤ 2 デバイスに接続されているインターフェイスに接続されます。直接接続されたマルチキャスト（IGMP）レシーバおよび送信元だけが、レイヤ2アクセス ドメインで許可されます。PIM受動インターフェイスは、受信した PIM 制御パケットを送信または処理しません。

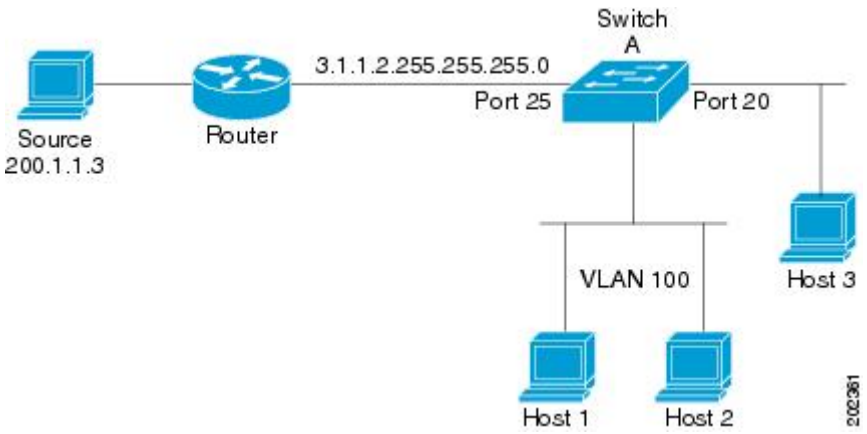
PIM スタブルーティングを使用しているときは、IP マルチキャストルーティングを使用し、device だけを PIM スタブルータとして設定するように、分散ルータおよびリモートルータを設定する必要があります。device は分散ルータ間の伝送トラフィックをルーティングしません。device のルーテッド アップリンク ポートも設定する必要があります。SVI の場合は、device のアップリンクポートを使用できません。SVI アップリンク ポートの PIM が必要な場合は、IP Services フィーチャ セットにアップグレードする必要があります。

（注） また、PIM スタブルーティングを設定するときは、EIGRP スタブルーティングも設定する必要があります。 device

冗長 PIM スタブ ルータ トポロジーはサポートされません。単一のアクセス ドメインにマルチキャスト トラフィックを転送している複数の PIM ルータがある場合、冗長トポロジーが存在します。PIM メッセージはブロックされ、PIM 資産および指定ルータ検出メカニズムは、PIM 受動インターフェイスでサポートされません。PIM スタブ機能では、非冗長アクセス ルータ トポロジーだけがサポートされます。非冗長トポロジーを使用することで、PIM 受動インターフェイスはそのアクセス ドメインで唯一のインターフェイスおよび指定ルータであると想定します。

図 *6 : PIM* スタブ ルータ設定

次の図では、デバイス A ルーテッド アップリンク ポート 25 がルータに接続され、PIM スタブ ルーティングが VLAN 100 インターフェイスとホスト 3 で有効になっています。この設定により、直接接続されたホストはマルチキャスト発信元 200.1.1.3 からトラフィックを受信できます。



IGMP ヘルパー

PIM スタブ ルーティングはルーティングされたトラフィックをエンド ユーザーの近くに移動させ、ネットワーク トラフィックを軽減します。スタブ ルータ（スイッチ）に IGMP ヘルパー機能を設定する方法でもトラフィックを軽減できます。

ip igmp helper-address *ip-address* インターフェイス コンフィギュレーション コマンドを使用してスタブルータ（スイッチ）を設定すると、スイッチによるネクストホップ インターフェイスへのレポート送信をイネーブルにできます。ダウンストリーム ルータに直接接続されていないホストはアップストリーム ネットワークの送信元マルチキャスト グループに加入できます。この機能が設定されていると、マルチキャスト ストリームへの加入を求めるホストからの IGMP パケットはアップストリームのネクストホップ デバイスに転送されます。アップストリームのセントラル ルータは、ヘルパー IGMP レポートまたは leave を受信すると、そのグループの発信インターフェイス リストからインターフェイスの追加または削除を行います。

ランデブー ポイント

ランデブー ポイント（RP）は、デバイスが PIM（Protocol Independent Multicast）スパース モード（SM）で動作している場合にデバイスが実行するロールです。RP が必要になるのは、PIM SM を実行しているネットワークだけです。PIM-SM モデルでは、マルチキャスト データを明示的に要求したアクティブなレシーバを含むネットワーク セグメントだけにトラフィックが転送されます。

マルチキャスト データの配信方法は、PIM デンス モード（PIM DM）とは対照的です。PIM DM では、マルチキャスト トラフィックが最初にネットワークのすべてのセグメントにフラッディングされます。ダウンストリーム ネイバーを持たないルータ、または直接レシーバに接続されているルータは、不要なトラフィックをプルーニングします。

RP は、マルチキャスト データのソースとレシーバの接点として機能します。PIM SIM ネットワークでは、ソースが RP にトラフィックを送信する必要があります。このトラフィックは、それから共有配信ツリーを下ってレシーバに転送されます。デフォルトでは、レシーバのファースト ホップ デバイスがソースを認識すると、ソースに Join メッセージを直接送信し、ソースからレシーバへのソース ベースの配信ツリーを作成します。ソースとレシーバ間の最短パス内に RP が配置されていない限り、このソース ツリーに RP は含まれません。

ほとんどの場合、ネットワークにおける RP の配置は複雑な判断を必要としません。デフォルトでは、RP が必要になるのは、ソースおよびレシーバとの新しいセッションを開始する場合だけです。その結果、RP では、トラフィックのフローまたは処理によるオーバーヘッドはほとんど発生しません。PIM バージョン 2 で実行される処理は PIM バージョン 1 よりも少なくなっています。これは、ソースを定期的に RP に登録するだけでステートを作成できるためです。

Auto-RP

PIM-SMの最初のバージョンでは、すべてのリーフルータ（ソースまたはレシーバに直接接続されたルータ）は、RP の IP アドレスを使用して手動で設定する必要がありました。このような設定は、スタティック RP 設定とも呼ばれます。スタティック RP の設定は、小規模のネットワークでは比較的容易ですが、大規模で複雑なネットワークでは困難を伴う可能性があります。

PIM-SM バージョン 1 の導入に続き、シスコは、Auto-RP 機能を備えた PIM-SM のバージョンを実装しました。Auto-RP は、PIM ネットワークにおけるグループから RP へのマッピングの配信を自動化します。Auto-RP には、次の利点があります。

- さまざまなグループにサービスを提供するために、ネットワーク内で複数の RP を設定することが比較的容易です。
- Auto-RP では、複数の RP 間で負荷を分散し、グループに加入するホストの場所に従って RP を配置できます。
- Auto-RP により、接続の問題の原因となる、矛盾した手動 RP 設定を回避できます。

複数の RP を使用して、異なるグループ範囲にサービスを提供したり、互いにバックアップとしての役割を果たしたりできます。Auto-RP が機能するためには、RP 通知メッセージを RP か

ら受信して競合を解決する RP マッピング エージェントとしてルータが指定されている必要があります。その場合、RP マッピング エージェントは、グループから RP への一貫したマッピングを他のすべてのルータに送信します。これにより、すべてのルータは、サポート対象のグループに使用する RP を自動的に検出します。

（注） PIM をスパース モードまたはデンス モードに設定し、Auto-RP を設定しない場合は、RP を静的に設定する必要があります。

（注） ルータ インターフェイスがスパース モードに設定されている場合、Auto-RP グループに対してすべてのルータが 1 つのスタティック アドレスで設定されているときは、引き続き Auto-RP グループを使用できます。

Auto-RP が機能するためには、RP 通知メッセージを RP から受信して競合を解決する RP マッピング エージェントとしてルータが指定されている必要があります。その後、RP マッピング エージェントは、デンス モード フラッディングにより、グループから RP への一貫したマッピングを他のすべてのルータに送信するようになります。これにより、すべてのルータは、サポート対象のグループに使用する RP を自動的に検出します。インターネット割り当て番号局（IANA）は、224.0.1.39 と 224.0.1.40 という 2 つのグループ アドレスを Auto-RP 用に割り当てています。Auto-RP の利点の 1 つは、指定した RP に対するすべての変更は、RP であるルータ上で設定するだけで、リーフ ルータ上で設定する必要がないことです。Auto-RP のもう 1 つの利点は、ドメイン内で RP アドレスのスコープを設定する機能を提供することです。スコーピングを設定するには、Auto-RP アドバタイズメントに許容されている存続可能時間（TTL）値を定義します。

RP の各設定方式には、それぞれの長所、短所、および複雑度のレベルがあります。従来の IP マルチキャスト ネットワーク シナリオにおいては、Auto-RP を使用して RP を設定することを推奨します。Auto-RP は、設定が容易で、十分にテストされており、安定しているためです。代わりの方法として、スタティック RP、Auto-RP、およびブートストラップ ルータを使用して RP を設定することもできます。

Auto-RP のスパース - デンス モード

Auto-RP の前提条件として、**ip pim sparse-dense-mode** インターフェイス コンフィギュレーション コマンドを使用してすべてのインターフェイスをスパース-デンスモードで設定する必要があります。スパース-デンス モードで設定されたインターフェイスは、マルチキャスト グループの動作モードに応じてスパース モードまたはデンス モードで処理されます。マルチキャスト グループ内に既知の RP が存在する場合、インターフェイスはスパース モードで処理されます。グループ内に既知の RP が存在しない場合、デフォルトでは、インターフェイスはデンス モードで処理され、このインターフェイス上にデータがフラッディングされます（デンス モード フォールバックを回避することもできます。「Configuring Basic IP Multicast」モジュールを参照してください）。

Auto-RP を正常に実装し、224.0.1.39 および 224.0.1.40 以外のグループがデンス モードで動作することを回避するには、「シンク RP」（「ラストリゾート RP」とも呼ばれます）を設定することを推奨します。シンク RP は、ネットワーク内に実際に存在するかどうかわからない静的に設定された RP です。デフォルトでは、Auto-RP メッセージはスタティック RP 設定よりも優先されるため、シンク RP の設定は Auto-RP の動作と干渉しません。未知のソースや予期しないソースをアクティブにできるため、ネットワーク内の可能なすべてのマルチキャスト グループにシンク RP を設定することを推奨します。ソースの登録を制限するように設定された RP がない場合は、グループがデンス モードに戻り、データがフラッディングされる可能性があります。

ブートストラップ ルータ

PIM-SM バージョン 2 では、Auto-RP に続いてブートストラップ ルータ（BSP）と呼ばれるもう 1 つの RP 選択モデルが導入されました。BSR は、RP 機能およびグループの RP 情報のリレーに候補ルータを使用するという点において Auto-RP と同様に動作します。RP 情報は、PIM メッセージ内で伝送される BSR メッセージを通じて配信されます。PIM メッセージは、PIM ルータから PIM ルータへ移動するリンクローカル マルチキャスト メッセージです。この RP 情報を配布するシングル ホップ方式により、BSR では TTL スコーピングを使用できません。BSR は、デンス モード動作に戻るリスクを冒さず、ドメイン内でスコーピング機能を提供しないこと以外は、RP と同様に実行します。

PIM ドメイン境界

IP マルチキャストの普及に伴い、PIMv2 ドメインと別の PIMv2 ドメインが境界を挟んで隣接する場合が増えています。2 つのドメインは同じ RP、BSR、候補 RP、候補 BSR のセットを共有していないことが多いため、PIMv2 BSR メッセージがドメインの内外に流れないようにする必要があります。メッセージのドメイン境界通過を許可すると、通常の BSR 選択メカニズムに悪影響が及んだり、境界に位置するすべてのドメインで単一の BSR が選択されたり、候補 RP アドバタイズメントが混在し、間違ったドメイン内で RP が選択されたりします。

マルチキャスト転送

マルチキャスト トラフィックの転送は、マルチキャスト対応ルータによって行われます。このようなルータは、すべてのレシーバにトラフィックを配信するために、IP マルチキャストがネットワーク上でたどるパスを制御する配信ツリーを作成します。

マルチキャスト トラフィックは、すべてのソースをグループ内のすべてのレシーバに接続する配信ツリー上で、ソースからマルチキャスト グループに流れます。このツリーは、すべてのソースで共有できます（共有ツリー）。または、各ソースに個別の配信ツリーを作成することもできます（ソース ツリー）。共有ツリーは一方向または双方向です。

ソース ツリーと共有ツリーの構造を説明する前に、マルチキャスト ルーティング テーブルで使用する表記について触れておきます。これらの表記には次のものが含まれます。

- (S, G) = (マルチキャスト グループ G のユニキャスト ソース, マルチキャスト グループ G)
- (*, G) = (マルチキャスト グループ G のすべてのソース, マルチキャスト グループ G)

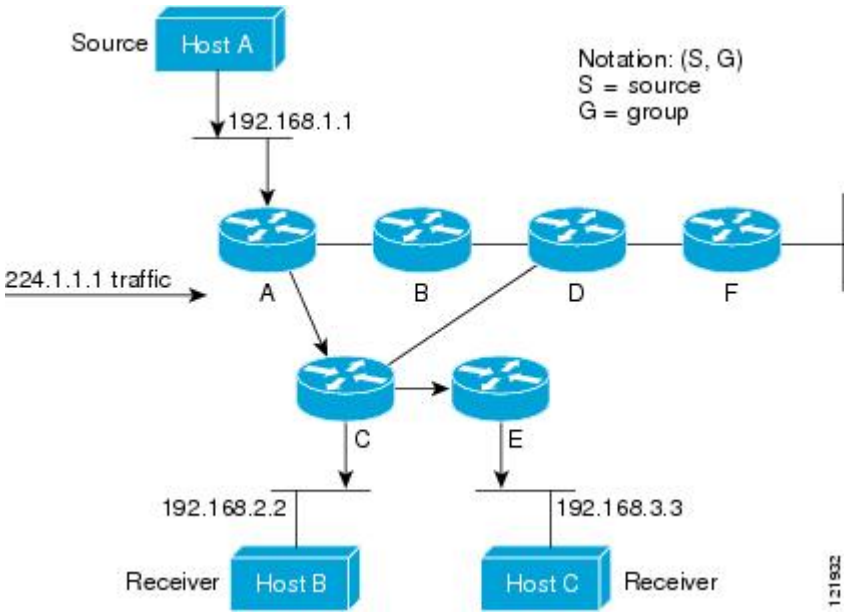
(S, G) という表記（「S カンマ G」と読みます）は、最短パス ツリーの列挙です。S はソースの IP アドレス、G はマルチキャスト グループ アドレスを表します。

共有ツリーは (*, G) で表されます。ソース ツリーは (S, G) で表され、常にソースでルーティングされます。

マルチキャスト配信のソース ツリー

マルチキャスト配信ツリーの最も単純な形式は、ソース ツリーです。ソース ツリーは、ソース ホストをルートとし、ネットワークを介してレシーバに接続するスパニングツリーを形成するブランチを持ちます。このツリーはネットワーク上での最短パスを使用するため、最短パス ツリー（SPT）とも呼ばれます。

次の図に、ソース（ホスト A）をルートとし、2 つのレシーバ（ホスト B およびホスト C）に接続するグループ 224.1.1.1 の SPT の例を示します。



標準表記を使用すると、図の例の SPT は（192.168.1.1, 224.1.1.1）となります。

(S, G) という表記は、各グループに送信する個々のソースに個別の SPT が存在することを意味します。

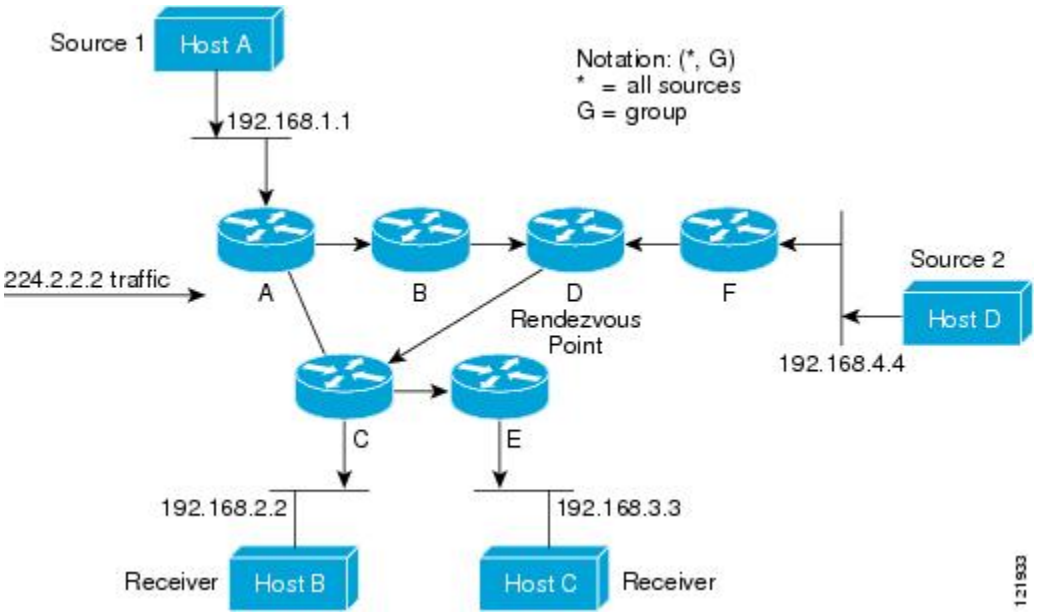
マルチキャスト配信の共有ツリー

ソースをルートとするソース ツリーとは異なり、共有ツリーはネットワーク内の選択されたポイントに配置された単一の共通ルートを使用します。この共有されたルートは、ランデブー ポイント（RP）と呼ばれます。

次の図に、ルータ D にルートが配置されたグループ 224.2.2.2 の共有ツリーを示します。この共有ツリーは単方向です。ソース トラフィックは、ソース ツリー上の RP に向けて送信されます。このトラフィックは、次に RP から共有ツリーを下方向に転送され、すべてのレシーバに

到達します（レシーバがソースと RP の間に配置されていない場合は、直接サービスが提供されます）。

図 7: 共有ツリー



この例では、ソース（ホスト A およびホスト D）からのマルチキャスト トラフィックがルート（ルータ D）に移動した後に共有ツリーから 2 つのレシーバ（ホスト B およびホスト C）へと到達します。マルチキャスト グループ内のすべての送信元が一般的な共有ツリーを使用するため、(*, G) というワイルドカード表記（「アスタリスク、カンマ、G」と読みます）でそのツリーを表します。この場合、* はすべてのソースを意味し、G はマルチキャスト グループを表します。したがって、図の共有ツリーは (*, 224.2.2.2) と表記します。

ソース ツリーと共有ツリーは、どちらもループフリーです。ツリーが分岐する場所でのみ、メッセージが複製されます。マルチキャスト グループのメンバは常に加入または脱退する可能性があるため、配信ツリーを動的に更新する必要があります。特定のブランチに存在するすべてのアクティブ レシーバが特定のマルチキャスト グループに対してトラフィックを要求しなくなると、ルータは配信ツリーからそのブランチをプルーニングし、そのブランチから下方向へのトラフィック転送を停止します。そのブランチの特定のレシーバがアクティブになり、マルチキャスト トラフィックを要求すると、ルータは配信ツリーを動的に変更し、トラフィック転送を再開します。

ソース ツリーの利点

ソース ツリーには、ソースとレシーバの間に最適なパスを作成するという利点があります。この利点により、マルチキャスト トラフィックの転送におけるネットワーク遅延を最小限に抑えることができます。ただし、この最適化は代償を伴います。ルータがソースごとにパス情報を維持する必要があるのです。何千ものソース、何千ものグループが存在するネットワークでは、このオーバーヘッドがすぐにルータ上でのリソースの問題につながる可能性があります。ネットワーク設計者は、マルチキャスト ルーティング テーブルのサイズによるメモリ消費について考慮する必要があります。

共有ツリーの利点

共有ツリーには、各ルータにおいて要求されるステートの量が最小限に抑えられるという利点があります。この利点により、共有ツリーだけが許容されるネットワークの全体的なメモリ要件が緩和されます。共有ツリーの欠点は、特定の状況でソースとレシーバの間のパスが最適パスではなくなり、パケット配信に遅延を生じる可能性があることです。たとえば、上の図のホスト A（ソース 1）とホスト 2（レシーバ）間の最短パスはルータ A とルータ B です。共有ツリーのルートとしてルータ D を使用するため、トラフィックはルータ A、B、D、そして次に C を通過する必要があります。ネットワーク設計者は、共有ツリー専用環境を実装する際にランデブー ポイント（RP）の配置を慎重に考慮する必要があります。

ユニキャスト ルーティングでは、トラフィックは、ネットワーク上でソースから宛先ホストまでの単一パスに沿ってルーティングされます。ユニキャスト ルータは、ソース アドレスを考慮せず、宛先アドレスおよびその宛先へのトラフィックの転送方法だけを考慮します。ルータは、ルーティング テーブル全体をスキャンして宛先アドレスを取得し、適正なインターフェイスから宛先の方向へユニキャスト パケットのコピーを転送します。

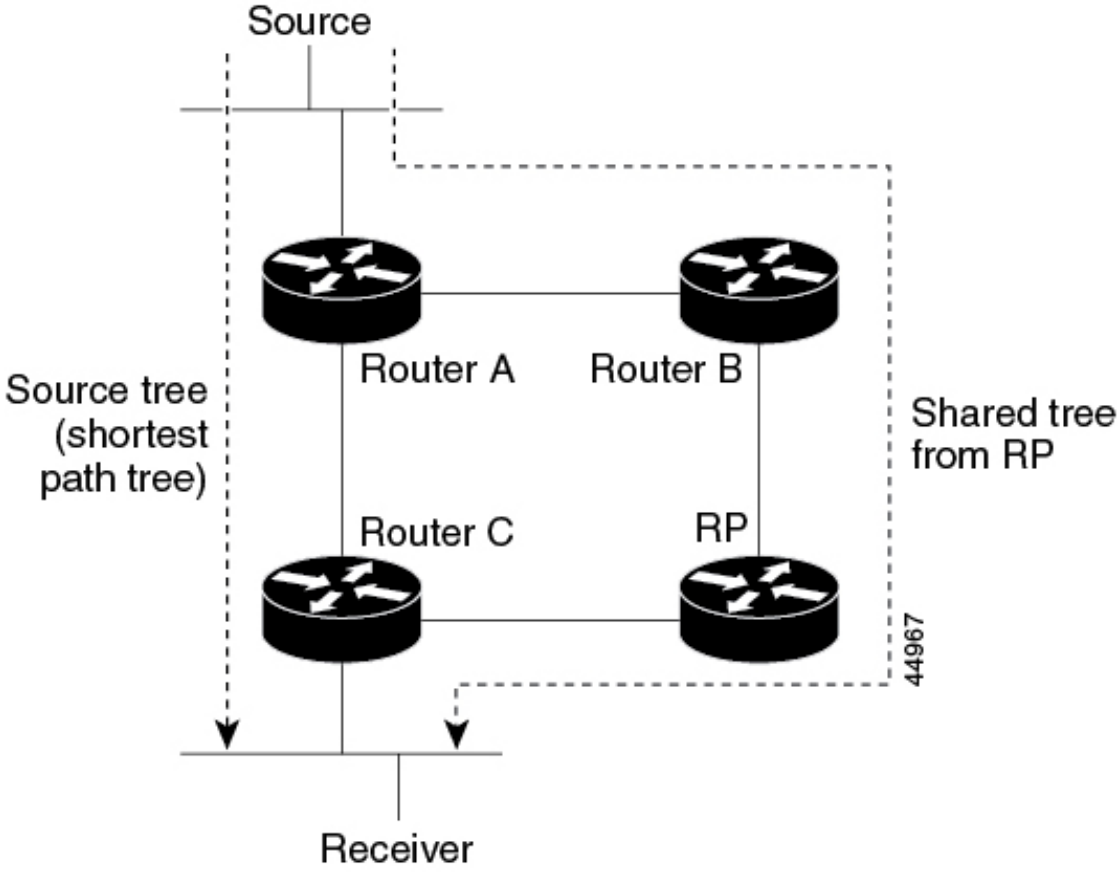
マルチキャスト転送では、ソースは、マルチキャスト グループ アドレスによって表される任意のホスト グループにトラフィックを送信します。マルチキャスト ルータは、どの方向が（ソースへ向かう）アップストリーム方向で、どの方向（1 方向または複数の方向）が（レシーバへ向かう）ダウンストリーム方向であるかを決定する必要があります。複数のダウンストリーム パスがある場合、ルータはパケットを複製し、それを適切なダウンストリーム パス（最善のユニキャスト ルート メトリック）で下方向に転送します。これらのパスがすべてであるとは限りません。レシーバの方向ではなく、ソースから遠ざかる方向へのマルチキャスト トラフィック転送は、Reverse Path Forwarding（RPF）と呼ばれます。RPF については、次の項を参照してください。

PIM 共有ツリーおよびソース ツリー

デフォルトでは、グループのメンバーで受信されるデータは、RP でルーティングされた単一のデータ配信ツリーを経由して、送信側からグループに送られます。

図 *8:* 共有ツリーおよびソース ツリー（最短パスツリー）

次の図に、このタイプの共有配信ツリーを示します。送信側からのデータは、RP に配信され、その共有ツリーに加入しているグループ メンバに配布されます。



データ レートによって保証されている場合は、送信元でルーティングされるデータ配信ツリーを、共有ツリーのリーフ ルータ（ダウンストリーム接続がないルータ）で使用できます。このタイプの配信ツリーは、SPT または送信元ツリーと呼ばれます。デフォルトでは、ソフトウェアは、送信元から最初のデータパケットを受信すると、ソースツリーに devices します。

共有ツリーから送信元ツリーへの移動プロセスは、次のとおりです。

1. レシーバがグループに加入します。リーフ ルータ C は Join メッセージを RP に向けて送信します。
2. RP はルータ C とのリンクを発信インターフェイス リストに格納します。
3. 送信元がデータを送信します。ルータ A はデータをカプセル化して登録メッセージに格納し、RP に送信します。
4. RP はデータをルータ C に向けて共有ツリーの下方向に転送し、送信元に向けて Join メッセージを送信します。この時点で、データはルータ C に 2 回着信する可能性があります（カプセル化されたデータ、およびネイティブ状態のデータ）。

5. データがネイティブ状態（カプセル化されていない状態）で着信すると、RP は登録停止メッセージをルータ A に送信します。
6. デフォルトでは、最初のデータ パケット受信時に、ルータ C が Join メッセージを送信元に送信するよう要求します。
7. ルータ C が（S, G）でデータを受信すると、ルータ C は共有ツリーの上位方向にある送信元に prune メッセージを送信します。
8. RP が（S, G）の発信インターフェイスからルータ C へのリンクを削除します。RP は送信元に向けてプルーニング メッセージを送信します。

送信元および RP に join および prune メッセージが送信されます。これらのメッセージはホップ単位で送信され、送信元または RP へのパス上にある各 PIM デバイスで処理されます。register および register-stop メッセージは、ホップバイホップで送信されません。これらのメッセージは、送信元に直接接続されている指定ルータによって送信され、グループの RP によって受信されます。

グループへ送信する複数の送信元で、共有ツリーが使用されます。共有ツリー上に存在するように、PIM デバイスを設定できます。

最初のデータ パケットがラスト ホップ ルータに着信すると、共有ツリーからソース ツリーへと変更されます。この変更は、**ip pim spt-threshold** グローバル コンフィギュレーション コマンドを使用して設定したしきい値によって異なります。

SPT には共有ツリーよりも多くのメモリが必要ですが、遅延が短縮されます。SPT の使用を延期することもできます。リーフ ルータを SPT にすぐ移動せず、トラフィックがしきい値に最初に到達したあとで移動するように指定できます。

PIM リーフ ルータが、指定グループの SPT に加入する時期を設定できます。送信元の送信速度が指定速度（キロビット/秒）以上の場合、マルチレイヤ スイッチは PIM Join メッセージを送信元に向けて送信し、送信元ツリー（SPT）を構築します。送信元からのトラフィック速度がしきい値を下回ると、リーフ ルータは共有ツリーに再び切り替わり、プルーニング メッセージを送信元に送信します。

SPT しきい値を適用するグループを指定するには、グループ リスト（標準アクセス リスト）を使用します。値 0 を指定する場合、またはグループ リストを使用しない場合、しきい値はすべてのグループに適用されます。

Reverse Path Forwarding

ユニキャスト ルーティングでは、トラフィックは、ネットワーク上でソースから宛先ホストまでの単一パスに沿ってルーティングされます。ユニキャスト ルータは、ソース アドレスを考慮せず、宛先アドレスおよびその宛先へのトラフィックの転送方法だけを考慮します。ルータは、ルーティング テーブル全体をスキャンして宛先ネットワークを取得し、適正なインターフェイスから宛先の方向へユニキャスト パケットのコピーを転送します。

マルチキャスト転送では、ソースは、マルチキャスト グループ アドレスによって表される任意のホスト グループにトラフィックを送信します。マルチキャスト ルータは、どの方向が（ソースへ向かう）アップストリーム方向で、どの方向（1 方向または複数の方向）が（レシー

バへ向かう）ダウンストリーム方向であるかを決定する必要があります。複数のダウンストリームパスがある場合、ルータはパケットを複製し、それを適切なダウンストリームパス（最善のユニキャスト ルート メトリック）で下方向に転送します。これらのパスがすべてであるとは限りません。レシーバの方向ではなく、ソースから遠ざかる方向へのマルチキャスト トラフィック転送は、Reverse Path Forwarding（RPF）と呼ばれます。RPF は、マルチキャスト データグラムの転送に使用されるアルゴリズムです。

Protocol Independent Multicast（PIM）は、ユニキャスト ルーティング情報を使用して、レシーバからソースへ向かうリバース パスに沿って配信ツリーを作成します。その後、マルチキャスト ルータは、その配信ツリーに沿ってソースからレシーバにパケットを転送します。RPF は、マルチキャスト転送における重要な概念です。RPF により、ルータは、配信ツリーの下方向へ正しくマルチキャスト トラフィックを転送できます。RPF は、既存のユニキャスト ルーティング テーブルを使用して、アップストリーム ネイバーとダウンストリーム ネイバーを決定します。ルータは、アップストリーム インターフェイスで受信した場合にのみ、マルチキャスト パケットを転送します。この RPF チェックにより、配信ツリーがループフリーであることを保証できます。

RPF チェック

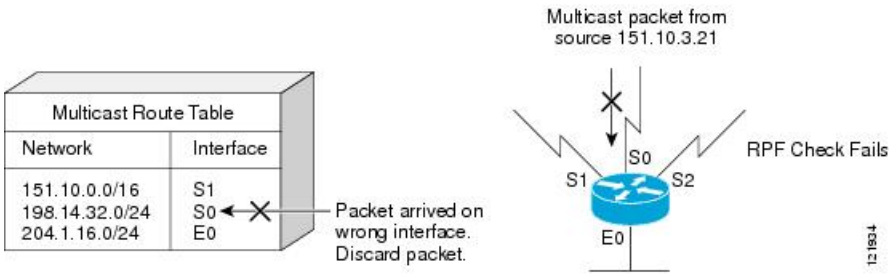
マルチキャスト パケットがルータに到達すると、ルータはそのパケットに対して RPF チェックを実行します。RPF チェックが成功すると、パケットが転送されます。そうでない場合、パケットはドロップされます。

ソース ツリーを下方向へ流れるトラフィックに対する RPF チェック手順は次のとおりです。

1. ルータは、ユニキャスト ルーティング テーブルでソース アドレスを検索して、ソースへのリバース パス上にあるインターフェイスにパケットが到達したかどうかを判定します。
2. ソースに戻すインターフェイスにパケットが到達した場合、RPF チェックは成功し、マルチキャスト ルーティング テーブル エントリの発信インターフェイス リストに示されているインターフェイスからパケットが転送されます。
3. ステップ 2 で RPF チェックに失敗した場合は、パケットがドロップされます。

図に、RPF チェックの失敗例を示します。

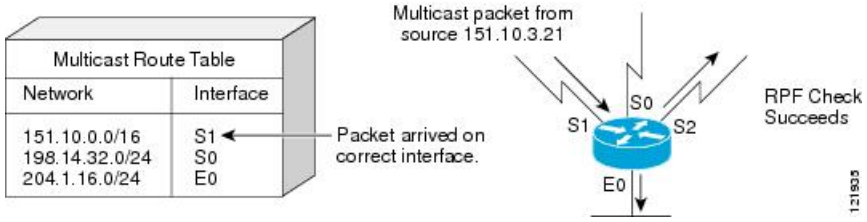
図 *9 : RPF* チェックの失敗



図に示すように、ソース 151.10.3.21 からのマルチキャスト パケットはシリアル インターフェイス 0（S0）上で受信されています。ユニキャスト ルート テーブルのチェック結果は、このルータが 151.10.3.21 にユニキャスト データを転送するために使用するインターフェイスは S1 であることを示しています。パケットはインターフェイス S0 に到達しているため、このパケットは廃棄されます。

図に RPF チェックの成功例を示します。

図 *10 : RPF* チェックの成功



この例では、マルチキャスト パケットはインターフェイス S1 に到達しています。ルータはユニキャスト ルーティング テーブルを参照し、S1 が適正なインターフェイスであることを知ります。RPF チェックが成功し、パケットが転送されます。

PIM ルーティングのデフォルト設定

次の表に、device の PIM ルーティングのデフォルト設定を示します。

表 *27 :* マルチキャスト ルーティングのデフォルト設定

機能	デフォルト設定
マルチキャスト ルーティング	すべてのインターフェイスでディセーブル
PIM のバージョン	バージョン 2
PIM モード	モードは未定義
PIM スタブ ルーティング	未設定
PIM RP アドレス	未設定
PIM ドメイン境界	ディセーブル。
PIM マルチキャスト境界	なし
候補 BSR	ディセーブル。
候補 RP	ディセーブル。
SPT しきい値レート	0 kb/s
PIM ルータ クエリー メッセージ インターバル	30 秒

PIM の設定方法

PIM スタブ ルーティングのイネーブル化

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip pim passive**
5. **end**
6. **show ip pim interface**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	PIM スタブ ルーティングをイネーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip pim passive** 例： `スイッチ(config-if)# ip pim passive`	インターフェイスに PIM スタブ機能を設定します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ip pim interface** 例： スイッチ# **show ip pim interface**	（任意）各インターフェイスで有効になっている PIM スタブを表示します。
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ランデブー ポイントの設定

インターフェイスがスパース - デンス モードで、グループをスパース グループとして扱う場合には、ランデブー ポイント（RP）を設定する必要があります。次の方法を使用できます。

- RP をマルチキャスト グループに手動で割り当てる
- PIMv1 から独立した、以下を含むスタンドアロンとしてのシスコ独自のプロトコル
  - 新規インターネットワークでの自動 RP の設定
  - 既存のスパースモード クラウドへの自動 RP の追加
  - 問題のある RP への Join メッセージの送信禁止
  - 着信 RP アナウンスメント メッセージのフィルタリング
- Internet Engineering Task Force（IETF）の標準追跡プロトコルの使用（PIMv2 BSR の設定を含む）

（注） 動作中の PIM バージョン、およびネットワーク内のルータ タイプに応じて、自動 RP、BSR、またはこれらを組み合わせて使用できます。ネットワーク内の異なるバージョンの PIM を利用する方法については、「PIMv1 および PIMv2 の相互運用性」のセクションを参照してください。

マルチキャスト グループへの RP の手動割り当て

ダイナミック メカニズム（自動 RP や BSR など）を使用してグループのランデブー ポイント（RP）を取得する場合、RP を手動で割り当てる必要はありません。

マルチキャスト トラフィックの送信側は、送信元の先頭ホップ ルータ（指定ルータ）から受信して RP に転送される登録メッセージを通し、自身の存在をアナウンスします。マルチキャストパケットの受信側はRPを使用し、マルチキャスト グループに加入します。この場合は、明示的な Join メッセージが使用されます。

（注） RPはマルチキャスト グループのメンバーではなく、マルチキャスト送信元およびグループ メンバーの合流地点として機能します。

アクセス リストで定義される複数のグループに、単一の RP を設定できます。グループに RP が設定されていない場合、マルチレイヤdeviceはデンスとしてグループに応答し、デンスモードの PIM 技術を使用します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip pim rp-address** *ip-address* [*access-list-number*] [**override**]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip pim rp-address** *ip-address* [*access-list-number*] [**override**] 例： `スイッチ(config)# ip pim rp-address 10.1.1.1 20 override`	PIM RP のアドレスを設定します。 デフォルトで、PIM RP アドレスは設定されていません。すべてのルータおよびマルチレイヤdevices（RP を含む）で、RP の IP アドレスを設定する必要があります。 （注） グループに RP が設定されていない場合、deviceは PIM DM 技術を使用し、グループをデンスとして処理します。 1 台の PIM デバイスを、複数のグループの RP にできます。1 つの PIM ドメイン内で一度に使用できる RP アドレスは、1 つだけです。アクセス リスト条件により、デバイスがどのグループの RP であるかを指定します。 • *ip-address* には、RP のユニキャスト アドレスをドット付き 10 進表記で入力します。 •（任意）*access-list-number* を指定する場合は、1 ～ 99 の IP 標準アクセス リスト番号を入力します。アクセス リストが設定されていない場合は、すべてのグループに RP が使用されます。 •（任意）**override** キーワードを指定すると、このコマンドによって設定された RP と、自動 RP または BSR で取得された RP との間に矛盾が生じた場合に、このコマンドによって設定された RP が優先されます。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： `スイッチ(config)# access-list 25 permit 10.5.0.1 255.224.0.0`	標準アクセスリストを作成し、コマンドを必要な回数だけ実行します。 • *access-list-number* には、ステップ 2 で指定したアクセス リスト番号を入力します。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。 • **permit** キーワードは、条件が一致した場合にアクセスを許可します。

	コマンドまたはアクション	目的
		• *source* には、RP が使用されるマルチキャスト グループのアドレスを入力します。 • （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

新規インターネットワークでの Auto-RP の設定

新規インターネットワーク内に自動 RP を設定している場合は、すべてのインターフェイスが SM-DM に設定されるため、デフォルトの RP は不要です。

（注） PIM ルータをローカル グループの RP として設定する場合は、次の手順のステップ 3 を省略します。

手順の概要

1. **enable**
2. **show running-config**
3. **configure terminal**
4. **ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*
5. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]

6. **ip pim send-rp-discovery scope** *ttl*
7. **end**
8. **show running-config**
9. **show ip pim rp mapping**
10. **show ip pim rp**
11. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** 例： スイッチ# **show running-config**	すべての PIM デバイス上でデフォルトの RP が設定されていること、および RP が SM ネットワーク内にあることを確認します。RP は、**ip pim rp-address** グローバル コンフィギュレーション コマンドによって設定済みです。 （注） SM-DM 環境の場合、このステップは不要です。 選択された RP は接続が良好で、ネットワークで使用可能となる必要があります。この RP は、グローバル グループ（224.x.x.x やその他のグローバル グループなど）に対して使用されます。この RP で処理されるグループ アドレス範囲は再設定しないでください。自動 RP によって動的に検出された RP は、静的に設定された RP よりも優先されます。ローカル グループ用に 2 番めの RP を使用することもできます。
ステップ 3	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds* 例： スイッチ(config)# **ip pim send-rp-announce**	別の PIM デバイスをローカル グループの候補 RP として設定します。 • *interface-id* には、RP アドレスを識別するインターフェイス タイプおよび番号を入力します。

	コマンドまたはアクション	目的
	gigabitethernet 1/0/5 scope 20 group-list 10 interval 120	有効なインターフェイスは、物理ポート、ポート チャネル、VLAN などです。 • **scope** *ttl* には、ホップの存続可能時間の値を指定します。RP アナウンス メッセージがネットワーク内のすべてのマッピング エージェントに到達するように、十分な大きさのホップ数を入力します。デフォルト設定はありません。指定できる範囲は 1 ～ 255 です。 • **group-list** *access-list-number* には、1 ～ 99 の範囲で標準の IP アクセスリスト番号を入力します。アクセス リストが設定されていない場合は、すべてのグループに RP が使用されます。 • **interval** *seconds* には、アナウンスメントメッセージを送信する頻度を指定します。デフォルトは 60 秒です。指定できる範囲は 1 ～ 16383 です。
ステップ 5	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： `スイッチ(config)# access-list 10 permit 10.10.0.0`	標準アクセス リストを作成し、コマンドを必要な回数だけ実行します。 • *access-list-number* には、ステップ 3 で指定したアクセス リスト番号を入力します。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。 • **permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *source* には、RP が使用されるマルチキャスト グループのアドレス範囲を入力します。 • （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 （注） アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。
ステップ 6	**ip pim send-rp-discovery scope** *ttl* 例：	接続が中断される可能性がない device を検索し、RP マッピング エージェントの役割を割り当てます。

	コマンドまたはアクション	目的
	スイッチ(config)# **ip pim send-rp-discovery scope 50**	**scope** *ttl* には、ホップの存続可能時間の値を指定し、RP ディスカバリパケットを制限します。ホップ数内にあるすべてのデバイスは、送信元デバイスから自動 RP ディスカバリ メッセージを受信します。これらのメッセージは他のデバイスに対し、矛盾（グループ/RP範囲の重なりなど）を回避するために使用されるグループ/RPマッピングを通知します。デフォルト設定はありません。指定できる範囲は 1 ～ 255 です。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 9	**show ip pim rp mapping** 例： スイッチ# **show ip pim rp mapping**	関連するマルチキャスト ルーティング エントリとともに保管されているアクティブなRPを表示します。
ステップ 10	**show ip pim rp** 例： スイッチ# **show ip pim rp**	ルーティング テーブルに保管されている情報を表示します。
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

既存のスパース モード クラウドへの Auto-RP の追加

ここでは、最初に自動 RP を既存の SM クラウドに導入し、既存のマルチキャスト インフラストラクチャができるだけ破壊されないようにする方法について説明します。

この手順は任意です。

手順の概要

1. **enable**
2. **show running-config**
3. **configure terminal**
4. **ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*
5. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
6. **ip pim send-rp-discovery scope** *ttl*
7. **end**
8. **show running-config**
9. **show ip pim rp mapping**
10. **show ip pim rp**
11. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show running-config** 例： `スイッチ# show running-config`	すべての PIM デバイス上でデフォルトの RP が設定されていること、および RP が SM ネットワーク内にあることを確認します。RP は、**ip pim rp-address** グローバル コンフィギュレーション コマンドによって設定済みです。 （注） SM-DM 環境の場合、このステップは不要です。 選択された RP は接続が良好で、ネットワークで使用可能となる必要があります。この RP は、グローバル グループ（224.x.x.x やその他のグローバル グループなど）に対して使用されます。この RP で処理されるグループ アドレス範囲は再設定しないでください。自動 RP によって動的に検出された RP は、静的に設定された RP よりも優先されます。ローカル グループ用に 2 番めの RP を使用することもできます。
ステップ 3	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds* 例： スイッチ(config)# **ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120**	別の PIM デバイスをローカル グループの候補 RP として設定します。 • *interface-id* には、RP アドレスを識別するインターフェイスタイプおよび番号を入力します。有効なインターフェイスは、物理ポート、ポート チャネル、VLAN などです。 • **scope** *ttl* には、ホップの存続可能時間の値を指定します。RP アナウンス メッセージがネットワーク内のすべてのマッピング エージェントに到達するように、十分な大きさのホップ数を入力します。デフォルト設定はありません。指定できる範囲は 1 ～ 255 です。 • **group-list** *access-list-number* には、1 ～ 99 の範囲で標準の IP アクセスリスト番号を入力します。アクセス リストが設定されていない場合は、すべてのグループに RP が使用されます。 • **interval** *seconds* には、アナウンスメントメッセージを送信する頻度を指定します。デフォルトは 60 秒です。指定できる範囲は 1 ～ 16383 です。
ステップ 5	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： スイッチ(config)# **access-list 10 permit 224.0.0.0 15.255.255.255**	標準アクセス リストを作成し、コマンドを必要な回数だけ実行します。 • *access-list-number* には、ステップ 3 で指定したアクセス リスト番号を入力します。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。 • **permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *source* には、RP が使用されるマルチキャスト グループのアドレス範囲を入力します。 • （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。

	コマンドまたはアクション	目的
ステップ 6	**ip pim send-rp-discovery scope** *ttl* 例： スイッチ(config)# **ip pim send-rp-discovery scope 50**	接続が中断される可能性がないdeviceを検索し、RP マッピング エージェントの役割を割り当てます。 **scope** *ttl* には、ホップの存続可能時間の値を指定し、RP ディスカバリパケットを制限します。ホップ数内にあるすべてのデバイスは、送信元デバイスから自動 RP ディスカバリ メッセージを受信します。これらのメッセージは他のデバイスに対し、矛盾（グループ/RP範囲の重なりなど）を回避するために使用されるグループ/RPマッピングを通知します。デフォルト設定はありません。指定できる範囲は 1 ～ 255 です。 （注） RPマッピングエージェントとして設定された device を削除するには、**no ip pim send-rp-discovery** グローバル コンフィギュレーション コマンドを使用します。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 9	**show ip pim rp mapping** 例： スイッチ# **show ip pim rp mapping**	関連するマルチキャスト ルーティング エントリとともに保管されているアクティブなRPを表示します。
ステップ 10	**show ip pim rp** 例： スイッチ# **show ip pim rp**	ルーティング テーブルに保管されている情報を表示します。
ステップ 11	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

単一スタティック RP でのスパース モードの設定（CLI）

ランデブー ポイント（RP）は Protocol Independent Multicast Sparse Mode（PIM-SM）を実行しているネットワークで必要です。PIM-SM でトラフィックは、明示的にマルチキャスト データを要求したアクティブなレシーバを持つネットワーク セグメントにのみ転送されます。

ここでは、単一のスタティック RP を使用したスパース モードの設定方法について説明します。

始める前に

単一のスタティック RP を使用してスパース モードを設定するときに必要なすべてのアクセス リストは、設定作業を開始する前に設定しておく必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip multicast-routing** [**distributed**]
4. **interface** *type number*
5. **ip pim sparse-mode**
6. IP マルチキャストを使用するすべてのインターフェイスでステップ 1 ～ 5 を繰り返します。
7. **exit**
8. **ip pim rp-address** *rp-address* [*access-list*] [**override**]
9. **end**
10. **show ip pim rp** [**mapping**] [*rp-address*]
11. **show ip igmp groups** [*group-name* | *group-address*| *interface-type interface-number*] [**detail**]
12. **show ip mroute**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**ip multicast-routing** [**distributed**] 例： `device(config)# ip multicast-routing`	IP マルチキャスト ルーティングを有効にします。 • **distributed** キーワードを使用して、マルチキャスト分散スイッチングを有効にします。
ステップ 4	**interface** *type number* 例： `device(config)# interface gigabitethernet 1/0/0`	PIM をイネーブルにできるホストに接続されているインターフェイスを選択します。
ステップ 5	**ip pim sparse-mode** 例： `device(config-if)# ip pim sparse-mode`	インターフェイスに対して PIM をイネーブルにします。スパース モードを使用する必要があります。
ステップ 6	IP マルチキャストを使用するすべてのインターフェイスでステップ 1 ～ 5 を繰り返します。	--
ステップ 7	**exit** 例： `device(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 8	**ip pim rp-address** *rp-address* [*access-list*] [**override**] 例： `device(config)# ip pim rp-address 192.168.0.0`	特定のグループの PIM RP のアドレスを設定します。 • マルチキャスト グループを RP に静的にマッピングされるよう定義する標準アクセス リストに名前を付けたり、番号を指定するために、オプションの *access-list* 引数が使用されます。 （注） アクセス リストが定義されていない場合、RP がすべてのマルチキャスト グループ 224/4 にマッピングされます。 • ダイナミックとスタティックのグループと RP 間のマッピングが共に使用され、RP アドレスが競合している場合、スタティックのグループと RP 間のマッピングに設定された RP アドレスが優先されるよう指定するには、オプションの **override** キーワードを使用します。

	コマンドまたはアクション	目的
		（注） **override** キーワードが指定されておらず、RP アドレスが競合している場合、ダイナミック グループと RP 間のマッピングがスタティック グループと RP 間のマッピングよりも優先されます。
ステップ 9	**end** 例： `device(config)# end`	現在のコンフィギュレーション セッションを終了して、EXEC モードに戻ります。
ステップ 10	**show ip pim rp** [**mapping**] [*rp-address*] 例： `device# show ip pim rp mapping`	（任意）ネットワークで既知の RP を表示し、ルータが各 RP について学習する方法を示します。
ステップ 11	**show ip igmp groups** [*group-name* | *group-address*| *interface-type interface-number*] [**detail**] 例： `device# show ip igmp groups`	（任意）ルータに直接接続されているレシーバと IGMP によって学習されたレシーバを持つマルチキャスト グループを表示します。 • レシーバ情報が結果の画面に表示されるには、レシーバがこのコマンドが発行された時点でネットワーク上でアクティブである必要があります。
ステップ 12	**show ip mroute** 例： `device# show ip mroute`	（任意）IP mroute テーブルの内容を表示します。

問題のある RP への Join メッセージの送信禁止

ip pim accept-rp コマンドがネットワーク全体に設定されているかどうかを判別するには、**show running-config** 特権 EXEC コマンドを使用します。**ip pim accept-rp** コマンドが設定されていないデバイスがある場合は、後でこの問題を解決できます。ルータまたはマルチレイヤdevicesが **ip pim accept-rp** コマンドによってすでに設定されている場合は、このコマンドを再入力し、新規にアドバタイズされる RP を許可する必要があります。

自動 RP によってアドバタイズされるすべての RP を許可し、他のすべての RP をデフォルトで拒否するには、**ip pim accept-rp auto-rp** グローバル コンフィギュレーション コマンドを使用します。

この手順は任意です。

着信 RP アナウンスメント メッセージのフィルタリング

マッピング エージェントにコンフィギュレーション コマンドを追加すると、故意に不正設定されたルータが候補 RP として動作し問題を引き起こさないようにできます。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number*
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number* 例： `スイッチ(config)# ip pim rp-announce-filter rp-list 10 group-list 14`	着信 RP アナウンスメント メッセージをフィルタリングします。 ネットワーク内のマッピング エージェントごとに、このコマンドを入力します。このコマンドを使用しないと、すべての着信 RP アナウンスメント メッセージがデフォルトで許可されます。 **rp-list** *access-list-number* には、候補 RP アドレスのアクセスリストを設定します。アクセスリストが許可されている場合は、**group-list** *access-list-number* 変数で指定されたグループ範囲に対してアクセスリストを使用できます。この変数を省略すると、すべてのマルチキャスト グループにフィルタが適用されます。

	コマンドまたはアクション	目的
		複数のマッピングエージェントを使用する場合は、グループ/RP マッピング情報に矛盾が生じないようにするため、すべてのマッピングエージェント間でフィルタを統一する必要があります。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： `スイッチ(config)# access-list 10 permit 10.8.1.0 255.255.224.0`	標準アクセスリストを作成し、コマンドを必要な回数だけ実行します。 • *access-list-number* には、ステップ 2 で指定したアクセス リスト番号を入力します。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。 • **permit** キーワードは、条件が一致した場合にアクセスを許可します。 • どのルータおよびマルチレイヤdevicesからの候補 RP アナウンスメント（rp-list アクセスコントロールリスト（ACL））がマッピングエージェントによって許可されるかを指定するアクセスリストを作成します。 • 許可または拒否するマルチキャスト グループの範囲を指定するアクセスリスト（グループリスト ACL）を作成します。 • *source* には、RP が使用されるマルチキャスト グループのアドレス範囲を入力します。 • （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例：	入力を確認します。

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

PIMv2 BSR の設定

PIMv2 BSR を設定するプロセスには、次のオプションの作業が含まれることがあります。

- PIM ドメイン境界の定義
- IP マルチキャスト境界の定義
- 候補 BSR の設定
- 候補 RP の設定

PIM ドメイン境界の定義

PIM ドメイン境界を設定するには、次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip pim bsr-border**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip pim bsr-border** 例： `スイッチ(config-if)# ip pim bsr-border`	PIM ドメイン用の PIM ブートストラップ メッセージ境界を定義します。 境界に位置する他の PIM ドメインに接続されているインターフェイスごとに、このコマンドを入力します。このコマンドを実行すると、deviceは、このインターフェイス上で PIMv2 BSR メッセージを送受信しないように指示されます。 （注） PIM 境界を削除するには、 **no ip pim bsr-border** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IP マルチキャスト境界の定義

自動 RP メッセージが PIM ドメインに入らないようにする場合は、マルチキャスト境界を定義します。自動 RP 情報を伝達する 224.0.1.39 および 224.0.1.40 宛てのパケットを拒否するアクセス リストを作成します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **deny** *source* [*source-wildcard*]
4. **interface** *interface-id*
5. **ip multicast boundary** *access-list-number*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**access-list** *access-list-number* **deny** *source* [*source-wildcard*] 例： スイッチ(config)# **access-list 12 deny 224.0.1.39** **access-list 12 deny 224.0.1.40**	標準アクセス リストを作成し、コマンドを必要な回数だけ実行します。 • *access-list-number* の範囲は 1 ～ 99 です。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。 • *source* には、自動 RP 情報を伝達するマルチキャスト アドレス 224.0.1.39 および 224.0.1.40 を入力します。 • （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進

	コマンドまたはアクション	目的
		表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在します。
ステップ 4	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**ip multicast boundary** *access-list-number* 例： `スイッチ(config-if)# ip multicast boundary 12`	ステップ 2 で作成したアクセス リストを指定し、境界を設定します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

候補 BSR の設定

候補 BSR を、1 つまたは複数設定できます。候補 BSR として機能するデバイスは、他のデバイスと正しく接続され、ネットワークのバックボーン部分に配置されている必要があります。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**

3. **ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*] 例： スイッチ(config)# **ip pim bsr-candidate gigabitethernet 1/0/3 28 100**	候補 BSR となるように device を設定します。 • *interface-id* には、スイッチを候補 BSR に設定するときに BSR アドレスの取得元となる device 上のインターフェイスを入力します。このインターフェイスは PIM を使用してイネーブルにする必要があります。有効なインターフェイスは、物理ポート、ポート チャネル、VLAN などです。 • *hash-mask-length* には、ハッシュ機能を呼び出す前にグループ アドレスとの AND 条件となるマスク長（最大 32 ビット）を指定します。ハッシュ元が同じであるすべてのグループは、同じ RP に対応します。たとえば、マスク長が 24 の場合、グループ アドレスの最初の 24 ビットだけが使用されます。 • （任意）*priority* を指定する場合は、0 ～ 255 の番号を入力します。プライオリティが大きな BSR が優先されます。このプライオリティ値が同じである場合は、大きな IP アドレスを持つデバイスが BSR として選択されます。デフォルトは 0 です。
ステップ 4	**end** 例：	特権 EXEC モードに戻ります。

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

候補 RP の設定

候補 RP を、1 つまたは複数設定できます。BSR と同様、RP は他のデバイスと正しく接続され、ネットワークのバックボーン部分に配置されている必要があります。RP は IP マルチキャスト アドレス空間全体、またはその一部を処理します。候補 RP は候補 RP アドバタイズを BSR に送信します。

この手順は任意です。

始める前に

RP となるデバイスを決定するときは、次の可能性を考慮してください。

- 自動 RP だけが使用されているシスコのルータおよびマルチレイヤ devices で構成されるネットワークでは、すべてのデバイスを RP として設定できます。
- シスコの PIMv2 ルータおよびマルチレイヤ devices と、他のベンダーのルータだけで構成されるネットワークでは、すべてのデバイスを RP として使用できます。
- シスコの PIMv1 ルータ、PIMv2 ルータ、および他のベンダーのルータで構成されるネットワークでは、シスコ PIMv2 ルータおよびマルチレイヤ devices を RP として設定できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*] 例： スイッチ(config)# **ip pim rp-candidate gigabitethernet 1/0/5 group-list 10**	候補 RP となるように device を設定します。 • *interface-id* には、対応する IP アドレスが候補 RP アドレスとしてアドバタイズされるインターフェイスを指定します。有効なインターフェイスは、物理ポート、ポート チャネル、VLAN などです。 • （任意）**group-list** *access-list-number* を指定する場合は、1 ～ 99 の IP 標準アクセス リスト番号を入力します。group-list を指定しない場合は、deviceがすべてのグループの候補 RP となります。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： スイッチ(config)# **access-list 10 permit 239.0.0.0 0.255.255.255**	標準アクセス リストを作成し、コマンドを必要な回数だけ実行します。 • *access-list-number* には、ステップ 2 で指定したアクセス リスト番号を入力します。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *source* には、パケットの送信元であるネットワークまたはホストの番号を入力します。 • （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

PIM 最短パス ツリーの使用の延期

マルチキャストルーティングが送信元ツリーから最短パスツリーに切り替わる前に到達する必要があるトラフィック レートしきい値を設定するには、次の手順を実行します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
4. **ip pim spt-threshold** {*kbps* | **infinity**} [**group-list** *access-list-number*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： `スイッチ(config)# access-list 16 permit 225.0.0.0 0.255.255.255`	標準アクセス リストを作成します。 • *access-list-number* の範囲は 1 ～ 99 です。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。 • **permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *source* には、しきい値が適用されるマルチキャスト グループを指定します。 •（任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在します。
ステップ 4	**ip pim spt-threshold** {*kbps* | **infinity**} [**group-list** *access-list-number*] 例： `スイッチ(config)# ip pim spt-threshold infinity group-list 16`	最短パスツリー（SPT）に移行するまでに到達する必要があるしきい値を指定します。 • *kbps* を指定する場合は、トラフィック レートをキロビット/秒で指定します。デフォルト値は 0 キロビット/秒です。 （注） 有効範囲は 0 ～ 4294967 ですが、deviceハードウェアの制限により、0 キロビット/秒以外は無効です。 • **infinity** を指定すると、指定されたグループのすべての送信元で共有ツリーが使用され、送信元ツリーに切り替わらなくなります。 •（任意）**group-list** *access-list-number* には、ステップ 2 で作成したアクセスリストを指定します。値 0 を指定する場合、またはグループ リストを使用しない場合、しきい値はすべてのグループに適用されます。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

PIM ルータクエリー メッセージ間隔の変更

PIM ルータおよびマルチレイヤdevicesでは、各 LAN セグメント（サブネット）の指定ルータ（DR）になるデバイスを検出するため、PIM ルータクエリメッセージが送信されます。DR は、直接接続された LAN 上のすべてのホストに IGMP ホストクエリー メッセージを送信します。

PIM DM 動作では、IGMPv1 が使用中の場合だけ、DR は意味を持ちます。IGMPv1 には IGMP クエリア選択プロセスがないため、選択された DR は IGMP クエリアとして機能します。PIM-SM 動作では、マルチキャスト送信元に直接接続されたデバイスが DR になります。DR は PIM 登録メッセージを送信し、送信元からのマルチキャスト トラフィックを共有ツリーの下方向へ転送する必要があることを RP に通知します。この場合、DR は最大の IP アドレスを持つデバイスです。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip pim query-interval** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip pim query-interval** *seconds* 例： スイッチ(config-if)# **ip pim query-interval 45**	deviceが PIM ルータクエリメッセージを送信する頻度を設定します。 デフォルトは 30 秒です。指定できる範囲は 1 ～ 65535 です。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ip igmp interface** [*interface-id*] 例： スイッチ# **show ip igmp interface**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

PIM の動作の確認

PIM-SM ネットワークまたは PIM-SSM ネットワークでの IP マルチキャスト動作の確認

PIM-SM ネットワークまたは PIM-SSM ネットワークでの IP マルチキャスト動作を確認するには、次の作業を実行します。これらの作業は、ソースとレシーバが想定どおりに動作しない場合に障害のあるホップを検出するのに役立ちます。

（注） パケットが想定された宛先に到達しない場合は、IP マルチキャストのファスト スイッチングをディセーブルにすることを検討してください。ディセーブルにすると、ルータがプロセス スイッチング モードになります。IP マルチキャストのファスト スイッチングをディセーブルにした後、パケットが正しい宛先に到達するようになった場合、問題は IP マルチキャストのファスト スイッチングに関連している可能性があります。

ファースト ホップ ルータでの IP マルチキャストの確認

ファースト ホップ ルータでの IP マルチキャスト動作を確認するには、ファースト ホップ ルータに次のコマンドを入力します。

手順の概要

1. **enable**
2. **show ip mroute** [*group-address*]
3. **show ip mroute active** [*kb/s*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show ip mroute** [*group-address*] 例： スイッチ# **show ip mroute 239.1.2.3** (*, 239.1.2.3), 00:18:10/stopped, RP 172.16.0.1, flags: SPF Incoming interface: Serial1/0, RPF nbr 172.31.200.2 Outgoing interface list: Null (10.0.0.1, 239.1.2.3), 00:18:10/00:03:22, flags: FT Incoming interface: GigabitEthernet0/0/0, RPF nbr 0.0.0.0	ファースト ホップ ルータの mroute に F フラグが設定されていることを確認します。

	コマンドまたはアクション	目的
	Outgoing interface list: Serial1/0, Forward/Sparse-Dense, 00:18:10/00:03:19	
ステップ 3	**show ip mroute active** [*kb/s*] 例： スイッチ# **show ip mroute active** Active IP Multicast Sources - sending >= 4 kbps Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 30 secs), 4 kbps(life avg)	グループに送信しているアクティブなマルチキャスト送信元に関する情報を表示します。このコマンドの出力では、アクティブなソースのマルチキャスト パケット レートに関する情報が示されます。 （注） デフォルトでは、**show ip mroute** コマンドと **active** キーワードによる出力では、4 kb/s 以上のレートでグループにトラフィックを送信するアクティブなソースの情報が表示されます。より低いレートのトラフィック（4 kb/s 未満のトラフィック）をグループに送信しているアクティブなソースに関する情報を表示する場合は、*kb/s* 引数に 1 の値を指定します。この引数に 1 の値を指定すると、1 kb/s 以上のレートでグループにトラフィックを送信しているアクティブなソースに関する情報が表示されます。これによって、存在する可能性があるすべてのアクティブなソース トラフィックに関する情報が効果的に表示されます。

SPT 上のルータでの IP マルチキャストの確認

PIM-SM または PIM-SSM ネットワーク内の SPT 上のルータでの IP マルチキャスト動作を確認するには、SPT 上のルータに次のコマンドを入力します。

手順の概要

1. **enable**
2. **show ip mroute** [*group-address*]
3. **show ip mroute active**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**show ip mroute** [*group-address*] 例： スイッチ# **show ip mroute 239.1.2.3** (*, 239.1.2.3), 00:17:56/00:03:02, RP 172.16.0.1, flags: S Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: GigabitEthernet0/0/0, Forward/Sparse-Dense, 00:17:56/00:03:02 (10.0.0.1, 239.1.2.3), 00:15:34/00:03:28, flags: T Incoming interface: Serial1/0, RPF nbr 172.31.200.1 Outgoing interface list: GigabitEthernet0/0/0, Forward/Sparse-Dense, 00:15:34/00:03:02	特定のグループの送信元に対する RPF ネイバーを確認します。
ステップ 3	**show ip mroute active** 例： スイッチ# **show ip mroute active** Active IP Multicast Sources - sending >= 4 kbps Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 30 secs), 4 kbps(life avg)	グループに送信しているアクティブなマルチキャスト送信元に関する情報を表示します。このコマンドの出力では、アクティブなソースのマルチキャスト パケット レートに関する情報が示されます。 （注） デフォルトでは、**show ip mroute** コマンドと **active** キーワードによる出力では、4 kb/s 以上のレートでグループにトラフィックを送信するアクティブなソースの情報が表示されます。より低いレートのトラフィック（4 kb/s 未満のトラフィック）をグループに送信しているアクティブなソースに関する情報を表示する場合は、*kb/s* 引数に 1 の値を指定します。この引数に 1 の値を指定すると、1 kb/s 以上のレートでグループにトラフィックを送信しているアクティブなソースに関する情報が表示されます。これによって、存在する可能性があるすべてのアクティブなソース トラフィックに関する情報が効果的に表示されます。

ラスト ホップ ルータでの IP マルチキャスト動作の確認

ラスト ホップ ルータでの IP マルチキャスト動作を確認するには、ラスト ホップ ルータで次のコマンドを入力します。

手順の概要

1. **enable**
2. **show ip igmp groups**

3. **show ip pim rp mapping**
4. **show ip mroute**
5. **show ip interface** [*type number*]
6. **show ip pim interface count**
7. **show ip mroute count**
8. **show ip mroute active** [*kb/s*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show ip igmp groups** 例： スイッチ# **show ip igmp groups** IGMP Connected Group Membership Group Address Interface Uptime Expires Last Reporter 239.1.2.3 GigabitEthernet1/0/0 00:05:14 00:02:14 10.1.0.6 224.0.1.39 GigabitEthernet0/0/0 00:09:11 00:02:08 172.31.100.1	ラスト ホップ ルータの IGMP メンバーシップを確認します。この情報によって、ラスト ホップ ルータに直接接続され、IGMP を介して認識されるレシーバが使用されているマルチキャスト グループが確認されます。
ステップ 3	**show ip pim rp mapping** 例： スイッチ# **show ip pim rp mapping** PIM Group-to-RP Mappings Group(s) 224.0.0.0/4 RP 172.16.0.1 (?), v2v1 Info source: 172.16.0.1 (?), elected via Auto-RP Uptime: 00:09:11, expires: 00:02:47	グループと RP 間のマッピングがラスト ホップ ルータで正しく生成されていることを確認します。 （注） PIM/SSM ネットワークでラスト ホップ ルータを確認する場合は、この手順を無視してください。PIM-SSM ではランデブー ポイント（RP）が使用されないため、**show ip pim rp mapping** コマンドは PIM/SSM ネットワーク内のルータでは動作しません。さらに、正しく設定されている場合は、PIM/SSM グループは **show ip pim rp mapping** コマンドの出力には表示されません。
ステップ 4	**show ip mroute** 例： スイッチ# **show ip mroute** (*, 239.1.2.3), 00:05:14/00:03:04, RP 172.16.0.1, flags: SJC Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1 Outgoing interface list:	mroute テーブルがラスト ホップ ルータに正しく入力されていることを確認します。

	コマンドまたはアクション	目的
	GigabitEthernet1/0, Forward/Sparse-Dense, 00:05:10/00:03:04 (10.0.0.1, 239.1.2.3), 00:02:49/00:03:29, flags: T Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse-Dense, 00:02:49/00:03:04 (*, 224.0.1.39), 00:10:05/stopped, RP 0.0.0.0, flags: DC Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse-Dense, 00:05:15/00:00:00 GigabitEthernet0/0, Forward/Sparse-Dense, 00:10:05/00:00:00 (172.16.0.1, 224.0.1.39), 00:02:00/00:01:33, flags: PTX Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1	
ステップ 5	**show ip interface** [*type number*] **例**： スイッチ# **show ip interface GigabitEthernet 0/0/0** GigabitEthernet0/0 is up, line protocol is up Internet address is 172.31.100.2/24 Broadcast address is 255.255.255.255 Address determined by setup command MTU is 1500 bytes Helper address is not set Directed broadcast forwarding is disabled Multicast reserved groups joined: 224.0.0.1 224.0.0.22 224.0.0.13 224.0.0.5 224.0.0.6 Outgoing access list is not set Inbound access list is not set Proxy ARP is enabled Local Proxy ARP is disabled Security level is default Split horizon is enabled ICMP redirects are always sent ICMP unreachables are always sent ICMP mask replies are never sent IP fast switching is enabled IP fast switching on the same interface is disabled IP Flow switching is disabled IP CEF switching is disabled IP Fast switching turbo vector IP multicast fast switching is enabled IP multicast distributed fast switching is disabled IP route-cache flags are Fast Router Discovery is disabled IP output packet accounting is disabled IP access violation accounting is disabled	マルチキャスト高速スイッチングがイネーブルになっており、ラスト ホップ ルータの発信インターフェイスでのパフォーマンスが最適化されていることを確認します。 （注） **no ip mroute-cache** インターフェイスコマンドを使用すると、IPマルチキャスト高速スイッチングがディセーブルになります。IPマルチキャスト高速スイッチングがディセーブルになると、プロセス スイッチドパスを介してパケットが転送されます。

	コマンドまたはアクション	目的
	TCP/IP header compression is disabled RTP/IP header compression is disabled Policy routing is disabled Network address translation is disabled WCCP Redirect outbound is disabled WCCP Redirect inbound is disabled WCCP Redirect exclude is disabled BGP Policy Mapping is disabled	
ステップ 6	**show ip pim interface count** 例： スイッチ# **show ip pim interface count** State: * - Fast Switched, D - Distributed Fast Switched H - Hardware Switching Enabled Address Interface FS Mpackets In/Out 172.31.100.2 GigabitEthernet0/0/0 * 4122/0 10.1.0.1 GigabitEthernet1/0/0 * 0/3193	マルチキャスト トラフィックがラスト ホップ ルータに転送されることを確認します。
ステップ 7	**show ip mroute count** 例： スイッチ# **show ip mroute count** IP Multicast Statistics 6 routes using 4008 bytes of memory 3 groups, 1.00 average sources per group Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc) Group: 239.1.2.3, Source count: 1, Packets forwarded: 3165, Packets received: 3165 RP-tree: Forwarding: 0/0/0/0, Other: 0/0/0 Source: 10.0.0.1/32, Forwarding: 3165/20/28/4, Other: 0/0/0 Group: 224.0.1.39, Source count: 1, Packets forwarded: 21, Packets received: 120 Source: 172.16.0.1/32, Forwarding: 21/1/48/0, Other: 120/0/99 Group: 224.0.1.40, Source count: 1, Packets forwarded: 10, Packets received: 10 Source: 172.16.0.1/32, Forwarding: 10/1/48/0, Other: 10/0/0	マルチキャスト トラフィックがラスト ホップ ルータに転送されることを確認します。
ステップ 8	**show ip mroute active** [*kb/s*] 例： スイッチ# **show ip mroute active** Active IP Multicast Sources - sending >= 4 kbps Group: 239.1.2.3, (?)	ラスト ホップ ルータ上のグループにトラフィックを送信しているアクティブなマルチキャスト ソースに関する情報を表示します。このコマンドの出力では、アクティブなソースのマルチキャスト パケット レートに関する情報が示されます。

	コマンドまたはアクション	目的
	`Source: 10.0.0.1 (?)` `Rate: 20 pps/4 kbps(1sec), 4 kbps(last 50 secs), 4 kbps(life avg)`	（注） デフォルトでは、**show ip mroute** コマンドと **active** キーワードによる出力では、4 kb/s 以上のレートでグループにトラフィックを送信するアクティブなソースの情報が表示されます。より低いレートのトラフィック（4 kb/s 未満のトラフィック）をグループに送信しているアクティブなソースに関する情報を表示する場合は、*kb/s* 引数に 1 の値を指定します。この引数に 1 の値を指定すると、1 kb/s 以上のレートでグループにトラフィックを送信しているアクティブなソースに関する情報が表示されます。これによって、存在する可能性があるすべてのアクティブなソース トラフィックに関する情報が効果的に表示されます。

PIM 対応ルータを使用した IP マルチキャストの到達可能性のテスト

管理しているすべての PIM 対応ルータおよびアクセス サーバーが、マルチキャスト グループのメンバで、すべてのルータが応答する原因となる ping が送信されます。これは、効果的な管理およびデバッグのツールです。

PIM 対応ルータを使用して IP マルチキャストの到達可能性をテストするには、次の作業を実行します。

マルチキャスト ping に応答するルータの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip igmp join-group** *group-address*
5. マルチキャスト ネットワークに加入しているルータ上のインターフェイスで、ステップ 3 とステップ 4 を繰り返します。
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： スイッチ(config)# **interface gigabitethernet 1/0/0**	インターフェイス コンフィギュレーション モードを開始します。 *type* 引数および *number* 引数には、ホストに直接接続されているインターフェイス、またはホストに対応しているインターフェイスを指定します。
ステップ 4	**ip igmp join-group** *group-address* 例： スイッチ(config-if)# **ip igmp join-group 225.2.2.2**	（任意）指定したグループに加入するようにルータ上のインターフェイスを設定します。 この作業の目的として、マルチキャスト ネットワークに加入しているルータ上のすべてのインターフェイス上で、*group-address* 引数に同じグループ アドレスを設定します。 （注） この方法では、ルータは、マルチキャスト パケットの転送に加えて、マルチキャスト パケットを受信します。マルチキャスト パケットを受信することにより、ルータの高速スイッチングは行われません。
ステップ 5	マルチキャスト ネットワークに加入しているルータ上のインターフェイスで、ステップ 3 とステップ 4 を繰り返します。	--
ステップ 6	**end** 例： スイッチ(config-if)# **end**	現在のコンフィギュレーション セッションを終了して、特権 EXEC モードに戻ります。

マルチキャスト ping に応答するように設定されたルータへの ping

マルチキャスト ping に応答するように設定されているルータに対して ping テストを開始するには、ルータで。このタスクは、ネットワーク内の IP マルチキャストの到達可能性のテストに使用します。

手順の概要

1. **enable**
2. **ping** *group-address*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**ping** *group-address* 例： スイッチ# **ping 225.2.2.2**	IP マルチキャスト グループ アドレスを ping します。 正常な応答は、グループ アドレスが機能していることを示します。

PIM のモニタリングとトラブルシューティング

PIM 情報のモニタリング

PIM 設定をモニターするには、次の表に記載された特権 EXEC コマンドを使用します。

表 *28 : PIM* モニタリング コマンド

コマンド	目的
show ip pim interface	Protocol Independent Multicast（PIM）のために設定されているインターフェイスに関する情報を表示します。
show ip pim neighbor	PIM ネイバー情報を表示します。
show ip pim rp[*group-name* | *group-address*]	スパース モードのマルチキャスト グループに関連付けられた RP ルータを表示します。このコマンドは、すべてのソフトウェア イメージで使用できます。

RP マッピングおよび BSR 情報のモニタリング

次の表に示す特権 EXEC モードを使用して、グループ/RP マッピングの一貫性を確認します。

表 *29 : RP* マッピングのモニタリング コマンド

コマンド	目的
show ip pim rp-hash *group*	指定したグループに選択されている RP を表示します。つまり、PIMv2 ルータまたはマルチレイヤ device 上で、PIMv1 システムで選択されている RP と同じ RP が使用されていることを確認します。*group* には、RP 情報を表示するグループ アドレスを入力します。

BSR の情報をモニターするには、次の表に示す特権 EXEC コマンドを使用します。

表 *30 : VTP* モニタリング コマンド

コマンド	目的
show ip pim bsr	選択された BSR に関する情報を表示します。

PIMv1 および PIMv2 の相互運用性に関するトラブルシューティング

PIMv1 および PIMv2 間の相互運用性に関する問題をデバッグするには、次の点を順にチェックします。

1. **show ip pim rp-hash** 特権 EXEC コマンドを使用して RP マッピングを確認し、すべてのシステムが同じグループの同じ RP に同意していることを確認します。
2. DR と RP の各バージョン間の相互運用性を確認し、RP が DR と適切に相互作用していることを確認します（この場合は、登録停止に応答し、カプセル化が解除されたデータ パケットをレジスタから転送します）。

PIM の設定例

例：PIM スタブ ルーティングのイネーブル化

次の例では、IP マルチキャスト ルーティングがイネーブルになっており、スイッチ A の PIM アップリンク ポート 25 はルーテッド アップリンク ポートとして設定されています（**spare-dense-mode** がイネーブル）。VLAN 100 インターフェイスとギガビット イーサネット ポート 20 で PIM スタブ ルーティングがイネーブルに設定されています。

```
スイッチ(config)# ip multicast-routing distributed
スイッチ(config)# interface GigabitEthernet3/0/25
```

```
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 3.1.1.2 255.255.255.0
スイッチ(config-if)# ip pim sparse-dense-mode
スイッチ(config-if)# exit
スイッチ(config)# interface vlan100
スイッチ(config-if)# ip pim passive
スイッチ(config-if)# exit
スイッチ(config)# interface GigabitEthernet3/0/20
スイッチ(config-if)# ip pim passive
スイッチ(config-if)# exit
スイッチ(config)# interface vlan100
スイッチ(config-if)# ip address 100.1.1.1 255.255.255.0
スイッチ(config-if)# ip pim passive
スイッチ(config-if)# exit
スイッチ(config)# interface GigabitEthernet3/0/20
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 10.1.1.1 255.255.255.0
スイッチ(config-if)# ip pim passive
スイッチ(config-if)# end
```

例：PIM スタブ ルーティングの確認

各インターフェイスの PIM スタブがイネーブルになっていることを確認するには、**show ip pim interfac** 特権 EXEC コマンドを使用します。

```
スイッチ# show ip pim interface
Address Interface Ver/ Nbr Query DR DR
Mode Count Intvl Prior
3.1.1.2 GigabitEthernet3/0/25 v2/SD 1 30 1 3.1.1.2
100.1.1.1 Vlan100 v2/P 0 30 1 100.1.1.1
10.1.1.1 GigabitEthernet3/0/20 v2/P 0 30 1 10.1.1.1
```

例：マルチキャスト グループへの RP の手動割り当て

次に、マルチキャスト グループ 225.2.2.2 の場合だけ、RP のアドレスを 147.106.6.22 に設定する例を示します。

```
スイッチ(config)# access-list 1 permit 225.2.2.2 0.0.0.0
スイッチ(config)# ip pim rp-address 147.106.6.22 1
```

例：Auto-RP の設定

次に、最大ホップ数が 31 であるすべての PIM 対応インターフェイスから RP アナウンスメントを送信する例を示します。ポート 1 の IP アドレスが RP です。アクセス リスト 5 には、この deviceが RP として機能するグループが記述されています。

```
スイッチ(config)# ip pim send-rp-announce gigabitethernet1/0/1 scope 31 group-list 5
スイッチ(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

例：Auto-RP 情報を拒否する IP マルチキャスト境界の定義

次に、自動 RP 情報を拒否する IP マルチキャスト境界のコンフィギュレーション例の一部を示します。

```
スイッチ(config)# access-list 1 deny 224.0.1.39
スイッチ(config)# access-list 1 deny 224.0.1.40
スイッチ(config)# access-list 1 permit all
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# ip multicast boundary 1
```

例：着信 RP アナウンスメント メッセージのフィルタリング

次に、候補 RP アナウンスメントが不正な候補 RP から許可されないようにするために使用される自動 RP マッピング エージェントの設定例を示します。

```
スイッチ(config)# ip pim rp-announce-filter rp-list 10 group-list 20
スイッチ(config)# access-list 10 permit host 172.16.5.1
スイッチ(config)# access-list 10 permit host 172.16.2.1
スイッチ(config)# access-list 20 deny 239.0.0.0 0.0.255.255
スイッチ(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

マッピング エージェントは 2 つのデバイス（172.16.5.1 および 172.16.2.1）からの候補 RP アナウンスだけを許可します。マッピング エージェントは 2 つのデバイスからの候補 RP アナウンスメントのうち、グループ範囲が 224.0.0.0 ～ 239.255.255.255 であるマルチキャスト グループ宛てのアナウンスメントだけを許可します。マッピングエージェントは、ネットワーク内の他のデバイスからの候補 RP アナウンスメントを許可しません。さらに、候補 RP アナウンスメントが 239.0.0.0 ～ 239.255.255.255 の範囲のグループに宛てたものである場合、マッピング エージェントは 172.16.5.1 または 172.16.2.1 からの候補 RP アナウンスメントを許可しません。この範囲は、管理の有効範囲付きアドレス範囲です。

例：問題のある RP への Join メッセージの送信禁止

すべてのインターフェイスが SM の場合はデフォルト設定の RP を使用し、既知のグループ 224.0.1.39 および 224.0.1.40 をサポートします。自動 RP はこれら 2 つの既知のグループを使用し、RP マッピング情報を収集、配信します。**ip pim accept-rp auto-rp** コマンドが設定されている場合は、RP を許可する別の **ip pim accept-rp** コマンドを次のように設定してください。

```
スイッチ(config)# ip pim accept-rp 172.10.20.1 1
スイッチ(config)# access-list 1 permit 224.0.1.39
```

```
スイッチ(config)# access-list 1 permit 224.0.1.40
```

例：候補 BSR の設定

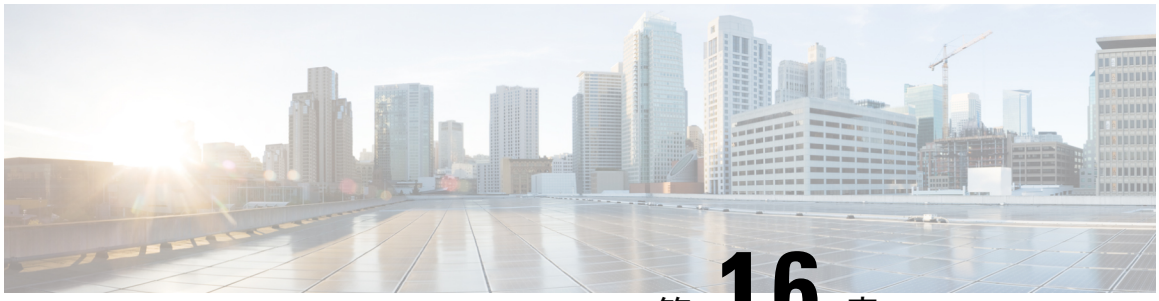
次に、候補 BSR の設定例を示します。この例では、アドバタイズ済み BSR アドレスとしてポートの IP アドレス 172.21.24.18 を、hash-mask-length として 30 ビットを使用します。プライオリティは 10 です。

```
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# ip address 172.21.24.18 255.255.255.0
スイッチ(config-if)# ip pim sparse-mode
スイッチ(config-if)# ip pim bsr-candidate gigabitethernet1/0/2 30 10
```

例：候補 RP の設定

次に、device が自身を候補 RP として PIM ドメイン内の BSR にアドバタイズするよう設定する例を示します。標準アクセス リスト番号 4 により、ポートで識別されるアドレスを持つ RP に対応するグループ プレフィックスが指定されます。この RP は、プレフィックスが 239 であるグループを処理します。

```
スイッチ(config)# ip pim rp-candidate gigabitethernet1/0/2 group-list 4
スイッチ(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```



第 16 章

HSRP 認識 PIM の設定

HSRP 認識 PIM

このモジュールでは、ホットスタンバイ ルータ プロトコル（HSRP）のアクティブ ルータ（AR）経由で転送するマルチキャスト トラフィックをイネーブルにし、PIM（Protocol Independent Multicast）を許可して HSRP の冗長性を活用し、潜在的なトラフィックの重複を回避し、フェールオーバーをイネーブルにできるように HSRP 認識 PIM 機能を設定する方法について説明します。

HSRP 認識 PIM の制約事項

- HSRP IPv6 はサポートされません。
- ステートフル フェールオーバーはサポートされません。PIM ステートレス フェールオーバー時は、HSRP グループの仮想 IP アドレスがスタンバイ ルータに転送されますが、mrouting ステート情報は転送されません。PIM はステート変更イベントをリッスンして応答し、フェールオーバー時に mroute ステートを作成します。
- 各インターフェイスの PIM がトラッキングできる HSRP グループの最大数は 16 です。
- PIM DR の冗長性プライオリティは、同じ RSRP グループがイネーブルになるか、または HSRP アクティブが DR の選択で成功しないデバイスの PIM DR プライオリティの設定値またはデフォルト値（1）よりも大きくする必要があります。
- デンス モードはサポートされません。
- PIM RP としての HSRP アドレスはサポートされていません。HSRP 認識 PIM は、PIM DR 選択と HSRP プライマリ選択を調整するためのものです。

HSRP 認識 PIM に関する情報

HSRP

Hot Standby Router Protocol（HSRP）はフォールトトレラント デフォルト ゲートウェイを確立するためのシスコ独自の冗長プロトコルです。

このプロトコルは、プライマリ ゲートウェイがアクセスできなくなった場合にデフォルト ゲートウェイのフェールオーバーを実現できるようにネットワーク デバイス間にフレームワークを確立します。複数のデバイスは、IP アドレスと MAC（レイヤ 2）アドレスを共有することで単一の仮想ルータとして機能できます。仮想ルータ グループのメンバは常にステータス メッセージを交換し、あるデバイスが予定されたまたは予定外の理由によって稼働しなくなった場合に、別のデバイスがルーティング処理を請け負うことができます。ホストは、一貫した IP および MAC アドレスに IP パケットを送信しつづけ、ルーティングを実行するデバイスは透過的に切り替えられます。

HSRP は、ホストが Router Discovery Protocol をサポートしておらず、選択されたデバイスのリロードや電源故障時に新しいデバイスに切り替えることができない場合に有効です。また、既存の TCP セッションはフェールオーバーが発生しても存続するため、このプロトコルでは IP トラフィックをルーティングするためにネクストホップを動的に選択するホストの回復をさらに透過的に実行できます。

HSRP をネットワーク セグメントに設定すると、HSRP が動作するデバイスのグループ間で仮想 MAC アドレスと IP アドレスを共有できるようになります。この HSRP グループのアドレスが仮想 IP アドレスと呼ばれます。このようなデバイスの 1 つが、アクティブ ルータ（AR）としてプロトコルによって選択されます。AR は、グループの MAC アドレス宛のパケットを受信してルーティングします。

HSRP では、プライオリティ メカニズムを使用して、デフォルトの AR にする HSRP 設定済みデバイスを決定します。デバイスを AR として設定するには、他のすべての HSRP 設定済みデバイスのプライオリティよりも高いプライオリティをそのデバイスに割り当てます。デフォルトのプライオリティは 100 です。したがって、100 よりも高いプライオリティを持つデバイスを 1 つだけ設定した場合、そのデバイスがデフォルトの AR になります。

HSRP を実行しているデバイスは、User Datagram Protocol（UDP）ベースのマルチキャスト hello メッセージを送信および受信して、デバイスの障害を検出したり、アクティブ デバイスとスタンバイ デバイスを割り当てたりします。AR が設定された時間内に hello メッセージを送信できなかった場合は、最高のプライオリティのスタンバイ デバイスが AR になります。このようにパケット転送機能が別のデバイスに移行しても、ネットワークのいずれのホストにもまったく影響はありません。

複数のホット スタンバイ グループをインターフェイスに設定できるので、冗長デバイスおよびロード シェアリングを余すところなく活用できるようになっています。

HSRP は IP ルートをアドバタイズせず、また、ルーティング テーブルに影響しないため、ルーティング プロトコルではありません。

HSRP には、デバイスの 1 つ以上のインターフェイスに障害が発生した場合にフェールオーバーをトリガーする機能があります。これは、ヘッドエンドに戻す 1 つのシリアル リンクをそれぞ

れ持つデュアル ブランチ デバイスに役立つ場合があります。プライマリ デバイスのシリアル リンクがダウンした場合、バックアップデバイスがプライマリ機能を引き継ぎ、ヘッドエンドへの接続を保持します。

HSRP 認識 PIM

PIM（Protocol Independent Multicast）には固有の冗長性機能がなく、その動作は Hot Standby Router Protocol（HSRP）グループ ステートに依存しません。その結果、IP マルチキャスト トラフィックは、HSRP によって選択されたものと同じデバイスによって必ずしも転送されるとは限りません。HSRP 認識 PIM 機能は、イネーブルになっている仮想ルーティング グループの冗長ネットワークで一貫した IP マルチキャスト転送を実現します。

HSRP 認識 PIM は HSRPアクティブ ルータ（AR）経由でのマルチキャスト トラフィックを転送することができるため、デバイスの HSRP ステートによっては、PIM は HSRP 冗長性を活用し、潜在的なトラフィックの重複を回避し、フェールオーバーをイネーブルにすることができます。PIM 代表ルータ（DR）は HSRP AR と同じゲートウェイで実行し、mroute ステートを維持します。

マルチアクセス セグメントで（LAN など）では、PIM DR 選択は冗長構成に対応していないため、選択した DR および HSRP AR が同じルータでない場合があります。PIM DR が RP または FHR に常に PIM Join/Prune メッセージを送信するようにするために、（HSRP グループが 1 つだけの場合は）HSRP AR が PIM DR になります。PIM はグループ ステートに基づく DR プライオリティの調整を担います。フェールオーバーが発生すると、HSRP グループによって選択された新しい AR 上にマルチキャスト ステートが作成され、その AR が HSRP 仮想 IP アドレスにアドレス指定されたすべてのトラフィックをルーティングし、転送する役割を引き受けます。

HSRP 認識 PIM をイネーブルにすると、PIM はデバイスが HSRP Active になった時点で PIM Hello 追加メッセージを各アクティブ HSRP グループの送信元アドレスとして HSRP 仮想 アドレスを使用して送信します。PIM Hello は、フェールオーバーに対応するための他のルータをトリガーするため、新しい GenID を伝送します。ダウンストリーム デバイスでこの PIM Hello を受信すると、仮想アドレスを PIM ネイバー リストに追加します。PIM Hello で伝送された新しい GenID はダウンストリームのルータをトリガーし、PIM Join メッセージを仮想アドレスに再送信します。アップストリーム ルータは、HSRP グループ ステートに基づいて PIM Join/Prunes（J/P）を処理します。

J/P の宛先が HSRP グループの仮想アドレスに一致し、宛先のデバイスが HSRP がアクティブ ステートである場合は、新しい AR が PIM DR として機能しているため、この AR が PIM Join を処理します。これにより、すべての PIM Join/Prune が HSRP グループの仮想アドレスに到達するため、ダウンストリーム ルータ側での変更とコンフィギュレーションが最小限に抑えられます。

IP ルーティング サービスが既存の仮想ルーティング プロトコルを使用して、基本的なステートレス フェールオーバー サービスを PIM などのクライアントアプリケーションに提供します。ローカルの HSRP グループ ステートの変更とスタンバイ ルータが担うタスクは対象のクライアント アプリケーションに通知されます。クライアント アプリケーションが IRS の最上部に構築され、ステートフルまたはステートレスのフェールオーバーを構築することがあります。HSRP クライアントとして PIM は HSRP からのステート変更通知をリッスンし、HSRP ス

テートに基づいて PIM DR のプライオリティを自動的に調整します。PIM クライアントも、新しい AR に mroute ステートを作成するためにフェールオーバーの時点でアップストリーム デバイスとダウンストリーム デバイス間の通信をトリガーします。

HSRP 認識 PIM の設定方法

インターフェイスでの HSRP グループの設定

始める前に

- デバイス上に IP マルチキャストがすでに設定されている必要があります。
- デバイス上に PIM がすでに設定されている必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [*secondary*]]
6. **standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime*
7. **standby** [*group-number*] **priority** *priority*
8. **standby** [*group-number*] **name** *group-name*
9. **end**
10. **show standby** [*type number* [*group*]] [**all** | **brief**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* [*name-tag*] 例： `Device(config)# interface ethernet 0/0`	設定するインターフェイスを指定します。インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address mask* 例：	インターフェイスに対するプライマリ IP アドレスまたはセカンダリ IP アドレスを設定します。

	コマンドまたはアクション	目的
	`Device(config-if)# ip address 10.0.0.2 255.255.255.0`	
ステップ 5	**standby** [*group-number*] **ip** [*ip-address* [*secondary*]] 例： `Device(config-if)# standby 1 ip 192.0.2.99`	HSRP をアクティブ化して HSRP グループを定義します。
ステップ 6	**standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime* 例： `Device(config-if)# standby 1 timers 5 15`	（任意）hello パケット間隔、および HSRP のアクティブ ルータまたはスタンバイ ルータのダウンを他のデバイスが宣言するまでの時間を設定します。
ステップ 7	**standby** [*group-number*] **priority** *priority* 例： `Device(config-if)# standby 1 priority 120`	（任意）HSRP のアクティブ ルータおよびスタンバイ ルータを選択しやすいように使用する HSRP プライオリティを割り当てます。
ステップ 8	**standby** [*group-number*] **name** *group-name* 例： `Device(config-if)# standby 1 name HSRP1`	（任意）HSRP グループの名前を定義します。 （注） HSRP 認識 PIM に使用する HSRP グループを設定する際は、**standby ip name** コマンドを常に設定することを推奨します。
ステップ 9	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show standby** [*type number* [*group*]] [**all** | **brief**] 例： `Device# show standby`	設定を確認するための HSRP グループ情報が表示されます。

PIM 冗長性の設定

始める前に

HSRP グループはインターフェイス上で設定済みになっている必要があります。「インターフェイスでの HSRP グループの設定」を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip address** *ip-address mask*
5. **ip pim redundancy** *group* **dr-priority** *priority*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* [*name-tag*] 例： `Device(config)# interface ethernet 0/0`	設定するインターフェイスを指定します。インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address mask* 例： `Device(config-if)# ip address 10.0.0.2 255.255.255.0`	インターフェイスに対するプライマリ IP アドレスまたはセカンダリ IP アドレスを設定します。
ステップ 5	**ip pim redundancy** *group* **dr-priority** *priority* 例： `Device(config-if)# ip pim redundancy HSRP1 dr-priority 60`	PIM 冗長性をイネーブルにし、冗長性プライオリティ値をアクティブな PIM 指定ルータ（DR）に割り当てます。 • HSRP グループ名では大文字と小文字が区別されるため、*group* 引数の値は **standby ip name** コマンドを使用して設定したグループ名と一致している必要があります。 • PIM DR の冗長性プライオリティは、同じ HSRP グループがイネーブルになっているデバイスの PIM DR プライオリティに設定されている値またはデフォルト値（1）よりも大きくする必要があります。
ステップ 6	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。

HSRP 認識 PIM の設定例

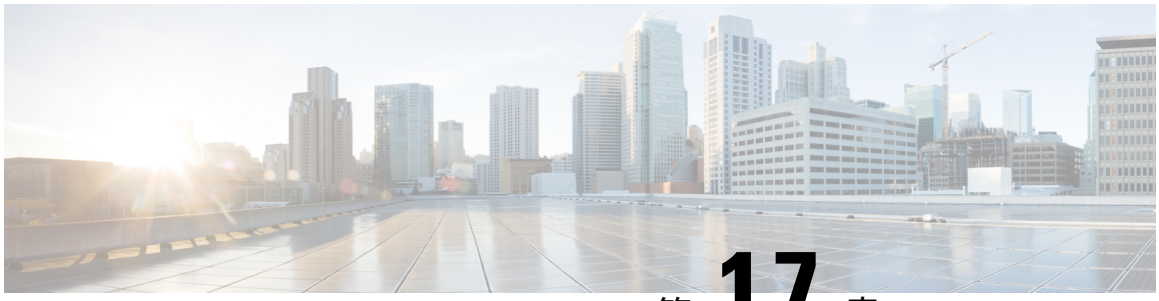
例：インターフェイスでの HSRP グループの設定

```
interface ethernet 0/0
```

```
 ip address 10.0.0.2 255.255.255.0
 standby 1 ip 192.0.2.99
 standby 1 timers 5 15
 standby 1 priority 120
 standby 1 name HSRP1
 !
!
```

例：PIM 冗長性の設定

```
interface ethernet 0/0
 ip address 10.0.0.2 255.255.255.0
 ip pim redundancy HSRP1 dr-priority 60
 !
!
```



第 17 章

VRRP 認識 PIM の設定

VRRP 認識 PIM

仮想ルータ冗長プロトコル（VRRP）によって、静的なデフォルトのルーティング環境に固有の単一障害点が除外されます。VRRPは、1つ以上の仮想ルータに対する責任をLAN上のVRRPルータに動的に割り当てて、マルチアクセス リンク上の複数のルータで同じ仮想 IP アドレスを利用できるようにする選定プロトコルです。

VRRP 認識 PIMは、VRRP と相互運用する PIM（Protocol Independent Multicast）の冗長性メカニズムです。このメカニズムでは、PIMがVRRPステートを追跡し、仮想ルーティング グループがイネーブルになっている冗長ネットワークでのフェールオーバー時にマルチキャスト トラフィックを保持できます。

ここでは、ネットワークの VRRP 認識 PIM の設定方法を説明します。

VRRP 認識 PIM の制約事項

- PIM スパース モード（SM）と Source Specific Multicast（SSM）モードがサポートされています。双方向（BiDir）PIM はサポートされません。
- Hot Standby Router Protocol（HSRP）IPv6 での PIM の相互運用性はサポートされません。
- PIMは、インターフェイスごとにVirtual Router Redundancy Protocol（VRRP）またはHSRPのいずれか1つの仮想グループのみをサポートします。
- VRRP 認識 PIMは中継ネットワークではサポートされません。PIM の冗長性対応インターフェイスは、ダウン ストリームからネットワークに参加する PIM をサポートしません。

VRRP 認識 PIM に関する情報

VRRP 認識 PIM の概要

Virtual Router Redundancy Protocol（VRRP）は、フォールトトレラント デフォルト ゲートウェイを確立するための冗長プロトコルです。このプロトコルは、プライマリ ゲートウェイがアク

セスできなくなった場合にデフォルト ゲートウェイのフェールオーバーを実現できるようにネットワーク デバイス間にフレームワークを確立します。

PIM（Protocol Independent Multicast）には固有の冗長性機能がないため、その動作は VRRP グループの状態に依存しません。したがって、IP マルチキャストのトラフィックは、VRRP によって選択されたものと同じデバイスによって転送されるとは限りません。VRRP 認識 PIM 機能は、イネーブルの状態の仮想ルーティング グループの冗長ネットワークで一貫した IP マルチキャスト転送を実行します。

マルチアクセスセグメント（LAN など）では、PIM 代表ルータ（DR）選択が冗長設定を認識しないため、選択された DR および VRRP のプライマリルータ（MR）は同じルータでない場合があります。PIM DR が常に PIM Join/Prume メッセージを RP または FHR に送信できるようにするため、VRRP MR が PIM DR になります（VRRP グループが 1 つだけの場合）。PIM はグループ ステートに基づく DR プライオリティの調整を担います。フェールオーバーが発生すると、マルチキャスト ステートが VRRP グループによって選択された新しい MR に作成され、その MR が VRRP 仮想 IP アドレスにアドレス指定されたすべてのトラフィックのルーティングと転送を担います。こうすることによって、PIM DR は VRRP MR と同じゲートウェイで実行され、mroute ステートが保持されます。これにより、マルチキャスト トラフィックが VRRP MR を通じて転送され、PIM が VRRP の冗長性を利用してトラフィックが重複する可能性をなくし、デバイスの VRRP 状態に応じてフェールオーバーを有効にします。

仮想ルータ冗長性サービス（FRRS）はクライアントにパブリック API を提供して VRRP との通信を行います。VRRP 認識 PIM は、IPv4 と IPv6 の両方で BRRPv3（ユニファイド VRRP）をサポートする VRRS の機能です。

VRRS クライアントとしての PIM は VRRS クライアント API を使用して一般的な First Hop Redundancy Protocol（FHRP）状態と設定情報を取得し、マルチキャスト冗長性機能を提供します。

PIM は、VRRS クライアントとして次の処理を実行します。

- 状態の変更をリッスンし、VRRS サーバ（VRRP）からの通知を更新します。
- VRRP の状態に基づいて PIM DR の優先度を調整します。
- VRRP がフェールオーバーすると、PIM はトラッキング対象の VRRS から状態変更通知を受け取り、VRRP MR からトラフィックが転送されるようにします。

VRRP 認識 PIM の設定方法

VRRP 認識 PIM の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **fhrp version vrrp** *version*
4. **interface** *type number*

5. **ip address** *address* {*primary* |*secondary*}
6. **vrrp** *group id* **address-family ipv4**
7. **vrrs leader** *group name*
8. **vrrp** *group id* **ip** *ip address*{*primary* |*secondary*}
9. **exit**
10. **interface** *type number*
11. **ip pim redundancy** *group name* **vrrp dr-priority** *priority-value*
12. **end**

手順の詳細

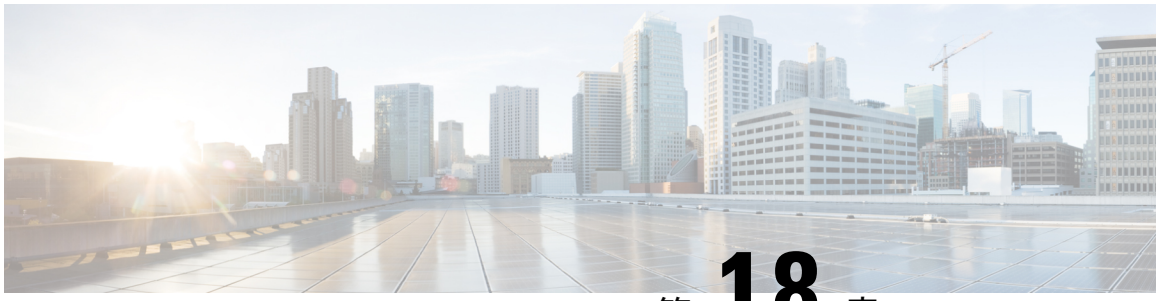
	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**fhrp version vrrp** *version* 例： `Device(config)# fhrp version vrrp v3`	VRRPv3 および VRRS を設定する機能をイネーブルにします。
ステップ 4	**interface** *type number* 例： `Device(config)# interface Ethernet0/0`	設定するインターフェイスを指定します。インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**ip address** *address* {*primary* |*secondary*} 例： `Device(config-if)# ip address 192.0.2.2`	VRRP グループのプライマリ アドレスまたはセカンダリ アドレスを指定します。
ステップ 6	**vrrp** *group id* **address-family ipv4** 例： `Device(config-if)# vrrp 1 address-family ipv4`	VRRP グループを作成し、VRRP コンフィギュレーション モードを開始します。
ステップ 7	**vrrs leader** *group name* 例： `Device(config-if-vrrp)# vrrs leader VRRP1`	指定されたネイバーとのコミュニティおよび（または）拡張コミュニティの交換をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 8	**vrrp** *group id* **ip** *ip address*{*primary* |*secondary*} 例： `Device(config-if-vrrp)# vrrp 1 ip 10.1.6.1`	アドレス ファミリ コンフィギュレーション モードを終了し、ルータ コンフィギュレーション モードに戻ります。
ステップ 9	**exit** 例： `Device(config-if-vrrp)# exit`	VRRP コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 10	**interface** *type number* 例： `Device(config)# interface Ethernet0/0`	設定するインターフェイスを指定します。インターフェイス コンフィギュレーション モードを開始します。
ステップ 11	**ip pim redundancy** *group name* **vrrp dr-priority** *priority-value* 例： `Device(config-if)# ip pim redundancy VRRP1 vrrp dr-priority 90`	ルータが指定ルータ（DR）として選択されるプライオリティを設定します。 • 冗長性の dr-priority 値は、VRRP 認識 PIM 機能でイネーブルにされたすべてのルータの値と同じにする必要があります。
ステップ 12	**end** 例： `Device(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

VRRP 認識 PIM の設定例

例：VRRP 認識 PIM

```
conf terminal
 fhrp version vrrp v3
interface Ethernet0/0
 ip  address 192.0.2.2
 vrrp 1 address-family ipv4

 vrrp 1 ip 10.1.6.1

 vrrs leader VRRP1
interface Ethernet0/0
 ip pim redundancy VRRP1 vrrp  dr-priority 90
 !
```



第 18 章

SSM の設定

SSM の設定の前提条件

次に、Source-Specific Multicast （SSM）および SSM マッピングを設定するための前提条件を示します。

- SSM および SSM マッピングを使用するには、3560-CX スイッチの IP Services フィーチャセットをイネーブルにする必要があります。
- SSM マッピングを設定する前に、次の作業を実行する必要があります。
  - IP マルチキャスト ルーティングをイネーブルにします。
  - PIM スパース モードをイネーブルにします。
  - SSM を設定します。
- スタティック SSM マッピングを設定する場合は、事前にアクセス コントロール リスト（ACL）を設定して、送信元アドレスにマッピングされるグループ範囲を定義する必要があります。
- SSM マッピングを設定し、DNS ルックアップで使用できるようにするには、稼動中の DNS サーバーにレコードを追加する必要があります。稼働中の DNS サーバーがない場合は、DNS サーバーをインストールする必要があります。

（注） 実行中の DNS サーバーにレコードを追加するには、*Cisco Network Registrar* などの製品を使用できます。

SSM 設定の制約事項

次に、SSM を設定する際の制約事項を示します。

- IGMPv3 で SSM を使用するには、Cisco IOS ルータ、アプリケーションが稼働しているホスト、そしてアプリケーション自体が SSM をサポートしている必要があります。
- SSM にまだ対応していないネットワーク内の既存のアプリケーションは、（S,G）チャネルの加入登録をサポートするように変更していない限り、SSM 範囲内では機能しません。そのため、既存のアプリケーションが指定の SSM 範囲内のアドレスを使用する場合、ネットワークで SSM をイネーブルにすると問題が発生することがあります。
- IGMP スヌーピング：IGMPv3 で使用される新しいメンバーシップ レポート メッセージは、旧型の IGMP スヌーピング devicesでは正しく認識されない場合があります。
- SSM をレイヤ 2 スイッチング メカニズムとともに使用する場合は、ある程度のアドレス管理が必要となります。Cisco Group Management Protocol（CGMP）、IGMP スヌーピング、または Router-Port Group Management Protocol（RGMP）でサポートされるのはグループ固有のフィルタリングだけであり、（S, G）チャネル固有のフィルタリングはサポートされていません。同じスイッチドネットワーク内の異なるレシーバーが異なる（S,G）チャネルを要求し、これらのチャネルが同じグループを共有している場合、レシーバーは上記のような既存メカニズムの利点を活用できません。どちらのレシーバーも、すべての（S, G）チャネル トラフィックを受信し、不要なトラフィックを入力から除外します。SSM は、独立した多くのアプリケーションに SSM 範囲のグループ アドレスを再利用できるので、このような状況では、スイッチド ネットワークのトラフィック フィルタリング機能が低下する可能性があります。そのため、アプリケーションに対して SSM 範囲の IP アドレスをランダムに使用し、SSM 範囲内の 1 つのアドレスがさまざまなアプリケーションに再利用される可能性を小さくすることが重要です。たとえば、TV チャネル セットを提供するアプリケーション サービスで、SSM を使用する場合は、各 TV（S, G）チャネルに異なるグループを使用する必要があります。このようにすれば、同じアプリケーション サービス内の異なるチャネルに複数のレシーバが接続されていても、レイヤ 2 デバイスを含むネットワークでトラフィック エイリアシングが発生しなくなります。
- PIM-SSM では、ラストホップ ルータは、そのインターフェイス上に適切な（S, G）加入登録があると、定期的に（S, G）Join メッセージを送信し続けます。このため、レシーバが（S, G）加入を送信する限り、ソースが長時間（または二度と）トラフィックを送信しなくてもレシーバからソースへの最短パス ツリー（SPT）状態が維持されます。

  送信元がトラフィックを送信し、レシーバーがグループに加入している場合にだけ（S, G）ステートが維持される PIM-SM では、これとは対照的な状況が発生します。PIM-SM では、送信元がトラフィックの送信を 3 分以上停止すると、（S, G）ステートは削除され、その送信元からのパケットが RPT を通じて再度到達した場合のみに再確立されます。PI-SSM では、送信元がアクティブであることをレシーバに通知するメカニズムがないので、レシーバが（S, G）チャネルの受信を要求している限り、（S, G）ステートを維持する必要があります。

次に、SSM マッピングを設定する際の制約事項を示します。

- SSM マッピング機能は、完全な SSM の利点を共有しません。SSM マッピングでは、ホストからグループ G の加入が取得され、1 つまたは複数のソースに関連付けられているアプリケーションでこのグループを指定できるため、グループ G ごとにこのようなアプリケーション 1 つのみをサポートできます。それにもかかわらず、完全な SSM アプリケーションは、SSM マッピングにも使用される同じグループを共有することができます。
- 完全な SSM への移行ソリューションとして SSM マッピングだけを使用する場合は、ラストホップ ルータの IGMPv3 をイネーブルにする際に十分に注意してください。

SSM および SSM マッピングに関する情報

SSM コンポーネント

SSM は、1 対多のアプリケーション（ブロードキャスト アプリケーション）に最適なデータグラム配信モデルです。

SSM は、オーディオおよびビデオ ブロードキャスト アプリケーション環境を対象とした IP マルチキャスト ソリューションのシスコによって実装されたコア ネットワーキング テクノロジーで、RFC 3569 に説明されています。次のコンポーネントを組み合わせることで、SSM の実装がサポートされます。

- Protocol Independent Multicast Source-Specific Mode（PIM-SSM）
- インターネット グループ管理プロトコル バージョン 3（IGMPv3）

Protocol Independent Multicast（PIM）SSM（PIM-SSM）は、SSM の実装をサポートするルーティング プロトコルで、PIM スパース モード（PIM-SM）から派生しました。IGMP は、ホストがルータにマルチキャスト グループ メンバーシップを伝えるために使用するインターネット技術特別調査委員会（IETF）標準トラック プロトコルです。IGMP バージョン 3 は、SSM に必要なソース フィルタリングをサポートします。SSM を IGMPv3 と共に実行するには、SSM が IOS ルータ、アプリケーションが実行されるホスト、およびアプリケーション自体でサポートされる必要があります。

Internet Standard Multicast と SSM の違い

インターネットと多くの企業イントラネットの標準 IP マルチキャスト インフラストラクチャは、PIM-SM プロトコルと Multicast Source Discovery Protocol（MSDP）に基づいています。これらのプロトコルは信頼でき、広範で、効率的であることが証明されています。しかし、インターネット標準マルチキャスト（ISM）サービス モデルの複雑さと機能性の制限があります。たとえば、ISM では、ネットワークは、実際にマルチキャスト トラフィックを送信しているホストについての情報を維持する必要があります。SSM では、この情報は IGMPv3 によって最後のホップ デバイスにリレーされた発信元アドレスを介して受信することで提供されます。SSM は、ISM に関連付けられた問題への対応を強化し、ネットワーク内で ISM 用に開発され

たプロトコルと共存することを目的としています。一般に、SSMはSSMを使用するアプリケーションにIP マルチキャスト サービスを提供します。

ISMサービスはRFC 1112で定義されています。このサービスは、任意のソースからマルチキャスト ホスト グループと呼ばれるレシーバのグループへのIPデータグラムの配信によって構成されています。マルチキャストホストグループのデータグラム トラフィックは、任意のIPユニキャスト送信元アドレスSとIP宛先アドレスとしてのマルチキャスト グループ アドレスGのデータグラムで構成されます。システムはホストグループのメンバになることによってこのトラフィックを受信します。ホスト グループのメンバーシップにはIGMP バージョン1、2、または3によるホスト グループのシグナリングが必要です。

SSM では、データグラムは（S, G）チャネルに基づいて配信されます。1 つの (S, G) チャネルのトラフィックは、IP宛先アドレスとしてIPユニキャスト ソース アドレスSとマルチキャスト グループ アドレスGを持つデータグラムで構成されています。システムは、（S, G）チャネルのメンバになることによって、このトラフィックを受信します。SSM と ISM のどちらでも、ソースになるためにシグナリングは必要ありません。ただし、SSMでは、レシーバーは特定の送信元からのトラフィックの受信または非受信を決めるために（S, G）への加入または脱退を行う必要があります。つまり、レシーバは加入した（S, G）チャネルからだけトラフィックを受信できます。一方、 ISM では、レシーバは受信するトラフィックの送信元の IP アドレスを知る必要はありません。提案されているチャネル加入シグナリングの標準的な方法では、IGMP INCLUDE モード メンバーシップ レポートを使用します。これは、IGMP バージョン 3 でのみサポートされています。

IP マルチキャスト グループ アドレス範囲の設定済みのサブセットにSSM配信モデルを適用することにより、SSM と ISM サービスを一緒に使用できます。インターネット割り当て番号局（IANA）は、SSMアプリケーションおよびプロトコル用に232.0.0.0～232.255.255.255のアドレス範囲を確保しています。ソフトウェアでは、224.0.0.0 ～ 239.255.255.255 の IP マルチキャスト アドレス範囲の任意のサブセットのSSM設定を許可します。SSM範囲が定義されると、アプリケーションが明示的な (S, G) チャネル加入登録を使用するように変更されているか、URL Rendezvous Directory（URD）によってSSMに対応していない限り、SSM範囲内でアドレスを使用しようとする場合に既存のIPマルチキャスト レシーバアプリケーションはトラフィックを受信しません。

SSM の動作

確立されているネットワークは、IPマルチキャストサービスがPIM SMに基づいているので、SSM サービスをサポートできます。ドメイン間の PIM-SM に必要なプロトコルがすべて揃っていないネットワークでも、SSMを単独で導入できます。つまり、SSMはRPを必要としないため、Auto-RP、MSDP、またはブートストラップ ルータ（BSR）などのRPメカニズムの必要がありません。

SSM がすでに PIM-SM 用に設定済みのネットワークで配備されている場合、ラスト ホップ ルータのみを SSM をサポートするソフトウェア イメージにアップグレードする必要があります。レシーバに直接接続されていないルータを SSM をサポートするソフトウェア イメージにアップグレードする必要はありません。一般的に、これらのラストホップではないルータは、SSM範囲でPIM-SMのみを実行する必要があります。これらは、MSDPシグナリング、登録、

または PIM-SM 共有ツリー動作が SSM 範囲内で発生することを抑制するために、追加のアクセス コントロール設定を必要とする場合もあります。

SSM モードの動作は、**ip pim ssm** グローバル コンフィギュレーション コマンドを使用して SSM 範囲を設定することによってイネーブルできます。この設定による影響は次のとおりです。

- SSM 範囲内のグループの場合、(S, G) チャネル加入は IGMPv3 INCLUDE モード メンバーシップ レポートによって受け入れられます。
- SSM 範囲のアドレスの PIM 動作は、PIM-SM の派生モードである PIM-SSM に変更されます。このモードでは、PIM (S, G) 加入およびプルーニング メッセージのみがルータによって生成されます。ランデブー ポイント ツリー（RPT）動作に関連した着信メッセージは無視されるか、拒否され、着信 PIM 登録メッセージは登録停止メッセージによってただちに応答されます。ラストホップ ルータ以外のルータでは、PIM-SSM は PIM-SM と下位互換性を保ちます。したがって、ラストホップ ルータ以外のルータは SSM グループに PIM-SM を使用できます（SSM をサポートしていない場合など）。
- SSM 範囲内のグループの場合、SSM 範囲内の MSDP Source-Active（SA）メッセージは受け入れ、生成、または転送されません。

IGMPv3 ホスト シグナリング

IGMPv3 は、ホストがマルチキャスト グループのラスト ホップ ルータにメンバーシップを伝える IETF 標準トラック プロトコルの第 3 バージョンです。IGMPv3 は、グループ メンバーシップを伝える能力をホストに与えます。これによってソースに関するフィルタリングが可能になります。ホストは、特定のソースを除いて、グループに送信するすべてのソースからトラフィックを受信したい（EXCLUDE と呼ばれるモード）、またはグループに送信する特定のソースからのみトラフィックを受信したい（INCLUDE と呼ばれるモード）と伝えることができます。

IGMPv3 は、ISM および SSM と同時に動作可能です。ISM では、EXCLUDE モードと INCLUDE モードの両方のレポートがラスト ホップ ルータによって受け入れられます。SSM では、INCLUDE モード レポートのみがラスト ホップ ルータによって受け入れられます。

の利点

IP マルチキャスト アドレス管理が不要

ISM サービスで、トラフィック ディストリビューションは使用する IP マルチキャスト グループ アドレスにのみ基づくため、アプリケーションは一意の IP マルチキャスト グループ アドレスを取得する必要があります。異なるソースとレシーバを持つ2つのアプリケーションが同じ IP マルチキャスト グループ アドレスを使用すると、両方のアプリケーションのレシーバが両方のアプリケーションのソースからトラフィックを受信します。適切にプログラムしている場合、レシーバは不要なトラフィックをフィルタできますが、この状態は一般的に許容できないレベルの不要なトラフィックを生み出します。

アプリケーションへの一意の IP マルチキャスト グループ アドレスの割り当ては問題となります。最も短期のアプリケーションはセッション記述プロトコル（SDP）やセッション通知プロトコル（SAP）のようなメカニズムを使用して、ランダムアドレスを取得します。これは、インターネット内のアプリケーションの増加によってうまく機能しないソリューションです。長期アプリケーションの現在のベスト ソリューションは、RFC 2770 に説明されていますが、このソリューションは各自律システムが 255 の使用可能な IP マルチキャスト アドレスのみに限定される制限の影響を受けます。

SSM で、他のソースからのトラフィックとは関係なく、各ソースからのトラフィックはネットワーク内のルータ間で転送されます。このため、異なるソースが SSM 範囲のマルチキャスト グループ アドレスを再利用できます。

望ましくないソースからの DoS 攻撃を防ぐ

SSM で、個別の各ソースからのマルチキャスト トラフィックは、（IGMPv3、IGMP v3lite または URD メンバーシップによって）レシーバから要求された場合にのみネットワーク中に転送されます。これに対し、ISM はマルチキャスト グループに送信するアクティブなソースからそのマルチキャスト グループを要求するすべてのレシーバにトラフィックを転送します。インターネット ブロードキャスト アプリケーションで、トラフィックを同じマルチキャスト グループにただ送信するだけで、望ましくないソースが実際のインターネットブロードキャスト ソースを簡単に妨害できるため、この ISM の動作は非常に望ましくありません。この状況は、レシーバ側で不要なトラフィックによって帯域幅を消耗させるため、インターネットブロードキャストの無停止の受信を妨害します。SSM では、トラフィックをマルチキャスト グループにただ送信するだけでは、このような種類の DoS 攻撃は行えません。

導入と管理が容易

ネットワークがマルチキャスト グループに送信しているアクティブ ソースについての情報を維持する必要がないため、SSM は簡単にインストールし、ネットワークでプロビジョニングできます。この要件は、（IGMPv1、IGMPv2、または IGMPv3 を使用する）ISM でのみ存在します。

ISM サービスの現在の標準ソリューションは PIM-SM と MSDP です。PIM-SM（Auto-RP または BSR の必要性を含む）および MSDP での Rendezvous Point（RP）管理は、ネットワークがアクティブ ソースについて学習するためにのみ必要です。この管理は SSM では必要ありません。このため、SSM は ISM よりインストールや管理が簡単で、配備での動作面の拡張も ISM より簡単です。SSM のインストールが簡単であるその他の要素は、既存の PIM-SM ネットワークを活用でき、ラスト ホップ ルータをアップグレードするだけで IGMPv3、IGMP v3lite、または URD をサポートできる点です。

インターネット ブロードキャスト アプリケーションに最適

上記の 3 つの利点により、次の理由で SSM はインターネット ブロードキャスト スタイルのアプリケーションに理想的です。

- 一意の IP マルチキャスト アドレスなしで SSM によって、インターネット ブロードキャスト サービスを提供できるため、コンテンツ プロバイダーはサービスを簡単に提供でき

ます（コンテンツ プロバイダーにとって、IP マルチキャスト アドレス割り当てはこれまで深刻な問題でした）。

- インターネット ブロードキャスト サービスは多数のレシーバに公開されることにより、DoS 攻撃の最も一般的な対象となるため、このような攻撃の阻止はインターネット ブロードキャスト サービスの重要な要素です。
- SSM はインストールや動作が簡単なため、特に、コンテンツを複数の独立した PIM ドメイン間で転送する必要のある場合（SSM のために PIM ドメイン間で MSDP を管理する必要がないため）、ネットワーク オペレータにとって理想的です。

SSM マッピングの概要

管理上または技術上の理由によりエンド システム上で SSM をサポートすることができない、または望ましくない場合、SSM マッピングは SSM 移行をサポートします。SSM を使用して IGMPv3 をサポートしていないレガシー STB に対して、ライブ ストリーミング ビデオを提供することは、SSM マッピングの一般的な応用例です。

典型的な STB 配置では、各 TV チャネルは独立した 1 つの IP マルチキャスト グループを使用し、その TV チャネルの送信を行うアクティブなサーバは 1 つです。1 つのサーバから複数の TV チャネルへの送信は可能ですが、各チャネルのグループはそれぞれ異なります。このようなネットワーク環境で、ルータが特定のグループ G の IGMPv1 または IGMPv2 のメンバーシップ レポートを受信した場合、レポートの宛先は暗黙的に、そのマルチキャスト グループに関連付けられている TV チャネルの well-known TV サーバになります。

SSM マッピングは、グループに送信しているソースをラスト ホップ ルータで検出する手段を提供します。SSM マッピングが設定されている場合、特定のグループ G の IGMPv1 または IGMPv2 のメンバーシップ レポートを受信したルータは、レポートを、このグループに関連付けられている既知のソースの 1 つ以上の (S, G) チャネル メンバーシップに変換します。

ルータはグループ G の IGMPv1 または IGMPv2 のメンバーシップ レポートを受信すると、SSM マッピングを使用して、グループ G の 1 つ以上のソース IP アドレスを決定します。その後、SSM マッピングは IGMPv3 レポートの INCLUDE (G, [S1, G], [S2, G]...[Sn, G]) に従ってメンバーシップ レポートを変換し、IGMPv3 レポートを受信したときと同様に続行します。ルータは、IGMPv1 または IGMPv2 メンバーシップ レポートを受信し続ける限り、さらに、グループの SSM マッピングが変更されない限り、PIM Join を（S1, G）から（Sn, G）までに送信し、これらのグループに加入し続けます。このため、SSM マッピングにより、IGMPv3 が未サポートであるレガシー STB への映像配信や、IGMPv3 ホスト スタックを利用しないアプリケーションに SSM を活用できます。

SSM マッピング機能を使用すると、ラスト ホップ ルータはスタティックに設定されたルータ上のテーブルまたは DNS サーバへの問い合わせを通じて、ソース アドレスを決定できます。スタティックに設定されたテーブルが変更された場合や、DNS マッピングが変更された場合、ルータは、現在のソースを加入したグループに関連付けたままにします。

スタティック SSM マッピング

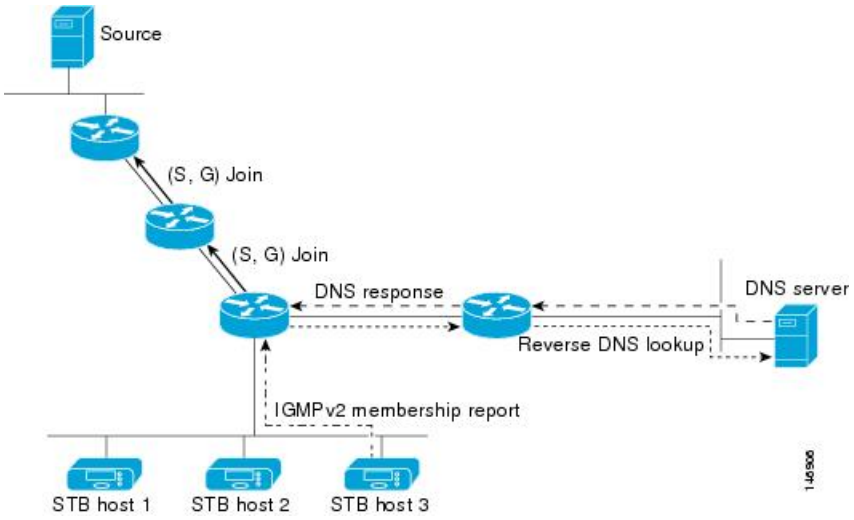
SSM スタティック マッピングを使用して、スタティック マップを使用してグループに送信するソースを決定するようにラスト ホップ ルータを設定できます。スタティック SSM マッピングを使用するには、グループ範囲を定義するアクセス リスト（ACL）を設定する必要があります。これらの ACL によって許可されたグループを **ip igmp static ssm-map** グローバル コンフィギュレーション コマンドを使用してソースにマッピングできます。

DNS が必要ない小規模なネットワークで、または一時的に不正確になった DNS マッピングをローカルに上書きするために、スタティック SSM マッピングを設定できます。設定されたスタティック SSM マッピングは、DNS マッピングよりも優先されます。

DNS ベースの SSM マッピング

DNS ベースの SSM マッピングを使用して、逆 DNS ルックアップを実行してグループを送信するソースを決定するようにラスト ホップ ルータを設定できます（次の図を参照）。DNS ベースの SSM マッピングが設定されると、ルータはグループ アドレス G を含むドメイン名を構築し、DNS への逆ルックアップを実行します。ルータにより、この構築されたドメイン名に戻される IP アドレス リソース レコード（IP ARR）がルックアップされ、戻された IP アドレスが、このグループに関連付けられるソース アドレスとして使用されます。SSM マッピングでサポートできる送信元の数は、グループごとに最大 20 です。ルータは各グループに設定されているすべてのソースに加入します。

図 *11 : DNS* ベースの *SSM* マッピング



ラスト ホップ ルータが 1 つのグループの複数のソースに加入できるようにする SSM マッピング メカニズムを使用すると、TV ブロードキャストのソース冗長性を提供できます。このコンテキストでは、同じ TV チャネルで 2 つのビデオ ソースに加入するために、SSM マッピングを使用しているラスト ホップ ルータによって、冗長性が提供されます。ただし、ラスト ホップ ルータでのビデオ トラフィックの重複を防ぐため、ビデオ ソースは、サーバ側のスイッチオーバー メカニズム（1 つのビデオ ソースがアクティブになる間、残りのバックアップ ビデオ ソースがパッシブになる）を使用する必要があります。パッシブの送信元は待機状態にな

り、アクティブな送信元の障害が検出された場合に、そのTVチャネルにビデオ トラフィックを送信します。このため、サーバ側のスイッチオーバー メカニズムによって、1台のサーバだけが TV チャネルにビデオ トラフィックを実際に送信するようになります。

G1、G2、G3、G4 を含むグループ G について 1 つ以上のソース アドレスをルックアップするには、次の DNS リソース レコード（RR）を DNS サーバで設定する必要があります。

G4.G3.G2.G1 [*multicast-domain*] [*timeout*]	IN A *source-address-1*
	IN A *source-address-2*
	IN A *source-address-n*

multicast-domain 引数は、設定可能なDNSプレフィックスです。デフォルトDNSプレフィックスは、in-addr.arpa です。インストールがインターネットから切り離されている場合、またはマッピングするグループ名が自分の所有するグローバル スコープのグループ アドレス（SSM用に設定する RFC 2770 タイプのアドレス）である場合にだけ、デフォルトのプレフィックスを使用します。

timeout 引数は、SSMマッピングを実行しているルータがDNSルックアップをキャッシュする時間を設定します。この引数はオプションで、エントリが設定されているゾーンのタイムアウトのデフォルトです。タイムアウトは、ルータがこのグループについて DNS サーバに問い合わせるまで、現在のマッピングを保持する期間を示します。タイムアウトはDNS RRエントリのキャッシュ時間から導出され、DNS サーバでグループ/ソースごとに設定できます。ルータによって生成されるDNSクエリー数を最小にする場合は、この時間に大きな値を設定します。新しいソースアドレスですべてのルータを早く更新する場合は、この時間に小さな値を設定します。

（注） DNS RR の設定に関する詳細については、DNS サーバのマニュアルを参照してください。

ソフトウェアでDNSベースのSSMマッピングを設定するには、いくつかのグローバル コマンドを設定する必要がありますが、チャネルごとに特定の設定をする必要はありません。追加チャネルが追加された場合も、SSM マッピングの設定は変更しません。DNS ベースの SSM マッピングが設定されるときに、1つまたは複数のDNSサーバによって、マッピングが全体的に処理されます。DNS ベースの SSM マッピングで、設定および冗長性管理に使用されるすべての DNS テクニックを必要なエントリに適用できます。

SSM マッピングの利点

- SSM マッピング機能は、IGMPv3 に基づく純粋な SSM ソリューションと同じくらいに、ネットワーク導入および管理を簡単にします。SSM マッピングをイネーブルにするために、いくつかの追加設定が必要です。
- SSM の利点である DoS 攻撃の禁止は、SSM マッピングの設定時に適用されます。SSM マッピングを設定した場合、まだ DoS 攻撃に対して脆弱な唯一のネットワーク セグメントが、ラスト ホップ ルータに接続された LAN のレシーバになります。これらのレシーバはまだ IGMPv1 および IGMPv2 を使用しているため、同じ LAN 上の不要なソースからの

攻撃に対して脆弱です。ただし、SSM マッピングは、ネットワーク上のあらゆる不要なソースからのマルチキャスト トラフィックからこれらのレシーバ（およびそれらに繋がるネットワーク パス）を保護します。

- SSMマッピングを使用したネットワーク内でのアドレスの割り当てには、調整が必要ですが、ネットワークからのコンテンツが他のネットワークに転送される場合でも、外部認証局からの割り当ては必要ではありません。

SSM および SSM マッピングの設定方法

SSM の設定

SSM を設定するには、次の手順を実行します。

この手順は任意です。

始める前に

Source Specific Multicast（SSM）範囲の定義にアクセスリストを使用する場合、**ip pim ssm** コマンドでアクセスリストを参照する前にアクセスリストを設定します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip pim ssm** [**default** | **range** *access-list*]
4. **interface** *type number*
5. **ip pim** {**sparse-mode** | **sparse-dense-mode**}
6. **ip igmp version 3**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**ip pim ssm** [**default** | **range** *access-list*] 例： スイッチ(config)# **ip pim ssm range 20**	IP マルチキャスト アドレスの SSM 範囲を定義します。
ステップ 4	**interface** *type number* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	IGMPv3 をイネーブルに設定可能なホストに接続されているインターフェイスを選択し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**ip pim** {**sparse-mode** | **sparse-dense-mode**} 例： スイッチ(config-if)# **ip pim sparse-mode**	インターフェイスに対して PIM をイネーブルにします。
ステップ 6	**ip igmp version 3** 例： スイッチ(config-if)# **ip igmp version 3**	このインターフェイス上で IGMPv3 をイネーブルにします。デフォルトでは、IGMP のバージョン 2 が設定されます。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SSM マッピングの設定

スタティック SSM マッピングの設定

スタティック SSM マッピングを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp ssm-map enable**
4. **no ip igmp ssm-map query dns**
5. **ip igmp ssm-map static** *access-list source-address*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp ssm-map enable** 例： `スイッチ(config)# ip igmp ssm-map enable`	設定されている SSM 範囲で、グループの SSM マッピングをイネーブルにします。 （注） このコマンドでは、デフォルトで、DNS ベースの SSM マッピングがイネーブルにされます。
ステップ 4	**no ip igmp ssm-map query dns** 例：	（任意）DNS ベースの SSM マッピングをディセーブルにします。

	コマンドまたはアクション	目的
	`スイッチ(config)# no ip igmp ssm-map query dns`	（注） スタティック SSM マッピングだけを使用する場合は、DNS ベースの SSM マッピングをディセーブルにします。デフォルトでは、**ip igmp ssm-map** コマンドによって DNS ベースの SSM マッピングがイネーブルになります。
ステップ 5	**ip igmp ssm-map static** *access-list source-address* 例： `スイッチ(config)# ip igmp ssm-map static 11 172.16.8.11`	スタティック SSM マッピングを設定します。 • *access-list* 引数に入力した ACL によって、*source-address* 引数に入力したソース IP アドレスにマッピングされるグループが決まります。 （注） 追加のスタティック SSM マッピングを設定することもできます。SSM マッピングを追加設定した場合、ルータが SSM 範囲のグループの IGMPv1 または IGMPv2 のメンバーシップレポートを受信すると、deviceは、設定されている各 **ip igmp ssm-map static** コマンドに基づいて、そのグループに関連付けられている送信元アドレスを特定します。deviceは各グループに最大 20 の送信元を関連付けます。 必要な場合は、ステップを繰り返して、追加のスタティック SSM マッピングを設定します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

DNS ベースの SSM マッピングの設定（CLI）

DNS ルックアップを実行してグループに送信を実行しているソースの IP アドレスを認識するよう、ラスト ホップ ルータを設定する場合は、この作業を実行します。

始める前に

- このタスクを実行する前に、IP マルチキャスト ルーティングをイネーブルにし、PIM スパース モードをイネーブルにし、SSM を設定します。詳細については、「Configuring Basic Multicast」モジュールを参照してください。
- SSM マッピングを設定し、DNS ルックアップで使用できるようにするためには、実行中の DNS サーバにレコードを追加できるようになる必要があります。稼働中の DNS サーバがない場合は、DNS サーバをインストールする必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip igmp ssm-map enable**
4. **ip igmp ssm-map query dns**
5. **ip domain multicast** *domain-prefix*
6. **ipname-server** *server-address1* [*server-address2server-address6*]
7. 冗長性のために追加の DNS サーバーを設定する場合は、必要に応じて、ステップ 6 を繰り返します。
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device# enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip igmp ssm-map enable** 例： `Device(config)# ip igmp ssm-map enable`	設定されている SSM 範囲で、グループの SSM マッピングをイネーブルにします。
ステップ 4	**ip igmp ssm-map query dns** 例：	（任意）DNS ベースの SSM マッピングをイネーブルにします。

	コマンドまたはアクション	目的
	`Device(config)# ip igmp ssm-map query dns`	• デフォルトでは、**ip igmp ssm-map** コマンドによってDNSベースのSSMマッピングがイネーブルになります。実行コンフィギュレーションに保存されるのは、このコマンドを **no** 形式で使用した場合だけです。 （注） DNSベースのSSMマッピングがディセーブルの場合、このコマンドを使用してDNS ベースの SSM マッピングを再度イネーブルにします。
ステップ 5	**ip domain multicast** *domain-prefix* 例： `Device(config)# ip domain multicast ssm-map.cisco.com`	（任意）Cisco IOS XE ソフトウェアが DNS ベースの SSM マッピングに使用するドメイン プレフィックスを変更します。 • デフォルトでは、ip-addr.arpa ドメイン プレフィックスが使用されます。
ステップ 6	**ipname-server** *server-address1* [*server-address2server-address6*] 例： `Device(config)# ip name-server 10.48.81.21`	名前とアドレスの解決に使用する 1 つまたは複数のネーム サーバーのアドレスを指定します。
ステップ 7	冗長性のために追加の DNS サーバーを設定する場合は、必要に応じて、ステップ 6 を繰り返します。	--
ステップ 8	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 9	**show running-config** 例： `Device# show running-config`	入力を確認します。
ステップ 10	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

SSM マッピングを使用したスタティック トラフィック転送の設定

ラスト ホップ ルータ上の SSM マッピングでスタティック トラフィック転送を設定する場合は、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp static-group** *group-address* **source ssm-map**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	SSM マッピングを使用してマルチキャスト グループにスタティックにトラフィックを転送するインターフェイスを選択し、インターフェイス コンフィギュレーション モードを開始します。 （注） SSM マッピングを使用したトラフィックのスタティック転送は、DNSベースのSSMマッピングとスタティックに設定されたSSMマッピングのいずれかで機能します。
ステップ 4	**ip igmp static-group** *group-address* **source ssm-map** 例： `スイッチ(config-if)# ip igmp static-group 239.1.2.1 source ssm-map`	そのインターフェイスから（S, G）チャネルへのスタティック転送用のSSMマッピングを設定します。 このコマンドは、特定グループに SSM トラフィックをスタティックに転送する場合に使用します。チャネルの送信元アドレスを決定するにはDNSベースの SSM マッピングを使用します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

SSM マッピングの設定と動作の確認

SSM マッピングの設定と動作を確認するには、次の手順を実行します。

手順の概要

1. **enable**
2. **show ip igmp ssm-mapping**
3. **show ip igmp ssm-mapping** *group-address*
4. **show ip igmp groups** [*group-name* | *group-address* | *interface-type interface-number*] [**detail**]
5. **show host**
6. **debug ip igmp** *group-address*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show ip igmp ssm-mapping** 例： `スイッチ# show ip igmp ssm-mapping` `SSM Mapping : Enabled` `DNS Lookup : Enabled` `Mcast domain : ssm-map.cisco.com` `Name servers : 10.0.0.3` `10.0.0.4`	（任意）SSMマッピングの設定に関する情報を表示します。
ステップ 3	**show ip igmp ssm-mapping** *group-address* 例：	（任意）SSMマッピングが特定のグループに使用するソースを表示します。

	コマンドまたはアクション	目的
	スイッチ# **show ip igmp ssm-mapping 232.1.1.4** Group address: 232.1.1.4 Database : DNS DNS name : 4.1.1.232.ssm-map.cisco.com Expire time : 860000 Source list : 172.16.8.5 : 172.16.8.6	次に、設定済みの DNS ベースの SSM マッピングに関する情報の例を示します。ルータはソース 172.16.8.5 および 172.16.8.6 にグループ 232.1.1.4 をマッピングする DNS ベースのマッピングを使用しています。このエントリのタイムアウトは、860000 ミリ秒（860 秒）です。
ステップ 4	**show ip igmp groups** [*group-name* | *group-address* | *interface-type interface-number*] [**detail**] 例： スイッチ# **show ip igmp group 232.1.1.4 detail** Interface: GigabitEthernet2/0/0 Group: 232.1.1.4 SSM Uptime: 00:03:20 Group mode: INCLUDE Last reporter: 0.0.0.0 CSR Grp Exp: 00:02:59 Group source list: (C - Cisco Src Report, U - URD, R - Remote, S - Static, M - SSM Mapping) Source Address Uptime v3 Exp CSR Exp Fwd Flags 172.16.8.3 00:03:20 stopped 00:02:59 Yes CM 172.16.8.4 00:03:20 stopped 00:02:59 Yes CM 172.16.8.5 00:03:20 stopped 00:02:59 Yes CM 172.16.8.6 00:03:20 stopped 00:02:59 Yes CM	（任意）ルータに直接接続されているレシーバと IGMPによって学習されたレシーバを持つマルチキャスト グループを表示します。 この例の「M」フラグは、SSMマッピングが設定されることを示します。
ステップ 5	**show host** 例： スイッチ# **show host** Default domain is cisco.com Name/address lookup uses domain service Name servers are 10.48.81.21 Codes: UN - unknown, EX - expired, OK - OK, ?? - revalidate temp - temporary, perm - permanent NA - Not Applicable None - Not defined Host Port Flags Age Type Address(es) 10.0.0.0.ssm-map.cisco.c None (temp, OK) 0 IP 172.16.8.5 172.16.8.6 172.16.8.3	（任意）デフォルト ドメイン名、名前のルックアップ サービスのスタイル、ネーム サーバ ホストのリスト、および、ホスト名とアドレスのキャッシュにあるリストを表示します。

	コマンドまたはアクション	目的
ステップ 6	**debug ip igmp** *group-address* 例： スイッチ# **debug ip igmp** IGMP(0): Convert IGMPv2 report (*,232.1.2.3) to IGMPv3 with 2 source(s) using STATIC. スイッチ# **debug ip igmp** IGMP(0): Convert IGMPv2 report (*,232.1.2.3) to IGMPv3 with 2 source(s) using DNS. スイッチ# **debug ip igmp** IGMP(0): DNS source lookup failed for (*, 232.1.2.3), IGMPv2 report failed	（任意）受信および送信した IGMP パケットとホスト関連イベントを表示します。 最初の例の出力は、ルータによってグループ G の IGMPv2 加入が IGMPv3 加入に変換されていることを示しています。 2 番目の例の出力は、DNS ルックアップが成功したことを示しています。 3 番目の例の出力は、DNS ベースの SSM マッピングがイネーブルで、DNS ルックアップが失敗したことを示しています。

SSM および SSM マッピングのモニタリング

SSM のモニタリング

SSM をモニタするには、必要に応じて特権 EXEC モードで次のコマンドを使用します。

コマンド	目的
スイッチ# **show ip igmp groups detail**	IGMPv3 で（S, G）チャネル加入を表示します。
スイッチ# **show ip mroute**	マルチキャスト グループが SSM サービスをサポートしているかどうか、または送信元固有のホスト レポートが受信されたかどうかを表示します。

SSM マッピングのモニタリング

SSM マッピングをモニターするには、次の表の特権 EXEC コマンドを使用します。

表 *31 : SSM* マッピングをモニターするコマンド

コマンド	目的
スイッチ# **show ip igmp ssm-mapping**	SSMマッピングについての情報を表示します。
スイッチ#**show ip igmp ssm-mapping** *group-address*	SSM マッピングが特定のグループに使用する送信元を表示します。

コマンド	目的
スイッチ#**show ip igmp groups** [*group-name* | *group-address* | *interface-type interface-number*] [**detail**]	ルータに直接接続されているレシーバと IGMP によって学習されたレシーバを持つマルチキャスト グループを表示します。
スイッチ#**show host**	デフォルトのドメイン名、名前ルックアップ サービス、ネーム サーバー ホストのリスト、およびホスト名とアドレスのキャッシュされたリストを表示します。
スイッチ#**debug ip igmp** *group-address*	送受信された IGMP パケットと IGMP ホスト関連イベントを表示します。

SSM および SSM マッピングの設定例

IGMPv3 を使用した SSM の例

次に、SSM 用に（IGMPv3 を実行する）デバイスを設定する例を示します。

```
ip multicast-routing
!
interface GigabitEthernet3/1/0
 ip address 172.21.200.203 255.255.255.0
 description backbone interface
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0
 ip address 131.108.1.2 255.255.255.0
 ip pim sparse-mode
 description ethernet connected to hosts
 ip igmp version 3
!
ip pim ssm default
```

SSM フィルタリング例

次に、SSM ルーティングをサポートしないソフトウェア リリースを実行しているレガシー RP ルータでフィルタリングを設定する例を示します。このフィルタリングは SSM 範囲で不要な PIM-SM および MSDP トラフィックをすべて抑制します。このフィルタリングがなくても SSM は動作しますが、レガシーのファースト ホップ ルータとラスト ホップ ルータがネットワークに存在する場合、追加の RPT トラフィックがある場合があります。

```
ip access-list extended  no-ssm-range
  deny   ip any 232.0.0.0 0.255.255.255 ! SSM range
  permit ip any any
! Deny registering in SSM range
ip pim accept-register list no-ssm-range
ip access-list extended msdp-nono-list
```

```
  deny   ip any 232.0.0.0 0.255.255.255 ! SSM Range
  ! .
  ! .
  ! .
  ! See ftp://ftpeng.cisco.com/ipmulticast/config-notes/msdp-sa-filter.txt for other SA
  ! messages that typically need to be filtered.
  permit ip any any
! Filter generated SA messages in SSM range. This configuration is only needed if there
! are directly connected sources to this router. The "ip pim accept-register" command
! filters remote sources.
ip msdp redistribute list msdp-nono-list
! Filter received SA messages in SSM range. "Filtered on receipt" means messages are
! neither processed or forwarded. Needs to be configured for each MSDP peer.
ip msdp sa-filter in msdp-peer1 list msdp-nono-list
! .
! .
! .
ip msdp sa-filter in msdp-peerN list msdp-nono-list
```

SSM マッピングの例

次に、SSMマッピング用にルータを設定する設定例を示します。この例では、機能間の互換性を示すために、他の IGMP および SSM 設定オプションの範囲も示します。例で使用されている機能のすべてを理解していない場合、この設定例をモデルとして使用しないでください。

（注） グローバル SSM 範囲 232.0.0.0/8 のアドレス割り当てはランダムです。この設定例の一部またはすべてをコピーする場合、この例で示されているように、232.1.1.x ではなくランダムアドレス範囲を選択してください。ランダムなアドレス範囲を使用することで、SSM マッピングの使用時に他の SSM の内容をインポートしたときに、アドレスの衝突が発生する可能性を最小限に抑え、競合を防ぐことができます。

```
!
no ip domain lookup
ip domain multicast ssm.map.cisco.com
ip name-server 10.48.81.21
!
!
ip multicast-routing distributed
ip igmp ssm-map enable
ip igmp ssm-map static 10 172.16.8.10
ip igmp ssm-map static 11 172.16.8.11
!
!
.
.
.
!
interface GigabitEthernet0/0/0
 description Sample IGMP Interface Configuration for SSM-Mapping Example
 ip address 10.20.1.2 255.0.0.0
 ip pim sparse-mode
 ip igmp last-member-query-interval 100
 ip igmp static-group 232.1.2.1 source ssm-map
 ip igmp version 3
 ip igmp explicit-tracking
```

```
 ip igmp limit 2
 ip igmp v3lite
 ip urd
!
.
.
.
!
ip pim ssm default
!
access-list 10 permit 232.1.2.10
access-list 11 permit 232.1.2.0 0.0.0.255
!
```

次の表で、SSM マッピング設定例に示されている重要なコマンドについて説明します。

表 *32 : SSM* マッピングの設定例で使用されているコマンドの説明

コマンド	説明
no ip domain lookup	IP DNS に基づいたホスト名からのアドレス変換をディセーブルにします。 （注） **no ip domain-list** コマンドは、IP DNS ベースのホスト名/アドレス間変換をディセーブルにすることにより、SSM マッピングの設定に矛盾が生じないことを示すためにのみ、設定に表示されます。このコマンドがイネーブルの場合、Cisco IOS XE ソフトウェアは、ホスト名として未知の文字列の解決を試みます。
ip domain multicast ssm-map.cisco.com	SSM マッピングのドメイン プレフィックスとして ssm-map.cisco.com を指定します。
ip name-server 10.48.81.21	SSM マッピングおよび DNS が利用されるソフトウェアの他のすべてのサービスで使用される DNS サーバの IP アドレスとして、10.48.81.21 を指定します。
ip multicast-routing	IP マルチキャスト ルーティングをイネーブルにします。
ip igmp ssm-map enable	SSM マッピングをイネーブルにします。
ip igmp ssm-map static 10 172.16.8.10	ソース アドレス 172.16.8.10 を使用するよう、ACL 10 によって許可されるグループを設定します。 • この例では、ACL 10 によって、232.1.2.10 を除く 232.1.2.0/25 範囲ですべてのグループが許可されます。
ip igmp ssm-map static 11 172.16.8.11	ソース アドレス 172.16.8.11 を使用するよう、ACL 11 によって許可されるグループを設定します。 • この例では、ACL 11 によって、グループ 232.1.2.10 が許可されます。

コマンド	説明
ip pim sparse-mode	PIM スパース モードをイネーブルにします。
ip igmp last-member-query-interval 100	IGMPv2 ホストの脱退遅延を減らします。 （注） このコマンドは、SSMマッピングの設定には必要ではありません。ただし、SSM マッピングに依存しているIGMPv2 ホストでは、このコマンドは効果的です。
ip igmp static-group 232.1.2.1 source ssm-map	グループ 232.1.2.1 に関連付けられているソースを特定するために使用されるよう、SSMマッピングを設定します。その結果得られる (S, G) チャネルは、静的に転送されます。
ip igmp version 3	このインターフェイス上で IGMPv3 をイネーブルにします。 （注） このコマンドは、IGMPv3 が SSM マッピングと同時に設定できることを示すためにのみ、設定で使用されますが、必須ではありません。
ip igmp explicit-tracking	マルチキャスト チャネルから脱退する IGMPv3 ホストの脱退遅延を最小限に抑えます。 （注） このコマンドは、SSMマッピングの設定には必要ではありません。
ip igmp limit 2	1 つのインターフェイス当たりのベースで、IGMP メンバーシップ状態から生じる IGMP 状態の数を制限します。 （注） このコマンドは、SSMマッピングの設定には必要ではありません。
ip igmp v3lite	このインターフェイスで IGMP v3lite メンバーシップ レポートの受け入れと処理をイネーブルにします。 （注） このコマンドは、IGMP v3lite が SSM マッピングと同時に設定できることを示すためにのみ、設定で使用されますが、必須ではありません。
ip urd	インターフェイスで確保された URD ポート 465 に送信された TCP パケットの代行受信と URD チャネル加入レポートの処理をイネーブルにします。 （注） このコマンドは、URD が SSM マッピングと同時に設定できることを示すためにのみ、設定で使用されますが、必須ではありません。

コマンド	説明
ip pim ssm default	SSM サービスを設定します。 **default** キーワードは SSM 範囲のアクセスリストを 232/8 と定義します。
access-list 10 permit 232.1.2.10 access-list 11 permit 232.1.2.0 0.0.0.255	スタティック SSM マッピングに使用されるよう、ACL を設定します。 （注） これらは、この設定例で **ip igmp ssm-map static** コマンドによって参照される ACL です。

DNS サーバの設定例

DNS ベースの SSM マッピングを設定するには、DNS サーバ ゾーンを作成するか、または既存のゾーンにレコードを追加する必要があります。DNS ベースの SSM マッピングを使用しているルータで、SSM マッピング以外の目的で DNS も使用している場合、通常設定の DNS サーバを使用する必要があります。そのルータで使用されている DNS 実装が DNS ベースの SSM マッピングだけの場合は、ルート ゾーンが空であるか、またはそれ自身を指すような疑似 DNS セットアップが可能です。

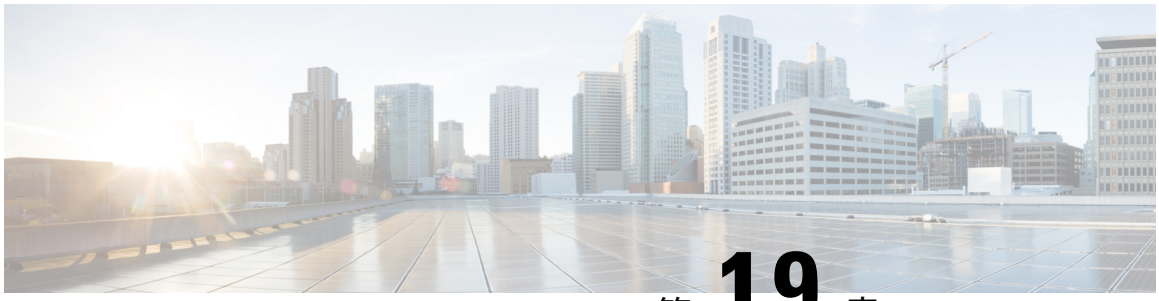
次に、ゾーンを作成し、Network Registrar を使用してゾーン データをインポートする例を示します。

```
Router> zone 1.1.232.ssm-map.cisco.com. create primary file=named.ssm-map
100 Ok
Router> dns reload
100 Ok
```

次に、BIND 8 の named.conf ファイルからゾーン ファイルをインポートする例を示します。

```
Router> ::import named.conf /etc/named.conf
Router> dns reload
100 Ok:
```

（注） ネットワーク レジストラ バージョン 8.0 およびそれ以降では、インポート BIND 8 形式の定義がサポートされます。



第 19 章

MSDP の設定

MSDP の前提条件

MSDP を使用するには、Catalyst 3560-CX スイッチで IP サービス フィーチャ セットをイネーブルにする必要があります。

Multicast Source Discovery Protocol に関する情報

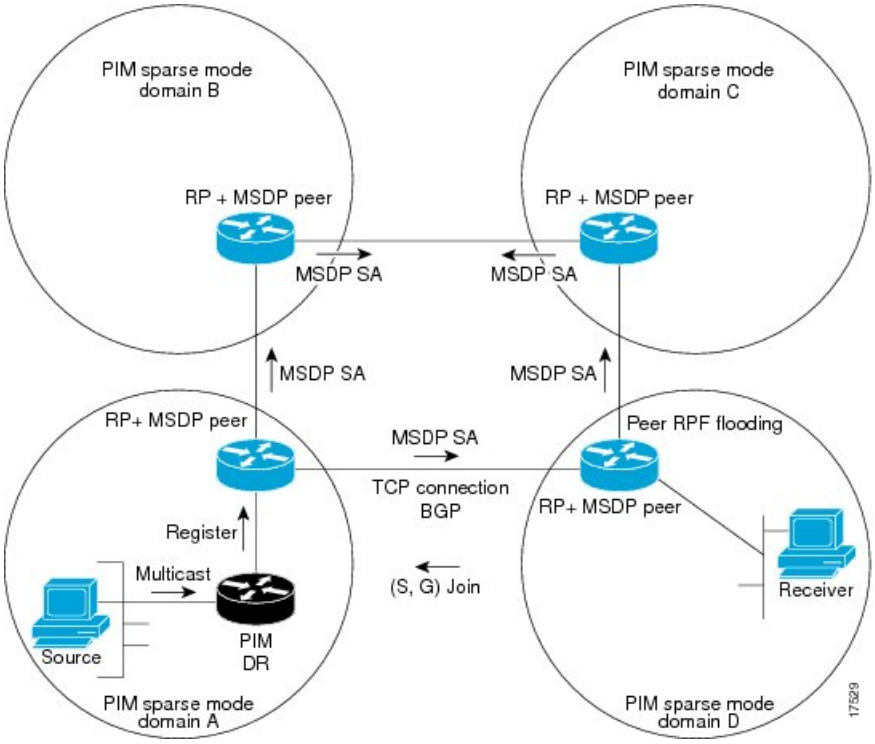
MSDP は複数の PIM-SM ドメインを接続するメカニズムです。MSDP は、他の PIM ドメイン内のマルチキャスト送信元を検出することを目的としています。MSDP の主な利点は、（一般的な共有ツリーではなく）ドメイン間ソース ツリーを PIM-SM ドメインで使用できるようにし、複数の PIM-SM ドメインを相互接続する複雑性を軽減することです。MSDP がネットワークで設定されている場合、RP は他のドメイン内の RP と送信元情報を交換します。RP は、レシーバがいるグループに送信するソースのドメイン間ソース ツリーに参加できます。RP は、そのドメイン内の共有ツリーのルートであり、アクティブ レシーバが存在するドメイン内のすべてのポイントへのブランチがあるため、これを行うことができます。PIM-SM ドメイン外の新しい送信元を（共有ツリーの送信元からのマルチキャスト パケットの到着によって）ラスト ホップ デバイスが認識すると、その送信元に加入要求を送信してドメイン間ソース ツリーに参加できます。

（注） RP に特定グループの共有ツリーがないか、発信インターフェイス リストがヌルの共有ツリーがある場合は、別のドメインの発信元に加入要求を送信しません。

MSDP がイネーブルになっている場合、PIM-SM ドメインの RP は、他のドメインの MSDP 対応デバイスとの MSDP ピアリング関係を維持します。このピアリング関係は TCP 接続を通じて発生します。交換されるのは主にマルチキャスト グループを送信する送信元のリストです。MSDP はピアリング接続に TCP（ポート 639）を使用します。BGP と同様に、ポイントツーポイント TCP ピアリングを使用する場合は、各ピアを明示的に設定する必要があります。さらに、RP 間の TCP 接続は基本的なルーティング システムによって実現されます。受信側の RP では、送信元リストを使用して送信元のパスが確立されます。マルチキャスト ソースがレシーバがいるドメインの対象である場合、マルチキャスト データは PIM-SM で提供される通常のソース ツリー構築メカニズムを使用して配信されます。MSDPは、グループを送信する送信元のアナウンスにも使用されます。これらのアナウンスは、ドメインの RP で発信する必要があります。

図に、2 つの MSDP ピア間の MSDP の動作を示します。PIM では、ドメインの RP に送信元を登録するための標準メカニズムとして、MSDP が使用されます。

図 *12 : RP* ピア間で動作する *MSDP*



MSDP が実装されている場合、次のイベント シーケンスが発生します。

1. 図に示すように、PIM 指定デバイス（DR）が送信元を RP に登録すると、その RP が Source-Active（SA）メッセージをすべての MSDP ピアに送信します。

（注） DR は、（ソースがアクティブになると）カプセル化されたデータをソースごとに 1 回だけ RP に送信します。ソースがタイムアウトした場合、ソースが再度アクティブになるとこのプロセスが実行されます。これは、発信元 RP に登録されているすべての発信元を含んでいる定期的な SA メッセージの場合とは異なります。これらの SA メッセージは MSDP 制御パケットであるため、アクティブな送信元からのカプセル化されたデータを含んでいません。

1. SA メッセージでは、ソース アドレス、ソースの送信先グループ、および RP のアドレスまたは発信者 ID が識別されます（設定されている場合）。

2. SA メッセージを受信する各 MSDP ピアは、発信者からのダウンストリームのすべてのピアに SA メッセージをフラッディングします。場合によっては（図の PIM-SM ドメイン B および C 内の RP の場合など）、RP は複数の MSDP ピアからの SA メッセージのコピーを受信することがあります。ループが作成されないように、RP は BGP ネクスト ホップ データベースに問い合わせて、SA メッセージの発信者へのネクスト ホップを識別します。MBGP とユニキャスト BGP の両方が設定されている場合、MBGP が最初に確認されてからユニキャスト BGP が確認されます。そのネクストホップ ネイバーが発信元の RPF ピアです。RPF ピアへのインターフェイス以外のインターフェイスにある発信元から受信した SA メッセージはドロップされます。そのため、SA メッセージ フラッディング プロセスはピア RPF フラッディングと呼ばれます。ピア RPF フラッディング メカニズムにより、BGP または MBGP は MSDP とともに実行する必要があります。

1. SA メッセージを受信した RP は、グループの (*, G) 送信インターフェイス リストにインターフェイスが存在するかどうかを確認することによって、そのドメイン内にアドバタイズされたグループのメンバが存在するかどうかを確認します。グループ メンバが存在しない場合、RP は何も実行しません。グループ メンバが存在する場合、RP は（S, G）加入要求を送信元に送信します。その結果、ドメイン間ソース ツリーのブランチが自律システムの RP との境界に構築されます。マルチキャスト パケットは、RP に着信すると、その共有ツリーを経由して RP のドメイン内のグループ メンバに転送されます。メンバの DR は、標準的な PIM-SM 手順を使用してソースへのランデブー ポイント ツリー（RPT）に加入することもできます。

2. 発信元 RP は、送信元がグループにパケットを送信し続ける限り、60 秒ごとに（S, G）ステートに関する SA メッセージを定期的に送信し続けます。RP は SA メッセージを受信すると、SA メッセージをキャッシュします。たとえば、発信元 RP 10.5.4.3 から (172.16.5.4, 228.1.2.3) に対する SA メッセージを受信したとします。RP は mroute テーブルを確認し、グループ 228.1.2.3 にアクティブなメンバが存在しないことを検出すると、SA メッセージを 10.5.4.3 のダウンストリームにあるピアに渡します。次に、ドメイン内のホストが加入要求をグループ 228.1.2.3 の RP に送信した場合、その RP はホストへのインターフェイスを (*, 224.1.2.3) エントリの発信インターフェイス リストに追加します。RP は SA メッセージをキャッシュするため、デバイスは (172.16.5.4, 228.1.2.3) のエントリを持ち、ホストが加入を要求するとすぐにソース ツリーに加入できます。

（注） 現行のすべてのサポート対象のソフトウェア リリースでは、MSDP SA メッセージのキャッシュは必須であり、手動でイネーブルまたはディセーブルにすることはできません。デフォルトでは、MSDP ピアが設定されると、 **ip multicast cache-sa-state** コマンドが自動的に実行コンフィギュレーションに追加されます。

MSDP の利点

MSDP には次の利点があります。

- 共有されたマルチキャスト配信ツリーが分割され、共有ツリーがドメインに対してローカルになるように設定できます。ローカル メンバーはローカル ツリーに加入します。共有ツリーへの Join メッセージはドメインから脱退する必要はありません。
- PIM SM ドメインは独自の RP だけを信頼するため、他のドメインの RP に対する信頼度が低下します。このため、送信元の情報がドメイン外部に漏れないようにでき、セキュリティが高まります。
- レシーバーだけが配置されているドメインは、グループ メンバーシップをグローバルにアドバタイズしなくても、データを受信できます。
- グローバルな送信元マルチキャスト ルーティング テーブル ステートが不要になり、メモリが削減されます。

デフォルト MSDP ピア

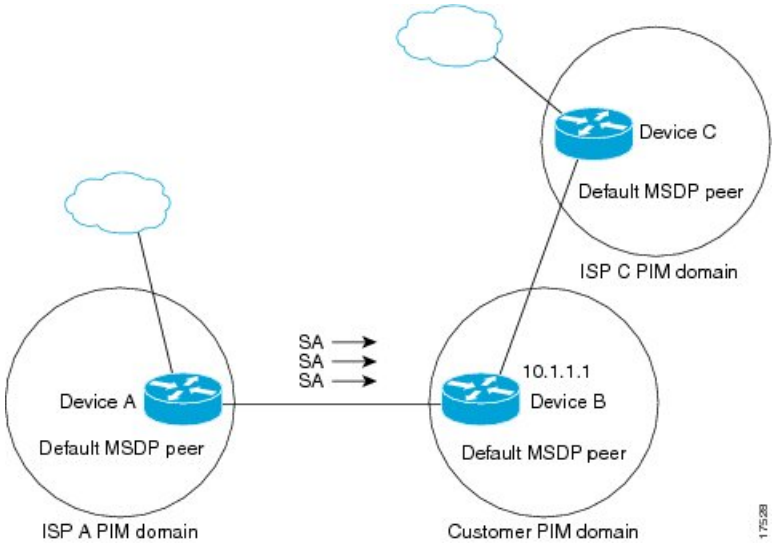
スタブ自律システムには、冗長性を実現するために複数の RP との MSDP ピアリングが必要な場合もあります。たとえば、RPF チェック メカニズムがないため、SA メッセージは複数のデフォルト ピアから受け入れられません。その代わりに、SA メッセージは 1 つのピアからだけ受け入れられます。そのピアに障害が発生した場合、SA メッセージは別のピアから受け入れられます。もちろん、デフォルトのピアが両方とも同じ SAメッセージを送信することがこの基本的な前提となっています。

下の図に、デフォルトの MSDP ピアが使用されるシナリオを示します。この図では、デバイス B を所有するカスタマーが 2 つのインターネット サービス プロバイダ（ISP）を介してインターネットに接続されています。一方の ISP はデバイス A を所有し、もう一方の ISP はデバイス C を所有しています。どちらもそれらの間で BGP も MBGP も実行していません。カスタマーが ISP ドメインまたは他のドメイン内のソースについて学習するために、デバイス B はデバイス A をデフォルト MSDP ピアとして識別します。デバイス B はデバイス A とデバイス C の両方に SA メッセージをアドバタイズしますが、デバイス A だけまたはデバイス C だけから SA メッセージを受け入れます。デバイス A が設定内の最初のデフォルト ピアである場合、デバイス A が稼働していれば デバイス A が使用されます。デバイス A が稼働していない場合に限り、デバイス B がデバイス C からの SA メッセージを受け入れます。

ISP は、プレフィックス リストを使用して、カスタマーのデバイスから受け入れるプレフィックスを定義する場合もあります。カスタマーは、複数のデフォルト ピアを定義します。各ピアには関連するプレフィックスを 1 つまたは複数設定します。

カスタマーは 2 つの ISP を使用しています。カスタマーはこの 2 つの ISP をデフォルト ピアとして定義します。設定内で最初のデフォルト ピアとして特定されているピアが稼働している限り、このピアがデフォルト ピアになり、カスタマーはそのピアから受信するすべての SA メッセージを受け入れます。

図 *13* : デフォルト *MSDP* ピアのシナリオ



デバイス B はデバイス A およびデバイス C に SA をアドバタイズしますが、デバイス A またはデバイス C だけを使用して SA メッセージを受け入れます。デバイス A が設定内の最初のデバイスである場合、デバイス A が稼働していればデバイス A が使用されます。デバイス A が稼働していない場合に限り、デバイス B がデバイス C から SA メッセージを受け入れます。これは、プレフィックス リストを使用しない動作です。

プレフィックス リストを指定すると、リスト内のプレフィックスに対してだけピアはデフォルト ピアになります。プレフィックス リストがそれぞれ関連付けられている場合は、複数のアクティブなデフォルト ピアを設定できます。プレフィックス リストがない場合も、複数のデフォルト ピアを設定できますが、アクティブなデフォルト ピアになるのは最初のピアだけです（このピアにデバイスが接続されていて、ピアがアクティブの場合に限ります）。最初に設定されたピアがダウンするか、このピアとの接続がダウンした場合、2 番目に設定されたピアがアクティブなデフォルト ピアになります。以下同様です。

MSDP メッシュ グループ

MSDP メッシュ グループは、MSDP によってフル メッシュ型に相互接続された MSDP スピーカーのグループです。つまり、グループの各 MSDP ピアには、グループ内の他のすべての MSDP ピアとの MSDP ピアリング関係（MSDP 接続）が必要です。MSDP メッシュ グループが MSDP ピアのグループ間に設定されている場合、SA メッセージのフラッディングが削減さ

れます。グループ内の MSDP ピアがグループ内の別の MSDP ピアから SA メッセージを受信すると、この SA メッセージはグループ内のその他のすべての MSDP ピアに送信されたとみなされるためです。その結果、受信側の MSDP ピアがグループ内の他の MSDP ピアに SA メッセージをフラッディングする必要はありません。

MSDP メッシュ グループの利点

- SA フラッディングの最適化：グループ内に複数のピアがある場合、SA フラッディングを最適化するために MSDP メッシュ グループは特に有用です。
- インターネットを通過する SA トラフィック量の削減：MSDP メッシュ グループを使用すると、SA メッセージは他のメッシュ グループ ピアにフラッディングされません。
- 着信 SA メッセージの RPF チェックの省略：MSDP メッシュ グループが設定されていると、メッシュ グループ ピアからの SA メッセージは常に受け入れられます。

SA 発信フィルタ

デフォルトでは、MSDP を実行するように設定されている RP は、それが RP であるすべてのローカル ソースの SA メッセージを発信します。そのため、RP に登録されているローカル ソースは SA メッセージでアドバタイズされますが、これが望ましくない場合もあります。たとえば、PIM-SM ドメイン内のソースがプライベート アドレス（たとえば、ネットワーク 10.0.0.0/8）を使用している場合、SA 発信フィルタを設定してこれらのアドレスがインターネット上の他の MSDP ピアにアドバタイズされないようにする必要があります。

SA メッセージでアドバタイズされるソースを制御するには、RP に SA 発信フィルタを設定します。SA 発信フィルタを作成すると、SA メッセージでアドバタイズされるソースを次のように制御できます。

- デバイスが SA メッセージでローカル ソースをアドバタイズしないように RP を設定できます。この場合もデバイスは通常の方法で他の MSDP ピアからの SA メッセージを転送します。ローカル ソースの SA メッセージは発信しません。
- 拡張アクセス リストで定義されている（S,G）ペアと一致する、特定のグループに送信するローカル ソースの SA メッセージだけを発信するようにデバイスを設定できます。その他のすべてのローカル ソースは SA メッセージでアドバタイズされません。
- AS パス アクセス リストで定義されている AS パスと一致する、特定のグループに送信するローカル ソースの SA メッセージだけを発信するようにデバイスを設定できます。その他のすべてのローカル ソースは SA メッセージでアドバタイズされません。
- ルート マップで定義されている基準と一致するローカル ソースの SA メッセージだけを発信するようにデバイスを設定できます。その他のすべてのローカル ソースは SA メッセージでアドバタイズされません。
- 拡張アクセス リスト、AS パス アクセス リスト、およびルート マップ（またはそれらのその組み合わせ）を含む SA 発信フィルタを設定します。この場合、ローカル ソースが SA メッセージでアドバタイズされる前に、すべての条件を満たしている必要があります。

MSDP での発信フィルタ リストの使用

デフォルトでは、MSDP 対応デバイスは、受信したすべての SA メッセージをその MSDP ピアすべてに転送します。ただし、発信フィルタ リストを作成することで、SA メッセージが MSDP ピアに転送されないようにできます。発信フィルタ リストは、ローカルに発信されたか別の MSDP ピアから受信したかに関係なくすべての SA メッセージに適用されますが、SA 発信フィルタはローカルに発信された SA メッセージだけに適用されます。ローカル デバイスから発信される MSDP SA メッセージのフィルタをイネーブルにする方法の詳細については、「ローカル ソースの RP によって発信された SA メッセージの制御」の項を参照してください。

発信フィルタ リストを作成すると、デバイスがピアへ転送する SA メッセージを次のように制御できます。

- 指定した MSDP ピアへ転送したすべての発信 SAメッセージをフィルタリングするには、MSDP ピアへの SA メッセージの転送を停止するようにデバイスを設定します。
- 指定した MSDP ピアへ転送した発信 SA メッセージのサブセットを拡張アクセス リストに定義された（S, G）ペアに基づいてフィルタリングするには、拡張アクセス リストで許可されている（S, G）ペアに一致する MSDP ピアへの SA メッセージだけを転送するようにデバイスを設定します。その他のすべての SA メッセージの MSDP ピアへの転送は停止されます。
- 指定した MSDP へ転送した発信 SA メッセージのサブセットをルート マップに定義された一致基準に基づいてフィルタリングするには、ルート マップに定義された基準に一致する SA メッセージだけを転送するようにデバイスを設定します。その他のすべての SA メッセージの MSDP ピアへの転送は停止されます。
- 指定したピアからの発信 SA メッセージのサブセットを SA メッセージに含まれているアナウンス側 RP アドレスに基づいてフィルタリングするには、SA メッセージが1 つ以上の MSDP ピアに送信されていても、それらの発信元に基づいて発信 SA メッセージをフィルタリングするようにデバイスを設定します。その他のすべての SA メッセージの MSDP ピアへの転送は停止されます。
- 拡張アクセス リスト、ルート マップ、および RP アクセス リストまたは RP ルート マップのいずれかを含む発信フィルタ リストを設定できます。この場合、MSDP ピアで発信 SA メッセージを転送するにはすべての条件を満たしている必要があります。

注意 SA メッセージの任意のフィルタリングを実行すると、ダウンストリーム MSDP ピアで正当なアクティブ ソースの SA メッセージを受信できなくなることがあります。そのため、このタイプのフィルタを使用する場合は注意が必要です。通常、発信フィルタ リストは、プライベート アドレスを使用するソースなど、望ましくないソースを拒否するためだけに使用します。

MSDP での着信フィルタ リストの使用

デフォルトでは、MSDP 対応デバイスは MSDP ピアからそのデバイスに送信されたすべての SA メッセージを受信します。ただし、着信フィルタ リストを作成することによって、MSDP ピアからデバイスが受信する送信元情報を制御できます。

着信フィルタ リストを作成すると、デバイスがピアから受信する着信 SA メッセージを次のように制御できます。

- 指定した MSDP ピアからのすべての着信 SAメッセージをフィルタリングするには、指定した MSDP ピアから送信されたすべての SAメッセージを無視するようにデバイスを設定します。
- 指定したピアからの着信 SA メッセージのサブセットを拡張アクセス リストに定義された（S, G）ペアに基づいてフィルタリングするには、拡張アクセス リストに定義された（S, G）ペアに一致する MSDP ピアからの SA メッセージだけを受信するようにデバイスを設定します。MSDP ピアからのその他のすべての着信 SA メッセージは無視されます。
- 指定したピアからの着信 SA 要求メッセージのサブセットをルート マップに定義された一致基準に基づいてフィルタリングするには、ルート マップに指定された基準に一致する SA メッセージだけを受信するようにデバイスを設定します。MSDP ピアからのその他のすべての着信 SA メッセージは無視されます。
- 指定したピアからの着信 SA メッセージのサブセットを拡張アクセス リストに定義された（S,G）ペアと、ルートマップに定義された基準の両方に基づいてフィルタリングするには、拡張アクセス リストに定義された（S, G）ペアと、ルート マップに定義された基準の両方に一致する着信 SA メッセージだけを受信するようにデバイスを設定します。MSDP ピアからのその他のすべての着信 SA メッセージは無視されます。
- 指定したピアからの着信 SA メッセージのサブセットを SA メッセージに含まれているアナウンス側 RP アドレスに基づいてフィルタリングするには、SA メッセージがすでに1 つ以上の MSDP ピア全体に送信されている可能性がある場合でも、それらの発信元に基づいて着信 SA メッセージをフィルタリングするようにデバイスを設定します。
- 拡張アクセス リスト、ルート マップ、および RP アクセス リストまたは RP ルート マップのいずれかを含む着信フィルタ リストを設定できます。この場合、MSDP ピアで着信 SA メッセージを受信するにはすべての条件を満たしている必要があります。

⚠

注意 SA メッセージの任意のフィルタリングを実行すると、ダウンストリーム MSDP ピアで正当なアクティブソースの SA メッセージを受信できなくなることがあります。そのため、このタイプのフィルタを使用する場合は注意が必要です。通常、着信フィルタ リストは、プライベート アドレスを使用するソースなど、望ましくないソースを拒否するためだけに使用されます。

MSDP の TTL しきい値

存続可能時間（TTL）値を使用して、ドロップされる前にパケットが取得できるホップの数を制限できます。特定の MSDP ピアに送信された、データがカプセル化された SA メッセージの TTL を指定するには、**ip multicast ttl-threshold** コマンドを使用します。デフォルトでは、パケットの TTL 値が 0（標準 TTL 動作）より大きい場合は、SA メッセージのマルチキャスト データ パケットは MSDP ピアに送信されます。

一般に、TTL しきい値の問題は、SA メッセージ内でソースの初期マルチキャスト パケットがカプセル化されることによって発生することがあります。マルチキャストパケットはユニキャスト SA メッセージ内部でカプセル化されるため（TTL は 255）、SA メッセージが MSDP ピアに送信されるときに TTL は減少しません。さらに、マルチキャスト トラフィックおよびユニキャスト トラフィックは MSDP ピア、したがってリモート PIM-SM ドメインへのまったく異なるパスに従うため、SA メッセージが通過するホップの総数は、通常のマルチキャスト パケットとは大きく異なります。その結果、カプセル化されたパケットは TTL しきい値に違反することになります。この問題を解決するには、**ip multicast ttl-threshold** コマンドを使用して、特定の MSDP ピアに送信された SA メッセージにカプセル化されているマルチキャストパケットに関連付けられた TTL しきい値を設定します。**ip msdp ttl-threshold** コマンドを使用すると、IP ヘッダーの TTL が *ttl-value* 引数に指定されている TTL 値未満であるマルチキャスト パケットが、ピアに送信される SA メッセージにカプセル化されないようにすることができます。

MSDP メッセージ タイプ

MSDP メッセージには 4 つの基本タイプがあり、それぞれが固有の Type、Length、および Value（TLV）データ フォーマットでエンコードされています。

SA メッセージ

SA メッセージを使用して、ドメイン内のアクティブなソースをアドバタイズします。また、これらの SA メッセージには送信元によって送信された最初のマルチキャスト データ パケットが含まれていることがあります。

SA メッセージには、発信元 RP の IPアドレスと、アドバタイズされる 1 つ以上の（S, G）ペアが含まれています。また、SA メッセージにカプセル化されたデータ パケットが含まれていることがあります。

SA 要求メッセージ

SA要求メッセージを使用して、特定のグループにアクティブなソースのリストを要求します。これらのメッセージは、SA キャッシュにアクティブな（S,G）ペアのリストを保持する MSDP SA キャッシュに送信されます。グループ内のすべてのアクティブなソースが発信元の RP によって再アドバタイズされるまで待つ代わりに、SA 要求メッセージを使用してアクティブなソースのリストを要求すると、加入遅延を短縮できます。

SA 応答メッセージ

SA 応答メッセージは SA 要求メッセージに応答する MSDP ピアによって送信されます。SA 応答メッセージには、発信元の RP の IP アドレスと、キャッシュに保存されている発信元 RP のドメイン内のアクティブなソースの 1 つ以上の（S, G）ペアが含まれています。

キープアライブ メッセージ

キープアライブ メッセージは 60 秒ごとに送信され、MSDP セッションをアクティブに保ちます。キープアライブ メッセージまたは SA メッセージを 75 秒間受信しなかった場合、MSDP セッションがリセットされます。

MSDP のデフォルト設定

MSDP はイネーブルになっていません。デフォルトの MSDP ピアはありません。

MSDP の設定方法

デフォルトの MSDP ピアの設定

始める前に

MSDP ピアを設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip msdp default-peer** *ip-address* | *name* [**prefix-list** *list*] 例：	すべての MSDP SA メッセージの受信元となるデフォルト ピアを定義します。

	コマンドまたはアクション	目的
	`Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a`	• *ip-address* | *name* には、MSDP デフォルト ピアの IP アドレスまたはドメイン ネーム システム（DNS）サーバー名を入力します。 • （任意） **prefix-list** *list* を指定する場合は、リスト内のプレフィックス専用のデフォルトピアとなるピアを指定するリスト名を入力します。プレフィックス リストがそれぞれ関連付けられている場合は、複数のアクティブなデフォルト ピアを設定できます。 **prefix-list** キーワードが指定された **ip msdp default-peer** コマンドを複数入力すると、複数の RP プレフィックスに対してすべてのデフォルトピアが同時に使用されます。この構文は通常、スタブ サイト クラウドに接続されたサービス プロバイダ クラウドで使用されます。 **prefix-list** キーワードを指定せずに **ip msdp default-peer** コマンドを複数入力すると、単一のアクティブピアですべての SA メッセージが受信されます。このピアに障害がある場合は、次の設定済みデフォルト ピアですべての SA メッセージが受信されます。この構文は通常、スタブ サイトで使用されます。
ステップ 4	**ip prefix-list** *name* [**description** *string*] | **seq** *number* {**permit** | **deny**} *network length* 例： `Router(config)# prefix-list site-a seq 3 permit 12 network length 128`	（任意）ステップ 2 で指定された名前を使用し、プレフィックス リストを作成します。 • （任意）**description** *string* を指定する場合は、このプレフィックスリストを説明する 80 文字以下のテキストを入力します。 • **seq** *number* には、エントリのシーケンス番号を入力します。指定できる範囲は 1 ～ 4294967294 です。 • **deny** キーワードを指定すると、条件が一致した場合にアクセスが拒否されます。 • **permit** キーワードを指定すると、条件が一致した場合にアクセスが許可されます。 • *network length* には、許可または拒否されているネットワークの番号およびネットワーク マスク長（ビット単位）を指定します。

	コマンドまたはアクション	目的
ステップ 5	**ip msdp description** {*peer-name* | *peer-address*} *text* 例： `Router(config)# ip msdp description peer-name site-b`	（任意）設定内で、または **show** コマンド出力内で簡単に識別できるように、指定されたピアの説明を設定します。 デフォルトでは、MSDP ピアに説明は関連付けられていません。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

SA ステートのキャッシング

メモリを消費して送信元情報の遅延を短縮する場合は、SA メッセージをキャッシュに格納するようにデバイスを設定できます。送信元とグループのペアのキャッシングをイネーブルにするには、次の手順を実行します。

送信元とグループのペアのキャッシングをイネーブルにするには、次の手順に従います。

手順

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**ip msdp cache-sa-state** [**list** *access-list-number*] 例： スイッチ(config)# **ip msdp cache-sa-state 100**	送信元とグループのペアのキャッシングをイネーブルにします（SA ステートを作成します）。アクセス リストを通過したこれらのペアがキャッシュに格納されます。 **list** *access-list-number* の範囲は 100 ～ 199 です。 （注） このコマンドの代わりに、**ip msdp sa-reques** グローバル コンフィギュレーション コマンドを使用できます。この代替コマンドを使用すると、グループの新しいメンバがアクティブになった場合に、SA 要求メッセージがデバイスから MSDP ピアに送信されます。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* 例： スイッチ(config)# **access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255**	IP 拡張アクセス リストを作成します。必要な回数だけこのコマンドを繰り返します。 • *access-list-number* の範囲は 100 ～ 199 です。ステップ 2 で作成した番号と同じ値を入力します。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *protocol* には、プロトコル名として **ip** を入力します。 • *source* には、パケットの送信元であるネットワークまたはホストの番号を入力します。 • *source-wildcard* には、送信元に適用するワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 • *destination* には、パケットの送信先であるネットワークまたはホストの番号を入力します。 • *destination-wildcard* には、宛先に適用するワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。

	コマンドまたはアクション	目的
		アクセスリストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MSDP ピアからの送信元情報の要求

グループへの送信元である接続 PIM SM ドメイン内のアクティブなマルチキャスト送信元を、グループの新しいメンバーが学習する必要がある場合は、新しいメンバーがグループに加入したときに、指定された MSDP ピアに SA 要求メッセージがデバイスから送信されるようにこのタスクを実行します。ピアは SA キャッシュ内の情報に応答します。ピアにキャッシュが設定されていない場合、このコマンドを実行しても何も起こりません。この機能を設定すると加入遅延は短縮されますが、メモリが消費されます。

新しいメンバーがグループに加入し、マルチキャストトラフィックを受信する必要が生じた場合、MSDP ピアに SA 要求メッセージを送信するようにデバイスを設定するには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip msdp sa-request** {*ip-address* | *name*} 例： `スイッチ(config)# ip msdp sa-request 171.69.1.1`	指定された MSDP ピアに SA 要求メッセージを送信するようにデバイスを設定します。 *ip-address* | *name* を指定する場合は、グループの新しいメンバーがアクティブになるときにローカルデバイスの SA メッセージの要求元になる MSDP ピアの IP アドレス、または名前を入力します。 SA メッセージを送信する必要がある MSDP ピアごとに、このコマンドを繰り返します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

スイッチから発信される送信元情報の制御

デバイスから発信されるマルチキャスト送信元情報を制御できます。

- アドバタイズ対象の送信元（送信元ベース）
- 送信元情報のレシーバー（要求元認識ベース）

詳細については、送信元の再配信 （304 ページ）およびSA 要求メッセージのフィルタリング（306 ページ）を参照してください。

送信元の再配信

SA メッセージは、送信元が登録されている RP で発信されます。デフォルトでは、RP に登録されているすべての送信元がアドバタイズされます。送信元が登録されている場合は、RP に *A* フラグが設定されています。このフラグは、フィルタリングされる場合を除き、送信元が SA に格納されてアドバタイズされることを意味します。

アドバタイズされる登録済みの送信元をさらに制限するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*] 例： `スイッチ(config)# ip msdp redistribute list 21`	SA メッセージに格納されてアドバタイズされる、マルチキャスト ルーティング テーブル内の（S, G）エントリを設定します。 デフォルトでは、ローカル ドメイン内の送信元だけがアドバタイズされます。 • （任意） **list** *access-list-name*：IP 標準または IP 拡張アクセスリストの名前または番号を入力します。標準アクセス リストの範囲は 1 ～ 99、拡張アクセス リストの範囲は 100 ～ 199 です。アクセスリストによって、アドバタイズされるローカルな送信元、および送信されるグループが制御されます。 • （任意） **asn** *aspath-access-list-number*：1 ～ 199 の範囲の IP 標準または IP 拡張アクセスリスト番号を入力します。このアクセスリスト番号は、**ip as-path access-list** コマンドでも設定する必要があります。 • （任意） **route-map** *map*：1 ～ 199 の範囲の IP 標準または IP 拡張アクセスリスト番号を入力します。このアクセスリスト番号は、**ip as-path**

	コマンドまたはアクション	目的
		access-list コマンドでも設定する必要があります。 アクセスリストまたは自律システムパスアクセスリストに従って、デバイスが（S, G）ペアをアドバタイズします。
ステップ 4	次のいずれかを使用します。 • `access-list`*access-list-number* {`deny`|`permit`} *source* [*source-wildcard*] • `access-list`*access-list-number* {`deny`|`permit`} *protocol source source-wildcard destination destination-wildcard* **例**： `スイッチ(config)#` **`access list 21 permit 194.1.22.0`** または `スイッチ(config)#` **`access list 21 permit ip 194.1.22.0 10.1.1.10 194.3.44.0 10.1.1.10`**	IP標準アクセスリストを作成します。必要な回数だけこのコマンドを繰り返します。 または IP拡張アクセスリストを作成します。必要な回数だけこのコマンドを繰り返します。 • *access-list-number*：ステップ 2 で作成した同じ番号を入力します。標準アクセスリストの範囲は 1 ～ 99、拡張アクセス リストの範囲は 100 ～ 199 です。 • **deny**：条件に合致している場合、アクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *protocol*：プロトコル名として **ip** を入力します。 • *source*：パケットの送信元であるネットワークまたはホストの番号を入力します。 • *source-wildcard*：送信元に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 • *destination*：パケットの宛先であるネットワークまたはホストの番号を入力します。 • *destination-wildcard*：宛先に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。
ステップ 5	**end** **例**：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SA 要求メッセージのフィルタリング

デフォルトでは、SA情報をキャッシングしているデバイスだけが、SA要求に応答できます。このようなデバイスでは、デフォルトでMSDP ピアからのすべてのSA 要求メッセージが採用され、アクティブな送信元の IP アドレスが取得されます。

ただし、MSDP ピアからの SA 要求をすべて無視するように、デバイスを設定できます。標準アクセス リストに記述されたグループのピアからのSA要求メッセージだけを採用することもできます。アクセス リスト内のグループが指定された場合は、そのグループのピアからの SA 要求メッセージが受信されます。他のグループのピアからの他のメッセージは、すべて無視されます。

デフォルト設定に戻すには、 **no ip msdp filter-sa-request** {*ip-address*| *name*} グローバル コンフィギュレーション コマンドを使用します。

これらのオプションのいずれかを設定するには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
ステップ 3	次のいずれかを使用します。 • `ip msdp filter-sa-request` {*ip-addressname*} • `ip msdp filter-sa-request` {*ip-addressname*} `list` *access-list-number* 例： `スイッチ(config)# ip msdp filter sa-request 171.69.2.2`	指定された MSDP ピアからの SA 要求メッセージをすべてフィルタリングします。 または 標準アクセスリストを通過したグループに対して、指定された MSDP ピアからの SA 要求メッセージをフィルタリングします。アクセスリストには、複数のグループ アドレスが記述されています。 access-list-number の範囲は 1 ～ 99 です。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： `スイッチ(config)# access-list 1 permit 192.4.22.0 0.0.0.255`	IP標準アクセスリストを作成します。必要な回数だけこのコマンドを繰り返します。 • *access-list-number* の範囲は 1 ～ 99 です。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *source* には、パケットの送信元であるネットワークまたはホストの番号を入力します。 •（任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

スイッチで転送される送信元情報の制御

デフォルトでは、デバイスで受信されたすべての SA メッセージが、すべての MSDP ピアに転送されます。ただし、フィルタリングするか、または存続可能時間（TTL）値を設定し、発信メッセージがピアに転送されないようにできます。

フィルタの使用法

フィルタを作成すると、次のいずれかの処理を実行できます。

- すべての送信元とグループのペアのフィルタリング
- 特定の送信元とグループのペアだけが通過するように、IP 拡張アクセス リストを指定
- ルート マップの一致条件に基づくフィルタリング

フィルタを適用するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかを使用します。 • **ip msdp sa-filter out** {*ip-address name*} • **ip msdp sa-filter out** {*ip-address name*} list *access-list-number* • **ip msdp sa-filter out**	• 指定された MSDP ピアへの SA メッセージをフィルタリングします。 • 指定したピアに対する IP 拡張アクセス リストを通過した SA メッセージのみを渡します。拡張アクセスリスト番号の範囲は 100 ～ 199 です。 **list** と **route-map** の両方のキーワードを使用すると、すべての条件に一致しなければ、発信 SA

	コマンドまたはアクション	目的
	{*ip-address name*} `route-map` *map-tag* 例： `スイッチ(config)# ip msdp sa-filter out switch.cisco.com` または `スイッチ(config)# ip msdp sa-filter out list 100` または `スイッチ(config)# ip msdp sa-filter out switch.cisco.com route-map 22`	メッセージ内のいずれの（S,G）ペアも通過できません。 • 指定された MSDP ピアへのルート マップ *map-tag* で一致基準を満たす SA メッセージのみを渡します。 すべての一致基準に当てはまる場合、ルート マップの **permit** がフィルタを通してルートを通過します。**deny** はルートをフィルタ処理します。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* 例： `スイッチ(config)# access list 100 permit ip 194.1.22.0 10.1.1.10 194.3.44.0 10.1.1.10`	（任意）IP 拡張アクセス リストを作成します。必要な回数だけこのコマンドを繰り返します。 • *access-list-number* には、ステップ 2 で指定した番号を入力します。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *protocol* には、プロトコル名として **ip** を入力します。 • *source* には、パケットの送信元であるネットワークまたはホストの番号を入力します。 • *source-wildcard* には、送信元に適用するワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 • *destination* には、パケットの送信先であるネットワークまたはホストの番号を入力します。 • *destination-wildcard* には、宛先に適用するワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SA メッセージに格納されて送信されるマルチキャスト データの TTL による制限

TTL 値を使用して、各送信元の最初の SA メッセージにカプセル化されるデータを制御できます。IP ヘッダー TTL 値が *ttl* 引数以上であるマルチキャスト パケットだけが、指定された MSDP ピアに送信されます。たとえば、内部トラフィックの TTL 値を 8 に制限できます。他のグループを外部に送信する場合は、これらのパケットの TTL を 8 より大きく設定して送信する必要があります。

TTL しきい値を確立するには、次の手順に従います。

手順

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

	コマンドまたはアクション	目的
ステップ 3	**ip msdp ttl-threshold** {*ip-address* | *name*} *ttl* 例： `スイッチ(config)# ip msdp ttl-threshold switch.cisco.com 0`	指定された MSDP ピア宛ての最初の SA メッセージにカプセル化されるマルチキャスト データを制限します。 • *ip-address*|*name* には、TTL の制限が適用される MSDP ピアの IP アドレスまたは名前を入力します。 • *ttl* には、TTL 値を入力します。デフォルトは 0 です。この場合、すべてのマルチキャスト データ パケットは、TTL がなくなるまでピアに転送されます。指定できる範囲は 0 ～ 255 です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

スイッチで受信される送信元情報の制御

デフォルトでは、デバイスは、MSDP の RPF ピアによって送信されたすべての SA メッセージを受信します。ただし、着信 SA メッセージをフィルタリングし、MSDP ピアから受信する送信元情報を制御できます。つまり、特定の着信 SA メッセージを受信しないようにデバイスを設定できます。

次のいずれかの処理を実行できます。

- MSDP ピアからのすべての着信 SA メッセージのフィルタリング
- 特定の送信元とグループのペアが通過するように、IP 拡張アクセス リストを指定
- ルート マップの一致条件に基づくフィルタリング

フィルタを適用するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかを使用します。 • **ip msdp sa-filter in** {*ip-address name*} • **ip msdp sa-filter in** {*ip-address name*} list *access-list-number* • **ip msdp sa-filter in** {*ip-address name*} route-map *map-tag* 例： スイッチ(config)# **ip msdp sa-filter in switch.cisco.com** または スイッチ(config)# **ip msdp sa-filter in list 100** または スイッチ(config)# **ip msdp sa-filter in switch.cisco.com route-map 22**	• 指定された MSDP ピアへの SA メッセージをフィルタリングします。 • IP 拡張アクセス リストを通過する、指定されたピアからの SA メッセージのみを通過させます。拡張アクセス リスト *access-list-number* の範囲は 100 ～ 199 です。 **list** と **route-map** の両方のキーワードを使用すると、すべての条件に一致しなければ、発信 SA メッセージ内のいずれの（S,G）ペアも通過できません。 • ルート マップ *map-tag* 内の一致条件を満たす、指定された MSDP ピアからの SA メッセージのみを通過させます。 すべての一致基準に当てはまる場合、ルート マップの **permit** がフィルタを通してルートを通過します。**deny** はルートをフィルタ処理します。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* 例： スイッチ(config)# **access list 100 permit ip**	（任意）IP 拡張アクセス リストを作成します。必要な回数だけこのコマンドを繰り返します。 • *Access-list-number* には、ステップ 2 で指定した番号を入力します。

	コマンドまたはアクション	目的
	194.1.22.0 10.1.1.10 194.3.44.0 10.1.1.10	• **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *protocol* には、プロトコル名として **ip** を入力します。 • *source* には、パケットの送信元であるネットワークまたはホストの番号を入力します。 • *source-wildcard* には、送信元に適用するワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 • *destination* には、パケットの送信先であるネットワークまたはホストの番号を入力します。 • *destination-wildcard* には、宛先に適用するワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。
ステップ 5	**end** **例**： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** **例**： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** **例**： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MSDP メッシュ グループの設定

MSDP メッシュ グループを設定するには、次の任意の作業を実行します。

（注） デバイスごとに複数のメッシュ グループを設定できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip msdp mesh-group** *mesh-name* {*peer-address* | *peer-name*}
4. MSDP ピアをメッシュ グループのメンバとして追加するには、ステップ 3 を繰り返します。
5. **exit**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip msdp mesh-group** *mesh-name* {*peer-address* | *peer-name*} 例： `Device(config)# ip msdp mesh-group peermesh`	MSDP メッシュ グループを設定し、MSDP ピアがそのメッシュ グループに属することを指定します。 （注） メッシュ グループに参加しているデバイス上のすべての MSDP ピアは、そのグループ内の他のすべての MSDP ピアと完全にメッシュ構造になっている必要があります。各デバイスの各 MSDP ピアは、**ip msdp peer** コマンドを使用して、ピアとして設定する必要があります、また、**ip msdp mesh-group** コマンドを使用して、そのメッシュグループのメンバとしても設定する必要があります。

	コマンドまたはアクション	目的
ステップ 4	MSDP ピアをメッシュ グループのメンバとして追加するには、ステップ 3 を繰り返します。	--
ステップ 5	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `Device# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

MSDP ピアのシャットダウン

始める前に

MSDP が動作していて、MSDP ピアを設定する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip msdp shutdown** {*peer-name* | *peer-address*}
4. 別の MSDP ピアをシャットダウンするには、ステップ 3 を繰り返します。
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたらパスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip msdp shutdown** {*peer-name* | *peer-address*} 例： `Device(config)# ip msdp shutdown 192.168.1.3`	指定された MSDP ピアを管理シャットダウンします。
ステップ 4	別の MSDP ピアをシャットダウンするには、ステップ 3 を繰り返します。	--
ステップ 5	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `Device# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

境界 PIM デンス モード領域の MSDP への包含

デンスモード（DM）領域と PIM スパースモード（SM）領域の境界となるデバイスに MSDP を設定します。デフォルトでは、DM 領域のアクティブな送信元は MSDP に加入しません。

（注） **ip msdp border sa-address** グローバル コンフィギュレーション コマンドの使用は推奨できません。DM ドメイン内の送信元が SM ドメイン内の RP にプロキシ登録されるように SM ドメイン内の境界ルータを設定し、標準 MSDP 手順でこれらの送信元をアドバタイズするように SM ドメインを設定してください。

ip msdp originator-id グローバル コンフィギュレーション コマンドを実行すると、RP アドレスとして使用されるインターフェイスも識別されます。**ip msdp border sa-address** および **ip msdp**

originator-id グローバル コンフィギュレーション コマンドの両方が設定されている場合、**ip msdp originator-id** コマンドから取得されたアドレスが RP アドレスを指定します。

DM 領域でアクティブな送信元の SA メッセージを MSDP ピアに送信するように境界ルータを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip msdp border sa-address** *interface-id* 例： `スイッチ(config)# ip msdp border sa-address 0/1`	DM 領域内のアクティブな送信元に関する SA メッセージを送信するように、DM 領域と SM 領域の境界スイッチを設定します。 *interface-id* には、SA メッセージ内の RP アドレスとして使用される、IP アドレスの配信元となるインターフェイスを指定します。 インターフェイスの IP アドレスは、SA メッセージ内の RP フィールド [Originator-ID] の値として使用されます。
ステップ 4	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*] 例： `スイッチ(config)# ip msdp redistribute list 100`	SA メッセージに格納されてアドバタイズされる、マルチキャスト ルーティング テーブル内の（S, G）エントリを設定します。 詳細については、送信元の再配信 （304 ページ）を参照してください。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例：	入力を確認します。

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

RP アドレス以外の発信元アドレスの設定

SA メッセージを発信する MSDP スピーカーがそのインターフェイスの IP アドレスを SA メッセージ内の RP アドレスとして使用できるようにするには、次の任意の作業を実行します。

また、次のいずれかの理由により、発信元 ID を変更できます。

- Anycast RP の MSDP メッシュ グループに複数のデバイスを設定する場合。
- デバイスが PIM-SM ドメインと PIM-DM ドメインの境界にある場合。デバイスが PIM-SM ドメインと PIM-DM ドメインの境界にあり、PIM-DM ドメイン内のアクティブなソースをアドバタイズする場合は、SA メッセージ内の RP アドレスが発信元デバイスのインターフェイスのアドレスになるように設定します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip msdp originator-id**
4. **exit**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**ip msdp originator-id** 例： `スイッチ(config)# ip msdp originator-id ethernet 1`	発信元デバイスのインターフェイスのアドレスとなるように、SA メッセージ内の RP アドレスを設定します。
ステップ 4	**exit** 例： `スイッチ(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

MSDP のモニタリングおよびメンテナンス

MSDP のモニタリング

MSDP の SA メッセージ、ピア、ステート、およびピアのステータスをモニタリングするには、次の任意の作業を実行します。

手順の概要

1. **enable**
2. **debug ip msdp** [*peer-address* | *peer-name*] [**detail**] [**routes**]
3. **debug ip msdp resets**
4. **show ip msdp count** [*as-number*]
5. **show ip msdp peer** [*peer-address* | *peer-name*]
6. **show ip msdp sa-cache** [*group-address* | *source-address* | *group-name* | *source-name*] [*as-number*]
7. **show ip msdp summary**

手順の詳細

ステップ 1 **enable**

例：

```
Device# enable
```

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

ステップ 2 **debug ip msdp** [*peer-address* | *peer-name*] [**detail**] [**routes**]

このコマンドを使用して、MSDP アクティビティをデバッグします。

オプションの *peer-address* または *peer-name* 引数を使用して、デバッグ イベントをログに記録するピアを指定します。

次に、**debug ip msdp** コマンドの出力例を示します。

例：

```
Device# debug ip msdp
MSDP debugging is on
Device#
MSDP: 224.150.44.254: Received 1388-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 1388, ec: 115, RP: 172.31.3.92
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.92, used EMBGP peer
MSDP: 224.150.44.250: Forward 1388-byte SA to peer
MSDP: 224.150.44.254: Received 1028-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 1028, ec: 85, RP: 172.31.3.92
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.92, used EMBGP peer
MSDP: 224.150.44.250: Forward 1028-byte SA to peer
MSDP: 224.150.44.254: Received 1388-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 1388, ec: 115, RP: 172.31.3.111
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.111, used EMBGP peer
MSDP: 224.150.44.250: Forward 1388-byte SA to peer
MSDP: 224.150.44.250: Received 56-byte message from peer
MSDP: 224.150.44.250: SA TLV, len: 56, ec: 4, RP: 192.168.76.241
MSDP: 224.150.44.250: Peer RPF check passed for 192.168.76.241, used EMBGP peer
MSDP: 224.150.44.254: Forward 56-byte SA to peer
MSDP: 224.150.44.254: Received 116-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 116, ec: 9, RP: 172.31.3.111
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.111, used EMBGP peer
MSDP: 224.150.44.250: Forward 116-byte SA to peer
MSDP: 224.150.44.254: Received 32-byte message from peer
MSDP: 224.150.44.254: SA TLV, len: 32, ec: 2, RP: 172.31.3.78
MSDP: 224.150.44.254: Peer RPF check passed for 172.31.3.78, used EMBGP peer
MSDP: 224.150.44.250: Forward 32-byte SA to peer
```

ステップ 3 **debug ip msdp resets**

このコマンドを使用して、MSDP ピアのリセット理由をデバッグします。

例：

```
Device# debug ip msdp resets
```

ステップ 4 **show ip msdp count** [*as-number*]

このコマンドを使用して、MSDP SA メッセージ内で発信したソースおよびグループの数、および SA キャッシュ内の MSDP ピアからの SA メッセージの数を表示します。**ip msdp cache-sa-state** コマンドは、このコマンドによって出力が生成されるように設定する必要があります。

次に、**show ip msdp count** コマンドの出力例を示します。

例：

```
Device# show ip msdp count
SA State per Peer Counters, <Peer>: <# SA learned>
    192.168.4.4: 8
SA State per ASN Counters, <asn>: <# sources>/<# groups>
    Total entries: 8
    ?: 8/8
```

ステップ 5 **show ip msdp peer** [*peer-address* | *peer-name*]

このコマンドを使用して、MSDP ピアに関する詳細情報を表示します。

オプションの *peer-address* 引数または *peer-name* 引数を使用して、特定のピアに関する情報を表示します。

次に、**show ip msdp peer** コマンドの出力例を示します。

例：

```
Device# show ip msdp peer 192.168.4.4
MSDP Peer 192.168.4.4 (?), AS 64512 (configured AS)
  Connection status:
    State: Up, Resets: 0, Connection source: Loopback0 (2.2.2.2)
    Uptime(Downtime): 00:07:55, Messages sent/received: 8/18
    Output messages discarded: 0
    Connection and counters cleared 00:08:55 ago
  SA Filtering:
    Input (S,G) filter: none, route-map: none
    Input RP filter: none, route-map: none
    Output (S,G) filter: none, route-map: none
    Output RP filter: none, route-map: none
  SA-Requests:
    Input filter: none
  Peer ttl threshold: 0
  SAs learned from this peer: 8
  Input queue size: 0, Output queue size: 0
  MD5 signature protection on MSDP TCP connection: not enabled
```

ステップ 6 **show ip msdp sa-cache** [*group-address* | *source-address* | *group-name* | *source-name*] [*as-number*]

このコマンドを使用して、MSDP ピアから学習した (S, G) ステートを表示します。

次に、**show ip msdp sa-cache** コマンドの出力例を示します。

例：

```
Device# show ip msdp sa-cache
MSDP Source-Active Cache - 8 entries
(10.44.44.5, 239.232.1.0), RP 192.168.4.4, BGP/AS 64512, 00:01:20/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.1), RP 192.168.4.4, BGP/AS 64512, 00:01:20/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.2), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.3), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.4), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
```

```
(10.44.44.5, 239.232.1.5), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.6), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
(10.44.44.5, 239.232.1.7), RP 192.168.4.4, BGP/AS 64512, 00:01:19/00:05:32, Peer 192.168.4.4
```

ステップ 7 **show ip msdp summary**

このコマンドを使用して、MSDP ピアのステータスを表示します。

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

例：

```
Device# show ip msdp summary
MSDP Peer Status Summary
Peer Address     AS    State    Uptime/  Reset SA    Peer Name
                                Downtime Count Count
192.168.4.4      4     Up       00:08:05 0     8     ?
```

MSDP 接続統計情報および SA キャッシュ エントリの消去

MSDP 接続、統計情報または SA キャッシュ エントリを消去するには、次の任意の作業を実行します。

手順の概要

1. **enable**
2. **clear ip msdp peer** [*peer-address* | *peer-name*]
3. **clear ip msdp statistics** [*peer-address* | peer-name]
4. **clear ip msdp sa-cache** [*group-address*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** **例**： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**clear ip msdp peer** [*peer-address* | *peer-name*] **例**： `Device# clear ip msdp peer`	指定された MSDP ピアへの TCP 接続をクリアし、すべての MSDP メッセージ カウンタをリセットします。
ステップ 3	**clear ip msdp statistics** [*peer-address* | peer-name] **例**： `Device# clear ip msdp statistics`	指定された MSDP ピアの統計カウンタをクリアし、すべての MSDP メッセージ カウンタをリセットします。

	コマンドまたはアクション	目的
ステップ 4	**clear ip msdp sa-cache** [*group-address*] 例： `Device# clear ip msdp sa-cache`	SA キャッシュ エントリを消去します。 • **clear ip msdp sa-cache** コマンドにオプションの *group-address* 引数または *source-address* 引数を指定した場合、すべての SA キャッシュエントリが消去されます。 • 特定のグループに関連付けられたすべての SA キャッシュ エントリを消去するには、オプションの *group-address* 引数を使用します。

MSDP の設定例

デフォルト MSDP ピアの設定：例

次に、ルータ A およびルータ C の部分的な設定の例を示します 。これらの ISP にはそれぞれに複数のカスタマー（カスタマーと同様）がおり、 デフォルトのピアリング（BGP または MBGP なし）を使用しています。この場合、両方の ISP で類似した設定となります。つまり、両方の ISP では、対応するプレフィックス リストで SA が許可されている場合、デフォルト ピアからの SA だけが受信されます。

ルータ A

```
Router(config)# ip msdp default-peer 10.1.1.1
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

ルータ C

```
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

SA ステートのキャッシング：例

次に、グループ 224.2.0.0/16 への送信元である 171.69.0.0/16 のすべての送信元のキャッシュ ステートをイネーブルにする例を示します。

```
スイッチ(config)# ip msdp cache-sa-state 100
スイッチ(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255
```

MSDP ピアからの送信元情報の要求：例

次に、171.69.1.1 の MSDP ピアに SA 要求メッセージを送信するように、スイッチを設定する例を示します。

```
スイッチ(config)# ip msdp sa-request 171.69.1.1
```

スイッチから発信される送信元情報の制御：例

次に、171.69.2.2 の MSDP ピアからの SA 要求メッセージをフィルタリングするように、スイッチを設定する例を示します。ネットワーク 192.4.22.0 の送信元からの SA 要求メッセージはアクセス リスト 1 に合格して、受信されます。その他のすべてのメッセージは無視されます。

```
スイッチ(config)# ip msdp filter sa-request 171.69.2.2 list 1
スイッチ(config)# access-list 1 permit 192.4.22.0 0.0.0.255
```

スイッチから転送される送信元情報の制御：例

次に、アクセス リスト 100 を通過する（S,G）ペアだけが SA メッセージに格納され、*switch.cisco.com* という名前のピアに転送されるように設定する例を示します。

```
スイッチ(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet1/0/1
スイッチ(config)# ip msdp sa-filter out switch.cisco.com list 100
スイッチ(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.20 0 0.0.255.255
```

スイッチで受信される送信元情報の制御：例

次に、*switch.cisco.com* という名前のピアからのすべての SA メッセージをフィルタリングする例を示します。

```
スイッチ(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet1/0/1
スイッチ(config)# ip msdp sa-filter in switch.cisco.com
```

例：MSDP メッシュ グループの設定

次に、3 台のデバイスを MSDP メッシュ グループのフル メッシュ メンバになるように設定する例を示します。

デバイス A の設定

```
ip msdp peer 10.2.2.2
ip msdp peer 10.3.3.3
```

```
ip msdp mesh-group test-mesh-group 10.2.2.2
ip msdp mesh-group test-mesh-group 10.3.3.3
```

デバイス B の設定

```
ip msdp peer 10.1.1.1
ip msdp peer 10.3.3.3
ip msdp mesh-group test-mesh-group 10.1.1.1
ip msdp mesh-group test-mesh-group 10.3.3.3
```

デバイス C の設定

```
ip msdp peer 10.1.1.1
ip msdp peer 10.2.2.2
ip msdp mesh-group test-mesh-group 10.1.1.1
ip msdp mesh-group test-mesh-group 10.2.2.2
```

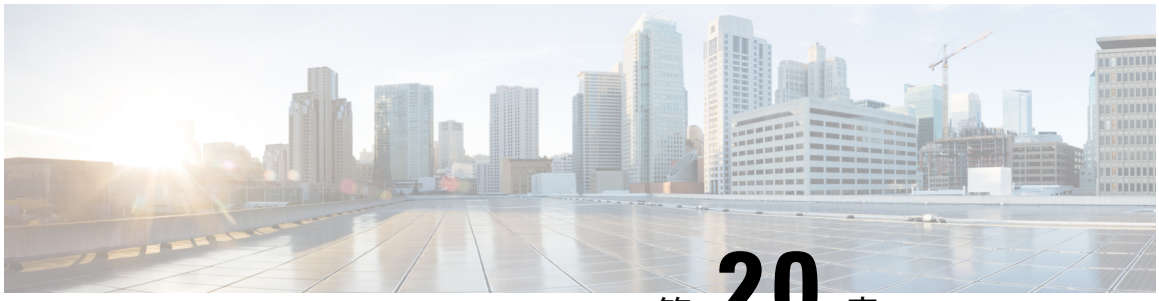
MSDP ピアからの送信元情報の要求：例

次に、171.69.1.1 の MSDP ピアに SA 要求メッセージを送信するように、スイッチを設定する例を示します。

```
スイッチ(config)# ip msdp sa-request 171.69.1.1
```

第 III 部

IPv6



第 20 章

MLD スヌーピングの設定

このモジュールには、MLD スヌーピングの設定の詳細が含まれています。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

IPv6 MLD スヌーピングの設定に関する情報

スイッチ上で Multicast Listener Discovery（MLD）スヌーピングを使用して、スイッチド ネットワーク内のクライアントおよびルータに IP Version 6（IPv6）マルチキャスト データを効率的に配信することができます。特に明記しない限り、スイッチという用語は、スタンドアロン スイッチを指します。

（注） この章で使用するコマンドの構文および使用方法の詳細については、このリリースに対応するコマンド リファレンスまたは手順に記載された Cisco IOS のマニュアルを参照してください。

MLD スヌーピングの概要

IP Version 4（IPv4）では、レイヤ 2 スイッチはインターネット グループ管理プロトコル（IGMP）スヌーピングを使用して、動的にレイヤ 2 インターフェイスを設定することにより、マルチキャスト トラフィックのフラッディングを抑制します。そのため、マルチキャスト トラフィックは IP マルチキャスト デバイスに対応付けられたインターフェイスにだけ転送されます。IPv6 では、MLD スヌーピングが同様の機能を実行します。MLD スヌーピングを使用すると、IPv6 マルチキャスト データは VLAN（仮想 LAN）内のすべてのポートにフラッディングされるのではなく、データを受信するポートのリストに選択的に転送されます。このリストは、IPv6 マルチキャスト制御パケットをスヌーピングすることにより構築されます。

MLD は IPv6 マルチキャスト ルータで使用されるプロトコルで、ルータに直接接続されたリンク上のマルチキャスト リスナー（IPv6 マルチキャスト パケットを受信するノード）の存在、および隣接ノードを対象とするマルチキャスト パケットを検出します。MLD は IGMP から派生しています。MLD バージョン 1（MLDv1）は IGMPv2 と、MLD バージョン 2（MLDv2）は IGMPv3 とそれぞれ同等です。MLD は Internet Control Message Protocol バージョン 6（ICMPv6）のサブプロトコルです。MLD メッセージは ICMPv6 メッセージのサブセットで、IPv6 パケット内で先頭の Next Header 値 58 により識別されます。

スイッチは、次の 2 つのバージョンの MLD スヌーピングをサポートします。

- MLDv1 スヌーピング：MLDv1 制御パケットを検出し、IPv6 宛先マルチキャスト アドレスに基づいてトラフィックのブリッジングを設定します。
- MLDv2 基本スヌーピング（MBSS）：MLDv2 制御パケットを使用して、IPv6 宛先マルチキャスト アドレスに基づいてトラフィックの転送を設定します。

スイッチは MLDv1 プロトコル パケットと MLDv2 プロトコル パケットの両方でスヌーピングでき、IPv6 宛先マルチキャスト アドレスに基づいて IPv6 マルチキャスト データをブリッジングします。

（注） スイッチは、IPv6 送信元および宛先マルチキャスト アドレスベースの転送を設定する MLDv2 拡張スヌーピングをサポートしません。

MLD スヌーピングは、グローバルまたは VLAN 単位でイネーブルまたはディセーブルに設定できます。MLD スヌーピングがイネーブルの場合、VLAN 単位の IPv6 マルチキャスト アドレス テーブルはソフトウェアおよびハードウェアで構築されます。その後、スイッチはハードウェアで IPv6 マルチキャストアドレスに基づくブリッジングを実行します。

MLD メッセージ

MLDv1 は、次の 3 種類のメッセージをサポートします。

- Listener Query：IGMPv2 クエリーと同等で、General Query または Mulicast-Address-Specific Query（MASQ）のいずれかになります。
- Multicast Listener Report：IGMPv2 レポートと同等です。

• Multicast Listener Done メッセージ：IGMPv2 Leave メッセージと同等です。

MLDv2 では、MLDv1 レポートおよび Done メッセージに加えて、MLDv2 クエリーおよび MLDv2 レポートもサポートします。

メッセージの送受信の結果生じるメッセージ タイマーおよびステート移行は、IGMPv2 メッセージの場合と同じです。リンクに対してローカルで有効な IPv6 送信元アドレスを持たない MLD メッセージは、MLD ルータおよび MLD スイッチで無視されます。

MLD クエリー

スイッチは MLD クエリーを送信し、IPv6 マルチキャスト アドレス データベースを構築し、MLD グループ固有クエリー、MLD グループおよび送信元固有クエリーを生成して、MLD Done メッセージに応答します。また、スイッチはレポート抑制、レポート プロキシング、即時脱退機能、およびスタティックな IPv6 マルチキャスト グループ アドレス設定もサポートします。

MLD スヌーピングがディセーブルの場合、すべての MLD クエリーが入力 VLAN でフラッディングされます。

MLD スヌーピングがイネーブルの場合、受信された MLD クエリーが入力 VLAN でフラッディングされ、クエリーのコピーは CPU に送信され、処理されます。MLD スヌーピングでは、受信されたクエリーから IPv6 マルチキャスト アドレス データベースを構築します。MLD スヌーピングは、マルチキャスト ルータ ポートを検出して、タイマーを維持し、レポート応答時間を設定します。また、VLAN のクエリア IP 送信元アドレス、VLAN 内のクエリア ポートを学習して、マルチキャスト アドレス エージングを維持します。

（注） IPv6 マルチキャスト ルータが Catalyst 6500 スイッチであり、拡張 VLAN（範囲 1006 ～ 4094）を使用する場合、Catalyst 2960、2960-S、2960-C、2960-X、または 2960-CX スイッチが VLAN 上でクエリを受信できるようにするため、IPv6 MLD スヌーピングを Catalyst 6500 スイッチの拡張 VLAN でイネーブルにする必要があります。標準範囲 VLAN（1 ～ 1005）の場合、IPv6 MLD スヌーピングを Catalyst 6500 スイッチの VLAN でイネーブルにする必要はありません。

グループが MLD スヌーピング データベースに存在する場合、スイッチは MLDv1 レポートを送信して、グループ固有のクエリーに応答します。このグループが不明の場合、グループ固有のクエリーは入力 VLAN にフラッディングされます。

ホストがマルチキャスト グループから脱退する場合、MLD Done メッセージ（IGMP Leave メッセージと同等）を送信できます。スイッチが MLDv1 Done メッセージを受信した際に、即時脱退がイネーブルでなければ、スイッチは メッセージを受信したポートに MASQ を送信して、ポートに接続する他のデバイスがマルチキャスト グループに残る必要があるかどうか判別します。

マルチキャスト クライアント エージングの堅牢性

クエリー数に基づいて、アドレスからのポート メンバーシップの削除を設定できます。1 つのアドレスに対するメンバーシップからポートが削除されるのは、設定された数のクエリーに関

してポート上のアドレスに対するレポートがない場合のみです。デフォルトの回数は 2 回です。

マルチキャスト ルータ検出

IGMP スヌーピングと同様に、MLD スヌーピングでは次の特性を持つマルチキャスト ルータ検出を行います。

- ユーザにより設定されたポートには、期限切れがありません。
- ダイナミックなポート学習は、MLDv1 スヌーピング クエリーおよび IPv6 PIMv2 パケットにより行われます。
- 複数のルータが同じレイヤ2インターフェイス上にある場合、MLD スヌーピングではポート上の単一のマルチキャスト ルータ（直前にルータ制御パケットを送信したルータ）を追跡します。
- マルチキャスト ルータ ポートのダイナミックなエージングは、デフォルト タイマーの 5 分に基づきます。ポート上で制御パケットが 5 分間受信されない場合、マルチキャスト ルータはルータのポート リストから削除されます。
- IPv6 マルチキャスト ルータ検出が実行されるのは、MLD スヌーピングがスイッチでイネーブルの場合のみです。
- 受信された IPv6 マルチキャスト ルータ制御パケットは、スイッチで MLD スヌーピングがイネーブルかどうかにかかわらず、常に入力 VLAN にフラッディングされます。
- 最初の IPv6 マルチキャスト ルータ ポートが検出された後は、不明の IPv6 マルチキャスト データは、検出されたルータ ポートに対してのみ転送されます（それまでは、すべての IPv6 マルチキャスト データは入力 VLAN にフラッディングされます）。

MLD レポート

MLDv1 join メッセージは、本質的には IGMPv2 と同じように処理されます。IPv6 マルチキャスト ルータが VLAN で検出されない場合は、レポートが処理されないか、またはスイッチから転送されません。IPv6 マルチキャスト ルータが検出され、MLDv1 レポートが受信されると、IPv6 マルチキャスト グループ アドレスが VLAN の MLD データベースに入力されます。その後、VLAN 内のグループに対するすべての IPv6 マルチキャスト トラフィックが、このアドレスを使用して転送されます。MLD スヌーピングがディセーブルの場合、レポートは入力 VLAN でフラッディングされます。

MLD スヌーピングがイネーブルの場合は、MLD レポート抑制（リスナー メッセージ抑制）は自動的にイネーブルになります。レポート抑制により、スイッチはグループで受信された最初の MLDv1 レポートを IPv6 マルチキャスト ルータに転送します。グループのそれ以降のレポートはルータに送信されません。MLD スヌーピングがディセーブルの場合は、レポート抑制がディセーブルになり、すべての MLDv1 レポートは入力 VLAN にフラッディングされます。

スイッチは、MLDv1 プロキシ レポーティングもサポートします。MLDv1 MASQ が受信されると、スイッチに他のポートのグループが存在する場合、およびクエリーを受信したポートと

アドレスの最後のメンバポートが異なる場合は、スイッチはクエリーを受信したアドレスに関する MLDv1 レポートで応答します。

MLD Done メッセージおよび即時脱退

即時脱退機能がイネーブルの場合にホストが MLDv1 Done メッセージ（IGMP Leave メッセージと同等）を送信すると、Done メッセージを受信したポートはグループからただちに削除されます。VLAN で即時脱退をイネーブルにする場合は（IGMP スヌーピングと同様に）、ポートに単一のホストが接続されている VLAN でのみこの機能を使用します。ポートがグループの最後のメンバである場合、グループも削除され、検出された IPv6 マルチキャスト ルータに脱退情報が転送されます。

VLAN で即時脱退がイネーブルでない場合に（1 つのポート上にグループのクライアントが複数ある場合）、Done メッセージがポートで受信されると、このポートで MASQ が生成されます。ユーザは、既存アドレスのポート メンバーシップが削除される時期を MASQ 数の観点から制御できます。アドレスに対するメンバーシップからポートが削除されるのは、設定された数のクエリーに関してポート上のアドレスに対する MLDv1 レポートがない場合です。

生成される MASQ 数は、 **ipv6 mld snooping last-listener-query count** グローバル コンフィギュレーション コマンドにより設定されます。デフォルトの回数は 2 回です。

MASQ は、Done メッセージが送信された IPv6 マルチキャスト アドレスに送信されます。スイッチの最大応答時間内に MASQ で指定された IPv6 マルチキャスト アドレスにレポートが送信されなければ、MASQ が送信されたポートは IPv6 マルチキャスト アドレス データベースから削除されます。最大応答時間は、 **ipv6 mld snooping last-listener-query-interval** グローバル コンフィギュレーション コマンドにより設定します。削除されたポートがマルチキャスト アドレスの最後のメンバである場合は、マルチキャストアドレスも削除され、スイッチは検出されたマルチキャスト ルータすべてにアドレス脱退情報を送信します。

TCN 処理

ipv6 mld snooping tcn query solicit グローバル コンフィギュレーション コマンドを使用して、トポロジ変更通知（TCN）送信要求を有効にすると、MLDv1 スヌーピングは、設定された数の MLDv1 クエリによりすべての IPv6 マルチキャストトラフィックをフラッディングするよう VLAN に設定してから、選択されたポートにのみマルチキャストデータの送信を開始します。この値は、 **ipv6 mld snooping tcn flood query count** グローバルコンフィギュレーションコマンドを使用して設定します。デフォルトでは、2つのクエリーが送信されます。スイッチが VLAN 内の STP ルートになる場合、またはスイッチがユーザにより設定された場合は、リンクに対してローカルで有効な IPv6 送信元アドレスを持つ MLDv1 グローバル Done メッセージも生成されます。これは IGMP スヌーピングの場合と同じです。

スイッチ スタックでの MLD スヌーピング

MLD IPv6 グループ アドレス データベースは、どのスイッチが IPv6 マルチキャスト グループを学習するかに関係なく、スタック内のすべてのスイッチ上で保持されます。レポート抑制とプロキシ レポーティングは、スタック全体で行われます。最大応答時間の間、1 つのグループに受信したレポートでマルチキャストルータに転送されるのは、どのスイッチにそのレポートが到達したかに関係なく、1 つだけです。

新しいアクティブスタックの選択は、IPv6マルチキャストデータの学習やブリッジングには影響しません。IPv6マルチキャストデータのブリッジングは、アクティブスタックの再選択中にも停止しません。新しいスイッチがスタックに追加されると、アクティブスタックからの学習済み IPv6 マルチキャスト情報との同期が取られます。同期が完了するまでは、新しく追加されたスイッチでのデータ入力は、不明マルチキャスト データとして扱われます。

IPv6 MLD スヌーピングの設定方法

MLD スヌーピングのデフォルト設定

表 *33 : MLD* スヌーピングのデフォルト設定

機能	デフォルト設定
MLD スヌーピング（グローバル）	ディセーブル。
MLD スヌーピング（VLAN 単位）	イネーブルVLAN MLD スヌーピングが実行されるためには、MLD スヌーピングがグローバルにイネーブルである必要があります。
IPv6 マルチキャスト アドレス	未設定
IPv6 マルチキャスト ルータ ポート	未設定
MLD スヌーピング即時脱退	ディセーブル。
MLD スヌーピングの堅牢性変数	グローバル：2、VLAN 単位：0 （注） VLAN 値はグローバル設定を上書きします。VLAN 値が 0 の場合、VLAN はグローバル数を使用します。
最後のリスナー クエリー カウント	グローバル：2、VLAN 単位：0 （注） VLAN 値はグローバル設定を上書きします。VLAN 値が 0 の場合、VLAN はグローバル数を使用します。
最後のリスナークエリーインターバル	グローバル：1000（1 秒）、VLAN：0 （注） VLAN 値はグローバル設定を上書きします。VLAN 値が 0 の場合、VLAN はグローバルのインターバルを使用します。
TCN クエリー送信請求	ディセーブル。

機能	デフォルト設定
TCN クエリー カウント	2
MLD リスナー抑制	有効。

MLD スヌーピング設定時の注意事項

MLD スヌーピングの設定時は、次の注意事項に従ってください。

- MLD スヌーピングの特性はいつでも設定できますが、設定を有効にする場合は、 **ipv6 mld snooping** グローバル コンフィギュレーション コマンドを使用して MLD スヌーピングをグローバルにイネーブルにする必要があります。
- IPv6 マルチキャストルータが Catalyst 6500 スイッチであり、拡張 VLAN（範囲 1006 ～ 4094）を使用する場合、スイッチが VLAN 上でクエリを受信できるようにするため、IPv6 MLD スヌーピングを Catalyst 6500 スイッチの拡張 VLAN でイネーブルにする必要があります。標準範囲 VLAN（1 ～ 1005）の場合、IPv6 MLD スヌーピングを Catalyst 6500 スイッチの VLAN でイネーブルにする必要はありません。
- MLD スヌーピングと IGMP スヌーピングは相互に独立して動作します。スイッチで両方の機能を同時にイネーブルにできます。

スイッチでのMLDスヌーピングのイネーブル化またはディセーブル化

デフォルトでは、IPv6 MLD スヌーピングはスイッチではグローバルにディセーブルで、すべての VLAN ではイネーブルです。MLD スヌーピングがグローバルにディセーブルの場合は、すべての VLAN でもディセーブルです。MLD スヌーピングをグローバルにイネーブルにすると、VLAN 設定はグローバル設定を上書きします。つまり、MLD スヌーピングはデフォルトステート（イネーブル）の VLAN インターフェイスでのみイネーブルになります。

VLAN 単位または VLAN 範囲で MLD スヌーピングをイネーブルおよびディセーブルにできますが、MLD スヌーピングをグローバルにディセーブルにした場合は、すべての VLAN でディセーブルになります。グローバル スヌーピングがイネーブルの場合、VLAN スヌーピングをイネーブルまたはディセーブルに設定できます。

スイッチでグローバルに MLD スヌーピングをイネーブルにするには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 mld snooping** 例： スイッチ(config)# **ipv6 mld snooping**	スイッチで MLD スヌーピングをイネーブルにします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**copy running-config startup-config** 例： スイッチ(config)# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。
ステップ 6	**reload** 例： スイッチ(config)# **reload**	OS（オペレーティング システム）をリロードします。

VLAN に対する MLD スヌーピングのイネーブル化またはディセーブル化

VLAN で MLD スヌーピングをイネーブルにするには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 mld snooping** 例： スイッチ(config)# **ipv6 mld snooping**	スイッチで MLD スヌーピングをイネーブルにします。
ステップ 4	**ipv6 mld snooping vlan** *vlan-id* 例： スイッチ(config)# **ipv6 mld snooping vlan 1**	VLAN で MLD スヌーピングをイネーブルにします。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。 （注） VLAN スヌーピングをイネーブルにするには、MLD スヌーピングがグローバルにイネーブルである必要があります。
ステップ 5	**end** 例： スイッチ(config)# **ipv6 mld snooping vlan 1**	特権 EXEC モードに戻ります。

スタティックなマルチキャスト グループの設定

ホストまたはレイヤ 2 ポートは、通常マルチキャスト グループにダイナミックに加入しますが、VLAN に IPv6 マルチキャスト アドレスおよびメンバ ポートをスタティックに設定することもできます。

マルチキャスト グループのメンバとしてレイヤ 2 ポートを追加するには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 mld snooping vlan** *vlan-id* **static** *ipv6_multicast_address* **interface** *interface-id* 例： スイッチ(config)# **ipv6 mld snooping vlan 1 static 3333.0000.1111 interface gigabitethernet 0/1**	マルチキャスト グループのメンバとしてレイヤ 2 ポートにマルチキャスト グループを設定します。 • *vlan-id* は、マルチキャスト グループの VLAN ID です。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。 • *ipv6_multicast_address* は、128 ビットのグループ IPv6 アドレスです。このアドレスは RFC 2373 で指定された形式でなければなりません。 • *interface-id* は、メンバ ポートです。物理インターフェイスまたはポートチャネル（1 ～ 48）に設定できます。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	次のいずれかを使用します。 • **show ipv6 mld snooping address** • **show ipv6 mld snooping address vlan** *vlan-id* 例： スイッチ# **show ipv6 mld snooping address** または スイッチ# **show ipv6 mld snooping vlan 1**	スタティック メンバ ポートおよび IPv6 アドレスを確認します。

マルチキャスト ルータ ポートの設定

（注） マルチキャスト ルータへのスタティック接続は、スイッチ ポートに限りサポートされます。

VLAN にマルチキャスト ルータ ポートを追加するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 mld snooping vlan** *vlan-id* **mrouter interface** *interface-id* 例： スイッチ(config)# **ipv6 mld snooping vlan 1 mrouter interface gigabitethernet 0/2**	マルチキャスト ルータの VLAN ID を指定して、マルチキャスト ルータにインターフェイスを指定します。 • 指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。 • このインターフェイスには物理インターフェイスまたはポート チャネルを指定できます。指定できるポートチャネルの範囲は 1 ～ 48 です。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show ipv6 mld snooping mrouter** [**vlan** *vlan-id*] 例： スイッチ# **show ipv6 mld snooping mrouter vlan 1**	VLAN インターフェイスで IPv6 MLD スヌーピングがイネーブルになっていることを確認します。

MLD 即時脱退のイネーブル化

MLDv1 即時脱退をイネーブルにするには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 mld snooping vlan** *vlan-id* **immediate-leave** 例： スイッチ(config)# **ipv6 mld snooping vlan 1 immediate-leave**	VLAN インターフェイスで MLD 即時脱退をイネーブルにします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show ipv6 mld snooping vlan** *vlan-id* 例： スイッチ# **show ipv6 mld snooping vlan 1**	VLAN インターフェイス上で即時脱退がイネーブルになっていることを確認します。

MLD スヌーピング クエリーの設定

スイッチまたはVLANにMLDスヌーピング クエリの特性を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 mld snooping robustness-variable** *value* 例：	（任意）スイッチが一般クエリーに応答しないリスナー（ポート）を削除する前に、送信されるクエ

	コマンドまたはアクション	目的
	スイッチ(config)# **ipv6 mld snooping robustness-variable 3**	リー数を設定します。指定できる範囲は 1 ～ 3 です。デフォルトは 2 です。
ステップ 4	**ipv6 mld snooping vlan** *vlan-id* **robustness-variable** *value* 例： スイッチ(config)# **ipv6 mld snooping vlan 1 robustness-variable 3**	（任意）VLAN 単位でロバストネス変数を設定します。これにより、MLD レポート応答がない場合にマルチキャスト アドレスがエージング アウトされるまでに、MLD スヌーピングが送信する一般クエリー数が決定されます。指定できる範囲は 1 ～ 3 です。デフォルトは 0 です。0 に設定すると、使用される数はグローバルな堅牢性変数の値になります。
ステップ 5	**ipv6 mld snooping last-listener-query-count** *count* 例： スイッチ(config)# **ipv6 mld snooping last-listener-query-count 7**	（任意）MLD クライアントがエージング アウトされる前にスイッチが送信する MASQ 数を設定します。指定できる範囲は 1 ～ 7 です。デフォルトは 2 です。クエリーは 1 秒後に送信されます。
ステップ 6	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-count** *count* 例： スイッチ(config)# **ipv6 mld snooping vlan 1 last-listener-query-count 7**	（任意）VLAN 単位でラスト リスナー クエリー カウントを設定します。この値はグローバルに設定された値を上書きします。指定できる範囲は 1 ～ 7 です。デフォルトは 0 です。0 に設定すると、グローバルなカウント値が使用されます。クエリーは 1 秒後に送信されます。
ステップ 7	**ipv6 mld snooping last-listener-query-interval** *interval* 例： スイッチ(config)# **ipv6 mld snooping last-listener-query-interval 2000**	（任意）スイッチが MASQ を送信したあと、マルチキャスト グループからポートを削除するまで待機する最大応答時間を設定します。指定できる範囲は、100 ～ 32,768 ミリ秒です。デフォルト値は 1000（1 秒）です。
ステップ 8	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-interval** *interval* 例： スイッチ(config)# **ipv6 mld snooping vlan 1 last-listener-query-interval 2000**	（任意）VLAN 単位で last-listener クエリー インターバルを設定します。この値はグローバルに設定された値を上書きします。指定できる範囲は、0 ～ 32,768 ミリ秒です。デフォルトは 0 です。0 に設定すると、グローバルな最後のリスナー クエリー インターバルが使用されます。
ステップ 9	**ipv6 mld snooping tcn query solicit** 例： スイッチ(config)# **ipv6 mld snooping tcn query solicit**	（任意）トポロジ変更通知（TCN）をイネーブルにします。これにより、VLAN は設定された数のクエリーに関する IPv6 マルチキャスト トラフィックすべてをフラッディングしてから、マルチキャスト データをマルチキャスト データの受信を要求するポートに対してのみ送信します。デフォルトでは、TCN はディセーブルに設定されています。

	コマンドまたはアクション	目的
ステップ 10	**ipv6 mld snooping tcn flood query count** *count* 例： スイッチ(config)# **ipv6 mld snooping tcn flood query count 5**	（任意）TCNがイネーブルの場合、送信されるTCNクエリー数を指定します。指定できる範囲は 1 ～ 10 で、デフォルトは 2 です。
ステップ 11	**end**	特権 EXEC モードに戻ります。
ステップ 12	**show ipv6 mld snooping querier** [**vlan** *vlan-id*] 例： スイッチ(config)# **show ipv6 mld snooping querier vlan 1**	（任意）スイッチまたはVLANのMLDスヌーピング クエリア情報を確認します。

MLD リスナー メッセージ抑制のディセーブル化

デフォルトでは、MLD スヌーピング リスナー メッセージ抑制はイネーブルに設定されています。この機能がイネーブルの場合、スイッチはマルチキャスト ルータ クエリーごとに 1 つの MLD レポートのみを転送します。メッセージ抑制がディセーブルの場合は、複数のマルチキャスト ルータに MLD レポートが転送されます。

MLD リスナー メッセージ抑制をディセーブルにするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no ipv6 mld snooping listener-message-suppression** 例： スイッチ(config)# **no ipv6 mld snooping listener-message-suppression**	MLD メッセージ抑制をディセーブルにします。
ステップ 4	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	
ステップ 5	**show ipv6 mld snooping** 例： スイッチ# **show ipv6 mld snooping**	IPv6 MLD スヌーピング レポート抑制がディセーブルであることを確認します。

MLD スヌーピング情報の表示

ダイナミックに学習された、あるいはスタティックに設定されたルータ ポートおよび VLAN インターフェイスのMLDスヌーピング情報を表示できます。また、MLDスヌーピング用に設定された VLAN の IPv6 グループ アドレス マルチキャスト エントリを表示することもできます。

表 *34 : MLD* スヌーピング情報表示用のコマンド

コマンド	目的
show ipv6 mld snooping [**vlan** *vlan-id*]	スイッチのすべての VLAN または指定された VLAN の MLD スヌーピング設定情報を表示します。 （任意）個々のVLANに関する情報を表示するには、**vlan** *vlan-id* を入力します。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。
show ipv6 mld snooping mrouter [**vlan** *vlan-id*]	ダイナミックに学習され、手動で設定されたマルチキャスト ルータ インターフェイスの情報を表示します。MLD スヌーピングをイネーブルにすると、スイッチはマルチキャスト ルータの接続先であるインターフェイスを自動的に学習します。これらのインターフェイスは動的に学習されます。 （任意）個々のVLANに関する情報を表示するには、**vlan** *vlan-id* を入力します。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。
show ipv6 mld snooping querier [**vlan** *vlan-id*]	VLAN 内で直前に受信した MLD クエリー メッセージの IPv6 アドレスおよび着信ポートに関する情報を表示します。 （任意） **vlan** *vlan-id* を入力して、単一の VLAN 情報を表示します。指定できる VLAN ID の範囲は 1 ～ 1001 および 1006 ～ 4094 です。

コマンド	目的
show ipv6 mld snooping address [**vlan** *vlan-id*] [**count** | **dynamic** | **user**]	すべての IPv6 マルチキャスト アドレス情報あるいはスイッチまたは VLAN の特定の IPv6 マルチキャスト アドレス情報を表示します。 • **count** を入力して、スイッチまたは VLAN のグループ数を表示します。 • **dynamic** を入力して、スイッチまたは VLAN の MLD スヌーピング学習済みグループ情報を表示します。 • **user** を入力して、スイッチまたは VLAN の MLD スヌーピングユーザ設定グループ情報を表示します。
show ipv6 mld snooping address vlan *vlan-id* [*ipv6-multicast-address*]	指定の VLAN および IPv6 マルチキャスト アドレスの MLD スヌーピングを表示します。

MLD スヌーピングの設定例

スタティックなマルチキャスト グループの設定：例

次に、IPv6 マルチキャスト グループをスタティックに設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ipv6 mld snooping vlan 2 static 3333.0000.1111 interface
gigabitethernet1/0/1
スイッチ(config)# end
```

マルチキャスト ルータ ポートの設定：例

次に、VLAN 200 にマルチキャスト ルータ ポートを追加する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ipv6 mld snooping vlan 200 mrouter interface gigabitethernet
          0/2
スイッチ(config)# exit
```

MLD 即時脱退のイネーブル化：例

次に、VLAN 130 で MLD 即時脱退をイネーブルにする例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ipv6 mld snooping vlan 130 immediate-leave
スイッチ(config)# exit
```

MLD スヌーピング クエリーの設定：例

次に、MLD スヌーピングのグローバルな堅牢性変数を 3 に設定する例を示します。

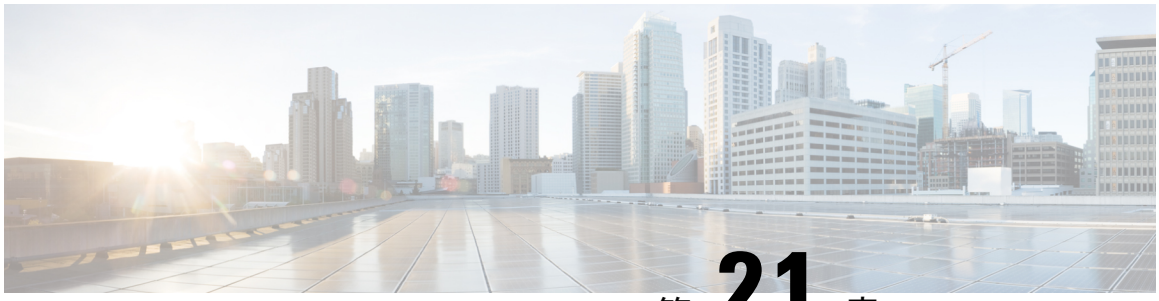
```
スイッチ# configure terminal
スイッチ(config)# ipv6 mld snooping robustness-variable 3
スイッチ(config)# exit
```

次に、VLAN の MLD スヌーピングの最後のリスナー クエリー カウントを 3 に設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ipv6 mld snooping vlan 200 last-listener-query-count 3
スイッチ(config)# exit
```

次に、MLD スヌーピングの最後のリスナー クエリー インターバル（最大応答時間）を 2000（2 秒）に設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ipv6 mld snooping last-listener-query-interval 2000
スイッチ(config)# exit
```



第 21 章

IPv6 ユニキャスト ルーティングの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

IPv6 ユニキャスト ルーティングの設定について

この章では、スイッチに IPv6 ユニキャスト ルーティングを設定する方法について説明します。

IPv6 の概要

IPv4 ユーザーは IPv6 に移行することができ、エンドツーエンドのセキュリティ、Quality of Service（QoS）、およびグローバルに一意なアドレスのようなサービスを利用できます。IPv6 アドレス スペースによって、プライベート アドレスの必要性が低下し、ネットワーク エッジの境界ルータで Network Address Translation（NAT; ネットワーク アドレス変換）処理を行う必要性も低下します。

シスコの IPv6 の実装方法については、次の URL を参照してください。

http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html

IPv6 およびこの章のその他の機能については、

- 『*Cisco IOS IPv6 Configuration Library*』を参照してください。
- Cisco.com の [Search] フィールドを使用して、Cisco IOS ソフトウェア マニュアルを特定します。たとえば、スタティック ルートについての情報が必要な場合は、[Search] フィールドで *Implementing Static Routes for IPv6* と入力すると、スタティック ルートについて調べられます。

IPv6 アドレス

スイッチがサポートするのは、IPv6ユニキャストアドレスのみです。サイトローカルユニキャスト アドレスおよびマルチキャスト アドレスはサポートされません。

IPv6 の 128 ビット アドレスは、コロンで区切られた一連の 8 つの 16 進フィールド（n:n:n:n:n:n:n:n. の形式）で表されます。次に、IPv6 アドレスの例を示します。

2031:0000:130F:0000:0000:09C0:080F:130B

実装を容易にするために、各フィールドの先行ゼロは省略可能です。上記アドレスは、先行ゼロを省略した次のアドレスと同じです。

2031:0:130F:0:0:9C0:80F:130B

2 つのコロン（::）を使用して、ゼロが連続する 16 進フィールドを表すことができます。ただし、この短縮形を使用できるのは、各アドレス内で 1 回のみです。

2031:0:130F::09C0:080F:130B

IPv6 アドレス形式、アドレス タイプ、および IPv6 パケット ヘッダーの詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipv6_basic/configuration/xe-3e/ip6b-xe-3e-book.html を参照してください。

「Implementing Addressing and Basic Connectivity」の章では、次の項の内容が Catalyst 2960、2960-S、2960-C、2960-X、2960-CX、3560-CX スイッチに適用されます。

- IPv6 アドレス形式
- IPv6 アドレス タイプ：マルチキャスト
- Ipv6 アドレス 出力表示
- 簡易 IPv6 パケット ヘッダー

サポート対象の IPv6 ユニキャスト ルーティング機能

ここでは、スイッチでサポートされている IPv6 プロトコル機能について説明します。

128 ビット幅のユニキャスト アドレス

スイッチは集約可能なグローバル ユニキャスト アドレスおよびリンク ローカル ユニキャスト アドレスをサポートします。サイト ローカル ユニキャスト アドレスはサポートされていません。

- 集約可能なグローバル ユニキャスト アドレスは、集約可能グローバル ユニキャスト プレフィックスの付いた IPv6 アドレスです。このアドレス構造を使用すると、ルーティング プレフィックスを厳格に集約することができ、グローバル ルーティング テーブル内のルーティング テーブル エントリ数が制限されます。これらのアドレスは、組織を経由して最終的にインターネット サービス プロバイダに至る集約リンク上で使用されます。

  これらのアドレスはグローバル ルーティング プレフィックス、サブネット ID、およびインターフェイス ID によって定義されます。現在のグローバル ユニキャスト アドレス割り当てには、バイナリ値 001（2000::/3）で開始するアドレス範囲が使用されます。プレフィックスが 2000::/3（001）～ E000::/3（111）のアドレスには、Extended Unique Identifier（EUI）64 フォーマットの 64 ビット インターフェイス ID を設定する必要があります。

- リンク ローカル ユニキャスト アドレスをすべてのインターフェイスに自動的に設定するには、修飾 EUI フォーマット内で、リンク ローカル プレフィックス FE80::/10（1111 1110 10）およびインターフェイスID を使用します。ネイバー探索プロトコル（NDP）およびステートレス自動設定プロセスでは、リンクローカルアドレスが使用されます。ローカルリンク上のノードは、リンクローカルアドレスを使用します。通信する場合に、グローバルに一意なアドレスは不要です。IPv6 ルータは、リンクローカルの送信元または宛先アドレスを持つパケットをその他のリンクに転送しません。

詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library* 』の「Implementing IPv6 Addressing and Basic Connectivity」の章にある IPv6 ユニキャスト アドレスに関する項を参照してください。

IPv6 の DNS

IPv6 は、ドメイン ネーム システム（DNS）のレコード タイプを、DNS 名前/アドレスおよびアドレス/名前の検索プロセスでサポートします。DNS AAAA リソース レコード タイプは IPv6 アドレスをサポートし、IPv4 の A アドレス レコードと同等です。スイッチは IPv4 および IPv6 の DNS 解決をサポートします。

IPv6 ユニキャストのパス MTU ディスカバリ

スイッチはシステム最大伝送単位（MTU）の IPv6 ノードへのアドバタイズおよびパス MTU ディスカバリをサポートします。パス MTU ディスカバリを使用すると、ホストは指定されたデータ パスを通るすべてのリンクの MTU サイズを動的に検出して、サイズに合せて調整できます。IPv6 では、パスを通るリンクの MTU サイズが小さくてパケット サイズに対応できない場合、パケットの送信元がフラグメンテーションを処理します。

ICMPv6

IPv6 のインターネット制御メッセージ プロトコル（ICMP）は、ICMP 宛先到達不能メッセージなどのエラー メッセージを生成して、処理中に発生したエラーや、その他の診断機能を報告します。IPv6 では、ネイバー探索プロトコルおよびパス MTU ディスカバリに ICMP パケットも使用されます。

ネイバー探索

スイッチは、IPv6 対応の NDP、ICMPv6 の最上部で稼働するプロトコル、および NDP をサポートしない IPv6 ステーション対応のスタティック ネイバー エントリをサポートします。IPv6 ネイバー探索プロセスは ICMP メッセージおよび送信請求ノード マルチキャスト アドレスを使用して、同じネットワーク（ローカル リンク）上のネイバーのリンク層アドレスを判別し、ネイバーに到達できるかどうかを確認し、近接ルータを追跡します。

スイッチは、マスク長が 64 未満のルートに対して ICMPv6 リダイレクトをサポートしています。マスク長が 64 ビットを超えるホスト ルートまたは集約ルートでは、ICMP リダイレクトがサポートされません。

ネイバー探索スロットリングにより、IPv6 パケットをルーティングするためにネクスト ホップ転送情報を取得するプロセス中に、スイッチ CPU に不必要な負荷がかかりません。IPv6 パケットのネクストホップがスイッチによってアクティブに解決しようとしている同じネイバーである場合は、そのようなパケットが追加されると、スイッチはそのパケットをドロップします。このドロップにより、CPU に余分な負荷がかからないようになります。

デフォルト ルータ プリファレンス

スイッチは、ルータのアドバタイズメント メッセージの拡張機能である、IPv6 Default Router Prefernce（DRP）をサポートします。DRP では、特にホストがマルチホーム構成されていて、ルータが異なるリンク上にある場合に、ホストが適切なルータを選択する機能が向上しました。スイッチは、Route Information Option（RFC 4191）をサポートしません。

IPv6 ホストは、オフリンク宛先へのトラフィック用にルータを選択する、デフォルト ルータ リストを維持します。次に、宛先用に選択されたルータは、宛先キャッシュに格納されます。IPv6 NDP では、到達可能であるルータまたは到達可能性の高いルータが、到達可能性が不明または低いルータよりも優先されます。NDP は、到達可能または到達できる可能性の高いルータとして、常に同じルータを選択するか、またはルータ リストを循環して選択できます。DRP を使用することにより、両方ともが到達可能または到達できる可能性の高い 2 台のルータの一方を他方に対して優先させるよう IPv6 ホストを設定することができます。

DRP for IPv6 の設定については、「*DRP* の設定」を参照してください。

DRP for IPv6 の詳細情報については、Cisco.com の『*Cisco IOS IPv6 Configuration Library*』を参照してください。

IPv6 のステートレス自動設定および重複アドレス検出

スイッチではステートレス自動設定が使用されているため、ホストやモバイル IP アドレスの管理のような、リンク、サブネット、およびサイトアドレス指定の変更を管理することができます。ホストは独自のリンクローカルアドレスを自動的に設定します。起動元ノードはルータに送信請求を送信して、インターフェイス設定をアドバタイズするようルータに要求します。

自動設定および重複アドレス検出の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing IPv6 Addressing and Basic Connectivity」の章を参照してください。

IPv6 アプリケーション

スイッチは、次のアプリケーションについて IPv6 をサポートします。

- Ping、traceroute、および Telnet
- IPv6 トランスポートによるセキュア シェル（SSH）
- IPv6 トランスポートによる HTTP サーバー アクセス
- IPv4 トランスポートによる AAAA の DNS レゾルバ
- IPv6 アドレスの Cisco Discovery Protocol（CDP）サポート

これらのアプリケーションの管理に関する詳細については、Cisco.com の『*Cisco IOS IPv6 Configuration Library*』を参照してください。

DHCP for IPv6 アドレスの割り当て

DHCPv6 を使用すると、DHCP サーバーは IPv6 ネットワーク アドレスなどの設定パラメータを IPv6 クライアントに渡すことができます。このアドレス割り当て機能により、ホストが接続するネットワークに基づいて、適切なプレフィックス内での重複しないアドレス割り当てが管理されます。アドレスは、1つまたは複数のプレフィックス プールから割り当てることができます。デフォルトのドメインおよびDNS ネーム サーバー アドレスなど、その他のオプションは、クライアントに戻すことができます。アドレス プールは、特定のインターフェイス、複数のインターフェイス上で使用する場合に割り当てられます。または、サーバーが自動的に適切なプールを検出できます。

DHCP for IPv6 の設定については、「*DHCP for IPv6* アドレス割り当ての設定」のセクションを参照してください。

DHCPv6 クライアント、サーバー、またはリレーエージェント機能の設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』を参照してください。

IPv6 のスタティック ルート

スタティック ルートは手動で設定され、2つのネットワーキング デバイス間のルートを明示的に定義します。スタティック ルートが有効なのは、外部ネットワークへのパスが 1 つしかない小規模ネットワークの場合、または大規模ネットワークで特定のトラフィック タイプにセキュリティを設定する場合です。

IPv6 のスタティック ルーティングの設定（CLI）

IPv6 用のスタティックルートの設定については、「*IPv6* 用のスタティックルーティングの設定」を参照してください。

スタティック ルートの詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing Static Routes for IPv6」の章を参照してください。

RIP for IPv6

IPv6 の Routing Information Protocol（RIP）は、ルーティング メトリックとしてホップ カウントを使用するディスタンスベクトル プロトコルです。IPv6 アドレスおよびプレフィックスのサポート、すべての RIP ルータを含むマルチキャスト グループ アドレス FF02::9 を RIP アップデート メッセージの宛先アドレスとして使用する機能などがあります。

IPv6 の RIP の設定については、「*IPv6 の RIP の設定*」を参照してください。

IPv6 の RIP の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing RIP for IPv6」の章を参照してください。

OSPF for IPv6

フィーチャ セットを実行しているスイッチは、IPv6 の Open Shortest Path First（OSPF）（IP のリンクステート プロトコル）をサポートします。詳細については、Cisco.com の『*Cisco IOS IPv6 Configuration Library*』を参照してください。

OSPFv3 グレースフル リスタート

OSPFv3 機能により、OSPFv3 ルーティング プロトコル情報が復元されている間も、既知のルート上でノンストップのデータの転送が可能になります。スイッチでは、グレースフル リスタートがリスタート モード（グレースフル リスタート対応スイッチの場合）とヘルパー モード（グレースフル リスタート認識スイッチの場合）のいずれかで使用されます。

グレースフル リスタート機能を使用するには、スイッチがハイアベイラビリティ ステートフル スイッチオーバー（SSO）モードである必要があります（デュアル ルート プロセッサ）。グレースフル リスタートに対応したスイッチでは、次の障害が発生した際にグレースフル リスタートが使用されます。

- スタンバイ ルート プロセッサへの切り替えが起こるルート プロセッサ障害
- 計画されたスタンバイ ルート プロセッサへのルート プロセッサの切り替え

グレースフル リスタート機能では、隣接スイッチがグレースフル リスタート認識である必要があります。

詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing OSFP for IPv6」の章を参照してください。

高速コンバージェンス：LSA および SPF スロットリング

OSPFv3 リンク ステート アドバタイズメント（LSA）および Shortest Path First（SPF）スロットリング機能は、ネットワークが不安定なときに、OSPFv3 でのリンク ステート アドバタイズメントの更新の速度を低下させる動的な方法ダイナミック方式を提供します。またこの機能を使用すると、LSA のレート制限をミリ秒単位で指定することにより、OSPFv3 コンバージェンス時間の短縮が可能になります。

OSPFv3 では以前はレート制限 SPF 計算および LSA 生成にスタティック タイマーを使用しました。これらのタイマーを設定することもできますが、値は秒単位で指定するため、OSPFv3 コンバージェンスに制限が課せられます。LSA および SPF スロットリングは、すばやく応答できる高度な SPF および LSA レート制限方式を提供することにより、1 秒未満単位でのコンバージェンスを実現し、長引く不安定期間中にも安定性および保護を提供します。

IPsec を使用した認証サポート

OSPF for IPv6（OSPFv3）パケットが変更されずにスイッチに再送信されるようにするには、OSPFv3 パケットを認証する必要があります。OSPFv3 は、IPsec セキュア ソケット API を使用

して OSPFv3 パケットに認証を追加します。この API は、IPv6 をサポートするように拡張されています。

OSPFv3 では、認証をイネーブルにするために IPsec を使用する必要があります。OSPFv3 で使用するために必要な IPsec API は暗号イメージのみに含まれるため、認証を使用するには暗号イメージが必要です。

IPv6 の HSRP の設定

HSRP は、任意の単一のルータのアベイラビリティに依存せず、ルーティング IPv6 トラフィックにルーティング冗長性を提供します。IPv6 ホストは、IPv6 ネイバー探索ルータのアドバタイズメントメッセージによって使用可能なルータを学習します。これらのメッセージは定期的にマルチキャストされるか、ホストにより送信請求されます。

HSRP IPv6 グループには、HSRP グループ番号に基づく仮想 MAC アドレス、およびデフォルトで HSRP 仮想 MAC アドレスに基づく HSRP の仮想 IPv6 リンクローカル アドレスがあります。HSRP グループがアクティブな場合、定期的なメッセージが HSRP 仮想 IPv6 リンクローカルアドレスに送信されます。グループがアクティブ ステートでなくなった場合、これらのメッセージは最後のメッセージが送信されたあとで停止します。

（注） IPv6 の HSRP を設定する場合、インターフェイス上で HSRP version 2（HSRPv2）をイネーブルにする必要があります。

EIGRP IPv6

IP サービス フィーチャ セットを実行中のスイッチは、IPv6 の Enhanced Interior Gateway Routing Protocol（EIGRP）をサポートします。IPv6 の EIGRP は稼働するインターフェイス上で設定されるため、グローバルな IPv6 アドレスは不要です。

（注） IP ベース フィーチャ セットを実行中のスイッチでは、IPv6 EIGRP スタブ ルーティングを含め、IPv6 EIGRP 機能はすべてサポートされません。

EIGRP IPv6 インスタンスでは、実行する前に暗示的または明示的なルータ ID が必要です。暗示的なルータ ID はローカルの IPv4 アドレスを基にして作成されるため、すべての IPv4 ノードには常に使用可能なルータ ID があります。ただし、EIGRP IPv6 は IPv6 ノードだけが含まれるネットワークで稼働するため、使用可能な IPv4 ルータ ID がない場合があります。

EIGRP for IPv6 の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing EIGRP for IPv6」の章を参照してください。

SNMP and Syslog Over IPv6

IPv4 と IPv6 の両方をサポートするには、IPv6 のネットワーク管理で IPv4 および IPv6 のトランスポートが必要になります。Syslog over IPv6 は、このトランスポートのアドレス データ タイプをサポートします。

Simple Network Management Protocol (SNMP) と syslog over IPv6 は、次の機能を提供します。

- IPv4 と IPv6 両方のサポート
- SNMP に対する IPv6 トランスポート、および SNMP 変更による IPv6 ホストのトラップのサポート
- IPv6 アドレス指定をサポートするための SNMP および syslog に関連する MIB
- IPv6 ホストをトラップ レシーバとして設定

Over IPv6 をサポートするため、SNMP は既存の IP トランスポート マッピングを変更して、IPv4 と IPv6 を同時にサポートします。次の SNMP 動作は、IPv6 トランスポート管理をサポートします。

- デフォルト設定のユーザー データグラム プロトコル（UDP）SNMP ソケットを開く
- *SR_IPV6_TRANSPORT* と呼ばれる新しいトランスポート メカニズムを提供
- IPv6 トランスポートによる SNMP 通知の送信
- IPv6 トランスポートの SNMP 名のアクセス リストのサポート
- IPv6 トランスポートを使用した SNMP プロキシ転送のサポート
- SNMP マネージャ機能と IPv6 トランスポートの連動確認

設定手順を含む、SNMP over IPv6 については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Managing Cisco IOS Applications over IPv6」の章を参照してください。

設定手順を含む、syslog over IPv6 については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing IPv6 Addressing and Basic Connectivity」の章を参照してください。

HTTP(S) Over IPv6

HTTP クライアントは要求を IPv4 HTTP サーバーと IPv6 HTTP サーバーの両方に送信し、これらのサーバーは IPv4 HTTP クライアントと IPv6 HTTP クライアントの両方からの要求に応答します。IPv6 アドレスを含む URL は、16 ビット値をコロンで区切った 16 進数で指定する必要があります。

受信ソケット コールは、IPv4 アドレス ファミリまたは IPv6 アドレス ファミリを選択します。受信ソケットは、IPv4 ソケットまたは IPv6 ソケットのいずれかです。リスニング ソケットは、接続を示す IPvv4 と IPv6 の両方の信号を待ち受け続けます。IPv6 リスニング ソケットは、IPv6 ワイルドカード アドレスにバインドされています。

基本 TCP/IP スタックは、デュアル スタック環境をサポートします。HTTP には、TCP/IP スタック、およびネットワーク層相互作用を処理するためのソケットが必要です。

HTTP 接続を確立するには、基本ネットワーク接続（**ping**）がクライアントとサーバーホストとの間に存在する必要があります。

詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Managing Cisco IOS Applications over IPv6」の章を参照してください。

サポートされていない IPv6 ユニキャスト ルーティング機能

スイッチは、次の IPv6 機能をサポートしません。

- サイトローカルアドレス宛ての IPv6 パケット
- IPv4/IPv6 や IPv6/IPv4 などのトンネリング プロトコル
- IPv4/IPv6 または IPv6/IPv4 トンネリング プロトコルをサポートするトンネル エンドポイントとしてのスイッチ

IPv6 機能の制限

スイッチでは IPv6 はハードウェアに実装されるため、ハードウェア メモリ内の IPv6 圧縮アドレスによる制限がいくつか発生します。これらのハードウェア制限により、機能の一部が失われて、制限されます。

機能の制限は次のとおりです。

- スイッチはハードウェアで SNAP カプセル化 IPv6 パケットを転送できません。これらはソフトウェアで転送されます。
- スイッチはソースルート IPv6 パケットに関する QoS 分類をハードウェアで適用できません。

IPv6 の設定

IPv6 のデフォルト設定

表 *35 : IPv6* のデフォルト設定

機能	デフォルト設定
SDM テンプレート	アドバンスデスクトップ。デフォルトは拡張テンプレート デフォルト
IPv6 アドレス	未設定

IPv6 アドレッシングの設定と IPv6 ルーティングの有効化

ここでは、IPv6 アドレスを各レイヤ 3 インターフェイスに割り当てて、IPv6 トラフィックをスイッチ上でグローバル転送する方法を説明します。

スイッチ上の IPv6 を設定する前に、次の注意事項に従ってください。

- 必ずデュアル IPv4/IPv6 SDM テンプレートを選択してください。

- **ipv6 address** インターフェイス コンフィギュレーション コマンドでは、16 ビット値を使用したコロン区切りの 16 進形式で指定したアドレスで *ipv6-address* 変数および *ipv6-prefix* 変数を入力する必要があります。*prefix-length* 変数（スラッシュ（/）で始まる）は、プレフィックス（アドレスのネットワーク部分）を構成するアドレスの上位連続ビット数を示す 10 進値です。

インターフェイス上の IPv6 トラフィックを転送するには、そのインターフェイス上でグローバル IPv6 アドレスを設定する必要があります。インターフェイス上で IPv6 アドレスを設定すると、リンクローカルアドレスの設定、およびそのインターフェイスに対する IPv6 のアクティブ化が自動的に行われます。設定されたインターフェイスは、次に示す、該当リンクの必須マルチキャスト グループに自動的に参加します。

- インターフェイスに割り当てられた各ユニキャストアドレスの送信要求ノードマルチキャスト グループ FF02:0:0:0:0:1:ff00::/104（このアドレスはネイバー探索プロセスで使用される）
- 全ノード向けリンクローカルマルチキャストグループ FF02::1
- 全ルータ向けリンクローカルマルチキャストグループ FF02::2

IPv6 ルーティングの設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing Addressing and Basic Connectivity for IPv6」の章を参照してください。

レイヤ 3 インターフェイスに IPv6 アドレスを割り当てて、IPv6 ルーティングをイネーブルにするは、特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例**： `スイッチ# configure terminal`	スイッチのリロード後、グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* **例**： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。インターフェイスは物理インターフェイス、スイッチ仮想インターフェイス（SVI）、またはレイヤ 3 EtherChannel に設定できます。
ステップ 3	**no switchport** **例**： `スイッチ(config-if)# no switchport`	レイヤ 2 コンフィギュレーション モードからインターフェイスを削除します（物理インターフェイスの場合）。

	コマンドまたはアクション	目的
ステップ 4	次のいずれかを使用します。 • **ipv6 address** *ipv6-prefix/prefix length* **eui-64** • **ipv6 address** *ipv6-address/prefix length* • **ipv6 address** *ipv6-address* **link-local** • **ipv6 enable** 例： `スイッチ(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64` `スイッチ(config-if)# ipv6 address 2001:0DB8:c18:1::/64` `スイッチ(config-if)# ipv6 address 2001:0DB8:c18:1:: link-local` `スイッチ(config-if)# ipv6 enable`	• IPv6 アドレスの下位 64 ビットの拡張固有識別子（EUI）を使用して、グローバル IPv6 アドレスを指定します。ネットワーク プレフィックスだけを指定します。最終の 64 ビットは、スイッチの MAC アドレスから自動的に計算されます。これにより、インターフェイス上で IPv6 処理が有効になります。 • インターフェイスの IPv6 アドレスを手動で設定します。 • インターフェイスで IPv6 が有効な場合に自動設定されるリンクローカルアドレスでなく、インターフェイス上の特定のリンクローカルなアドレスを使用するように指定します。このコマンドにより、インターフェイス上で IPv6 処理が有効になります。 • インターフェイスに IPv6 リンクローカルアドレスを自動設定し、インターフェイスでの IPv6 処理を有効にします。リンクローカルアドレスを使用できるのは、同じリンク上のノードと通信する場合だけです。
ステップ 5	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**ip routing** 例： `スイッチ(config)# ip routing`	スイッチ上で IP ルーティングをイネーブルにします。
ステップ 7	**ipv6 unicast-routing** 例： `スイッチ(config)# ipv6 unicast-routing`	IPv6 ユニキャスト データ パケットの転送を有効にします。
ステップ 8	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	
ステップ 9	**show ipv6 interface** *interface-id* 例： スイッチ# **show ipv6 interface gigabitethernet 1/0/1**	入力を確認します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 でのファースト ホップ セキュリティの設定

IPv6 でのファースト ホップ セキュリティの前提条件

- 必要な、IPv6 が有効になっている SDM テンプレートが設定されていること。
- **mls qos** コマンドを使用して CoPP ポリシーを設定する前に、スイッチで QoS を有効にする必要があります。

IPv6 でのファースト ホップ セキュリティの制約事項

- 次の制限は、FHS ポリシーを EtherChannel インターフェイスに適用する場合に該当します（ポート チャネル）。
  - FHS ポリシーがアタッチされた物理ポートは EtherChannel グループに参加することができません。
  - FHS ポリシーは、EtherChannel グループのメンバーである場合に物理ポートにアタッチすることができません。
- デフォルトでは、スヌーピング ポリシーにはセキュリティ レベルのガードがあります。そのようなスヌーピング ポリシーがアクセス スイッチに設定されると、ルータまたは DHCP サーバー/リレーに対応するアップリンク ポートが信頼できるポートとして設定されていても、IPv6（DHCPv6）サーバー パケットに対する外部 IPv6 ルータ アドバタイズメント(RA)または Dynamic Host Configuration Protocol はブロックされます。IPv6 RA または DHCPv6 サーバー メッセージを許可するには、次の手順を実行します。
  - IPv6 RA ガード ポリシー（RA の場合）または IPv6 DHCP ガード ポリシー（DHCP サーバー メッセージの場合）をアップリンク ポートに適用します。
  - 低いセキュリティ レベルでスヌーピング ポリシーを設定します（たとえば、glean や inspect など）。しかし、ファースト ホップ セキュリティ機能の利点が有効でないた

め、このようなスヌーピング ポリシーでは、低いセキュリティ レベルを設定することはお勧めしません。

- CSCvk32439 で報告された制限により、IPv6 SISF ベースのデバイス トラッキング ポリシーを使用した CoPP ポリシーには、次の制限が適用されます。
  - スイッチで IPv6 SISF ポリシーが設定されている場合、IPv6 NDP トラフィックを制限するには CoPP ポリシーが必要です。
  - NDP CoPP ポリシーが設定された後、制限されたトラフィックが CPU にヒットします。接続されているエンドポイントの合計に対応するには、NDP CoPP ポリシーの数を、スタック内の各スイッチに接続するユーザーの数よりわずかに多くする必要があります。スイッチに接続されているエンドポイントの数よりも少ない NDP CoPP ポリシーを設定すると、エンドポイントへの IP 割り当ては遅延しますが、完全に無視されるわけではありません。

（注） たとえば、5 つのスイッチのスタックに約 300 のユーザーがいる場合、NDP CoPP ポリシーは 300 を超える必要があります。

  - DHCPv6（サーバーからクライアントおよびクライアントからサーバー）CoPP ポリシーは、Lightweight DHCPv6 リレーエージェント（LDRA）がスイッチの IPv6 SISF ベースのデバイス トラッキング ポリシーで設定されている場合にのみ必要です。

IPv6 でのファースト ホップ セキュリティに関する情報

IPv6 のファーストホップ セキュリティ（FHS IPv6）は、ポリシーを物理インターフェイス、または VLAN にアタッチできる一連の IPv6 セキュリティ機能です。IPv6 ソフトウェア ポリシー データベース サービスは、これらのポリシーを保存しアクセスします。ポリシーを設定または変更すると、ポリシー属性はソフトウェア ポリシー データベースに保存または更新され、その後指定したとおりに適用されます。次の IPv6 ポリシーが現在サポートされています。

- IPv6 スヌーピング ポリシー：IPv6 スヌーピング ポリシーは、IPv6 内の FHS で使用できるほとんどの機能を有効にできるコンテナ ポリシーとして機能します。
- IPv6 FHS バインディング テーブルの内容：スイッチに接続された IPv6 ネイバーのデータベース テーブルはネイバー探索（ND）プロトコル スヌーピングなどの情報ソースから作成されます。このデータベースまたはバインディング テーブルは、リンク層アドレス（LLA）、IPv4 または IPv6 アドレス、およびスプーフィングやリダイレクト攻撃を防止するためにネイバーのプレフィックス バインディングを検証するために、さまざまな IPv6 ガード機能（IPv6 ND 検査など）によって使用されます。
- IPv6 ネイバー探索検査：IPv6 ND 検査は、レイヤ 2 ネイバー テーブル内のステートレス自動設定アドレスのバインディングを学習し、保護します。IPv6 ND 検査は、信頼できるバインディング テーブル データベースを構築するためにネイバー探索メッセージを分析します。準拠していない IPv6 ネイバー探索メッセージは破棄されます。ND メッセージは、

その IPv6 からメディア アクセス コントロール（MAC）へのマッピングが検証可能な場合に信頼できると見なされます。

この機能によって、DAD、アドレス解決、ルータ ディスカバリ、ネイバー キャッシュに対する攻撃などの、ND メカニズムに固有の脆弱性のいくつかが軽減されます。

- IPv6 ルータ アドバタイズメント ガード：IPv6 ルータ アドバタイズメント（RA）ガード機能を使用すると、ネットワーク管理者は、ネットワーク スイッチ プラットフォームに到着した不要または不正な RA ガード メッセージをブロックまたは拒否できます。RA は、リンクで自身をアナウンスするためにルータによって使用されます。RA ガード機能は、これらの RA を分析して、未承認のルータによって送信された偽の RA をフィルタリングして除外します。ホスト モードでは、ポートではルータ アドバタイズメントとルータ リダイレクト メッセージはすべて許可されません。RA ガード機能は、レイヤ 2 デバイスの設定情報を、受信した RA フレームで検出された情報と比較します。レイヤ 2 デバイスは、RA フレームとルータ リダイレクト フレームの内容を設定と照らし合わせて検証した後で、RA をユニキャストまたはマルチキャストの宛先に転送します。RA フレームの内容が検証されない場合は、RA は破棄されます。

- IPv6 DHCP ガード：IPv6 DHCP ガード機能は、承認されない DHCPv6 サーバーおよびリレー エージェントからの返信およびアドバタイズメント メッセージをブロックします。IPv6 DHCP ガードは、偽造されたメッセージがバインディング テーブルに入るのを防ぎ、DHCPv6 サーバーまたは DHCP リレーからデータを受信することが明示的に設定されていないポートで受信された DHCPv6 サーバー メッセージをブロックできます。この機能を使用するには、ポリシーを設定してインターフェイスまたは VLAN にアタッチします。DHCP ガード パケットをデバッグするには、**debug ipv6 snooping dhcp-guard** 特権 EXEC コマンドを使用します。

- IPv6 ソース ガード：IPv4 ソース ガードと同様、IPv6 ソース ガードは送信元アドレス スプーフィングを防ぐために、送信元アドレスまたはプレフィックスを検証します。

  ソース ガードでは、送信元または宛先アドレスに基づいてトラフィックを許可または拒否するようにハードウェアをプログラムします。ここでは、データ パケットのトラフィックのみを処理します。

  ソース ガード パケットをデバッグするには、debug ipv6 snooping source-guard 特権 EXEC コマンドを使用します。

  次の制約事項が適用されます。

  - FHS ポリシーは、EtherChannel グループのメンバーである場合に物理ポートにアタッチすることができません。

  - IPv6 ソース ガードがスイッチ ポートで有効になっている場合は、そのスイッチ ポートが属するインターフェイスで NDP または DHCP スヌーピングを有効にする必要があります。そうしないと、このポートからのすべてのデータ トラフィックがブロックされます。

  - IPv6 ソース ガード ポリシーを VLAN に適用することはできません。インターフェイス レベルのみでサポートされています。

- インターフェイスで IPv4 および IPv6 のソース ガードを一緒に設定する場合は、**ip verify source** の代わりに **ip verify source mac-check** の使用を推奨します。2 つの異なるフィルタリング ルール（IPv4（IP フィルタ）用と IPv6（IP-MAC フィルタ）用）が設定されているため、特定のポートの IPv4 接続が切断される可能性があります。
- IPv6 ソース ガードとプレフィックス ガードは同時に使用できません。ポリシーをインターフェイスに付加する際は、「アドレスを確認」するか「プレフィックスを確認」する必要はありますが、両方を確認する必要はありません。
- PVLAN と送信元/プレフィックス ガードは同時に適用できません。

IPv6 送信元ガードの詳細については、Cisco.comで『Cisco IOS IPv6 Configuration Guide Library』の「IPv6 Source Guard」の章を参照してください。

- IPv6 プレフィックス ガード：IPv6 プレフィックス ガードは、IPv6 送信元ガード機能内で動作し、デバイスがトポロジに不正なアドレスから発信されたトラフィックを拒否できるようにします。IPv6 プレフィックス ガードは、IPv6 プレフィックスが DHCP プレフィックス委任を使用してデバイス（ホーム ゲートウェイなど）に委任される場合によく使用されています。この機能は、リンクに割り当てられたアドレスの範囲を検出し、この範囲に入っていないアドレスを発信元とするトラフィックをブロックします。

  IPv6 プレフィックス ガードの詳細については、Cisco.comで『Cisco IOS IPv6 Configuration Guide Library』の「IPv6 Prefix Guard」の章を参照してください。

- IPv6 宛先ガード：IPv6 宛先ガード機能は、IPv6 ネイバー探索で動作し、リンク上でアクティブであると認識されているアドレスについてのみ、デバイスがアドレスを解決します。アドレス グリーニング機能に依存して、リンク上でアクティブなすべての宛先をバインディング テーブルに挿入してから、バインディング テーブルで宛先が見つからなかったときに実行される解決をブロックします。

  IPv6 宛先ガードに関する詳細については、Cisco.comで『Cisco IOS IPv6 Configuration Guide Library』の「IPv6 Destination Guard」の章を参照してください。

- IPv6 ネイバー検索マルチキャスト抑制：IPv6 ネイバー検索マルチキャスト抑制機能は、IPv6 のスヌーピング機能で、スイッチまたはワイヤレス コントローラで実行し、適切なリンク動作に必要な制御トラフィック量を削減するために使用されます。

- DHCPv6 リレー：Lightweight DHCPv6 リレー エージェント：Lightweight DHCPv6 リレー エージェント機能を使用するとリンクレイヤブリッジング（非ルーティング）機能を実行するアクセス ノードによってリレーエージェント情報が挿入されます。Lightweight DHCPv6 リレー エージェント（LDRA）機能は、DSL アクセス マルチプレクサ（DSLAM）や IPv6 制御やルーティング機能をサポートしないイーサネット スイッチなどの既存のアクセス ノードに実装できます。LDRA を使用して、DHCP バージョン 6（DHCPv6）メッセージ交換にリレーエージェント オプションを挿入して、主にクライアント側のインターフェイスを特定します。LDRA 機能は、インターフェイスと VLAN でイネーブルにできます。

（注） LDRA デバイスがクライアントに直接接続されている場合は、サーバー側で特定のサブネットまたはリンク情報を取得するために、インターフェイスにプール設定が必要です。この場合、LDRA デバイスが異なるサブネットまたはリンクに存在する場合、サーバーは正しいサブネットを取得できない場合があります。インターフェイスでプール名を設定して、クライアントに適切なサブネットまたはリンクを選択できるようになりました。

DHCPv6 リレーの詳細については、『IP Addressing: DHCP Configuration Guide, Cisco IOS Release 15.1SG』の「DHCPv6 Relay—Lightweight DHCPv6 Relay Agent」の項を参照してください。

IPv6 スヌーピング ポリシーの設定方法

手順の概要

1. **enable**
2. **configure terminal**
3. **IPv6 snooping policy** *policy -name*
4. ［**data-glean** ｜**default** ｜ **device-role** ［**node**｜**switch**］ ｜**limit**｛**address-count***value*｝ ｜ **no** ｜ **protocol** ［**all** ｜ **nodhcp**｜ **ndp**］ ｜**security-level** ［**glean**｜ **guard**｜ **inspect**］｜ **tracking** ［**disable** ｜ **enable**］｜ **trusted-port** ｝
5. **exit**
6. **show ipv6 snooping policy***policy-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**IPv6 snooping policy** *policy -name*	グローバル コンフィギュレーション モードでスヌーピング ポリシーを作成します。

	コマンドまたはアクション	目的
ステップ 4	[**data-glean** | **default** | **device-role** [**node** | **switch**] | **limit** {**address-count***value*} | **no** | **protocol** [**all** | **nodhcp** | **ndp**] | **security-level** [**glean** | **guard** | **inspect**] | **tracking** [**disable** | **enable**] | **trusted-port** }	データ アドレス グリーニングをイネーブルにし、さまざまな条件に対してメッセージを検証し、メッセージのセキュリティ レベルを指定します。 • （任意）**data-glean**：データ アドレス グリーニングをイネーブルにします。このオプションは、デフォルトで無効です。 • （任意）**default**：すべてのデフォルト オプションを設定します。 • （任意）**device-role** [**node** | **switch**]：ポートに接続されたデバイスのロールを認定します。 • （任意）**limit** {**address-count** *value*}：ターゲットごとに許可されるアドレス数を制限します。 • （任意）**no**：コマンドを無効にするか、またはそのデフォルトに設定します。 • （任意）**protocol** [**all** | **dhcp** | **ndp**]：分析のために、スヌーピング機能にどのプロトコルをリダイレクトするかを指定します。デフォルトは**all**です。デフォルトを変更するには、**no protocol** コマンドを使用します。 • （任意）**security-level** [**glean** | **guard** | **inspect**]：この機能によって適用されるセキュリティのレベルを指定します。 • **glean**：メッセージからアドレスを収集し、何も確認せずにバインディング テーブルに入力します。 • **guard**：アドレスを収集し、メッセージを検査します。さらに、RAおよびDHCPサーバ メッセージを拒否します。これがデフォルトのオプションです。 • **inspect**：アドレスを収集し、メッセージの一貫性と準拠を検証して、アドレスの所有権を適用します。 • （任意）**tracking** [**disable** | **enable**]：デフォルトのトラッキング動作を上書きし、トラッキング オプションを指定します。

	コマンドまたはアクション	目的
		• （任意）**trusted-port**：信頼できるポートを設定します。これにより、該当するターゲットに対するガードがディセーブルになります。信頼できるポートを経由して学習されたバインディングは、他のどのポートを経由して学習されたバインディングよりも優先されます。また、テーブル内にエントリを作成しているときに衝突が発生した場合も、信頼できるポートが優先されます。
ステップ 5	**exit**	スヌーピング ポリシー コンフィギュレーション モードを終了します。
ステップ 6	**show ipv6 snooping policy***policy-name*	スヌーピング ポリシー設定を表示します。

IPv6 スヌーピング ポリシーのインターフェイスまたは VLAN へのアタッチ方法

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかの作業を実行します。
   - **interface** *type number*
   - **switchport**
   - **ipv6 snooping** [**attach-policy** *policy_name*]

     または

   - **vlan configuration** *vlan list*
   - **ipv6 snooping attach-policy** *policy-name*
4. **show ipv6 snooping policy** *policy-name*
5. **show ipv6 neighbors binding**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかの作業を実行します。 • **interface** *type number* • **switchport** • **ipv6 snooping** [**attach-policy** *policy_name*] または • **vlan configuration** *vlan list* • **ipv6 snooping attach-policy** *policy-name*	インターフェイスのタイプおよび番号を指定し、インターフェイス コンフィギュレーション モードを開始します。 （注） typeは物理インターフェイスでも、イーサチャネルでもかまいません。 インターフェイスをレイヤ 2 ポートとして設定します。 スヌーピング ポリシー（データ グリーニングがイネーブル）をインターフェイスに適用します。ポートと、そのポートに適用されるポリシーを指定します。 （注） スヌーピング ポリシーで **data-glean** をイネーブルにした場合は、そのポリシーをVLANではなく、インターフェイスに適用する必要があります。
ステップ 4	**show ipv6 snooping policy** *policy-name*	スヌーピング ポリシー設定を表示します。
ステップ 5	**show ipv6 neighbors binding**	スヌーピング ポリシーによって入力されたバインディング テーブル エントリを表示します。

デバイスでの IPv6 ネイバー探索マルチキャスト抑制ポリシーのアタッチ方法

IPv6ネイバー探索マルチキャスト抑制ポリシーをデバイスにアタッチするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 nd suppress policy** *policy-name*
4. **mode dad-proxy**
5. **mode full-proxy**
6. **mode mc-proxy**

手順の詳細

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

	コマンドまたはアクション	目的
	`Device> enable`	
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 nd suppress policy** *policy-name*	ネイバー探索抑制ポリシー名を定義して、ネイバー探索抑制ポリシー コンフィギュレーション モードを開始します。
ステップ 4	**mode dad-proxy**	IPv6 DADプロキシ モードでネイバー探索抑制をイネーブルにします。
ステップ 5	**mode full-proxy**	プロキシマルチキャストおよびユニキャストのネイバー送信要求メッセージに対するネイバー探索抑制をイネーブルにします。
ステップ 6	**mode mc-proxy**	プロキシマルチキャスト ネイバー送信要求メッセージに対するネイバー探索抑制をイネーブルにします。

インターフェイスでの *IPv6* ネイバー探索マルチキャスト抑制ポリシーのアタッチ方法

IPv6 ネイバー探索マルチキャスト抑制ポリシーをインターフェイスにアタッチするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかの作業を実行します。
   - **interface** *type number*
   - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]

     または

   - **vlan configuration** *vlan-id*
   - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]
4. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかの作業を実行します。 • **interface** *type number* • **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]] または • **vlan configuration** *vlan-id* • **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]	インターフェイスのタイプと番号を指定し、デバイスをインターフェイス コンフィギュレーション モードにします。 IPv6 ネイバー探索マルチキャスト ポリシーをインターフェイスまたは VLAN にアタッチします。
ステップ 4	**exit**	インターフェイス コンフィギュレーション モードを終了します。

レイヤ *2 EtherChannel* インターフェイスへの *IPv6* ネイバー探索マルチキャスト抑制ポリシーのアタッチ方法

IPv6 ネイバー探索マルチキャスト抑制ポリシーを EtherChannel インターフェイスにアタッチするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかの作業を実行します。
   • **interface port-channel** *port-channel-number*
   • **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]
   または
   • **vlan configuration** *vlan-id*
   • **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]

4. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかの作業を実行します。 • **interface port-channel** *port-channel-number* • **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]] または • **vlan configuration** *vlan-id* • **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]	インターフェイスのタイプとポート番号を指定し、スイッチをポート チャネル コンフィギュレーション モードにします。 IPv6 ネイバー探索マルチキャスト ポリシーをインターフェイスまたは VLAN にアタッチします。
ステップ 4	**exit**	インターフェイス コンフィギュレーション モードを終了します。

IPv6 DHCP ガード ポリシーの設定方法

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 dhcp guard policy** *policy-name*
4. [**default** | **device-role** [**client** | **server**] |**no** | **exit** | **trusted-port**]
5. **exit**
6. 次のいずれかの作業を実行します。
   - **interface** *type number*
   - **ipv6 dhcp guard attach-policy** *policy-name*

     または
   - **vlan configuration** *vlan-id*
   - **ipv6 dhcp guard attach-policy** *policy-name*

7. **show ipv6 dhcp guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 dhcp guard policy** *policy-name*	DHCPv6 ガード ポリシー名を指定し、DHCPv6 ガード ポリシー コンフィギュレーション モードを開始します。
ステップ 4	[**default** | **device-role** [**client** | **server**] |**no** | **exit** | **trusted-port**]	（任意）特定の役割のデバイスからのものではないポート上の DHCPv6 応答および DHCPv6 アドバタイズメントをフィルタします。デフォルトは **client** です。 • **client**：デフォルト値。アタッチされたデバイスがクライアントであることを指定します。サーバー メッセージにはこのポートで破棄されます。 • **server**：適用されたデバイスが DHCPv6 サーバーであることを指定します。このポートでは、サーバー メッセージが許可されます。 （任意）**trusted-port**：ポートを信頼モードに設定します。このポートでは、これ以上のポリシングは実行されません。 （注） 信頼できるポートを設定した場合、device-role オプションは使用できません。
ステップ 5	**exit**	DHCP ガードポリシー グローバル コンフィギュレーション モードを終了します。
ステップ 6	次のいずれかの作業を実行します。 • **interface** *type number* • **ipv6 dhcp guard attach-policy** *policy-name* または	インターフェイスのタイプおよび番号を指定し、インターフェイス コンフィギュレーション モードを開始します。 DHCP ガード ポリシーをインターフェイスまたは VLAN に適用します。

	コマンドまたはアクション	目的
	• **vlan configuration** *vlan-id* • **ipv6 dhcp guard attach-policy** *policy-name*	
ステップ 7	**show ipv6 dhcp guard policy** *policy_name*	DHCP ガード ポリシー設定を表示します。

DHCPv6 ガード設定の例

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy pol1
 device-role server
 match server access-list acl1
 match reply prefix-list abc
 preference min 0
 preference max 255
 trusted-port
interface GigabitEthernet 0/2/0
 switchport
 ipv6 dhcp guard attach-policy pol1 vlan add 1
 vlan configuration 1
  ipv6 dhcp guard attach-policy pol1
show ipv6 dhcp guard policy pol1
```

IPv6 ソース ガードの設定方法

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 source-guard policy** *policy_name*
4. [**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }]
5. **ipv6 source-guard** [**attach-policy** *policy-name*]
6. **exit**
7. **show ipv6 source-guard policy***policy_name*

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 source-guard policy** *policy_name*	IPv6 ソース ガード ポリシー名を指定し、IPv6 ソース ガード ポリシー コンフィギュレーション モードを開始します。
ステップ 4	[**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }]	IPv6 ソース ガード ポリシーを定義します。 • **deny global-autoconf**：自動設定されたグローバル アドレスからのデータ トラフィックを拒否します。これは、リンク上のすべてのグローバル アドレスが DHCP によって割り当てられている際に、管理者が、自己設定されたアドレスを持つホストによるトラフィックの送信をブロックしたい場合に役立ちます。 • **permit link-local**：リンクローカル アドレスから送信されたすべてのデータ トラフィックを許可します。
ステップ 5	**ipv6 source-guard**［**attach-policy** *policy-name*］	ポリシー名を指定します。 （任意）**attach-policy** *policy-name*：ポリシー名に基づいてフィルタリングします。
ステップ 6	**exit**	ソース ガード ポリシー コンフィギュレーション モードを終了します。
ステップ 7	**show ipv6 source-guard policy***policy_name*	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

デフォルト ルータ プリファレンス（DRP）の設定

ルータアドバタイズメント（RA）メッセージは、**ipv6 nd router-preference** インターフェイス コンフィギュレーション コマンドによって設定されるデフォルト ルータ プリファレンス（DRP）とともに送信されます。DRP が設定されていない場合は、RA はプリファレンス「中」とともに送信されます。

リンク上の 2 つのルータが等価ではあっても、等コストではないルーティングを提供する可能性がある場合、およびポリシーでホストがいずれかのルータを選択するよう指示された場合は、DRP が有効です。

IPv6 の DRP の設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing IPv6 Addresses and Basic Connectivity」の章を参照してください。

インターフェイス上のルータの DRP を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	インターフェイス コンフィギュレーション モードを開始して、DRP を指定するレイヤ 3 インターフェイスを特定します。
ステップ 4	**ipv6 nd router-preference** {**high** | **medium** | **low**} 例： スイッチ(config-if)# **ipv6 nd router-preference medium**	スイッチ インターフェイス上のルータに DRP を指定します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ipv6 interface** 例： スイッチ# **show ipv6 interface**	設定を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 ICMP レート制限の設定

ICMP レート制限はデフォルトで有効です。エラー メッセージのデフォルト間隔は 100 ミリ秒、デフォルト バケット サイズ（バケットに格納される最大トークン数）は 10 です。

ICMP のレート制限パラメータを変更するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 icmp error-interval** *interval* [*bucketsize*] 例： `スイッチ(config)# ipv6 icmp error-interval 50 20`	IPv6 ICMP エラー メッセージの間隔とバケット サイズを設定します。 • *interval*：バケットに追加されるトークンの間隔（ミリ秒）。指定できる範囲は 0 ～ 2147483647 ミリ秒です。 • *bucketsize*：（任意）バケットに格納される最大トークン数。指定できる範囲は 1 ～ 200 です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show ipv6 interface** [*interface-id*] 例： `スイッチ# show ipv6 interface gigabitethernet0/1`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 用のシスコ エクスプレス フォワーディングおよび分散型シスコ エクスプレス フォワーディングの設定

シスコ エクスプレス フォワーディングは、ネットワークパフォーマンスを最適化するためのレイヤ 3 IP スイッチングテクノロジーです。シスコ エクスプレス フォワーディングには高度な IP 検索および転送アルゴリズムが実装されているため、レイヤ 3 スイッチングのパフォーマンスを最大化できます。高速スイッチング ルート キャッシュよりも CPU にかかる負担が少ないため、CEF はより多くの CPU 処理能力をパケット転送に振り分けることができます。IPv4 用のシスコ エクスプレス フォワーディングおよび分散型シスコ エクスプレス フォワーディングはデフォルトで有効になっています。IPv6 用のシスコ エクスプレス フォワーディングおよび分散型シスコ エクスプレス フォワーディングはデフォルトでは無効になっていますが、IPv6 ルーティングを設定すると自動的に有効になります。

IPv6 ルーティングの設定を解除すると IPv6 用のシスコ エクスプレス フォワーディングおよび分散型シスコ エクスプレス フォワーディングは自動的に無効になります。IPv6 用のシスコ エクスプレス フォワーディングおよび分散型シスコ エクスプレス フォワーディングを設定で無効にすることはできません。IPv6 の状態を確認するには、**show ipv6 cef**特権 EXEC コマンドを入力します。

IPv6 ユニキャストパケットをルーティングするには、最初に **ipv6 unicast-routing** グローバル コンフィギュレーション コマンドを使用して、IPv6 ユニキャストパケットの転送をグローバルに設定してから、**ipv6 address** インターフェイス コンフィギュレーション コマンドを使用して、特定のインターフェイスに IPv6 アドレスおよび IPv6 処理を設定する必要があります。

シスコ エクスプレス フォワーディングおよび分散型シスコ エクスプレス フォワーディングの設定の詳細については、Cisco.com の『*Cisco IOS IPv6 Configuration Library*』を参照してください。

IPv6 のスタティック ルーティングの設定

スタティック IPv6 ルーティングの設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing Static Routes for IPv6」の章を参照してください。

スタティック IPv6 ルーティングを設定するには、次の手順を実行します。

始める前に

ip routing グローバル コンフィギュレーション コマンドを使用してルーティングを有効にし、**ipv6 unicast-routing** グローバル コンフィギュレーション コマンドを使用して IPv6 パケットの転送を有効にします。また、インターフェイスに IPv6 アドレスを設定して少なくとも 1 つのレイヤ 3 インターフェイス上で IPv6 を有効にする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address* | *interface-id* [*ipv6-address*]} [*administrative distance*] 例： スイッチ(config)# **ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130**	スタティック IPv6 ルートを設定します。 • *ipv6-prefix*：スタティック ルートの宛先となる IPv6 ネットワーク。スタティック ホスト ルートを設定する場合は、ホスト名も設定できます。 • */prefix length*：IPv6 プレフィックスの長さ。プレフィックス（アドレスのネットワーク部分）を構成するアドレスの上位連続ビット数を示す 10 進値です。10 進数値の前にスラッシュ記号が必要です。 • *ipv6-address*：指定したネットワークに到達するために使用可能なネクスト ホップの IPv6 アドレス。ネクスト ホップの IPv6 アドレスを直接接続する必要はありません。再帰処理が実行されて、直接接続されたネクスト ホップの IPv6 アドレスが検出されます。このアドレスはRFC 2373 に記載された形式（16 ビット値を使用したコロン区切りの 16 進表記で指定）で設定する必要があります。 • *interface-id*：Point-To-Point（ポイントツーポイント）インターフェイスおよびブロードキャスト インターフェイスからのダイレクト スタティック ルートを指定します。ポイントツーポイント インターフェイスの場合、ネクストホップの IPv6 アドレスを指定する必要はありません。ブロードキャスト インターフェイスの場合は、常にネクストホップの IPv6 アドレスを指定するか、または指定したプレフィックスをリンクに割り当てて、リンクローカルアドレスをネ

	コマンドまたはアクション	目的
		クストホップとして指定する必要があります。パケットの送信先となるネクストホップのIPv6アドレスを指定することもできます。 （注） リンクローカルアドレスをネクストホップとして使用する場合は、*interface-id* を指定する必要があります（リンクローカルのネクストホップを隣接ルータに設定する必要もあります）。 • *administrative distance*：（任意）アドミニストレーティブ ディスタンス。指定できる範囲は 1 ～ 254 です。デフォルト値は 1 で、この場合、接続されたルートを除くその他のどのルート タイプよりも、スタティック ルートが優先します。フローティング スタティック ルートを設定する場合は、ダイナミック ルーティング プロトコルよりも大きなアドミニストレーティブ ディスタンスを使用します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	次のいずれかを使用します。 • **show ipv6 static** [*ipv6-address* | *ipv6-prefix/prefix length*] [**interface** *interface-id*] [**detail**]][**recursive**] [**detail**] • **show ipv6 route static** [*updated*] 例： `スイッチ# show ipv6 static 2001:0DB8::/32 interface gigabitethernet2/0/1` または `スイッチ# show ipv6 route static`	IPv6 ルーティング テーブルの内容を表示して、設定を確認します。 • **interface** *interface-id*：（任意）出力インターフェイスとして指定されたインターフェイスを含むスタティック ルートのみを表示します。 • **recursive**：（任意）再帰スタティック ルートのみを表示します。**recursive** キーワードは **interface** キーワードと相互に排他的です。ただし、コマンド構文に IPv6 プレフィックスが指定されているかどうかに関係なく、使用できます。 • **detail**：（任意）次に示す追加情報を表示します。 • 有効な再帰ルートの場合、出力パスセットおよび最大分解深度 • 無効なルートの場合、ルートが無効な理由

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

IPv6 RIP の設定

IPv6 の RIP ルーティングの設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing RIP for IPv6」の章を参照してください。

IPv6 の RIP ルーティングを設定するには、次の手順を実行します。

始める前に

IPv6 RIP を実行するようにスイッチを設定する前に、**ip routing** グローバル コンフィギュレーション コマンドを使用してルーティングを有効にし、**ipv6 unicast-routing** グローバル コンフィギュレーション コマンドを使用して IPv6 パケットの転送を有効にして、IPv6 RIP を有効にするレイヤ 3 インターフェイス上で IPv6 を有効にする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 router rip** *name* 例： スイッチ(config)# **ipv6 router rip cisco**	IPv6 RIP ルーティング プロセスを設定し、このプロセスに対してルータ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**maximum-paths** *number-paths* 例： `スイッチ(config-router)# maximum-paths 6`	（任意）IPv6 RIP がサポートできる等コスト ルートの最大数を定義します。指定できる範囲は 1 ～ 32 で、デフォルトは 16 ルートです。
ステップ 5	**exit** 例： `スイッチ(config-router)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。
ステップ 7	**ipv6 rip** *name* **enable** 例： `スイッチ(config-if)# ipv6 rip cisco enable`	指定された IPv6 RIP ルーティング プロセスをインターフェイス上で有効にします。
ステップ 8	**ipv6 rip** *name* **default-information** {**only** | **originate**} 例： `スイッチ(config-if)# ipv6 rip cisco default-information only`	（任意）IPv6 デフォルト ルート（::/0）を RIP ルーティング プロセス アップデートに格納して、指定インターフェイスから送信します。 （注） 任意のインターフェイスから IPv6 デフォルト ルート（::/0）を送信したあとに、ルーティング ループが発生しないようにするために、ルーティング プロセスは任意のインターフェイスで受信したすべてのデフォルト ルートを無視します。 • **only**：このインターフェイスから送信するアップデートに、デフォルト ルートを格納し、その他のすべてのルートを含めない場合に選択します。 • **originate**：このインターフェイスから送信するアップデートに、デフォルト ルートおよびその他のすべてのルートを格納する場合に選択します。

	コマンドまたはアクション	目的
ステップ 9	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 10	次のいずれかを使用します。 • **show ipv6 rip** [*name*] [**interface** *interface-id*] [**database**] [**next-hops**] • **show ipv6 rip** 例： `スイッチ# show ipv6 rip cisco interface gigabitethernet 2/0/1` または `スイッチ# show ipv6 rip`	• 現在の IPv6 RIP プロセスに関する情報を表示します。 • IPv6 ルーティング テーブルの現在の内容を表示します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 OSPF の設定

IPv6 の OSPF ルーティングの設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing OSPF for IPv6」の章を参照してください。

IPv6 の OSPF ルーティングを設定するには、次の手順を実行します。

始める前に

ネットワークでは、IPv6 の OSPF をカスタマイズできます。ただし、IPv6 の OSPF のデフォルト設定は、ほとんどのお客様および機能の要件を満たします。

次の注意事項に従ってください。

- IPv6 コマンドのデフォルト設定を変更する場合は注意してください。デフォルト設定を変更すると、IPv6 ネットワークの OSPF に悪影響が及ぶことがあります。
- インターフェイスで IPv6 OSPF を有効にする前に、**ip routing** グローバル コンフィギュレーション コマンドを使用してルーティングを有効にし、**ipv6 unicast-routing** グローバル コンフィギュレーション コマンドを使用して IPv6 パケットの転送を有効にし、IPv6 OSPF を有効にするレイヤ 3 インターフェイスで IPv6 を有効にする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 router ospf** *process-id* 例： スイッチ(config)# **ipv6 router ospf 21**	プロセスに対して OSPF ルータ コンフィギュレーション モードを有効にします。プロセス ID は、IPv6 OSPF ルーティング プロセスを有効にする場合に管理上割り当てられる番号です。この ID はローカルに割り当てられ、1 ～ 65535 の正の整数を指定できます。
ステップ 4	**area** *area-id* **range** {*ipv6-prefix/prefix length*} [**advertise** | **not-advertise**] [**cost** *cost*] 例： スイッチ(config)# **area .3 range 2001:0DB8::/32 not-advertise**	（任意）エリア境界でルートを統合および集約します。 • *area-id*：ルートをサマライズするエリアの ID。10 進数または IPv6 プレフィックスのどちらかを指定できます。 • *ipv6-prefix/prefix length*：宛先 IPv6 ネットワーク、およびプレフィックス（アドレスのネットワーク部分）を構成するアドレスの上位連続ビット数を示す 10 進数。10 進値の前にスラッシュ（/）を付加する必要があります。 • **advertise**：（任意）アドバタイズするアドレス範囲ステータスを設定し、タイプ 3 のサマリー リンクステート アドバタイズメント（LSA）を生成します。 • **not-advertise**：（任意）アドレス範囲ステータスを DoNotAdvertise に設定します。Type3 サマリー LSA は抑制され、コンポーネント ネットワークは他のネットワークから隠された状態のままです。 • **cost** *cost*：（任意）現在のサマリー ルートのメトリックまたはコストを設定します。宛先への最短パスを判別する場合に、OSPF SPF 計算で

	コマンドまたはアクション	目的
		使用します。指定できる値は 0 ～ 16777215 です。
ステップ 5	**maximum paths** *number-paths* 例： `スイッチ(config)# maximum paths 16`	（任意）IPv6 OSPF がルーティング テーブルに入力する必要がある、同じ宛先への等コスト ルートの最大数を定義します。指定できる範囲は 1 ～ 32 で、デフォルトは 16 です。
ステップ 6	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 7	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ3インターフェイスを指定します。
ステップ 8	**ipv6 ospf** *process-id* **area** *area-id* [**instance** *instance-id*] 例： `スイッチ(config-if)# ipv6 ospf 21 area .3`	インターフェイスで IPv6 の OSPF を有効にします。 • **instance** *instance-id*：（任意）インスタンス ID
ステップ 9	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 10	次のいずれかを使用します。 • **show ipv6 ospf** [*process-id*] [*area-id*] **interface** [*interface-id*] • **show ipv6 ospf** [*process-id*] [*area-id*] 例： `スイッチ# show ipv6 ospf 21 interface gigabitethernet2/0/1` または `スイッチ# show ipv6 ospf 21`	• OSPF インターフェイスに関する情報を表示します。 • OSPF ルーティング プロセスに関する一般情報を表示します。

	コマンドまたはアクション	目的
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

OSPFv3 高速コンバージェンスに対する LSA および SPF タイマーの調整

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 router ospf***process-id*
4. **timers lsa arrival** *milliseconds*
5. **timers pacing flood***milliseconds*
6. **timers pacing lsa-group***seconds*
7. **timers pacing retransmission***milliseconds*
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 router ospf***process-id*	OSPFv3 ルータ コンフィギュレーション モードをイネーブルにします。
ステップ 4	**timers lsa arrival** *milliseconds*	ソフトウェアが OSPFv3 ネイバーから同じ LSA を受け入れる最小間隔を設定します。
ステップ 5	**timers pacing flood***milliseconds*	LSA フラッド パケット ペーシングを設定します。

	コマンドまたはアクション	目的
ステップ 6	**timers pacing lsa-group***seconds*	OSPFv3 LSA を収集してグループ化し、リフレッシュ、チェックサム、またはエージングを行う間隔を変更します。
ステップ 7	**timers pacing retransmission***milliseconds*	OSPFv3 での LSA 再送信パケット ペーシングを設定します。
ステップ 8	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

OSPFv3 高速コンバージェンスに対する LSA および SPF スロットリングの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 router ospf***process-id*
4. **timers throttle spf** *spf-start spf-hold spf-max-wait*
5. **timers throttle lsa***start-intervalhold-intervalmax-interval*
6. **timers lsa arrival***milliseconds*
7. **timers pacing flood***milliseconds*
8. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**ipv6 router ospf***process-id*	OSPFv3 ルータ コンフィギュレーション モードをイネーブルにします。
ステップ 4	**timers throttle spf** *spf-start spf-hold spf-max-wait*	SPF スロットリングをオンにします。
ステップ 5	**timers throttle lsa***start-intervalhold-intervalmax-interval*	OSPFv3 LSA 生成に対するレート制限値を設定します。
ステップ 6	**timers lsa arrival***milliseconds*	ソフトウェアが OSPFv3 ネイバーから同じ LSA を受け入れる最小間隔を設定します。
ステップ 7	**timers pacing flood***milliseconds*	LSA フラッド パケット ペーシングを設定します。
ステップ 8	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

IPv6 の EIGRP の設定

IPv6 EIGRP を実行するようにスイッチを設定する前に、 **ip routing global configuration** グローバル コンフィギュレーション コマンドを入力してルーティングを有効にし、**ipv6 unicast-routing global** グローバル コンフィギュレーション コマンドを入力して IPv6 パケットの転送を有効にし、IPv6 EIGRP を有効にするレイヤ 3 インターフェイス上で IPv6 を有効にします。

明示的なルータ ID を設定するには、**show ipv6 eigrp** コマンドを使用して設定済みのルータ ID を確認してから、**router-id** コマンドを使用します。

EIGRP IPv4 の場合と同様に、EIGRPv6 を使用して EIGRP IPv6 インターフェイスを指定し、これらのサブセットを受動インターフェイスとして選択できます。**passive-interface** コマンドを使用してインターフェイスをパッシブに設定してから、選択したインターフェイスで **no passive-interface** コマンドを使用してこれらのインターフェイスをアクティブにします。受動インターフェイスでは、EIGRP IPv6 を設定する必要がありません。

設定手順の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing EIGRP for IPv6」の章を参照してください。

IPv6 の HSRP の設定

IPv6 の Hot Standby Router Protocol（HSRP）は、任意の単一のルータのアベイラビリティに依存せず、ルーティング IPv6 トラフィックにルーティング冗長性を提供します。

スイッチで IPv6 の HSRP がイネーブルである場合、IPv6 ホストは IPv6 ネイバー探索ルータのアドバタイズメント メッセージから使用可能な IPv6 ルータを学習します。HSRP IPv6 グループには、HSRP グループ番号に基づいて作成される仮想 MAC アドレスがあります。グループ

には、デフォルトで、HSRP 仮想 MAC アドレスに基づいて作成される仮想 IPv6 リンクローカル アドレスがあります。HSRP グループがアクティブな場合、定期的なメッセージが HSRP 仮想 IPv6 リンクローカル アドレスに送信されます。

IPv6 の HSRP を設定する場合、インターフェイス上で HSRP version 2（HSRPv2）をイネーブルにする必要があります。

（注） IPv6 の HSRP グループを設定する前に、**ipv6 unicast-routing** グローバル コンフィギュレーション コマンドを使用して IPv6 パケットの転送をイネーブルにし、IPv6 の HSRP グループを設定するインターフェイス上で IPv6 をイネーブルにする必要があります。

HSRP バージョン 2 のイネーブル化

IPv6 の HSRP の設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Configuring First Hop Redundancy Protocols in IPv6」の章を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始して、スタンバイ バージョンを指定するレイヤ 3 インターフェイスを入力します。
ステップ 3	**standby version** {**1** | **2**} 例： `スイッチ(config-if)# standby version 2`	HSRP バージョンを設定します。HSRP バージョンを変更するには、**2** を入力します。デフォルトは 1 です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

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

IPv6 の HSRP グループのイネーブル化

ここでは、レイヤ 3 インターフェイス上で IPv6 の HSRP を作成するかイネーブルにする方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	インターフェイス コンフィギュレーション モードを開始し、IPv6 の HSRP をイネーブルにするレイヤ 3 インターフェイスを入力します。
ステップ 3	**standby** [*group-number*] **ipv6** {*link-local-address* | **autoconfig**} 例： スイッチ(config-if)# **standby 2 ipv6 auto config**	IPv6 グループの HSRP を作成（またはイネーブルに）します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 4095 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • ホット スタンバイ ルータ インターフェイスのリンクローカルアドレスを入力するか、リンクローカル プレフィックスおよび変更された

	コマンドまたはアクション	目的
		EUI-64 形式のインターフェイス ID から自動的に生成されるリンクローカルアドレスをイネーブルにします。この場合、EUI-64 インターフェイス ID は、関連する HSRP 仮想 MAC アドレスから作成されます。
ステップ 4	**standby** [*group-number*] **preempt** [**delay** {**minimum** *seconds* | **reload** *seconds* | **sync** *seconds*}] 例： `スイッチ(config-if)# standby 2 preempt delay reload 0`	ルータを **preempt** に設定し、ローカルルータのプライオリティがアクティブルータよりも高い場合は、アクティブルータとして制御を行います。 • （任意）*group-number*：コマンドが適用されるグループ番号です。 • （任意）**delay**：ローカルルータがアクティブルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒です（1 時間）。デフォルトは 0 です（引き継ぐまで遅延がない）。 • （任意）**reload**：リロード後のプリエンプション遅延（秒）を設定します。遅延時間は、ルータのリロード後の最初のインターフェイスアップ イベントに対してだけ適用されます。 • （任意）**sync**：IP 冗長クライアントの最大同期化時間（秒）を設定します。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 5	**standby** [*group-number*] **priority** *priority* 例： `スイッチ(config-if)# standby 2 priority 200`	アクティブルータを選択するときに使用される **priority**値を設定します。指定できる範囲は 1 ～ 255 です。デフォルト プライオリティは 100 です。最大の値が、最高のプライオリティを表します。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show standby** [*interface-id* [*group-number*]] 例：	設定を確認します。

	コマンドまたはアクション	目的
	スイッチ# **show standby gigabitethernet 1/0/1 2**	
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

Multi-VRF CE の設定

スイッチ上で IP サービスまたは拡張 IP サービス フィーチャ セットが稼働している場合、スイッチはカスタマー エッジ（CE）デバイスの複数の VRF ルーティング/転送（Multi-VRF）インスタンスをサポートします（Multi-VRF CE）。サービス プロバイダは、Multi-VRF CE により、重複する IP アドレスで複数の VPN をサポートできます。

（注） スイッチでは、VPNのサポートのためにマルチプロトコル ラベル スイッチング（MPLS）が使用されません。

IPv6 マルチキャスト ルーティングは VRF 関連インターフェイスではサポートされません。

Multi-VRF CE のデフォルト設定

表 *36 : VRF* のデフォルト設定

機能	デフォルト設定
VRF	ディセーブル。VRF は定義されていません。
マップ	インポート マップ、エクスポート マップ、ルート マップは定義されていません。
転送テーブル	インターフェイスのデフォルトは、グローバル ルーティング テーブルです。

VRF の設定

コマンドの完全な構文と使用方法については、このリリースに対応するスイッチ コマンド リファレンスおよび『*Cisco IOS Switching Services Command Reference*』を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ipv6 unicast-routing** 例： `スイッチ(config)# ipv6 unicast routing`	IPv6 ユニキャスト ルーティングをイネーブルにします。
ステップ 3	**vrf definition** *vrf-name* 例： `スイッチ(config)# vrf definition vpn1`	VRF 名を指定し、VRF コンフィギュレーション モードを開始します。
ステップ 4	**address family** *ipv6* 例： `スイッチ(config)# address family ipv6`	IPv6 アドレス ファミリを指定し、アドレス ファミリ コンフィギュレーション モードを開始します。
ステップ 5	**rd** *route-distinguisher* 例： `スイッチ(config-vrf)# rd 100:2`	ルート識別子を指定して VRF テーブルを作成します。AS 番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。
ステップ 6	**route-target** {**export** | **import** | **both**} *route-target-ext-community* 例： `スイッチ(config-vrf)# route-target both 100:2`	指定された VRF のインポート、エクスポート、またはインポートおよびエクスポート ルート ターゲット コミュニティのリストを作成します。AS システム番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。*route-target-ext-community* は、ステップ 4 で入力した *route-distinguisher* と同一にする必要があります。
ステップ 7	**import map** *route-map* 例： `スイッチ(config-vrf)# import map importmap1`	（任意）VRF にルート マップを対応付けます。
ステップ 8	**interface** *interface-id* 例： `スイッチ(config-vrf)# interface gigabitethernet 1/0/1`	VRF に関連付けるレイヤ 3 インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスにはルーテッド ポートまたは SVI を設定できます。

	コマンドまたはアクション	目的
ステップ 9	**vrf forwarding** *vrf-name* 例： `スイッチ(config-if)# vrf forwarding vpn1`	VRF をレイヤ 3 インターフェイスに対応付けます。
ステップ 10	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 11	**show vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] 例： `スイッチ# show vrf interfaces vpn1`	設定を確認します。設定した VRF に関する情報を表示します。
ステップ 12	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VRF 認識サービスの設定

次のサービスは、VRF 認識です。

- ARP
- ping
- 簡易ネットワーク管理プロトコル（SNMP）
- ホットスタンバイ ルータ プロトコル（HSRP）
- ユニキャスト RPF（uRPF）
- Syslog
- traceroute
- FTP および TFTP

（注） このスイッチでは、ユニキャスト RPF（uRPF）およびネットワーク タイム プロトコル（NTP）に対して VRF 認識のサービスはサポートされません。

ネイバー探索用 VRF 認識サービスの設定

コマンドの完全な構文と使用方法については、このリリースに対応するスイッチ コマンド リファレンスおよび『*Cisco IOS Switching Services Command Reference, Release 12.4*』を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**show ipv6 neighbors vrf***vrf-name* 例： `スイッチ# show ipv6 neighbors vrf vpn1`	指定された VRF 内の ARP テーブルを表示します。

ping 用 VRF 認識サービスの設定

コマンドの完全な構文と使用方法については、このリリースに対応するスイッチ コマンド リファレンスおよび『*Cisco IOS Switching Services Command Reference, Release*』を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**ping vrf***vrf-name***ipv6***ipv6-address* 例： `スイッチ# ping vrf vpn1 ipv6`	指定された VRF 内の ARP テーブルを表示します。

HSRP 用 VRF 認識サービスの設定

コマンドの完全な構文と使用方法については、このリリースに対応するスイッチ コマンド リファレンスおよび『*Cisco IOS Switching Services Command Reference, Release 12.4*』を参照してください。

手順

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

	コマンドまたはアクション	目的
ステップ 2	**interface** *interface-id* 例： スイッチ# **interface gigabitethernet1/0/1**	インターフェイス コンフィギュレーション モードを開始し、HSRP をイネーブルにするレイヤ 3 インターフェイスを入力します。
ステップ 3	**no switchport** 例： スイッチ# no switchport	レイヤ 2 コンフィギュレーション モードからインターフェイスを削除します（物理インターフェイスの場合）。
ステップ 4	**vrf forwarding***vrf-name* 例： スイッチ# vrf forwarding vpn1	インターフェイス上で VRF を設定します。
ステップ 5	**ipv6 address***ipv6 address* 例： スイッチ# ipv6 address 2001::DB8:1/64	インターフェイスの IPv6 アドレスを入力します。
ステップ 6	**standby 1 ipv6***ipv6 address* 例： スイッチ# standby 1 ipv6 2001::DB8:1/64	HSRP をイネーブルにし、仮想 IP アドレスを設定します。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

traceroute 用 VRF 認識サービスの設定

コマンドの完全な構文と使用方法については、このリリースに対応するスイッチ コマンド リファレンスおよび『*Cisco IOS Switching Services Command Reference, Release*』を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**traceroute vrf***vrf-nameipv6-address* 例： スイッチ# traceroute vrf**vpn1 2001::DB8:1/64**	宛先アドレスを取得する VPN VRF の名前を指定します。

FTP および TFTP 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip ftp source-interface** *interface-type interface-number* 例： スイッチ(config)# ip ftp source-interface gigabitethernet 1/0/2	FTP 接続の発信元 IP アドレスを指定します。
ステップ 3	**end** 例： スイッチ(config)#end	特権 EXEC モードに戻ります。
ステップ 4	**configure terminal** 例： スイッチ# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 5	**ip tftp source-interface** *interface-type interface-number* 例： スイッチ(config)# ip tftp source-interface gigabitethernet 1/0/2	TFTP 接続用の送信元 IP アドレスを指定します。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**end** 例： スイッチ(config)#end	特権 EXEC モードに戻ります。

VPN ルーティング セッションの設定

VPN 内のルーティングは、サポートされるルーティング プロトコル（OSPF、EIGRP、または BGP）、またはスタティック ルーティングで設定できます。ここで説明する設定はOSPF のものですが、その他のプロトコルでも手順は同じです。

（注） VRF インスタンス内で EIGRP ルーティングプロセスが実行されるように設定するには、**autonomous-system** *autonomous-system-number* アドレス ファミリ コンフィギュレーション モード コマンドを入力して、自律システム番号を設定する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router ospfv3** *process-id* 例： `スイッチ(config)# router ospfv3 1`	OSPF ルーティングをイネーブルにして VPN 転送テーブルを指定し、ルータ コンフィギュレーション モードを開始します。
ステップ 3	**router** *router-id* 例： `スイッチ(config)# router router-id`	この OSPFv3 プロセスの OSPF ルータ ID を IP アドレス形式で指定します。
ステップ 4	**log-adjacency-changes** 例： `スイッチ(config-router)# log-adjacency-changes`	（任意）隣接ステートの変更を記録します。これは、デフォルトの状態です。
ステップ 5	**address-family ipv6 unicast vrf** *vrf-name* 例： `スイッチ(config-router)# address-family ipv6 unicast vrf vpn1`	その VRF に対してアドレス ファミリ コマンド モードを開始します。
ステップ 6	**area** *area-id normal* 例： `スイッチ(config-router)# area 2`	OSPFv3 エリア パラメータとタイプを指定します。

	コマンドまたはアクション	目的
ステップ 7	**redistribute bgp** *autonomous-system-number* 例： `スイッチ(config-router)# redistribute bgp 10`	BGP ルーティング プロセスから OSPF ルーティング プロセスにルートを再配布します。
ステップ 8	**end** 例： `スイッチ(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 9	**show ospfv3 vrf** *vrf-name* 例： `スイッチ# show ospfv3 vrf vpn1`	OSPFv3 ネットワークの設定を確認します。
ステップ 10	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

BGP PE/CE ルーティング セッションの設定

手順

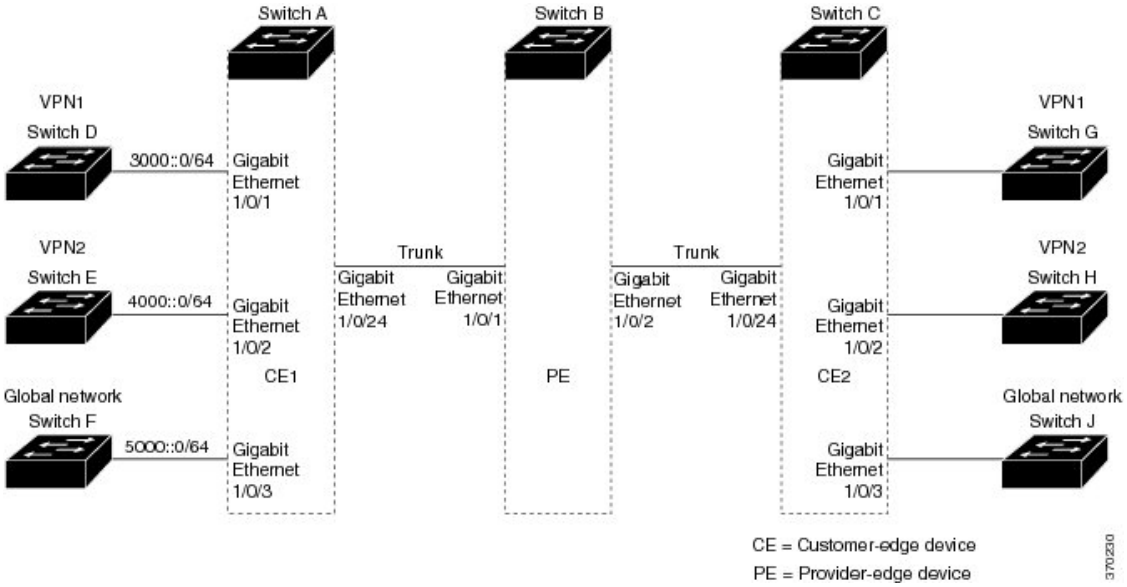
	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router bgp** *autonomous-system-number* 例： `スイッチ(config)# router bgp 2`	その他の BGP ルータに AS 番号を渡す BGP ルーティング プロセスを設定し、ルータ コンフィギュレーション モードを開始します。
ステップ 3	**bgp router id** *router-id* 例： `スイッチ(config)# bgp router-id`	（任意）固定 32 ビット ルータ ID を、BGP を実行するローカル ルータの ID として設定します。
ステップ 4	**redistribute ospf** *process-id* 例：	OSPF 内部ルートを再配布するようにスイッチを設定します。

	コマンドまたはアクション	目的
	スイッチ(config-router)# redistribute ospf 1	
ステップ 5	**address-family ipv6 vrf** *vrf-name* 例： スイッチ(config-router)# address-family ipv6 vrf vpn1	PE/CE ルーティング セッションの BGP パラメータを定義し、VRF アドレス ファミリ モードを開始します。
ステップ 6	**network** *ipv6 network-number* 例： スイッチ(config-router)# network ipv6 255.255.255.0	BGP を使用して IPv6 ネットワーク番号をアナウンスするように指定します。
ステップ 7	**neighbor** *ipv6 address* **remote-as** *as-number* 例： スイッチ(config-router)# neighbor 10.1.1.2 remote-as 2	PE と CE ルータの間の BGP セッションを定義します。
ステップ 8	**neighbor** *address* **activate** 例： スイッチ(config-router)# neighbor 10.2.1.1 activate	IPv4 アドレス ファミリのアドバタイズメントをアクティブ化します。
ステップ 9	**end** 例： スイッチ(config-router)# end	特権 EXEC モードに戻ります。
ステップ 10	**show bgp vrf** *vrf-name* 例： スイッチ# show ip bgp ipv4 neighbors	VRF の BGP 設定を確認します。
ステップ 11	**copy running-config startup-config** 例： スイッチ# copy running-config startup-config	（任意）コンフィギュレーション ファイルに設定を保存します。

Multi-VRF CE の設定例

VPN1、VPN2、およびグローバル ネットワークで使用されるプロトコルは OSPF です。CE/PE 接続には BGP が使用されます。図のあとに続く出力は、スイッチを CE スイッチ A として設定する例、およびカスタマー スイッチ D と E の VRF 設定を示しています。CE スイッチ C とその他のカスタマー スイッチを設定するコマンドは含まれていませんが、内容は同様です。

図 *14 : Multi-VRF CE* の設定例



スイッチ A では、ルーティングをイネーブルにして VRF を設定します。

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# ipv6 unicast-routing
スイッチ(config)# vrf definition v11
スイッチ(config-vrf)# rd 11:1
スイッチ(config-vrf)# address-family ipv6
スイッチ(config-vrf)# exit
スイッチ(config-vrf)# vrf definition v12
スイッチ(config-vrf)# rd 12:1
スイッチ(config-vrf)# address-family ipv6
スイッチ(config-vrf-af)# end
```

スイッチ A の物理インターフェイスを設定します。ギガビット イーサネット インターフェイス 1/0/24 は PE へのトランク接続です。ギガビット イーサネット ポート 1/0/1 と 1/0/2 は VPN に接続されます。

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# interface GigabitEthernet 1/0/1
スイッチ(config-if)# switchport access vlan 208
スイッチ(config-if)# no ip address
```

```
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitEthernet 1/0/2
スイッチ(config-if)# switchport access vlan 118
スイッチ(config-if)# no ip address
スイッチ(config-if)# exit
スイッチ(config)# interface GigabitEthernet 1/0/24
スイッチ(config-if)# switchport trunk encapsulation dot1q
スイッチ(config-if)# switchport mode trunk
スイッチ(config-if)# exit
```

スイッチ A で使用する VLAN を設定します。VLAN 10 は、CE と PE 間の VRF 11 によって使用されます。VLAN 20 は、CE と PE 間の VRF 12 によって使用されます。VLAN 118 と 208 は、それぞれスイッチ E とスイッチ D を含む VPN に使用されます。

```
スイッチ(config)# interface vlan10
スイッチ(config-if)# vrf forwarding v11
スイッチ(config-if)# ipv6 address 1000::1/64
スイッチ(config-if)# exit

スイッチ(config)# interface vlan20
スイッチ(config-if)# vrf forwarding v12
スイッチ(config-if)# ipv6 address 2000::1/64
スイッチ(config-if)# exit

スイッチ(config)# interface vlan208
スイッチ(config-if)# vrf forwarding v11
スイッチ(config-if)# ipv6 address 3000::1/64
スイッチ(config-if)# exit

スイッチ(config)# interface vlan118
スイッチ(config-if)# vrf forwarding v12
スイッチ(config-if)# ipv6 address 4000::1/64
スイッチ(config-if)# exit
```

VPN1 と VPN2 で OSPFv3 ルーティングを設定します。

```
スイッチ(config)# router ospfv3 1
スイッチ(config-router)# router-id 10.1.1.10
スイッチ(config-router)# address-family ipv6 unicast vrf v11
スイッチ(config-router-af)# area 0 normal
スイッチ(config-router-af)# redistribute bgp 800
スイッチ(config-router)# exit
スイッチ(config)# router ospfv3 2
スイッチ(config-router)# router-id 2.2.2.2
スイッチ(config-router)# address-family ipv6 unicast vrf v12
スイッチ(config-router-af)# area 0 normal
スイッチ(config-router-af)# redistribute bgp 800
スイッチ(config-router-af)# exit
スイッチ(config-router)# exit
スイッチ(config)# exit
```

CE/PE ルーティングに BGP を設定します。

```
スイッチ(config)# router bgp 800
スイッチ(config-router)# bgp router-id 8.8.8.8
スイッチ(config-router)# address-family ipv6 vrf v11
スイッチ(config-router-af)# redistribute ospf 1
スイッチ(config-router-af)# neighbor 1000::2 remote-as 100
スイッチ(config-router-af)# neighbor 1000::2 activate
スイッチ(config-router-af)# network 3000::/64
スイッチ(config-router-af)# exit

スイッチ(config)# address-family ipv6 vrf v12
スイッチ(config-router-af)# redistribute ospf 2
スイッチ(config-router-af)# neighbor 2000::2 remote-as 100
スイッチ(config-router-af)# neighbor 2000::2 activate
スイッチ(config-router-af)# network 4000::/64
```

スイッチ D は VPN 1 に属します。次のコマンドを使用して、スイッチ A への接続を設定します。

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# ipv6 unicast-routing
スイッチ(config)# interface GigabitEthernet 5/0/16
スイッチ(config-if)# no switchport
スイッチ(config-if)# ipv6 address 3000::2/64
スイッチ(config-if)# exit

スイッチ(config-router)# router ospfv3 101
スイッチ(config-router)# address-family ipv6
スイッチ(config-router-af)# area 0 normal
スイッチ(config-router-af)# redistribute connected
スイッチ(config-router-af)# exit
スイッチ(config-router)# exit
```

スイッチ E は VPN 2 に属します。次のコマンドを使用して、スイッチ A への接続を設定します。

```
スイッチ(config)# ipv6 unicast-routing
スイッチ(config)# interface GigabitEthernet 3/0/13
スイッチ(config-if)# switchport access vlan 20
スイッチ(config-if)# exit
スイッチ(config)# interface vlan 20
スイッチ(config-if)# ipv6 address 4000::2/64


スイッチ(config)# router ospfv3 101
スイッチ(config-router)# address-family ipv6
スイッチ(config-router-af)# area 0 normal
スイッチ(config-router-af)# redistribute connected
スイッチ(config-router-af)# end
```

このコマンドをスイッチB（PEルータ）で使用すると、CEデバイス、スイッチAに対する接続だけが設定されます。

```
スイッチ(config)# vrf definition v1
スイッチ(config-vrf)# rd 1:1
スイッチ(config-vrf)# address-family ipv6
スイッチ(config-vrf-af)# exit
スイッチ(config-vrf)# exit

スイッチ(config)# vrf definition v2
スイッチ(config-vrf)# rd 2:1
スイッチ(config-vrf)# address-family ipv6
スイッチ(config-vrf-af)# exit
スイッチ(config-vrf)# exit

スイッチ(config-if)# interface g 1/0/2
スイッチ(config-if)# vrf forwarding v1
スイッチ(config-if)# ipv6 address 1000::2/64
スイッチ(config-if)# exit
スイッチ(config)# interface g 1/0/4
スイッチ(config-if)# vrf forwarding v2
スイッチ(config-if)# ipv6 address 2000::2/64

スイッチ(config-if)# interface gigabitEthernet 1/0/1
スイッチ(config-if)# switchport trunk encapsulation dot1q
スイッチ(config-if)# switchport mode trunk

スイッチ(config)# router bgp 100
スイッチ(config-router)# address-family ipv6 vrf v1
スイッチ(config-router-af)# neighbor 1000::1 remote-as 100
スイッチ(config-router-af)# neighbor 1000::1 activate
スイッチ(config-router-af)# network 3000::/64
スイッチ(config-router-af)# exit
スイッチ(config-router)# address-family ipv6 vrf v2
スイッチ(config-router-af)# neighbor 2000::1 remote-as 100
スイッチ(config-router-af)# neighbor 2000::1 activate
スイッチ(config-router-af)# network 4000::/64
```

Multi-VRF CE ステータスの表示

表 *37 : Multi-VRF CE* 情報を表示するコマンド

コマンド	目的
show ipv6 protocols vrf*vrf -name*	VRFに対応付けられたルーティングプロトコル情報を表示します。
show ipv6 route vrf*vrf -name* [**connected**] [*protocol* [*as-number*]] [**list**] [**mobile**] [**odr**] [**profile**] [**static**] [**summary**] [**supernets-only**]	VRFに対応付けられたIPルーティングテーブル情報を表示します。

コマンド	目的
show ipv6 vrf [**brief** | **detail** | **interfaces**] [*vrf-name*]	定義された VRF インスタンスに関する情報を表示します。

IPv6 の表示

次のコマンドの構文および使用方法の詳細については、Cisco IOS のコマンド リファレンスを参照してください。

表 *38 : IPv6* をモニタリングするコマンド

コマンド	目的
show ipv6 access-list	アクセス リストのサマリーを表示します。
show ipv6 cef	IPv6 の Cisco エクスプレス フォワーディングを表示します。
show ipv6 interface *interface-id*	IPv6 インターフェイスのステータスと設定を表示します。
show ipv6 mtu	宛先キャッシュごとに IPv6 MTU を表示します。
show ipv6 neighbors	IPv6 ネイバーキャッシュ エントリを表示します。
show ipv6 prefix-list	IPv6 プレフィックス リストを表示します。
show ipv6 protocols	スイッチの IPv6 ルーティング プロトコルのリストを表示します。
show ipv6 rip	IPv6 RIP ルーティング プロトコル ステータスを表示します。
show ipv6 route	IPv6 ルート テーブル エントリを表示します。
show ipv6 static	IPv6 スタティック ルートを表示します。
show ipv6 traffic	IPv6 トラフィックの統計情報を表示します。

DHCP for IPv6 アドレス割り当ての設定

この項では、DHCPv6 のアドレス割り当てについてだけ説明します。DHCPv6 クライアント、サーバー、またはリレー エージェント機能の設定の詳細については、Cisco.com で『*Cisco IOS IPv6 Configuration Library*』の「Implementing DHCP for IPv6」の章を参照してください。

DHCPv6 アドレス割り当てのデフォルト設定

デフォルトで、DHCPv6 機能はスイッチに設定されています。

DHCPv6 アドレス割り当ての設定時の注意事項

DHCPv6 アドレス割り当てを設定する場合は、次の注意事項に従ってください。

- 以下の手順では、次に示すレイヤ 3 インターフェイスの 1 つを指定する必要があります。
  - DHCPv6 IPv6 ルーティングは、レイヤ 3 インターフェイス上で有効である必要があります。
  - SVI：**interface vlan** *vlan_id* コマンドを使用して作成された VLAN インターフェイスです。
  - レイヤ 3 モードの EtherChannel ポートチャネル：**interface port-channel port-channel-number** コマンドを使用して作成されたポートチャネル論理インターフェイス。
- スイッチは、DHCPv6 クライアント、サーバー、またはリレーエージェントとして動作できます。DHCPv6 クライアント、サーバー、およびリレー機能は、インターフェイスで相互に排他的です。

DHCPv6 サーバー機能の有効化（CLI）

DHCPv6 プールの特性を変更するには、**no** 形式の DHCP プール コンフィギュレーション モード コマンドを使用します。インターフェイスに対して DHCPv6 サーバー機能を無効にするには、**no ipv6 dhcp server** インターフェイス コンフィギュレーション コマンドを使用します。

インターフェイスで DHCPv6 サーバー機能を有効にするには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
ステップ 3	**ipv6 dhcp pool** *poolname* 例： `スイッチ(config)# ipv6 dhcp pool 7`	DHCP プール コンフィギュレーション モードを開始して、IPv6 DHCP プールの名前を定義します。プール名は、記号文字列（Engineering など）または整数（0 など）です。
ステップ 4	**address prefix** *IPv6-prefix* {**lifetime**} {*t1 t1* | **infinite**} 例： `スイッチ(config-dhcpv6)# address prefix 2001:1000::0/64 lifetime 3600`	（任意）アドレス割り当て用のアドレスプレフィックスを指定します。 このアドレスは、16 ビット値をコロンで区切った 16 進数で指定する必要があります。 **lifetime** *t1 t1*：IPv6 アドレス プレフィックスが有効な状態を維持するタイム インターバル（秒）を指定します。指定できる範囲は 5 ～ 4294967295 秒です。時間間隔なしの場合は、**infinite** を指定します。
ステップ 5	**link-address** *IPv6-prefix* 例： `スイッチ(config-dhcpv6)# link-address 2001:1002::0/64`	（任意）link-address IPv6 プレフィックスを指定します。 着信インターフェイス上のアドレスまたはパケットのリンクアドレスが指定した IPv6 プレフィックスに一致する場合、サーバーは設定情報プールを使用します。 このアドレスは、16 ビット値をコロンで区切った 16 進数で指定する必要があります。
ステップ 6	**vendor-specific** *vendor-id* 例： `スイッチ(config-dhcpv6)# vendor-specific 9`	（任意）ベンダー固有のコンフィギュレーション モードを開始して、ベンダー固有の ID 番号を指定します。この番号は、ベンダーの IANA プライベート エンタープライズ番号です。指定できる範囲は 1 ～ 4294967295 です。
ステップ 7	**suboption** *number* { **address** *IPv6-address* | **ascii** *ASCII-string* | **hex** *hex-string*} 例： `スイッチ(config-dhcpv6-vs)# suboption 1 address 1000:235D::`	（任意）ベンダー固有のサブオプション番号を入力します。指定できる範囲は 1 ～ 65535 です。IPv6 アドレス、ASCII テキスト、または 16 進文字列をサブオプション パラメータで定義されているように入力します。
ステップ 8	**exit** 例： `スイッチ(config-dhcpv6-vs)# exit`	DHCP プール コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
ステップ 9	**exit** 例： `スイッチ(config-dhcpv6)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 10	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するインターフェイスを指定します。
ステップ 11	**ipv6 dhcp server** [*poolname* | **automatic**] [**rapid-commit**] [**preference** *value*] [**allow-hint**] 例： `スイッチ(config-if)# ipv6 dhcp server automatic`	インターフェイスに対して DHCPv6 サーバー機能を有効にします。 • *poolname*：（任意）IPv6 DHCP プールのユーザー定義の名前。プール名は、記号文字列（Engineering など）または整数（0 など）です。 • **automatic**：（任意）サーバーが、クライアントにアドレスを割り当てるときに使用するプールを自動的に決定できるようにします。 • **rapid-commit**：（任意）2 つのメッセージを交換する方式を許可します。 • **preference** 値：（任意）サーバーによって送信されるアドバタイズメント メッセージ内のプリファレンス オプションで指定するプリファレンス値を設定します。範囲は 0 ～ 255 です。デフォルトのプリファレンス値は 0 です。 • **allow-hint**：（任意）サーバーが SOLICIT メッセージに含まれるクライアントの提案を考慮するかどうかを指定します。デフォルトでは、サーバーはクライアントのヒントを無視します。
ステップ 12	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 13	次のいずれかを実行します。	• DHCPv6 プール設定を確認します。

	コマンドまたはアクション	目的
	• **show ipv6 dhcp pool** • **show ipv6 dhcp interface** 例： スイッチ# **show ipv6 dhcp pool** または スイッチ# **show ipv6 dhcp interface**	• DHCPv6 サーバー機能がインターフェイス上で有効であることを確認します。
ステップ 14	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

DHCPv6 クライアント機能の有効化

インターフェイスで DHCPv6 クライアントを有効にするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	インターフェイス コンフィギュレーション モードを開始し、設定するインターフェイスを指定します。
ステップ 4	**ipv6 address dhcp** [**rapid-commit**] 例： スイッチ(config-if)# **ipv6 address dhcp rapid-commit**	インターフェイスで DHCPv6 サーバーから IPv6 アドレスを取得できるようにします。 **rapid-commit**：（任意）アドレス割り当てに 2 つのメッセージを交換する方式を許可します。

	コマンドまたはアクション	目的
ステップ 5	**ipv6 dhcp client request** [**vendor-specific**] 例： `スイッチ(config-if)# ipv6 dhcp client request vendor-specific`	（任意）インターフェイスでベンダー固有のオプションを要求できるようにします。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show ipv6 dhcp interface** 例： `スイッチ# show ipv6 dhcp interface`	DHCPv6 クライアントがインターフェイスで有効になっていることを確認します。

IPv6 ユニキャスト ルーティングの設定例

IPv6 アドレッシングの設定と IPv6 ルーティングの有効化：例

次に、IPv6 プレフィックス 2001:0DB8:c18:1::/64 に基づく、リンクローカルアドレスおよびグローバルアドレスを使用して、IPv6 を有効にする例を示します。EUI-64 インターフェイス ID が、両方のアドレスの下位 64 ビットで使用されます。**show ipv6 interface** EXEC コマンドの出力は、インターフェイスのリンクローカル プレフィックス FE80::/64 にインターフェイス ID（20B:46FF:FE2F:D940）を付加する方法を示すために追加されています。

```
スイッチ(config)# ipv6 unicast-routing
スイッチ(config)# interface gigabitethernet0/11

スイッチ(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
スイッチ(config-if)# end
スイッチ# show ipv6 interface gigabitethernet0/11
GigabitEthernet0/11 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
  2001:0DB8:c18:1:20B:46FF:FE2F:D940, subnet is 2001:0DB8:c18:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
```

```
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
Hosts use stateless autoconfig for addresses.
```

デフォルト ルータ プリファレンスの設定：例

次に、インターフェイス上のルータに高い DRP を設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# ipv6 nd router-preference high
スイッチ(config-if)# end
```

IPv6 の HSRP グループのイネーブル化：例

次に、ポートのグループ 1 で IPv6 の HSRP をアクティブにする例を示します。ホットスタンバイ グループで使用される IP アドレスは、IPv6 の HSRP を使用して学習されます。

（注） これは、IPv6 の HSRP をイネーブルにするために必要な最小限の手順です。その他の設定は任意です。

```
スイッチ# configure terminal
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# no switchport
スイッチ(config-if)# standby 1 ipv6 autoconfig
スイッチ(config-if)# end
スイッチ# show standby
```

DHCPv6 サーバー機能の有効化：例

次の例では、*engineering* という IPv6 アドレス プレフィックスを持つプールを設定する方法を示します。

```
スイッチ# configure terminal
スイッチ(config)# ipv6 dhcp pool engineering
スイッチ(config-dhcpv6)#address prefix 2001:1000::0/64
スイッチ(config-dhcpv6)# end
```

次に、3 リンクアドレスおよび IPv6 アドレス プレフィックスを持つ *testgroup* と呼ばれるプールを設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# ipv6 dhcp pool testgroup
スイッチ(config-dhcpv6)# link-address 2001:1001::0/64
スイッチ(config-dhcpv6)# link-address 2001:1002::0/64
スイッチ(config-dhcpv6)# link-address 2001:2000::0/48
スイッチ(config-dhcpv6)# address prefix 2001:1003::0/64
スイッチ(config-dhcpv6)# end
```

次の例では、*350* というベンダー固有オプションを持つプールを設定する方法を示します。

```
スイッチ# configure terminal
スイッチ(config)# ipv6 dhcp pool 350
スイッチ(config-dhcpv6)# address prefix 2001:1005::0/48
スイッチ(config-dhcpv6)# vendor-specific 9
スイッチ(config-dhcpv6-vs)# suboption 1 address 1000:235D::1
スイッチ(config-dhcpv6-vs)# suboption 2 ascii "IP-Phone"
スイッチ(config-dhcpv6-vs)# end
```

DHCPv6 クライアント機能の有効化：例

次に、IPv6 アドレスを取得して、rapid-commit オプションを有効にする例を示します。

```
スイッチ(config)# interface gigabitethernet2/0/1
スイッチ(config-if)# ipv6 address dhcp rapid-commit
```

IPv6 ICMP レート制限の設定：例

次に、IPv6 ICMP エラー メッセージ間隔を 50 ミリ秒に、バケット サイズを 20 トークンに設定する例を示します。

```
スイッチ(config)#ipv6 icmp error-interval 50 20
```

IPv6 のスタティック ルーティングの設定：例

次に、アドミニストレーティブ ディスタンスが 130 のフローティング スタティック ルートをインターフェイスに設定する例を示します。

```
スイッチ(config)# ipv6 route 2001:0DB8::/32 gigabitethernet 1/0/1 130
```

IPv6 の RIP の設定：例

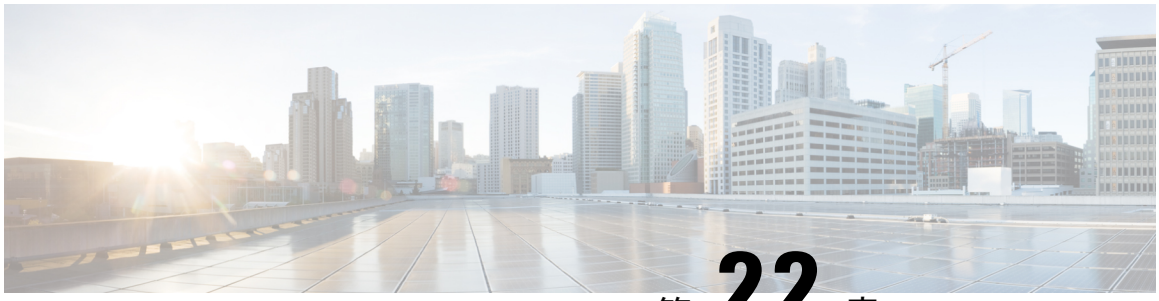
次に、最大 8 の等コスト ルートにより RIP ルーティング プロセス *cisco* を有効にし、インターフェイス上でこれを有効にする例を示します。

```
スイッチ(config)# ipv6 router rip cisco
スイッチ(config-router)# maximum-paths 8
スイッチ(config)# exit
スイッチ(config)# interface gigabitethernet2/0/11
スイッチ(config-if)# ipv6 rip cisco enable
```

IPv6 の表示：例

次に、**show ipv6 interface** 特権 EXEC コマンドの出力例を示します。

```
スイッチ# show ipv6 interface
Vlan1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
<output truncated>
```



第 22 章

IPv6 マルチキャストの実装

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

IPv6 マルチキャスト ルーティングの実装に関する情報

この章では、スイッチに IPv6 マルチキャスト ルーティングを実装する方法について説明します。

従来の IP 通信では、ホストはパケットを単一のホスト（ユニキャスト伝送）またはすべてのホスト（ブロードキャスト伝送）に送信できます。IPv6 マルチキャストは、第三の方式を提供するものであり、ホストが単一のデータ ストリームをすべてのホストのサブセット（グループ伝送）に同時に送信できるようにします。

（注） IPv6 マルチキャストルーティングは Cisco Catalyst 3560-CX スイッチでのみサポートされます。

IPv6 マルチキャストの概要

IPv6 マルチキャスト グループは、特定のデータ ストリームを受信する受信側の任意のグループです。このグループには、物理的境界または地理的境界はありません。受信側は、インターネット上または任意のプライベート ネットワーク内の任意の場所に配置できます。特定のグループへのデータ フローの受信に関与する受信側は、ローカル スイッチに対してシグナリングすることによってそのグループに加入する必要があります。このシグナリングは、MLD プロトコルを使用して行われます。

スイッチは、MLD プロトコルを使用して、直接接続されているサブネットにグループのメンバが存在するかどうかを学習します。ホストは、MLD レポート メッセージを送信することによってマルチキャスト グループに加入します。ネットワークでは、各サブネットでマルチキャストデータのコピーを 1 つだけ使用して、潜在的に無制限の受信側にデータが伝送されます。トラフィックの受信を希望する IPv6 ホストはグループ メンバと呼ばれます。

グループ メンバに伝送されるパケットは、単一のマルチキャスト グループ アドレスによって識別されます。マルチキャスト パケットは、IPv6 ユニキャスト パケットと同様に、ベストエフォート型の信頼性を使用してグループに伝送されます。

マルチキャスト環境は、送信側と受信側で構成されます。どのホストも、グループのメンバであるかどうかにかかわらず、グループに送信できます。ただし、グループのメンバだけがメッセージをリッスンして受信できます。

マルチキャストアドレスがマルチキャストグループの受信先として選択されます。送信者は、データグラムの宛先アドレスとしてグループのすべてのメンバに到達するためにそのアドレスを使用します。

マルチキャストグループ内のメンバーシップはダイナミックです。ホストはいつでも加入および脱退できます。マルチキャストグループ内のメンバの場所または数に制約はありません。ホストは、一度に複数のマルチキャスト グループのメンバにすることができます。

マルチキャストグループがどの程度アクティブであるか、その期間、およびメンバーシップはグループおよび状況によって異なります。メンバを含むグループにアクティビティがない場合もあります。

IPv6 マルチキャスト ルーティングの実装

Cisco IOS ソフトウェアでは、IPv6 マルチキャスト ルーティングを実装するため、次のプロトコルがサポートされています。

- MLD は、直接接続されているリンク上のマルチキャスト リスナー（特定のマルチキャスト アドレスを宛先としたマルチキャスト パケットを受信するために使用するノード）を検出するために IPv6 スイッチで使用されます。MLD には 2 つのバージョンがあります。MLD バージョン 1 はバージョン 2 のインターネット グループ管理プロトコル（IGMP）for IPv4 をベースとしています。MLD バージョン 2 はバージョン 3 の IGMP for IPv4 をベースとしています。Cisco IOS ソフトウェアの IPv6 マルチキャストでは、MLD バージョン 2 と MLD バージョン 1 の両方が使用されます。MLD バージョン 2 は、MLD バージョン 1 と完全な下位互換性があります（RFC 2710 で規定）。MLD バージョン 1 だけをサポートするホストは、MLD バージョン 2 を実行しているスイッチと相互運用します。MLD バー

ジョン 1 ホストと MLD バージョン 2 ホストの両方が混在する LAN もサポートされています。

- PIM-SM は、相互に転送されるマルチキャスト パケット、および直接接続されている LAN に転送されるマルチキャスト パケットを追跡するためにスイッチ間で使用されます。
- PIM in Source Specific Multicast（PIM-SSM）は PIM-SM と類似していますが、IP マルチキャスト アドレスを宛先とした特定の送信元アドレス（または特定の送信元アドレスを除くすべてのアドレス）からのパケットを受信する対象をレポートする機能を別途備えています。

MLD アクセス グループ

MLD アクセス グループは、Cisco IOS IPv6 マルチキャスト スイッチでの受信側アクセス コントロールを実現します。この機能では、受信側が加入できるグループのリストを制限し、SSM チャネルへの加入に使用される送信元を許可または拒否します。

受信側の明示的トラッキング

明示的トラッキング機能を使用すると、スイッチが IPv6 ネットワーク内のホストの動作を追跡できるようになります。また、この機能により、高速脱退メカニズムを MLD バージョン 2 のホスト レポートで使用できるようになります。

IPv6 マルチキャスト ユーザ認証およびプロファイル サポート

IPv6 マルチキャストは、ネットワーク内の任意のホストがマルチキャスト グループの受信側または送信元になれる設計になっています。したがって、ネットワークのマルチキャスト トラフィックを制御するには、マルチキャスト アクセス コントロールが必要です。アクセス コントロール機能は、主に、送信元のアクセス コントロールとアカウンティング、受信側のアクセス コントロールとアカウンティング、およびこのアクセス コントロール メカニズムのプロビジョニングで構成されます。

マルチキャスト アクセス コントロールは、マルチキャストと認証、許可、アカウンティング（AAA）間のインターフェイスを提供し、ラストホップ スイッチ、マルチキャストにおける受信側アクセス コントロール機能、およびマルチキャストにおけるグループまたはチャネル ディセーブル化機能でのプロビジョニング、許可、およびアカウンティングを実現します。

新しいマルチキャスト サービス環境を展開する場合、ユーザ認証を追加し、インターフェイス単位でユーザ プロファイルのダウンロードを行う必要があります。AAA と IPv6 マルチキャストを使用すると、マルチキャスト環境でのユーザ認証とユーザ プロファイルのダウンロードがサポートされます。

RADIUS サーバからアクセス スイッチへのマルチキャスト アクセス コントロール プロファイルのダウンロードをトリガーするイベントは、アクセス スイッチへの MLD join の着信です。このイベントが発生すると、ユーザは認可キャッシュのタイムアウトを発生させて定期的なダウンロードを要求するか、または適切な multicast clear コマンドを使用してプロファイルが変更された場合に新規ダウンロードをトリガーできます。

アカウンティングはRADIUSアカウンティングを使用して行われます。開始および停止アカウンティング レコードは、アクセス スイッチから RADIUS サーバに送信されます。リソースの消費をストリーム単位で追跡できるように、これらのアカウンティング レコードには、マルチキャスト送信元およびグループに関する情報が含まれています。ラストホップ スイッチが新しい MLD レポートを受信すると、開始レコードが送信され、MLD leave を受信するか、何らかの理由によりグループまたはチャネルが削除されると、停止レコードが送信されます。

IPv6 MLD プロキシ

MLD プロキシ機能は、スイッチのアップストリーム インターフェイス上で、スイッチがすべての（*, G）および（S, G）エントリに対して MLD メンバーシップ レポートを生成するか、またはこれらのエントリのユーザ定義サブセットを生成するメカニズムを提供します。MLD プロキシ機能により、デバイスは、プロキシ グループ メンバーシップ情報を学習し、その情報に基づいてマルチキャスト パケットを転送できるようになります。

スイッチが mroute プロキシ エントリの RP として動作する場合、これらのエントリの MLD メンバーシップ レポートを、ユーザが指定したプロキシ インターフェイス上で生成できます。

プロトコル独立マルチキャスト

PIM（Protocol Independent Multicast）は、相互に転送されるマルチキャスト パケット、および直接接続されている LAN に転送されるマルチキャスト パケットを追跡するためにスイッチ間で使用されます。PIM は、ユニキャスト ルーティング プロトコルとは独立して動作し、他のプロトコルと同様に、マルチキャスト ルート アップデートの送受信を実行します。ユニキャスト ルーティング テーブルに値を入力するために LAN でどのユニキャスト ルーティング プロトコルが使用されているかどうかにかかわらず、Cisco IOS PIM では、独自のルーティング テーブルを構築および管理する代わりに、既存のユニキャスト テーブル コンテンツを使用して、Reverse Path Forwarding（RPF）チェックを実行します。

PIM-SM または PIM-SSM のいずれかを使用するように IPv6 マルチキャストを設定することも、ネットワークで PIM-SM と PIM-SSM の両方を使用することもできます。

PIM スパース モード

IPv6 マルチキャストでは、PIM-SM を使用したドメイン内マルチキャスト ルーティングがサポートされています。PIM-SM は、ユニキャスト ルーティングを使用して、マルチキャスト ツリー構築用のリバースパス情報を提供しますが、特定のユニキャスト ルーティング プロトコルには依存しません。

PIM-SM は、トラフィックに対して明示的な要求がある場合を除いて、各マルチキャストに関与しているスイッチの数が比較的少なく、これらのスイッチがグループのマルチキャスト パケットを転送しないときに、マルチキャスト ネットワークで使用されます。PIM-SMは、共有ツリー上のデータ パケットを転送することによって、アクティブな送信元に関する情報を配布します。PIM-SMは最初に共有ツリーを使用しますが、これにはRPの使用が必要となります。

要求は、ツリーのルート ノードに向けてホップバイホップで送信される PIM join を使用して行われます。PIM-SM のツリーのルート ノードは、共有ツリーの場合は RP、最短パス ツリー（SPT）の場合はマルチキャスト送信元に直接接続されているファーストホップ スイッチにな

ります。RP はマルチキャスト グループを追跡し、マルチキャスト パケットを送信するホストはそのホストのファーストホップ スイッチによって RP に登録されます。

PIM join がツリーの上位方向に送信されると、要求されたマルチキャスト トラフィックがツリーの下位方向に転送されるように、パス上のスイッチがマルチキャスト転送ステートを設定します。マルチキャスト トラフィックが不要になったら、スイッチはルート ノードに向けてツリーの上位方向に PIM prune を送信し、不必要なトラフィックをプルーニング（削除）送信します。この PIM prune がホップごとにツリーを上位方向に移動する際、各スイッチはその転送状態を適切に更新します。最終的に、マルチキャスト グループまたは送信元に関連付けられている転送ステートは削除されます。

マルチキャスト データの送信側は、マルチキャスト グループを宛先としたデータを送信します。送信側の指定スイッチ（DR）は、これらのデータ パケットを受け取り、ユニキャストでカプセル化し、RP に直接送信します。RP は、カプセル化されたこれらのデータ パケットを受信し、カプセル化を解除し、共有ツリー上に転送します。そのあと、パケットは、RP ツリー上のスイッチの（*, G）マルチキャスト ツリー ステートに従って、RP ツリー ブランチの任意の場所に複製され、そのマルチキャスト グループのすべての受信側に最終的に到達します。RP へのデータ パケットのカプセル化のプロセスは登録と呼ばれ、カプセル化されたパケットは PIM レジスタ パケットと呼ばれます。

指定スイッチ

Cisco スイッチは、LAN セグメント上に複数のスイッチが存在する場合、PIM-SM を使用してマルチキャスト トラフィックを転送し、選択プロセスに従って指定スイッチを選択します。

指定スイッチは、PIM register メッセージ、PIM join メッセージ、および PIM prune メッセージを RP に送信し、アクティブな送信元およびホスト グループ メンバーシップに関する情報を通知します。

LAN 上に複数の PIM-SM スイッチが存在する場合は、指定スイッチを選択して、接続されているホストに対するマルチキャスト トラフィックの重複を回避する必要があります。ipv6 pim dr-priority コマンドを使用して DR の選択を強制することを選択しない限り、最も大きい IPv6 アドレスの PIM スイッチが LAN の DR になります。このコマンドでは、LAN セグメント上の各スイッチの DR プライオリティ（デフォルトのプライオリティ = 1）を指定して、最もプライオリティの高いスイッチが DR として選択されるようにすることができます。LAN セグメント上のすべてのスイッチのプライオリティが同じ場合にも、最上位 IPv6 アドレスを持つスイッチが使用されます。

DR で障害が発生した場合、PIM-SM はスイッチ A の障害を検出し、フェールオーバー DR を選択する手段を提供します。DR（スイッチ A）が動作不能になった場合、スイッチ A とネイバーとの隣接関係がタイムアウトすると、スイッチ B はその状況を検出します。スイッチ B はホスト A から MLD メンバーシップ レポートを受けているため、このインターフェイスでグループ A の MLD ステートをすでに持ち、新しい DR になると即座に RP に join を送信します。この段階で、スイッチ B を経由する共有ツリーの新しいブランチの下位方向へのトラフィック フローが再び確立されます。また、ホスト A がトラフィックをソーシングしていた場合、スイッチ B は、ホスト A から次のマルチキャスト パケットを受信した直後に、新しい登録プロセスを開始します。このアクションで、RP による、スイッチ B を経由する新しいブランチを介したホスト A への SPT 加入がトリガーされます。

（注）
- 2 つの PIM スイッチが直接接続されている場合、これらのスイッチはネイバーになります。PIM ネイバーを表示するには、show ipv6 pim neighbor 特権 EXEC コマンドを使用します。
- DR 選択プロセスは、マルチアクセス LAN のみで必要です。

ランデブー ポイント

IPv6 PIM では、組み込み RP がサポートされています。組み込み RP サポートを利用すると、スイッチは、スタティックに設定されている RP の代わりに、マルチキャスト グループ宛先アドレスを使用して RP 情報を学習できるようになります。スイッチが RP である場合、RP としてスタティックに設定する必要があります。

スイッチは、MLD レポート内、または PIM メッセージおよびデータ パケット内の組み込み RP グループ アドレスを検索します。このようなアドレスが見つかったら、スイッチはアドレス自体からグループの RP を学習します。この学習された RP は、グループのすべてのプロトコル アクティビティに使用されます。スイッチが RP である場合、組み込み RP を RP として設定する必要があり、スイッチはそのようにアドバタイズされます。

組み込み RP よりも優先するスタティック RP を選択するには、特定の組み込み RP グループ範囲またはマスクをスタティック RP のアクセス リストに設定する必要があります。PIM がスパース モードで設定されている場合は、RP として動作する 1 つ以上のスイッチを選択する必要もあります。RP は、共有配布ツリーの選択ポイントに配置された単一の共通ルートであり、各ボックスでスタティックに設定されます。

PIM DR は、共有ツリーの下位方向に配布するために、直接接続されているマルチキャスト送信元から RP にデータを転送します。データは次の 2 つの方法のいずれかを使用して RP に転送されます。

- データは、登録パケットにカプセル化され、DR として動作するファーストホップ スイッチによって直接 RP にユニキャストされます。
- RP 自身が送信元ツリーに加入している場合は、PIM スパース モードの項で説明したように、RPF 転送アルゴリズムに従ってマルチキャスト転送されます。

RP アドレスは、パケットをグループに送信するホストの代わりに、ファーストホップ スイッチによって PIM register メッセージを送信するために使用されます。また、RP アドレスは、ラストホップ スイッチによって PIM join および prune メッセージを RP に送信してグループ メンバーシップについて通知するためにも使用されます。すべてのスイッチ（RP スイッチを含む）で RP アドレスを設定する必要があります。

1 つの PIM スイッチを複数のグループの RP にすることができます。特定のグループの PIM ドメイン内で一度に使用できる RP アドレスは 1 つだけです。アクセス リストで指定されている条件によって、スイッチがどのグループの RP であるかが判別されます。

IPv6 マルチキャストでは、PIM accept register 機能がサポートされています。これは、RP で PIM-SM register メッセージのフィルタリングを実行するための機能です。ユーザーは、アクセ

ス リストを照合するか、または登録されている送信元の AS パスとルート マップに指定されている AS パスを比較できます。

PIMv6 エニーキャスト RP ソリューションの概要

IPv6 PIM のエニーキャスト RP ソリューションは、IPv6 ネットワークによる PIM-SM RP のエニーキャスト サービスのサポートを可能にします。これにより、PIM のみを実行するドメイン内でエニーキャスト RP を使用できるようになります。この機能は、ドメイン間接続が不要な場合に便利です。エニーキャスト RP は、IPv4 および IPv6 で使用できますが、IPv4 だけで動作する Multicast Source Discovery Protocol（MSDP）には依存しません。

エニーキャスト RP は、PIM RP のデバイスに障害が発生した場合に、高速コンバージェンスを取得するために ISP ベースのバックボーンが使用するメカニズムです。受信側および送信元が最も近くの RP にランデブーできるようにするには、送信元からのパケットがすべての RP に到達して、加入している受信側を検出する必要があります。

ユニキャスト IP アドレスは RP アドレスとして選択されます。このアドレスは、静的に設定されるか、またはダイナミック プロトコルを使用して、ドメイン全体のすべての PIM デバイスに配信されます。ドメイン内の一連のデバイスが、この RP アドレスの RP として動作するように選択されます。これらのデバイスは、エニーキャスト RP セットと呼ばれます。エニーキャスト RP セット内の各デバイスは、RP アドレスを使用してループバック インターフェイスで設定されます。また、エニーキャスト RP セット内の各デバイスには、RP 間の通信に使用する別の物理 IP アドレスも必要です。

RP アドレス、または RP アドレスに対応するプレフィックスは、ドメイン内部のユニキャスト ルーティング システムに挿入されます。エニーキャスト RP セット内の各デバイスは、エニーキャスト RP セット内のその他すべてのデバイスのアドレスで設定されます。また、この設定は、セット内のすべての RP で一致している必要があります。

IPv6 BSR：RP マッピングの設定

ドメイン内の PIM スイッチは、各マルチキャスト グループを正しい RP アドレスにマッピングできる必要があります。PIM-SM 対応の BSR プロトコルは、グループと RP のマッピング情報をドメイン全体に迅速に配布するためのダイナミック適応メカニズムを備えています。IPv6 BSR 機能を使用すると、到達不能になった RP が検出され、マッピング テーブルが変更されます。これにより、到達不能な RP が今後使用されなくなり、新しいテーブルがドメイン全体に迅速に配布されるようになります。

すべての PIM-SM マルチキャスト グループを RP の IP または IPv6 アドレスに関連付ける必要があります。新しいマルチキャスト送信側が送信を開始すると、そのローカル DR がこれらのデータ パケットを PIM register メッセージにカプセル化し、そのマルチキャスト グループの RP に送信します。新しいマルチキャスト受信側が加入すると、そのローカル DR がそのマルチキャスト グループの RP に PIM join メッセージを送信します。PIM スイッチは、（*, G）join メッセージを送信するとき、RP 方向への次のスイッチを認識して、G（グループ）がそのスイッチにメッセージを送信できるようにする必要があります。また、PIM スイッチは、（*, G）ステートを使用してデータ パケットを転送するとき、G を宛先としたパケットの正しい着信インターフェイスを認識する必要があります。これは、他のインターフェイスに着信するパケットを拒否する必要があるためです。

ドメイン内の少数のスイッチが候補ブートストラップ スイッチ（C-BSR）として設定され、単一の BSR がそのドメイン用に選択されます。また、ドメイン内の一連のスイッチが候補 RP（C-RP）として設定されます。通常、これらのスイッチは、C-BSR として設定されているものと同じスイッチです。候補 RP は、候補 RP アドバタイズメント（C-RP-Adv）メッセージをそのドメインの BSR に定期的にユニキャストし、RP になる意思をアドバタイズします。C-RP-Adv メッセージには、アドバタイズを行っている C-RP のアドレス、およびグループ アドレスとマスク長のフィールドの任意のリストが含まれています。これらのフィールドは、立候補のアドバタイズの対象となるグループ プレフィックスを示します。BSR は、定期的に発信するブートストラップ メッセージ（BSM）にこれらの一連の C-RP とそれに対応するグループ プレフィックスを含めます。BSM は、ドメイン全体にホップバイホップで配布されます。

双方向 BSR がサポートされているため、双方向 RP を C-RP メッセージおよび BSM の双方向範囲でアドバタイズできます。システム内のすべてのスイッチは、BSM で双方向範囲を使用できる必要があります。使用できない場合は、双方向 RP 機能が機能しません。

PIM-Source Specific Multicast (PIM-SSM)

PIM-SSM は、SSM の実装をサポートするルーティング プロトコルであり、PIM-SM から派生したものです。ただし、PIM-SM では PIM join を受けてすべてのマルチキャスト送信元からデータが送信されるのに対し、SSM機能では、受信側が明示的に加入しているマルチキャスト送信元だけからその受信側にデータグラム トラフィックが転送されます。これにより、帯域利用率が最適化され、不要なインターネット ブロードキャスト トラフィックが拒否されます。さらに、SSM では、RP と共有ツリーを使用する代わりに、マルチキャスト グループの送信元アドレスで見つかった情報を使用します。この情報は、MLD メンバーシップ レポートによってラストホップ スイッチにリレーされる送信元アドレスを通して受信側から提供されます。その結果として、送信元に直接つながる最短パス ツリーが得られます。

SSM では、データグラムは（S, G）チャネルに基づいて配信されます。1 つの（S, G）チャネルのトラフィックは、IPv6 ユニキャスト送信元アドレス S とマルチキャスト グループ アドレス G を IPv6 宛先アドレスとして使用するデータグラムで構成されます。システムは、（S, G）チャネルのメンバになることによって、このトラフィックを受信します。シグナリングは不要ですが、受信側は特定の送信元からのトラフィックを受信する場合は（S, G）チャネルに加入し、トラフィックを受信しない場合はチャネルから脱退する必要があります。

SSM を動作させるには、MLD バージョン 2 が必要です。MLD を使用すると、ホストが送信元の情報を提供できるようになります。MLD を使用して SSM を動作させるには、Cisco IOS IPv6 スイッチ、アプリケーションが実行されているホスト、およびアプリケーション自体で SSM がサポートされている必要があります。

IPv6 用の SSM マッピング

IPv6 用の SSM マッピングでは、MLD バージョン 1 の受信側用にスタティックとダイナミックの両方のドメイン ネーム システム（DNS）マッピングがサポートされています。この機能を使用すると、TCP/IP ホスト スタックおよび IP マルチキャスト受信アプリケーションで MLD バージョン 2 サポートを提供できないホストで IPv6 SSM を展開できます。

SSM マッピングにより、スイッチは実行コンフィギュレーションまたは DNS サーバのいずれかでマルチキャスト MLD バージョン 1 レポートの送信元を検索できるようになります。そのあと、スイッチは送信元に対する（S, G）join を開始できます。

PIM 共有ツリーおよびソース ツリー（最短パス ツリー）

デフォルトでは、グループのメンバは、RPをルートとする単一のデータ配布ツリーを通じて、送信側からグループへのデータを受信します。このタイプの配布ツリーは、共有ツリーまたはランデブー ポイント ツリー（RPT）と呼ばれます（下の図を参照）。送信側からのデータは、RP に配信され、その共有ツリーに加入しているグループ メンバに配布されます。

データしきい値で保証される場合、共有ツリー上のリーフ スイッチは、送信元をルートとするデータ配布ツリーへの切り替えを開始できます。このタイプの配布ツリーは、最短パス ツリーまたはソース ツリーと呼ばれます。デフォルトでは、Cisco IOS ソフトウェアは、送信元から最初のデータ パケットを受信した時点で、ソース ツリーへの切り替えを行います。

次に、共有ツリーからソース ツリーに切り替わるプロセスの詳細を示します。

1. 受信側がグループに加入します。リーフ スイッチ C が RP に join メッセージを送信します。
2. RP がスイッチ C へのリンクを発信インターフェイス リストに登録します。
3. 送信元がデータを送信します。スイッチ A が register にデータをカプセル化し、それを RP に送信します。
4. RP が共有ツリーの下位方向のスイッチ C にデータを転送し、送信元に join メッセージを送信します。この時点で、データはスイッチ C に 2 回（カプセル化された状態で 1 回、ネイティブの状態で 1 回）着信する可能性があります。
5. データがネイティブの（カプセル化されていない）状態で RP に着信すると、RP はスイッチ A に register-stop メッセージを送信します。
6. デフォルトでは、最初のデータ パケット受信時に、スイッチ C が Join メッセージを送信元に送信するよう要求します。
7. スイッチ C は、（S, G）でデータを受信すると、共有ツリーの上位方向にある送信元に prune メッセージを送信します。
8. RP が（S, G）の発信インターフェイスからスイッチ C へのリンクを削除します。
9. RP が送信元への prune メッセージをトリガーします。

送信元および RP に join および prune メッセージが送信されます。これらのメッセージはホップバイホップで送信され、送信元または RP へのパス上にある各 PIM スイッチで処理されます。register および register-stop メッセージは、ホップバイホップで送信されません。これらのメッセージは、送信元に直接接続されている指定スイッチによって送信され、グループの RP によって受信されます。

Reverse Path Forwarding

Reverse Path Forwarding は、マルチキャスト データグラムの転送に使用されます。これは、次のように機能します。

- スイッチで、送信元へのユニキャストパケットの送信に使用しているインターフェイスでデータグラムを受信すると、パケットは RPF インターフェイスに着信しています。
- パケットが RPF インターフェイスに着信した場合、スイッチは、マルチキャスト ルーティング テーブル エントリの発信インターフェイス リストに存在するインターフェイスにパケットを転送します。
- パケットが RPF インターフェイスに着信しない場合、パケットはループを回避するためにサイレントにドロップされています。

PIM では、送信元ツリーと RP をルートとする共有ツリーの両方を使用してデータグラムを転送します。RPF チェックは、次のようにそれぞれ異なる方法で実行されます。

- PIM スイッチが送信元ツリー ステートである場合（つまり、（S, G）エントリがマルチキャスト ルーティング テーブル内にある場合）、マルチキャスト パケットの送信元の IPv6 アドレスに対して RPF チェックが実行されます。
- PIM スイッチが共有ツリー ステートである場合（および送信元ツリー ステートが明示されていない場合）、（メンバがグループに加入している場合は既知である）RP のアドレスに対して RPF チェックが実行されます。

空間モード PIM では、RPF ルックアップ機能を使用して、join および prune の送信先を決定します。（S, G）join（送信元ツリー ステート）は送信元に向けて送信されます。（*, G）join（共有ツリー ステート）は RP に向けて送信されます。

ルーティング可能アドレスの hello オプション

IPv6 内部ゲートウェイ プロトコルを使用してユニキャスト ルーティング テーブルを構築する場合、アップストリーム スイッチ アドレスを検出するための手順では、PIM ネイバーとネクストホップ スイッチが同じスイッチを表しているかぎり、これらのアドレスは常に同じであるものと想定されます。ただし、スイッチがリンク上に複数のアドレスを持つ場合は、このことが当てはまるとはかぎりません。

この状況は IPv6 において、2 つの一般的な状況で発生することがあります。1 つめの状況は、ユニキャスト ルーティング テーブルが IPv6 内部ゲートウェイ プロトコル（マルチキャスト BGP など）によって構築されない場合に発生します。2 つめの状況は、RP のアドレスがダウンストリーム スイッチとサブネット プレフィックスを共有している場合に発生します（RP スイッチ アドレスはドメインワイドにする必要があるため、リンクローカル アドレスにはできないことに注意してください）。

ルーティング可能アドレスの hello オプションによって、PIM プロトコルでこのような状況を回避できます。このためには、PIM hello メッセージがアドバタイズされるインターフェイス上のすべてのアドレスを含む PIM hello メッセージ オプションを追加します。PIM スイッチが何らかのアドレスのアップストリーム スイッチを検出すると、RPF 計算の結果は、PIM ネイバーのアドレス自体に加えて、このオプションのアドレスとも比較されます。このオプション

にはそのリンク上の PIM スイッチの考えられるアドレスがすべて含まれているため、対象の PIM スイッチがこのオプションをサポートしている場合、常に RPF 計算の結果が含まれます。

PIM メッセージにサイズ制限があることと、ルーティング可能アドレスの hello オプションが単一の PIM hello メッセージ内に収まる必要があるため、インターフェイスで設定できるアドレスの制限は 16 個になっています。

双方向 PIM

双方向 PIM により、マルチキャスト スイッチは、PIM-SM の単方向共有ツリーと比較して、保持するステート情報を減らすことができます。双方向共有ツリーは、データを送信元からランデブーポイントアドレス（RPA）に伝送し、それらを RPA から受信側に配布します。PIM-SM とは異なり、双方向 PIM は送信元ツリーへの切り替えは実行しません。また、送信元から RP へのデータの登録カプセル化は行われません。

指定された単一のフォワーダ（DF）が、双方向 PIM ドメイン内のすべてのリンク（マルチアクセスおよびポイントツーポイント リンクを含む）の各 RPA 用に存在しています。唯一の例外は、DF が存在しない RPL です。DF は、MRIB が提供するメトリックとの比較で決定される、RPA への最適なルートを持つリンク上のスイッチです。指定された RPA の DF は、リンクにダウンストリーム トラフィックを転送し、リンクからのアップストリーム トラフィックをランデブー ポイント リンク（RPL）に転送します。DF は、RPA にマップするすべての双方向グループに対してこの機能を実行します。また、リンク上の DF は、リンク上のダウンストリーム スイッチからの Join メッセージを処理するとともに、MLD などのローカル メンバーシップ メカニズムによって検出されたローカル受信者にパケットが転送されることを保証します。

双方向 PIM は、中レートまたは低レートの送信元が多数存在する場合に役立ちます。ただし、双方向共有ツリーの遅延特性は、PIM-SM で構築された送信元ツリーよりもさらに劣る可能性があります（トポロジに依存）。

IPv6 では、双方向 RP のスタティック設定だけがサポートされています。

スタティック mroute

IPv6 スタティック mroute は、RPF チェックを変化させるために使用する IPv4 スタティック mroute とほぼ同様に動作します。IPv6 スタティック mroute は、IPv6 スタティック ルートと同じデータベースを共有し、RPF チェックに対するスタティック ルート サポートを拡張することによって実装されます。スタティック mroute では、等コスト マルチパス mroute がサポートされています。また、ユニキャスト専用スタティック ルートもサポートされています。

MRIB

マルチキャストルーティング情報ベース（MRIB）は、マルチキャストルーティング プロトコル（ルーティング クライアント）によってインスタンス化されるマルチキャスト ルーティング エントリのプロトコル非依存リポジトリです。その主要機能は、ルーティング プロトコルとマルチキャスト転送情報ベース（MFIB）間の非依存性を実現することです。また、クライアント間の調整および通信ポイントとしても機能します。

ルーティング クライアントは、MRIB が提供するサービスを使用して、ルーティング エントリをインスタンス化し、他のクライアントによってルーティングエントリに加えられた変更を取得します。MRIB では、ルーティング クライアント以外に、転送クライアント（MFIB インスタンス）や特別なクライアント（MLD など）も扱われます。MFIB は、MRIB からその転送エントリを取得し、パケットの受信に関連するイベントについてMRIBに通知します。これらの通知は、ルーティング クライアントによって明示的に要求されることも、MFIB によって自発的に生成されることもあります。

MRIB のもう 1 つの重要な機能は、同じマルチキャスト セッション内でマルチキャスト接続を確立する際に、複数のルーティング クライアントの調整を可能にすることです。また、MRIB では、MLD とルーティング プロトコル間の調整も可能です。

MFIB

MFIB は、IPv6 ソフトウェア用のプラットフォーム非依存およびルーティング プロトコル非依存ライブラリです。その主な目的は、転送テーブルが変更されたときに、Cisco IOS プラットフォームに、IPv6マルチキャスト転送テーブルおよび通知を読み取るインターフェイスを提供することです。MFIB が提供する情報には、明確に定義された転送セマンティクスが含まれています。この情報は、プラットフォームが特定のハードウェアまたはソフトウェア転送メカニズムに容易に変換できる設計になっています。

ネットワーク内でルーティングまたはトポロジが変更されると、IPv6 ルーティング テーブルがアップデートされ、これらの変更が MFIB に反映されます。MFIB は、IPv6 ルーティング テーブル内の情報に基づいて、ネクストホップ アドレス情報を管理します。MFIB エントリとルーティング テーブル エントリの間には 1 対 1 の相互関係があるため、MFIB には既知のすべてのルートが含まれ、高速スイッチングや最適スイッチングなどのスイッチング パスに関連付けられているルート キャッシュ管理の必要がなくなります。

IPv6 マルチキャスト VRF Lite

IPv6 マルチキャスト VRF Lite 機能は、複数の仮想ルーティングおよび転送（VRF）コンテキストに対する IPv6 マルチキャスト サポートを提供します。これらの VRF のスコープは、VRF が定義されているスイッチに制限されています。

この機能により、別の VRF に属するデバイス間の通信は、明示的に設定されていない限り許可されないため、より高いレベルのセキュリティでのルーティングと転送の切り分けができます。IPv6 マルチキャスト VRF Lite 機能は、特定の VRF に属するトラフィックの管理とトラブルシューティングを容易にします。

IPv6マルチキャストのプロセス スイッチングおよび高速スイッチング

統合 MFIB は、IPv6 マルチキャストでの PIM-SM および PIM-SSM に対するファスト スイッチングおよびプロセス スイッチングの両サポートを提供するために使用されます。プロセス スイッチングでは、のが各パケットの調査、書き換え、および転送を行う必要があります。最初にパケットが受信され、システム メモリにコピーされます。次に、スイッチがルーティング テーブル内でレイヤ 3 ネットワーク アドレスを検索します。そのあと、レイヤ 2 フレームがネ

クストホップの宛先アドレスで書き換えられ、発信インターフェイスに送信されます。また、は、巡回冗長検査（CRC）も計算します。このスイッチング方式は、IPv6パケットをスイッチングする方式の中でスケーラビリティが最も低い方式です。

IPv6 マルチキャストの高速スイッチングを使用すると、スイッチは、プロセス スイッチングよりも高いパケット転送パフォーマンスを実現できます。従来ルート キャッシュに格納される情報は、IPv6 マルチキャスト スイッチング用にいくつかのデータ構造に格納されます。これらのデータ構造では、ルックアップが最適化され、パケット転送を効率的に行えるようになっています。

IPv6 マルチキャスト転送では、PIM プロトコル ロジックで許可されていれば、最初のパケットのファスト スイッチングが行われます。IPv6マルチキャストの高速スイッチングでは、MACカプセル化ヘッダーが事前に計算されます。IPv6 マルチキャストの高速スイッチングでは、MFIB を使用して、IPv6 送信先プレフィックス ベースのスイッチング判定が行われます。IPv6 マルチキャストの高速スイッチングでは、MFIB に加えて、隣接関係テーブルを使用して、レイヤ 2 アドレッシング情報が付加されます。隣接関係テーブルでは、すべての MFIB エントリのレイヤ 2 ネクストホップ アドレスが管理されます。

隣接が検出されると、隣接関係テーブルにそのデータが入力されます。（ARP などを使用して）隣接エントリが作成されるたびに、その隣接ノードのリンク層ヘッダーが事前に計算され、隣接関係テーブルに格納されます。ルートが決定されると、そのヘッダーはネクスト ホップおよび対応する隣接エントリを指します。そのあと、そのヘッダーはパケット スイッチング時のカプセル化に使用されます。

ロード バランシングと冗長性の両方に対応するようにスイッチが設定されている場合など、ルートには送信先プレフィックスへの複数のパスが存在することがあります。解決されたパスごとに、そのパスのネクストホップインターフェイスに対応する隣接へのポインタが追加されます。このメカニズムは、複数のパスでのロード バランシングに使用されます。

IPv6 マルチキャスト アドレス ファミリのマルチプロトコル BGP

IPv6 マルチキャスト アドレス ファミリのマルチプロトコル BGP 機能では、マルチプロトコル BGP for IPv6 拡張を提供し、IPv4 BGP と同じ機能と機能性をサポートします。マルチキャスト BGP に対する IPv6 拡張には、IPv6 マルチキャスト アドレス ファミリ、ネットワーク層到達可能性情報（NLRI）、および IPv6 アドレスを使用するネクスト ホップ（宛先へのパス内の次のスイッチ）属性のサポートが含まれています。

マルチキャスト BGP は、ドメイン間 IPv6 マルチキャストの配布を可能にする、拡張された BGP です。マルチプロトコル BGP では、複数のネットワーク層プロトコル アドレス ファミリ（IPv6 アドレス ファミリなど）および IPv6 マルチキャスト ルートに関するルーティング情報を伝送します。IPv6 マルチキャスト アドレス ファミリには、IPv6 PIM プロトコルによる RPF ルックアップに使用される複数のルートが含まれており、マルチキャスト BGP IPv6 は、同じドメイン間転送を提供します。ユニキャスト BGP が学習したルートは IPv6 マルチキャストには使用されないため、ユーザーは、BGP で IPv6 マルチキャストを使用する場合は、マルチプロトコル BGP for IPv6 マルチキャストを使用する必要があります。

マルチキャスト BGP 機能は、個別のアドレス ファミリ コンテキストを介して提供されます。Subsequent Address Family Identifier（SAFI）では、属性で伝送されるネットワーク層到達可能

性情報のタイプに関する情報を提供します。マルチプロトコル BGP ユニキャストでは SAFI 1 メッセージを使用し、マルチプロトコル BGP マルチキャストでは SAFI 2 メッセージを使用します。SAFI 1 メッセージは、ルートは IP ユニキャストだけに使用でき、IP マルチキャストには使用できないことを示します。この機能があるため、IPv6 ユニキャスト RIB 内の BGP ルートは、IPv6 マルチキャスト RPF ルックアップでは無視される必要があります。

IPv6 マルチキャスト RPF ルックアップを使用して、異なるポリシーおよびトポロジ（IPv6 ユニキャストとマルチキャストなど）を設定するよう、個別の BGP ルーティング テーブルが維持されています。マルチキャスト RPF ルックアップは、IP ユニキャスト ルート ルックアップと非常によく似ています。

IPv6 マルチキャスト BGP テーブルと関連付けられている MRIB はありません。ただし、必要な場合、IPv6 マルチキャスト BGP は、ユニキャスト IPv6 RIB で動作します。マルチキャスト BGP では、IPv6 ユニキャスト RIB へのルートの挿入や更新は行いません。

IPv6 マルチキャストでの NSF と SSO のサポート

IPv6マルチキャストでは、ノンストップ フォワーディング（NSF）およびステートフル スイッチオーバー（SSO）がサポートされています。

IPv6 マルチキャストの帯域幅ベースの CAC

IPv6 マルチキャストの帯域幅ベースのコール アドミッション制御（CAC）機能は、コスト乗数を使用してインターフェイス単位の mroute ステート リミッタをカウントする手段を実装します。この機能を使用すると、マルチキャスト フローで異なる量の帯域幅が使用されるネットワーク環境で、インターフェイス単位の帯域幅ベースの CAC を提供できます。

この機能では、IPv6 マルチキャスト ステートを詳細に制限および考慮します。この機能を設定すると、IPv6マルチキャストPIM トポロジの着信インターフェイスまたは発信インターフェイスとして使用できる回数にインターフェイスを制限できます。

この機能を使用すると、スイッチ管理者はアクセス リストと一致するステートに対してグローバル制限コスト コマンドを設定して、インターフェイス制限に対してこのようなステートを考慮するときに使用するコスト乗数を指定できます。この機能では、異なる帯域幅要件に応じてコスト乗数を適切に調整することによって、帯域幅ベースのローカル CAC ポリシーを柔軟に実装できます。

IPv6 マルチキャストの実装

IPv6 マルチキャスト ルーティングのイネーブル化

IPV6 マルチキャストルーティングを有効にするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 multicast-routing** 例： スイッチ(config)# **ipv6 multicast-routing**	すべての IPv6 対応インターフェイスでマルチキャスト ルーティングをイネーブルにし、イネーブルになっているすべてのスイッチ インターフェイスで PIM および MLD に対してマルチキャスト転送をイネーブルにします。
ステップ 4	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MLD プロトコルのカスタマイズおよび確認

インターフェイスでの MLD のカスタマイズおよび確認

インターフェイスの MLD をカスタマイズして確認するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： スイッチ(config)# **interface GigabitEthernet 1/0/1**	インターフェイスのタイプと番号を指定し、スイッチをインターフェイスコンフィギュレーションモードにします。

	コマンドまたはアクション	目的
ステップ 4	**ipv6 mld join-group** [*group-address*] [**include** | **exclude**] {*source-address* | **source-list** [*acl*]} 例： `スイッチ(config-if)# ipv6 mld join-group FF04::10`	指定したグループおよび送信元に対して MLD レポートを設定します。
ステップ 5	**ipv6 mld access-group** *access-list-name* 例： `スイッチ(config-if)# ipv6 access-list acc-grp-1`	ユーザーに IPv6 マルチキャストの受信側アクセスコントロールの実行を許可します。
ステップ 6	**ipv6 mld static-group** [*group-address*] [**include** | **exclude**] {*source-address* | source-list [*acl*]} 例： `スイッチ(config-if)# ipv6 mld static-group ff04::10 include 100::1`	指定したインターフェイスにマルチキャストグループのトラフィックをスタティックに転送し、MLD ジョイナがインターフェイスに存在するかのようにインターフェイスが動作するようにします。
ステップ 7	**ipv6 mld query-max-response-time** *seconds* 例： `スイッチ(config-if)# ipv6 mld query-timeout 130`	スイッチがインターフェイスのクエリアとして引き継ぐまでのタイムアウト値を設定します。
ステップ 8	**exit** 例： `スイッチ(config-if)# exit`	このコマンドを 2 回入力して、インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。
ステップ 9	**show ipv6 mld groups [link-local]** [*group-name* | *group-address*] [*interface-type interface-number*] [**detail** | **explicit**] 例： `スイッチ# show ipv6 mld groups GigabitEthernet 1/0/1`	スイッチに直接接続されており、MLD を介して学習したマルチキャスト グループを表示します。
ステップ 10	**show ipv6 mld groups summary** 例： `スイッチ# show ipv6 mld groups summary`	MLD キャッシュに存在する (*, G) および (S, G) メンバーシップ レポートの番号を表示します。
ステップ 11	**show ipv6 mld interface** [*type number*] 例：	インターフェイスのマルチキャスト関連情報を表示します。

	コマンドまたはアクション	目的
	スイッチ# **show ipv6 mld interface GigabitEthernet 1/0/1**	
ステップ 12	**debug ipv6 mld** [*group-name* | *group-address* | *interface-type*] 例： スイッチ# **debug ipv6 mld**	MLD プロトコル アクティビティに対するデバッグをイネーブルにします。
ステップ 13	**debug ipv6 mld explicit** [*group-name* | *group-address* 例： スイッチ# **debug ipv6 mld explicit**	ホストの明示的トラッキングに関連する情報を表示します。
ステップ 14	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MLD グループ制限の実装

インターフェイス単位の MLD 制限とグローバル MLD 制限は相互に独立して機能します。インターフェイス単位の MLD 制限とグローバル MLD 制限の両方を同じスイッチで設定できます。MLD 制限の数は、グローバルの場合もインターフェイス単位の場合も、デフォルトでは設定されません。ユーザーが制限を設定する必要があります。インターフェイス単位のステート制限またはグローバル ステート制限を超えるメンバーシップ レポートは無視されます。

MLD グループ制限のグローバルな実装

MLD グループ制限をグローバルに実装するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 mld** [*vrf vrf-name*] **state-limit** *number*
4. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 mld** [*vrf vrf-name*] **state-limit** *number* 例： スイッチ(config)# **ipv6 mld state-limit 300**	MLD ステートの数をグローバルに制限します。
ステップ 4	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MLD グループ制限のインターフェイス単位での実装

MLD グループ制限をインターフェイスごとに実装するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface type** *number*
4. **ipv6 mld limit** *number* [**except**]*access-list*
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface type** *number* 例： スイッチ(config)# **interface GigabitEthernet 1/0/1**	インターフェイスのタイプと番号を指定し、スイッチをインターフェイス コンフィギュレーション モードにします。

	コマンドまたはアクション	目的
ステップ 4	**ipv6 mld limit** *number* [**except**]*access-list* 例： `スイッチ(config-if)# ipv6 mld limit 100`	MLD ステートの数をインターフェイス単位で制限します。
ステップ 5	**copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

受信側の明示的トラッキングによってホストの動作を追跡するための設定

明示的トラッキング機能を使用すると、スイッチが IPv6 ネットワーク内のホストの動作を追跡できるようになります。また、高速脱退メカニズムを MLD バージョン 2 のホスト レポートで使用できるようになります。

受信側の明示的トラッキングを設定してホストの動作を追跡するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `スイッチ(config)# interface GigabitEthernet 1/0/1`	インターフェイスのタイプと番号を指定し、スイッチをインターフェイス コンフィギュレーション モードにします。
ステップ 4	**ipv6 mld explicit-tracking** *access-list-name* 例： `スイッチ(config-if)# ipv6 mld explicit-tracking list1`	ホストの明示的トラッキングをイネーブルにします。
ステップ 5	**copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

マルチキャスト ユーザ認証およびプロファイル サポートの設定

マルチキャスト ユーザ認証およびプロファイル サポートを設定する前に、次の制約事項を認識しておく必要があります。

- ポート、インターフェイス、VC、または VLAN ID がユーザまたは加入者アイデンティティになります。ホスト名、ユーザ ID、またはパスワードを使用したユーザアイデンティティはサポートされていません。
- IPv6 マルチキャストに対する AAA アクセス コントロールのイネーブル化
- 方式リストの指定およびマルチキャスト アカウンティングのイネーブル化
- スイッチでの未認証マルチキャスト トラフィック受信のディセーブル化
- MLD インターフェイスでの許可ステータスのリセット

IPv6 マルチキャストに対する AAA アクセス コントロールのイネーブル化

特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**aaa new-model** 例： `スイッチ(config)# aaa new-model`	AAA アクセス コントロール システムをイネーブルにします。
ステップ 3	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

方式リストの指定およびマルチキャスト アカウンティングのイネーブル化

次の作業では、AAA 認可およびアカウンティングに使用される方式リストを指定する方法、およびインターフェイス上の指定したグループまたはチャネルでマルチキャストアカウンティングをイネーブルにする方法を示します。

特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**aaa authorization multicast default** [*method3* | *method4*] 例：	AAA 認可をイネーブルにし、IPv6 マルチキャスト ネットワークへのユーザ アクセスを制限するパラメータを設定します。

	コマンドまたはアクション	目的
	`Switch (config)# aaa authorization multicast default`	
ステップ 3	**aaa accounting multicast default** [**start-stop** | **stop- only** [**broadcast**] [*method1*] [*method2*] [*method3*] [*method2]* 例： `Switch (config)# aaa accounting multicast default`	課金、またはRADIUSを使用する際のセキュリティのために、IPv6 マルチキャスト サービスの AAA アカウンティングをイネーブルにします。
ステップ 4	**interface** *type number* 例： `Switch (config)# interface FastEthernet 1/0`	インターフェイスのタイプと番号を指定し、スイッチをインターフェイス コンフィギュレーション モードにします。
ステップ 5	**ipv6 multicast aaa account receive access-list-name** *access-list-name*[**throttle***throttle-number*] 例： `Switch (config-if)# ipv6 multicast aaa account receive list1`	指定したグループまたはチャネル **copy running-config startup-config** で AAA アカウンティングをイネーブルにします。
ステップ 6	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

スイッチでの未認証マルチキャスト トラフィックの受信のディセーブル化

状況によっては、アクセス コントロール プロファイルに従って加入者の認証とチャネルの認可が行われていないかぎり、マルチキャスト トラフィックの受信を防止することが必要となる場合があります。つまり、アクセス コントロール プロファイルで特に指定がなければ、トラフィックを完全になくす必要があります。

未認証グループまたは未認可チャネルからマルチキャスト トラフィックをスイッチが受信しないようにするには、次の作業を実行します。

特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ipv6 multicast** [vrf*vrf-name*] **group-range** [*access-list-name* 例： `Switch (config)# ipv6 multicast group-range`	スイッチのすべてのインターフェイスで未認可グループまたはチャネルのマルチキャスト プロトコル アクションおよびトラフィック転送をディセーブルにします。

	コマンドまたはアクション	目的
ステップ 3	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 での MLD プロキシのイネーブル化

特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ipv6 mld host-proxy** [*group-acl*] 例： `Switch (config)# ipv6 mld host-proxy proxy-group`	MLD プロキシ機能をイネーブルにします。
ステップ 3	**ipv6 mld host-proxy interface**[*group-acl*] 例： `Switch (config)# ipv6 mld host-proxy interface Ethernet 0/0`	RP 上の指定したインターフェイス上で MLD プロキシ機能をイネーブルにします。
ステップ 4	**show ipv6 mld host-proxy**[*interface-type interface-number*] **group** [*group-address*]] 例： `Switch (config)# show ipv6 mld host-proxy Ethernet0/0`	IPv6 MLD ホスト プロキシ情報を表示します。
ステップ 5	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MLD インターフェイスでの許可ステータスのリセット

インターフェイスを指定しない場合は、すべての MLD インターフェイスで認可がリセットされます。

特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**clear ipv6 multicast aaa authorization** [*interface-type interface-number*] 例： `Switch # clear ipv6 multicast aaa authorization FastEthernet 1/0`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MLD トラフィック カウンタのリセット

MLD トラフィックカウンタをリセットするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**clear ipv6 mld traffic** 例： `スイッチ# clear ipv6 mld traffic`	すべての MLD トラフィック カウンタをリセットします。
ステップ 4	**show ipv6 mld traffic** 例： `スイッチ# show ipv6 mld traffic`	MLD トラフィック カウンタを表示します。
ステップ 5	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MLD インターフェイス カウンタのクリア

MLD インターフェイスカウンタをクリアするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**clear ipv6 mld counters** *interface-type* 例： スイッチ# **clear ipv6 mld counters Ethernet1/0**	MLD インターフェイス カウンタをクリアします。
ステップ 4	**copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

PIM の設定

ここでは、PIM の設定方法について説明します。

PIM-SM の設定およびグループ範囲の PIM-SM 情報の表示

PIM-SM を設定し、グループ範囲の PIM-SM 情報を表示するには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	ipv6 pim **rp-address** *ipv6-address*[*group-access-list*] 例： スイッチ(config)# **ipv6 pim rp-address 2001:DB8::01:800:200E:8C6C acc-grp-1**	特定のグループ範囲の PIM RP のアドレスを設定します。
ステップ 4	**exit** 例： スイッチ(config)# **exit**	グローバル コンフィギュレーション モードを終了し、スイッチを特権 EXEC モードに戻します。
ステップ 5	**show ipv6 pim interface** [state-on] [**state-off**] [*type-number*] 例： スイッチ# **show ipv6 pim interface**	PIM に対して設定されたインターフェイスに関する情報を表示します。
ステップ 6	**show ipv6 pim group-map** [*group-name* | group-address] | [*group-range* | *group-mask*] [**info-source {bsr** | **default** | **embedded-rp** | **static**}] 例： スイッチ# **show ipv6 pim group-map**	IPv6 マルチキャスト グループ マッピング テーブルを表示します。
ステップ 7	**show ipv6 pim neighbor** [**detail**] [*interface-type interface-number* | **count**] 例： スイッチ# **show ipv6 pim neighbor**	Cisco IOS ソフトウェアで検出された PIM ネイバーを表示します。
ステップ 8	**show ipv6 pim range-list** [**config**] [*rp-address* | *rp-name*] 例： スイッチ# **show ipv6 pim range-list**	IPv6 マルチキャスト範囲リストに関する情報を表示します。
ステップ 9	**show ipv6 pim tunnel** [*interface-type interface-number*] 例： スイッチ# **show ipv6 pim tunnel**	インターフェイス上の PIM レジスタのカプセル化およびカプセル化解除トンネルに関する情報を表示します。

	コマンドまたはアクション	目的
ステップ 10	**debug ipv6 pim** [*group-name* | *group-address* | **interface** *interface-type* | **bsr** | **group** | **mvpn** | **neighbor**] 例： スイッチ# **debug ipv6 pim**	PIM プロトコル アクティビティに対するデバッグをイネーブルにします。
ステップ 11	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

PIM オプションの設定

PIM オプションを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 pim spt-threshold infinity** [**group-list** *access-list-name*] 例： スイッチ(config)# **ipv6 pim spt-threshold infinity group-list acc-grp-1**	PIM リーフ スイッチが指定したグループの SPT に加入するタイミングを設定します。
ステップ 4	**ipv6 pim accept-register** { **list** *access-list* | **route-map** *map-name*} 例： スイッチ(config)# **ipv6 pim accept-register route-map reg-filter**	RP のレジスタを許可または拒否します。
ステップ 5	**interface** *type number* 例： スイッチ(config)# **interface GigabitEthernet 1/0/1**	インターフェイスのタイプと番号を指定し、スイッチをインターフェイス コンフィギュレーション モードにします。

	コマンドまたはアクション	目的
ステップ 6	**ipv6 pim dr-priority** *value* 例： `スイッチ(config-if)# ipv6 pim dr-priority 3`	PIM スイッチの DR プライオリティを設定します。
ステップ 7	**ipv6 pim hello-interval** *seconds* 例： `スイッチ(config-if)# ipv6 pim hello-interval 45`	インターフェイスにおける PIM hello メッセージの頻度を設定します。
ステップ 8	**ipv6 pim join-prune-interval** *seconds* 例： `スイッチ(config-if)# ipv6 pim join-prune-interval 75`	指定したインターフェイスに対して join および prune の定期的な通知間隔を設定します。
ステップ 9	**exit** 例： `スイッチ(config-if)# exit`	このコマンドを 2 回入力して、インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。
ステップ 10	**ipv6 pim join-prune statistic** [*interface-type*] 例： `スイッチ(config-if)# show ipv6 pim join-prune statistic`	各インターフェイスの最後の集約パケットに関する平均 join-prune 集約を表示します。
ステップ 11	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

双方向 PIM の設定および双方向 PIM 情報の表示

特権 EXEC モードで次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ipv6 pim** [**vrf** *vrf-name*] **rp-address** *ipv6-address* [*group-access-list]* [**bidir**] 例：	特定のグループ範囲の PIM RP のアドレスを設定します。bidir キーワードを使用すると、そのグループ範囲が双方向共有ツリー転送に使用されるようになります。

	コマンドまたはアクション	目的
	`Switch (config) # ipv6 pim rp-address 2001:DB8::01:800:200E:8C6C bidir`	
ステップ 3	**exit** 例： `Switch (config-if) # exit`	グローバル コンフィギュレーション モードを終了し、スイッチを特権 EXEC モードに戻します。
ステップ 4	**show ipv6 pim** [**vrf** *vrf-name*] **df** [*interface-type interface-number*] [*rp-address*] 例： `Switch (config) # show ipv6 pim df`	RP の各インターフェイスの Designated Forwarder（DF）選択ステートを表示します。
ステップ 5	**show ipv6 pim** [**vrf** *vrf-name*] **df winner** [*interface-type interface-number*] [*rp-address*] 例： `Switch (config-if) # show ipv6 pim df winner ethernet 1/0 200::1`	各 RP の各インターフェイスの DF 選択ウィナーを表示します。
ステップ 6	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

PIM トラフィック カウンタのリセット

PIM が誤動作する場合、または予想される PIM パケット数が送受信されていることを確認するために、ユーザーは PIM トラフィック カウンタをクリアできます。トラフィック カウンタがクリアされたら、ユーザーは show ipv6 pim traffic コマンドを入力して、PIM が正しく機能していること、および PIM パケットが正しく送受信されていることを確認できます。

PIM トラフィックカウンタをリセットするには、次の手順を実行します。

手順

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

	コマンドまたはアクション	目的
ステップ 3	**clear ipv6 pim traffic** 例： スイッチ# **clear ipv6 pim traffic**	PIM トラフィック カウンタをリセットします。
ステップ 4	**show ipv6 pim traffic** 例： スイッチ# **show ipv6 pim traffic**	PIM トラフィック カウンタを表示します。
ステップ 5	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

PIM トポロジ テーブルをクリアすることによる MRIB 接続のリセット

MRIB を使用するのに設定は不要です。ただし、特定の状況においては、ユーザーは PIM トポロジ テーブルをクリアして MRIB 接続をリセットし、MRIB 情報を確認する必要がある場合があります。

PIM トポロジテーブルをクリアして MRIB 接続をリセットするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**clear ipv6 pim topology** [*group-name* | *group-address*] 例： スイッチ# **clear ipv6 pim topology FF04::10**	PIM トポロジ テーブルをクリアします。
ステップ 4	**show ipv6 mrib client** [**filter**] [**name** {*client-name* | *client-name* **: client-id**}] 例： スイッチ# **show ipv6 mrib client**	インターフェイスのマルチキャスト関連情報を表示します。

	コマンドまたはアクション	目的
ステップ 5	**show ipv6 mrib route** {**link-local** | **summary** | [*sourceaddress-or-name* | *] [*groupname-or-address*[*prefix-length*]]] 例： `Switchスイッチ# show ipv6 mrib route`	MRIB ルート情報を表示します。
ステップ 6	**show ipv6 pim topology** [*groupname-or-address* [*sourceaddress-or-name*] | **link-local** | **route-count** [**detail**]] 例： `スイッチ# show ipv6 pim topology`	特定のグループまたはすべてのグループの PIM トポロジ テーブル情報を表示します。
ステップ 7	**debug ipv6 mrib client** 例： `スイッチ# debug ipv6 mrib client`	MRIB クライアント管理アクティビティに対するデバッグをイネーブルにします。
ステップ 8	**debug ipv6 mrib io** 例： `スイッチ# debug ipv6 mrib io`	MRIB I/O イベントに対するデバッグをイネーブルにします。
ステップ 9	**debug ipv6 mrib proxy** 例： `スイッチ# debug ipv6 mrib proxy`	分散型スイッチ プラットフォームにおけるスイッチ プロセッサとラインカード間の MRIB プロキシ アクティビティに対するデバッグをイネーブルにします。
ステップ 10	**debug ipv6 mrib route** [*group-name* | *group-address*] 例： `スイッチ# debug ipv6 mrib route`	MRIB ルーティング エントリ関連のアクティビティに関する情報を表示します。
ステップ 11	**debug ipv6 mrib table** 例： `スイッチ# debug ipv6 mrib table`	MRIB テーブル管理アクティビティに対するデバッグをイネーブルにします。
ステップ 12	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

BSR の設定

ここでの作業について、以下に説明します。

BSR の設定および BSR 情報の確認

BSR 情報を設定および確認するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 pim bsr candidate bsr** *ipv6-address*[*hash-mask-length*] [**priority** *priority-value*] 例： スイッチ(config)# **ipv6 pim bsr candidate bsr 2001:DB8:3000:3000::42 124 priority 10**	候補 BSR になるようにスイッチを設定します。
ステップ 4	**interface** *type number* 例： スイッチ(config)# **interface GigabitEthernet 1/0/1**	インターフェイスのタイプと番号を指定し、スイッチをインターフェイス コンフィギュレーション モードにします。
ステップ 5	**ipv6 pim bsr border** 例： スイッチ(config-if)# **ipv6 pim bsr border**	インターフェイスのタイプと番号を指定し、スイッチをインターフェイス コンフィギュレーション モードにします。
ステップ 6	**exit** 例： スイッチ(config-if)# **exit**	このコマンドを 2 回入力して、インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。
ステップ 7	**show ipv6 pim** bsr {**election** | **rp-cache** | **candidate-rp**} 例：	PIM BSR プロトコル処理に関連する情報を表示します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **show ipv6 pim bsr election**	
ステップ 8	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

BSR への PIM RP アドバタイズメントの送信

BSR に PIM RP アドバタイズメントを送信するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 pim bsr candidate rp** *ipv6-address* [**group-list** *access-list-name*] **[priority** *priority-value*] [**interval** seconds] 例： スイッチ(config)# **ipv6 pim bsr candidate rp 2001:DB8:3000:3000::42 priority 0**	BSR に PIM RP アドバタイズメントを送信します。
ステップ 4	**interface** *type number* 例： スイッチ(config)# **interface GigabitEthernet 1/0/1**	インターフェイスのタイプと番号を指定し、スイッチをインターフェイス コンフィギュレーションモードにします。
ステップ 5	**ipv6 pim bsr border** 例： スイッチ(config-if)# **ipv6 pim bsr border**	指定したインターフェイスの任意のスコープの全 BSM に対して境界を設定します。
ステップ 6	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

限定スコープ ゾーン内で BSR を使用できるようにするための設定

スコープゾーン内で使用する BSR を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 pim bsr candidate rp** *ipv6-address* [*hash-mask-length*] [**priority** priority-value] 例： スイッチ(config)# **ipv6 pim bsr candidate bsr 2001:DB8:1:1:4**	候補 BSR になるようにスイッチを設定します。
ステップ 4	**ipv6 pim bsr candidate rp** *ipv6-address* [**group-list** *access-list-name*] [**priority** *priority-value*] [**interval** seconds] 例： スイッチ(config)# **ipv6 pim bsr candidate rp 2001:DB8:1:1:1 group-list list scope 6**	BSR に PIM RP アドバタイズメントを送信するように候補 RP を設定します。
ステップ 5	**interface** *type number* 例： スイッチ(config-if)# **interface GigabitEthernet 1/0/1**	インターフェイスのタイプと番号を指定し、スイッチをインターフェイス コンフィギュレーション モードにします。
ステップ 6	**ipv6 multicast boundary scope** *scope-value* 例： スイッチ(config-if)# **ipv6 multicast boundary scope 6**	指定されたスコープのインターフェイスでマルチキャスト境界を設定します。
ステップ 7	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

BSR スイッチにスコープと RP のマッピングをアナウンスさせるための設定

IPv6 BSR スイッチは、スコープと RP のマッピングを候補 RP メッセージから学習するのではなく、直接アナウンスするようにスタティックに設定できます。ユーザーは、スコープと RP のマッピングをアナウンスするように BSR スイッチを設定して、BSR をサポートしていない RP がその BSR にインポートされるように設定できます。この機能をイネーブルにすると、ローカルの候補 BSR スイッチの既知のリモート RP が、企業の BSR ドメインの外部に配置されている RP を学習できるようになります。

スコープと RP のマッピングをアナウンスするように BSR スイッチを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 pim bsr announced rp** *ipv6-address* [**group-list** *access-list-name*] [**priority** *priority-value*] 例： スイッチ(config)# **ipv6 pim bsr announced rp 2001:DB8:3000:3000::42 priority 0**	指定した候補 RP の BSR からスコープと RP のマッピングを直接アナウンスします。
ステップ 4	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SSM マッピングの設定

SSM マッピング機能をイネーブルにすると、DNS ベースの SSM マッピングが自動的にイネーブルになります。つまり、スイッチは、マルチキャスト MLD バージョン 1 レポートの送信元を DNS サーバーから検索するようになります。

スイッチ設定に応じて、DNS ベースのマッピングまたはスタティック SSM マッピングのいずれかを使用できます。スタティック SSM マッピングを使用する場合は、複数のスタティック SSM マッピングを設定できます。複数のスタティック SSM マッピングを設定すると、一致するすべてのアクセス リストの送信元アドレスが使用されるようになります。

（注） DNS ベースの SSM マッピングを使用するには、スイッチは正しく設定されている DNS サーバーを少なくとも 1 つ見つける必要があります。スイッチは、その DNS サーバーに直接接続される可能性があります。

SSM マッピングを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 mld ssm-map enable** 例： スイッチ(config)# **ipv6 mld ssm-map enable**	設定済みの SSM 範囲内のグループに対して SSM マッピング機能をイネーブルにします。
ステップ 4	**no ipv6 mld ssm-map query dns** 例： スイッチ(config)# **no ipv6 mld ssm-map query dns**	DNS ベースの SSM マッピングをディセーブルにします。
ステップ 5	**ipv6 mld ssm-map static** *access-list source-address* 例： スイッチ(config-if)# **ipv6 mld ssm-map static SSM_MAP_ACL_2 2001:DB8:1::1**	スタティック SSM マッピングを設定します。
ステップ 6	**exit** 例： スイッチ(config-if)# **exit**	グローバル コンフィギュレーション モードを終了し、スイッチを特権 EXEC モードに戻します。
ステップ 7	**show ipv6 mld ssm-map** [*source-address*] 例：	SSM マッピング情報を表示します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **show ipv6 mld ssm-map**	
ステップ 8	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

スタティック mroute の設定

IPv6 のスタティック マルチキャスト ルート（mroute）は、IPv6 スタティック ルートの拡張として実装できます。スイッチを設定する際には、ユニキャスト ルーティング専用としてスタティック ルートを使用するか、マルチキャスト RPF 選択専用としてスタティック マルチキャスト ルートを使用するか、またはユニキャスト ルーティングとマルチキャスト RPF 選択の両方にスタティック ルートを使用するように設定できます。

静的 mroute を設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 route** {*ipv6-prefix / prefix-length ipv6-address* | *interface-type interface-number ipv6-address*]} [*administrative-distance*] [*administrative-multicast-distance* | *unicast* | *multicast*] [**tag** *tag*] 例： スイッチ(config)# **ipv6 route 2001:DB8::/64 6::6 100**	スタティック IPv6 ルートを確立します。この例は、ユニキャスト ルーティングとマルチキャスト RPF 選択の両方に使用されるスタティック ルートを示しています。
ステップ 4	**exit** 例： スイッチ# **exit**	グローバル コンフィギュレーション モードを終了し、スイッチを特権 EXEC モードに戻します。

	コマンドまたはアクション	目的
ステップ 5	**show ipv6 mroute** [link-local | [*group-name* | *group-address* [*source-address* | *source-name*]] **[summary]** [**count**] 例： `スイッチ# show ipv6 mroute ff07::1`	IPv6 マルチキャスト ルーティング テーブルの内容を表示します。
ステップ 6	**show ipv6 mroute** [**link-local** | *group-name* | *group-address*] **active** [*kbps*] 例： `スイッチ(config-if)# show ipv6 mroute active`	スイッチ上のアクティブなマルチキャスト ストリームを表示します。
ステップ 7	**show ipv6 rpf** [*ipv6-prefix*] 例： `スイッチ(config-if)# show ipv6 rpf 2001::1:1:2`	特定のユニキャスト ホスト アドレスおよびプレフィックスの RPF 情報を確認します。
ステップ 8	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 マルチキャストでの MFIB の使用

IPv6 マルチキャスト ルーティングをイネーブルにすると、マルチキャスト転送が自動的にイネーブルになります。

IPv6 マルチキャストでの MFIB の動作の確認

IPv6 マルチキャストで MFIB の動作を確認するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 パスワードを入力します（要求された場合）。
ステップ 2	**show ipv6 mfib** [**link-local** | **verbose** | *group-address-name* | *ipv6-prefix / prefix-length* | *source-address-name* | **count** | **interface** | **status** | **summary**] 例：	IPv6 MFIB での転送エントリおよびインターフェイスを表示します。

	コマンドまたはアクション	目的
	スイッチ# **show ipv6 mfib**	
ステップ 3	**show ipv6 mfib** [all | linkscope | group-name | group-address [source-name | source-address]] **count** 例： スイッチ# **show ipv6 mfib ff07::1**	IPv6 マルチキャスト ルーティング テーブルの内容を表示します。
ステップ 4	**show ipv6 mfib interface** 例： スイッチ# **show ipv6 mfib interface**	IPv6マルチキャスト対応インターフェイスとその転送ステータスに関する情報を表示します。
ステップ 5	**show ipv6 mfib status** 例： スイッチ# **show ipv6 mfib status**	一般的なMFIB設定と動作ステータスを表示します。
ステップ 6	**show ipv6 mfib summary** 例： スイッチ# **show ipv6 mfib summary**	IPv6 MFIB エントリおよびインターフェイスの数に関するサマリー情報を表示します。
ステップ 7	**debug ipv6 mfib** [*group-name* | *group-address*] [**adjacency** | **db** | **fs** | **init** | **interface** | **mrib** [**detail**] | **nat** | **pak** | **platform** | **ppr** | **ps** | **signal** | **table**] 例： スイッチ# **debug ipv6 mfib FF04::10 pak**	IPv6 MFIB に対するデバッグ出力をイネーブルにします。

MFIB トラフィック カウンタのリセット

MFIB トラフィックカウンタをリセットするには、次の手順を実行します。

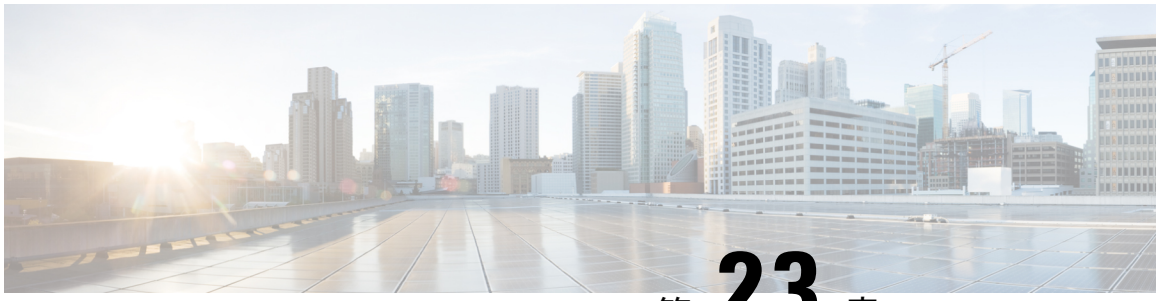
手順

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

	コマンドまたはアクション	目的
ステップ 2	**clear ipv6 mfib counters** [*group-name* | **group-address** [*source-address* | *source-name*]] 例： スイッチ# **clear ipv6 mfib counters FF04::10**	アクティブなすべての MFIB トラフィック カウンタをリセットします。

第 IV 部

レイヤ 2



第 23 章

IEEE 802.1Q トンネリングおよびレイヤ 2 プロトコル トンネリングの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

トンネリング設定の前提条件

ここでは、IEEE 802.1Q およびレイヤ 2 プロトコル トンネリングを設定するための前提条件と考慮事項について説明します。

IEEE 802.1Q トンネリング

IEEE 802.1Q トンネリングはレイヤ 2 パケット スイッチングで適切に動作しますが、一部のレイヤ 2 機能およびレイヤ 3 スイッチングの間には非互換性があります。

- トンネル ポートはルーテッド ポートにできません。
- IEEE 802.1Q トンネル ポートを含む VLAN では IP ルーティングがサポートされません。トンネルポートから受信したパケットは、レイヤ2情報だけに基づいて転送されます。トンネル ポートを含むdevice仮想インターフェイス（SVI）でルーティングが有効になっている場合、トンネルポートから受信したタグなしIPパケットは、deviceに認識されてルーティングされます。カスタマーは、ネイティブ VLAN を介してインターネットにアクセスできます。このアクセスが必要ない場合は、トンネル ポートを含む VLAN で SVI を設定しないでください。
- フォールバック ブリッジングは、トンネル ポートでサポートされません。トンネル ポートから受信したすべてのIEEE 802.1Qタグ付きパケットはIP以外のパケットとして扱われるので、トンネル ポートが設定されている VLAN でフォールバック ブリッジングが有効である場合、IP パケットは VLAN を越えて不適切にブリッジングされます。このため、トンネル ポートを含む VLAN ではフォールバック ブリッジングを有効にしないでください。
- トンネル ポートでは IP アクセス コントロール リスト（ACL）がサポートされません。
- レイヤ 3 の Quality of Service（QoS）ACL およびレイヤ 3 情報に関連する他の QoS 機能は、トンネル ポートではサポートされていません。MAC ベース QoS はトンネル ポートでサポートされます。
- IEEE 802.1Q 設定が EtherChannel ポート グループ内で矛盾しない場合、EtherChannel ポート グループにはトンネル ポートとの互換性があります。
- ポート集約プロトコル（PAgP）、Link Aggregation Control Protocol（LACP）、単一方向リンク検出（UDLD）は、IEEE 802.1Q トンネル ポートでサポートされます。
- トンネル ポートとトランク ポートで非対称リンクを手動で設定する必要があるので、ダイナミック トランキング プロトコル（DTP）にはIEEE 802.1Q トンネリングとの互換性がありません。
- VLANトランキング プロトコル（VTP）は、非対称リンクで接続されているデバイス間、またはトンネルを通して通信を行うデバイス間で動作しません。
- IEEE 802.1Q トンネル ポートでは、ループバック検出がサポートされます。
- IEEE 802.1Q トンネル ポートとしてポートを設定すると、スパニングツリー ブリッジ プロトコルデータユニット（BPDU）フィルタリングがインターフェイスで自動的に有効になります。Cisco Discovery Protocol（CDP）および Layer Link Discovery Protocol（LLDP）は、インターフェイスで自動的に無効になります。

レイヤ 2 プロトコル トンネリング

- deviceでは、CDP、STP（Multiple STP（MSTP）を含む）、VTP のトンネリングがサポートされます。プロトコル トンネリングはデフォルトでディセーブルになっていますが、IEEE 802.1Q トンネル ポート、またはアクセス ポートでプロトコルごとにイネーブルにできます。
- deviceでは、switchport モードが dynamic auto または dynamic desirable に設定されているポートにおいて、レイヤ 2 プロトコル トンネリングがサポートされません。
- DTP はレイヤ 2 プロトコル トンネリングと互換性がありません。
- サービスプロバイダ ネットワークのアウトバウンド側のエッジ devicesでは、適切なレイヤ 2 プロトコル情報および MAC アドレス情報が復元され、同じメトロ VLAN のすべてのトンネル ポートおよびアクセス ポートにパケットが転送されます。
- サードパーティ ベンダー devicesとの相互運用性のため、deviceではレイヤ 2 プロトコル トンネル バイパス機能がサポートされます。バイパス モードでは、プロトコル トンネリングの制御方法が異なるベンダー devicesに制御 PDU が透過的に転送されます。deviceの入力ポートでレイヤ 2 プロトコル トンネリングがイネーブルになっている場合、出力トランク ポートは特殊なカプセル化を使用してトンネリング パケットを転送します。出力トランク ポートでもレイヤ 2 プロトコル トンネリングをイネーブルにすると、この動作がバイパスされて、deviceは加工や変更を行わずに制御 PDU を転送します。
- deviceでは、ポイントツーポイント ネットワーク トポロジのエミュレートに関して PAgP、LACP、UDLD のトンネリングがサポートされます。プロトコル トンネリングはデフォルトでディセーブルになっていますが、IEEE 802.1Q トンネル ポート、またはアクセス ポートでプロトコルごとにイネーブルにできます。
- PAgP トンネリングまたは LACP トンネリングの場合は、リンク障害検出を高速にするため、インターフェイスで UDLD もイネーブルにすることを推奨します。
- PAgP パケット、LACP パケット、UDLD パケットのレイヤ 2 プロトコル トンネリングでは、ループバック検出はサポートされません。
- IEEE 802.1Q 設定が EtherChannel ポート グループ内で矛盾しない場合、EtherChannel ポート グループにはトンネル ポートとの互換性があります。
- 独自の宛先 MAC アドレスでカプセル化された PDU が、レイヤ 2 トンネリングがイネーブルになっているトンネル ポートまたはアクセス ポートから受信される場合、トンネル ポートは、ループを防止するためにシャットダウンされます。このポートは、プロトコル用に設定されたシャットダウンしきい値に達した場合にもシャットダウンされます。**shutdown** コマンドに続けて **no shutdown** コマンドを入力すると、ポートを手動で再びイネーブルにできます。errdisable recovery がイネーブルである場合は、指定された時間間隔で動作が再試行されます。
- カプセル化が解除された PDU だけがカスタマー ネットワークに転送されます。サービスプロバイダ ネットワーク上で動作しているスパニングツリー インスタンスでは、BPDU

がトンネル ポートに転送されません。CDP パケットはトンネル ポートから転送されません。

- インターフェイスでプロトコル トンネリングがイネーブルである場合は、カスタマーネットワークによって生成されたPDU用に、プロトコルごとのシャットダウンしきい値やポートごとのシャットダウンしきい値を設定できます。制限を超えると、ポートはシャットダウンされます。QoS ACL およびポリシー マップをトンネル ポートで使用すると、BPDU レートを制限することもできます。
- インターフェイスでプロトコル トンネリングがイネーブルである場合は、カスタマーネットワークによって生成された PDU 用に、プロトコルごとのドロップしきい値やポートごとのドロップしきい値を設定できます。制限を超えると、ポートが PDU を受信するレートがドロップしきい値未満になるまで、ポートで PDU がドロップされます。
- トンネリングされたPDU（特にSTP BPDU）は、カスタマーの仮想ネットワークが正しく動作するためにすべてのリモートサイトに配信される必要があるので、同じトンネルポートから受信されるデータ パケットよりも PDU のプライオリティをサービスプロバイダ ネットワーク内で高くできます。デフォルトの場合、PDUではデータ パケットと同じCoS値が使用されます。

EtherChannel のレイヤ 2 トンネリング

EtherChannel の自動作成を容易にするためにレイヤ 2 ポイントツーポイント トンネリングを設定するには、サービスプロバイダ（SP）エッジスイッチおよびカスタマー deviceの両方を設定する必要があります。

トンネリングについて

IEEE 802.1Q およびレイヤ 2 プロトコルの概要

バーチャル プライベート ネットワーク（VPN）では、多くの場合にイーサネットベースの共有インフラストラクチャである企業規模の接続に、プライベート ネットワークと同じセキュリティ、プライオリティ、信頼性、管理の容易さが提供されます。トンネリングは、サービスプロバイダのネットワークを越えて複数のカスタマーのトラフィックを運び、その他のカスタマーのトラフィックに影響を与えずに、それぞれのカスタマーの VLAN およびレイヤ 2 プロトコルの設定を維持する必要があるサービス プロバイダ用に設計された機能です。

（注） IEEE 802.1Q およびレイヤ 2 プロトコル トンネリングは Cisco Catalyst 3560-CX スイッチでのみサポートされています。

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

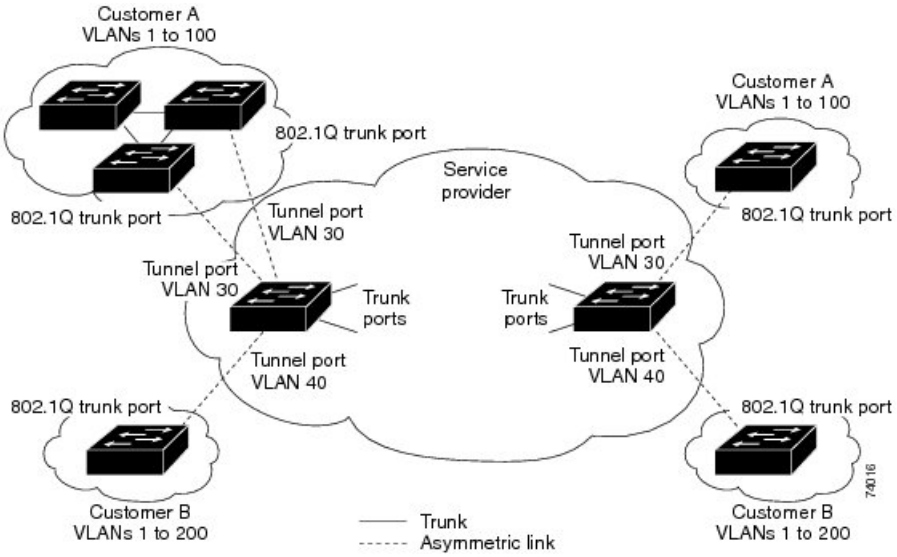
IEEE 802.1Q トンネリング

サービス プロバイダーのビジネス カスタマーには、多くの場合、サポートする VLAN ID および VLAN の数に固有の要件があります。同一サービスプロバイダー ネットワークのさまざまなカスタマーが必要とする VLAN 範囲は重複し、インフラストラクチャを通るカスタマーのトラフィックは混合してしまうことがあります。それぞれのカスタマーに VLAN ID の固有の範囲を割り当てると、カスタマーの設定が制限され、IEEE 802.1Q 仕様の VLAN 制限（4096）を簡単に超えてしまうことがあります。

サービス プロバイダーは、IEEE 802.1Q トンネリング機能を使用すると、単一の VLAN を使用して、複数の VLAN を含むカスタマーをサポートできます。カスタマーの VLAN ID は、同一 VLAN にあるように見えても保護され、さまざまなカスタマーのトラフィックは、サービスプロバイダー ネットワーク内で区別されます。IEEE 802.1Q トンネリングを使用する場合、VLAN-in-VLAN 階層構造およびタグ付きパケットへの再タグ付けによって、VLAN スペースを拡張できます。IEEE 802.1Q トンネリングをサポートするように設定したポートは、トンネル ポートと呼ばれます。トンネリングを設定する場合は、トンネリング専用の VLAN ID にトンネルポートを割り当てます。それぞれのカスタマーには別個のサービスプロバイダー VLAN ID が必要ですが、その VLAN ID ではすべてのカスタマーの VLAN がサポートされます。

適切な VLAN ID で通常どおりにタグ付けされたカスタマーのトラフィックは、カスタマー デバイスの IEEE 802.1Q トランク ポートからサービスプロバイダーのエッジ deviceのトンネル ポートに発信されます。カスタマーデバイスとエッジ device間のリンクは、片方が IEEE 802.1Q トランク ポートとして設定され、もう一方がトンネル ポートとして設定されるため、非対称です。それぞれのカスタマーに固有のアクセス VLAN ID には、トンネル ポート インターフェイスを割り当てます。

図 *15*：サービス プロバイダー ネットワークにおける *IEEE 802.1Q* トンネル ポート



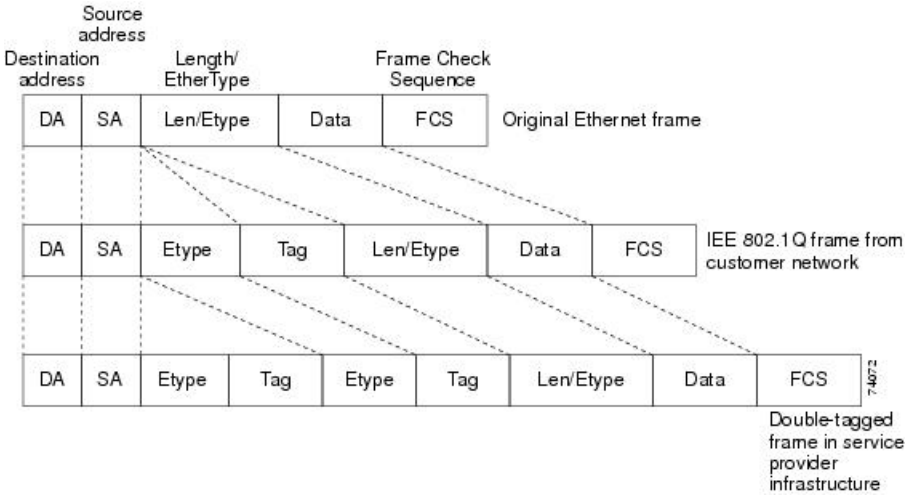
カスタマーのトランク ポートからサービス プロバイダーのエッジ deviceのトンネル ポートに発信されるパケットには、通常、適切な VLAN ID とともに IEEE 802.1Q タグが付いています。これらのタグ付きパケットは、device内部ではそのまま保持され、トランク ポートを出てサー

ビスプロバイダー ネットワークに入る時点で、カスタマーに固有の VLAN ID を含む、IEEE 802.1Q タグのもう 1 つのレイヤ（メトロ タグと呼ばれる）でカプセル化されます。カスタマーの元の IEEE 802.1Q タグは、カプセル化されたパケット内で保護されます。このため、サービスプロバイダー ネットワークに入るパケットには、カスタマーのアクセス VLAN ID を含む外部（メトロ）タグ、および着信トラフィックのものである内部 VLAN ID という、二重のタグが付きます。

二重タグ パケットがサービスプロバイダー コア deviceの別のトランク ポートに入ると、device がパケットを処理するときに外部タグが外されます。パケットがその同じコア deviceの別のトランク ポートを出るとき、同じメトロ タグがパケットに再び追加されます。

図 *16*：元の（通常）イーサネット パケット、*IEEE 802.1Q* イーサネット パケット、二重タグ イーサネット パケットの形式

この図は、二重タグ付きパケットのタグ構造を示しています。



パケットがサービス プロバイダー出力deviceのトランク ポートに入ると、deviceがパケットを内部処理する間に外部タグが再び外されます。ただし、パケットがエッジ deviceのトンネル ポートからカスタマー ネットワークに送信されるとき、メトロ タグは追加されません。パケットは通常の IEEE 802.1Q タグ フレームとして送信され、カスタマー ネットワーク内で元の VLAN 番号は保護されます。

上記のネットワークの図では、カスタマー A に VLAN 30、カスタマー B に VLAN 40 が割り当てられています。エッジデバイスのトンネルポートに入る、IEEE 802.1Q タグが付いたパケットは、サービスプロバイダ ネットワークに入るとき、VLAN ID 30 または 40 を適切に含む外部タグ、および VLAN 100 などの元の VLAN 番号を含む内部タグが付いて二重タグになります。カスタマー A とカスタマー B の両方が、それぞれのネットワーク内で VLAN 100 を含んでいても、外部タグが異なるので、サービスプロバイダー ネットワーク内で区別されます。それぞれのカスタマーは、その他のカスタマーが使用する VLAN 番号スペース、およびサービスプロバイダー ネットワークが使用する VLAN 番号スペースから独立した、独自の VLAN 番号スペースを制御します。

アウトバウンド トンネル ポートでは、カスタマーのネットワーク上の元の VLAN 番号が回復されます。トンネリングとタグ付けを複数レベルにすることもできますが、このリリースのデバイスでは 1 レベルだけがサポートされます。

カスタマー ネットワークから発信されるトラフィックにタグ（ネイティブ VLAN フレーム）が付いていない場合、そのパケットのブリッジングまたはルーティングは通常パケットとして行われます。エッジデバイスのトンネルポートを通ってサービスプロバイダ ネットワークに入るすべてのパケットは、タグが付いていないか、IEEE 802.1Q ヘッダーですでにタグが付いているかに関係なく、タグなしパケットとして扱われます。パケットは、IEEE 802.1Q トランク ポートでサービスプロバイダー ネットワークを通じて送信される場合、メトロ タグ VLAN ID（トンネル ポートのアクセス VLAN に設定）でカプセル化されます。メトロ タグの優先度フィールドは、トンネル ポートで設定されているインターフェイス サービス クラス（CoS）優先度に設定されます（設定されていない場合、デフォルトはゼロです）。

IEEE 802.1Q トンネリング設定時の注意事項

IEEE 802.1Q トンネリングを設定する場合は、カスタマー デバイスおよびエッジ deviceの間で非対称リンクを常に使用する必要があります。カスタマーデバイスのポートを IEEE 802.1Q トランク ポートに、エッジ deviceのポートをトンネル ポートとして設定してください。

トンネリングに使用する VLAN だけにトンネル ポートを割り当ててください。

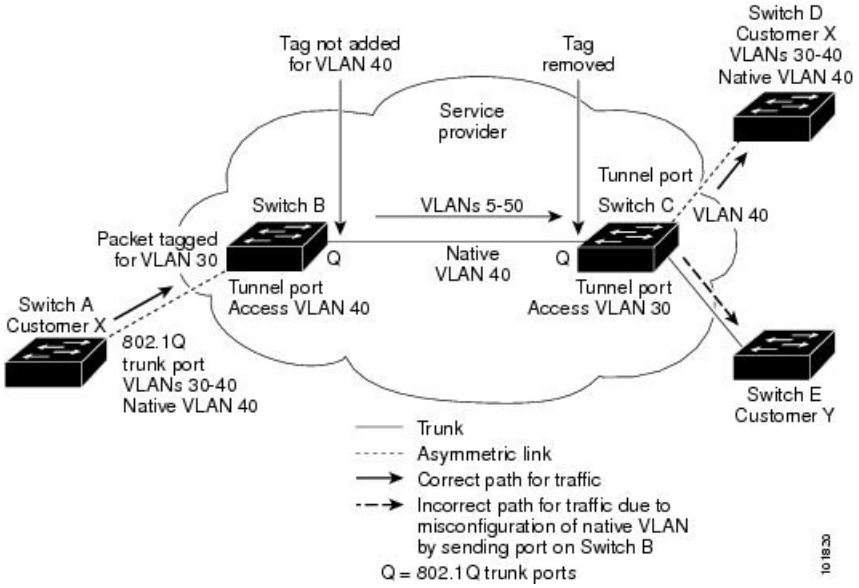
ネイティブ VLANおよび最大伝送単位（MTU）の設定要件については、次の項で説明します。

ネイティブ VLAN

エッジ deviceで IEEE 802.1Q トンネリングを設定する場合、サービスプロバイダー ネットワークにパケットを送信するために、IEEE 802.1Q トランク ポートを使用する必要があります。ただし、サービスプロバイダー ネットワークのコアを通過するパケットは、IEEE 802.1Q トランク、ISL トランク、非トランキング リンクのいずれかで送信できます。コア devicesで IEEE 802.1Q トランクを使用する場合、IEEE 802.1Q トランクのネイティブ VLAN は、同一deviceの非トランキング（トンネリング）ポートのネイティブ VLAN と同じであってはなりません。これは、ネイティブ VLANのトラフィックは、IEEE 802.1Q 送信トランク ポートではタグ付けされないためです。

次のネットワーク図で、VLAN 40 は、サービスプロバイダー ネットワークの入力エッジdevice（デバイス B）にある、カスタマー X からの IEEE 802.1Q トランク ポートのネイティブ VLAN として設定されています。カスタマー X のデバイス A は、VLAN 30 のタグ付きパケットを、アクセス VLAN 40 に属する、サービスプロバイダー ネットワークのデバイス B の入力トンネル ポートに送信します。トンネル ポートのアクセス VLAN（VLAN 40）は、エッジ deviceのトランク ポートのネイティブ VLAN（VLAN 40）と同じであるため、トンネル ポートから受信したタグ付きパケットにメトロ タグが追加されません。パケットには VLAN 30 タグだけが付いて、サービスプロバイダー ネットワークで出力エッジ device（デバイス C）のトランク ポートに送信され、出力device トンネルによってカスタマー Y に間違って送信されます。

図 *17 : IEEE 802.1Q* トンネリングおよびネイティブ *VLAN* に潜在する問題



この問題の解決方法は次のとおりです。

-
- **vlan dot1q tag native** グローバル コンフィギュレーション コマンドを使用することで、（ネイティブ VLAN を含む）IEEE 802.1Q トランクから発信されるすべてのパケットがタグ付けされるようにエッジdevicesを設定します。すべての IEEE 802.1Q トランクでネイティブ VLAN パケットにタグを付けるようにdevicesを設定した場合、devicesはタグなしパケットを受け入れますが、タグ付きパケットだけを送信します。
- エッジ devicesのトランク ポートのネイティブ VLAN ID が、カスタマー VLAN 範囲に含まれないようにしてください。たとえばトランク ポートが VLAN100 ～ 200 のトラフィックを運ぶ場合は、この範囲以外の番号をネイティブ VLAN に割り当てます。

システム MTU

device上のトラフィックに関するデフォルトのシステム MTU は、1500 バイトです。

system mtu jumbo グローバル コンフィギュレーション コマンドを使用すると、10 ギガビット イーサネット ポートおよびギガビット イーサネット ポートで 1500 バイトを超えるフレームをサポートするように設定できます。

システム MTU 値とシステム ジャンボ MTU 値には、IEEE 802.1Q ヘッダーは含まれていません。IEEE 802.1Q トンネリング機能では、メトロ タグが追加されるとフレーム サイズが 4 バイト増加するため、システム最大伝送単位サイズとシステム ジャンボ最大伝送単位サイズに最低 4 バイトを追加することによって、サービスプロバイダ ネットワークのすべてのdevicesが最大フレームを処理できるように設定する必要があります。

たとえば、deviceは、次のいずれかの設定で、1496 バイトの最大フレーム サイズをサポートします。

- deviceのシステムジャンボ最大伝送単位値が 1500 バイトで、 **switchport mode dot1q tunnel** インターフェイス コンフィギュレーション コマンドを使って 10 ギガビットイーサネットまたはギガビットイーサネット device ポートが設定されている。
- device メンバのシステム最大伝送単位値が 1500 バイトで、**switchport mode dot1q tunnel** インターフェイス コンフィギュレーション コマンドを使ってメンバのファストイーサネットポートが設定されている。

IEEE 802.1Q トンネリングのデフォルト設定

デフォルトでは、デフォルト switchport モードが dynamic auto であるため、IEEE 802.1Q トンネルはディセーブルです。すべての IEEE 802.1Q トランク ポートにおける IEEE 802.1Q ネイティブ VLAN パケットのタグ付けもディセーブルです。

レイヤ 2 プロトコル トンネリングの概要

サービスプロバイダー ネットワークを越えて接続されている、さまざまなサイトに散在するカスタマーは、さまざまなレイヤ 2 プロトコルを使用してトポロジをスケールし、すべてのリモート サイトおよびローカル サイトを含める必要があります。STP を適切に動作させる必要があり、サービスプロバイダー ネットワークを越えたローカル サイトおよびすべてのリモートサイトを含む、適切なスパニングツリーをすべてのVLANで構築する必要があります。Cisco Discovery Protocol（CDP）では、隣接するシスコ デバイスをローカル サイトおよびリモートサイトから検出する必要があります。VLAN トランキング プロトコル（VTP）では、カスタマー ネットワークのすべてのサイトで矛盾しないVLAN 設定を提供する必要があります。

プロトコル トンネリングが有効である場合、サービスプロバイダ ネットワークのインバウンド側エッジ devicesでは、特殊 MAC アドレスでレイヤ 2 プロトコルパケットがカプセル化され、サービスプロバイダ ネットワークに送信されます。ネットワークのコア devicesでは、このパケットが処理されずに通常のパケットとして転送されます。CDP、STP、VTP のレイヤ 2 プロトコル データ ユニット（PDU）は、サービスプロバイダ ネットワークをまたがり、サービスプロバイダ ネットワークのアウトバウンド側のカスタマー devicesに配信されます。同一パケットは同じVLANのすべてのカスタマー ポートで受信され、次のような結果になります。

- それぞれのカスタマー サイトのユーザは STP を適切に実行でき、すべての VLAN では（ローカルサイトだけではなく）すべてのサイトからのパラメータに基づいて、正しいスパニングツリーが構築されます。
- CDP では、サービスプロバイダー ネットワークによって接続されているその他のシスコ デバイスに関する情報が検出されて表示されます。
- VTP ではカスタマー ネットワーク全体で一貫した VLAN 設定が提供され、サービス プロバイダを通してすべてのdevicesに伝播されます。

（注） サードパーティ ベンダーとの相互運用性を提供するには、レイヤ 2 プロトコルトンネル バイパス機能を使用します。バイパス モードでは、プロトコル トンネリングの制御方法が異なるベンダー devicesに、制御 PDU が透過的に転送されます。バイパス モードを実装するには、出力トランク ポートでレイヤ 2 プロトコル トンネリングを有効にします。レイヤ 2 プロトコル トンネリングがトランク ポートで有効の場合、カプセル化された MAC アドレスが削除されて、プロトコル パケットに通常の MAC アドレスを持つようになります。

レイヤ 2 プロトコル トンネリングは個別に使用できます。レイヤ 2 プロトコル トンネリングでは、IEEE 802.1Q トンネリングを向上させることができます。IEEE 802.1Q トンネリング ポートでプロトコル トンネリングが有効になっていない場合、サービスプロバイダ ネットワークの受信側のリモート devicesでは PDU が受信されず、STP、CDP、VTP を適切に実行できません。プロトコルのトンネリングが有効である場合、それぞれのカスタマー ネットワークのレイヤ2プロトコルは、サービスプロバイダ ネットワーク内で動作しているものから完全に区別されます。IEEE 802.1Q トンネリングでサービスプロバイダ ネットワークを通してトラフィックを送信する、さまざまなサイトのカスタマー devicesでは、カスタマー VLAN が完全に認識されます。IEEE 802.1Q トンネリングを使用しない場合は、アクセス ポートでカスタマー device に接続し、サービスプロバイダのアクセス ポートでトンネリングを有効にすることで、レイヤ 2 プロトコル トンネリングを有効にできます。

たとえば、次の図（レイヤ 2 プロトコル トンネリング）では、カスタマー X の 4 つのdevices が同じVLAN上にあり、サービスプロバイダ ネットワークを通して互いに接続されています。ネットワークで PDU がトンネリングされない場合、ネットワークの遠端側のdevicesでは、STP、CDP、VTP を適切に実行できません。たとえば、カスタマーXのサイト1のdeviceでは、VLAN の STP は、カスタマー X のサイト 2 のdevicesに基づくコンバージェンス パラメータを考慮せずに、サイト 1 のdevice上にスパニングツリーを構築します。これにより、「適切なコンバージェンスを含まないレイヤ2ネットワーク トポロジ」の図に示されているようなトポロジになる可能性があります。

図 *18 :* レイヤ *2* プロトコル トンネリング

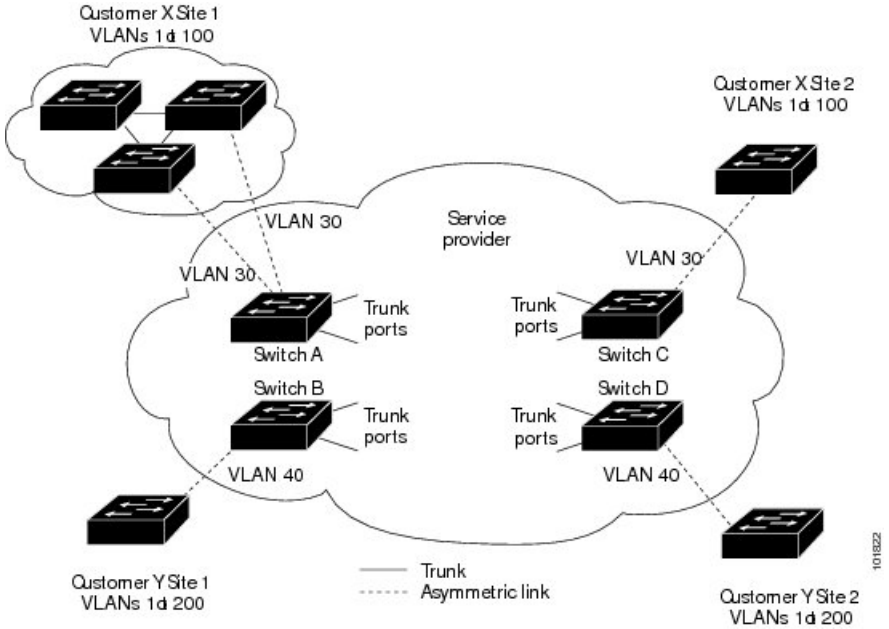
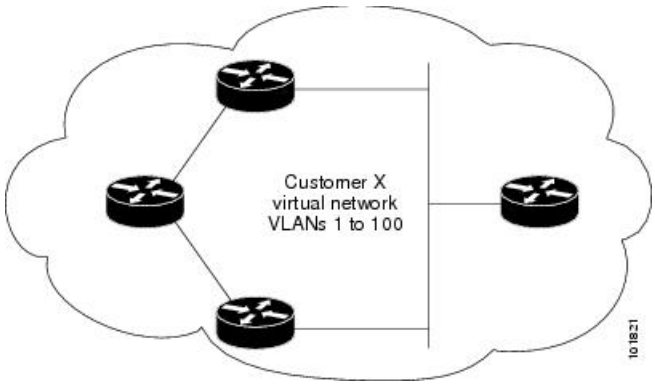


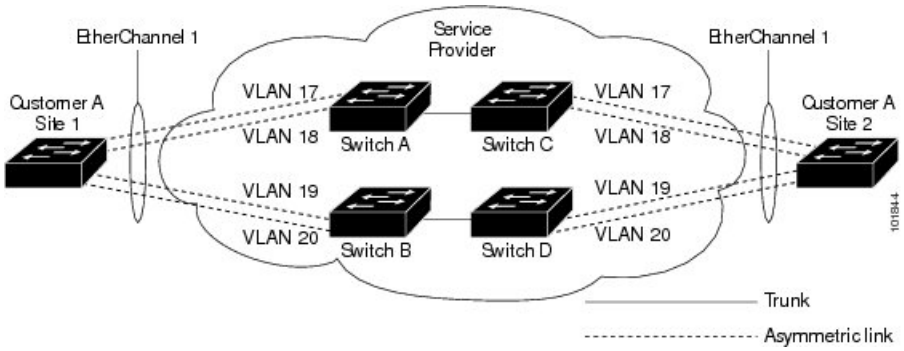
図 *19 :* 適切なコンバージェンスを含まないレイヤ *2* ネットワーク トポロジ



サービスプロバイダ ネットワークでは、レイヤ 2 プロトコル トンネリングを使用し、ポイントツーポイント ネットワーク トポロジをエミュレートして、EtherChannel の作成を向上させることができます。サービスプロバイダ device でプロトコル トンネリング（PAgP または LACP）を有効にすると、リモート カスタマー devices では PDU が受信され、EtherChannel の自動作成をネゴシエーションできるようになります。

たとえば、次の図（EtherChannels のレイヤ 2 プロトコル トンネリング）では、カスタマー A の 2 つのdevicesが同じ VLAN 上にあり、サービスプロバイダ ネットワークを介して接続されています。ネットワークで PDU がトンネリングされると、ネットワークの遠端側のdevicesでは、専用回線を必要とせずに EtherChannel の自動作成をネゴシエーションできます。

図 *20 : EtherChannel* のレイヤ *2* プロトコル トンネリング



ポートでのレイヤ 2 プロトコル トンネリング

サービスプロバイダ ネットワークのエッジ devicesで、カスタマーに接続されているポートにおいて、レイヤ2プロトコル トンネリングを（プロトコルごとに）イネーブルにできます。カスタマー devicesに接続されているサービス プロバイダ エッジ deviceでは、トンネリング処理が実行されます。エッジ device トンネル ポートは、カスタマーの IEEE 802.1Q トランク ポートに接続されます。エッジ device アクセス ポートは、カスタマー アクセス ポートに接続されます。カスタマーdevicesに接続されるエッジ deviceでは、トンネリング処理が実行されます。

アクセス ポートまたはトンネル ポートのいずれかとして設定されているポートでは、レイヤ2プロトコル トンネリングをイネーブルにできます。**switchport mode dynamic auto** モード（デフォルトモード）または **switchport mode dynamic desirable** モードに設定されているポートでは、レイヤ 2 プロトコルトンネリングをイネーブルにできません。

deviceでは、CDP、STP、VTP のレイヤ 2 プロトコル トンネリングがサポートされます。ポイントツーポイント ネットワーク トポロジのエミュレートの場合は、PAgP、LACP、UDLD のプロトコルもサポートされます。deviceは、LLDP のレイヤ2プロトコル トンネリングをサポートしません。

（注） PAgP、LACP、UDLD プロトコル トンネリングでは、ポイントツーポイント トポロジのエミュレートだけが目的です。設定を間違えたことによりトンネリング パケットが多くのポートに送信されると、ネットワーク障害が発生する可能性があります。

レイヤ2プロトコルがイネーブルになっているポート経由でサービスプロバイダのインバウンド エッジ deviceに入ったレイヤ 2 PDU が、トランク ポートからサービスプロバイダ ネットワークに出て行くとき、deviceでは、カスタマー PDU 宛先 MAC アドレスが、周知のシスコ固有のマルチキャスト アドレス（01-00-0c-cd-cd-d0）で上書きされます。IEEE 802.1Q トンネリングがイネーブルである場合、パケットにはタグが二重に付きます。このうち外部タグはカスタマーのメトロ タグ、内部タグはカスタマーの VLAN タグです。コア devicesでは内部タグが無視され、同じメトロ VLAN のすべてのトランク ポートにパケットが転送されます。アウトバウンド側のエッジ devicesでは、適切なレイヤ 2 プロトコル情報および MAC アドレス情報が復元され、同じメトロ VLAN のすべてのトンネル ポートまたはすべてのアクセス ポートにパ

ケットが転送されます。このため、レイヤ 2 PDU はそのまま残り、サービスプロバイダ インフラストラクチャを越えてカスタマー ネットワークの反対側に配信されます。

レイヤ2プロトコル トンネリングの概要（461 ページ）のレイヤ2プロトコル トンネリングの図を参照してください（それぞれアクセス VLAN 30、40 のカスタマー X とカスタマー Y）。非対称リンクにより、サイト 1 のカスタマーは、サービスプロバイダ ネットワークのエッジ devicesに接続されています。サイト 1 のカスタマー Y からデバイス B に発信されたレイヤ 2 PDU（たとえば BPDU）は、周知の MAC アドレスが宛先 MAC アドレスになっている二重タグ パケットとしてインフラストラクチャに転送されます。この二重タグ パケットには、40 というメトロ VLAN タグ、および VLAN 100 などの内部 VLAN タグが付いています。二重タグ パケットがデバイス D に入ると、外部 VLAN タグ 40 が外されて、周知の MAC アドレスがそれぞれのレイヤ 2 プロトコル MAC アドレスで置き換わり、パケットは、VLAN 100 の 1 重タグ フレームとしてサイト 2 のカスタマー Y に送信されます。

また、カスタマー deviceのアクセス ポートまたはトランク ポートに接続されているエッジ deviceのアクセス ポートでも、レイヤ2プロトコル トンネリングをイネーブルにできます。この場合は、カプセル化プロセスとカプセル開放プロセスが、前の段落で説明したものと同じですが、パケットはサービスプロバイダーネットワークで二重タグになりません。カスタマー固有のアクセス VLAN タグの 1 重タグになります。

レイヤ 2 プロトコル トンネリングのデフォルト設定

次の表に、レイヤ 2 プロトコル トンネリングのデフォルト設定を記載します。

表 *39 :* レイヤ *2* イーサネット インターフェイス *VLAN* のデフォルト設定

機能	デフォルト設定
レイヤ 2 プロトコル トンネリング	ディセーブル。
シャットダウンしきい値	未設定。
ドロップしきい値	未設定。
CoS 値	インターフェイスで CoS 値が設定されている場合は、その値がレイヤ 2 プロトコル トンネリングの BPDU CoS 値を設定するために使用されます。インターフェイス レベルで CoS 値が設定されてない場合は、L2プロトコル トンネリング BPDU の CoS マーキングのデフォルト値は5になります。これはデータ トラフィックに適用されません。

トンネリングの設定方法

IEEE 802.1Q トンネリング ポートの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport access vlan** *vlan-id*
5. **switchport mode dot1q-tunnel**
6. **exit**
7. **vlan dot1q tag native**
8. **end**
9. 次のいずれかを使用します。
   - **show dot1q-tunnel**
   - **show running-config interface**
10. **show vlan dot1q tag native**
11. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet2/0/1**	トンネル ポートとして設定するインターフェイスのインターフェイス コンフィギュレーション モードを開始します。これは、カスタマーdeviceに接続するサービスプロバイダ ネットワーク内のエッジ ポートである必要があります。有効なインターフェイスには、物理インターフェイスおよびポートチャ

	コマンドまたはアクション	目的
		ネル論理インターフェイス（ポート チャネル 1 ～ 48）が含まれます。
ステップ 4	**switchport access vlan** *vlan-id* 例： `スイッチ(config-if)# switchport access vlan 2`	インターフェイスがトランキングを停止した場合に使用されるデフォルト VLAN を指定します。この VLAN ID は特定カスタマーに固有です。
ステップ 5	**switchport mode dot1q-tunnel** 例： `スイッチ(config-if)# switchport mode dot1q-tunnel`	IEEE 802.1Q トンネル ポートとしてインターフェイスを設定します。 （注） ポートを dynamic desirable デフォルト状態に戻すには、 **no switchport mode dot1q-tunnel** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 6	**exit** 例： `スイッチ(config-if)# exit`	特権 EXEC モードに戻ります。
ステップ 7	**vlan dot1q tag native** 例： `スイッチ(config)# vlan dot1q tag native`	（任意）すべての IEEE 802.1Q トランク ポートでネイティブ VLAN パケットのタギングがイネーブルになるようにdeviceを設定します。これを設定せず、カスタマー VLAN ID がネイティブ VLAN と同じである場合、トランク ポートはメトロ タグを適用せず、パケットは誤った宛先に送信される可能性があります。 （注） ネイティブ VLAN パケットのタグ付けをディセーブルにするには、 **no vlan dot1q tag native** グローバル コンフィギュレーション コマンドを使用します。
ステップ 8	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 9	次のいずれかを使用します。 • **show dot1q-tunnel** • **show running-config interface**	IEEE 802.1Q トンネリング用に設定されたポートを表示します。 トンネリング モードになっているポートを表示します。

	コマンドまたはアクション	目的
	例： スイッチ# **show dot1q-tunnel** または スイッチ# **show running-config interface**	
ステップ 10	**show vlan dot1q tag native** 例： スイッチ# **show vlan dot1q native**	IEEE 802.1Q ネイティブ VLAN タギング ステータスを表示します。
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

レイヤ 2 プロトコル トンネリングの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. 次のいずれかを使用します。
   - **switchport mode access**
   - **switchport mode dot1q-tunnel**
5. **l2protocol-tunnel**[**cdp** | **lldp** | **point-to-point** | **stp** | **vtp**]
6. **l2protocol-tunnel shutdown-threshold**[*packet_second_rate_value* | **cdp**|**lldp point-to-point** |**stp** | **vtp**]
7. **l2protocol-tunnel drop-threshold**[*packet_second_rate_value* | **cdp**|**lldp** | **point-to-point**|**stp** | **vtp**]
8. **exit**
9. **errdisable recovery cause l2ptguard**
10. **l2protocol-tunnel cos** *value*
11. **end**
12. **show l2protocol**
13. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	IP Phone に接続するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	次のいずれかを使用します。 • **switchport mode access** • **switchport mode dot1q-tunnel** 例： スイッチ# **switchport mode access** または スイッチ# **switchport mode dot1q-tunnel**	アクセス ポートまたは IEEE 802.1Q トンネル ポートとしてインターフェイスを設定します。
ステップ 5	**l2protocol-tunnel**[**cdp** | **lldp** | **point-to-point** | **stp** | **vtp**] 例： スイッチ# **l2protocol-tunnel cdp**	目的のプロトコルに対してプロトコル トンネリングをイネーブルにします。キーワードを入力しない場合、トンネリングは、3 つのすべてのレイヤ 2 プロトコルでイネーブルになります。 （注） いずれかのレイヤ 2 プロトコルまたは 3 つすべてのレイヤ 2 プロトコルのプロトコル トンネリングをディセーブルにするには、 **no l2protocol-tunnel** [**cdp** | **lldp** | **point-to-point** | **stp** | **vtp**] インターフェイス コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 6	**l2protocol-tunnel shutdown-threshold**[*packet_second_rate_value* | **cdp**|**lldp point-to-point** |**stp** | **vtp**] 例： スイッチ# **l2protocol-tunnel shutdown-threshold 100 cdp**	（任意）1 秒間にカプセル化可能なパケット数を表すしきい値を設定します。設定したしきい値を超えると、インターフェイスは無効になります。プロトコル オプションを指定しない場合、しきい値は、それぞれのトンネリングされたレイヤ 2 プロトコル タイプに適用されます。指定できる範囲は 1 ～ 4096 です。デフォルトでは、しきい値は設定されません。 （注） このインターフェイスでドロップしきい値も設定する場合は、**shutdown-threshold** 値を **drop-threshold** の値以上にする必要があります。 （注） **no l2protocol-tunnel shutdown-threshold** [*packet_second_rate_value* | **cdp** | **lldp**| **point-to-point** | **stp** | **vtp**] および **no l2protocol-tunnel drop-threshold** [*packet_second_rate_value* | **cdp** | **lldp**| **point-to-point** |**stp** | **vtp**] コマンドを使用し、シャットダウンとドロップのしきい値をデフォルト設定に戻します。
ステップ 7	**l2protocol-tunnel drop-threshold**[*packet_second_rate_value* | **cdp**|**lldp** | **point-to-point**|**stp** | **vtp**] 例： スイッチ# **l2protocol-tunnel drop-threshold 100 cdp**	（任意）1 秒間にカプセル化可能なパケット数を表すしきい値を設定します。設定したしきい値を超えると、インターフェイスによってパケットがドロップされます。プロトコル オプションを指定しない場合、しきい値は、それぞれのトンネリングされたレイヤ 2 プロトコル タイプに適用されます。指定できる範囲は 1 ～ 4096 です。デフォルトでは、しきい値は設定されません。 （注） このインターフェイスでシャットダウンしきい値も設定する場合は、**drop-threshold** 値を **shutdown-threshold** の値以上にする必要があります。 （注） **no l2protocol-tunnel shutdown-threshold** [**cdp** | | **lldppoint-to-pointstp** | **vtp**] および **no l2protocol-tunnel drop-threshold** [**cdp** | **stp** | **vtp**] コマンドを使用し、シャットダウンおよびドロップしきい値がデフォルト設定に戻ります。

	コマンドまたはアクション	目的
ステップ 8	**exit** 例： スイッチ# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 9	**errdisable recovery cause l2ptguard** 例： スイッチ(config)# **errdisable recovery cause l2ptguard**	（任意）インターフェイスが再び有効になって再試行できるように、レイヤ 2 最大レート エラーからの復旧メカニズムを設定します。errdisable recovery はデフォルトでディセーブルになっています。イネーブルにした場合、デフォルトの間隔は300秒です。
ステップ 10	**l2protocol-tunnel cos** *value* 例： スイッチ(config)# **l2protocol-tunnel cos value 7**	（任意）トンネリングされたすべてのレイヤ2PDUに対してCoS値を設定します。範囲は0～7です。デフォルトは、インターフェイスのデフォルトCoS値です。設定されていない場合、デフォルトは5です。
ステップ 11	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 12	**show l2protocol** 例： スイッチ# **show l2protocol**	deviceのレイヤ2トンネル ポートを表示します（設定されているプロトコル、しきい値、カウンタを含む）。
ステップ 13	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

サービスプロバイダー エッジ スイッチの設定

始める前に

EtherChannels の場合は、SP（サービス プロバイダ）エッジ devicesおよびカスタマー devicesをレイヤ 2 プロトコル トンネリング用に設定する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode dot1q-tunnel**
5. **l2protocol-tunnel point-to-point**[**pagp** |**lacp** |**udld**]
6. **l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value*
7. **l2protocol-tunnel drop-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value*
8. **no cdp enable**
9. **spanning-tree bpdu filter enable**
10. **exit**
11. **errdisable recovery cause l2ptguard**
12. **l2protocol-tunnel cos** *value*
13. **end**
14. **show l2protocol**
15. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/1`	IP Phone に接続するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport mode dot1q-tunnel** 例： `スイッチ(config-if)# switchport mode dot1q-tunnel`	IEEE 802.1Q トンネル ポートとしてインターフェイスを設定します。

	コマンドまたはアクション	目的
ステップ 5	**l2protocol-tunnel point-to-point**[**pagp** |**lacp** |**udld**] 例： `スイッチ(config-if)# l2protocol-tunnel point-to-point pagp`	（任意）目的のプロトコルに関するポイントツーポイント プロトコル トンネリングを有効にします。キーワードを入力しない場合、トンネリングは、3つすべてのプロトコルで有効になります。 （注） ネットワーク障害を避けるため、ネットワークがポイントツーポイント トポロジになっていることを確認してから、PAgPパケット、LACPパケット、UDLDパケットのうちいずれかのトンネリングをイネーブルにしてください。 （注） **no l2protocol-tunnel** [**point-to-point** [**pagp** | **lacp** | **udld**]] インターフェイス コンフィギュレーションを使用し、1 つまたは 3 つすべてのレイヤ 2 プロトコルのポイントツーポイントプロトコル トンネリングを無効にします。
ステップ 6	**l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value* 例： `スイッチ(config-if)# l2protocol-tunnel shutdown-threshold point-to-point pagp 100`	（任意）1秒間にカプセル化可能なパケット数を表すしきい値を設定します。設定したしきい値を超えると、インターフェイスは無効になります。プロトコル オプションを指定しない場合、しきい値は、それぞれのトンネリングされたレイヤ2プロトコルタイプに適用されます。指定できる範囲は1～4096です。デフォルトでは、しきい値は設定されません。 （注） このインターフェイスでドロップしきい値も設定する場合は、**shutdown-threshold** 値を **drop-threshold** の値以上にする必要があります。 （注） **no l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] および **no l2protocol-tunnel drop-threshold** [[**point-to-point** [**pagp** | lacp | **udld**]] コマンドを使用し、シャットダウンおよびドロップしきい値がデフォルト設定に戻ります。
ステップ 7	**l2protocol-tunnel drop-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value* 例： `スイッチ(config-if)# l2protocol-tunnel`	（任意）1秒間にカプセル化可能なパケット数を表すしきい値を設定します。設定したしきい値を超えると、インターフェイスによってパケットがドロップされます。プロトコル オプションを指定しない場合、しきい値は、それぞれのトンネリングされた

	コマンドまたはアクション	目的
	drop-threshold point-to-point pagp 500	レイヤ 2 プロトコル タイプに適用されます。指定できる範囲は 1 ～ 4096 です。デフォルトでは、しきい値は設定されません。 （注） このインターフェイスでシャットダウンしきい値も設定する場合は、**drop-threshold** 値を **shutdown-threshold** の値以上にする必要があります。
ステップ 8	**no cdp enable** 例： `スイッチ(config-if)# no cdp enable`	インターフェイス上で CDP を無効にします。
ステップ 9	**spanning-tree bpdu filter enable** 例： `スイッチ(config-if)# spanning-tree bpdu filter enable`	インターフェイス上で BPDU フィルタリングをイネーブルにします。
ステップ 10	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 11	**errdisable recovery cause l2ptguard** 例： `スイッチ(config)# errdisable recovery cause l2ptguard`	（任意）インターフェイスが再び有効になって再試行できるように、レイヤ 2 最大レート エラーからの復旧メカニズムを設定します。errdisable recovery はデフォルトでディセーブルになっています。イネーブルにした場合、デフォルトの間隔は 300 秒です。
ステップ 12	**l2protocol-tunnel cos** *value* 例： `スイッチ(config)# l2protocol-tunnel cos 2`	（任意）トンネリングされたすべてのレイヤ 2 PDU に対して CoS 値を設定します。範囲は 0 ～ 7 です。デフォルトは、インターフェイスのデフォルト CoS 値です。設定されていない場合、デフォルトは 5 です。
ステップ 13	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 14	**show l2protocol** 例： スイッチ) # **show l2protocol**	deviceのレイヤ2 トンネル ポートを表示します（設定されているプロトコル、しきい値、カウンタを含む）。
ステップ 15	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

カスタマー デバイスの設定

始める前に

EtherChannel の場合は、サービスプロバイダ エッジ deviceおよびカスタマー devicesをレイヤ 2 プロトコル トンネリング用に設定する必要があります

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport trunk encapsulation dot1q**
5. **switchport mode trunk**
6. **udld port**
7. **channel-group** *channel-group-number* **mode desirable**
8. **exit**
9. **interface port-channel port-channel** *number*
10. **shutdown**
11. **no shutdown**
12. **end**
13. **show l2protocol**
14. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	IP Phone に接続するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport trunk encapsulation dot1q** 例： スイッチ(config)# **switchport trunk encapsulation dot1q**	トランキング カプセル化形式を IEEE 802.1Q に設定します。
ステップ 5	**switchport mode trunk** 例： スイッチ(config-if)# **switchport mode trunk**	インターフェイスでトランキングをイネーブルにします。
ステップ 6	**udld port** 例： スイッチ(config-if)# **udld port**	インターフェイス上で UDLD を通常モードでイネーブルにします。
ステップ 7	**channel-group** *channel-group-number* **mode desirable** 例： スイッチ(config-if)# **channel-group 25 mode desirable**	チャネル グループにインターフェイスを割り当て、PAgP モードに desirable を指定します。
ステップ 8	**exit** 例： スイッチ(config-if)# **exit**	グローバル コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
ステップ 9	**interface port-channel port-channel** *number* 例： スイッチ(config)# **interface port-channel port-channel 25**	ポートチャネル インターフェイス モードを開始します。
ステップ 10	**shutdown** 例： スイッチ(config)# **shutdown**	インターフェイスをシャットダウンします。
ステップ 11	**no shutdown** 例： スイッチ(config)# **no shutdown**	インターフェイスを有効にします。
ステップ 12	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 13	**show l2protocol** 例： スイッチ# **show l2protocol**	deviceのレイヤ2 トンネル ポートを表示します（設定されているプロトコル、しきい値、カウンタを含む）。
ステップ 14	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。 （注） インターフェイスをデフォルト設定に戻すには、 **no switchport mode trunk**、**no udld enable**、および **no channel group** *channel-group-number* **mode desirable** インターフェイス コンフィギュレーション コマンドを使用します。

IEEE 802.1Q およびレイヤ 2 プロトコル トンネリングの設定例

例：IEEE 802.1Q トンネリング ポートの設定

以下の例では、トンネル ポートとしてインターフェイスを設定してネイティブ VLAN パケットのタグ付けをイネーブルにし、設定を確認する方法を示します。

```
Switch(config)# interface gigabitethernet1/0/7
Switch(config-if)# switchport access vlan 22
% Access VLAN does not exist. Creating vlan 22
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# exit
Switch(config)# vlan dot1q tag native
Switch(config)# end
Switch# show dot1q-tunnel interface gigabitethernet1/0/7
Port
-----
Gi1/0/1Port
-----
Switch# show vlan dot1q tag native
dot1q native vlan tagging is enabled
```

例：レイヤ 2 プロトコル トンネリングの設定

以下の例では、CDP、STP、VTP のレイヤ 2 プロトコル トンネリングを設定し、設定を確認する方法を示します。

```
Switch(config)# interface gigabitethernet1/0/11
Switch(config-if)# l2protocol-tunnel cdp
Switch(config-if)# l2protocol-tunnel stp
Switch(config-if)# l2protocol-tunnel vtp
Switch(config-if)# l2protocol-tunnel shutdown-threshold 1500
Switch(config-if)# l2protocol-tunnel drop-threshold 1000
Switch(config-if)# exit
Switch(config)# l2protocol-tunnel cos 7
Switch(config)# end
Switch# show l2protocol

COS for Encapsulated Packets: 7
Port Protocol Shutdown Drop Encapsulation Decapsulation Drop
Threshold Threshold Counter Counter Counter
------- -------- --------- --------- ------------- ------------- -------------
Gi0/11 cdp 1500 1000 2288 2282 0
stp 1500 1000 116 13 0
vtp 1500 1000 3 67 0
pagp ---- ---- 0 0 0
lacp ---- ---- 0 0 0
udld ---- ---- 0 0 0
```

例：サービスプロバイダー エッジ スイッチとカスタマー スイッチの設定

以下は、サービス プロバイダーのエッジ スイッチ 1 およびエッジ スイッチ 2 を設定する方法の例です。VLAN 17、18、19、20 はアクセス VLAN、ファスト イーサネット インターフェイス 1 および 2 は PAgP および UDLD がイネーブルになっているポイントツーポイント トンネル ポート、ドロップしきい値は 1000、ファスト イーサネット インターフェイス 3 はトランク ポートです。

サービスプロバイダー エッジ スイッチ 1 の設定は次のとおりです。

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport access vlan 17
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport access vlan 18
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)#
Switch(config-if)# switchport mode trunk
```

サービスプロバイダー エッジ スイッチ 2 の設定は次のとおりです。

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport access vlan 19
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport access vlan 20
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)#
Switch(config-if)# switchport mode trunk
```

次は、サイト 1 のカスタマー スイッチを設定する方法の例です。ファスト イーサネット インターフェイス 1、2、3、4 は IEEE 802.1Q トランキング用に設定されており、UDLD はイネーブル、EtherChannel グループ 1 はイネーブル、ポート チャネルはシャットダウンされた後でイネーブルになり EtherChannel 設定がアクティブになります。

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface port-channel 1
Switch(config-if)# shutdown
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

トンネリング ステータスのモニタリング

次の表では、トンネリング ステータスをモニタするために使用するコマンドについて説明します。

表 *40 :* トンネリングのモニタリング コマンド

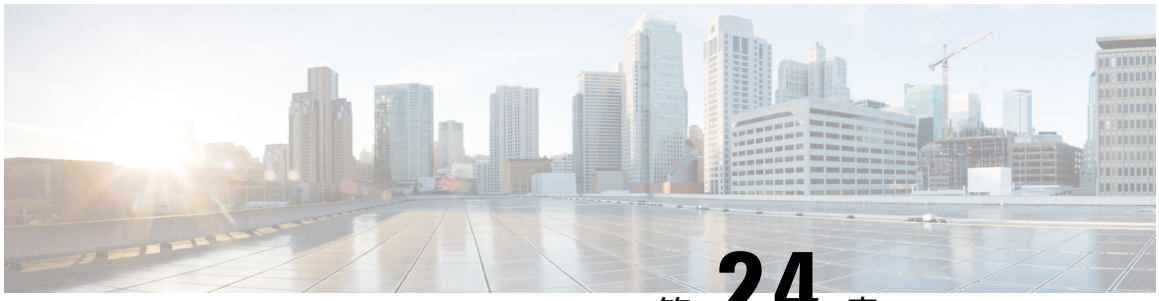
コマンド	目的
show dot1q-tunnel	device の IEEE 802.1Q トンネルポートを表示します。
show dot1q-tunnel interface *interface-id*	特定のインターフェイスがトンネル ポートであるかどうかを確認します。
show vlan dot1q tag native	deviceのネイティブ VLAN タギングのステータスを表示します。

次の作業

次の設定を行えます。

- VTP
- VLAN
- VLAN トランキング
- プライベート VLAN
- VLAN メンバーシップ ポリシー サーバー（VMPS）
- 音声 VLAN

次の作業



第 24 章

スパニングツリー プロトコルの設定

この章では、Catalyst devicesのポートベース VLAN（仮想 LAN）上でスパニングツリープロトコル（STP）を設定する方法について説明します。このdeviceは、IEEE 802.1D 標準に準拠した Per-VLAN Spanning-Tree plus（PVST+）とシスコ独自の拡張機能の組み合わせか、もしくは IEEE 802.1w 標準に準拠した Rapid Per-VLAN Spanning-Tree plus（Rapid PVST+）プロトコルのいずれかを使用できます。スイッチ スタックは、ネットワークのその他の部分に対しては単一のスパニングツリー ノードに見え、すべてのスタック メンバが同一のブリッジ ID を使用します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

STP の制約事項

- ルート deviceとしてdeviceを設定しようとする場合、ルート deviceにするために必要な値が 1 未満だと、失敗します。

- ネットワークが、拡張システム ID をサポートするdevicesとサポートしないものの両方で構成されている場合、拡張システム ID をサポートするdeviceがルート deviceになる可能性は低くなります。古いソフトウェアを実行している接続devicesの優先度より VLAN 番号が大きい場合は常に、拡張システム ID によってdevice 優先度の値が増加します。
- 各スパニングツリーインスタンスのルートdeviceは、バックボーンまたはディストリビューション deviceでなければなりません。アクセス deviceをスパニングツリープライマリ ルートとして設定しないでください。

スパニング ツリー プロトコルに関する情報

スパニングツリー プロトコル

スパニングツリープロトコル（STP）は、ネットワーク内のループを回避しながらパスを冗長化するためのレイヤ2リンク管理プロトコルです。レイヤ2イーサネット ネットワークが正常に動作するには、任意の2つのステーション間で存在できるアクティブ パスは1つだけです。エンドステーション間に複数のアクティブパスがあると、ネットワークにループが生じます。ネットワークにループが存在すると、エンドステーションがメッセージを重複して受信する可能性があります。また、Devices が複数のレイヤ 2 インターフェイス上のエンドステーション MAC アドレスを学習する可能性もあります。このような状況によって、ネットワークが不安定になります。スパニングツリーの動作は透過的であり、エンドステーション側で、単一LAN セグメントに接続されているのか、複数セグメントからなるスイッチド LAN に接続されているのかを検出することはできません。

STP は、スパニングツリーアルゴリズムを使用し、スパニングツリーのルートとして冗長接続ネットワーク内のdeviceを 1 つ選択します。アルゴリズムは、次に基づき、各ポートに役割を割り当て、スイッチドレイヤ2ネットワークを介して最良のループフリーパスを算出します。アクティブ トポロジでのポートの役割：

- ルート：スパニングツリー トポロジに対して選定される転送ポート
- 指定：各スイッチド LAN セグメントに対して選定される転送ポート
- 代替：スパニングツリーのルート ブリッジへの代替パスとなるブロック ポート
- バックアップ：ループバック コンフィギュレーションのブロック ポート

すべてのポートに役割が指定されているdevice、またはバックアップの役割が指定されているスイッチはルート deviceです。 少なくとも *1* つのポートに役割が指定されているdeviceは、指定deviceを意味します。

冗長データ パスはスパニングツリーによって、強制的にスタンバイ（ブロックされた）ステートにされます。スパニングツリーのネットワークセグメントでエラーが発生したときに冗長パスが存在する場合は、スパニングツリー アルゴリズムがスパニングツリートポロジを再計算し、スタンバイパスをアクティブにします。Devices は次のように呼ばれるスパニングツリーフレームを送受信します。 （ブリッジ プロトコル データ ユニット（BPDU）と呼ばれる）を

定期間隔で送受信します。devicesはこのフレームを転送しませんが、このフレームを使用してループフリー パスを構築します。 BPDU には、deviceおよび MAC アドレス、deviceの優先順位、ポートの優先順位、およびパス コストを含む、送信側deviceとそのポートに関する情報が含まれます。スパニングツリーはこの情報を使用して、スイッチド ネットワーク用のルート deviceおよびルート ポートを選定し、さらに、各スイッチド セグメントのルート ポートおよび指定ポートを選定します。

deviceの 2 つのポートがループの一部である場合、 spanning-tree および、パス コスト設定は、どのポートがフォワーディング ステートになるか、およびどのポートがブロッキング ステートになるかを制御します。スパニングツリー ポート プライオリティ値は、ネットワーク トポロジにおけるポートの位置とともに、トラフィック転送におけるポートの位置がどれだけ適切であるかを表します。The コスト値は、メディア速度を表します。

（注） デフォルトではdeviceは、Small Form-Factor Pluggable（SFP）モジュールを備えていないインターフェイスにだけ、（接続が稼働していることを確認するために）キープアライブ メッセージを送信します。[**no**] **keepalive** インターフェイス コンフィギュレーション コマンドをキーワードなしで入力すると、インターフェイスのデフォルトを変更できます。

スパニングツリー トポロジと BPDU

スイッチド ネットワーク内の安定したアクティブ スパニングツリー トポロジは、次の要素によって制御されます。

- device上の各 VLAN に関連付けられた一意のブリッジ ID（device優先度および MAC アドレス）。
- ルート deviceに対するスパニングツリー パス コスト。
- 各レイヤ 2 インターフェイスに対応付けられたポート ID（ポート プライオリティおよび MAC アドレス）。

ネットワーク内のdevicesに電源が入ると、各機能はルート deviceとして機能します。各deviceは、そのすべてのポートからコンフィギュレーション BPDU を送信します。BPDU によって通信が行われ、スパニングツリー トポロジが計算されます。各設定 BPDU には、次の情報が含まれています。

- 送信deviceがルート deviceとして識別するdeviceの一意のブリッジ ID。
- ルートまでのスパニングツリー パス コスト
- 送信deviceのブリッジ ID。
- メッセージ エージ
- 送信側インターフェイス ID
- hello タイマー、転送遅延タイマー、および max-age プロトコル タイマーの値

deviceは、優位な情報（より小さいブリッジID、より低いパス コストなど）が含まれているコンフィギュレーションBPDUを受信すると、そのポートに対する情報を保存します。このBPDUをdeviceのルート ポート上で受信した場合、そのdeviceが指定deviceとなっているすべての接続 LAN に、更新したメッセージを付けて BPDU を転送します。

deviceは、そのポートに現在保存されている情報よりも下位の情報を含むコンフィギュレーション BPDU を受信した場合は、その BPDU を廃棄します。deviceが下位 BPDU を受信した LAN の指定deviceである場合、そのポートに保存されている最新情報を含む BPDU をその LAN に送信します。このようにして下位情報は廃棄され、優位情報がネットワークで伝播されます。

BPDU の交換によって、次の処理が行われます。

- ネットワーク内の 1 つのdeviceがとして選択されます。 ルート device（スイッチド ネットワークのスパニングツリー トポロジーの論理的な中心）。箇条書きの項目の下の図を参照してください。

  VLAN ごとに、device優先度が最も高い（最も小さい数字の優先順位の値）deviceがルート deviceとして選択されます。すべてのdevicesがデフォルトの優先度（32768）で設定されている場合、VLAN 内で MAC アドレスの最も小さいdeviceがルート deviceになります。device の優先順位の値は、ブリッジ ID の最上位ビットを占めます。

- deviceごとに（ルート deviceを除く）、ルート ポートが 1 つ選択されます。このポートは、deviceからルート deviceにパケットを転送するときに最適パス（最小コスト）を提供します。
- ルート deviceへの最短距離は、パス コストに基づいてdeviceごとに計算されます。
- LAN セグメントごとに指定deviceが選択されます。指定deviceは、その LAN からルート deviceにパケットを転送するときの最小パス コストを提供します。DP は、指定deviceが LAN に接続されているポートです。

（注） **logging event spanning tree**コマンドが複数のインターフェイスに設定され、トポロジが変更されると、複数のロギングメッセージが発生し、CPU 使用率が高くなることがあります。これにより、スイッチが STP Bpdu の処理をドロップまたは遅延させる可能性があります。

この動作を防ぐには、**logging event spanning tree** および **logging event status** コマンドを削除するか、コンソールへのロギングを無効にします。

スイッチドネットワーク上のいずれの地点からもルート deviceに到達する場合に必要のないパスはすべて、スパニングツリー ブロッキング モードになります。

ブリッジ ID、デバイス プライオリティ、および拡張システム ID

IEEE 802.1D 標準では、それぞれのdeviceに固有のルートdeviceの選択を制御するブリッジ識別子（ブリッジ ID）が必要です。各 VLAN は PVST+ と Rapid PVST+ によって異なる論理ブリッジと見なされるので、同一のdeviceは設定された各 VLAN とは異なるブリッジ ID を保有している必要があります。device上の各 VLAN には一意の 8 バイト ブリッジ ID が設定されます。

上位の 2 バイトはdeviceプライオリティに使用され、残りの 6 バイトがdeviceの MAC アドレスから取得されます。

deviceでは IEEE 802.1t スパニングツリー拡張機能がサポートされ、従来はdevice プライオリティに使用されていたビットの一部が VLAN ID として使用されるようになりました。その結果、deviceに割り当てられる MAC アドレスが少なくなり、より広い範囲の VLAN ID をサポートできるようになり、しかもブリッジ ID の一意性を損なうこともありません。

従来はdevice プライオリティに使用されていた 2 バイトが、4 ビットのプライオリティ値と 12 ビットの拡張システム ID 値（VLAN ID と同じ）に割り当てられています。

表 *41* : デバイス プライオリティ値および拡張システム *ID*

プライオリティ値				拡張システム ID（VLAN ID と同設定）											
ビット 16	ビット 15	ビット 14	ビット 13	ビット 12	ビット 11	ビット 10	ビット 9	ビット 8	ビット 7	ビット 6	ビット 5	ビット 4	ビット 3	ビット 2	ビット 1
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

スパニングツリーは、ブリッジ ID を VLAN ごとに一意にするために、拡張システム ID、device プライオリティ、および割り当てられたスパニングツリー MAC アドレスを使用します。

拡張システム ID のサポートにより、ルート device、セカンダリ ルート device、および VLAN のdevice プライオリティの手動での設定方法に影響が生じます。たとえば、deviceのプライオリティ値を変更すると、deviceがルート deviceとして選定される可能性も変更されることになります。大きい値を設定すると可能性が低下し、値が小さいと可能性が増大します。

指定された VLAN のルート deviceに 24576 に満たないdevice プライオリティが設定されている場合は、deviceはその VLAN について、自身のプライオリティを最小のdevice プライオリティより 4096 だけ小さい値に設定します4096 は、表に示すように 4 ビット device スイッチ プライオリティ値の最下位ビットの値です。

ポート プライオリティとパス コスト

ループが発生した場合、スパニングツリーはポート プライオリティを使用して、フォワーディング ステートにするインターフェイスを選択します。最初に選択されるインターフェイスには高いプライオリティ値（小さい数値）を割り当て、最後に選択されるインターフェイスには低いプライオリティ値（高い数値）を割り当てることができます。すべてのインターフェイスに同じプライオリティ値が与えられている場合、スパニングツリーはインターフェイス番号が最小のインターフェイスをフォワーディング ステートにし、他のインターフェイスをブロックします。

スパニングツリー パス コストのデフォルト値は、インターフェイスのメディア速度に基づきます。ループが発生した場合、スパニングツリーはコストを使用して、フォワーディング ステートにするインターフェイスを選択します。最初に選択されるインターフェイスには低いコスト値を割り当て、最後に選択されるインターフェイスには高いコスト値を割り当てることができます。すべてのインターフェイスに同じコスト値が与えられている場合、スパニングツ

リーはインターフェイス番号が最小のインターフェイスをフォワーディング ステートにし、他のインターフェイスをブロックします。

deviceがdevice スタックのメンバーの場合は、最初に選択させたいインターフェイスには小さいコスト値を与え、最後に選択させたいインターフェイスには（ポートプライオリティを調整せずに）大きいコスト値を与えます。詳細については、関連項目を参照してください。

スパニングツリー インターフェイス ステート

プロトコル情報がスイッチド LAN を通過するとき、伝播遅延が生じることがあります。その結果、スイッチド ネットワークのさまざまな時点および場所でトポロジーの変化が発生します。インターフェイスがスパニングツリートポロジに含まれていない状態からフォワーディング ステートに直接移行すると、一時的にデータ ループが形成されることがあります。インターフェイスは新しいトポロジ情報がスイッチド LAN 上で伝播されるまで待機し、フレーム転送を開始する必要があります。インターフェイスはさらに、古いトポロジで使用されていた転送フレームのフレーム存続時間を満了させることも必要です。

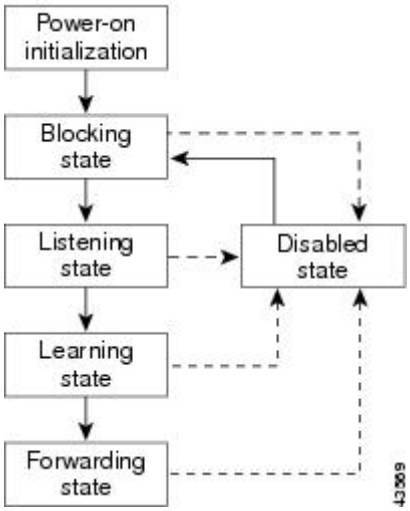
スパニングツリーを使用しているdeviceの各レイヤ 2 インターフェイスは、次のいずれかのステートになります。

- ブロッキング：インターフェイスはフレーム転送に関与しません。
- リスニング：インターフェイスをフレーム転送に関与させることをスパニングツリーが決定した場合、ブロッキング ステートから最初に移行するステートです。
- ラーニング：インターフェイスはフレーム転送に関与する準備をしている状態です。
- フォワーディング：インターフェイスはフレームを転送します。
- ディセーブル：インターフェイスはスパニングツリーに含まれません。シャットダウン ポートであるか、ポート上にリンクがないか、またはポート上でスパニングツリー インスタンスが稼働していないためです。

インターフェイスは次のように、ステートを移行します。

- 初期化からブロッキング
- ブロッキングからリスニングまたはディセーブル
- リスニングからラーニングまたはディセーブル
- ラーニングからフォワーディングまたはディセーブル
- フォワーディングからディセーブル

図 *21* : スパニングツリー インターフェイス ステート



インターフェイスはこれらのステート間を移動します。

デフォルト設定では、deviceを起動するとスパニングツリーがイネーブルになります。その後、deviceの各インターフェイス、VLAN、ネットワークがブロッキング ステートからリスニングおよびラーニングという移行ステートを通過します。スパニングツリーは、フォワーディング ステートまたはブロッキング ステートで各インターフェイスを安定させます。

スパニングツリー アルゴリズムがレイヤ 2 インターフェイスをフォワーディング ステートにする場合、次のプロセスが発生します。

1. スパニングツリーがインターフェイスをブロッキング ステートに移行させるプロトコル情報を待つ間、インターフェイスはリスニング ステートになります。
2. スパニングツリーは転送遅延タイマーの満了を待ち、インターフェイスをラーニング ステートに移行させ、転送遅延タイマーをリセットします。
3. ラーニング ステートの間、deviceが転送データベースのエンド ステーションの位置情報を学習しているとき、インターフェイスはフレーム転送をブロックし続けます。
4. 転送遅延タイマーが満了すると、スパニングツリーはインターフェイスをフォワーディング ステートに移行させ、このときラーニングとフレーム転送の両方が可能になります。

ブロッキング ステート

ブロッキング ステートのレイヤ2インターフェイスはフレームの転送に関与しません。初期化後、deviceの各インターフェイスにBPDUが送信されます。deviceは最初、他のdevicesとBPDUを交換するまで、ルートとして動作します。この交換により、ネットワーク内でどのdeviceがルートまたはルート deviceになるかが確立されます。ネットワーク内にdeviceが 1 つしかない場合は交換は行われず、転送遅延タイマーが満了し、インターフェイスがリスニング ステートになります。インターフェイスはdeviceの初期化後、必ずブロッキング ステートになります。

ブロッキング ステートのインターフェイスは、次の機能を実行します。

- インターフェイス上で受信したフレームを廃棄します。

- 転送用に他のインターフェイスからスイッチングされたフレームを廃棄します。
- アドレスを学習しません。
- BPDU を受信します。

リスニング ステート

リスニング ステートは、ブロッキング ステートを経て、レイヤ 2 インターフェイスが最初に移行するステートです。インターフェイスがリスニング ステートになるのは、スパニングツリーによってそのインターフェイスのフレーム転送への関与が決定された場合です。

リスニング ステートのインターフェイスは、次の機能を実行します。

- インターフェイス上で受信したフレームを廃棄します。
- 転送用に他のインターフェイスからスイッチングされたフレームを廃棄します。
- アドレスを学習しません。
- BPDU を受信します。

ラーニング ステート

ラーニング ステートのレイヤ2インターフェイスは、フレームの転送に関与できるように準備します。インターフェイスはリスニング ステートからラーニング ステートに移行します。

ラーニング ステートのインターフェイスは、次の機能を実行します。

- インターフェイス上で受信したフレームを廃棄します。
- 転送用に他のインターフェイスからスイッチングされたフレームを廃棄します。
- アドレスを学習します。
- BPDU を受信します。

フォワーディング ステート

フォワーディング ステートのレイヤ2インターフェイスは、フレームを転送します。インターフェイスはラーニング ステートからフォワーディング ステートに移行します。

フォワーディング ステートのインターフェイスは、次の機能を実行します。

- インターフェイス上でフレームを受信して転送します。
- 他のインターフェイスからスイッチングされたフレームを転送します。
- アドレスを学習します。
- BPDU を受信します。

ディセーブル ステート

ブロッキング ステートのレイヤ2インターフェイスは、フレームの転送やスパニングツリーに関与しません。ディセーブル ステートのインターフェイスは動作不能です。

ディセーブル インターフェイスは、次の機能を実行します。

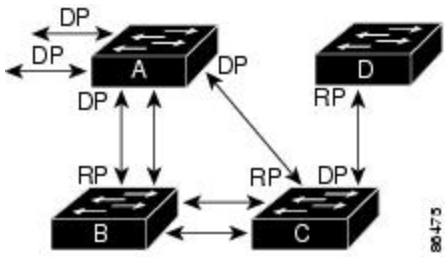
- インターフェイス上で受信したフレームを廃棄します。
- 転送用に他のインターフェイスからスイッチングされたフレームを廃棄します。
- アドレスを学習しません。
- BPDU を受信しません。

デバイス またはポートがルート デバイスまたはルート ポートになる仕組み

ネットワーク上のすべてのdevicesがデフォルトのスパニングツリー設定で有効になっている場合、最小の MAC アドレスを持つdeviceがルート deviceになります。

図 *22 :* スパニングツリー トポロジ

デバイス A はルート deviceとして選択されます。すべてのdevicesのdeviceの優先度がデフォルト（32768）に設定されており、デバイス A の MAC アドレスが最も小さいためです。ただし、トラフィック パターン、転送インターフェイスの数、またはリンク タイプによっては、デバイス A が最適なルート deviceとは限りません。ルート deviceになるように、最適なdeviceのプライオリティを引き上げる（数値を引き下げる）と、スパニングツリーの再計算が強制的に行われ、最適なdeviceをルートとした新しいトポロジが形成されます。



RP = Root Port
DP = Designated Port

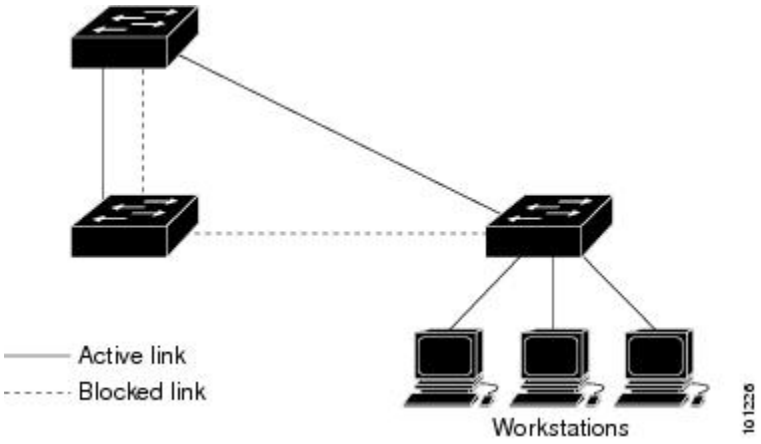
スパニングツリー トポロジがデフォルトのパラメータに基づいて算出された場合、スイッチド ネットワークの送信元エンド ステーションから宛先エンド ステーションまでのパスが最適にならない場合があります。たとえば、ルート ポートよりプライオリティの高いインターフェイスに高速リンクを接続すると、ルート ポートが変更される可能性があります。最高速のリンクをルート ポートにすることが重要です。

たとえば、デバイス B のあるポートがギガビット イーサネット リンクで、デバイス上の別のポート（10/100 リンク）がルート ポートであると仮定します。ネットワーク トラフィックはギガビット イーサネット リンクに流す方が効率的です。ギガビット イーサネット ポートのスパニングツリー ポート プライオリティをルート ポートより高くする（数値を小さくする）と、ギガビット イーサネット ポートが新しいルート ポートになります。

スパニングツリーおよび冗長接続

図 *23 :* スパニングツリーおよび冗長接続

2 つのdevice インターフェイスを別の 1 台のデバイス、または 2 台の異なるデバイスに接続することにより、スパニングツリーを使用して冗長バックボーンを作成できます。スパニングツリーは一方のインターフェイスを自動的にディセーブルにし、他方でエラーが発生した場合にはそのディセーブルにしていた方をイネーブルにします。一方のリンクが高速で、他方が低速の場合、必ず、低速の方のリンクがディセーブルになります。速度が同じ場合、ポート優先度とポート ID が加算され、最大値を持つリンクがスパニングツリーによって無効にされます。



EtherChannel グループを使用して、devices間に冗長リンクを設定することもできます。

スパニングツリー アドレスの管理

IEEE 802.1D では、各種ブリッジ プロトコルに使用させるために、0x00180C2000000 ～ 0x0180C2000010 の範囲で 17 のマルチキャスト アドレスが規定されています。これらのアドレスは削除できないスタティック アドレスです。

スパニングツリー ステートに関係なく、スタック内の各deviceは 0x0180C2000000 ～ 0x0180C2000000 のアドレス宛てのパケットを受信しますが、転送は行いません。

スパニングツリーが有効になっている場合、device またはスタック 内の各 device の CPU は 0x0180C2000000 および 0x0180C2000010 宛てのパケットを受信します。スパニングツリーが無効になっている場合は、device またはスタック 内の各 device は、それらのパケットを不明のマルチキャストアドレスとして転送します。

接続を維持するためのエージング タイムの短縮

ダイナミックアドレスのエージングタイムはデフォルトで 5 分です。これは、**mac address-table aging-time** グローバル コンフィギュレーション コマンドのデフォルトの設定です。ただし、スパニングツリーの再構成により、多数のステーションの位置が変更されることがあります。このようなステーションは、再構成中、5 分以上にわたって到達できないことがあるので、アドレステーブルからステーションアドレスを削除し、改めて学習できるように、アドレスエージング タイムが短縮されます。スパニングツリー再構成時に短縮されるエージングタイムは、

転送遅延パラメータ値（**spanning-tree vlan** *vlan-id* **forward-time** *seconds* グローバル コンフィギュレーション コマンド）と同じです。

各 VLAN はそれぞれ独立したスパニングツリー インスタンスであるため、deviceは VLAN 単位でエージング タイムを短縮します。あるVLANでスパニングツリーの再構成が行われると、その VLAN で学習されたダイナミック アドレスがエージング タイム短縮の対象になります。他のVLANのダイナミック アドレスは影響を受けず、deviceで設定されたエージング間隔がそのまま保持されます。

スパニングツリー モードおよびプロトコル

このdeviceでサポートされるモードおよびプロトコルは、次のとおりです。

- PVST+：このスパニングツリー モードは、IEEE 802.1D 標準およびシスコ独自の拡張機能に準拠します。PVST+ はdevice上の各 VLAN でサポートされる最大数まで動作し、各 VLAN にネットワーク上でのループフリーパスを提供します。

  PVST+ は、対象となる VLAN にレイヤ 2 ロード バランシングを提供します。ネットワーク上の VLAN を使用してさまざまな論理トポロジを作成し、特定のリンクに偏らないようにすべてのリンクを使用できるようにします。VLAN 上のPVST+インスタンスごとに、それぞれ 1 つのルート deviceがあります。このルート deviceは、その VLAN に対応するスパニングツリー情報を、ネットワーク上の他のすべてのdevicesに伝送します。このプロセスにより、各deviceがネットワークに関する共通の情報を持つため、ネットワーク トポロジが確実に維持されます。

- Rapid PVST+：このスパニングツリーモードは、IEEE 802.1w 標準に準拠した高速コンバージェンスを使用する以外はPVST+ と同じです。15.2(4)E リリース以降、STP のデフォルト モードは Rapid PVST+ です。高速コンバージェンスを行うため、Rapid PVST+ はトポロジ変更を受信すると、ポート単位でダイナミックに学習した MAC アドレス エントリをただちに削除します。このような場合、PVST+ では、ダイナミックに学習した MAC アドレス エントリには短いエージング タイムが使用されます。

  Rapid PVST+ はPVST+ と同じ設定を使用しているので（特に明記する場合を除く）、device で必要なことは最小限の追加設定のみです。Rapid PVST+ の利点は、大規模な PVST+ のインストール ベースを Rapid PVST+ に移行する際に、複雑なマルチ スパニングツリー プロトコル（MSTP）設定の学習やネットワーク再設定の必要がないことです。Rapid PVST+ モードでは、各 VLAN は独自のスパニングツリー インスタンスを最大数実行します。

- MSTP：このスパニングツリーモードはIEEE 802.1s 標準に準拠しています。複数の VLAN を同一のスパニングツリー インスタンスにマッピングし、多数の VLAN をサポートする場合に必要となるスパニングツリーインスタンスの数を減らすことができます。MSTP は Rapid Spanning-Tree Protocol（RSTP）（IEEE 802.1w 準拠）上で実行され、転送遅延を解消し、ルート ポートおよび指定ポートをフォワーディング ステートにすばやく移行することにより、スパニングツリーの高速コンバージェンスを可能にします。deviceスタックでは、クロススタック高速移行（CSRT）機能が RSTP と同じ機能を実行します。RSTP または CSRT を使用しなければ、MSTP は稼働できません。

サポートされるスパニングツリー インスタンス

PVST+ または Rapid PVST+ モードでは、deviceまたはdeviceスタックは最大 128 のスパニングツリー インスタンスをサポートします。

MSTP モードでは、deviceまたはdeviceスタックは最大 65 の MST インスタンスをサポートします。特定の MST インスタンスにマッピング可能な VLAN 数に制限はありません。

スパニングツリーの相互運用性と下位互換性

MSTP および PVST+ が混在したネットワークでは、Common Spanning-Tree（CST）のルートは MST バックボーンの内側に配置する必要があり、PVST+ deviceを複数の MST リージョンに接続することはできません。

ネットワーク内に Rapid PVST+ を実行しているdevicesと PVST+ を実行しているdevicesが存在する場合、Rapid PVST+ devicesと PVST+ devicesを別のスパニングツリー インスタンスに設定することを推奨します。Rapid PVST+ スパニングツリー インスタンスでは、ルート deviceは Rapid PVST+ deviceでなければなりません。PVST+ インスタンスでは、ルート deviceは PVST+ deviceでなければなりません。PVST+ devicesはネットワークのエッジに配置する必要があります。

すべてのスタック メンバーが、同じバージョンのスパニングツリーを実行します（すべて PVST+、すべて Rapid PVST+、またはすべて MSTP）。

表 *42 : PVST+*、*MSTP*、*Rapid PVST+* の相互運用性と互換性

	PVST+	**MSTP**	**Rapid PVST+**
PVST+	あり	あり（制限あり）	あり（PVST+に戻る）
MSTP	あり（制限あり）	あり	あり（PVST+に戻る）
Rapid PVST+	あり（PVST+に戻る）	あり（PVST+に戻る）	対応

STP および IEEE 802.1Q トランク

VLAN トランクに関する IEEE 802.1Q 規格は、ネットワークのスパニングツリー ストラテジに一定の制限を設けています。この規格では、トランク上で使用できるすべての VLAN に対して、1 つのスパニングツリー インスタンスしか認められません。ただし、IEEE 802.1Q トランクを介して接続される Cisco devicesのネットワークにおいて、devicesはトランク上で許容される VLAN ごとに 1 つのスパニングツリー インスタンスを維持します。

IEEE 802.1Q トランクを介して Cisco deviceを他社製のデバイスに接続する場合、Cisco deviceは PVST+ を使用してスパニングツリーの相互運用性を実現します。Rapid PVST+ がイネーブルの場合、device は PVST+ ではなく Rapid PVST+ を使用します。deviceは、トランクの IEEE 802.1Q VLAN のスパニングツリー インスタンスと他社の IEEE 802.1Q deviceのスパニングツリー インスタンスを結合します。

ただし、PVST+ または Rapid PVST+ の情報はすべて、他社製の IEEE 802.1Q devicesからなるクラウドにより分離された Cisco devicesによって維持されます。Cisco devicesを分離する他社製の IEEE 802.1Q クラウドは、devices間の単一トランク リンクとして扱われます。

PVST+ は IEEE 802.1Q トランクで自動的に有効になるので、ユーザー側で設定する必要はありません。アクセスポートおよび ISL（スイッチ間リンク）トランクポートでの外部スパニングツリーの動作は、PVST+ の影響を受けません。

VLAN ブリッジ スパニングツリー

シスコ VLAN ブリッジ スパニングツリーは、フォールバック ブリッジング機能（ブリッジ グループ）で使用し、DECnet などの IP 以外のプロトコルを 2 つ以上の VLAN ブリッジ ドメインまたはルーテッドポート間で伝送します。VLAN ブリッジ スパニングツリーにより、ブリッジ グループは個々の VLAN スパニングツリーの上部にスパニングツリーを形成できるので、VLAN 間で複数の接続がある場合に、ループが形成されないようにします。また、ブリッジングされている VLAN からの個々のスパニングツリーが単一のスパニングツリーに縮小しないようにする働きもします。

VLAN ブリッジ スパニングツリーをサポートするには、一部のスパニングツリー タイマーを増やします。フォールバック ブリッジング機能を使用するには、deviceで IP サービス フィーチャ セットをイネーブルにする必要があります。

スパニングツリー機能のデフォルト設定

表 *43 :* スパニングツリー機能のデフォルト設定

機能	デフォルト設定
イネーブル ステート	VLAN 1 上でイネーブル
スパニングツリー モード	Rapid PVST+（PVST+ と MSTP はディセーブル）
デバイス priority	32768
スパニングツリー ポート プライオリティ（インターフェイス単位で設定可能）	128
スパニングツリー ポート コスト（インターフェイス単位で設定可能）	1000 Mb/s：4 100 Mb/s：19 10 Mb/s：100
スパニングツリー VLAN ポート プライオリティ（VLAN 単位で設定可能）	128

機能	デフォルト設定
スパニングツリー VLAN ポート コスト（VLAN 単位で設定可能）	1000 Mb/s：4 100 Mb/s：19 10 Mb/s：100
スパニングツリー タイマー	hello タイム：2 秒 転送遅延時間：15 秒 最大エージング タイム：20 秒 転送保留カウント：6 BPDU

（注） Cisco IOS Release 15.2(4)E 以降では、デフォルトの STP モードは Rapid PVST+ です。

スパニングツリー機能の設定方法

スパニングツリー モードの変更

スイッチは次の 3 つのスパニングツリー モードをサポートします。Per-VLAN Spanning-Tree Plus（PVST+）、Rapid PVST+、またはマルチ スパニングツリー プロトコル（MSTP）。デフォルトでは、device Rapid PVST+ プロトコルを実行します。

デフォルト モード以外のモードをイネーブルにする場合、この手順は必須です。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mode** {**pvst** | **mst** | **rapid-pvst**}
4. **interface** *interface-id*
5. **spanning-tree link-type point-to-point**
6. **end**
7. **clear spanning-tree detected-protocols**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mode** {**pvst** | **mst** | **rapid-pvst**} 例： スイッチ(config)# **spanning-tree mode pvst**	スパニングツリー モードを設定します。 すべてのスタック メンバーは、同じバージョンのスパニング ツリーを実行します。 • PVST+ をイネーブルにするには、**pvst** を選択します。 • MSTP をイネーブルにするには、**mst** を選択します。 • rapid PVST+ をイネーブルにするには、**rapid-pvst** を選択します。
ステップ 4	**interface** *interface-id* 例： スイッチ(config)# **interface GigabitEthernet1/0/1**	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。有効なインターフェイスとしては、物理ポート、VLAN、ポート チャネルなどがあります。VLAN ID の範囲は 1 ～ 4094 です。指定できるポートチャネルの範囲は 1 ～ です。
ステップ 5	**spanning-tree link-type point-to-point** 例： スイッチ(config-if)# **spanning-tree link-type point-to-point**	このポートのリンク タイプがポイントツーポイントであることを指定します。 このポート（ローカル ポート）をポイントツーポイント リンクでリモート ポートと接続し、ローカル ポートが指定ポートになると、deviceはリモート ポートとネゴシエーションし、ローカル ポートをフォワーディング ステートにすばやく変更します。
ステップ 6	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**clear spanning-tree detected-protocols** 例：	device 上のいずれかのポートがレガシー IEEE 802.1D device 上のポートに接続されている場合は、このコ

	コマンドまたはアクション	目的
	スイッチ# **clear spanning-tree detected-protocols**	マンドにより device 全体のプロトコル移行プロセスを再開します。 このステップは、このdeviceで Rapid PVST+ が稼働していることを指定deviceが検出する場合のオプションです。

スパニング ツリーのディセーブル化

スパニングツリーはデフォルトで、VLAN 1 およびスパニングツリー限度を上限として新しく作成されたすべての VLAN 上でイネーブルです。スパニングツリーをディセーブルにするのは、ネットワーク トポロジにループがないことが確実な場合だけにしてください。

⚠

注意 スパニングツリーがディセーブルでありながら、トポロジにループが存在していると、余分なトラフィックが発生し、パケットの重複が無限に繰り返されることによって、ネットワークのパフォーマンスが大幅に低下します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **no spanning-tree vlan** *vlan-id*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no spanning-tree vlan** *vlan-id* 例：	*vlan-id* に指定できる範囲は 1 ～ 4094 です。

	コマンドまたはアクション	目的
	スイッチ(config)# **no spanning-tree vlan 300**	
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

ルート デバイスの設定

特定の VLAN で device をルートとして設定するには、**spanning-tree vlan** *vlan-id* **root** グローバル コンフィギュレーション コマンドを使用して、deviceのプライオリティをデフォルト値（32768）から、それより大幅に小さい値に変更します。 このコマンドを入力すると、ソフトウェアが各 VLAN について、ルート devicesのdevice プライオリティを確認します。拡張システム ID をサポートするため、deviceは指定された VLAN の自身のプライオリティを 24576 に設定します。この値によって、このdeviceを指定された VLAN のルートに設定できます。

レイヤ 2 ネットワークの直径（つまり、レイヤ 2 ネットワーク上の任意の 2 つのエンドステーション間deviceの最大ホップカウント）を指定するには、**diameter** キーワードを指定します。ネットワーク直径を指定すると、deviceはその直径を持つネットワークに最適な hello タイム、転送遅延時間、および最大エージング タイムを自動的に設定します。その結果、コンバージェンスに要する時間が大幅に短縮されます。**hello** キーワードを使用して、自動的に計算される hello タイムを上書きできます。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter*
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter* 例： スイッチ(config)# **spanning-tree vlan 20-24 root primary diameter 4**	指定された VLAN のルートになるように、deviceを設定します。 • *vlan-id* には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。 •（任意）**diameter** *net-diameter* には、任意の 2 つのエンドステーション間devicesの最大数を指定します。範囲は 2 ～ 7 です。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

次のタスク

ルートdeviceとしてdeviceを設定した後で、**spanning-tree vlan** *vlan-id* **hello-time**、**spanning-tree vlan** *vlan-id* **forward-time**、および **spanning-tree vlan** *vlan-id* **max-age** グローバル コンフィギュレーション コマンドを使用して、hello タイム、転送遅延時間、および最大エージングタイムを手動で設定することは推奨できません。

セカンダリ ルート デバイスの設定

deviceをセカンダリ ルートとして設定すると、deviceプライオリティがデフォルト値（32768）から 28672 に変更されます。このプライオリティでは、deviceがプライマリ ルート deviceが失敗した場合の、指定された VLAN のルートdeviceになる可能性があります。ここでは、その他のネットワーク devicesが、デフォルトのdeviceプライオリティの 32768 を使用しているためにルート deviceになる可能性が低いことが前提となっています。

このコマンドを複数のdeviceに対して実行すると、複数のバックアップ ルート devicesを設定できます。**spanning-tree vlan** *vlan-id* **root primary** グローバル コンフィギュレーション コマンドでプライマリルート device を設定したときと同じネットワーク直径および hello タイム値を使用してください。

この手順は任意です。

手順の概要

1. **enable**

2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter* 例： `スイッチ(config)# spanning-tree vlan 20-24 root secondary diameter 4`	指定された VLAN のセカンダリ ルートになるように、deviceを設定します。 • *vlan-id* には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。 •（任意）**diameter** *net-diameter* には、任意の 2 つのエンドステーション間devicesの最大数を指定します。指定できる範囲は 2 ～ 7 です。 プライマリ ルート deviceを設定したときと同じネットワーク直径を使用してください。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

ポート プライオリティの設定

（注） device が device スタックのメンバーである場合、**spanning-tree** [**vlan** *vlan-id*] **port-priority** *priority* インターフェイス コンフィギュレーション コマンドの代わりに、**spanning-tree** [**vlan** *vlan-id*] **cost** *cost* インターフェイス コンフィギュレーション コマンドを使用して、フォワーディングステートにするインターフェイスを選択する必要があります。最初に選択させるインターフェイスには、低いコスト値を割り当て、最後に選択させるインターフェイスには高いコスト値を割り当てます。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree port-priority** *priority*
5. **spanning-tree vlan** *vlan-id* **port-priority** *priority*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーションモードを開始します。 有効なインターフェイスは、物理ポートおよびポートチャネル論理インターフェイス（**port-channel** *port-channel-number*）です。
ステップ 4	**spanning-tree port-priority** *priority* 例：	インターフェイスのポートプライオリティを設定します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **spanning-tree port-priority 0**	*priority* に指定できる範囲は 0 ～ 240 で、16 ずつ増加します。デフォルトは 128 です。有効な値は 0、16、32、48、64、80、96、112、128、144、160、176、192、208、224、240 です。その他の値はすべて拒否されます。値が小さいほど、プライオリティが高くなります。
ステップ 5	**spanning-tree vlan** *vlan-id* **port-priority** *priority* 例： スイッチ(config-if)# **spanning-tree vlan 20-25 port-priority 0**	VLAN のポート プライオリティを設定します。 • *vlan-id* には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。 • *priority* に指定できる範囲は 0 ～ 240 で、16 ずつ増加します。デフォルトは 128 です。有効な値は 0、16、32、48、64、80、96、112、128、144、160、176、192、208、224、240 です。その他の値はすべて拒否されます。値が小さいほど、プライオリティが高くなります。
ステップ 6	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

パス コストの設定

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree cost** *cost*
5. **spanning-tree vlan** *vlan-id* **cost** *cost*
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するインターフェイスを指定し、インターフェイス コンフィギュレーションモードを開始します。有効なインターフェイスは、物理ポートおよびポートチャネル論理インターフェイス（**port-channel** *port-channel-number*）です。
ステップ 4	**spanning-tree cost** *cost* 例： スイッチ(config-if)# **spanning-tree cost 250**	インターフェイスのコストを設定します。 ループが発生した場合、スパニングツリーはパス コストを使用して、フォワーディング ステートにするインターフェイスを選択します。低いパス コストは高速送信を表します。 *cost* の範囲は 1 ～ 200000000 です。デフォルト値はインターフェイスのメディア速度から派生します。
ステップ 5	**spanning-tree vlan** *vlan-id* **cost** *cost* 例： スイッチ(config-if)# **spanning-tree vlan 10,12-15,20 cost 300**	VLAN のコストを設定します。 ループが発生した場合、スパニングツリーはパス コストを使用して、フォワーディング ステートにするインターフェイスを選択します。低いパス コストは高速送信を表します。 • *vlan-id* には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。 • *cost* の範囲は 1 ～ 200000000 です。デフォルト値はインターフェイスのメディア速度から派生します。
ステップ 6	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

show spanning-tree interface *interface-id* 特権 EXEC コマンドによって表示されるのは、リンクアップ動作可能状態のポートの情報だけです。そうでない場合は、**show running-config** 特権 EXEC コマンドを使用して設定を確認してください。

VLAN のデバイス プライオリティの設定

deviceのプライオリティを設定して、スタンドアロンdeviceまたはスタックにあるdeviceがルートdeviceとして選択される可能性を高めることができます。

（注） このコマンドの使用には注意してください。deviceのプライオリティを変更する場合は通常、**spanning-tree vlan** *vlan-id* **root primary** および **spanning-tree vlan** *vlan-id* **root secondary** グローバル コンフィギュレーション コマンドを使用することを推奨します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **priority** *priority*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree vlan** *vlan-id* **priority** *priority* 例： `スイッチ(config)# spanning-tree vlan 20 priority 8192`	VLAN のdevice プライオリティの設定 • *vlan-id* には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。

	コマンドまたはアクション	目的
		• *priority* の範囲は 0 ～ 61440 で、4096 ずつ増加します。デフォルトは 32768 です。この値が低いほど、deviceがルート deviceとして選択される可能性が高くなります。 有効なプライオリティ値は4096、8192、12288、16384、20480、24576、28672、32768、36864、40960、45056、49152、53248、57344、61440 です。その他の値はすべて拒否されます。
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

hello タイムの設定

hello タイムはルート deviceによって設定メッセージが生成されて送信される時間の間隔です。

この手順は任意です。

手順の概要

1. **enable**
2. **spanning-tree vlan** *vlan-id* **hello-time** *seconds*
3. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**spanning-tree vlan** *vlan-id* **hello-time** *seconds* 例： スイッチ(config)# **spanning-tree vlan 20-24 hello-time 3**	VLAN の hello タイムを設定します。hello タイムはルート deviceによって設定メッセージが生成されて送信される時間の間隔です。このメッセージは、deviceが活動中であることを表します。 • *vlan-id* には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。

	コマンドまたはアクション	目的
		• *seconds* に指定できる範囲は 1 ～ 10 です。デフォルトは 2 です。
ステップ 3	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

VLAN の転送遅延時間の設定

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **forward-time** *seconds*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree vlan** *vlan-id* **forward-time** *seconds* 例： スイッチ(config)# **spanning-tree vlan 20,25 forward-time 18**	VLAN の転送時間を設定します。転送遅延時間は、スパニングツリー ラーニング ステートおよびリスニング ステートからフォワーディング ステートに移行するまでに、インターフェイスが待機する秒数です。 • *vlan-id* には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。

	コマンドまたはアクション	目的
		• *seconds* に指定できる範囲は 4 ～ 30 です。デフォルトは 15 です。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

VLAN の最大エージング タイムの設定

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **max-age** *seconds*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree vlan** *vlan-id* **max-age** *seconds* 例： スイッチ(config)# **spanning-tree vlan 20 max-age 30**	VLAN の最大エージング タイムを設定します。最大エージング タイムは、deviceが再設定を試す前にスパニングツリー設定メッセージを受信せずに待機する秒数です。 • *vlan-id* には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。

	コマンドまたはアクション	目的
		• *seconds* に指定できる範囲は 6 ～ 40 です。デフォルトは 20 です。
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

転送保留カウントの設定

転送保留カウント値を変更することで、BPDU のバースト サイズを設定できます。

（注） このパラメータをより高い値に変更すると、（特に Rapid PVST+ モードで）CPU の使用率に大きく影響します。逆に、この値を低く設定すると、セッションによってはコンバージェンスを抑えることができます。この値は、デフォルト設定で使用することを推奨します。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree transmit hold-count** *value*
4. **end**

手順の詳細

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

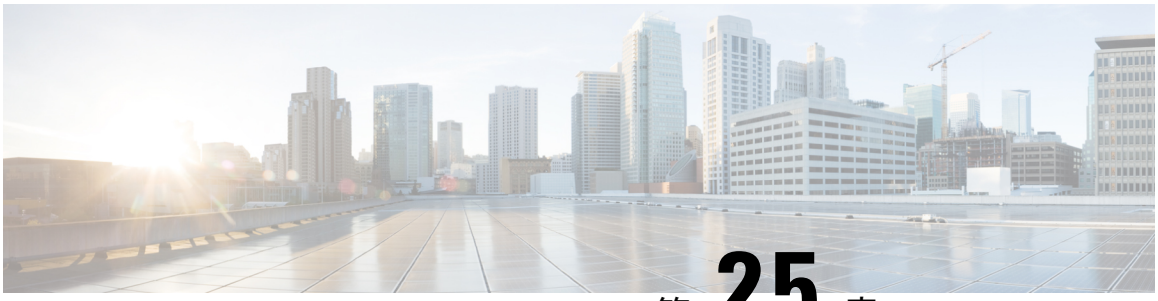
	コマンドまたはアクション	目的
ステップ 3	**spanning-tree transmit hold-count** *value* 例： `スイッチ(config)# spanning-tree transmit hold-count 6`	1 秒間停止する前に送信できる BPDU 数を設定します。 *value* に指定できる範囲は 1 ～ 20 です。デフォルト値は 6 です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

スパニングツリー ステータスのモニタリング

表 *44:* スパニングツリー ステータス表示用のコマンド

show spanning-tree active	アクティブ インターフェイスに関するスパニングツリー情報だけを表示します。
show spanning-tree detail	インターフェイス情報の詳細サマリーを表示します。
show spanning-tree vlan *vlan-id*	指定した VLAN のスパニング ツリー情報を表示します。
show spanning-tree interface *interface-id*	指定したインターフェイスのスパニングツリー情報を表示します。
show spanning-tree interface *interface-id* **portfast**	指定したインターフェイスのスパニングツリー portfast 情報を表示します。
show spanning-tree summary [**totals**]	インターフェイス ステートのサマリーを表示します。または STP ステート セクションのすべての行を表示します。

スパニングツリーカウンタをクリアするには、**clear spanning-tree** [**interface** *interface-id*] 特権 EXEC コマンドを使用します。



第 25 章

複数のスパニング ツリー プロトコルの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MSTP の前提条件

- 2つ以上のdevicesを同じマルチスパニングツリー（MST）リージョンに設定するには、その 2 つに同じ VLAN/インスタンス マッピング、同じコンフィギュレーション リビジョン番号、同じ名前を設定しなければなりません。

- ネットワーク内の冗長パスでロード バランシングを機能させるには、すべての VLAN/インスタンスマッピングの割り当てが一致している必要があります。一致していないと、すべてのトラフィックが 1 つのリンク上で伝送されます。
- Per-VLAN Spanning-Tree Plus（PVST+）と MST クラウドの間、または Rapid- PVST+ と MST クラウドの間でロードバランシングが機能するためには、すべての MST 境界ポートがフォワーディングでなければなりません。MST クラウドの内部スパニングツリー（IST）のルートが共通スパニングツリー（CST）のルートである場合、MST 境界ポートはフォワーディングです。MST クラウドが複数の MST リージョンから構成されている場合、いずれかの MST リージョンに CST ルートを含める必要があり、その他すべての MST リージョンに、PVST+ クラウドまたは高速 PVST+ クラウドを通るパスよりも、MST クラウド内に含まれるルートへのパスが良くする必要があります。クラウド内のdevicesを手動で設定しなければならない場合もあります。

MSTP の制約事項

- device スタックは最大 65 の MST インスタンスをサポートします。特定の MST インスタンスにマッピング可能な VLAN 数に制限はありません。
- PVST+、Rapid PVST+、および MSTP はサポートされますが、アクティブにできるのは 1 つのバージョンだけです（たとえば、すべての VLAN で PVST+ を実行する、すべての VLAN で Rapid PVST+ を実行する、またはすべての VLAN で MSTP を実行します）。
- MST コンフィギュレーションの VLAN トランキング プロトコル（VTP）伝搬はサポートされません。ただし、コマンドライン インターフェイス（CLI）または簡易ネットワーク管理プロトコル（SNMP）サポートを通じて、MST リージョン内の各deviceで MST コンフィギュレーション（リージョン名、リビジョン番号、および VLAN とインスタンスのマッピング）を手動で設定することは可能です。
- ネットワークを多数のリージョンに分割することは推奨できません。ただし、どうしても分割せざるを得ない場合は、スイッチド LAN をルータまたは非レイヤ 2 デバイスで相互接続された小規模な LAN に分割することを推奨します。
- リージョンは、同じ MST コンフィギュレーションを持つ 1 つまたは複数のメンバーで構成されます。リージョンの各メンバーは高速スパニングツリープロトコル（RSTP）ブリッジ プロトコル データ ユニット（BPDU）を処理する機能を備えている必要があります。ネットワーク内の MST リージョンの数に制限はありませんが、各リージョンは最大 65 のスパニングツリー インスタンスのみをサポートできます。VLAN には、一度に 1 つのスパニングツリー インスタンスのみ割り当てることができます。

MSTP について

MSTP の設定

高速コンバージェンスのために RSTP を使用する MSTP では、複数の VLAN をグループ化して同じスパニングツリーインスタンスにマッピングすることが可能で、多くの VLAN をサポートするのに必要なスパニングツリー インスタンスの数を軽減できます。MSTP は、データ トラフィックに複数の転送パスを提供し、ロード バランシングを実現して、多数の VLAN をサポートするのに必要なスパニングツリーインスタンスの数を減らすことができます。MSTP を使用すると、1 つのインスタンス（転送パス）で障害が発生しても他のインスタンス（転送パス）は影響を受けないので、ネットワークのフォールトトレランスが向上します。

（注） マルチ スパニングツリー（MST）実装は IEEE 802.1s 標準に準拠しています。

MSTP を導入する場合、最も一般的なのは、レイヤ 2 スイッチド ネットワークのバックボーンおよびディストリビューション レイヤへの導入です。MSTP の導入により、サービス プロバイダー環境に求められる高可用性ネットワークを実現できます。

device が MST モードの場合、IEEE 802.1w 準拠の RSTP が自動的にイネーブルになります。RSTP は、IEEE 802.1D の転送遅延を軽減し、ルート ポートおよび指定ポートをフォワーディング ステートにすばやく移行する明示的なハンドシェイクによって、スパニングツリーの高速コンバージェンスを実現します。

MSTP と RSTP は、既存のシスコ独自の Multiple Instance STP（MISTP）、および既存の Cisco PVST+ と Rapid Per-VLAN Spanning-Tree plux（Rapid PVST+）を使用して、スパニングツリーの動作を改善し、（オリジナルの）IEEE 802.1D スパニング ツリーに準拠した機器との下位互換性を保持しています。

device スタックは、ネットワークのその他の部分に対しては単一のスパニングツリー ノードに見え、すべてのスタック メンバーが同一の device ID を使用します。

MSTP 設定時の注意事項

- **spanning-tree mode mst** グローバル コンフィギュレーション コマンドを使用して、MST をイネーブルにすると、RSTP が自動的にイネーブルになります。
- UplinkFast、BackboneFast、クロススタック UplinkFast の設定のガイドラインについては、関連項目のセクションの該当するセクションを参照してください。
- device が MST モードの場合は、パス コスト値の計算に、ロング パス コスト計算方式（32 ビット）が使用されます。ロング パス コスト計算方式では、次のパス コスト値がサポートされます。

速度	パス コスト値
10 Mb/s	2,000,000
100 Mb/s	200,000
1 Gb/s	20,000
10 Gb/s	2,000
100 Gb/s	200

ルート スイッチ

deviceは、マッピングされているVLANグループのスパニングツリーインスタンスを保持しています。device IDは、deviceのプライオリティおよびdeviceのMACアドレスで構成されており、各インスタンスに関連付けられます。VLANのグループでは、最小のdevice IDをもつdeviceがルート deviceになります。

deviceをルートとして設定する場合は、deviceプライオリティをデフォルト値（32768）からそれより大幅に低い値に変更し、deviceが、指定したスパニング ツリー インスタンスのルート deviceになるようにします。このコマンドを入力すると、deviceはルート devicesのdevice プライオリティをチェックします。拡張システム IDをサポートしているため、24576という値でdevicesが指定したスパニングツリーインスタンスのルートとなる場合、そのdeviceは指定したインスタンスに対する自身のプライオリティを24576に設定します。

指定されたインスタンスのルート deviceに24576に満たないdevice プライオリティが設定されている場合は、deviceは自身のプライオリティを最小のdevice プライオリティより4096だけ小さい値に設定します（4096は4ビット device プライオリティの最下位ビットの値です）。詳細については、関連項目の「ブリッジ ID、スイッチ プライオリティ、および拡張システム ID デバイス」リンクを参照してください。

ネットワークが、拡張システム IDをサポートするdevicesとサポートしないものの両方で構成されている場合、拡張システム IDをサポートするdeviceがルート deviceになる可能性は低くなります。古いソフトウェアを実行している接続deviceのプライオリティより VLAN番号が大きい場合は常に、拡張システム IDによってスイッチ プライオリティ値が増加します。

各スパニングツリー インスタンスのルート deviceは、バックボーンまたはディストリビューション deviceでなければなりません。アクセス deviceをスパニングツリープライマリ ルートとして設定しないでください。

レイヤ2ネットワークの直径（つまり、レイヤ2ネットワーク上の任意の2つのエンドステーション間の最大 device ホップカウント）を指定するには、**diameter** キーワード（MSTインスタンスが0の場合のみ使用できる）を指定します。ネットワーク直径を指定すると、deviceはその直径を持つネットワークに最適な helloタイム、転送遅延時間、および最大エージング タイムを自動的に設定します。その結果、コンバージェンスに要する時間が大幅に短縮されます。**hello** キーワードを使用して、自動的に計算される helloタイムを上書きできます。

MST リージョン

スイッチを MST インスタンスに加入させるには、同じ MST コンフィギュレーション情報を使用して矛盾のないようにスイッチを設定する必要があります。同じ MST 設定の相互接続スイッチの集まりによって MST リージョンが構成されます。

MST 設定では、それぞれのdeviceが属する MST リージョンが制御されます。この設定には、領域の名前、バージョン番号、MST VLAN とインスタンスの割り当てマップが含まれます。その中で MST リージョンの設定を指定することにより、リージョンのdeviceを設定します。MST インスタンスに VLAN をマッピングし、リージョン名を指定して、リビジョン番号を設定できます。手順と例については、関連項目の「MST リージョン設定の指定と MSTP のイネーブル化」リンクをクリックします。

リージョンには、同一の MST コンフィギュレーションを持った 1 つまたは複数のメンバが必要です。さらに、各メンバは、RSTP ブリッジプロトコル データ ユニット（BPDU）を処理できる必要があります。ネットワーク内の MST リージョンの数に制限はありませんが、各リージョンは最大 65 のスパニングツリー インスタンスをサポートできます。インスタンスは、0 ～ 4094 の範囲の任意の番号で識別できます。VLAN には、一度に 1 つのスパニングツリー インスタンスのみ割り当てることができます。

IST、CIST、CST

すべてのスパニングツリー インスタンスが独立している PVST+ および Rapid PVST+ とは異なり、MSTP は次の 2 つのタイプのスパニングツリーを確立して保持しています。

- Internal Spanning-Tree（IST）は、1 つの MST リージョン内で稼働するスパニングツリーです。

  各 MST リージョン内の MSTP は複数のスパニングツリー インスタンスを維持しています。インスタンス 0 は、リージョンの特殊なインスタンスで、IST と呼ばれています。その他すべての MSTI には、1 ～ 4094 の番号が付きます。

  IST は、BPDU を送受信する唯一のスパニングツリー インスタンスです。他のスパニングツリーの情報はすべて、MSTP BPDU 内にカプセル化されている M レコードに格納されています。MSTP BPDU はすべてのインスタンスの情報を伝送するので、複数のスパニングツリー インスタンスをサポートする処理が必要な BPDU の数を大幅に減少できます。

  同一リージョン内の すべての MST インスタンスは同じプロトコル タイマーを共有しますが、各 MST インスタンスは独自のトポロジ パラメータ（ルート device ID、ルート パス コストなど）を持っています。デフォルトでは、すべての VLAN が IST に割り当てられます。

  MSTI はリージョンにローカルです。たとえばリージョン A およびリージョン B が相互接続されていても、リージョン A の MSTI 1 は、リージョン B の MSTI 1 に依存しません。

- Common and Internal Spanning-Tree（CIST）は、各 MST リージョン内の IST と、MST リージョンおよびシングル スパニングツリーを相互接続する Common Spanning-Tree（CST）の集合です。

1 つのリージョン内で計算されたスパニングツリーは、スイッチド ドメイン全体を網羅する CST のサブツリーと見なされます。CIST は、IEEE 802.1w、IEEE 802.1s、および IEEE 802.1D 標準をサポートするスイッチ間で実行されるスパニングツリーアルゴリズムによって形成されます。MST リージョン内の CIST は、リージョン外の CST と同じです。

MST リージョン内の動作

IST は 1 つのリージョン内のすべての MSTP スイッチを接続します。IST が収束すると、IST のルートは CIST リージョナル ルートになります。これは、リージョン内で最も小さいdevice ID、および CIST ルートに対するパス コストをもつdeviceです。ネットワークに領域が 1 つしかない場合、CIST リージョナル ルートは CIST ルートにもなります。CIST ルートがリージョンの外部にある場合、リージョンの境界に位置する MSTP スイッチの 1 つが CIST リージョナル ルートとして選択されます。

MSTP deviceは初期化時に、自身が CIST のルートおよび CIST リージョナル ルートであることを主張するために CIST ルートと CIST リージョナル ルートへのパス コストがいずれもゼロに設定された BPDU を送信します。deviceはすべての MSTI を初期化し、そのすべてのルートであることを主張します。deviceは、ポート用に現在保存されているものより上位の MST ルート情報（低いdevice ID、低いパス コストなど）を受信した場合、CIST リージョナル ルートとしての主張を放棄します。

リージョンには、初期化中に多くのサブ リージョンが含まれて、それぞれに独自の CIST リージョナル ルートが含まれることがあります。スイッチは、優位の IST 情報を受信すると、古いサブリージョンを脱退して、真の CIST リージョナル ルートが含まれている新しいサブリージョンに加入します。真の CIST リージョナル ルートが含まれている以外のサブリージョンは、すべて縮小します。

正常な動作のためには、MST リージョン内のすべてのスイッチが同じ CIST リージョナル ルートを承認する必要があります。共通の CIST リージョナル ルートに収束する場合、そのリージョン内にある 2 つのスイッチは、1 つの MST インスタンスに対するポートの役割のみを同期させます。

MST リージョン間の動作

ネットワーク内に複数のリージョンまたはレガシー IEEE 802.1D devicesが混在している場合、MSTP は、ネットワーク内のすべての MST リージョンとすべてのレガシー STP devicesから構成される CST を構築して保持します。MSTI は、リージョンの境界にある IST と組み合わさり、CST になります。

IST はリージョン内のすべての MSTP devicesを接続し、スイッチド ドメイン全体を囲む CIST のサブツリーとして認識されます。サブツリーのルートは CIST リージョナル ルートです。MST リージョンは、隣接する STP deviceおよび MST リージョンへの仮想devicesとして認識されます。

CST インスタンスのみが BPDU を送受信し、MST インスタンスはスパニングツリー情報を BPDU に追加して隣接するdevicesと相互作用し、最終的なスパニングツリー トポロジーを算出します。したがって、BPDU 伝送に関連するスパニングツリー パラメータ（hello タイム、転送時間、最大エージング タイム、最大ホップ カウントなど）は、CST インスタンスだけで設

定されますが、その影響はすべての MST インスタンスに及びます。スパニングツリー トポロジに関連するパラメータ（device プライオリティ、ポート VLAN コスト、ポート VLAN プライオリティなど）は、CST インスタンスと MST インスタンスの両方で設定できます。

MSTP devicesは、バージョン 3 RSTP BPDU または IEEE 802.1D STP BPDU を使用して、レガシー IEEE 802.1D devicesと通信します。MSTP devicesは、MSTP BPDU を使用して MSTP devices と通信します。

IEEE 802.1s の用語

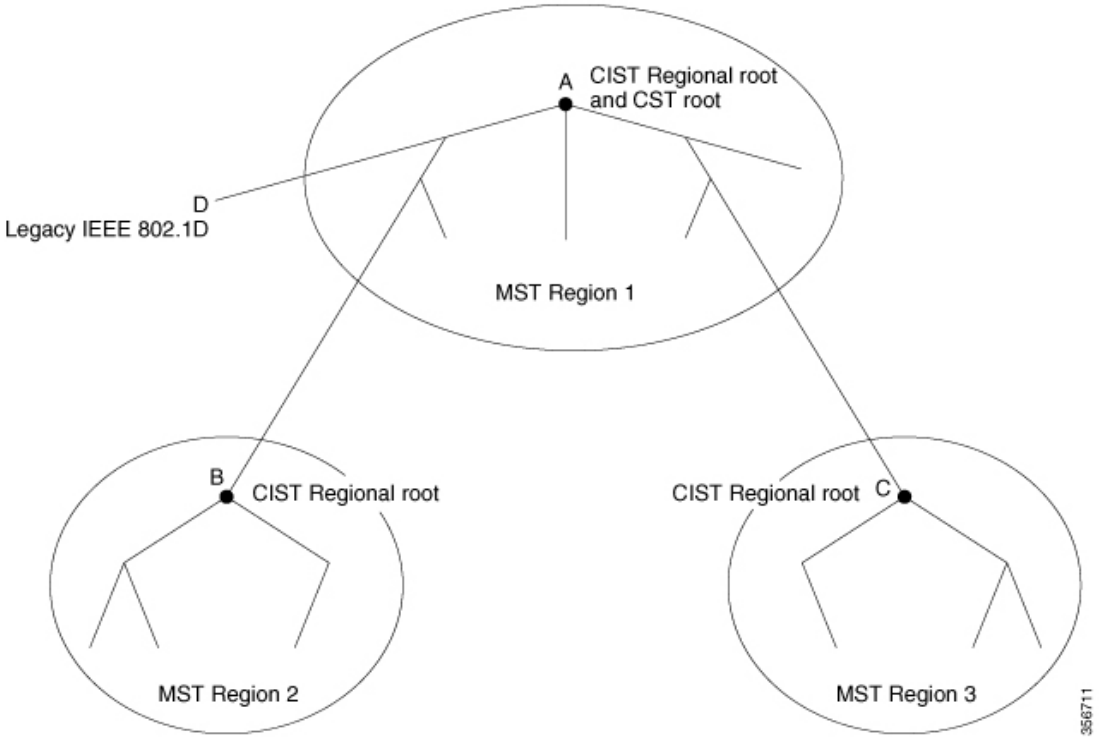
シスコの先行標準実装で使用される一部のMST命名規則は、一部の内部パラメータまたはリージョンパラメータを識別するように変更されました。これらのパラメータは、ネットワーク全体に関連している外部パラメータと違い、MST リージョン内でのみ影響があります。CIST はネットワーク全体を網羅するスパニングツリー インスタンスのため、CIST パラメータのみ、内部修飾子やリージョナル修飾子ではなく外部修飾子が必要です。

- CIST ルートは、ネットワーク全体を網羅する一意のインスタンスのためのルート deviceです。
- CIST 外部ルート パス コストは、CIST ルートまでのコストです。このコストは MST 領域内で変化しません。MST リージョンは、CIST への単一deviceと見なすことに注意してください。CIST 外部ルート パス コストは、これらの仮想devices、およびどのリージョンにも属さないdevicesの間で算出されるルート パス コストです。
- CIST ルートが領域内にある場合、CIST リージョナル ルートは CIST ルートです。CIST ルートがリージョン内にない場合、CIST リージョナル ルートは、リージョン内の CIST ルートに最も近いdeviceです。CIST リージョナル ルートは、IST のルート deviceとして動作します。
- CIST 内部ルート パス コストは、領域内の CIST リージョナル ルートまでのコストです。このコストは、IST つまりインスタンス 0 だけに関連します。

MST リージョンの図

この図は、3 個の MST リージョンとレガシー IEEE 802.1D device（D）を示しています。リージョン 1 の CIST リージョナル ルート（A）は、CIST ルートでもあります。リージョン 2 の CIST リージョナル ルート（B）、およびリージョン 3 の CIST リージョナル ルート（C）は、CIST 内のそれぞれのサブツリーのルートです。RSTP はすべてのリージョンで稼働しています。

図 *24 : MST* リージョン、*CIST* リージョナルルート、*CST* ルート



ホップ カウント

IST および MST インスタンスは、スパニングツリー トポロジの計算に、コンフィギュレーション BPDU のメッセージ有効期間と最大エージング タイムの情報を使用しません。その代わりに、IP Time To Live（TTL）メカニズムに似た、ルートまでのパス コストおよびホップ カウント メカニズムを使用します。

spanning-tree mst max-hops グローバル コンフィギュレーション コマンドを使用すると、領域内で最大ホップ カウントを設定し、その領域の IST およびすべての MST インスタンスに適用できます。ホップ カウントを設定すると、メッセージ エージ情報を設定するのと同様の結果が得られます（再構成の開始時期を決定します）。インスタンスのルート deviceは、コストが 0 でホップ カウントが最大値に設定されている BPDU（M レコード）を常に送信します。device は、この BPDU を受信すると、受信した残りのホップ カウントから 1 を引き、生成する BPDU で残りのホップ カウントとしてこの値を伝播します。カウントがゼロに達すると、deviceは BPDU を廃棄し、ポート用に維持されている情報を期限切れにします。

BPDU の RSTP 部分に格納されているメッセージ有効期間と最大エージング タイムの情報は、リージョン全体で同じままであり、そのリージョンの境界に位置する指定ポートによって同じ値が伝播されます。

境界ポート

シスコ先行標準の実装では、境界ポートは、RSTP が稼働する単一のスパニングツリー リージョン、PVST+ または Rapid PVST+ が稼働する単一のスパニングツリー リージョン、または異なる MST コンフィギュレーションを持つ別の MST リージョンに MST リージョンを接続します。境界ポートは、LAN、単一のスパニングツリーdeviceまたはMST設定が異なるdeviceの指定deviceにも接続します。

IEEE 802.1s 標準では、境界ポートの定義はなくなりました。IEEE 802.1Q-2002 標準では、ポートが受信できる 2 種類のメッセージを識別します。

- 内部（同一リージョンから）
- 外部（別のリージョンから）

メッセージが内部の場合、CIST の部分は CIST によって受信されるので、各 MST インスタンスは個々の M レコードだけを受信します。

メッセージが外部である場合、CIST だけが受信します。CIST の役割がルートや代替ルートの場合、または外部 BPDU のトポロジが変更された場合は、MST インスタンスに影響する可能性があります。

MST リージョンには、devicesおよびLANの両方が含まれます。セグメントは、DPのリージョンに属します。そのため、セグメントの指定ポートではなく異なるリージョンにあるポートは境界ポートになります。この定義では、リージョン内部の2つのポートが、別のリージョンに属するポートとセグメントを共有し、内部メッセージおよび外部メッセージの両方を 1 つのポートで受信できるようになります。

シスコ先行標準の実装との主な違いは、STP 互換モードを使用している場合、指定ポートが境界ポートとして定義されない点です。

（注） レガシー STP deviceがセグメントに存在する場合、メッセージは常に外部と見なされます。

シスコ先行標準の実装から他に変更された点は、送信device ID を持つ RSTP またはレガシー IEEE 802.1Q deviceの部分に、CIST リージョナル ルート device ID フィールドが加えられたことです。リージョン全体は、一貫した送信者device ID をネイバー devicesに送信し、単一仮想deviceのように動作します。この例では、A または B がセグメントに指定されているかどうかに関係なく、ルートの一貫した送信者device ID が同じである BPDU をdevice C が受信します。

IEEE 802.1s の実装

シスコの IEEE MST 標準の実装には、標準の要件を満たす機能だけでなく、すでに公開されている標準には含まれていない一部の（要望されている）先行標準の機能が含まれています。

ポートの役割名の変更

境界の役割は最終的に MST 標準に含まれませんでしたが、境界の概念自体はシスコの実装に投影されています。ただし、リージョン境界にある MST インスタンスのポートは、対応する CIST ポートのステートに必ずしも従うわけではありません。現在、2 つの境界の役割が存在しています。

- 境界ポートが CIST リージョナル ルートのルート ポートである場合：CIST インスタンス ポートを提案されて同期中の場合、対応するすべての MSTI ポートの同期を取り終わった後であれば（その後フォワーディングします）、その場合のみ合意を返信してフォワーディング ステートに移行できます。MSTI ポートには、特別なプライマリ ロールがあります。
- 境界ポートが CIST リージョナル ルートのルート ポートでない：MSTI ポートは、CIST ポートのステートおよび役割に従います。標準では提供される情報が少ないため、MSTI ポートが BPDU（M レコード）を受信しない場合、MSTI ポートが BPDU を代わりにブロックできる理由がわかりにくい場合があります。この場合、境界の役割自体は存在していませんが、**show** コマンドで見ると、出力される *type* カラムで、ポートが境界ポートとして認識されていることがわかります。

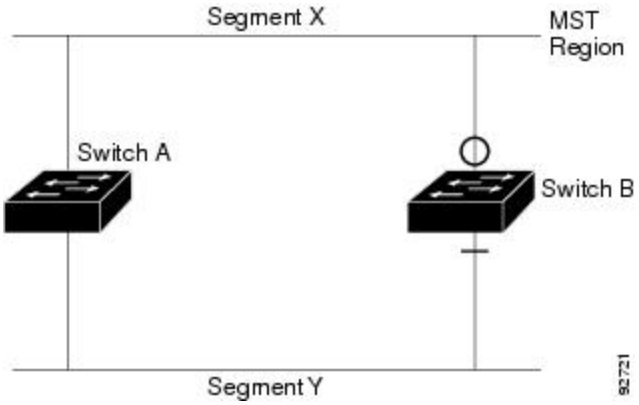
レガシーおよび規格Devicesの相互運用

準規格devicesの自動検出はエラーになることがあるので、インターフェイス コンフィギュレーション コマンドを使用して準規格ポートを識別できます。deviceの規格と準規格の間にリージョンを形成することはできませんが、CIST を使用して相互運用することができます。このような特別な方法を採用しても、失われる機能は、異なるインスタンス上のロードバランシングだけです。ポートが先行標準の BPDU を受信すると、CLI（コマンドライン インターフェイス）にはポートの設定に応じて異なるフラグが表示されます。deviceが準規格 BPDU 送信用に設定されていないポートで準規格 BPDU を初めて受信したときは、Syslog メッセージも表示されます。

図 *25 :* 規格および準規格のデバイスの相互運用

A が規格のdeviceで、B が準規格のdeviceとして、両方とも同じリージョンに設定されているとします。A は CIST のルート device です。B のセグメント X にはルートポート（BX）、セグメント Y には代替ポート（BY）があります。セグメント Y がフラップして BY のポートが代替になってから準規格 BPDU を 1 つ送信すると、AY は準規格 device が Y に接続されていることを検出できず、規格 BPDU の送信を続けます。ポート BY は境界に固定され、A と B との間でのロード ランシングは不可能になります。セグメント X にも同じ問題がありますが、B は

トポロジの変更であれば送信する場合があります。



（注） 規格 MST 実装と準規格 MST 実装間の相互作用を最低限に抑えることを推奨します。

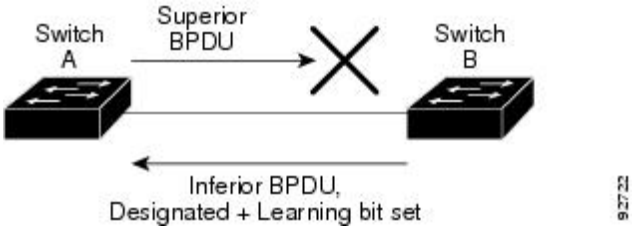
単一方向リンク障害の検出

IEEE MST 標準にはこの機能が存在していませんが、Cisco IOS Release には加えられています。ソフトウェアは、受信した BPDU でポートのロールおよびステートの一貫性をチェックし、ブリッジング ループの原因となることがある単方向リンク障害を検出します。

指定ポートは、矛盾を検出すると、その役割を維持しますが、廃棄ステートに戻ります。一貫性がない場合は、接続を中断した方がブリッジング ループを解決できるからです。

図 *26 :* 単一方向リンク障害の検出

次の図に、ブリッジングループの一般的な原因となる単方向リンク障害を示します。デバイス A はルートdeviceであり、device B へのリンクで BPDU は失われます。RSTP および MST BPDU には、送信側ポートの役割と状態が含まれます。device A はこの情報を使用し、ルータ A が送信する上位 BPDU にdevice B が反応しないこと、およびdevice B がルート deviceではなく指定ブリッジであることを検出できます。この結果、device A は、そのポートをブロックし（またはブロックし続け）、ブリッジング ループが防止されます。



IEEE 802.1D STP との相互運用性

MSTP が稼働しているdeviceは、IEEE 802.1D 準拠のレガシー devicesとの相互運用を可能にする組み込み型のプロトコル移行メカニズムをサポートします。このdeviceは、レガシー IEEE 802.1D コンフィギュレーション BPDU（プロトコルバージョンが 0 に設定されている BPDU）

を受信すると、そのポート上では IEEE 802.1D BPDU のみを送信します。また、MSTP device は、レガシー BPDU、別のリージョンに関連付けられている MSTP BPDU（バージョン 3）、または RSTP BPDU（バージョン 2）を受信することによって、ポートがリージョンの境界に位置していることを検出できます。

ただし、deviceが IEEE 802.1D BPDU を受信していない場合は、自動的に MSTP モードに戻りません。これはレガシー deviceが指定deviceでない限り、レガシー deviceがリンクから削除されたかどうか検出できないためです。このdeviceが接続するdeviceがリージョンに加入していると、deviceはポートに境界の役割を割り当て続ける場合があります。プロトコル移行プロセスを再開するには（強制的にネイバー devices と再びネゴシエーションするには）、**clear spanning-tree detected-protocols** 特権 EXEC コマンドを使用します。

リンク上のすべてのレガシー devicesが RSTP devicesであれば、これらのスイッチは、RSTP BPDU 同様に MSTP BPDU を処理できます。したがって、MSTP devicesは、バージョン 0 コンフィギュレーションと TCN BPDU またはバージョン 3 MSTP BPDU のいずれかを境界ポートで送信します。境界ポートは、LAN、単一スパニングツリー deviceまたは MST 設定が異なる deviceのいずれかの指定のdeviceに接続します。

RSTP 概要

RSTP は、ポイントツーポイントの配線を利用して、スパニングツリーの高速コンバージェンスを実現します。また、1 秒未満の間に、スパニングツリーを再構成できます（IEEE 802.1D スパニングツリーのデフォルトに設定されている 50 秒とは異なります）。

ポートの役割およびアクティブ トポロジ

RSTP は、ポートに役割を割り当てて、アクティブ トポロジを学習することによって高速コンバージェンスを実現します。RSTP は device をルート deviceとして最も高いdevice プライオリティ（プライオリティの数値が一番小さい）に選択するために、IEEE 802.1D STP 上に構築されます。RSTP は、次のうちいずれかのポートの役割をそれぞれのポートに割り当てます。

- ルート ポート：device がルートdevice にパケットを転送するとき、最適なパス（最低コスト）を提供します。
- 指定ポート：指定deviceに接続し、その LAN からルート deviceにパケットを転送するとき、パス コストを最低にします。DP は、指定deviceが LAN に接続されているポートです。
- 代替ポート：現在のルート ポートが提供したパスに代わるルート deviceへの代替パスを提供します。
- バックアップ ポート：指定ポートが提供した、スパニングツリーのリーフに向かうパスのバックアップとして機能します。バックアップ ポートは、2 つのポートがループバック内でポイントツーポイント リンクによって接続されるか、共有 LAN セグメントとの複数の接続がdeviceにある場合に限って存在できます。
- ディセーブル ポート：スパニングツリーの動作において何も役割が与えられていません。

ルート ポートまたは指定ポートのロールを持つポートは、アクティブなトポロジに含まれます。代替ポートまたはバックアップ ポートのロールがあるポートは、アクティブ トポロジから除外されます。

ネットワーク全体のポートの役割に矛盾のない安定したトポロジでは、RSTPは、すべてのルートポートおよび指定ポートがただちにフォワーディング ステートに移行し、代替ポートとバックアップ ポートが必ず廃棄ステート（IEEE 802.1D のブロッキング ステートと同じ）になるように保証します。ポートのステートにより、転送処理および学習処理の動作が制御されます。

表 *45 :* ポート ステートの比較

運用ステータス	STP ポート ステート（IEEE 802.1D）	RSTP ポート ステート	ポートがアクティブ トポロジに含まれているか
イネーブル	ブロッキング	廃棄	×
イネーブル	リスニング	廃棄	×
イネーブル	ラーニング	ラーニング	○
イネーブル	転送	転送	○
ディセーブル	ディセーブル	廃棄	×

Cisco STP の実装との一貫性を保つため、このマニュアルでは、ポート ステートを廃棄ではなくブロッキングとして定義します。DP はリスニング ステートから開始します。

高速コンバージェンス

RSTP は、device、device ポート、LAN のうちいずれかの障害のあと、接続の高速回復を提供します。エッジ ポート、新しいルート ポート、ポイントツーポイント リンクで接続したポートに、高速コンバージェンスが次のように提供されます。

- エッジポート：**spanning-tree portfast** インターフェイス コンフィギュレーション コマンドを使用して RSTP deviceでエッジポートとしてポートを設定した場合、エッジポートはフォワーディング ステートにすぐに移行します。エッジ ポートは Port Fast 対応ポートと同じであり、単一エンドステーションに接続しているポートだけでイネーブルにする必要があります。
- ルート ポート：RSTP は、新しいルート ポートを選択した場合、古いルート ポートをブロックし、新しいルート ポートをフォワーディング ステートにすぐに移行します。
- ポイントツーポイント リンク：ポイントツーポイント リンクによってあるポートと別のポートを接続することでローカルポートが指定ポートになると、提案合意ハンドシェイクを使用して他のポートと急速な移行がネゴシエートされ、トポロジにループがなくなります。

図 *27*: 高速コンバージェンスの提案と合意のハンドシェイク

デバイス A がデバイス B にポイントツーポイント リンクで接続され、すべてのポートはブロッキング ステートになっています。デバイス A の優先度がデバイス B の優先度よりも数値的に小さいとします。デバイス A は提案メッセージ（提案フラグを設定した設定 BPDU）をデバイス B に送信し、指定deviceとしてそれ自体を提案します。

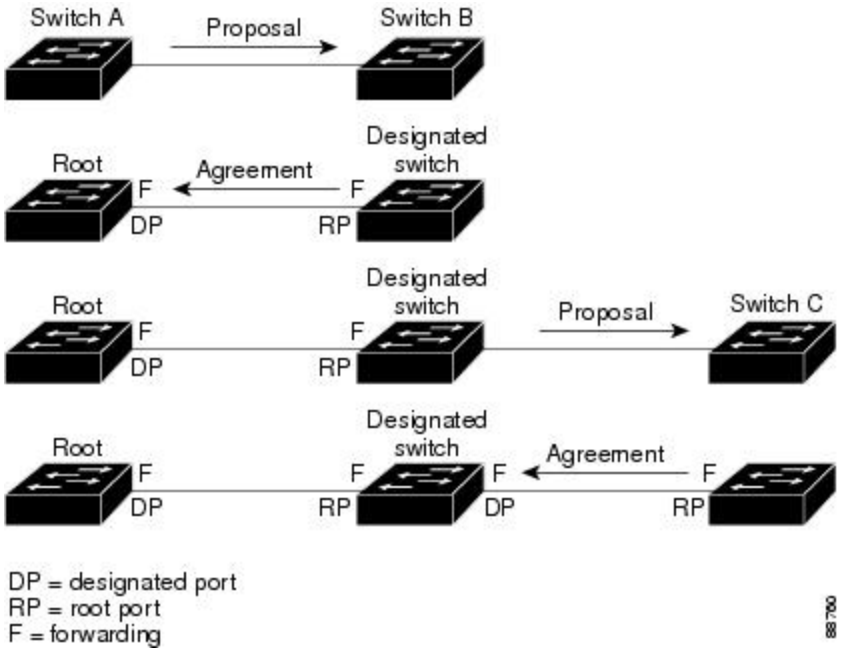
デバイス B は、提案メッセージの受信後、提案メッセージを受信したポートを新しいルート ポートとして選択し、エッジ以外のすべてのポートを強制的にブロッキング ステートにして、新しいルートポートを介して合意メッセージ（合意フラグを設定した BPDU）を送信します。

デバイス A も、デバイス B の合意メッセージの受信後、指定ポートをフォワーディング ステートにすぐに移行します。デバイス B はすべてのエッジ以外のポートをブロックし、Devices A およびルータ B の間にポイントツーポイント リンクがあるので、ネットワークにループは形成されません。

デバイス C がデバイス B に接続すると、同様のセットのハンドシェーク メッセージが交換されます。デバイス C はデバイス B に接続されているポートをルート ポートとして選択し、両端がフォワーディング ステートにすぐに移行します。このハンドシェーク処理を繰り返して、もう 1 つのdeviceがアクティブ トポロジーに加わります。ネットワークが収束すると、この提案/合意ハンドシェイクがルートからスパニングツリーのリーフへと進みます。

device スタックでは、Cross-Stack Rapid Transition（CSRT）機能を使用すると、ポートがフォワーディング ステートに移行する前に、スタック メンバで、提案/合意ハンドシェイク中にすべてのスタック メンバーから確認メッセージを受信できます。deviceが MST モードの場合、CSRT は自動的に有効にされます。

deviceはポートのデュプレックス モードによってリンク タイプを学習します。全二重ポートはポイントツーポイント接続と見なされ、半二重接続は共有接続と見なされます。デュプレックス設定によって制御されるデフォルト設定を無効にするには、**spanning-tree link-type** インターフェイス コンフィギュレーション コマンドを入力します。



ポート ロールの同期

deviceがそのルータのポートの 1 つで提案メッセージを受信し、そのポートが新しいルート ポートとして選択されると、RSTP によってその他すべてのポートが新しいルートの情報と強制的に同期化します。

その他すべてのポートが同期化されている場合、deviceはルート ポートで受信した上位ルート情報で同期化されます。deviceのそれぞれのポートは、次のような場合に同期化します。

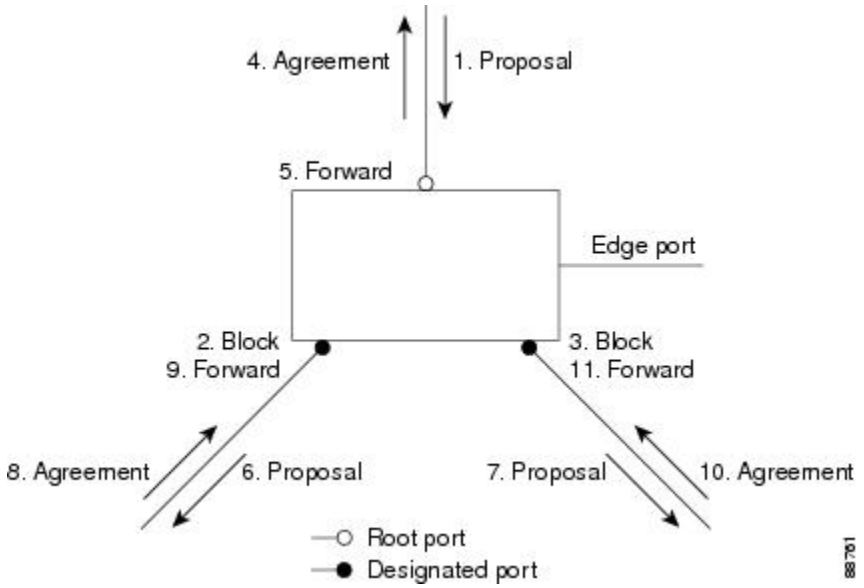
- ポートがブロッキング ステートである。
- エッジ ポートである（ネットワークのエッジに存在するように設定されたポート）。

指定ポートがフォワーディング ステートでエッジ ポートとして設定されていない場合、RSTP によって新しいルート情報と強制的に同期されると、その指定ポートはブロッキング ステートに移行します。一般的に RSTP がルート情報でポートを強制的に同期化し、ポートが上の条件を満たしていない場合、そのポート ステートはブロッキングに設定されます。

図 *28*：高速コンバージェンス中のイベントのシーケンス

deviceは、すべてのポートが同期化されたことを確認した後で、ルート ポートに対応する指定deviceに合意メッセージを送信します。ポイントツーポイント リンクで接続されたdevicesが

ポートの役割で合意すると、RSTP はポート ステートをフォワーディングにすぐに移行しま



す。

ブリッジ プロトコル データ ユニットの形式および処理

RSTP BPDU のフォーマットは、プロトコル バージョンが 2 に設定されている点を除き、IEEE 802.1D BPDU のフォーマットと同じです。新しい 1 バイトのバージョン 1 の Length フィールドは 0 に設定されます。これはバージョン 1 のプロトコルの情報がないことを示しています。

表 *46 : RSTP BPDU* フラグ

ビット	機能
0	トポロジーの変化（TC）
1	提案
2 ～ 3: 00 01 10 11	ポートの役割： 不明 代替ポート ルートポート 指定ポート
4	ラーニング
5	転送
6	合意
7	トポロジー変更確認応答（TCA）

送信側deviceは RSTP BPDU の提案フラグを設定し、そのLAN の指定deviceとして自分自身を提案します。提案メッセージのポートの役割は、常に DP に設定されます。

送信側deviceは、RSTP BPDU の合意フラグを設定して以前の提案を受け入れます。合意メッセージ内のポート ロールは、常にルート ポートに設定されます。

RSTP には個別のトポロジ変更通知（TCN）BPDU はありません。TC フラグが使用されて、TC が示されます。ただし、IEEE 802.1D devicesとの相互運用性を保つために、RSTP deviceは TCN BPDU の処理と生成を行います。

ラーニング フラグおよびフォワーディング フラグは、送信側ポートのステートに従って設定されます。

優位 BPDU 情報の処理

ポートに現在保存されているルート情報よりも優位のルート情報（小さいdevice ID、低いパスコストなど）をポートが受け取ると、RSTP は再構成を開始します。ポートが新しいルートポートとして提案されて選択されると、RSTP は強制的にその他すべてのポートを同期化します。

受信した BPDU が、提案フラグが設定されている RSTP BPDU である場合、deviceはその他すべてのポートが同期化されてから合意メッセージを送信します。BPDU が IEEE 802.1D BPDU の場合、deviceは提案フラグを設定せずに、そのポートの転送遅延タイマーを起動します。新しいルートポートでは、フォワーディング ステートに移行するために、2 倍の転送遅延時間が必要となります。

ポートで優位の情報が受信されたために、そのポートがバックアップ ポートまたは代替ポートになる場合、RSTP はそのポートをブロッキング ステートに設定し、合意メッセージは送信しません。DP は、転送遅延タイマーが失効するまで、提案フラグを設定して BPDU を送信し続け、転送遅延タイマーの失効時に、ポートはフォワーディング ステートに移行します。

下位 BPDU 情報の処理

指定ポートの役割を持つ下位 BPDU（そのポートに現在保存されている値より大きいdevice ID、高いパスコストなど）を指定ポートが受信した場合、その指定ポートはただちに現在の自身の情報で応答します。

トポロジの変更

ここでは、スパニングツリー トポロジの変更処理について、RSTP と IEEE 802.1D の相違を説明します。

- 検出：IEEE 802.1D では、どのようなブロッキング ステートとフォワーディング ステートとの間の移行でもトポロジの変更が発生しますが、RSTP でトポロジの変更が発生するのは、ブロッキング ステートからフォワーディング ステートに移行する場合だけです（トポロジの変更と見なされるのは、接続数が増加する場合だけです）。エッジポートにおけるステート変更は、TC の原因になりません。RSTP deviceは、TC を検出すると、TCN を受信したポートを除く、エッジ以外のすべてのポートで学習した情報を削除します。

- 通知：IEEE 802.1D は TCN BPDU を使用しますが、RSTP は使用しません。ただし、IEEE 802.1D との相互運用性を保つために、RSTP deviceは TCN BPDU の処理と生成を行います。
- 確認：RSTP deviceは、指定ポートで IEEE 802.1D deviceから TCN メッセージを受信した場合、TCA ビットが設定された IEEE 802.1D コンフィギュレーション BPDU で応答します。ただし、IEEE 802.1D deviceに接続されたルート ポートで TC 時間タイマー（IEEE 802.1D のトポロジ変更タイマーと同じ）がアクティブであり、TCA ビットが設定されたコンフィギュレーション BPDU が受信された場合、TC 時間タイマーはリセットされます。

  この処理は、IEEE 802.1D devicesをサポートする目的でのみ必要とされます。RSTP BPDU は TCA ビットが設定されていません。
- 伝播：RSTP deviceは、DP またはルート ポートを介して別のdeviceから TC メッセージを受信すると、エッジ以外のすべての DP、およびルート ポート（TC メッセージを受信したポートを除く）に変更を伝播します。deviceはこのようなすべてのポートで TC-while タイマーを開始し、そのポートで学習した情報を消去します。
- プロトコルの移行：IEEE 802.1D devicesとの下位互換性を保つため、RSTP は IEEE 802.1D コンフィギュレーション BPDU および TCN BPDU をポート単位で必要に応じて送信します。

  ポートが初期化されると、移行遅延タイマーが開始され（RSTP BPDU が送信される最低時間を指定）、RSTP BPDU が送信されます。このタイマーがアクティブである間、device はそのポートで受信したすべての BPDU を処理し、プロトコル タイプを無視します。

  deviceはポートの移行遅延タイマーが満了した後に IEEE 802.1D BPDU を受信した場合、IEEE 802.1D deviceに接続されていると想定し、IEEE 802.1D BPDU のみの使用を開始します。ただし、RSTP deviceが 1 つのポートで IEEE 802.1D BPDU を使用していて、タイマーが満了した後に RSTP BPDU を受信した場合、タイマーが再起動し、そのポートで RSTP BPDU の使用が開始されます。

プロトコル移行プロセス

MSTP が稼働しているdeviceは、IEEE 802.1D 準拠のレガシー devicesとの相互運用を可能にする組み込み型のプロトコル移行メカニズムをサポートします。このdeviceは、レガシー IEEE 802.1D コンフィギュレーション BPDU（プロトコルバージョンが 0 に設定されている BPDU）を受信すると、そのポート上では IEEE 802.1D BPDU のみを送信します。また、MST deviceは、レガシー BPDU、別のリージョンに関連付けられている MSTP BPDU（バージョン 3）、または RST BPDU（バージョン 2）を受信することによって、ポートがリージョンの境界に位置していることを検出できます。

ただし、deviceが IEEE 802.1D BPDU を受信していない場合は、自動的に MSTP モードに戻りません。これはレガシー deviceが指定deviceでない限り、レガシー deviceがリンクから削除されたかどうか検出できないためです。また、接続するdeviceがリージョンに加入していると、deviceはポートに境界の役割を割り当て続ける場合があります。

MSTP のデフォルト設定

表 *47 : MSTP* のデフォルト設定

機能	デフォルト設定
スパニングツリー モード	MSTP
スイッチ プライオリティ（CIST ポートごとに設定可能）	32768
スパニングツリー ポート プライオリティ（CIST ポート単位で設定可能）	128
スパニングツリー ポート コスト（CIST ポート単位で設定可能）	1000 Mb/s：20000 100 Mb/s：20000 10 Mb/s：20000
hello タイム	3 秒
転送遅延時間	20 秒
最大エージング タイム	20 秒
最大ホップ カウント	20 ホップ

MST と PVST+ の相互運用性について（PVST+ シミュレーション）

PVST+ シミュレーション機能は、MST と Rapid PVST+ との間にシームレスな相互運用性を実現します。ポート単位またはグローバルに有効化または無効化できます。PVST+ シミュレーションは、デフォルトでイネーブルになっています。

ただし、MST と Rapid PVST+ との接続を制御し、MST 対応ポートを Rapid PVST+ 対応ポートに誤って接続するのを防止することが必要な場合もあります。Rapid PVST+ はデフォルト STP モードのため、Rapid PVST+ がイネーブルな多数の接続が検出されることがあります。

この機能を無効にすると、スイッチは MST 領域と PVST+ 領域との対話を停止します。MST 対応ポートは、Rapid PVST+ 対応ポートに接続されたことを検出すると、PVST ピア不整合（ブロッキング）状態に移行します。このポートは、Shared Spanning Tree Protocol（SSTP）BPDU の受信を停止するまでは不整合状態を維持し、受信停止後は通常の STP 送信プロセスを再開します。

たとえば、PVST+ シミュレーションを無効にすることにより、正しく設定されていないスイッチと、STP モードが MSTP 以外であるネットワーク（デフォルト モードは PVST+）との接続を、防止することができます。

（同一リージョン内の）MST スイッチを PVST+ スイッチと対話させるよう設定する場合は、次の注意事項に従ってください。

- MST リージョン内のすべての VLAN に対するルートを設定します。次の例を参照してください。

```
Switch# show spanning-tree mst interface gigabitethernet 1/1
GigabitEthernet1/1 of MST00 is root forwarding
Edge port: no             (trunk) port guard : none    (default)
Link type: point-to-point (auto)  bpdu filter: disable (default)
Boundary : boundary       (PVST)  bpdu guard : disable (default)
Bpdus sent 10, received 310

Instance Role Sts Cost  Prio.Nbr    Vlans mapped
-------- ---- --- --------- -------- --------------------------------
0        Root FWD 20000 128.1       1-2,4-2999,4000-4094
3        Boun FWD 20000 128.1       3,3000-3999
```

MST スイッチに属する境界ポートは、PVST+ をシミュレートし、すべての VLAN に PVST+ BPDU を送信します。

PVST+ スイッチ上でループ ガードをイネーブルにすると、MST スイッチの設定が変更されたときに、ポートが loop-inconsistent ステートに変化する可能性があります。loop-inconsistent 状態を解消するには、PVST+ スイッチ上でループ ガードをいったん無効にしてから再有効化する必要があります。

- MST スイッチの PVST+ サイド内にある VLAN の一部またはすべてに対して、ルートを配置しないでください。境界の MST スイッチが指定ポート上の VLAN のすべてまたは一部に対する PVST+ BPDU を受信すると、ルート ガードによってそのポートがブロッキング ステートになります。

- PVST+ スイッチを 2 つの異なる MST リージョンに接続すると、PVST+ スイッチからのトポロジ変更が最初の MST リージョンから先へ伝達されません。この場合、トポロジ変更は VLAN がマッピングされているインスタンスで伝播されるだけです。トポロジ変更は最初の MST リージョンに対してローカルのままで、その他のリージョンの Cisco Access Manager（CAM）エントリはフラッシュされません。他の MST リージョンにもトポロジ変更が認識されるようにするには、IST に VLAN をマッピングするか、またはアクセス リンクを介して 2 つのリージョンに PVST+ スイッチを接続します。

- PVST+ シミュレーションを無効にすると、ポートがすでに他の不整合状態にある間、PVST+ ピア不整合も起こる可能性があるので、注意してください。たとえば、すべての STP インスタンスのルート ブリッジは、MST または Rapid PVST+ のどちらかの側に属している必要があります。すべての STP インスタンスのルート ブリッジがどちらか一方の側に属していないと、ポートは PVST+ シミュレーション不整合状態になります。

（注） すべての STP インスタンスのルート ブリッジを、MST 側に配置することを推奨します。

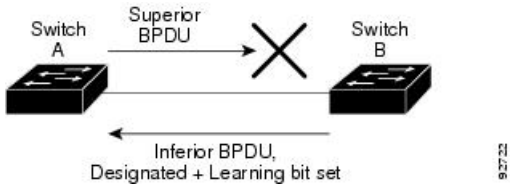
単方向リンク障害の検出について

IEEE 802.1D-2004 RSTP および IEEE 802.1Q-2005 MSTP 標準には単方向リンク障害を検出する解決メカニズムが含まれており、ユーザによる設定は必要ありません。

スイッチにより、受信するBPDUのポートのロールおよびステートの一貫性がチェックされ、ブリッジングループを発生させる可能性のある単方向リンク障害が検出されます。指定ポートが矛盾を検出するとロールは維持されますが、状態は廃棄（ブロッキング）ステートに戻ります。これは、接続に矛盾が生じた場合、ブリッジングループを開始するよりも接続を中断する方が好ましいためです。

たとえば、次の図では、スイッチ A がルート ブリッジ スイッチで、スイッチ B が指定ポートです。スイッチ A からの BPDU は、スイッチ B に向かうリンク上で失われます。

図 *29 :* 単一方向リンク障害の検出

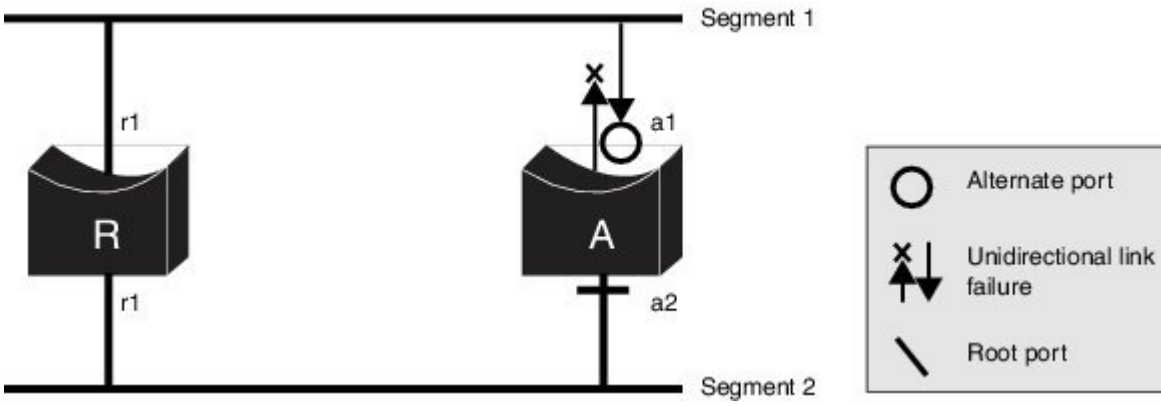


Rapid PVST+（802.1w）および MST BPDU には送信ポートのロールとステートが含まれるので、ロールがルート ブリッジではなく指定ポートであるという理由からスイッチ B が送信対象の優位 BPDU に反応しないことを、スイッチ A は（下位 BPDU から）検出します。 結果として、スイッチAは自身のポートをブロックし（またはブロックを維持して）、ブリッジ処理のループを回避します。結果として、スイッチAは自身のポートをブロックし（またはブロックを維持して）、ブリッジ処理のループを回避します。

解決メカニズムに関して、次のガイドラインと制約事項に留意してください。

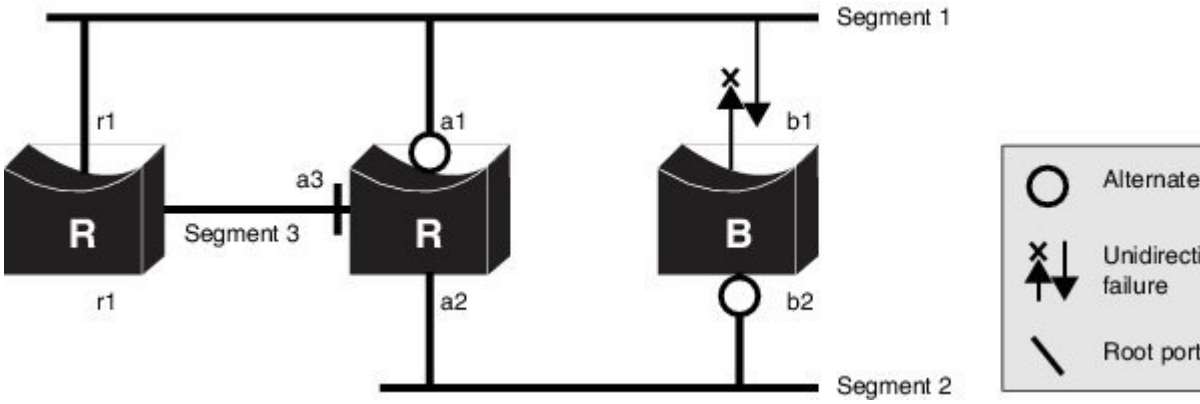
- RSTP またはMST を実行するスイッチ上でのみ機能します（解決メカニズムは、BPDU を開始するポートのロールとステートを読み取る必要があります）。
- 接続が失われる原因になることがあります。たとえば、次の図のブリッジ A は、ルート ポートとして選択したポートでの送信ができません。この状況の結果として、接続が失われます（r1 と r2 は指定ポート、a1 はルート ポート、a2 は代替ポートです。A と R の間には 1 方向の接続しかありません）。

図 *30 :* 接続の消失



- 共有セグメントで永久ブリッジングループが発生する原因になることがあります。たとえば、次の図で、ブリッジ R の優先順位が最も高く、ポート b1 は共有セグメント 1 からのトラフィックを受信できずセグメント 1 の下位指定情報を送信していると仮定します。r1 と a1 はどちらもこの不整合を検出できます。ただし、現在の解決メカニズムでは、廃棄に戻るのは r1 のみであり、ルート ポート a1 は永久ループを開きます。ただし、この問題は、ポイントツーポイント リンクによって接続されたレイヤ 2 スイッチド ネットワークでは発生しません。

図 *31* : 共有セグメントのブリッジング ループ



MSTP 機能の設定方法

MST リージョン設定の指定と MSTP のイネーブル化

2 つ以上のスイッチを同じ MST リージョンに設定するには、その 2 つのスイッチに同じ VLAN/インスタンス マッピング、同じコンフィギュレーション リビジョン番号、同じ名前を設定しなければなりません。

リージョンには、MST 設定が同一である、1 つ以上のメンバーを含めることができます。各メンバーでは、RSTP BPDU を処理できる必要があります。ネットワーク内の MST リージョンの数に制限はありませんが、各リージョンは最大 65 のスパニングツリー インスタンスのみをサポートできます。VLAN には、一度に 1 つのスパニングツリー インスタンスのみ割り当てることができます。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst configuration**
4. **instance** *instance-id* **vlan** *vlan-range*
5. **name** *name*
6. **revision** *version*

7. **show pending**
8. **exit**
9. **spanning-tree mode mst**
10. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst configuration** 例： `スイッチ(config)# spanning-tree mst configuration`	MST コンフィギュレーション モードを開始します。
ステップ 4	**instance** *instance-id* **vlan** *vlan-range* 例： `スイッチ(config-mst)# instance 1 vlan 10-20`	VLAN を MSTI にマップします。 • *instance-id* に指定できる範囲は、0 ～ 4094 です。 • **vlan** *vlan-range* に指定できる範囲は、1 ～ 4094 です。 VLAN を MSTI にマップする場合、マッピングは増加され、コマンドに指定した VLAN は、以前マッピングした VLAN に追加されるか、そこから削除されます。 VLAN の範囲を指定するには、ハイフンを使用します。たとえば **instance 1 vlan 1-63** では、VLAN 1 ～ 63 が MSTI 1 にマップされます。 VLAN を列挙して指定する場合は、カンマを使用します。たとえば **instance 1 vlan 10, 20, 30** と指定すると、VLAN 10、20、30 が MST インスタンス 1 にマッピングされます。

	コマンドまたはアクション	目的
ステップ 5	**name** *name* 例： `スイッチ(config-mst)# name region1`	コンフィギュレーション名を指定します。*name* 文字列の最大の長さは 32 文字であり、大文字と小文字が区別されます。
ステップ 6	**revision** *version* 例： `スイッチ(config-mst)# revision 1`	設定リビジョン番号を指定します。指定できる範囲は 0 ～ 65535 です。
ステップ 7	**show pending** 例： `スイッチ(config-mst)# show pending`	保留中の設定を表示し、設定を確認します。
ステップ 8	**exit** 例： `スイッチ(config-mst)# exit`	すべての変更を適用し、グローバル コンフィギュレーション モードに戻ります。
ステップ 9	**spanning-tree mode mst** 例： `スイッチ(config)# spanning-tree mode mst`	MSTP をイネーブルにします。RSTP もイネーブルになります。 スパニングツリー モードを変更すると、すべてのスパニングツリー インスタンスは以前のモードであるため停止し、新しいモードで再起動するので、トラフィックを中断させる可能性があります。 MSTP と PVST+ または MSTP と Rapid PVST+ を同時に実行することはできません。
ステップ 10	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

ルート デバイスの設定

この手順は任意です。

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

指定された MST インスタンス ID も把握する必要があります。この例のステップ 2 では、インスタンス ID として 0 を使用します。これは「関連項目」で示されている手順によって設定されたインスタンス ID が 0 であるためです。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst** *instance-id* **root primary**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst** *instance-id* **root primary** 例： `スイッチ(config)# spanning-tree mst 0 root primary`	ルート deviceとしてdeviceを設定します。 • *instance-id* には、単一のインスタンス、ハイフンで区切られた範囲のインスタンス、またはカンマで区切られた一連のインスタンスを指定できます。指定できる範囲は 0 ～ 4094 です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

セカンダリ ルートの設定デバイス

拡張システム ID をサポートするdeviceをセカンダリ ルートとして設定する場合、device プライオリティはデフォルト値（32768）から 28672 に修正されます。プライマリ ルート deviceで障害が発生した場合は、このdeviceが指定インスタンスのルート deviceになる可能性があります。ここでは、その他のネットワーク devicesが、デフォルトのdevice プライオリティの 32768 を使用しているためにルート deviceになる可能性が低いことが前提となっています。

このコマンドを複数のdeviceに対して実行すると、複数のバックアップルート devicesを設定できます。**spanning-tree mst** *instance-id* **root primary** グローバル コンフィギュレーション コマンドでプライマリルート device を設定したときと同じネットワーク直径および hello タイム値を使用してください。

この手順は任意です。

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

指定された MST インスタンス ID も把握する必要があります。この例では、インスタンス ID として 0 を使用します。これは「関連項目」で示されている手順によって設定されたインスタンス ID が 0 であるためです。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst** *instance-id* **root secondary**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst** *instance-id* **root secondary** 例： `スイッチ(config)# spanning-tree mst 0 root secondary`	セカンダリ ルート deviceとしてdeviceを設定します。 • *instance-id* には、単一のインスタンス、ハイフンで区切られた範囲のインスタンス、またはカンマで区切られた一連のインスタンスを指定できます。指定できる範囲は 0 ～ 4094 です。
ステップ 4	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	

ポート プライオリティの設定

ループが発生した場合、MSTPはポートプライオリティを使用して、フォワーディング ステートにするインターフェイスを選択します。最初に選択されるインターフェイスには高いプライオリティ値（小さい数値）を割り当て、最後に選択されるインターフェイスには低いプライオリティ値（高い数値）を割り当てることができます。すべてのインターフェイスに同じプライオリティ値が与えられている場合、MSTP はインターフェイス番号が最小のインターフェイスをフォワーディング ステートにし、他のインターフェイスをブロックします。

（注） device が device スタックのメンバーの場合、**spanning-tree mst** [*instance-id*] **port-priority** *priority* インターフェイス コンフィギュレーション コマンドの代わりに、**spanning-tree mst** [*instance-id*] **cost** *cost* インターフェイス コンフィギュレーション コマンドを使用し、フォワーディングステートにするインターフェイスを選択する必要があります。最初に選択させたいポートには、より小さいコスト値を割り当て、最後に選択させたいポートには、より大きいコスト値を割り当てることができます。詳細については、関連項目の下に表示されるパス コストのトピックを参照してください。

この手順は任意です。

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

指定されたMSTインスタンスIDと使用されるインターフェイスも把握する必要があります。この例では、インスタンス ID として 0 を使用し、インターフェイスとして GigabitEthernet0/1 を使用します。これは「関連トピック」で示されている手順によってインスタンス ID とインターフェイスがそのように設定されているためです。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst** *instance-id* **port-priority** *priority*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**spanning-tree mst** *instance-id* **port-priority** *priority* 例： スイッチ(config-if)# **spanning-tree mst 0 port-priority 64**	ポート プライオリティを設定します。 • *instance-id* には、単一のインスタンス、ハイフンで区切られた範囲のインスタンス、またはカンマで区切られた一連のインスタンスを指定できます。指定できる範囲は 0 ～ 4094 です。 • *priority* 値の範囲は 0 ～ 240 で、16 ずつ増加します。デフォルト値は 128 です。値が小さいほど、プライオリティが高くなります。 使用可能な値は、0、16、32、48、64、80、96、112、128、144、160、176、192、208、224、240 だけです。その他の値はすべて拒否されます。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

show spanning-tree mst interface *interface-id* 特権 EXEC コマンドで情報が表示されるのは、ポートがリンクアップ動作可能の状態にある場合に限られます。そうでない場合は、**show running-config interface** 特権 EXEC コマンドを使用して設定を確認してください。

パス コストの設定

MSTPパス コストのデフォルト値は、インターフェイスのメディア速度に基づきます。ループが発生した場合、MSTPはコストを使用して、フォワーディング ステートにするインターフェイスを選択します。最初に選択されるインターフェイスには低いコスト値を割り当て、最後に選択されるインターフェイスには高いコスト値を割り当てることができます。すべてのインターフェイスに同じコスト値が与えられている場合、MSTPはインターフェイス番号が最小のインターフェイスをフォワーディング ステートにし、他のインターフェイスをブロックします。

この手順は任意です。

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

指定されたMSTインスタンスIDと使用されるインターフェイスも把握する必要があります。この例では、インスタンスIDとして0を使用し、インターフェイスとしてGigabitEthernet1/0/1を使用します。これは「関連トピック」で示されている手順によってインスタンス IDとインターフェイスがそのように設定されているためです。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst** *instance-id* **cost** *cost*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例：	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。有効なインターフェイスには、物理ポートとポート

	コマンドまたはアクション	目的
	スイッチ(config)# **interface gigabitethernet 1/0/1**	チャネル論理インターフェイスがあります。指定できるポートチャネルの範囲は 1 ～ 48 です。
ステップ 4	**spanning-tree mst** *instance-id* **cost** *cost* 例： スイッチ(config-if)# **spanning-tree mst 0 cost 17031970**	コストを設定します。 ループが発生した場合、MSTP はパス コストを使用して、フォワーディング ステートにするインターフェイスを選択します。低いパス コストは高速送信を表します。 • *instance-id* には、単一のインスタンス、ハイフンで区切られた範囲のインスタンス、またはカンマで区切られた一連のインスタンスを指定できます。指定できる範囲は 0 ～ 4094 です。 • *cost* の範囲は 1 ～ 200000000 です。デフォルト値はインターフェイスのメディア速度から派生します。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

show spanning-tree mst interface *interface-id* 特権 EXEC コマンドによって表示されるのは、リンクアップ動作可能状態のポートの情報だけです。そうでない場合は、**show running-config** 特権 EXEC コマンドを使用して設定を確認してください。

デバイス プライオリティの設定

deviceのプライオリティを変更すると、スタンドアロンdeviceまたはスタック内のdeviceであるかどうかに関係なく、ルートdeviceとして選択される可能性が高くなります。

（注） このコマンドの使用には注意してください。通常のネットワーク設定では、**spanning-tree mst** *instance-id* **root primary** および **spanning-tree mst** *instance-id* **root secondary** グローバル コンフィギュレーション コマンドを使用して、deviceをルートまたはセカンダリルートdeviceとして指定することをお勧めします。これらのコマンドが動作しない場合にのみdeviceプライオリティを変更する必要があります。

この手順は任意です。

始める前に

マルチスパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

使用する指定されたMSTインスタンスIDも把握する必要があります。この例では、インスタンス ID として 0 を使用します。これは「関連項目」で示されている手順によって設定されたインスタンス ID が 0 であるためです。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst** *instance-id* **priority** *priority*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst** *instance-id* **priority** *priority* 例： `スイッチ(config)# spanning-tree mst 0 priority 40960`	deviceのプライオリティを設定します。 • *instance-id* には、単一のインスタンス、ハイフンで区切られた範囲のインスタンス、またはカンマで区切られた一連のインスタンスを指定できます。指定できる範囲は 0 ～ 4094 です。 • *priority* の範囲は 0 ～ 61440 で、4096 ずつ増加します。デフォルトは 32768 です。この値が低いほど、deviceがルートdeviceとして選択される可能性が高くなります。 使用可能な値は、0、4096、8192、12288、16384、20480、24576、28672、32768、36864、40960、45056、49152、53248、57344、61440 です。これらは唯一の許容値です。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

hello タイムの設定

hello タイムはルート deviceによって設定メッセージが生成されて送信される時間の間隔です。

この手順は任意です。

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst hello-time** *seconds*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst hello-time** *seconds* 例： スイッチ(config)# **spanning-tree mst hello-time 4**	すべての MST インスタンスについて、hello タイムを設定します。hello タイムはルート deviceによって設定メッセージが生成されて送信される時間の間隔です。このメッセージは、deviceが活動中であることを表します。

	コマンドまたはアクション	目的
		seconds に指定できる範囲は 1 ～ 10 です。デフォルトは 3 です。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

転送遅延時間の設定

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst forward-time** *seconds*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst forward-time** *seconds* 例： スイッチ(config)# **spanning-tree mst forward-time 25**	すべての MST インスタンスについて、転送時間を設定します。転送遅延時間は、スパニングツリー ラーニング ステートおよびリスニング ステートからフォワーディング ステートに移行するまでに、ポートが待機する秒数です。

	コマンドまたはアクション	目的
		seconds に指定できる範囲は 4 ～ 30 です。デフォルトは 20 です。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

最大エージング タイムの設定

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-age** *seconds*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst max-age** *seconds* 例： スイッチ(config)# **spanning-tree mst max-age 40**	すべての MST インスタンスについて、最大経過時間を設定します。最大エージング タイムは、device が再設定を試す前にスパニングツリー設定メッセージを受信せずに待機する秒数です。 *seconds* に指定できる範囲は 6 ～ 40 です。デフォルトは 20 です。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

最大ホップ カウントの設定

この手順は任意です。

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-hops** *hop-count*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst max-hops** *hop-count* 例： スイッチ(config)# **spanning-tree mst max-hops 25**	BPDU を廃棄してポート用に保持していた情報を期限切れにするまでの、リージョンでのホップ数を設定します。 *hop-count* に指定できる範囲は 1 ～ 255 です。デフォルト値は 20 です。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

高速移行を確実にするためのリンク タイプの指定

ポイントツーポイント リンクでポート間を接続し、ローカル ポートが DP になると、RSTP は提案と合意のハンドシェークを使用して別のポートと高速移行をネゴシエーションし、ループがないトポロジを保証します。

デフォルトの場合、リンク タイプはインターフェイスのデュプレックス モードから制御されます。全二重ポートはポイントツーポイント接続、半二重ポートは共有接続と見なされます。MSTP を実行しているリモート deviceの単一ポートに、半二重リンクを物理的にポイントツーポイントで接続した場合は、リンク タイプのデフォルト設定を無効にして、フォワーディング ステートへの高速移行をイネーブルにすることができます。

この手順は任意です。

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

指定された MST インスタンス ID と使用されるインターフェイスも把握する必要があります。この例では、インスタンス ID として 0 を使用し、インターフェイスとして GigabitEthernet1/0/1 を使用します。これは「関連トピック」で示されている手順によってインスタンス ID とインターフェイスがそのように設定されているためです。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree link-type point-to-point**
5. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。有効なインターフェイスには、物理ポート、VLAN、およびポート チャネル論理インターフェイスがあります。VLAN ID の範囲は 1 ～ 4094 です。指定できるポートチャネルの範囲は 1 ～ 48 です。
ステップ 4	**spanning-tree link-type point-to-point** 例： スイッチ(config-if)# **spanning-tree link-type point-to-point**	ポートのリンク タイプがポイントツーポイントであることを指定します。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

ネイバー タイプの設定

トポロジには、先行標準に準拠したデバイスと IEEE 802.1s 標準準拠のデバイスの両方を加えることができます。デフォルトの場合、ポートは準規格デバイスを自動的に検出できますが、規格 BPDU および準規格 BPDU の両方を受信できます。デバイスとそのネイバーの間に不一致がある場合は、CIST だけがインターフェイスで動作します。

準規格 BPDU だけを送信するようにポートを設定できます。先行標準のフラグは、ポートが STP 互換モードにある場合でも、すべての **show** コマンドで表示されます。

この手順は任意です。

始める前に

マルチ スパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst pre-standard**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。有効なインターフェイスには、物理ポートが含まれます。
ステップ 4	**spanning-tree mst pre-standard** 例： スイッチ(config-if)# **spanning-tree mst pre-standard**	ポートが準規格 BPDU だけを送信できることを指定します。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

プロトコルの移行プロセスの再開

この手順では、プロトコル移行プロセスを再開し、ネイバー devicesとの再ネゴシエーションを強制します。また、deviceを MST モードに戻します。これは、IEEE 802.1D BPDU の受信後に deviceがそれらを受信しない場合に必要です。

deviceでプロトコルの移行プロセスを再開する（隣接するdevicesで再ネゴシエーションを強制的に行う）手順については、これらの手順に従ってください。

始める前に

マルチスパニングツリー（MST）が、deviceで指定されて有効になっている必要があります。詳細については、関連項目を参照してください。

コマンドのインターフェイス バージョンを使用する場合は、使用する MST インターフェイスが分かっている必要があります。この例では、インターフェイスとして GigabitEthernet1/0/1 を使用します。それが「関連項目」で示されている手順によって設定されたインターフェイスであるからです。

手順の概要

1. **enable**
2. 次のいずれかのコマンドを入力します。
   - **clear spanning-tree detected-protocols**
   - **clear spanning-tree detected-protocols interface** *interface-id*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	次のいずれかのコマンドを入力します。 • **clear spanning-tree detected-protocols** • **clear spanning-tree detected-protocols interface** *interface-id* 例： `スイッチ# clear spanning-tree detected-protocols` または `スイッチ# clear spanning-tree detected-protocols interface gigabitethernet 1/0/1`	deviceが MSTP モードに戻り、プロトコルの移行プロセスが再開されます。

次のタスク

この手順は、deviceでさらにレガシー IEEE 802.1D コンフィギュレーション BPDU（プロトコル バージョンが 0 に設定された BPDU）を受信する場合に、繰り返しが必要なことがあります。

PVST+ シミュレーションの設定

PVST+シミュレーションは、デフォルトでイネーブルになっています。つまり、すべてのポートが、Rapid PVST+ モードで動作する接続先デバイスと自動的に相互運用します。機能を無効にしてから再設定したい場合は、次の作業を参照してください。

PVST+ シミュレーションをグローバルに有効にするには、次の作業を行います。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree mst simulate pvst global**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree mst simulate pvst global** 例： `スイッチ(config)# spanning-tree mst simulate pvst global`	PVST+シミュレーションをグローバルに有効化します。 Rapid PVST+ モードで動作する接続先デバイスとスイッチとの自動的な相互運用を回避するには、コマンドの **no** バージョンを入力します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

ポート上での PVST+ シミュレーションの有効化

特定のポート上で PVST+ シミュレーションを有効化するには、次の作業を行います。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst simulate pvst**
5. **end**
6. **show spanning-tree summary**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gi1/0/1`	設定するポートを選択します。
ステップ 4	**spanning-tree mst simulate pvst** 例： `スイッチ(config-if)# spanning-tree mst simulate pvst`	特定のインターフェイスで PVST+ シミュレーションを有効化します。 指定したインターフェイスと MST を実行していない接続スイッチとの自動的な相互運用を回避するには、**spanning-tree mst simulate pvst disable** コマンドを入力します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show spanning-tree summary** 例： `スイッチ# show spanning-tree summary`	設定を確認します。

例

例：PVST+ シミュレーション

次の例は、Rapid PVST+ を実行している接続スイッチと自動的に相互運用することを防止するようにスイッチを設定する方法を示しています。

```
Switch# configure terminal
Switch(config)# no spanning-tree mst simulate pvst global
```

次に、Rapid PVST+ を実行している接続先デバイスとポートが自動的に相互運用しないようにする例を示します。

```
Switch(config)# interface1/0/1
Switch(config-if)# spanning-tree mst simulate pvst disable
```

次の出力例は、PVST+ シミュレーション無効時にポートで SSTP BPDU を受信した場合に受け取るシステム メッセージを示しています。

```
Message
SPANTREE_PVST_PEER_BLOCK: PVST BPDU detected on port %s [port number].

Severity
Critical

Explanation
A PVST+ peer was detected on the specified interface on the switch.
PVST+ simulation feature is disabled, as a result of which the interface
 was moved to the spanning tree
Blocking state.

Action
Identify the PVST+ switch from the network which might be configured
incorrectly.
```

次の出力例は、インターフェイスのピア不整合が解消したときに受け取るシステムメッセージを示しています。

```
Message
SPANTREE_PVST_PEER_UNBLOCK: Unblocking port %s [port number].

Severity
Critical

Explanation
The interface specified in the error message has been restored to normal
 spanning tree state.
```

```
Action
None.
```

この例は、ポート **1/0/1** を設定して PVST+ シミュレーションを無効にし、そのポートがピアタイプ不整合状態にあるときの、スパニング ツリー ステータスを示しています。

```
Switch# show spanning-tree
VLAN0010
  Spanning tree enabled protocol mstp
  Root ID Priority  32778
          Address   0002.172c.f400
          This bridge is the root
          Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)
            Address  0002.172c.f400
          Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
          Aging Time 300
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------
Gi1/0/1           Desg BKN*4         128.270 P2p *PVST_Peer_Inc
```

次に、MSTP モードで PVST+ シミュレーションが有効である場合のスパニング ツリーの概要の例を示します。

```
Switch# show spanning-tree summary
Switch is in mst mode (IEEE Standard)
Root bridge for: MST0
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is long
PVST Simulation Default is enabled
Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
MST0                          2         0        0          0
 2
---------------------- -------- --------- -------- ---------- ----------
1 mst                         2         0        0          0
 2
```

次に、STP モードで PVST+ シミュレーションが無効である場合のスパニング ツリーの概要の例を示します。

```
Switch# show spanning-tree summary
Switch is in mst mode (IEEE Standard)
Root bridge for: MST0
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
```

```
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is long
PVST Simulation Default is disabled
Name                 Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
MST0                          2         0        0          0
 2
---------------------- -------- --------- -------- ---------- ----------
1 mst                         2         0        0          0
 2
```

次に、スイッチが MSTP モードでない場合、つまりスイッチが PVST または Rapid-PVST モードの場合のスパニング ツリーの概要の例を示します。出力文字列は現在の STP モードを表示します。

```
Switch# show spanning-tree summary
Switch is in rapid-pvst mode
Root bridge for: VLAN0001, VLAN2001-VLAN2002
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is short
PVST Simulation Default is enabled but inactive in rapid-pvst mode
Name                 Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
VLAN0001                      2         0        0          0
 2
VLAN2001                      2         0        0          0
 2
VLAN2002                      2         0        0          0
 2
---------------------- -------- --------- -------- ---------- ----------
3 vlans                       6         0        0          0
 6
```

この例は、PVST+シミュレーションがグローバルに有効な場合（デフォルト設定）のインターフェイスの詳細を示しています。

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
```

```
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   PVST Simulation is enabled by default
   BPDU: sent 132, received 1
```

この例は、PVST+シミュレーションがグローバルに無効な場合のインターフェイスの詳細を示しています。

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   PVST Simulation is disabled by default
   BPDU: sent 132, received 1
```

この例は、PVST+シミュレーションがポートで明示的に有効化されている場合のインターフェイスの詳細を示しています。

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   PVST Simulation is enabled
   BPDU: sent 132, received 1
```

この例は、ポートで PVST+ シミュレーション機能が無効になっており PVST ピア不整合が検出された場合のインターフェイスの詳細を示しています。

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is broken (PVST Peer Inconsistent)
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   PVST Simulation is disabled
   BPDU: sent 132, received 1
```

例：単方向リンク障害の検出

この例は、ポート **1/0/1detail** を設定して PVST+ シミュレーションを無効にし、ポートが現在ピア タイプ不整合状態にあるときの、スパニング ツリー ステータスを示しています。

```
Switch# show spanning-tree
VLAN0010
```

```
  Spanning tree enabled protocol rstp
  Root ID    Priority 32778
             Address  0002.172c.f400
             This bridge is the root
             Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID  Priority 32778 (priority 32768 sys-id-ext 10)
             Address  0002.172c.f400
             Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
             Aging Time 300

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------
Gi1/0/1          Desg BKN 4         128.270  P2p Dispute
```

この例は、競合する状態が検出された場合のインターフェイスの詳細を示しています。

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is designated blocking (dispute)
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   BPDU: sent 132, received 1
```

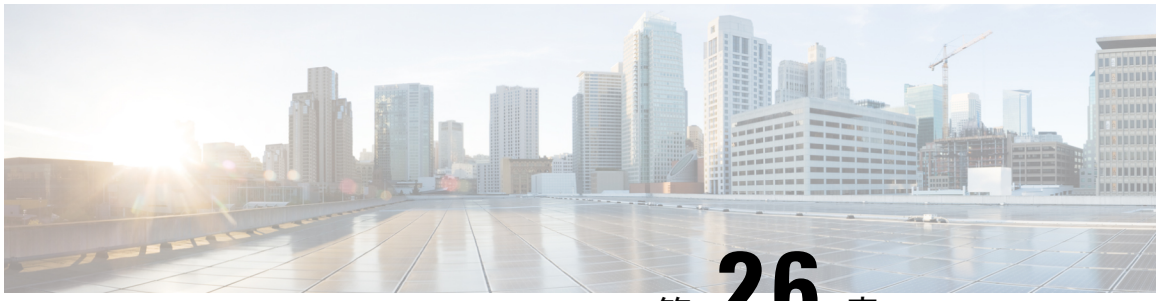
MST の設定およびステータスのモニタリング

表 *48 : MST* ステータスを表示するコマンド

show spanning-tree mst configuration	MST リージョンの設定を表示します。
show spanning-tree mst configuration digest	現在の MSTCI に含まれる MD5 ダイジェストを表示します。
show spanning-tree mst *instance-id*	指定インスタンスの MST 情報を表示します。 （注） このコマンドは、ポートがリンクアップ動作可能状態の場合にのみ情報を表示します。
show spanning-tree mst interface *interface-id*	指定インターフェイスの MST 情報を表示します。

MSTP の機能情報

リリース	変更内容
Cisco IOS Release 15.2(3)E	この機能が導入されました。



第 26 章

オプションのスパニングツリー機能の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

オプションのスパニングツリー機能の制約事項

- PortFast は、スパニング ツリーがコンバージェンスするまでにインターフェイスが待機する時間を最短にするため、これはエンドステーションに接続されているインターフェイスで使用される場合のみ有用です。他のスイッチに接続するインターフェイスで PortFast をイネーブルにすると、スパニングツリーのループが生じることがあります。

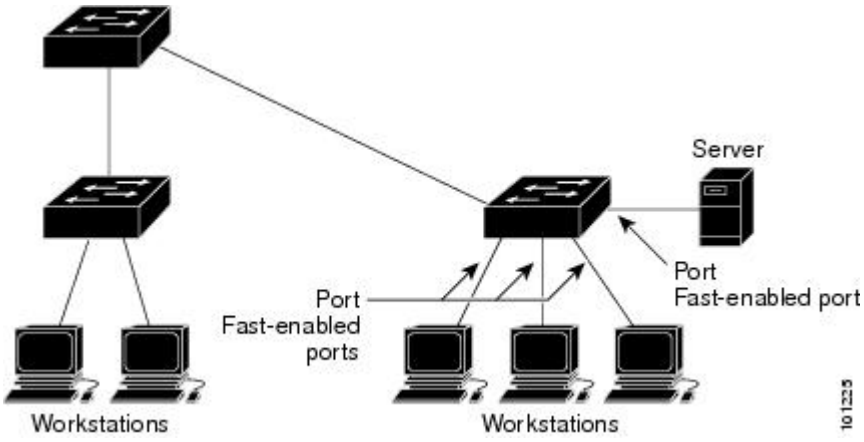
オプションのスパニングツリー機能について

PortFast

PortFast機能を使用すると、アクセス ポートまたはトランク ポートとして設定されているインターフェイスが、リスニング ステートおよびラーニング ステートを経由せずに、ブロッキング ステートから直接フォワーディング ステートに移行します。

図 *32 : PortFast* が有効なインターフェイス

1 台のワークステーションまたはサーバに接続されているインターフェイス上で PortFast を使用すると、スパニングツリーが収束するのを待たずにデバイスをすぐにネットワークに接続できます。



1 台のワークステーションまたはサーバに接続されたインターフェイスがブリッジプロトコル データ ユニット（BPDU）を受信しないようにする必要があります。スイッチを再起動すると、PortFast が有効に設定されているインターフェイスは通常のスパニングツリー ステータスの遷移をたどります。

インターフェイスまたはすべての非トランク ポートで有効にして、この機能を有効にできます。

BPDU ガード

ブリッジプロトコルデータユニット（BPDU）ガード機能はスイッチ上でグローバルにイネーブルにすることも、ポート単位でイネーブルにすることもできます。ただし、これらの動作は次の点で異なります。

PortFast エッジ対応ポート上でグローバル レベルで BPDU ガードをイネーブルにすると、スパニング ツリーは、BPDU が受信されると、PortFast エッジ動作ステートのポートをシャット ダウンします。有効な設定では、PortFast エッジ対応ポートは BPDU を受信しません。PortFast エッジ対応ポートが BPDU を受信した場合は、許可されていないデバイスの接続などの無効な設定が存在することを示しており、BPDU ガード機能によってポートは error-disabled ステート

になります。この状態になると、スイッチは違反が発生したポート全体をシャットダウンします。

PortFast エッジ機能をイネーブルにせずにインターフェイス レベルでポート上の BPDU ガードをイネーブルにした場合、ポートが BPDU を受信すると、error-disabled ステートになります。

インターフェイスを手動で再び動作させなければならない場合、無効な設定を防ぐには、BPDU ガード機能が役に立ちます。サービスプロバイダー ネットワーク内でアクセス ポートがスパニングツリーに参加しないようにするには、BPDU ガード機能を使用します。

BPDU フィルタリング

BPDU フィルタリング機能はスイッチ上でグローバルにイネーブルにすることも、インターフェイス単位でイネーブルにすることもできます。ただし、これらの動作は次の点で異なります。

グローバル レベルでは、PortFast エッジ対応インターフェイスで BPDU フィルタリングをイネーブルにすると、PortFast エッジ動作ステートにあるインターフェイスでの BPDU の送受信が防止されます。ただし、リンクが確立してからスイッチが発信 BPDU のフィルタリングを開始するまでの間に、このインターフェイスから BPDU がいくつか送信されます。これらのインターフェイスに接続されたホストが BPDU を受信しないようにするには、スイッチ上で BPDU フィルタリングをグローバルにイネーブルにする必要があります。PortFast エッジ対応インターフェイスでは、BPDU を受信すると、PortFast エッジ動作ステートが解除され、BPDU フィルタリングがディセーブルになります。

PortFast エッジ機能をイネーブルにせずに、インターフェイスで BPDU フィルタリングをイネーブルにすると、インターフェイスでの BPDU の送受信が防止されます。

注意 BPDU フィルタリングを特定のインターフェイス上でイネーブルにすることは、そのインターフェイス上でスパニングツリーをディセーブルにすることと同じであり、スパニング ツリー ループが発生することがあります。

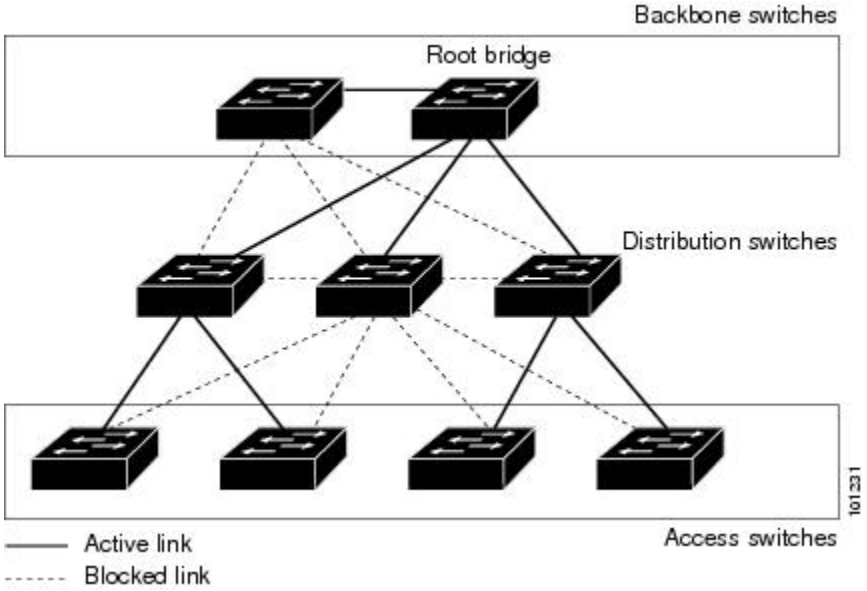
スイッチ全体または 1 つのインターフェイスで BPDU フィルタリング機能をイネーブルにできます。

UplinkFast

図 *33* : 階層型ネットワークのスイッチ

階層型ネットワークに配置されたスイッチは、バックボーン スイッチ、ディストリビューション スイッチ、およびアクセス スイッチに分類できます。この複雑なネットワークには、ディストリビューション スイッチとアクセス スイッチがあり、ループを防止するために、スパニ

ング ツリーがブロックする冗長リンクが少なくとも 1 つあります。



スイッチの接続が切断されると、スイッチはスパニングツリーが新しいルートポートを選択すると同時に代替パスの使用を開始します。リンクやスイッチに障害が発生した場合、またはスパニングツリーが UplinkFast の有効化によって自動的に再設定された場合に、新しいルートポートを短時間で選択できます。ルートポートは、通常のスパニングツリー手順とは異なり、リスニング ステートおよびラーニング ステートを経由せず、ただちにフォワーディング ステートに移行します。

スパニングツリーが新規ルートポートを再設定すると、他のインターフェイスはネットワークにマルチキャストパケットをフラッディングし、インターフェイス上で学習した各アドレスにパケットを送信します。max-update-rate パラメータの値を小さくすることで、これらのマルチキャストトラフィックのバーストを制限できます（このパラメータはデフォルトで毎秒 150 パケットです）。ただし、0 を入力すると、ステーション学習フレームが生成されないので、接続切断後スパニングツリー トポロジがコンバージェンスする速度が遅くなります。

（注） UplinkFast は、ネットワークのアクセスまたはエッジに位置する、ワイヤリング クローゼットのスイッチで非常に有効です。バックボーン デバイスには適していません。他のアプリケーションにこの機能を使用しても、有効とは限りません。

UplinkFast は、直接リンク障害発生後に高速コンバージェンスを行い、アップリンク グループを使用して、冗長レイヤ 2 リンク間でロードバランシングを実行します。アップリンク グループは、（VLAN ごとの）レイヤ 2 インターフェイスの集合であり、いかなるときも、その中の 1 つのインターフェイスだけが転送を行います。つまり、アップリンク グループは、（転送を行う）ルートポートと、（セルフループを行うポートを除く）ブロックされたポートの集合で構成されます。アップリンク グループは、転送中のリンクで障害が起きた場合に代替パスを提供します。

図 *34* : 直接リンク障害が発生する前の *UplinkFast* の例

このトポロジにはリンク障害がありません。ルート スイッチであるスイッチ A は、リンク L1 を介してスイッチ B に、リンク L2 を介してスイッチ C に直接接続されています。スイッチ B に直接接続されているスイッチ C のレイヤ 2 インターフェイスは、ブロッキング ステートです。

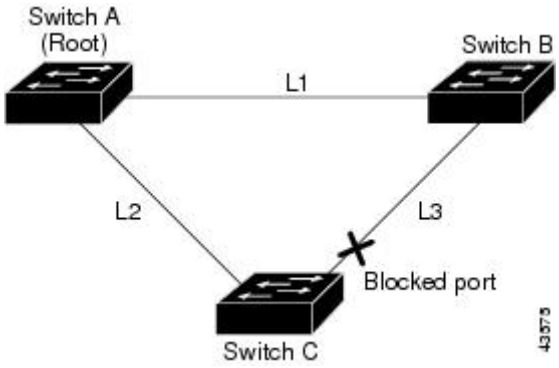
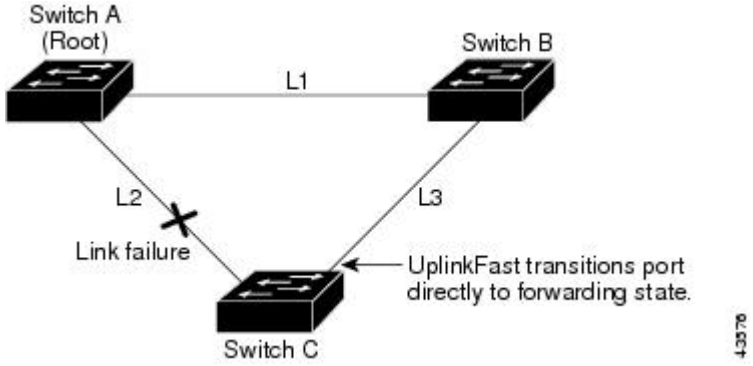


図 *35* : 直接リンク障害が発生したあとの *UplinkFast* の例

スイッチ C が、ルート ポートの現在のアクティブ リンクである L2 でリンク障害（直接リンク障害）を検出すると、UplinkFast がスイッチ C でブロックされていたインターフェイスのブロックを解除し、リスニング ステートおよびラーニング ステートを経由せずに、直接フォワーディング ステートに移行させます。この切り替えに必要な時間は、約 1 ～ 5 秒です。



クロススタック UplinkFast

クロススタック UplinkFast（CSUF）は、スイッチ スタック全体にスパニングツリー高速移行（通常のネットワーク状態の下では1秒未満の高速コンバージェンス）を提供します。高速移行の間は、スタック上の代替冗長リンクがフォワーディング ステートになり、一時的なスパニングツリーループもバックボーンへの接続の損失も発生させません。一部の設定では、この機能により、冗長性と復元力を備えたネットワークが得られます。CSUF は UplinkFast 機能をイネーブルにすると、自動的にイネーブルになります。

CSUF で高速移行が得られない場合もあります。この場合は、通常のスパニングツリー移行が発生し、30 ～ 40 秒以内に完了します。詳細については、「関連項目」を参照してください。

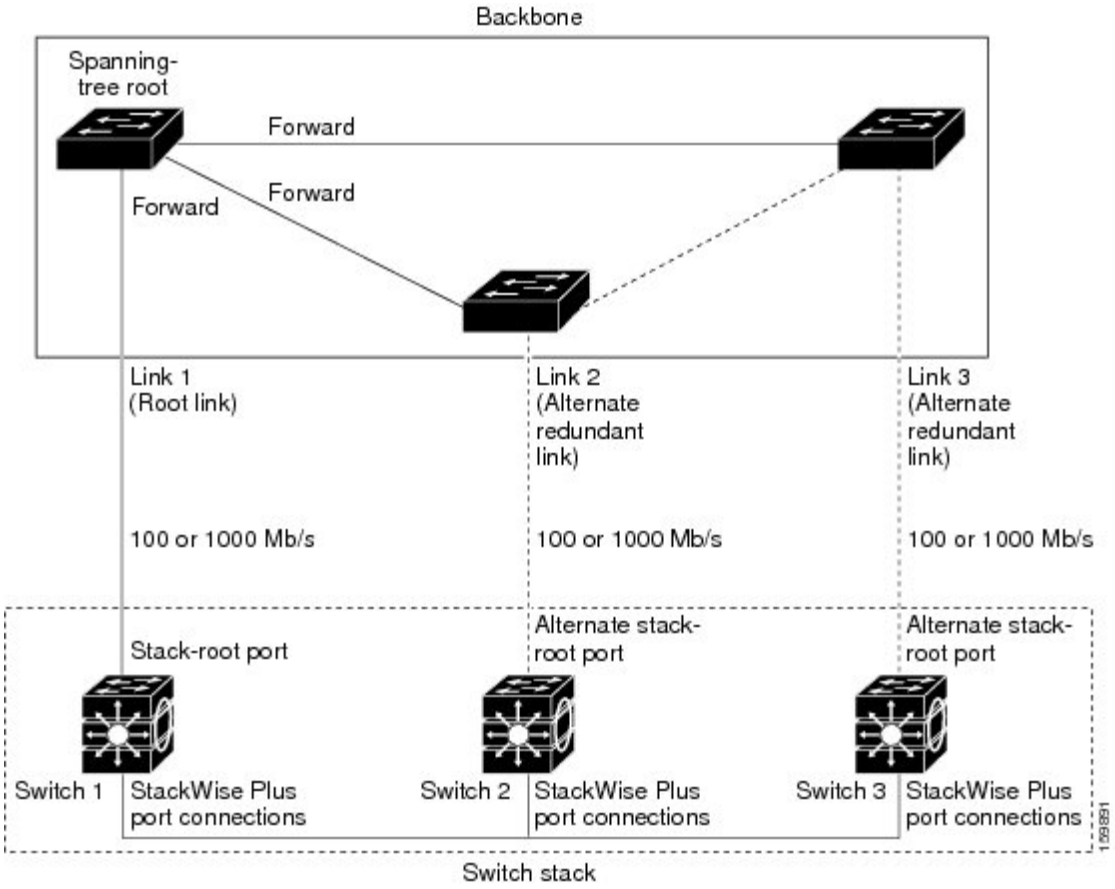
クロススタック UplinkFast の動作

クロススタック UplinkFast（CSUF）によって、ルートへのパスとしてスタック内で 1 つのリンクが確実に選択されます。

図 *36 :* クロススタック *UplinkFast* トポロジ

スイッチ 1 のスタックルート ポートは、スパニングツリーのルートへパスを提供しています。スイッチ 2 およびスイッチ 3 の代替スタックルート ポートは、現在のスタックルート スイッチに障害が発生したか、またはそのスパニングツリールートへのリンクに障害が発生した場合に、スパニングツリー ルートへの代替パスを提供できます。

ルート リンクである Link 1 は、スパニングツリー フォワーディング ステートになっています。Link 2 と Link 3 は、スパニングツリー ブロッキング ステートになっている代替冗長リンクです。スイッチ 1 に障害が発生したか、そのスタック ルート ポートに障害が発生したか、または Link 1 に障害が発生した場合には、CSUF が、1 秒未満でスイッチ 2 またはスイッチ 3 のいずれかにある代替スタックルート ポートを選択して、それをフォワーディング ステートにします。



特定のリンク損失またはスパニングツリーイベントが発生した場合（次のトピックを参照）、Fast Uplink Transition Protocol は、ネイバー リストを使用して、高速移行要求をスタック メンバーに送信します。

高速移行要求を送信するスイッチは、ルート ポートとして選択されたポートをフォワーディング ステートへ高速移行する必要があります。また、高速移行を実行するには、事前に各スタックから確認応答を取得しておく必要があります。

スタック内の各スイッチが、ルート、コスト、およびブリッジ ID を比較することにより、このスパニングツリー インスタンスのスタック ルートとなるよりも送信スイッチの方がよりよい選択肢であるかどうかを判断します。スタック ルートとして送信スイッチが最も良い選択である場合は、スタック内の各スイッチが確認応答を返します。それ以外の場合は、高速移行要求を送信します。この時点では、送信スイッチは、すべてのスタック スイッチから確認応答を受け取っていません。

すべてのスタック スイッチから確認応答を受け取ると、送信スイッチの Fast Uplink Transition Protocol は代替スタック ルート ポートをすぐにフォワーディング ステートに移行させます。送信スイッチがすべてのスタック スイッチからの確認応答を取得しなかった場合、通常のスパニングツリー移行（ブロッキング、リスニング、ラーニング、およびフォワーディング）が行われ、スパニングツリー トポロジが通常のレート（2 × 転送遅延時間 + 最大エージング タイム）で収束します。

Fast Uplink Transition Protocol は、VLAN ごとに実装されており、一度に 1 つのスパニングツリー インスタンスにしか影響しません。

高速コンバージェンスを発生させるイベント

CSUF 高速コンバージェンスは、ネットワーク イベントまたはネットワーク障害に応じて、発生する場合もあれば発生しない場合もあります。

高速コンバージェンス（通常のネットワーク状態で 1 秒未満）は、次のような状況で発生します。

- スタック ルート ポート リンクに障害が発生した。

  スタック内の 2 つのスイッチがルートへの代替パスを持つ場合、それらのスイッチの片方だけが高速移行を行います。

- スタック ルートをスパニングツリー ルートに接続するリンクに障害が発生し、回復した。
- ネットワークの再設定により、新しいスタック ルート スイッチが選択された。
- ネットワークの再設定により、現在のスタック ルート スイッチ上で新しいポートがスタック ルート ポートとして選択された。

（注） 複数のイベントが同時に発生すると、高速移行が行われなくなる場合もあります。たとえば、スタック メンバの電源がオフになり、それと同時にスタック ルートをスパニングツリー ルートに接続しているリンクが回復した場合、通常のスパニングツリー コンバージェンスが発生します。

通常のスパニングツリー コンバージェンス（30 ～ 40 秒）は、次のような状況で発生します。

- スタック ルート スイッチの電源がオフになったか、またはソフトウェアに障害が発生した。
- 電源がオフになっていたか、または障害が発生していたスタック ルート スイッチの電源が入った。
- スタック ルートになる可能性のある新しいスイッチがスタックに追加された。

BackboneFast

BackboneFast は、バックボーンのコアにおける間接障害を検出します。BackboneFast は、UplinkFast 機能を補完するテクノロジーです。UplinkFast は、アクセス スイッチに直接接続されたリンクの障害に対応します。BackboneFastは、最大エージング タイマーを最適化します。最大エージング タイマーによって、スイッチがインターフェイスで受信したプロトコル情報を保存しておく時間の長さが制御されます。スイッチが別のスイッチの指定ポートから下位BPDUを受信した場合、BPDUは他のスイッチでルートまでのパスが失われた可能性を示すシグナルとなり、BackboneFast はルートまでの別のパスを見つけようとします。

スイッチのルート ポートまたはブロックされたインターフェイスが、指定スイッチから下位 BPDU を受け取ると、BackboneFast が開始します。下位 BPDU は、ルート ブリッジと指定スイッチの両方を宣言しているスイッチを識別します。スイッチが下位BPDUを受信した場合、そのスイッチが直接接続されていないリンク（間接リンク）で障害が発生したことを意味します（指定スイッチとルート スイッチ間の接続が切断されています）。スパニングツリーのルールに従い、スイッチは最大エージング タイム（デフォルトは 20 秒）の間、下位 BPDU を無視します。

スイッチは、ルート スイッチへの代替パスの有無を判別します。下位 BPDU がブロック インターフェイスに到達した場合、スイッチ上のルート ポートおよび他のブロック インターフェイスがルート スイッチへの代替パスになります（セルフループ ポートはルート スイッチの代替パスとは見なされません）。下位 BPDU がルート ポートに到達した場合には、すべてのブロック インターフェイスがルート スイッチへの代替パスになります。下位 BPDU がルート ポートに到達し、しかもブロック インターフェイスがない場合、スイッチはルート スイッチへの接続が切断されたものと見なし、ルート ポートの最大エージング タイムが経過するまで待ち、通常のスパニングツリー ルールに従ってルート スイッチになります。

スイッチが代替パスでルート スイッチに到達できる場合、スイッチはその代替パスを使用して、Root Link Query（RLQ）要求を送信します。スイッチは、スタック メンバーがルート スイッチへの代替ルートを持つかどうかを学習するために、すべての代替パスに RLQ 要求を送信し、ネットワーク内およびスタック内の他のスイッチからのRLQ応答を待機します。スイッチは、すべての代替パスに RLQ 要求を送信し、ネットワーク内の他のスイッチからの RLQ 応答を待機します。

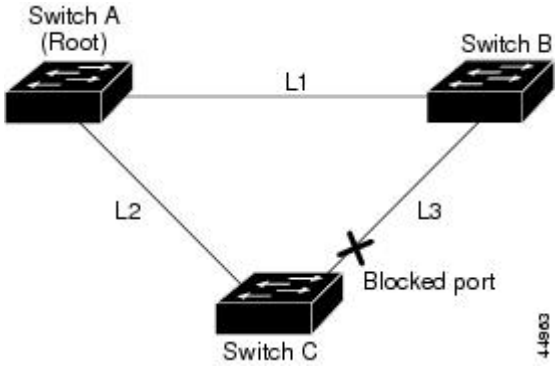
スタック メンバが、ブロック インターフェイス上の非スタック メンバから RLQ 応答を受信し、その応答が他の非スタック スイッチ宛てのものであった場合、そのスタック メンバは、スパニングツリー インターフェイス ステートに関係なく、その応答パケットを転送します。

スタック メンバが非スタック メンバからRLQ応答を受信し、その応答がスタック宛てのものであった場合、そのスタック メンバは、他のすべてのスタック メンバがその応答を受信するようにその応答を転送します。

ルートへの代替パスがまだ存在していると判断したスイッチは、下位 BPDU を受信したインターフェイスの最大エージング タイムが経過するまで待ちます。ルート スイッチへのすべての代替パスが、スイッチとルートスイッチ間の接続が切断されていることを示している場合、スイッチは RLQ 応答を受信したインターフェイスの最大エージング タイムを満了させます。1 つまたは複数の代替パスからルート スイッチへ引き続き接続できる場合、スイッチは下位BPDUを受信したすべてのインターフェイスを指定ポートにして、（ブロッキング ステートになっていた場合）ブロッキング ステートを解除し、リスニング ステート、ラーニング ステートを経てフォワーディング ステートに移行させます。

図 *37* : 間接リンク障害が発生する前の *BackboneFast* の例

これは、リンク障害が発生していないトポロジ例です。ルート スイッチであるスイッチ A はリンク L1 を介してスイッチ B に、リンク L2 を介してスイッチ C に直接接続されています。スイッチ B に直接接続されているスイッチ C のレイヤ 2 インターフェイスは、ブロッキング



ステートです。

図 *38* : 間接リンク障害が発生したあとの *BackboneFast* の例

リンク L1 で障害が発生した場合、スイッチ C はリンク L1 に直接接続されていないので、この障害を検出できません。一方スイッチ B は、L1 によってルート スイッチに直接接続されているため障害を検出し、スイッチB自身をルートとして選定して、自らをルートとして特定した状態でBPDUをスイッチCへ送信し始めます。スイッチBから下位BPDUを受信したスイッチCは、間接障害が発生していると見なします。この時点で、BackboneFastは、スイッチCのブロック インターフェイスを、インターフェイスの最大エージング タイムが満了するまで待たずに、ただちにリスニング ステートに移行させます。BackboneFast は、次に、スイッチ C のレイヤ2インターフェイスをフォワーディングステートに移行させ、スイッチBからスイッチ A へのパスを提供します。ルートスイッチの選択には約 30 秒必要です。これは転送遅延時間がデフォルトの 15 秒に設定されていればその倍の時間です。BackboneFast がリンク L1 で発

生した障害に応じてトポロジを再設定します。

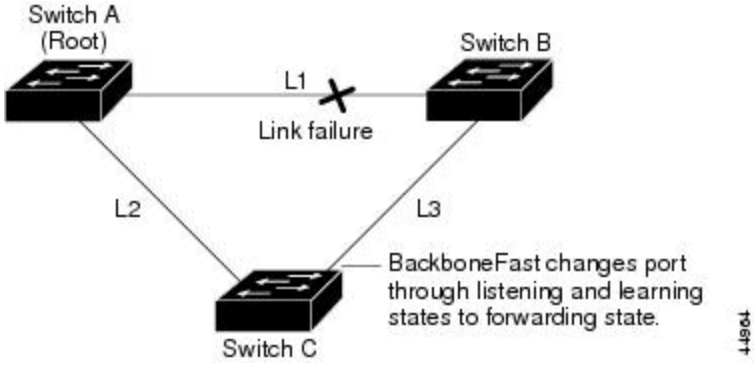
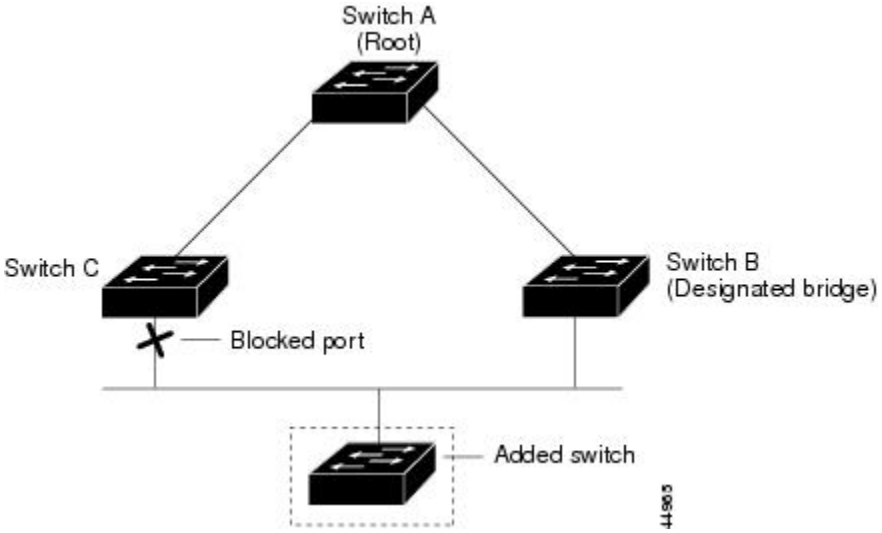


図 *39 :* メディア共有型トポロジにおけるスイッチの追加

新しいスイッチがメディア共有型トポロジに組み込まれた場合、認識された指定スイッチ（スイッチ B）から下位 BPDU が届いていないので、BackboneFast はアクティブになりません。新しいスイッチは、自身がルート スイッチであることを伝える下位 BPDU の送信を開始します。ただし、他のスイッチはこれらの下位 BPDU を無視し、新しいスイッチはスイッチ B がルート スイッチであるスイッチ A への指定スイッチであることを学習します。



EtherChannel ガード

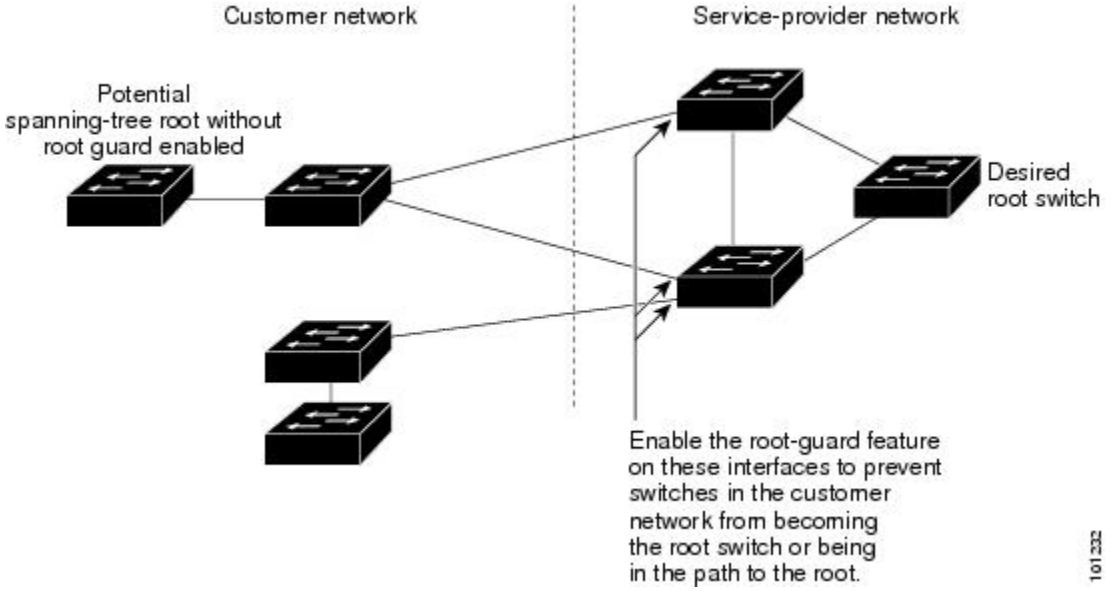
EtherChannel ガードを使用すると、スイッチと接続したデバイス間での EtherChannel の設定の矛盾を検出できます。スイッチ インターフェイスは EtherChannel として設定されているものの、もう一方のデバイスのインターフェイスではその設定が行われていない場合、設定の矛盾が発生します。また、EtherChannel の両端でチャネルのパラメータが異なる場合にも、設定の矛盾が発生します。

スイッチが、他のデバイス上で設定の矛盾を検出した場合、EtherChannel ガードは、スイッチのインターフェイスを errdisable ステートにし、エラー メッセージを表示します。

ルート ガード

図 *40* : サービス プロバイダー ネットワークのルート ガード

サービス プロバイダー（SP）のレイヤ 2 ネットワークには、SP 以外が所有するスイッチへの接続が多く含まれている場合があります。このようなトポロジでは、スパニングツリーが再構成され、カスタマー スイッチをルート スイッチとして選択する可能性があります。この状況を防ぐには、カスタマー ネットワーク内のスイッチに接続する SP スイッチ インターフェイス上でルート ガード機能を有効に設定します。スパニングツリーの計算によってカスタマー ネットワーク内のインターフェイスがルート ポートとして選択されると、ルート ガードがそのインターフェイスを root-inconsistent（ブロッキング）ステートにして、カスタマーのスイッチがルート スイッチにならないようにするか、ルートへのパスに組み込まれないようにします。



SP ネットワーク外のスイッチがルート スイッチになると、インターフェイスがブロックされ（root-inconsistent ステートになり）、スパニングツリーが新しいルート スイッチを選択します。カスタマーのスイッチがルート スイッチになることはありません。ルートへのパスに組み込まれることもありません。

スイッチが MST モードで動作している場合、ルート ガードが強制的にそのインターフェイスを指定ポートにします。また、境界ポートがルート ガードによって Internal Spanning-Tree（IST）インスタンスでブロックされている場合にも、このインターフェイスはすべての MST インスタンスでもブロックされます。境界ポートは、指定スイッチが IEEE 802.1D スイッチまたは異なる MST リージョン設定を持つスイッチのいずれかである LAN に接続されるインターフェイスです。

1 つのインターフェイス上でルート ガードをイネーブルにすると、そのインターフェイスが所属するすべての VLAN にルート ガードが適用されます。VLAN は、MST インスタンスに対してグループ化された後、マッピングされます。

⚠

注意 ルート ガード機能を誤って使用すると、接続が切断されることがあります。

ループ ガード

ループ ガードを使用すると、代替ポートまたはルート ポートが、単一方向リンクの原因となる障害によって指定ポートになることを防ぎます。この機能は、スイッチドネットワーク全体でイネーブルにした場合に最も効果があります。ループ ガードによって、代替ポートおよびルート ポートが指定ポートになることが防止され、スパニングツリーがルート ポートまたは代替ポートで BPDU を送信することはありません。

スイッチが PVST+ または Rapid PVST+ モードで動作している場合、ループ ガードによって、代替ポートおよびルートポートが指定ポートになることが防止され、スパニングツリーがルート ポートまたは代替ポートで BPDU を送信することはありません。

スイッチが MST モードで動作しているとき、ループ ガードによってすべての MST インスタンスでインターフェイスがブロックされている場合でのみ、非境界ポートで BPDU を送信しません。境界ポートでは、ループ ガードがすべての MST インスタンスでインターフェイスをブロックします。

STP PortFast ポート タイプ

スパニングツリー ポートは、エッジ ポート、ネットワーク ポート、または標準ポートとして構成できます。ポートは、ある一時点において、これらのうちいずれか 1 つの状態をとります。デフォルトのスパニング ツリー ポート タイプは「標準」です。ポート タイプは、グローバル単位でもインターフェイス単位でも設定できます。

インターフェイスが接続されているデバイスのタイプによって、スパニング ツリー ポートを下記のいずれかのポート タイプに設定できます。

- PortFast エッジ ポート：レイヤ 2 ホストに接続されます。これにはアクセス ポートまたはエッジ トランク ポート（**portfast edge trunk**）のいずれかを使用できます。このタイプのポートインターフェイスは、リスニング ステートとラーニング ステートをバイパスして、直接フォワーディング ステートに移行します。1 台のワークステーションまたはサーバに接続されたレイヤ 2 アクセス ポート上で PortFast エッジを使用すると、スパニング ツリーのコンバージェンスを待たずに、デバイスがただちにネットワークに接続されます。

  インターフェイスでブリッジ プロトコル データ ユニット（BPDU）が受信されても、スパニングツリーがポートをブロッキング ステートにしません。スパニング ツリーは、設定されたステートが *port fast edge* のままでトポロジ変更への参加を開始している場合でも、ポートの動作ステートを *non-port fast* に設定します。

（注） レイヤ 2 スイッチまたはブリッジに接続しているポートをエッジ ポートとして設定すると、ブリッジング ループが発生することがあります。

- PortFast ネットワーク ポート：レイヤ 2 スイッチまたはブリッジのみに接続されます。Bridge Assurance は PortFast ネットワーク ポート上でのみ有効になります。詳細については、*Bridge Assurance* を参照してください。

（注） レイヤ2にホスト接続されたポートをスパニング ツリー ネットワーク ポートとして設定すると、そのポートは自動的にブロッキング ステートになります。

- PortFast 標準ポート：スパニング ツリー ポートのデフォルト タイプです。

（注） Cisco IOS リリース 15.2(4) E または IOS XE 3.8.0E 以降、グローバルまたはインターフェイス コンフィギュレーション モードで **spanning-tree portfast** [trunk] コマンドを入力すると、このコマンドが **spanning-tree portfast edge** [trunk] として自動的に保存されます。

Bridge Assurance

Bridge Assurance は、単方向リンク（リンクまたはポートの一方向のみのトラフィック）または隣接スイッチの機能不全が原因で発生するループ状態を防止するのに役立ちます。ここで言う機能不全とは、トラフィックの転送はまだ可能だが STP の実行ができなくなってしまったスイッチ（ブレインデッド スイッチ）のことを指します。

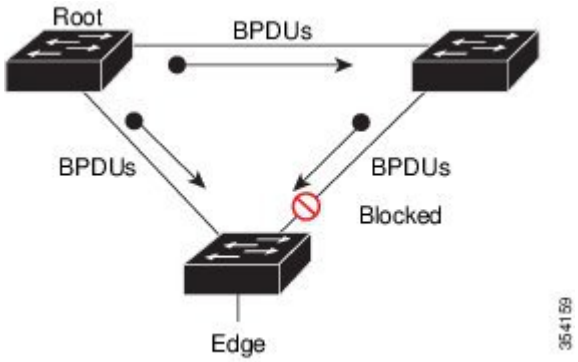
動作中のすべてのネットワーク ポート（代替ポートとバックアップ ポートを含む）に、BPDU が hello タイムごとに送出されます。Bridge Assurance では、すべてのネットワーク ポートのポイントツーポイント リンクでの BPDU の受信がモニタされます。割り当てられた hello タイム期間内にポートが BPDU を受信しない場合、ポートはブロック状態（フレームの転送が停止するポート不整合状態と同じ）になります。ポートが BPDU の受信を再開すると、ポートは通常のスパニング ツリー動作を再開します。

（注） Bridge Assurance をサポートするのは、Rapid PVST+ および MST スパニング ツリー プロトコルのみです。PVST+ は Bridge Assurance をサポートしません。

次に、Bridge Assurance によってネットワークをブリッジング ループから保護する例を示します。

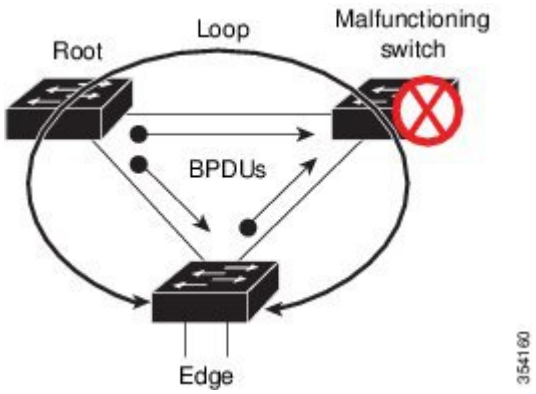
次の図は、標準的な STP トポロジを使用するネットワークを示しています。

図 *41* : 標準的な *STP* トポロジのネットワーク



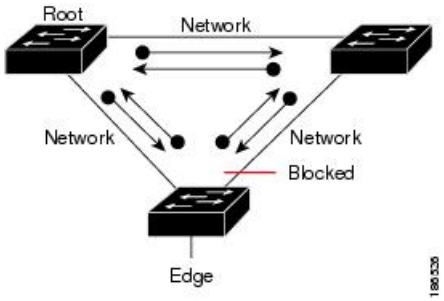
次の図は、デバイスで障害が発生し（ブレインデッド）、Bridge Assurance が有効でないときにネットワークで発生する可能性のある問題を示しています。

図 *42* : スイッチの機能不全によるネットワーク ループ



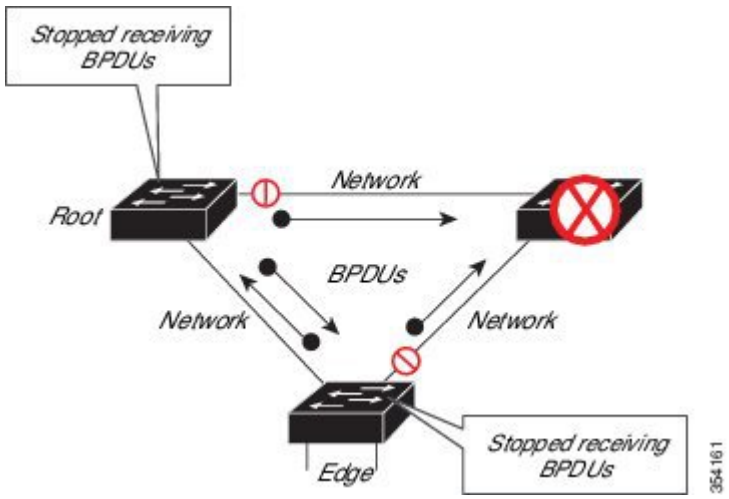
次の図は、Bridge Assurance が有効になっているネットワークで、すべての STP ネットワーク ポートから双方向 BPDU が発行される一般的な STP トポロジを示しています。

図 *43* : *Bridge Assurance* を実行している *STP* トポロジのネットワーク



次の図は、スイッチの機能不全によるネットワーク ループの図に示した潜在的なネットワーク問題を、ネットワークで Bridge Assurance を有効にすることによって回避する様子を示しています。

図 *44 : Bridge Assurance* によるネットワーク上の問題の回避



ポートがブロック/ブロック解除されると、システムは syslog メッセージを生成します。次の出力例は、それぞれの場合に生成されるログを示しています。

BRIDGE_ASSURANCE_BLOCK

```
Sep 17 09:48:16.249 PDT: %SPANTREE-2-BRIDGE_ASSURANCE_BLOCK: Bridge Assurance blocking
port GigabitEthernet1/0/1 on VLAN0001.
```

BRIDGE_ASSURANCE_UNBLOCK

```
Sep 17 09:48:58.426 PDT: %SPANTREE-2-BRIDGE_ASSURANCE_UNBLOCK: Bridge Assurance unblocking
 port GigabitEthernet1/0/1 on VLAN0001.
```

Bridge Assurance を有効にする際は、次の注意事項に従ってください。

- グローバルな有効化または無効化のみ可能です。
- これは、代替ポートとバックアップ ポートを含め、動作中のすべてのネットワーク ポートに適用されます。
- Bridge Assurance をサポートするのは、Rapid PVST+ および MST スパニング ツリー プロトコルのみです。PVST+ は Bridge Assurance をサポートしません。
- Bridge Assurance が正しく動作するには、ポイントツーポイント リンクの両端で Bridge Assurance がサポートおよび設定されている必要があります。リンクの一端のデバイスで Bridge Assurance が有効であっても、他端のデバイスで有効になっていない場合、接続ポートはブロックされ、Bridge Assurance 不整合状態となります。Bridge Assurance は、ネットワーク全体でイネーブルにすることを推奨します。
- ポート上で Bridge Assurance をイネーブルにするには、BPDU フィルタリングと BPDU Guard をディセーブルにする必要があります。

- Bridge Assurance は、Loop Guard とともにイネーブルにできます。
- Bridge Assurance は、ルート ガードとともにイネーブルにできます。後者は、ネットワークでのルート ブリッジの配置を強制する方法を提供するように設計されています。

オプションのスパニングツリー機能の設定方法

PortFast のイネーブル化

PortFast 機能がイネーブルに設定されているインターフェイスは、標準の転送遅延時間の経過を待たずに、すぐにスパニングツリー フォワーディング ステートに移行されます。

音声 VLAN 機能をイネーブルにすると、PortFast 機能が自動的にイネーブルになります。音声 VLAN をディセーブルにしても、PortFast 機能は自動的にディセーブルになりません。

スイッチで PVST+、Rapid PVST+、または MSTP が稼働している場合、この機能をイネーブルにできます。

⚠

注意 PortFast を使用するのは、1 つのエンド ステーションがアクセス ポートまたはトランク ポートに接続されている場合に限定されます。スイッチまたはハブに接続するインターフェイス上でこの機能をイネーブルにすると、スパニングツリーがネットワーク ループを検出または阻止できなくなり、その結果、ブロードキャスト ストームおよびアドレス ラーニングの障害が起きる可能性があります。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree portfast** [**trunk**]
5. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**spanning-tree portfast** [**trunk**] 例： `スイッチ(config-if)# spanning-tree portfast trunk`	単一ワーク ステーションまたはサーバーに接続されたアクセス ポート上で PortFast をイネーブルにします。 **trunk** キーワードを指定すると、トランクポート上で PortFast をイネーブルにできます。 （注） トランク ポートで PortFast をイネーブルにするには、**spanning-tree portfast trunk** インターフェイス コンフィギュレーション コマンドを使用する必要があります。**spanning-tree portfast** コマンドは、トランクポート上では機能しません。 トランク ポート上で PortFast をイネーブルにする場合は、事前に、トランク ポートとワークステーションまたはサーバーの間にループがないことを確認してください。 デフォルトでは、PortFast はすべてのインターフェイスでディセーブルです。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

次のタスク

spanning-tree portfast default グローバル コンフィギュレーション コマンドを使用すると、すべての非トランクポート上で PortFast 機能をグローバルにイネーブルにできます。

BPDU ガードのイネーブル化

スイッチで PVST+、Rapid PVST+、または MSTP が稼働している場合、BPDU ガード機能をイネーブルにできます。

⚠

注意 PortFastエッジは、エンドステーションに接続するインターフェイスのみに設定します。それ以外に設定すると、予期しないトポロジ ループが原因でデータのパケット ループが発生し、スイッチおよびネットワークの動作が妨げられることがあります。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree portfast edge**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	エンドステーションに接続するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**spanning-tree portfast edge** 例： `スイッチ(config-if)# spanning-tree portfast edge`	PortFast エッジ機能をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

次のタスク

ポートをシャットダウンしないようにするには、**errdisable detect cause bpduguard shutdown vlan** グローバルコンフィギュレーションコマンドを使用して、違反が発生したポート上の原因となっている VLAN だけをシャットダウンします。

PortFast 機能をイネーブルにしなくても、**spanning-tree bpduguard enable** インターフェイス コンフィギュレーション コマンドを使用して、任意のポートで BPDU ガードをイネーブルにすることもできます。BPDU を受信したポートは、errdisable ステートになります。

BPDU フィルタリングのイネーブル化

PortFast エッジ機能をイネーブルにしなくても、**spanning-tree bpdufilter enable** インターフェイス コンフィギュレーション コマンドを使用して、任意のインターフェイスで BPDU フィルタリングをイネーブルにすることもできます。このコマンドを実行すると、インターフェイスは BPDU を送受信できなくなります。

⚠

注意 BPDUフィルタリングを特定のインターフェイス上でイネーブルにすることは、そのインターフェイス上でスパニングツリーをディセーブルにすることと同じであり、スパニングツリー ループが発生することがあります。

スイッチで PVST+、Rapid PVST+、または MSTP が稼働している場合、BPDU フィルタリング機能をイネーブルにできます。

⚠

注意 PortFastエッジは、エンドステーションに接続するインターフェイスのみに設定します。それ以外に設定すると、予期しないトポロジ ループが原因でデータのパケット ループが発生し、スイッチおよびネットワークの動作が妨げられることがあります。

この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast edge bpdufilter default**
4. **interface** *interface-id*
5. **spanning-tree portfast edge**

6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree portfast edge bpdufilter default** 例： スイッチ(config)# **spanning-tree portfast edge bpdufilter default**	BPDU フィルタリングをグローバルにイネーブルにします。 BPDU フィルタリングは、デフォルトではディセーブルに設定されています。
ステップ 4	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/2**	エンドステーションに接続するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**spanning-tree portfast edge** 例： スイッチ(config-if)# **spanning-tree portfast edge**	指定したインターフェイスで PortFast エッジ機能をイネーブルにします。
ステップ 6	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

冗長リンクで使用するための UplinkFast のイネーブル化

（注） UplinkFastをイネーブルにすると、スイッチまたはスイッチスタックのすべてのVLANに影響します。個々の VLAN について UplinkFast を設定することはできません。

Rapid PVST+ または MSTP に対して UplinkFast または Cross-Stack UplinkFast（CSUF）機能を設定できますが、この機能は、スパニングツリーのモードを PVST+ に変更するまではディセーブル（非アクティブ）になったままです。

この手順は任意です。UplinkFast および CSUF をイネーブルにするには、次の手順に従います。

始める前に

スイッチ プライオリティが設定されている VLAN 上で UplinkFast をイネーブルにすることはできません。スイッチプライオリティが設定されている VLAN 上で UplinkFast をイネーブルにする場合は、最初に **no spanning-tree vlan** *vlan-id* **priority** グローバル コンフィギュレーション コマンドを使用することによって、VLAN のスイッチプライオリティをデフォルト値に戻す必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*]
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*] 例： `スイッチ(config)# spanning-tree uplinkfast max-update-rate 200`	UplinkFast をイネーブルにします。 （任意）*pkts-per-second* に指定できる範囲は毎秒 0 ～ 32000 パケットです。デフォルト値は 150 です。 0 を入力すると、ステーション学習フレームが生成されないので、接続切断後スパニングツリートポロジがコンバージェンスする速度が遅くなります。 このコマンドを入力すると、すべての非スタック ポート インターフェイス上で CSUF もイネーブルになります。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

UplinkFast をイネーブルにすると、すべての VLAN のスイッチ プライオリティは 49152 に設定されます。UplinkFast をイネーブルにする場合、または UplinkFast がすでにイネーブルに設定されている場合に、パス コストを 3000 未満の値に変更すると、すべてのインターフェイスおよび VLAN トランクのパス コストが 3000 だけ増加します（パス コストを 3000 以上の値に変更した場合、パス コストは変更されません）。スイッチ プライオリティおよびパス コストを変更すると、スイッチがルート スイッチになる可能性が低くなります。

デフォルト値を変更していない場合、UplinkFast をディセーブルにすると、すべての VLAN のスイッチ プライオリティとすべてのインターフェイスのパス コストがデフォルト値に設定されます。

次の手順に従って UplinkFast 機能をイネーブルにすると、CSUF は非スタック ポート インターフェイスで自動的にグローバルにイネーブルになります。

UplinkFast のディセーブル化

この手順は任意です。

UplinkFast および Cross-Stack UplinkFast（SUF）をディセーブルにするには、次の手順に従います。

始める前に

UplinkFast を有効にする必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **no spanning-tree uplinkfast**
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no spanning-tree uplinkfast** 例： スイッチ(config)# **no spanning-tree uplinkfast**	スイッチおよびそのスイッチのすべての VLAN で UplinkFast および CSUF をディセーブルにします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

デフォルト値を変更していない場合、UplinkFast をディセーブルにすると、すべての VLAN のスイッチ プライオリティとすべてのインターフェイスのパス コストがデフォルト値に設定されます。

次の手順に従って UplinkFast 機能をディセーブルにすると、CSUF は非スタック ポート インターフェイスで自動的にグローバルにディセーブルになります。

BackboneFast をイネーブル化

BackboneFast をイネーブルにすると、間接リンク障害を検出し、スパニングツリーの再構成をより早く開始できます。

Rapid PVST+ または MSTP に対して BackboneFast 機能を設定できます。ただし、スパニングツリーモードを PVST+ に変更するまで、この機能はディセーブル（非アクティブ）のままです。

この手順は任意です。スイッチ上で BackboneFast をイネーブルにするには、次の手順に従います。

始める前に

BackboneFast を使用する場合は、ネットワーク上のすべてのスイッチでイネーブルする必要があります。BackboneFast は、トークンリング VLAN ではサポートされません。この機能は他社製スイッチでの使用にサポートされています。

手順の概要

1. **enable**

2. **configure terminal**
3. **spanning-tree backbonefast**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree backbonefast** 例： `スイッチ(config)# spanning-tree backbonefast`	BackboneFast をイネーブルにします。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

EtherChannel ガードのイネーブル化

deviceで PVST+、Rapid PVST+、または MSTP が稼働している場合、EtherChannel の設定の矛盾を検出する EtherChannel ガード機能をイネーブルにできます。

この手順は任意です。

deviceで EtherChannel ガードをイネーブルにするには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree etherchannel guard misconfig**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree etherchannel guard misconfig** 例： スイッチ(config)# **spanning-tree etherchannel guard misconfig**	EtherChannel ガードをイネーブルにします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

次のタスク

show interfaces status err-disabled 特権 EXEC コマンドを使用することで、EtherChannel の設定矛盾が原因でディセーブルになっている device ポートを表示できます。リモートデバイス上では、**show etherchannel summary** 特権 EXEC コマンドを使用して、EtherChannel の設定を確認できます。

設定を修正した後、誤って設定していたポート チャネル インターフェイス上で、**shutdown** および **no shutdown** インターフェイス コンフィギュレーション コマンドを入力してください。

ルート ガードのイネーブル化

1 つのインターフェイス上でルート ガードをイネーブルにすると、そのインターフェイスが所属するすべての VLAN にルート ガードが適用されます。UplinkFast 機能が使用するインターフェイスで、ルート ガードをイネーブルにしないでください。UplinkFast を使用すると、障害発生時に（ブロック ステートの）バックアップ インターフェイスがルート ポートになります。ただし、同時にルート ガードもイネーブルになっていた場合は、UplinkFast 機能が使用するすべてのバックアップ インターフェイスが root-inconsistent（ブロック）ステートになり、フォワーディング ステートに移行できなくなります。

（注） ルート ガードとループ ガードの両方を同時にイネーブルにすることはできません。

スイッチで PVST+、Rapid PVST+、または MSTP が稼働している場合、この機能をイネーブルにできます。

この手順は任意です。

スイッチ上でルート ガードをイネーブルにするには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree guard root**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**spanning-tree guard root** 例： `スイッチ(config-if)# spanning-tree guard root`	インターフェイス上でルート ガードをイネーブルにします。 デフォルトでは、ルート ガードはすべてのインターフェイスでディセーブルです。
ステップ 5	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **end**	

ループ ガードのイネーブル化

ループ ガードを使用すると、代替ポートまたはルート ポートが、単一方向リンクの原因となる障害によって指定ポートになることを防ぎます。この機能は、スイッチドネットワーク全体に設定した場合に最も効果があります。ループ ガードは、スパニングツリーがポイントツーポイントと見なすインターフェイス上でのみ動作します。

（注） ループ ガードとルート ガードの両方を同時にイネーブルにすることはできません。

deviceで PVST+、Rapid PVST+、または MSTP が稼働している場合、この機能をイネーブルにできます。

この手順は任意です。deviceでループ ガードをイネーブルにするには、次の手順に従います。

手順の概要

1. 次のいずれかのコマンドを入力します。
   - **show spanning-tree active**
   - **show spanning-tree mst**
2. **configure terminal**
3. **spanning-tree loopguard default**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	次のいずれかのコマンドを入力します。 • **show spanning-tree active** • **show spanning-tree mst** 例： スイッチ# **show spanning-tree active** または スイッチ# **show spanning-tree mst**	どのインターフェイスが代替ポートまたはルート ポートであるかを確認します。
ステップ 2	**configure terminal** 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**spanning-tree loopguard default** 例： スイッチ(config)# **spanning-tree loopguard default**	ループ ガードをイネーブルにします。 ループ ガードは、デフォルトではディセーブルに設定されています。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

PortFast ポート タイプの有効化

このセクションでは、PortFast ポート タイプを有効化するさまざまな手順について説明します。

デフォルト ポート ステートのグローバル設定

デフォルト PortFast のステートを設定するには、次の作業を行います。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast** [**edge | network | normal**] **default**
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**spanning-tree portfast** [**edge** | **network** | **normal**] **default** 例： `スイッチ(config)# spanning-tree portfast default`	スイッチ上のすべてのインターフェイスのデフォルト状態を設定します。次のオプションがあります。 • （任意）**edge**：すべてのインターフェイスをエッジ ポートとして設定します。このコマンドでは、すべてのポートがホストまたはサーバに接続されているものとします。 • （任意）**network**：すべてのインターフェイスをスパニング ツリー ネットワーク ポートとして設定します。このコマンドでは、すべてのポートがスイッチまたはブリッジに接続されているものとします。Bridge Assurance は、デフォルトですべてのネットワーク ポート上で有効化されています。 • （任意）**normal**：すべてのインターフェイスを通常のスパニング ツリー ポートとして設定します。標準ポートは、任意のタイプのデバイスに接続できます。 • **default**：デフォルトのポート タイプは「normal」です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

指定したインターフェイスでの PortFast エッジの設定

エッジポートとして設定されたインターフェイスは、リンク アップ時に、ブロッキング ステートやラーニング ステートを経由することなく、フォワーディング ステートに直接移行します。

（注） このタイプのポートの目的は、アクセス ポートがスパニング ツリーのコンバージェンスを待機する時間を最小限に抑えることです。したがって、アクセス ポートで使用したときに最も効果を発揮します。別のスイッチに接続しているポートでPortFastエッジを有効にすると、スパニング ツリー ループが作成されるリスクがあります。

指定のインターフェイスにエッジ ポートを設定する手順は、次のとおりです。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id* | **port-channel** *port_channel_number*
4. **spanning-tree portfast edge** [**trunk**]
5. end
6. **show running interface** *interface-id* | **port-channel** *port_channel_number*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id*	**port-channel** *port_channel_number* 例： `スイッチ(config)# interface gigabitethernet 1/0/1 | port-channel port_channel_number`	設定するインターフェイスを選択します。
ステップ 4	**spanning-tree portfast edge** [**trunk**] 例： `スイッチ(config-if)# spanning-tree portfast trunk`	エンドワークステーションまたはサーバに接続されたレイヤ 2 アクセス ポート上でエッジの動作を有効にします。 • （任意）**trunk** キーワード：トランク ポート上のエッジの動作を有効化します。リンクがトランクである場合、このキーワードを使用します。このコマンドを使用するのは、VLAN の終端となっており、そこからの STP BPDU がポートで受信されることのない、エンドホストのデバイスに接続されているポート上のみとします。このようなエンド ホスト デバイスには、ブリッジングをサポートするように設定されていないルータ上のワークステーション、サーバ、ポートなどがあります。 • PortFast エッジを無効にするには、コマンドの **no** バージョンを使用します。

	コマンドまたはアクション	目的
ステップ 5	end 例： スイッチ(config-if)# **end**	設定モードを終了します。
ステップ 6	**show running interface** *interface-id* | **port-channel** *port_channel_number* 例： スイッチ# **show running interface gigabitethernet 1/0/1| port-channel *port_channel_number***	設定を確認します。

指定したインターフェイスでの PortFast ネットワーク ポートの設定

レイヤ2スイッチおよびブリッジに接続されているポートをネットワーク ポートとして設定できます。

（注） Bridge Assurance は PortFast ネットワーク ポート上でのみ有効になります。詳細については、*Bridge Assurance* を参照してください。

ポートをネットワーク ポートとして設定するには、次の作業を行います。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id* | **port-channel** *port_channel_number*
4. **spanning-tree portfast network**
5. end
6. **show running interface** *interface-id* | **port-channel** *port_channel_number*

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**interface** *interface-id* | **port-channel** *port_channel_number* 例： スイッチ(config)# **interface gigabitethernet 1/0/1| port-channel *port_channel_number***	設定するインターフェイスを選択します。
ステップ 4	**spanning-tree portfast network** 例： スイッチ(config-if)# **spanning-tree portfast network**	エンドワークステーションまたはサーバに接続されたレイヤ 2 アクセス ポート上でエッジの動作を有効にします。 • ポートをネットワーク ポートとして設定します。Bridge Assurance をグローバルに有効化している場合、スパニングツリーネットワーク ポート上で Bridge Assurance が自動的に実行されます。 • PortFast を無効にするには、コマンドの **no** バージョンを使用します。
ステップ 5	end 例： スイッチ(config-if)# **end**	設定モードを終了します。
ステップ 6	**show running interface** *interface-id* | **port-channel** *port_channel_number* 例： スイッチ# **show running interface gigabitethernet 1/0/1 | port-channel *port_channel_number***	設定を確認します。

Bridge Assurance の有効化

Bridge Assurance を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **spanning-tree bridge assurance**
4. **end**
5. **show spanning-tree summary**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spanning-tree bridge assurance** 例： スイッチ(config)# **spanning-tree bridge assurance**	スイッチのすべてのネットワーク ポートで Bridge Assurance をイネーブルにします。 デフォルトでは、[Bridge Assurance] はイネーブルになっています。 この機能を無効にするには、このコマンドの **no** バージョンを使用します。ブリッジ保証をディセーブルにすると、すべての設定済みネットワーク ポートが標準のスパニングツリー ポートとして動作します。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show spanning-tree summary** 例： スイッチ# **show spanning-tree summary**	スパニング ツリー情報を表示し、Bridge Assurance が有効になっているかを示します。

例

例：指定したインターフェイスでの PortFast エッジの設定

次の例は、GigabitEthernetインターフェイス **1/0/1** でエッジの動作を有効化する方法を示しています。

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
```

```
Switch(config-if)# spanning-tree portfast edge
Switch(config-if)# end
Switch#
```

次に、設定を確認する例を示します。

```
Switch# show running-config interface gigabitethernet1/0/1
Building configuration...
Current configuration:
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport access vlan 200
switchport mode access
spanning-tree portfast edge
end
```

次の例は、ポートGigabitEthernet**1/0/1**が現在エッジ状態にあることを表示するための方法を示しています。

```
Switch# show spanning-tree vlan 200
VLAN0200
Spanning tree enabled protocol rstp
Root ID Priority 2
Address 001b.2a68.5fc0
Cost 3
Port 125 (GigabitEthernet5/9)
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 2 (priority 0 sys-id-ext 2)
Address 7010.5c9c.5200
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 0 sec
Interface Role Sts Cost Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/1 Desg FWD 4 128.1 P2p Edge
```

例：指定したインターフェイスでの PortFast ネットワーク ポートの設定

この例は、GigabitEthernet インターフェイス **1/0/1** をネットワーク ポートとして設定する方法を示しています。

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# spanning-tree portfast network
Switch(config-if)# end
Switch#
```

次に、設定を確認する例を示します。

```
Switch# show running-config interface gigabitethernet1/0/1
Building configuration...
Current configuration:
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport access vlan 200
switchport mode access
```

```
spanning-tree portfast network
end
```

この例は、show spanning-tree vlan の出力を示しています。

```
Switch# show spanning-tree vlan
Sep 17 09:51:36.370 PDT: %SYS-5-CONFIG_I: Configured from console by console2

VLAN0002
  Spanning tree enabled protocol rstp
  Root ID    Priority    2
             Address     7010.5c9c.5200
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    2      (priority 0 sys-id-ext 2)
             Address     7010.5c9c.5200
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  0   sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Gi1/0/1             Desg FWD 4         128.1    P2p Edge
Po4                 Desg FWD 3         128.480  P2p Network
Gi4/0/1             Desg FWD 4         128.169  P2p Edge
Gi4/0/47            Desg FWD 4         128.215  P2p Network

Switch#
```

例：Bridge Assurance の設定

この出力は、ポート GigabitEthernet **1/0/1** がネットワーク ポートとして設定され、現在 Bridge Assurance 不整合状態にあることを示しています。

（注） この出力ではポート タイプがネットワークおよび*BA_Inc と表示されています。これは、ポートが不整合状態にあることを示しています。

```
Switch# show spanning-tree
VLAN0010
Spanning tree enabled protocol rstp
Root ID Priority 32778
Address 0002.172c.f400
This bridge is the root
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)
Address 0002.172c.f400
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 300
Interface Role Sts   Cost    Prio.   Nbr    Type
---------------- ---- --- --------- -------- --------------------------------
Gi1/0/1    Desg BKN*4 128.270 Network, P2p *BA_Inc
```

この例は、show spanning-tree summary の出力を示しています。

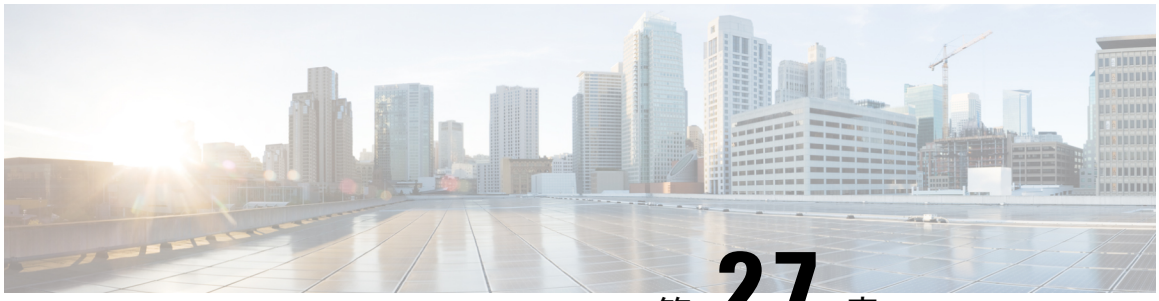
```
Switch#sh spanning-tree summary
```

```
Switch is in rapid-pvst mode
Root bridge for: VLAN0001-VLAN0002, VLAN0128
EtherChannel misconfig guard            is enabled
Extended system ID                      is enabled
Portfast Default                        is network
Portfast Edge BPDU Guard Default        is disabled
Portfast Edge BPDU Filter Default       is disabled
Loopguard Default                       is enabled
PVST Simulation Default                 is enabled but inactive in rapid-pvst mode
Bridge Assurance                        is enabled
UplinkFast                              is disabled
BackboneFast                            is disabled
Configured Pathcost method used is short

Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
VLAN0001                    0         0        0          5          5
VLAN0002                    0         0        0          4          4
VLAN0128                    0         0        0          4          4
---------------------- -------- --------- -------- ---------- ----------
3 vlans                     0         0        0         13         13

Switch#
```

スパニングツリー ステータスのモニタリング

表 *49:* スパニングツリー ステータスをモニタリングするコマンド

コマンド	目的
show spanning-tree active	アクティブ インターフェイスに関するスパニングツリー情報だけを表示します。
show spanning-tree detail	インターフェイス情報の詳細サマリーを表示します。
show spanning-tree interface *interface-id*	指定したインターフェイスのスパニングツリー情報を表示します。
show spanning-tree mst interface *interface-id*	指定インターフェイスのMST情報を表示します。
show spanning-tree summary [**totals**]	インターフェイス ステートのサマリーを表示します。またはスパニングツリー ステート セクションのすべての行を表示します。
show spanning-tree mst interface *interface-id* **portfast edge**	指定したインターフェイスのスパニングツリー portfast 情報を表示します。



第 27 章

双方向フォワーディング検出の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

双方向フォワーディング検出の前提条件

BFD の前提条件は次のとおりです。

- スイッチのフィーチャ セットは、IP Base またはそれ以上です。IP Base フィーチャ セットは Enhanced Interior Gateway Routing Protocol（EIGRP）スタブ ルーティングのみをサポートします。BFD は使用しません。IP Service フィーチャ セットは BFD を使用して EIGRP をサポートします。
- IP ルーティングは、参加しているすべてのスイッチでイネーブルにする必要があります。

- BFD を展開する前に、スイッチの BFD でサポートされている IP ルーティング プロトコルのいずれかを設定します。また、使用する予定のルーティング プロトコルの高速コンバージェンスも実装します。

双方向フォワーディング検出の制約事項

BFD の制約事項は次のとおりです。

- BFD は直接接続されたネイバーだけに対して動作します。BFD のネイバーは 1 ホップ以内に限られます。マルチホップのコンフィギュレーションはサポートされません。
- スイッチでは、最小 hello 間隔 100 ms、倍率 3 で最大 100 の BFD セッションがサポートされます。この倍率は、セッションがダウンしたと宣言される前に失われた可能性のある連続するパケットの最小数を指定します。
- エコー モードをイネーブルにするには、ピア システムを no ip redirects コマンドで設定する必要があります。

双方向フォワーディング検出について

BFD の動作

BFD は、インターフェイス、データリンク、および転送プレーンを含めて、2 つの隣接ルータ間の転送パスで、オーバーヘッドの少ない短期間の障害検出方法を提供します。

BFD はインターフェイス レベルおよびルーティング プロトコル レベルでイネーブルにする検出プロトコルです。シスコでは BFD 非同期モードをサポートしています。これは、ルータ間の BFD ネイバー セッションをアクティブにして維持するための、2 台のシステム間の BFD 制御パケットの送信に依存します。したがって、BFD セッションを作成するには、両方のシステムで（または BFD ピアで）BFD を設定する必要があります。適切なルーティング プロトコルに対して、インターフェイス レベルおよびルータ レベルで BFD がイネーブルになっている場合、BFD セッションが作成されて BFD タイマーがネゴシエートされ、ネゴシエートされた間隔で BFD ピアが互いに BFD 制御パケットの送信を開始します。

シスコは、BFD エコー モードをサポートしています。エコー パケットはフォワーディング エンジンによって送信され、検出を実行するために同じパスに沿って返信されます。もう一方の BFD セッションは、エコー パケットの実際のフォワーディングに参加しません。

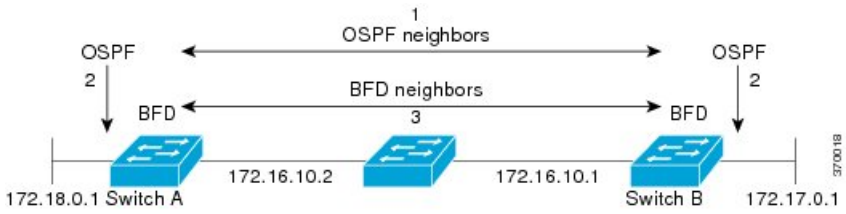
ここでは、次の内容について説明します。

ネイバー関係

BFD はあらゆるメディア タイプ、カプセル化、トポロジ、ルーティング プロトコル BGP、EIGRP、IS-IS、および OSPF の個別の高速 BFD ピア障害検出時間を提供します。ローカル ルー

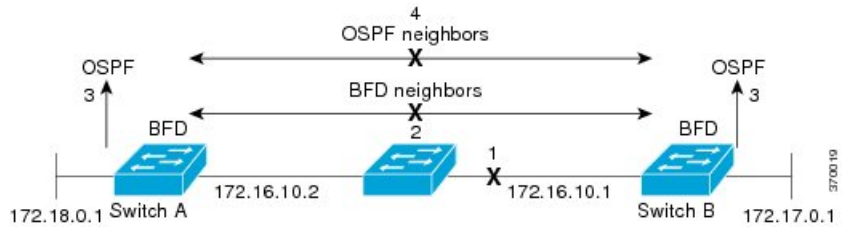
タのルーティング プロトコルに高速障害検出通知を送信して、ルーティング テーブル再計算プロセスを開始すると、BFD はネットワーク コンバージェンス時間全体を大幅に短縮できます。下の図に、OSPF と BFD を実行する 2 台のルータがある単純なネットワークを示します。OSPF がネイバー（1）を検出すると、OSPF ネイバー ルータ（2）で BFD ネイバー セッションを開始する要求が、ローカル BFD プロセスに送信されます。OSPF ネイバー ルータでの BFD ネイバー セッションが確立されます（3）。

図 *45 : BFD* ネイバー関係の確立



以下の図に、ネットワークで障害が発生した場合を示します（1）。OSPF ネイバー ルータでの BFD ネイバー セッションが停止されます（2）。BFD はローカル OSPF プロセスに BFD ネイバーに接続できなくなったことを通知します（3）。ローカル OSPF プロセスは OSPF ネイバー関係を解除します（4）。代替パスを使用できる場合、ルータはただちにコンバージェンスを開始します。

図 *46 : OSPF* ネイバー関係の解除



ルーティング プロトコルでは、取得したネイバーそれぞれについて、BFD で登録する必要があります。ネイバーが登録されると、セッションがまだ存在していない場合、BFD によって、ネイバーとのセッションが開始されます。

次のとき、OSPF では、BFD を使用して登録が行われます。

- ネイバーの有限状態マシン（FSM）は、Full ステートに移行します。
- OSPF BFD と BFD の両方が有効にされます。

ブロードキャスト インターフェイスでは、OSPF によって、指定ルータ（DR）とバックアップ指定ルータ（BDR）とともにのみ、BFD セッションが確立されますが、DROTHER ステートのすべての 2 台のルータ間では確立されません。

BFD の障害検出

BFD セッションが確立され、タイマーの取り消しが完了すると、BFD ピアは IGP hello プロトコルと同様に動作する（ただし、より高速な）、BFD 制御パケットを送信して状態を検出します。次の点に注意する必要があります。

- BFD はフォワーディング パスの障害検出プロトコルです。BFD は障害を検出しますが、障害が発生したピアをバイパスするには、ルーティング プロトコルがアクションを実行する必要があります。
  - 通常、BFD はどのプロトコル レイヤでも使用できます。ただし、シスコの BFD 実装では、特に BGP、EIGRP、IS-IS、および OSPF ルーティング プロトコル、およびスタティック ルーティングのレイヤ 3 クライアントだけがサポートされます。
- シスコの BFD 実装では、シスコ デバイスが複数のクライアント プロトコルに 1 つの BFD セッションを使用します。たとえば、同じピアへの同じリンクを介してネットワークで OSPF および EIGRP を実行している場合、1 つの BFD セッションだけが確立され、BFD で両方のルーティング プロトコルとセッション情報を共有します。ただし、IPv4 および IPv6 クライアントは BFD セッションを共有できません。

BFD バージョンの相互運用性

スイッチは、BFDバージョン1およびBFDバージョン0をサポートします。デフォルトでは、すべての BFD セッションがバージョン 1 で実行され、バージョン 0 と相互運用可能です。システムで自動的にFDバージョン検出が実行される場合、ネイバー間のBFDセッションがネイバー間の最も一般的な BFD バージョンで実行されます。たとえば、BFD ネイバーが BFD バージョン0を実行し、他のBFDネイバーがバージョン1を実行している場合、セッションでBFDバージョン 0 が実行されます。**show bfd neighbors** [**details**] コマンドの出力で、BFD ネイバーが実行している BFD バージョンを確認できます。

BFD セッションの制限

作成できる BFD セッションの最小数は、「hello」間隔によって異なることがあります。100 ms の「hello」間隔では、100 セッションが許可されます。より大きい hello 間隔では、より多くのセッションが許可されます。VLAN インターフェイスでは、最小「hello」間隔は 600 ms です。

非ブロードキャスト メディア インターフェイスに対する BFD サポート

BFD 機能はスイッチの VLAN インターフェイスでサポートされています。

bfd interval コマンドは、BFD モニタリングを開始するインターフェイスで設定する必要があります。

ステートフル スイッチオーバーでのノンストップ フォワーディングの BFD サポート

通常、ネットワーキング デバイスを再起動すると、そのデバイスのすべてのルーティング ピアがデバイスの終了および再起動を検出します。この遷移によってルーティング フラップが発生し、そのために複数のルーティング ドメインに分散される可能性があります。ルーティングの再起動によって発生したルーティング フラップによって、ルーティングが不安定になります。これはネットワーク全体のパフォーマンスに悪影響を及ぼします。ノンストップ フォワーディング（NSF）は、ステートフル スイッチオーバー（SSO）がイネーブルになっているデバ

イスのルーティング フラップを抑制するのに役立ち、それによってネットワークの不安定さが減少します。

NSF では、ルーティング プロトコル情報がスイッチオーバー後に保存されるとき、既知のルータでデータ パケットのフォワーディングを継続できます。NSF を使用すると、ピア ネットワーキング デバイスでルーティング フラップが発生しません。データ トラフィックはインテリジェント ラインカードまたはデュアル フォワーディング プロセッサを介して転送されますが、スタンバイ RP では、スイッチオーバー中に障害が発生したアクティブな RP からの制御と見なされます。ラインカードおよびフォワーディング プロセッサの機能はスイッチオーバーによって維持され、アクティブな RP の転送情報ベース（FIB）が NSF 動作で最新状態が維持されます。

デュアル RP をサポートするデバイスでは、SSO が RP の 1 つをアクティブなプロセッサとして確立し、他の RP はスタンバイ プロセッサに割り当てられ、それらの間で情報が同期されます。アクティブな RP に障害が発生したとき、ネットワーキング デバイスから削除されたとき、または手動でメンテナンスから排除されたときに、アクティブなプロセッサとスタンバイ プロセッサからのスイッチオーバーが発生します。

ステートフル スイッチオーバーの BFD サポート

BFD プロトコルでは、隣接するフォワーディング エンジン間でパスに短期間の障害検出が行われます。デュアル RP スイッチ（冗長性のため）を使用するネットワーク導入では、スイッチにグレースフル リスタート メカニズムがあり、アクティブな RP とスタンバイ RP の間のスイッチオーバー時にフォワーディング状態が保護されます。

スタンバイ RP のステートフル BFD

スタンバイ RP へのスイッチオーバーを成功させるために、BFD プロトコルでチェックポイント メッセージを使用して、アクティブな RP Cisco IOS インスタンスからセッション情報をスタンバイ RP Cisco IOS インスタンスに送信します。セッション情報には、ローカル識別子およびリモート識別子、隣接ルータのタイマー情報、BFD セットアップ情報、およびセッション固有の情報（セッションのタイプやセッションのバージョンなど）が含まれます。さらに、BFD プロトコルはセッションの作成および削除のチェックポイント メッセージを送信して、スタンバイ RP でセッションを作成または削除します。

スタンバイ RP の BFD セッションはパケットの送受信を行わず、期限切れになったタイマーを処理しません。このようなセッションは、スイッチオーバーの発生を待ってからアクティブ セッションのパケットを送信し、セッションが隣接スイッチでタイムアウトにならないようにします。

スタンバイ RP の BFD プロトコルはスイッチオーバーの通知を受けると、状態をアクティブに変更し、自分自身をシスコ エクスプレス フォワーディングに登録することで、パケットを受信し、期限切れになったすべての要素にパケットを送信できるようにします。

また、BFD ではチェックポイント メッセージを使用して、アクティブな RP でクライアントによって作成されたセッションをスイッチオーバー時に維持します。スイッチオーバーが発生すると、BFD は SSO 再要求タイマーを起動します。クライアントは再要求タイマーによって指定された期間内のセッションを再要求する必要があります。そうしないと、セッションが削除されます。

タイマーの値は、BFD セッションの数およびプラットフォームによって異なります。

表 *50* : スイッチの *BFD* タイマー値

BFD セッションの最大数	BFD セッションタイプ	最小タイマー値（ms）	クライアント	注
100	非同期/エコー	100 x 3	すべて（All）	SSO スイッチでは、5 の倍数の使用が推奨されます。

スタティック ルーティングの BFD サポート

OSPF や BGP などの動的なルーティング プロトコルとは異なり、スタティック ルーティングにはピア検出の方法がありません。したがって、BFD が設定されると、ゲートウェイの到達可能性は完全に指定されたネイバーへの BFD セッションの状態に依存します。BFD セッションが開始されない限り、スタティック ルートのゲートウェイは到達不能と見なされ、したがって、影響を受けるルートが適切なルーティング情報ベース（RIB）にインストールされません。

BFD セッションが正常に確立されるように、ピア上のインターフェイスで BFD を設定し、ピア上の BFD クライアントに BFD ネイバーのアドレスを登録する必要があります。インターフェイスがダイナミック ルーティング プロトコルで使用される場合、後者の要件は通常、BFD の各ネイバーでルーティング プロトコル インスタンスを設定することによって満たされます。インターフェイスがスタティック ルーティングに排他的に使用される場合、この要件はピア上でスタティック ルートを設定することによって満たす必要があります。

BFD セッションが起動状態のときに BFD 設定がリモート ピアから削除された場合、BFD セッションの最新状態がスタティック スタティックに送信されません。その結果、スタティック ルートが RIB に残ります。唯一の回避策は、IPv4 スタティック BFD ネイバー設定を削除して、スタティック ルートが BFD セッション状態を追跡しないようにすることです。

障害検出に BFD を使用することの利点

機能を導入するときは、あらゆる代替策を検討し、トレードオフに注意することが重要です。

EIGRP、BGP、および OSPF の通常の導入で BFD に最も近い代替策は、EIGRP、BGP、および OSPF ルーティング プロトコルの変更された障害検出メカニズムを使用することです。

EIGRP の hello およびホールド タイマーを絶対最小値に設定する場合、EIGRP の障害検出速度が 1～2 秒程度に下がります。

BGP または OSPF に fast hello を使用する場合、これらの Interior Gateway Protocol（IGP）プロトコルによって障害検出メカニズムが最小 1 秒に減少します。

ルーティング プロトコルの減少したタイマー メカニズムで BFD を実装すると、いくつかの利点があります。

- EIGRP、BGP、および OSPF タイマーによって 1 秒または 2 秒の最小検出タイマーを実現できますが、障害検出が 1 秒未満になる場合もあります。

- BFDは特定のルーティングプロトコルに関連付けられていないため、EIGRP、BGP、およびOSPF の汎用の整合性のある障害検出メカニズムとして使用できます。
- BFD の一部をデータ プレーンに分散できるため、コントロール プレーンに全体が存在する分散 EIGRP、BGP、および OSPF タイマーよりも CPU の負荷を軽くすることができます。

双方向フォワーディング検出の設定方法

インターフェイスで BFD を設定して、BFD プロセスを開始します。BFD プロセスが開始されると、隣接するデータベースにエントリが作成されません。つまり、BFD制御パケットが送受信されません。BFD バージョン 1 でサポートされる BFD エコー モード。

BFD 制御パケットに加えて、BFD エコー パケットが送受信されます。適用可能なルーティング プロトコルの BFD サポートを設定すると、隣接作成が実行されます。ここでは、次の手順について説明します。

インターフェイスでの BFD セッション パラメータの設定

ここでは、BFD セッションのベースライン パラメータをインターフェイスで設定して、インターフェイスで BFD を設定する作業を行います。BFD ネイバーに対して BFD セッションを実行するインターフェイスごとに、次の作業を繰り返します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *interval-multiplier*
5. **no bfd echo**
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface** *type number* 例： `Switch(config)# interface GigabitEthernet 6/1`	インターフェイスのタイプと番号を指定し、デバイスをインターフェイス コンフィギュレーション モードにします。
ステップ 4	**bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *interval-multiplier* 例： `Switch(config-if)# no bfd echo`	インターフェイスで BFD をイネーブルにします。 ハードウェア オフロードをイネーブルにするために、BFD エコー モードをディセーブルにします。
ステップ 5	**no bfd echo** 例： `Switch(config-if)# no bfd echo`	ハードウェア オフロードをイネーブルにするために、BFD エコー モードをディセーブルにします。
ステップ 6	**end** 例： `Switch(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ダイナミック ルーティング プロトコルに対する BFD サポートの設定

デバイスレベルでダイナミック ルーティング プロトコルの BFD サポートをイネーブルにして、すべてのインターフェイスに対してグローバルに BFD サポートをイネーブルにするか、またはインターフェイスレベルでインターフェイスごとに BFD を設定することができます。

ここでは、次の作業について説明します。

BGP に対する BFD サポートの設定

この作業は、ボーダー ゲートウェイ プロトコル（BGP）が BFD に登録済みのプロトコルになり、BFD から転送パス検出障害メッセージを受信するように、BGP に対する BFD サポートを設定する場合に実行します。

始める前に

BGP は、参加しているすべてのスイッチで実行されている必要があります。

BFD セッションを BFD ネイバーに対して実行するインターフェイスで、BFD セッションの基本パラメータを設定する必要があります。詳細については、「インターフェイスでの BFD セッション パラメータの設定」の項を参照してください。

（注） **show bfd neighbors details** コマンドの出力には、設定された間隔が表示されます。ハードウェア オフロードされた BFD セッションが 50 ms の倍数でない Tx および Rx 間隔で設定されていたために変更された間隔は出力に表示されません。

手順の概要

1. **enable**
2. **configure terminal**
3. **router bgp** *as-tag*
4. **neighbor** *ip-address* **fall-over bfd**
5. **end**
6. **show bfd neighbors** [**details**]
7. **show ip bgp neighbor**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**router bgp** *as-tag* 例： `Switch(config)# router bgp tag1`	BGP プロセスを指定し、ルータ コンフィギュレーション モードを開始します。
ステップ 4	**neighbor** *ip-address* **fall-over bfd** 例： `Switch(config-router)# neighbor 172.16.10.2 fall-over bfd`	フェールオーバーに対する BFD サポートを有効にします。
ステップ 5	**end** 例： `Switch(config-router)# end`	ルータ コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 6	**show bfd neighbors** [**details**] 例： `Switch# show bfd neighbors detail`	（任意）BFD ネイバーがアクティブで、BFD が登録したルーティング プロトコルが表示されることを確認します。
ステップ 7	**show ip bgp neighbor** 例： `Switch# show ip bgp neighbor`	（任意）ネイバーへの BGP および TCP 接続についての情報を表示します。

EIGRP に対する BFD サポートの設定

ここでは、EIGRP が BFD の登録プロトコルとなり、BFD から転送パスの検出障害メッセージを受信するように、EIGRP に対する BFD サポートを設定する手順について説明します。EIGRP に対する BFD サポートをイネーブルにするには、2 つの方法があります。

- ルータ コンフィギュレーション モードで **bfd all-interfaces** コマンドを使用して、EIGRP がルーティングしているすべてのインターフェイスに対して BFD を有効にできます。
- ルータ設定モードで **bfd interface** *type number* コマンドを使用して、EIGRP がルーティングしているインターフェイスのサブセットに対して BFD を有効にできます。

始める前に

EIGRP は、関連するすべてのスイッチで実行する必要があります。

BFD セッションを BFD ネイバーに対して実行するインターフェイスで、BFD セッションの基本パラメータを設定する必要があります。詳細については、「インターフェイスでの BFD セッション パラメータの設定」を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **router eigrp** *as-number*
4. 次のいずれかを実行します。
   - **bfd all-interfaces**
   - **bfd interface** *type number*
5. **end**
6. **show bfd neighbors** [**details**]
7. **show ip eigrp interfaces** [*type number*] [*as-number*] [**detail**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**router eigrp** *as-number* 例： `Switch(config)# router eigrp 123`	EIGRP ルーティング プロセスを設定し、ルータ コンフィギュレーション モードを開始します。
ステップ 4	次のいずれかを実行します。 • **bfd all-interfaces** • **bfd interface** *type number* 例： `Switch(config-router)# bfd all-interfaces` 例： `Switch(config-router)# bfd interface FastEthernet 6/1`	EIGRP ルーティング プロセスに関連付けられたすべてのインターフェイスで、BFD をグローバルにイネーブルにします。 または EIGRP ルーティング プロセスに関連付けられた 1 つ以上のインターフェイスに対して、インターフェイスごとに BFD をイネーブルにします。
ステップ 5	**end** 例： `Switch(config-router) end`	ルータ コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。
ステップ 6	**show bfd neighbors** [**details**] 例： `Switch# show bfd neighbors details`	（任意）BFD ネイバーがアクティブで、BFD が登録したルーティング プロトコルが表示されることを確認します。
ステップ 7	**show ip eigrp interfaces** [*type number*] [*as-number*] [**detail**] 例： `Switch# show ip eigrp interfaces detail`	（任意）EIGRP に対する BFD サポートがイネーブルになっているインターフェイスを表示します。

OSPF に対する BFD サポートの設定

ここでは、OSPF が BFD の登録プロトコルとなり、BFD から転送パスの検出障害メッセージを受信するように、OSPF に対する BFD サポートを設定する手順について説明します。すべてのインターフェイスでグローバルに OSPF に対する BFD を設定するか、または 1 つ以上のインターフェイスで選択的に設定することができます。

OSPF に対する BFD サポートを有効にするには、2 つの方法があります。

- ルータ コンフィギュレーション モードで **bfd all-interfaces** コマンドを使用して、OSPF がルーティングしているすべてのインターフェイスに対して BFD を有効にできます。インターフェイス コンフィギュレーション モードで **ip ospf bfd** [**disable**] コマンドを使用して、個々のインターフェイスで BFD サポートを無効にできます。
- インターフェイス コンフィギュレーション モードで **ip ospf bfd** コマンドを使用すると、OSPF がルーティングしているインターフェイスのサブセットに対して BFD を有効にできます。

OSPF に対する BFD サポートのタスクについては、次の項を参照してください。

すべてのインターフェイスの OSPF に対する BFD サポートの設定

すべての OSPF インターフェイスの BFD を設定するには、次の作業を実行します。

すべての OSPF インターフェイスに対して BFD を設定するのではなく、特定の 1 つ以上のインターフェイスに対して BFD サポートを設定する場合は、「Configuring OSPF Support for BFD over IPv4 for One or More Interfaces」の項を参照してください。

始める前に

Open Shortest Path First（OSPF）は、参加しているすべてのスイッチで実行されている必要があります。

BFD セッションを BFD ネイバーに対して実行するインターフェイスで、BFD セッションの基本パラメータを設定する必要があります。詳細については、「インターフェイスでの BFD セッション パラメータの設定」の項を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **switch ospf** *process-id*
4. **bfd all-interfaces**
5. **end**
6. **show bfd neighbors** [**details**]
7. **show ip ospf**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**switch ospf** *process-id* 例： `Switch(config)# router ospf 4`	OSPF プロセスを指定し、ルータ コンフィギュレーション モードを開始します。
ステップ 4	**bfd all-interfaces** 例： `Switch(config-router)# bfd all-interfaces`	OSPF ルーティング プロセスに関連付けられたすべてのインターフェイスで、BFD をグローバルに有効にします。
ステップ 5	**end** 例： `Switch(config-router)# end`	ルータ コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。
ステップ 6	**show bfd neighbors** [**details**] 例： `Switch# show bfd neighbors detail`	（任意）BFD ネイバーがアクティブで、BFD が登録したルーティング プロトコルが表示されるかどうかの検証に使用できる情報を表示します。
ステップ 7	**show ip ospf** 例： `Switch# show ip ospf`	（任意）OSPF に対して BFD が有効になっているかどうかを検証するために使用できる情報を表示します。

1 つ以上のインターフェイスの OSPF に対する BFD サポートの設定

すべての OSPF インターフェイスの BFD を設定するには、次の作業を実行します。

すべての OSPF インターフェイスに対して BFD を設定するのではなく、特定の 1 つ以上のインターフェイスに対して BFD サポートを設定する場合は、「Configuring OSPF Support for BFD over IPv4 for One or More Interfaces」の項を参照してください。

始める前に

OSPF は、参加しているすべてのスイッチで実行されている必要があります。

BFD セッションを BFD ネイバーに対して実行するインターフェイスで、BFD セッションの基本パラメータを設定する必要があります。詳細については、「インターフェイスでのBFDセッション パラメータの設定」の項を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip ospf bfd** [**disable**]
5. **end**
6. **show bfd neighbors** [**details**]
7. **show ip ospf**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Switch(config)# interface fastethernet 6/1`	（任意）インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip ospf bfd** [**disable**] 例： `Switch(config-if)# ip ospf bfd`	（任意）OSPF ルーティング プロセスに関連付けられた 1 つ以上のインターフェイスに対して、インターフェイスごとに BFD をイネーブルまたはディセーブルにします。 （注） ルータ コンフィギュレーション モードで **bfdall-interfaces** コマンドを使用して OSPF が関連付けられたすべてのインターフェイスでBFDをイネーブルにした場合にだけ、**disable** キーワードを使用します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `Switch(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	**show bfd neighbors** [**details**] 例： `Switch# show bfd neighbors detail`	（任意）BFD ネイバーがアクティブで、BFD が登録したルーティング プロトコルが表示されるかどうかの検証に使用できる情報を表示します。
ステップ 7	**show ip ospf** 例： `Switch# show ip ospf`	（任意）OSPF に対して BFD が有効になっているかどうかを検証するために使用できる情報を表示します。

スタティック ルーティングに対する BFD サポートの設定

スタティック ルーティングのための BFD サポートを設定するには、このタスクを実行します。各 BFD ネイバーに対してこの手順を繰り返します。詳細については、「例：スタティック ルーティングのための BFD サポートの設定」の項を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no switchport**
5. **ip address** *ip-address mask*
6. **bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *interval-multiplier*
7. **exit**
8. **ip route static bfd** *interface-type interface-number ip-address* [**group** *group-name* [**passive**]]
9. **ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*]
10. **exit**
11. **show ip static route**
12. **show ip static route bfd**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Switch(config)# interface gigabitethernet 6/1`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**no switchport** 例： `Switch(config)# no switchport`	レイヤ 3 にインターフェイスを変更します。
ステップ 5	**ip address** *ip-address mask* 例： `Switch(config-if)# ip address 10.201.201.1 255.255.255.0`	インターフェイスに IP アドレスを設定します。
ステップ 6	**bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *interval-multiplier* 例： `Switch(config-if)# bfd interval 500 min_rx 500 multiplier 5`	インターフェイスで BFD をイネーブルにします。
ステップ 7	**exit** 例： `Switch(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 8	**ip route static bfd** *interface-type interface-number ip-address* [**group** *group-name* [**passive**]] 例： `Switch(config)# ip route static bfd serial 2/0 10.1.1.1 group group1 passive`	スタティック ルートの BFD ネイバーを指定します。 • BFD が直接接続されたネイバーだけでサポートされているため、*interface-type*、*interface-number*、および *ip-address* 引数は必須です。
ステップ 9	**ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*] 例：	スタティック ルートの BFD ネイバーを指定します。

	コマンドまたはアクション	目的
	`Switch(config)# ip route 10.0.0.0 255.0.0.0 Gigabitethernet 6/1 10.201.201.2`	
ステップ 10	**exit** 例： `Switch(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 11	**show ip static route** 例： `Switch# show ip static route`	（任意）スタティック ルート データベース情報を表示します。
ステップ 12	**show ip static route bfd** 例： `Switch# show ip static route bfd`	（任意）設定された BFD グループおよび non-group エントリからスタティック BFD の設定に関する情報を表示します。

BFD エコー モードの設定

デフォルトでは BFD エコー モードが有効になっていますが、方向ごとに個別に実行できるように、無効にすることもできます。

BFD エコー モードは非同期 BFD で動作します。エコー パケットはフォワーディング エンジンによって送信され、検出を実行するために、同じパスで転送されます。反対側の BFD セッションはエコー パケットの実際のフォワーディングに関与しません。エコー機能およびフォワーディング エンジンが検出プロセスを処理するため、2 つの BFD ネイバー間で送信される BFD 制御パケットの数が減少します。また、フォワーディング エンジンが、リモート システムを介さずにリモート（ネイバー）システムの転送パスをテストするため、パケット間の遅延のばらつきが向上する可能性があり、それによって BFD バージョン 0 を BFD セッションの BFD 制御パケットで使用する場合に、障害検出時間を短縮できます。

エコー モードを両端で実行している（両方の BFD ネイバーがエコー モードを実行している）場合は、非対称性がないと表現されます。

前提条件

BFD は、参加しているすべてのスイッチで実行されている必要があります。

CPU 使用率の上昇を避けるために、BFD エコーモードを使用する前に、**no ip redirects** コマンドを入力して、Internet Control Message Protocol（ICMP）リダイレクトメッセージの送信を無効にする必要があります。

BFD セッションを BFD ネイバーに対して実行するインターフェイスで、BFD セッションの基本パラメータを設定する必要があります。詳細については、「インターフェイスでの BFD セッション パラメータの設定」の項を参照してください。

機能制限

BFD バージョン 1 でサポートされる BFD エコー モード。

（注） BFD エコー モードは、ユニキャスト リバース パス転送（uRPF）の設定との組み合わせでは動作しません。BFD エコー モードと uRPF の設定がイネーブルの場合、セッションはフラップします。

BFD 低速タイマーの設定

このタスクでは、BFD の slow timer 値を変更する方法を示します。各 BFD スイッチに対してこのタスクを繰り返します。

手順の概要

1. **enable**
2. **configure terminal**
3. **bfd slow-timer** *milliseconds*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**bfd slow-timer** *milliseconds* 例： `Switch(config)# bfd slow-timer 12000`	BFD の slow timer を設定します。
ステップ 4	**end** 例： `Switch(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

非対称性のない BFD エコー モードの無効化

このタスクでは、非対称性のない BFD エコー モードをディセーブルにする方法を示します。スイッチからエコー パケットが送信されず、スイッチはネイバー スイッチが受信した BFD エコー パケットを転送しません。

各 BFD スイッチに対してこのタスクを繰り返します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no bfd echo**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no bfd echo** 例： `Switch(config)# no bfd echo`	BFD エコー モードを無効にします。 • **no** 形式を使用すると、BFD エコーモードを無効にできます。
ステップ 4	**end** 例： `Switch(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

BFD のモニタリングとトラブルシューティング

ここでは、維持とトラブルシューティングのために BFD 情報を取得する方法について説明します。これらのタスクのコマンドを必要に応じて任意の順序で入力できます。

BFD のモニタリングとトラブルシューティングを行うには、次の手順を実行します。

手順の概要

1. **enable**

2. **show bfd neighbors** [**details**]
3. **debug bfd** [**packet** | **event**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show bfd neighbors** [**details**] 例： `Switch# show bfd neighbors details`	（任意）BFD 隣接関係データベースを表示します。 • **details** キーワードを指定すると、すべての BFD プロトコル パラメータとネイバーごとにタイマーが表示されます。
ステップ 3	**debug bfd** [**packet**	**event**] 例： `Switch# debug bfd packet`	（任意）BFD パケットのデバッグ情報を表示します。

双方向フォワーディング検出の設定例

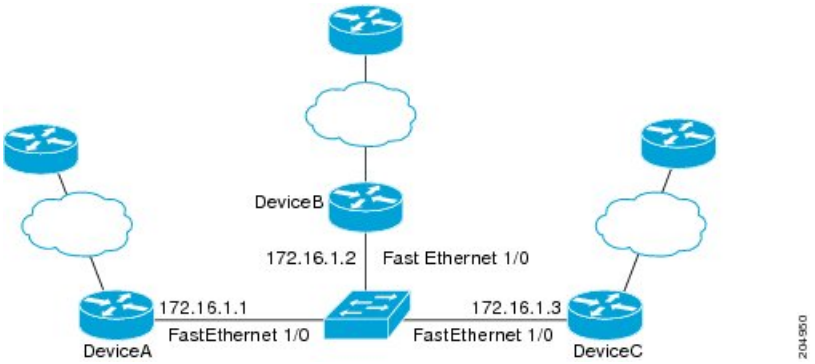
ここでは、次の設定例について説明します。

例：エコー モードがデフォルトでイネーブルになった EIGRP ネットワークでの BFD の設定

次の例では、EIGRP ネットワークにデバイス A、デバイス B およびデバイス C が含まれています。デバイス A のファスト イーサネット インターフェイス 1/0 がデバイス B のファスト イーサネット インターフェイス 1/0 と同じネットワークに接続されています。デバイス B のファスト イーサネット 1/0 が デバイス C のファスト イーサネット インターフェイス 1/0 と同じネットワークに接続されています。

デバイス A とデバイス B はエコー モードをサポートする BFD バージョン 1 を実行しており、デバイス C はエコー モードをサポートしない BFD バージョン 0 を実行しています。エコー モードはデバイス A とデバイス B の転送パスで動作するため、デバイス C とその BFD ネイバーの間の BFD セッションは非対称のエコー モードで実行されます。BFD セッションおよび障害検出のため、エコー パケットは同じパスで返されます。また、BFD ネイバー デバイス C は BFD バージョン 0 を実行し、BFD セッションおよび障害検出のために BFD 制御パケットを使用します。

下の図に、複数のデバイスがある大規模な EIGRP ネットワークを示します。その中の 3 台は、ルーティング プロトコルとして EIGRP を実行している BFD ネイバーです。



この例は、グローバル コンフィギュレーション モードから開始し、BFD の設定を示します。

デバイス A の設定

```
interface Fast Ethernet0/0
no shutdown
ip address 10.4.9.14 255.255.255.0
duplex auto
speed auto
!
interface Fast Ethernet1/0
ip address 172.16.1.1 255.255.255.0
bfd interval 50 min_rx 50 multiplier 3
no shutdown
duplex auto
speed auto
!
router eigrp 11
network 172.16.0.0
bfd all-interfaces
auto-summary
!
ip default-gateway 10.4.9.1
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 10.4.9.1
ip route 172.16.1.129 255.255.255.255 10.4.9.1
!
no ip http server
!
logging alarm informational
!
control-plane
!
line con 0
exec-timeout 30 0
stopbits 1
line aux 0
stopbits 1
line vty 0 4
login
!
!
end
```

デバイス B の設定

```
!
interface Fast Ethernet0/0
no shutdown
ip address 10.4.9.34 255.255.255.0
duplex auto
speed auto
!
interface Fast Ethernet1/0
ip address 172.16.1.2 255.255.255.0
bfd interval 50 min_rx 50 multiplier 3
no shtdown
duplex auto
speed auto
!
router eigrp 11
network 172.16.0.0
bfd all-interfaces
auto-summary
!
ip default-gateway 10.4.9.1
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 10.4.9.1
ip route 172.16.1.129 255.255.255.255 10.4.9.1
!
no ip http server
!
logging alarm informational
!
control-plane
!
line con 0
exec-timeout 30 0
stopbits 1
line aux 0
stopbits 1
line vty 0 4
login
!
!
end
```

デバイス C の設定

```
!
!
interface Fast Ethernet0/0
no shutdown
ip address 10.4.9.34 255.255.255.0
duplex auto
speed auto
!
interface Fast Ethernet1/0
ip address 172.16.1.2 255.255.255.0
bfd interval 50 min_rx 50 multiplier 3
no shutdown
duplex auto
speed auto
!
router eigrp 11
network 172.16.0.0
```

```
bfd all-interfaces
auto-summary
!
ip default-gateway 10.4.9.1
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 10.4.9.1
ip route 172.16.1.129 255.255.255.255 10.4.9.1
!
no ip http server
!
logging alarm informational
!
control-plane
!
line con 0
exec-timeout 30 0
stopbits 1
line aux 0
stopbits 1
line vty 0 4
login
!
!
end
```

デバイス A からの **show bfd neighbors details** コマンドの出力で、3 台のすべてのデバイス間に BFD セッションが作成され、EIGRP が BFD サポートに登録されることを確認できます。出力の最初のグループは、IP アドレスが 172.16.1.3 のデバイス C が BFD バージョン 0 を実行しているため、エコー モードを使用しないことを示します。出力の 2 番目のグループは、IP アドレスが 172.16.1.2 のデバイス B が BFD バージョン 1 を実行していて、50 ミリ秒の BFD interval パラメータが使用されていることを示します。この出力では、対応するコマンド出力が太字で表示されています。

```
DeviceA# show bfd neighbors details

OurAddr
       NeighAddr
      LD/RD  RH/RS    Holdown(mult)  State     Int
172.16.1.1    172.16.1.3
     5/3    1(RH)    150 (3 )       Up   Fa1/0
Session state is UP and not using echo function.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 150(0), Hello (hits): 50(1364284)
Rx Count: 1351813, Rx Interval (ms) min/max/avg: 28/64/49 last: 4 ms ago
Tx Count: 1364289, Tx Interval (ms) min/max/avg: 40/68/49 last: 32 ms ago
Registered protocols: EIGRP
Uptime: 18:42:45
Last packet: Version: 0
            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 3          - Your Discr.: 5
             Min tx interval: 50000    - Min rx interval: 50000
             Min Echo interval: 0
OurAddr       NeighAddr
     LD/RD  RH/RS   Holdown(mult)  State     Int
172.16.1.1    172.16.1.2
```

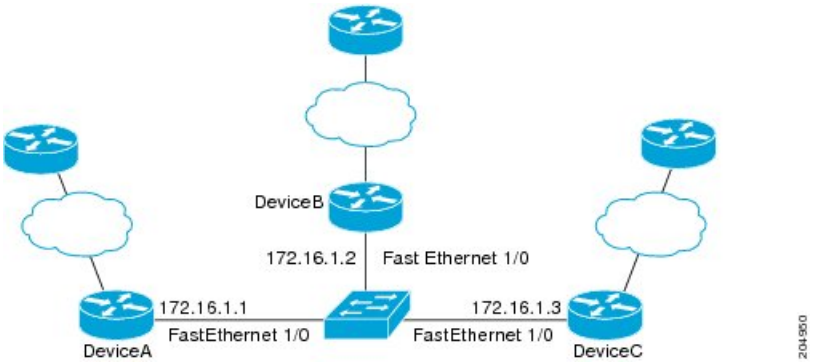
```
    6/1    Up        0    (3 )   Up        Fa1/0
Session state is UP and using echo function with 50 ms interval.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 1000000, Received Multiplier: 3
Holdown (hits): 3000(0), Hello (hits): 1000(317)
Rx Count: 305, Rx Interval (ms) min/max/avg: 1/1016/887 last: 448 ms ago
Tx Count: 319, Tx Interval (ms) min/max/avg: 1/1008/880 last: 532 ms ago
Registered protocols: EIGRP
Uptime: 00:04:30
Last packet: Version: 1

        - Diagnostic: 0
            State bit: Up         - Demand bit: 0
            Poll bit: 0           - Final bit: 0
            Multiplier: 3         - Length: 24
            My Discr.: 1          - Your Discr.: 6
            Min tx interval: 1000000    - Min rx interval: 1000000
            Min Echo interval: 50000
```

デバイス B の **show bfd neighbors details** コマンドによる出力で、BFD セッションが作成され、EIGRP が BFD サポートに対して登録されていることを確認できます。前述のように、デバイス A は BFD バージョン 1 を実行するため、エコー モードを実行しており、デバイス C は BFD バージョン 0 を実行するため、エコー モードを実行しません。この出力では、対応するコマンド出力が太字で表示されています。

```
DeviceB# show bfd neighbors details

OurAddr       NeighAddr
     LD/RD  RH/RS   Holdown(mult)  State     Int
172.16.1.2    172.16.1.1
    1/6    Up        0    (3 )   Up        Fa1/0
Session state is UP and using echo function with 50 ms interval.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 1000000, Received Multiplier: 3
Holdown (hits): 3000(0), Hello (hits): 1000(337)
Rx Count: 341, Rx Interval (ms) min/max/avg: 1/1008/882 last: 364 ms ago
Tx Count: 339, Tx Interval (ms) min/max/avg: 1/1016/886 last: 632 ms ago
Registered protocols: EIGRP
Uptime: 00:05:00
Last packet: Version: 1
           - Diagnostic: 0
             State bit: Up         - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 6          - Your Discr.: 1
             Min tx interval: 1000000    - Min rx interval: 1000000
             Min Echo interval: 50000
OurAddr       NeighAddr

 LD/RD  RH/RS   Holdown(mult)  State     Int
172.16.1.2    172.16.1.3
     3/6    1(RH)     118  (3 )   Up        Fa1/0
Session state is UP and not using echo function.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 150(0), Hello (hits): 50(5735)
```

```
Rx Count: 5731, Rx Interval (ms) min/max/avg: 32/72/49 last: 32 ms ago
Tx Count: 5740, Tx Interval (ms) min/max/avg: 40/64/50 last: 44 ms ago
Registered protocols: EIGRP
Uptime: 00:04:45
Last packet: Version: 0
            - Diagnostic: 0
             I Hear You bit: 1      - Demand bit: 0
             Poll bit: 0            - Final bit: 0
             Multiplier: 3          - Length: 24
             My Discr.: 6           - Your Discr.: 3
             Min tx interval: 50000    - Min rx interval: 50000
             Min Echo interval: 0
```

下の図は、デバイス B のファスト イーサネット インターフェイス 1/0 に障害が発生したことを示しています。デバイス B でファスト イーサネット インターフェイス 1/0 をシャット ダウンした場合、デバイス A とデバイス B の対応する BFD セッションの BFD 統計情報が少なくなります。



デバイス B のファスト イーサネット インターフェイス 1/0 に障害が発生すると、BFD はデバイス A またはデバイス C の BFD ネイバーとしてデバイス B を検出しなくなります。この例では、デバイス B でファスト イーサネット インターフェイス 1/0 が管理的上の理由でシャットダウンされています。

デバイス A での **show bfd neighbors** コマンドによる次の出力では、EIGRP ネットワークのデバイス A の唯一の BFD ネイバーが表示されます。この出力では、対応するコマンド出力が太字で表示されています。

```
DeviceA# show bfd neighbors
OurAddr       NeighAddr

    LD/RD  RH/RS   Holdown(mult)  State     Int
172.16.1.1    172.16.1.3

    5/3    1(RH)     134   (3 )   Up        Fa1/0
```

デバイス C での **show bfd neighbors** コマンドによる次の出力でも、EIGRP ネットワークのデバイス C の唯一の BFD ネイバーが表示されます。この出力では、対応するコマンド出力が太字で表示されています。

```
DeviceC# show bfd neighbors

OurAddr       NeighAddr
```

```
   LD/RD RH  Holdown(mult)  State     Int
172.16.1.3    172.16.1.1

    3/5  1   114  (3 )      Up        Fa1/0
```

例：OSPF ネットワークでの BFD の設定

次に、OSPF インターフェイスで BFD を設定する例を示します。次の例では、デバイス A とデバイス B でシンプルな OSPF ネットワークが構成されています。デバイス A のファストイーサネット インターフェイス 1/0 はデバイス B のファスト イーサネット インターフェイス 6/0 と同じネットワークに接続されています。グローバル コンフィギュレーション モードで始まるこの例には、BFD の設定が示されています。デバイス A と B に対して、OSPF プロセスに関連付けられたすべてのインターフェイスで、BFD がグローバルに設定されます。

デバイス A の設定

```
!
interface Fast Ethernet 0/1
ip address 172.16.10.1 255.255.255.0
bfd interval 50 min_rx 50 multiplier 3
!
interface Fast Ethernet 3/0.1
ip address 172.17.0.1 255.255.255.0
!
router ospf 123
log-adjacency-changes detail
network 172.16.0.0 0.0.0.255 area 0
network 172.17.0.0 0.0.0.255 area 0
bfd all-interfaces
```

デバイス B の設定

```
!
interface Fast Ethernet 6/0
 ip address 172.16.10.2 255.255.255.0
 bfd interval 50 min_rx 50 multiplier 3
!
interface Fast Ethernet 6/1
 ip address 172.18.0.1 255.255.255.0
!
router ospf 123
 log-adjacency-changes detail
 network 172.16.0.0 0.0.255.255 area 0
 network 172.18.0.0 0.0.255.255 area 0
 bfd all-interfaces
```

show bfd neighbors details コマンドによる出力で、BFD セッションが作成され、BFD サポートに対して OSPF が登録されることを確認できます。

デバイス A

```
DeviceA# show bfd neighbors details

OurAddr       NeighAddr     LD/RD RH  Holdown(mult)  State     Int
172.16.10.1   172.16.10.2    1/2  1   532  (3 )      Up        Fa0/1
```

```
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5
Received MinRxInt: 1000, Received Multiplier: 3
Holdown (hits): 600(22), Hello (hits): 200(84453)
Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332 last: 68 ms ago
Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196 last: 192 ms ago
Registered protocols: OSPF


Uptime: 02:18:49
Last packet: Version: 0
            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 2          - Your Discr.: 1
             Min tx interval: 50000    - Min rx interval: 1000
             Min Echo interval: 0
```

デバイス B からの **show bfd neighbors details** コマンドによる出力で、BFD セッションが作成されたことを確認できます。

デバイス B

```
DeviceB# attach 6
Entering Console for 8 Port Fast Ethernet in Slot: 6
Type "exit" to end this session
Press RETURN to get started!

Device> show bfd neighbors details
Cleanup timer hits: 0
OurAddr       NeighAddr     LD/RD RH  Holdown(mult)  State     Int
172.16.10.2   172.16.10.1    8/1  1   1000 (5 )      Up        Fa6/0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 1000, Multiplier: 3
Received MinRxInt: 200000, Received Multiplier: 5
Holdown (hits): 1000(0), Hello (hits): 200(5995)
Rx Count: 10126, Rx Interval (ms) min/max/avg: 152/248/196 last: 0 ms ago
Tx Count: 5998, Tx Interval (ms) min/max/avg: 204/440/332 last: 12 ms ago
Last packet: Version: 0            - Diagnostic: 0
I Hear You bit: 1     - Demand bit: 0
Poll bit: 0           - Final bit: 0
Multiplier: 5         - Length: 24
My Discr.: 1          - Your Discr.: 8
Min tx interval: 200000    - Min rx interval: 200000
Min Echo interval: 0
Uptime: 00:33:13
SSO Cleanup Timer called: 0
SSO Cleanup Action Taken: 0
Pseudo pre-emptive process count: 239103 min/max/avg: 8/16/8 last: 0 ms ago
IPC Tx Failure Count: 0
IPC Rx Failure Count: 0
Total Adjs Found: 1
```

show ip ospf コマンドによる出力で、BFD が OSPF に対してイネーブルになっていることを確認できます。

デバイス A

```
DeviceA# show ip ospf
```

```
 Routing Process "ospf 123" with ID 172.16.10.1
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 0. Checksum Sum 0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 External flood list length 0
 BFD is enabled


    Area BACKBONE(0)
        Number of interfaces in this area is 2 (1 loopback)
        Area has no authentication
        SPF algorithm last executed 00:00:08.828 ago
        SPF algorithm executed 9 times
        Area ranges are
        Number of LSA 3. Checksum Sum 0x028417
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

デバイス B

```
DeviceB# show ip ospf

 Routing Process "ospf 123" with ID 172.18.0.1
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 0. Checksum Sum 0x0
 Number of opaque AS LSA 0. Checksum Sum 0x0
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 Number of areas transit capable is 0
 External flood list length 0
 BFD is enabled
```

```
    Area BACKBONE(0)
        Number of interfaces in this area is 2 (1 loopback)
        Area has no authentication
        SPF algorithm last executed 02:07:30.932 ago
        SPF algorithm executed 7 times
        Area ranges are
        Number of LSA 3. Checksum Sum 0x28417
        Number of opaque link LSA 0. Checksum Sum 0x0
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

show ip ospf interface コマンドによる出力で、デバイス A とデバイス B を接続しているインターフェイスで OSPF に対して BFD がイネーブルになっていることを確認できます。

デバイス A

```
DeviceA# show ip ospf interface Fast Ethernet 0/1

show ip ospf interface Fast Ethernet 0/1
Fast Ethernet0/1 is up, line protocol is up
Internet Address 172.16.10.1/24, Area 0
Process ID 123, Router ID 172.16.10.1, Network Type BROADCAST, Cost: 1
Transmit Delay is 1 sec, State BDR, Priority 1, BFD enabled
Designated Router (ID) 172.18.0.1, Interface address 172.16.10.2
Backup Designated router (ID) 172.16.10.1, Interface address 172.16.10.1
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
oob-resync timeout 40
Hello due in 00:00:03
Supports Link-local Signaling (LLS)
Index 1/1, flood queue length 0
Next 0x0(0)/0x0(0)
Last flood scan length is 1, maximum is 1
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 1, Adjacent neighbor count is 1
Adjacent with neighbor 172.18.0.1  (Designated Router)
Suppress hello for 0 neighbor(s)
```

デバイス B

```
DeviceB# show ip ospf interface Fast Ethernet 6/1

Fast Ethernet6/1 is up, line protocol is up
Internet Address 172.18.0.1/24, Area 0
Process ID 123, Router ID 172.18.0.1, Network Type BROADCAST, Cost: 1
Transmit Delay is 1 sec, State DR, Priority 1, BFD enabled
Designated Router (ID) 172.18.0.1, Interface address 172.18.0.1
No backup designated router on this network
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
oob-resync timeout 40
Hello due in 00:00:01
Supports Link-local Signaling (LLS)
Index 1/1, flood queue length 0
Next 0x0(0)/0x0(0)
Last flood scan length is 0, maximum is 0
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 0, Adjacent neighbor count is 0
Suppress hello for 0 neighbor(s)
```

例：スタティック ルーティングに対する BFD サポートの設定

次の例では、ネットワークはデバイス A とデバイス B で構成されています。デバイス A のシリアル インターフェイス 2/0 は、デバイス B のシリアル インターフェイス 2/0 と同じネットワークに接続されています。BFD セッションを起動するには、デバイス B を設定する必要があります。

デバイス A

```
configure terminal
interface Serial 2/0
ip address 10.201.201.1 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.2
ip route 10.0.0.0 255.0.0.0 Serial 2/0 10.201.201.2
```

デバイス B

```
configure terminal
interface Serial 2/0
ip address 10.201.201.2 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.1
ip route 10.1.1.1 255.255.255.255 Serial 2/0 10.201.201.1
```

デバイス B のスタティック ルートが単独で存在していて、10.201.201.1 と 10.201.201.2 の間で BFDセッションをイネーブルにすることに注意してください。設定する必要のある有益なスタティック ルートがない場合、パケットの転送に影響しないプレフィックス、たとえば、ローカルで設定されたループバック インターフェイスを選択します。

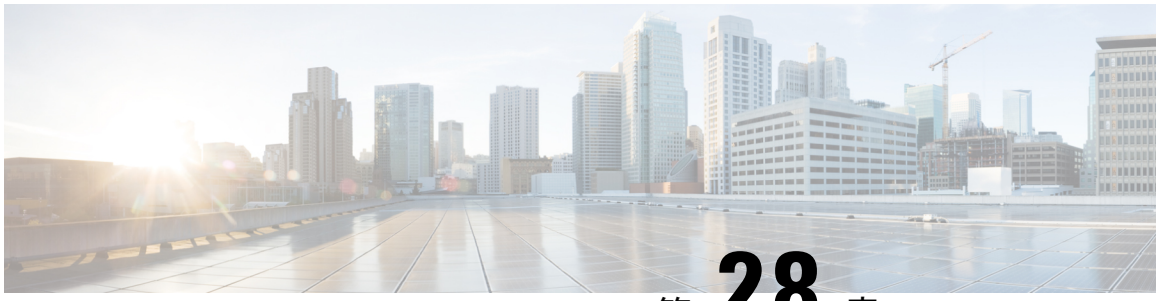
次の例では、BFD グループ testgroup のイーサネット インターフェイス 0/0 を介して 209.165.200.225 に到達するアクティブなスタティック BFD 設定があります。設定されたスタティック BFD によってトラッキングされるスタティック ルートが設定されるとすぐに、単一のホップ BFD セッションがイーサネット インターフェイス 0/0 を介して 209.165.200.225 に開始されます。BFD セッションが正常に確立されると、プレフィックス 10.0.0.0/8 が RIB に追加されます。

```
configure terminal
ip route static bfd Ethernet 0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225
```

次の例では、イーサネット インターフェイス 0/0.1001 を介した 209.165.200.226 への BFD セッションがグループ testgroup を使用するようにマークされます。つまり、この設定はパッシブなスタティック BFD です。2 つ目のスタティック BFD 設定によってトラッキングされるスタティック ルートがあるものの、209.165.200.226 に対する BFD セッションはイーサネット インターフェイス 0/0.1001 を介しては開始されません。プレフィックス 10.1.1.1/8 と 10.2.2.2/8 の存在は、アクティブなスタティック BFD セッション（イーサネット インターフェイス 0/0 209.165.200.225）によって制御されます。

```
configure terminal
ip route static bfd Ethernet 0/0 209.165.200.225 group testgroup
```

```
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225
ip route static bfd Ethernet 0/0.1001 209.165.200.226 group testgroup passive
ip route 10.1.1.1 255.255.255.224 Ethernet 0/0.1001 209.165.200.226
ip route 10.2.2.2 255.255.255.224 Ethernet 0/0.1001 209.165.200.226
```



第 28 章

EtherChannel の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

EtherChannel の制約事項

- EtherChannel のすべてのポートは同じ VLAN に割り当てるか、またはトランク ポートとして設定する必要があります。
- EtherChannel のポートがトランク ポートとして設定されている場合、すべてのポートを同じモード（Inter-Switch Link（ISL）または IEEE 802.1Q）で設定する必要があります。
- Port Aggregation Protocol（PAgP）は単一スイッチの EtherChannel 設定でのみイネーブルにできます。PAgP はクロススタック EtherChannel ではイネーブルにできません。

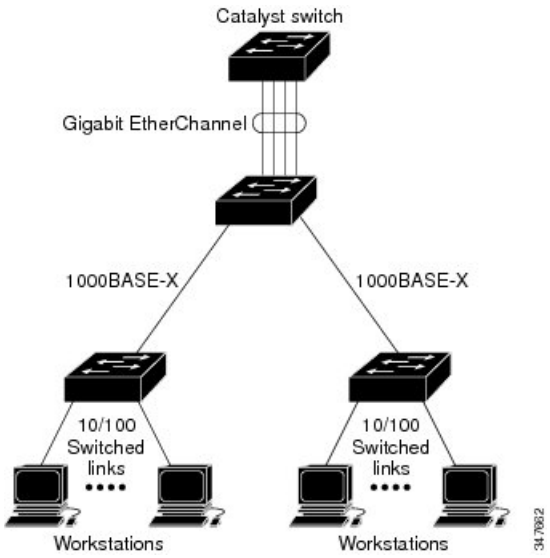
EtherChannel について

EtherChannel の概要

EtherChannel は、スイッチ、ルータ、およびサーバ間にフォールトトレラントな高速リンクを提供します。EtherChannel を使用して、ワイヤリング クローゼットとデータセンター間の帯域幅を増やすことができます。さらに、ボトルネックが発生しやすいネットワーク上のあらゆる場所に EtherChannel を配置できます。EtherChannel は、他のリンクに負荷を再分散させることによって、リンク切断から自動的に回復します。リンク障害が発生した場合、EtherChannel は自動的に障害リンクからチャネル内の他のリンクにトラフィックをリダイレクトします。

EtherChannel は、単一の論理リンクにバンドルする個別のイーサネット リンクで構成されます。

図 *47 :* 一般的な *EtherChannel* 構成



各 EtherChannel は、互換性のある設定のイーサネット ポートを 8 つまで使用して構成できます。

EtherChannel のモード

EtherChannel は、ポート集約プロトコル（PAgP）、Link Aggregation Control Protocol（LACP）、または On のいずれかのモードに設定できます。EtherChannel の両端は同じモードで設定します。

- EtherChannel の一方の端を PAgP または LACP モードに設定すると、システムはもう一方の端とネゴシエーションし、アクティブにするポートを決定します。リモート ポートが EtherChannel とネゴシエーションができない場合、ローカル ポートは独立ステートにな

り、他の単一リンクと同様にデータ トラフィックを引き続き伝送します。ポート設定は変更されませんが、ポートは EtherChannel に参加しません。

- EtherChannel を **on** モードに設定すると、ネゴシエーションは実行されません。スイッチはEtherChannel内で互換性のあるすべてのポートを強制的にアクティブにします。チャネルの他端（その他のスイッチ上）も**on**モードに設定する必要があります。そうでないと、パケット損失が発生する可能性があります。

Devices上の EtherChannel

device上、スタックの単一device上、またはスタックの複数devices上（クロススタック EtherChannel とも呼ぶ）で EtherChannel を作成できます。

EtherChannel リンクのフェールオーバー

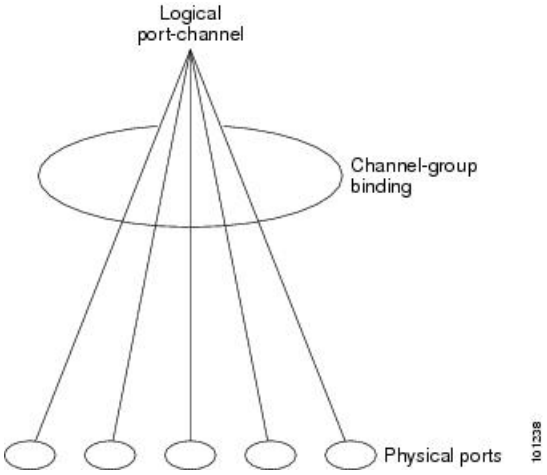
EtherChannel 内のリンクで障害が発生すると、それまでその障害リンクで伝送されていたトラフィックがEtherChannel内の残りのリンクに切り替えられます。スイッチでトラップがイネーブルになっている場合、スイッチ、EtherChannel、および失敗したリンクを区別したトラップが送信されます。EtherChannel の 1 つのリンク上の着信ブロードキャストおよびマルチキャスト パケットは、EtherChannel の他のリンクに戻らないようにブロックされます。

チャネル グループおよびポートチャネル インターフェイス

EtherChannel は、チャネル グループとポートチャネル インターフェイスから構成されます。チャネル グループはポートチャネル インターフェイスに物理ポートをバインドします。ポートチャネル インターフェイスに適用した設定変更は、チャネル グループにまとめてバインドされるすべての物理ポートに適用されます。

図 *48 :* 物理ポート、チャネル グループおよびポートチャネル インターフェイスの関係

channel-group コマンドは、物理ポートおよびポートチャネル インターフェイスをまとめてバインドします。各 EtherChannel には 1 ～ までの番号が付いたポートチャネル論理インターフェイスがあります。ポートチャネル インターフェイス番号は、**channel-group** インターフェイス コンフィギュレーション コマンドで指定した番号に対応しています。



- レイヤ 2 ポートの場合は、**channel-group** インターフェイス コンフィギュレーション コマンドを使用して、ポートチャネル インターフェイスを動的に作成します。

  また、**interface port-channel** *port-channel-number* グローバル コンフィギュレーション コマンドを使用して、ポートチャネル論理インターフェイスを手動で作成することもできます。ただし、その場合、論理インターフェイスを物理ポートにバインドするには、**channel-group** *channel-group-number* コマンドを使用する必要があります。*channel-group-number* は *port-channel-number* と同じ値に設定することも、違う値を使用することもできます。新しい番号を使用した場合、**channel-group** コマンドは動的に新しいポートチャネルを作成します。

Port Aggregation Protocol; ポート集約プロトコル

ポート集約プロトコル（PAgP）はシスコ独自のプロトコルで、Cisco devicesおよび PAgP をサポートするベンダーによってライセンス供与されたdevicesでのみ稼働します。PAgP を使用すると、イーサネット ポート間で PAgP パケットを交換することにより、EtherChannel を自動的に作成できます。

device または device スタックは PAgP を使用することによって、PAgP をサポートできるパートナーの識別情報、および各ポートの機能を学習します。次に、設定が類似している（device 内の単一 上の）ポートを、単一の論理リンク（チャネルまたは集約ポート）に動的にグループ化します。設定が類似しているポートをグループ化する場合の基準は、ハードウェア、管理、およびポート パラメータ制約です。たとえば、PAgP は速度、デュプレックス モード、ネイティブ VLAN、VLAN 範囲、トランキング ステータス、およびトランキング タイプが同じポートをグループとしてまとめます。リンクを EtherChannel にグループ化した後で、PAgP は単一device ポートとして、スパニングツリーにそのグループを追加します。

PAgP モード

PAgP モードは、PAgP ネゴシエーションを開始する PAgP パケットをポートが送信できるか、または受信した PAgP パケットに応答できるかを指定します。

表 *51 : EtherChannel PAgP* モード

モード	説明
auto	ポートをパッシブ ネゴシエーション ステートにします。この場合、ポートは受信する PAgP パケットに応答しますが、PAgP パケット ネゴシエーションを開始することはありません。これにより、PAgP パケットの送信は最小限に抑えられます。
desirable	ポートをアクティブ ネゴシエーション ステートにします。この場合、ポートは PAgP パケットを送信することによって、相手ポートとのネゴシエーションを開始します。

スイッチポートは、**auto** モードまたは **desirable** モードに設定された相手ポートとだけ PAgP パケットを交換します。**on** モードに設定されたポートは、PAgP パケットを交換しません。

auto モードおよび **desirable** モードはともに、相手ポートとネゴシエーションして、ポート速度などの条件に基づいて（レイヤ 2 EtherChannel の場合は、トランクステートおよび VLAN 番号などの基準に基づいて）、ポートで EtherChannel を形成できるようにします。

PAgP モードが異なっていても、モード間で互換性がある限り、ポートは EtherChannel を形成できます。次に例を示します。

- **desirable** モードのポートは、**desirable** または **auto** モードの別のポートと EtherChannel を形成できます。
- **auto** モードのポートは、**desirable** モードの別のポートと EtherChannel を形成できます。

両ポートとも LACP ネゴシエーションを開始しないため、**auto** モードのポートは、**auto** モードの別のポートと EtherChannel を形成することはできません。

サイレント モード

PAgP 対応のデバイスにスイッチを接続する場合、**non-silent** キーワードを使用すると、スイッチポートを非サイレント動作用に設定できます。**auto** モードまたは **desirable** モードとともに **non-silent** モードを指定しなかった場合は、サイレントモードが指定されていると見なされます。

サイレント モードを使用するのは、PAgP 非対応で、かつほとんどパケットを送信しないデバイスにスイッチを接続する場合です。サイレント パートナーの例は、トラフィックを生成しないファイル サーバ、またはパケット アナライザなどです。この場合、サイレント パートナーに接続された物理ポート上で PAgP を稼働させると、このスイッチ ポートが動作しなくなります。ただし、サイレントを設定すると、PAgP が動作してチャネル グループにポートを結合し、このポートが伝送に使用されます。

PAgP 学習方式およびプライオリティ

ネットワーク デバイスは、PAgP 物理ラーナーまたは集約ポート ラーナーに分類されます。物理ポートによってアドレスを学習し、その知識に基づいて送信を指示するデバイスは物理ラーナーです。集約（論理）ポートによってアドレスを学習するデバイスは、集約ポート ラーナーです。学習方式は、リンクの両端で同一の設定にする必要があります。

デバイスとそのパートナーが両方とも集約ポート ラーナーの場合、論理ポート チャネル上のアドレスを学習します。デバイスは EtherChannel のいずれかのポートを使用することによって、送信元にパケットを送信します。集約ポート ラーナーの場合、どの物理ポートにパケットが届くかは重要ではありません。

PAgP は、パートナー デバイスが物理ラーナーの場合およびローカル デバイスが集約ポート ラーナーの場合には自動検出できません。したがって、物理ポートでアドレスを学習するには、ローカル デバイスに手動で学習方式を設定する必要があります。また、負荷の分散方式を送信元ベース分散に設定して、指定された送信元 MAC アドレスが常に同じ物理ポートに送信されるようにする必要もあります。

グループ内の 1 つのポートですべての伝送を行うように設定して、他のポートをホット スタンバイに使用することもできます。選択された 1 つのポートでハードウェア信号が検出されなくなった場合は、数秒以内に、グループ内の未使用のポートに切り替えて動作させることができます。パケット伝送用に常に選択されるように、ポートを設定するには、 **pagp port-priority** インターフェイス コンフィギュレーション コマンドを使用してプライオリティを変更します。プライオリティが高いほど、そのポートが選択される可能性が高まります。

（注） CLI で **physical-port** キーワードを指定した場合でも、device がサポートするのは、集約ポート上でのアドレス ラーニングのみです。**pagp learn-method** コマンドおよび **pagp port-priority** コマンドは、device のハードウェアには影響を及ぼしませんが、Catalyst 1900 スイッチなど、物理ポートによるアドレス ラーニングのみをサポートしているデバイスと PAgP の相互運用性を確保するために必要です。

device のリンク パートナーが物理ラーナーである場合、**pagp learn-method physical-port** インターフェイス コンフィギュレーション コマンドを使用して物理ポート ラーナーとして device を設定することを推奨します。また、**port-channel load-balance src-mac** グローバル コンフィギュレーション コマンドを使用して、送信元 MAC アドレスに基づいて負荷分散方式を設定することを推奨します。すると、deviceは送信元アドレスを学習した EtherChannel 内の同じポートを使用して、物理ラーナーにパケットを送信します。この状況では、**pagp learn-method** コマンドのみを使用します。

PAgP と仮想スイッチとの相互作用およびデュアルアクティブ検出

仮想スイッチは、仮想スイッチ リンク（VSL）により接続された複数のコア スイッチであり、それらのスイッチ間で制御情報とデータ トラフィックを伝送します。スイッチのうちの 1 つはアクティブ モードです。その他のスイッチはスタンバイ モードです。冗長性のため、リモート スイッチはリモート サテライト リンク（RSL）によって仮想スイッチに接続されます。

2つのスイッチ間のVSLに障害が発生すると、一方のスイッチは他方のスイッチのステータスを認識しません。両方のスイッチがアクティブモードになり、ネットワークを、重複したコンフィギュレーション（IP アドレスおよびブリッジ ID の重複を含む）を伴うデュアルアクティブの状態にする可能性があります。ネットワークがダウンする場合もあります。

デュアルアクティブの状態を防止するために、コア スイッチはPAgPプロトコルデータ ユニット（PDU）を RSL を介してリモート スイッチに送信します。PAgP PDU はアクティブ スイッチを識別し、リモート スイッチは、コア スイッチが同期化するようにPDUをコア スイッチに転送します。アクティブ スイッチに障害が発生した場合、またはアクティブ スイッチがリセットされた場合は、スタンバイ スイッチがアクティブスイッチの役割を引き継ぎます。VSL がダウンした場合は、1つのコア スイッチが他のコア スイッチのステータスを認識し、その状態を変更しません。

PAgP と他の機能との相互作用

ダイナミック トランキング プロトコル（DTP）および Cisco Discovery Protocol（CDP）は、EtherChannelの物理ポートを使用してパケットを送受信します。トランク ポートは、番号が最も小さい VLAN 上で PAgP プロトコル データ ユニット（PDU）を送受信します。

レイヤ 2 EtherChannel では、チャネル内で最初に起動するポートが EtherChannel に MAC アドレスを提供します。このポートがバンドルから削除されると、バンドル内の他のポートの1つが EtherChannel に MAC アドレスを提供します。

PAgP が PAgP PDU を送受信するのは、PAgP が auto モードまたは desirable モードでイネーブルになっている、稼働状態のポート上だけです。

Link Aggregation Control Protocol（LACP）

LACP は IEEE 802.3ad で定義されており、Cisco devicesが IEEE 802.3ad プロトコルに適合したdevices間のイーサネット チャネルを管理できるようにします。LACP を使用すると、イーサネット ポート間でLACPパケットを交換することにより、EtherChannelを自動的に作成できます。

device または device スタックは LACP を使用することによって、LACP をサポートできるパートナーの識別情報、および各ポートの機能を学習します。次に、設定が類似しているポートを単一の倫理リンク（チャネルまたは集約ポート）に動的にグループ化します。設定が類似しているポートをグループ化する場合の基準は、ハードウェア、管理、およびポートパラメータ制約です。たとえば、LACPは速度、デュプレックス モード、ネイティブ VLAN、VLAN 範囲、トランキング ステータス、およびトランキング タイプが同じポートをグループとしてまとめます。リンクをまとめて EtherChannel を形成した後で、LACP は単一device ポートとして、スパニングツリーにそのグループを追加します。

ポート チャネル内のポートの独立モード動作が変更されます。CSCtn96950 では、デフォルトでスタンドアロン モードが有効になっています。LACP ピアから応答が受信されない場合、ポート チャネル内のポートは中断状態に移動されます。

LACP モード

LACP モードでは、ポートが LACP パケットを送信できるか、LACP パケットの受信のみができるかどうかを指定します。

表 *52 : EtherChannel LACP* モード

モード	説明
active	ポートをアクティブ ネゴシエーション ステートにします。この場合、ポートは LACP パケットを送信することによって、相手ポートとのネゴシエーションを開始します。
passive	ポートはパッシブ ネゴシエーション ステートになります。この場合、ポートは受信する LACP パケットに応答しますが、LACP パケット ネゴシエーションを開始することはありません。これにより、LACPパケットの送信を最小限に抑えます。

active モードおよび **passive LACP** モードはともに、相手ポートとネゴシエーションして、ポート速度などの条件に基づいて（レイヤ 2 EtherChannel の場合は、トランクステートおよびVLAN番号などの基準に基づいて）、ポートで EtherChannel を形成できるようにします。

LACP モードが異なっていても、モード間で互換性がある限り、ポートは EtherChannel を形成できます。次に例を示します。

- **active** モードのポートは、**active** または **passive** モードの別のポートと EtherChannel を形成できます。
- 両ポートとも LACP ネゴシエーションを開始しないため、**passive** モードのポートは、**passive** モードの別のポートと EtherChannel を形成することはできません。

LACP と他の機能との相互作用

DTP および CDP は、EtherChannel の物理ポートを介してパケットを送受信します。トランクポートは、番号が最も小さい VLAN 上で LACP PDU を送受信します。

レイヤ 2 EtherChannel では、チャネル内で最初に起動するポートが EtherChannel に MAC アドレスを提供します。このポートがバンドルから削除されると、バンドル内の他のポートの 1 つが EtherChannel に MAC アドレスを提供します。

LACP が LACP PDU を送受信するのは、LACP が active モードまたは passive モードでイネーブルになっている稼働状態のポートとの間だけです。

EtherChannel の On モード

EtherChannel **on** モードは、EtherChannel を手動で設定するために使用できます。**on** モードでは、ネゴシエーションを行わずにポートは強制的に EtherChannel に参加されます。**on** モードは、リモートデバイスが PAgP または LACP をサポートしていない場合に役立つことがありま

す。**on** モードでは、リンクの両端の devices が **on** モードに設定されている場合のみ、使用可能な EtherChannel が存在します。

同じチャネルグループ内で**on**モードに設定されているポートは、互換性のあるポート特性（速度やデュプレックスなど）を備えている必要があります。互換性のないポートは、**on** モードに設定されている場合でも、一時停止されます。

注意 **on** モードを使用する場合は、注意する必要があります。これは手動の設定であり、EtherChannel の両端のポートには、同一の設定が必要です。グループの設定を誤ると、パケット損失またはスパニングツリーループが発生することがあります。

ロードバランシングおよび転送方式

EtherChannel は、フレーム内のアドレスに基づいて形成されたバイナリ パターンの一部を、チャネル内の 1 つのリンクを選択する数値に縮小することによって、チャネル内のリンク間でトラフィックのロードバランシングを行います。MAC アドレス、IP アドレス、送信元アドレス、宛先アドレス、または送信元と宛先両方のアドレスに基づいた負荷分散など、複数の異なるロードバランシング モードから 1 つを指定できます。選択したモードは、device上で設定されているすべての EtherChannel に適用されます。

（注） レイヤ 3 等コスト マルチ パス（ECMP）のロード バランシングは、送信元 IP アドレス、宛先 IP アドレス、送信元ポート、宛先ポート、およびレイヤ 4 プロトコルに基づいています。フラグメント化されたパケットは、これらのパラメータを使用して計算されたアルゴリズムに基づいて 2 つの異なるリンクで処理されます。これらのパラメータのいずれかを変更すると、ロード バランシングが実行されます。

MAC アドレス転送

送信元 MAC アドレス転送の場合、EtherChannel に転送されたパケットは、着信パケットの送信元 MAC アドレスに基づいてチャネル ポート間で分配されます。したがって、ロード バランシングを行うために、送信元ホストが異なるパケットはそれぞれ異なるチャネル ポートを使用しますが、送信元ホストが同じパケットは同じチャネル ポートを使用します。

宛先 MAC アドレス転送の場合、EtherChannel に転送されたパケットは、着信パケットの宛先ホストの MAC アドレスに基づいてチャネル ポート間で分配されます。したがって、宛先が同じパケットは同じポートに転送され、宛先の異なるパケットはそれぞれ異なるチャネル ポートに転送されます。

送信元および宛先 MAC アドレス転送の場合、EtherChannel に転送されたパケットは、送信元および宛先の両方の MAC アドレスに基づいてチャネル ポート間で分配されます。この転送方式は、負荷分散の送信元 MAC アドレス転送方式と宛先 MAC アドレス転送方式を組み合わせたものです。特定のdeviceに対して送信元MACアドレス転送と宛先MACアドレス転送のどちらが適切であるかが不明な場合に使用できます。送信元および宛先 MAC アドレス転送の場

合、ホスト A からホスト B、ホスト A からホスト C、およびホスト C からホスト B に送信されるパケットは、それぞれ異なるチャネル ポートを使用できます。

IP アドレス転送

送信元 IP アドレスベース転送の場合、パケットは、着信パケットの送信元 IP アドレスに基づいて EtherChannel ポート間で分配されます。ロード バランシングを行うために、IP アドレスが異なるパケットはチャネルでそれぞれ異なるポートを使用しますが、IP アドレスが同じパケットはチャネルで同じポートを使用します。

宛先 IP アドレスベース転送の場合、パケットは着信パケットの宛先 IP アドレスに基づいて EtherChannel ポート間で分配されます。ロード バランシングを行うために、同じ送信元 IP アドレスから異なる宛先 IP アドレスに送信されるパケットは、チャネルの異なるチャネル ポートに送信できます。異なる送信元 IP アドレスから同じ宛先 IP アドレスに送信されるパケットは、常にチャネルの同じポートに送信されます。

送信元と宛先 IP アドレスベース転送の場合、パケットは着信パケットの送信元および宛先の両方の IP アドレスに基づいて EtherChannel ポート間で分配されます。この転送方式は、送信元 IP アドレスベース転送方式と宛先 IP アドレスベース転送方式を組み合わせたもので、特定のdeviceに対して送信元 IP アドレスベース転送と宛先 IP アドレスベース転送のどちらが適切であるか不明な場合に使用できます。この方式では、IP アドレス A から IP アドレス B に、IP アドレス A から IP アドレス C に、および IP アドレス C から IP アドレス B に送信されるパケットは、それぞれ異なるチャネル ポートを使用できます。

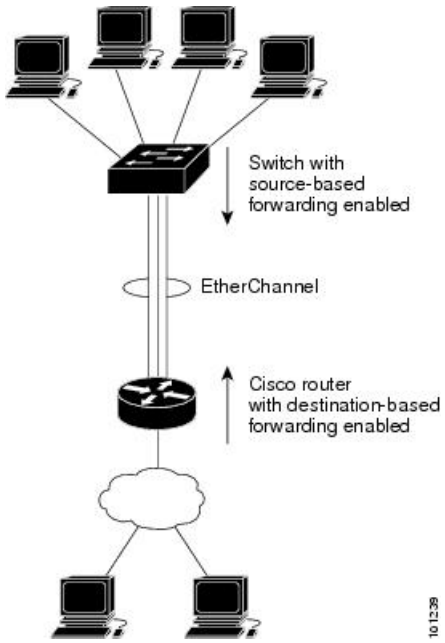
ロードバランシングの利点

ロードバランシング方式には異なる利点があるため、ネットワーク内のdeviceの位置、および負荷分散が必要なトラフィックの種類に基づいて特定のロードバランシング方式を選択する必要があります。

図 *49* : 負荷の分散および転送方式

次の図では、4 台のワーク ステーションの EtherChannel がルータと通信します。ルータは単一 MAC アドレス デバイスであるため、device EtherChannel で送信元ベース転送を行うことにより、deviceが、ルータで使用可能なすべての帯域幅を使用することが保証されます。ルータは、宛先アドレスベース転送を行うように設定されます。これは、多数のワークステーションで、

トラフィックがルータ EtherChannel から均等に分配されることになっているためです。



設定で一番種類が多くなるオプションを使用してください。たとえば、チャネル上のトラフィックが単一 MAC アドレスを宛先とする場合、宛先 MAC アドレスを使用すると、チャネル内の同じリンクが常に選択されます。ただし、送信元アドレスまたはIPアドレスを使用した方が、ロードバランシングの効率がよくなる場合があります。

EtherChannel ロード延期の概要

Instant Access システムでは、EtherChannel ロード遅延機能により、ポートをポート チャネルにバンドルすることができますが、そのポートにはグループ マスク値が割り当てられません。これにより、トラフィックが新規インスタント アクセス スタック メンバーに転送されるのを回避し、ステートフル スイッチオーバー（SSO）後のデータ損失を抑えることができます。

Cisco Catalyst Instant Access は、ディストリビューション スイッチとアクセス レイヤ スイッチを包括する単一のネットワーク タッチ ポイントと単一の設定ポイントを作成します。Instant Access により、物理的なディストリビューション スイッチとアクセス レイヤ スイッチを、単一の設定、管理、およびトラブルシューティング ポイントを備えた単一の論理エンティティにマージすることができます。次の図は、ポート チャネル経由でスタック構成クライアント（Member 1 および Member 2）に接続されているスイッチ（Catalyst 2960-X シリーズ スイッチ）と Instant Access システムが通信するサンプル ネットワークを表しています。

EtherChannel ロード延期機能が設定されている状態で、新しい Instant Access クライアント スタック メンバーが始動すると、この新規参加スタック メンバーのポートはポート チャネルにバンドルされます。移行期間中は、データ パスがディストリビューション スイッチ（Catalyst 6000 シリーズ スイッチ）に完全には確立されず、アクセス レイヤ スイッチ（Catalyst 2960-X シリーズ スイッチが）から送信されたトラフィックは未確立のポートに到達するので、トラフィックが失われます。

ポートチャネルでロードシェアリング延期が有効な場合、メンバーポートのロードシェアリングの割り当ては、**port-channel load-defer** コマンドによってグローバルに設定された期間の分だけ遅延されます。延期期間中、延期メンバ ポートのロード シェアは 0 に設定されます。この状態では、延期ポートによるデータおよびコントロール トラフィックの受信と、コントロール トラフィックの送信は可能ですが、ポートがデータ トラフィックを仮想スイッチング システム（VSS）に送信することはできません。グローバル延期タイマーの期限切れに伴い、延期メンバ ポートは延期状態を終了し、ポートは通常に設定されたロード シェアと認識するようになります。

ロードシェアの延期は、ポート チャネルの少なくとも 1 つのメンバー ポートがゼロ以外のロード シェアで現時点においてアクティブになっている場合にだけ適用されます。ロード シェアの延期をイネーブルにされたポートが EtherChannel に対する最初のメンバである場合、延期機能は適用されず、ポートは即座にトラフィックを転送します。

この機能はポート チャネル単位で有効になります。ただし、ロード延期タイマーは、ポート チャネル単位ではなくグローバルに設定されます。その結果、新しいポートがバンドルされても、すでに実行中の場合はタイマーがスタートしません。他のポートがすでに延期期間に入っていれば、新しいポートも、その残り時間の間だけ延期されます。

ロード延期は、いずれか 1 つの延期対象ポート チャネルのメンバーがバンドル解除されると、すぐに停止します。その結果、延期期間中にバンドル解除が発生した場合、延期されていたすべてのポートにグループマスクが割り当てられます。

（注） スタック メンバー スイッチでこの機能の有効化を試みると、次のメッセージが表示されます。

```
Load share deferral is supported only on stand-alone stack.
```

EtherChannel のデフォルト設定

EtherChannel のデフォルト設定を、次の表に示します。

表 *53 : EtherChannel* のデフォルト設定

機能	デフォルト設定
チャネル グループ	割り当てなし
ポートチャネル論理インターフェイス	未定義
PAgP モード	デフォルトなし
PAgP 学習方式	すべてのポートで集約ポート ラーニング
PAgP プライオリティ	すべてのポートで 128
LACP モード	デフォルトなし

機能	デフォルト設定
LACP 学習方式	すべてのポートで集約ポート ラーニング
LACP ポート プライオリティ	すべてのポートで 32768
LACP システム プライオリティ	32768
LACP システム ID	LACP システムのプライオリティおよびスイッチまたはスタックの MAC アドレス
ロード バランシング	着信パケットの送信元MACアドレスに基づいてスイッチ上で負荷を分散

EtherChannel 設定時の注意事項

EtherChannelポートを正しく設定していない場合は、ネットワーク ループおよびその他の問題を回避するために、一部の EtherChannel インターフェイスが自動的にディセーブルになります。設定上の問題を回避するために、次の注意事項に従ってください。

- PAgP EtherChannelは、同じタイプのイーサネット ポートを8つまで使用して設定します。
- 同じタイプのイーサネット ポートを最大で 16 個備えた LACP EtherChannel を設定してください。最大 8 つのポートを active モードに、最大 8 つのポートを standby モードにできます。
- EtherChannel内のすべてのポートを同じ速度および同じデュプレックス モードで動作するように設定します。
- EtherChannel 内のすべてのポートをイネーブルにします。**shutdown** インターフェイス コンフィギュレーション コマンドを使用して無効にされたEtherChannel内のポートはリンク障害として扱われ、そのトラフィックはEtherChannel内の残りのポートのいずれかに転送されます。
- グループを初めて作成した際には、そのグループに最初に追加されたポートのパラメータ設定値をすべてのポートが引き継ぎます。次のパラメータのいずれかで設定を変更した場合は、グループ内のすべてのポートでも変更する必要があります。
  - 許可 VLAN リスト
  - 各 VLAN のスパニングツリー パス コスト
  - 各 VLAN のスパニングツリー ポート プライオリティ
  - スパニングツリー PortFast の設定
- 1つのポートが複数のEtherChannelグループのメンバになるように設定しないでください。
- EtherChannel は、PAgP と LACP の両方のモードには設定しないでください。PAgP および LACP を実行している EtherChannel グループは、同じ device 上、またはスタック 内の異

なる devices 上で共存できます。個々の EtherChannel グループは PAgP または LACP のいずれかを実行できますが、相互運用することはできません。

- アクティブまたはアクティブでない EtherChannel メンバであるポートを IEEE 802.1x ポートとして設定しないでください。EtherChannel ポートで IEEE 802.1x をイネーブルにしようとすると、エラー メッセージが表示され、IEEE 802.1x はイネーブルになりません。
- EtherChannel が device インターフェイスに設定されている場合は、**dot1x system-auth-control** グローバル コンフィギュレーション コマンドを使用して、device 上で IEEE 802.1x をグローバルにイネーブルにする前に、インターフェイスから EtherChannel 構成を削除します。

レイヤ 2 EtherChannel 設定時の注意事項

レイヤ 2 EtherChannels を設定する場合は、次の注意事項に従ってください。

- EtherChannel 内のすべてのポートを同じ VLAN に割り当てるか、またはトランクとして設定してください。複数のネイティブ VLAN に接続されるポートは、EtherChannel を形成できません。
- EtherChannel は、トランキング レイヤ 2 EtherChannel 内のすべてのポート上で同じ VLAN 許容範囲をサポートしています。VLAN 許容範囲が一致していないと、PAgP が **auto** モードまたは **desirable** モードに設定されていても、ポートは EtherChannel を形成しません。
- スパニングツリー パス コストが異なるポートは、設定上の矛盾がない限り、EtherChannel を形成できます。異なるスパニングツリー パス コストを設定すること自体は、EtherChannel を形成するポートの矛盾にはなりません。

Auto-LAG

Auto-LAG 機能は、スイッチに接続されたポートで EtherChannel を自動的に作成できる機能です。デフォルトでは、Auto-LAG がグローバルに無効にされ、すべてのポート インターフェイスで有効になっています。Auto-LAG は、グローバルに有効になっている場合にのみ、スイッチに適用されます。

Auto-LAG をグローバルに有効にすると、次のシナリオが可能になります。

- パートナー ポート インターフェイス上に EtherChannel が設定されている場合、すべてのポート インターフェイスが自動 EtherChannel の作成に参加します。詳細については、次の表「アクターとパートナー デバイス間でサポートされる *Auto-LAG* 設定」を参照してください。
- すでに手動 EtherChannel の一部であるポートは、自動 EtherChannel の作成に参加することはできません。
- Auto-LAG がすでに自動で作成された EtherChannel の一部であるポート インターフェイスで無効になっている場合、ポート インターフェイスは自動 EtherChannel からバンドル解除されます。

次の表に、アクターとパートナー デバイス間でサポートされる Auto-LAG 設定を示します。

表 *54*：アクターとパートナー デバイス間でサポートされる *Auto-LAG* 設定

アクター/パートナー	アクティブ	パッシブ	自動
アクティブ	対応	対応	対応
パッシブ	対応	非対応	対応
自動	対応	対応	対応

Auto-LAG をグローバルに無効にすると、自動で作成されたすべての Etherchannel が手動 EtherChannel になります。

既存の自動で作成された EtherChannel で設定を追加することはできません。追加するには、最初に **port-channel**<*channel-number*>**persistent** を実行して、手動 EtherChannel に変換する必要があります。

（注） Auto-LAG は自動 EtherChannel の作成に LACP プロトコルを使用します。一意のパートナー デバイスで自動的に作成できる EtherChannel は 1 つだけです。

Auto-LAG 設定時の注意事項

Auto-LAG 機能を設定するときには、次の注意事項に従ってください。

- Auto-LAG がグローバルで有効な場合、およびポート インターフェイスで有効な場合に、ポート インターフェイスを自動 EtherChannel のメンバーにしたくない場合は、ポート インターフェイスで Auto-LAG を無効にします。
- ポート インターフェイスは、すでに手動 EtherChannel のメンバーである場合、自動 EtherChannel にバンドルされません。自動 EtherChannel にバンドルされるようにするには、まずポート インターフェイスで手動 EtherChannel のバンドルを解除します。
- Auto-LAG が有効になり、自動 EtherChannel が作成されると、同じパートナー デバイスで複数の EtherChannel を手動で作成できます。ただし、デフォルトでは、ポートはパートナー デバイスで自動 EtherChannel の作成を試行します。
- Auto-LAG は、レイヤ 2 EtherChannel でのみサポートされています。レイヤ 3 インターフェイスおよびレイヤ 3 EtherChannel ではサポートされていません。
- Auto-LAG は、Cross-Stack EtherChannel でサポートされています。

EtherChannel の設定方法

EtherChannelの設定後、ポートチャネルインターフェイスに適用した設定変更は、そのポートチャネルインターフェイスに割り当てられたすべての物理ポートに適用されます。また、物理ポートに適用した設定変更は、設定を適用したポートだけに作用します。

レイヤ 2 EtherChannel の設定

レイヤ2 EtherChannelを設定するには、**channel-group**インターフェイス コンフィギュレーション コマンドを使用して、チャネルグループにポートを割り当てます。このコマンドにより、ポートチャネル論理インターフェイスが自動的に作成されます。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {**access** | **trunk**}
4. **switchport access vlan** *vlan-id*
5. **channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive**}
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/1`	物理ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。 指定できるインターフェイスは、物理ポートです。 PAgP EtherChannelの場合、同じタイプおよび速度のポートを 8 つまで同じグループに設定できます。 LACP EtherChannelの場合、同じタイプのイーサネット ポートを 16 まで設定できます。最大 8 つのポートを active モードに、最大 8 つのポートを standby モードにできます。

	コマンドまたはアクション	目的
ステップ 3	**switchport mode** {**access** | **trunk**} 例： `スイッチ(config-if)# switchport mode access`	すべてのポートをスタティックアクセス ポートとして同じ VLAN に割り当てるか、またはトランクとして設定します。 ポートをスタティックアクセス ポートとして設定する場合は、ポートを 1 つの VLAN にのみ割り当ててください。指定できる範囲は 1 ～ 4094 です。
ステップ 4	**switchport access vlan** *vlan-id* 例： `スイッチ(config-if)# switchport access vlan 22`	ポートをスタティックアクセス ポートとして設定する場合は、ポートを 1 つの VLAN にのみ割り当ててください。指定できる範囲は 1 ～ 4094 です。
ステップ 5	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive**} 例： `スイッチ(config-if)# channel-group 5 mode auto`	チャネル グループにポートを割り当て、PAgP モードまたは LACP モードを指定します。 **mode** には、次のキーワードのいずれか 1 つを選択します。 • **auto** –PAgP 装置が検出された場合に限り、PAgP をイネーブルにします。ポートをパッシブ ネゴシエーション ステートにします。この場合、ポートは受信する PAgP パケットに応答しますが、PAgP パケット ネゴシエーションを開始することはありません。。 • **desirable** –無条件に PAgP をイネーブルにします。ポートをアクティブ ネゴシエーション ステートにします。この場合、ポートは PAgP パケットを送信することによって、相手ポートとのネゴシエーションを開始します。。 • **on** – : PAgP または LACP を使用せずにポートが強制的にチャネル化されます。**on** モードでは、使用可能な EtherChannel が存在するのは、**on** モードのポートグループが、**on** モードの別のポートグループに接続する場合だけです。 • **non-silent** –（任意）device が PAgP 対応のパートナーに接続されている場合、ポートが **auto** または **desirable** モードになると非サイレント動作を行うように device ポートを設定します。**non-silent** を指定しなかった場合は、サイレントが指定されたものと見なされます。サイレント設定は、ファイル サーバまたはパケット アナライザとの接続に適しています。サイレント

	コマンドまたはアクション	目的
		を設定すると、PAgPが動作してチャネル グループにポートを結合し、このポートが伝送に使用されます。 • **active**：LACP 装置が検出された場合に限り、LACPをイネーブルにします。ポートをアクティブ ネゴシエーション ステートにします。この場合、ポートはLACP パケットを送信することによって、相手ポートとのネゴシエーションを開始します。 • **passive** –：ポート上で LACP をイネーブルにして、ポートをパッシブ ネゴシエーション ステートにします。この場合、ポートは受信するLACP パケットに応答しますが、LACP パケット ネゴシエーションを開始することはありません。
ステップ 6	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

EtherChannel ロード バランシングの設定

送信元ベースまたは宛先ベースの転送方式を使用することによって、EtherChannelのロードバランシングを設定できます。

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **port-channel load-balance** { **dst-ip** | **dst-mac** | **src-dst-ip** | **src-dst-mac** | **src-ip** | **src-mac** }
3. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**port-channel load-balance** { **dst-ip** | **dst-mac** | **src-dst-ip** | **src-dst-mac** | **src-ip** | **src-mac** } 例： `スイッチ(config)# port-channel load-balance src-mac`	EtherChannel のロードバランシング方式を設定します。 デフォルトは **src-mac**です。 次のいずれかの負荷分散方式を選択します。 • **dst-ip**：宛先ホストの IP アドレスを指定します。 • **dst-mac**：着信パケットの宛先ホストの MAC アドレスを指定します。 • **src-dst-ip**：送信元および宛先ホストの IP アドレスを指定します。 • **src-dst-mac**：送信元および宛先ホストの MAC アドレスを指定します。 • **src-ip**：送信元ホストの IP アドレスを指定します。 • **src-mac**：着信パケットの送信元 MAC アドレスを指定します。
ステップ 3	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

ポート チャネル ロード延期の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **port-channel load-defer** *seconds*
4. **interface** *type number*
5. **port-channel load-defer**
6. **end**
7. **show etherchannel** *channel-group* **port-channel**
8. **show platform pm group-masks**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**port-channel load-defer** *seconds* 例： `Switch(config)# port-channel load-defer 60`	すべてのポート チャネルに対し、ポートのロード シェアリング延期間隔を設定します。 • *seconds*：遅延するポート チャネルのロード シェアリングが初期状態で 0 となっている時間。指定できる範囲は 1 ～ 1,800 秒です。デフォルトは 120 秒です。
ステップ 4	**interface** *type number* 例： `Switch(config)# interface port-channel 10`	ポート チャネル インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**port-channel load-defer** 例： `Switch(config-if)# port-channel load-defer`	ポート チャネルでポートのロード シェアリング遅延をイネーブルにします。
ステップ 6	**end** 例： `Switch(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 7	**show etherchannel** *channel-group* **port-channel** 例： `Switch# show etherchannel 1 port-channel`	ポート チャネルの情報を表示します。
ステップ 8	**show platform pm group-masks** 例： `Switch# show platform pm group-masks`	EtherChannel グループ マスク情報を表示します。

例

次に **show etherchannel** *channel-group* **port-channel** コマンドの出力例を示します。*channel-group* 引数を指定しなかった場合は、このコマンドにより、すべてのチャネル グループに関する情報が表示されます。

```
Switch# show etherchannel 1 port-channel

  Port-channels in the group:
  ---------------------------

Port-channel: Po1
------------

Age of the Port-channel   = 0d:00h:37m:08s
Logical slot/port   = 9/1          Number of ports = 0
GC                  = 0x00000000      HotStandBy port = null
Port state          = Port-channel Ag-Not-Inuse
Protocol            =    -
Port security       = Disabled
Load share deferral = Enabled   defer period = 120 sec   time left = 0 sec
```

次に、**show platform pm group-masks** コマンドの出力例を示します。延期タイマー実行中、延期されているポートのグループ マスクは 0xFFFF となります。

```
Switch# show platform pm group-masks

=====================================================================
                    Etherchannel members and group masks table
Group #ports group frame-dist slot port mask interface index
---------------------------------------------------------------------
 1    0      1     src-mac
 2    0      2     src-mac
 3    0      3     src-mac
 4    0      4     src-mac
 5    0      5     src-mac
 6    0      6     src-mac
 7    0      7     src-mac
 8    0      8     src-mac
 9    0      9     src-mac
 10   3      10    src-mac
                                1    12   0000 Gi1/0/12  3
                                1    10   FFFF Gi1/0/10  6
                                1    11   FFFF Gi1/0/11  7
 11   0      11    src-mac
 12   0      12    src-mac
 13   0      13    src-mac
 14   0      14    src-mac
 15   0      15    src-mac
```

PAgP 学習方式およびプライオリティの設定

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **pagp learn-method physical-port**
4. **pagp port-priority** *priority*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/2**	伝送ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**pagp learn-method physical-port** 例： スイッチ(config-if)# **pagp learn-method physical port**	PAgP 学習方式を選択します。 デフォルトでは、**aggregation-port learning** が選択されています。つまり、EtherChannel 内のポートのいずれかを使用して、deviceがパケットを送信元に送信します。集約ポート ラーナーの場合、どの物理ポートにパケットが届くかは重要ではありません。 物理ポートラーナー isである別のdeviceに接続する**physical-port**を選択します。 **port-channel load-balance** グローバル コンフィギュレーション コマンドを **src-mac** に設定してください。 学習方式はリンクの両端で同じ方式に設定する必要があります。
ステップ 4	**pagp port-priority** *priority* 例： スイッチ(config-if)# **pagp port-priority 200**	選択したポートがパケット伝送用として選択されるように、プライオリティを割り当てます。 *priority* に指定できる範囲は 0 ～ 255 です。デフォルト値は 128 です。プライオリティが高いほど、ポートがPAgP伝送に使用される可能性が高くなります。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

LACP ホット スタンバイ ポートの設定

イネーブルの場合、LACP はチャネル内の LACP 互換ポート数を最大に設定しようとします（最大 16 ポート）。同時にアクティブになれる LACP リンクは 8 つだけです。リンクが追加されるとソフトウェアによってホット スタンバイモードになります。アクティブ リンクの 1 つが非アクティブになると、ホット スタンバイ モードのリンクが代わりにアクティブになります。

9 つ以上のリンクが EtherChannel グループとして設定された場合、ソフトウェアは LACP プライオリティに基づいてアクティブにするホット スタンバイ ポートを決定します。ソフトウェアは、LACP を操作するシステム間のすべてのリンクに、次の要素（プライオリティ順）で構成された一意のプライオリティを割り当てます。

- LACP システム プライオリティ
- システム ID（device MAC アドレス）
- LACP ポート プライオリティ
- ポート番号

プライオリティの比較においては、数値が小さいほどプライオリティが高くなります。プライオリティは、ハードウェア上の制約がある場合に、すべての互換ポートが集約されないように、スタンバイ モードにするポートを決定します。

アクティブ ポートかホット スタンバイ ポートかを判別するには、次の（2 つの）手順を使用します。まず、数値的に低いシステム プライオリティとシステム ID を持つシステムの方を選びます。次に、ポートプライオリティおよびポート番号の値に基づいて、そのシステムのアクティブ ポートとホット スタンバイ ポートを決定します。他のシステムのポート プライオリティとポート番号の値は使用されません。

ソフトウェアのアクティブおよびスタンバイ リンクの選択方法に影響を与えるように、LACP システムプライオリティおよびLACPポートプライオリティのデフォルト値を変更できます。

LACP システム プライオリティの設定

lacp system-priority グローバル コンフィギュレーション コマンドを使用して、LACP をイネーブルにしているすべてのEtherChannelに対してシステムプライオリティを設定できます。LACP を設定済みの各チャネルに対しては、システム プライオリティを設定できません。デフォルト値を変更すると、ソフトウェアのアクティブおよびスタンバイ リンクの選択方法に影響します。

どのポートがホットスタンバイモードにあるか確認するには、**show etherchannel summary** 特権 EXEC コマンドを使用します（H ポートステートフラグで表示）。

LACPシステムプライオリティを設定するには、次の手順に従います。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**

3. **lacp system-priority** *priority*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**lacp system-priority** *priority* 例： スイッチ(config)# **lacp system-priority 32000**	LACP システム プライオリティを設定します。 指定できる範囲は 1 ～ 65535 です。デフォルトは 32768 です。 値が小さいほど、システム プライオリティは高くなります。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

LACP ポート プライオリティの設定

デフォルトでは、すべてのポートは同じポート プライオリティです。ローカル システムのシステム プライオリティおよびシステム ID の値がリモート システムよりも小さい場合は、LACP EtherChannel ポートのポート プライオリティをデフォルトよりも小さな値に変更して、最初にアクティブになるホット スタンバイ リンクを変更できます。ホット スタンバイ ポートは、番号が小さい方が先にチャネルでアクティブになります。どのポートがホット スタンバイ モードにあるか確認するには、**show etherchannel summary** 特権 EXEC コマンドを使用します（H ポート ステート フラグで表示）。

（注） LACP がすべての互換ポートを集約できない場合（たとえば、ハードウェアの制約が大きいリモート システム）、EtherChannel 中でアクティブにならないポートはすべてホット スタンバイ ステートになり、チャネル化されたポートのいずれかが機能しない場合に限り使用されます。

LACP ポート プライオリティを設定するには、次の手順に従います。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lacp port-priority** *priority*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**lacp port-priority** *priority* 例： `スイッチ(config-if)# lacp port-priority 32000`	LACP ポート プライオリティを設定します。 指定できる範囲は 1 ～ 65535 です。デフォルトは 32768 です。値が小さいほど、ポートが LACP 伝送に使用される可能性が高くなります。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

LACP ポート チャネルの最小リンク機能の設定

リンク アップ状態で、リンク アップ ステートに移行するポート チャネル インターフェイスの EtherChannel でバンドルする必要のあるアクティブ ポートの最小数を指定できます。EtherChannel の最小リンクを使用して、低帯域幅 LACP EtherChannel がアクティブになることを防止できます。また、LACP EtherChannel にアクティブ メンバー ポートが少なすぎて、必要な最低帯域幅を提供できない場合、この機能により LACP EtherChannel が非アクティブになります。

ポート チャネルに必要なリンクの最小数を設定する。次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *channel-number*
4. **port-channel min-links** *min-links-number*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface port-channel** *channel-number* 例： `スイッチ(config)# interface port-channel 2`	ポートチャネルのインターフェイス コンフィギュレーション モードを開始します。 *channel-number* に指定できる範囲は、1 ～ 63 です。
ステップ 4	**port-channel min-links** *min-links-number* 例： `スイッチ(config-if)# port-channel min-links 3`	リンク アップ状態で、リンク アップ ステートに移行するポート チャネル インターフェイスの EtherChannel でバンドルする必要のあるメンバ ポートの最小数を指定できます。 *min-links-number* の範囲は 2 ～ 8 です。
ステップ 5	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	

LACP 高速レート タイマーの設定

LACP タイマー レートを変更することにより、LACP タイムアウトの時間を変更することができます。**lacp rate** コマンドを使用し、LACP がサポートされているインターフェイスで受信される LACP 制御パケットのレートを設定します。タイムアウト レートは、デフォルトのレート（30 秒）から高速レート（1 秒）に変更することができます。このコマンドは、LACP がイネーブルになっているインターフェイスでのみサポートされます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** {**fastethernet** | **gigabitethernet** | **tengigabitethernet**} *slot/port*
4. **lacp rate** {**normal** | **fast**}
5. **end**
6. **show lacp internal**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** {**fastethernet**	**gigabitethernet**	**tengigabitethernet**} *slot/port* 例： スイッチ(config)# **interface gigabitEthernet 2/1**	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**lacp rate** {**normal** | **fast**} 例： `スイッチ(config-if)# lacp rate fast`	LACP がサポートされているインターフェイスで受信される LACP 制御パケットのレートを設定します。 • タイムアウトレートをデフォルトにリセットするには、**no lacp rate** コマンドを使用します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show lacp internal** 例： `スイッチ# show lacp internal` `スイッチ# show lacp counters`	設定を確認します。

グローバルな Auto-LAG の設定

手順の概要

1. **enable**
2. **configure terminal**
3. [**no**] **port-channel auto**
4. **end**
5. **show etherchannel auto**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	[no] port-channel auto 例： スイッチ(config)# port-channel auto	スイッチ上の Auto-LAG 機能をグローバルで有効にします。スイッチ上の Auto-LAG 機能をグローバルで無効にするには、このコマンドの no 形式を使用します。 （注） デフォルトでは、auto-LAG 機能は各ポート上でイネーブルになっています。
ステップ 4	end 例： スイッチ(config)# end	特権 EXEC モードに戻ります。
ステップ 5	show etherchannel auto 例： スイッチ# show etherchannel auto	EtherChannel が自動的に作成されたことが表示されます。

ポート インターフェイスでの Auto-LAG の設定

手順の概要

1. enable
2. configure terminal
3. interface *interface-id*
4. [no] channel-group auto
5. end
6. show etherchannel auto

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	Auto-LAG を有効にするポート インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	[**no**] **channel-group auto** 例： スイッチ(config-if)# **channel-group auto**	（任意）個々のポートインターフェイスでAuto-LAG 機能を有効にします。個々のポートインターフェイス上で Auto-LAG 機能を無効にするには、このコマンドの no 形式を使用します。 （注） デフォルトでは、auto-LAG 機能は各ポート上でイネーブルになっています。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show etherchannel auto** 例： スイッチ# **show etherchannel auto**	EtherChannel が自動的に作成されたことが表示されます。

次のタスク

Auto-LAG での持続性の設定

自動で作成された EtherChannel を手動のものに変更し、既存の EtherChannel に設定を追加するには、persistence コマンドを使用します。

手順の概要

1. **enable**
2. **port-channel** *channel-number* **persistent**
3. **show etherchannel summary**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**port-channel** *channel-number* **persistent** 例： スイッチ# **port-channel 1 persistent**	自動で作成された EtherChannel を手動のものに変更し、EtherChannel に設定を追加することができます。
ステップ 3	**show etherchannel summary** 例： スイッチ# **show etherchannel summary**	EtherChannel 情報を表示します。

EtherChannel、PAgP、および LACP ステータスのモニタ

この表に記載されているコマンドを使用して EtherChannel、PAgP、および LACP ステータスを表示できます。

表 *55 : EtherChannel*、*PAgP*、および *LACP* ステータスのモニタ用コマンド

コマンド	説明
clear lacp { *channel-group-number* **counters** | **counters** }	LACP チャネル グループ情報およびトラフィック カウンタをクリアします。
clear pagp { *channel-group-number* **counters** | **counters** }	PAgP チャネル グループ情報およびトラフィック カウンタをクリアします。
show etherchannel [*channel-group-number* { **detail** | **load-balance** | **port** | **port-channel** | **protocol** | **summary** }] [**detail** | **load-balance** | **port** | **port-channel** | **protocol** | **auto** | **summary**]	EtherChannel 情報が簡潔、詳細に、1 行のサマリー形式で表示されます。負荷分散方式またはフレーム配布方式、ポート、ポート チャネル、プロトコル、および Auto-LAG 情報も表示されます。
show pagp [*channel-group-number*] { **counters** | **internal** | **neighbor** }	トラフィック情報、内部 PAgP 設定、ネイバー情報などの PAgP 情報が表示されます。
show pagp [*channel-group-number*] **dual-active**	デュアルアクティブ検出ステータスが表示されます。
show lacp [*channel-group-number*] { **counters** | **internal** | **neighbor** | **sys-id**}	トラフィック情報、内部 LACP 設定、ネイバー情報などの LACP 情報が表示されます。
show running-config	設定エントリを確認します。
show etherchannel load-balance	ポート チャネル内のポート間のロードバランシング、またはフレーム配布方式を表示します。

EtherChannel の設定例

レイヤ 2 EtherChannel の設定：例

この例では、スタック内deviceの 1 つの に EtherChannel を設定する例を示します。2 つのポートを VLAN 10 のスタティックアクセスポートとして、PAgP モードが **desirable** であるチャネル 5 に割り当てます。

```
スイッチ# configure terminal
スイッチ(config)# interface range gigabitethernet2/0/1 -2
スイッチ(config-if-range)# switchport mode access
スイッチ(config-if-range)# switchport access vlan 10
スイッチ(config-if-range)# channel-group 5 mode desirable non-silent
スイッチ(config-if-range)# end
```

この例では、スタック内deviceの 1 つの に EtherChannel を設定する例を示します。2 つのポートは VLAN 10 のスタティックアクセス ポートとして、LACP モードが active であるチャネル 5 に割り当てられます。 **active :**

```
スイッチ# configure terminal
スイッチ(config)# interface range gigabitethernet2/0/1 -2
スイッチ(config-if-range)# switchport mode access
スイッチ(config-if-range)# switchport access vlan 10
スイッチ(config-if-range)# channel-group 5 mode active
スイッチ(config-if-range)# end
```

次の例では、クロススタック EtherChannel を設定する方法を示します。LACP パッシブ モードを使用して、VLAN 10 内のスタティックアクセス ポートとしてスタック メンバ 1 のポートを 2 つ、スタック メンバ 2 のポートを 1 つチャネル 5 に割り当てます。

```
スイッチ# configure terminal
スイッチ(config)# interface range gigabitethernet2/0/4 -5
スイッチ(config-if-range)# switchport mode access
スイッチ(config-if-range)# switchport access vlan 10
スイッチ(config-if-range)# channel-group 5 mode passive
スイッチ(config-if-range)# exit
スイッチ(config)# interface gigabitethernet3/0/3
スイッチ(config-if)# switchport mode access
スイッチ(config-if)# switchport access vlan 10
スイッチ(config-if)# channel-group 5 mode passive
スイッチ(config-if)# exit
```

PoE または LACP ネゴシエーションのエラーは、スイッチからアクセスポイント（AP）に 2 つのポートを設定した場合に発生する可能性があります。このシナリオは、ポートチャネルの設定をスイッチ側で行うと回避できます。詳細については、次の例を参照してください。

```
interface Port-channel1
  switchport access vlan 20
 switchport mode access
  switchport nonegotiate
  no port-channel standalone-disable   <--this one
  spanning-tree portfast
```

（注） ポートがポートのフラッピングに関する LACP エラーを検出した場合は、次のコマンドも含める必要があります。 **no errdisable detect cause pagp-flap**

例：ポート チャネル ロード延期の設定

```
Switch# configure terminal
Switch(config)# port-channel load-defer 60
Switch(config)# interface port-channel 10
Switch(config-if)# port-channel load-defer
Switch(config-if)# end
```

Auto-LAG の設定：例

次に、スイッチに Auto-LAG を設定する例を示します。

```
device> enable
device# configure terminal
device (config)# port-channel auto
device (config-if)# end
device#  show etherchannel auto
```

次の例は、自動的に作成された EtherChannel の概要を示します。

```
device# show etherchannel auto
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port
        A - formed by Auto LAG

Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
1      Po1(SUA)        LACP      Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)
```

次の例は、**port-channel** 1 **persistent** コマンドを実行した後の自動 EtherChannel の概要を示します。

```
device# port-channel 1 persistent

device# show etherchannel summary
Switch# show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port
        A - formed by Auto LAG

Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
1      Po1(SU)         LACP      Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)
```

LACP ポート チャネルの最小リンクの設定例

次の例は、LACP ポート チャネル最小リンク数の設定方法を示しています。

```
device > enable
device# configure terminal
device(config)# interface port-channel 5
device(config-if)# port-channel min-links 3
device#  show etherchannel 25 summary
device# end
```

スタンドアロン スイッチで最小リンク要件が満たされない場合、ポート チャネルにフラグが設定され SM/SN または RM/RN ステートが割り当てられます。

```
 device# show etherchannel 5 summary

Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use N- not in use, no aggregration
f - failed to allocate aggregator
M - not in use, no aggregation due to minimum links not met
m- not in use, port not aggregated due to minimum links not met
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
Number of channel-groups in use: 6
Number of aggregators: 6

 Group   Port-channel   Protocol    Ports
------+-------------+-----------+-----------------------------------------------
 6       Po25(RM)        LACP        Gi1/3/1(D) Gi1/3/2(D) Gi2/2/25(D) Gi2/2/26(W)
```

例：LACP 高速レート タイマーの設定

次の例は LACP レートの設定方法を示しています。

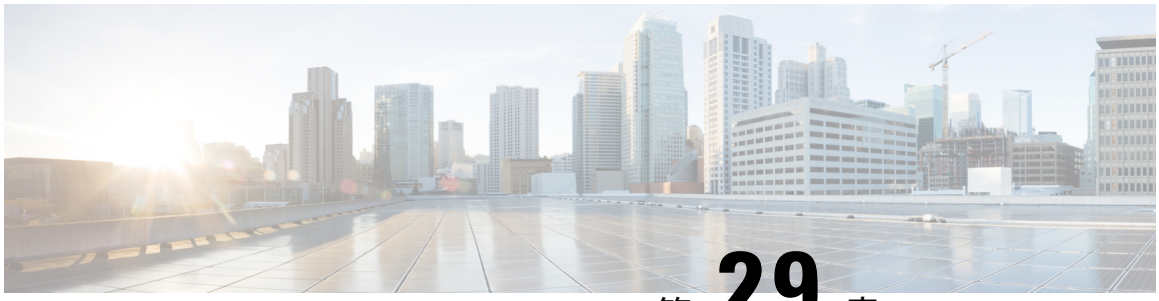
```
device> enable
device# configure terminal
device(config)# interface gigabitEthernet 2/1
device(config-if)# lacp rate fast
device(config-if)# exit
device(config)# end
device# show lacp internal
device# show lacp counters
```

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

```
device# show lacp internal
Flags: S - Device is requesting Slow LACPDUs
F - Device is requesting Fast LACPDUs
A - Device is in Active mode P - Device is in Passive mode
Channel group 25
LACP port Admin Oper Port Port
Port Flags State Priority Key Key Number State
Te1/49 FA bndl 32768 0x19 0x19 0x32 0x3F
Te1/50 FA bndl 32768 0x19 0x19 0x33 0x3F
Te1/51 FA bndl 32768 0x19 0x19 0x34 0x3F
Te1/52 FA bndl 32768 0x19 0x19 0x35 0x3F
```

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

```
device# show lacp counters

LACPDUs Marker Marker Response LACPDUs
Port Sent Recv Sent Recv Sent Recv Pkts Err
---------------------------------------------------------------------
Channel group: 24
Te1/1/27 2 2 0 0 0 0 0
Te2/1/25 2 2 0 0 0 0 0
```



第 29 章

リンクステート トラッキングの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

リンク ステート トラッキングの設定の制約事項

- スイッチ 1 つにつき、設定できるリンクステート グループは 2 つだけです。
- インターフェイスは、複数のリンクステート グループのメンバにはなれません。
- リンクステート グループ内でアップストリーム インターフェイスとして定義されているインターフェイスを、リンクステート グループ内でダウンストリーム インターフェイスとして定義することはできません。
- ダウンストリームの EtherChannel インターフェイスの一部となる個々のインスターフェイスでリンクステート トラッキングをイネーブルにしないでください。

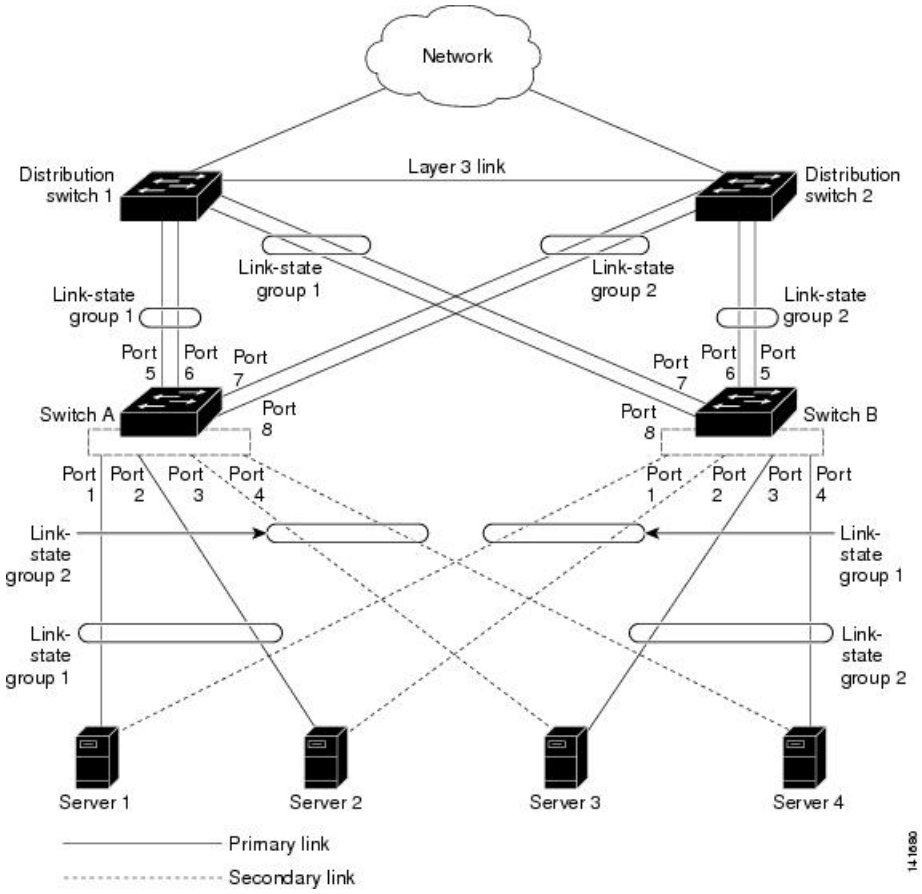
リンクステート トラッキングの概要

リンクステート トラッキングは、トランク フェールオーバーとも呼ばれ、複数のインターフェイスのリンクステートをバインドします。リンクステート トラッキングはサーバー NIC アダプタのチーミングと連動させることができ、ネットワークに冗長性を提供します。サーバー NIC アダプタがプライマリまたはセカンダリの関係で設定されており、プライマリ インターフェイスでリンクが失われた場合は、ネットワーク接続が透過的にセカンダリ インターフェイスに切り替えられます。

（注） ポートの集合（EtherChannel）またはアクセス モードかトランク モードのいずれかの単一の物理ポートをインターフェイスに指定できます。

次の図の設定では、ネットワーク トラフィック フローのバランスが確実に保たれます。

図 *50 :* 一般的なリンクステート トラッキングの設定



- スイッチと他のネットワーク デバイスへのリンクの場合

- サーバー 1 とサーバー 2 は、プライマリ リンクにスイッチ A を使用し、セカンダリ リンクにスイッチ B を使用しています。
- サーバー 3 とサーバー 4 は、プライマリ リンクにスイッチ B を使用し、セカンダリ リンクにスイッチ A を使用しています。

- スイッチ A のリンクステート グループ 1
  - スイッチ A はリンクステート グループ 1 を介して、プライマリ リンクをサーバー 1 およびサーバー 2 に使用します。ポート 1 はサーバー 1 に、ポート 2 はサーバー 2 にそれぞれ接続されます。ポート 1 およびポート 2 はリンクステート グループ 1 でダウンストリーム インターフェイスとして使用します。
  - ポート 5 およびポート 6 は、リンクステート グループ 1 を介して分散スイッチ 1 に接続されます。ポート 5 およびポート 6 は、リンクステート グループ 1 でアップストリーム インターフェイスとして使用します。

- スイッチ A のリンクステート グループ 2
  - スイッチ A はリンクステートグループ 2 を介して、セカンダリ リンクをサーバー 3 およびサーバー 4 に使用します。ポート 3 はサーバー 3 に、ポート 4 はサーバー 4 にそれぞれ接続されます。ポート 3 およびポート 4 はリンクステート グループ 2 でダウンストリーム インターフェイスとして使用します。
  - ポート 7 およびポート 8 は、リンクステート グループ 2 を介して分散スイッチ 2 に接続されます。ポート 7 およびポート 8 は、リンクステート グループ 2 でアップストリーム インターフェイスとして使用します。

- スイッチ B のリンクステート グループ 2
  - スイッチ B はリンクステートグループ 2 を介して、プライマリ リンクをサーバー 3 およびサーバー 4 に使用します。ポート 3 はサーバー 3 に、ポート 4 はサーバー 4 にそれぞれ接続されます。ポート 3 およびポート 4 はリンクステート グループ 2 でダウンストリーム インターフェイスとして使用します。
  - ポート 5 およびポート 6 は、リンクステート グループ 2 を介して分散スイッチ 2 に接続されます。ポート 5 およびポート 6 は、リンクステート グループ 2 でアップストリーム インターフェイスとして使用します。

- スイッチ B のリンクステート グループ 1
  - スイッチ B はリンクステート グループ 1 を介して、セカンダリ リンクをサーバー 1 およびサーバー 2 に使用します。ポート 1 はサーバー 1 に、ポート 2 はサーバー 2 にそれぞれ接続されます。ポート 1 およびポート 2 はリンクステート グループ 1 でダウンストリーム インターフェイスとして使用します。
  - ポート 7 およびポート 8 は、リンクステート グループ 1 を介して分散スイッチ 1 に接続されます。ポート 7 およびポート 8 は、リンクステート グループ 1 でアップストリーム インターフェイスとして使用します。

分散スイッチやルータに障害が発生したり、ケーブルが切断されたり、リンクが失われたために、リンクステート グループ内でアップストリーム ポートが利用不能や接続不能になる場合があります。これらは、リンクステート トラッキングがイネーブルの際の、ダウンストリーム インターフェイスとアップストリーム インターフェイス間の相互作用です。

- アップストリーム インターフェイスがリンクアップ ステートの場合、ダウンストリーム インターフェイスをリンクアップ ステートに変更したり、リンクアップ ステートのままにしたりすることができます。
- すべてのアップストリーム インターフェイスが利用不能になった場合、リンクステート トラッキングが自動的にダウンストリーム インターフェイスを errdisable ステートにします。サーバー間の接続は、自動的にプライマリ サーバー インターフェイスからセカンダリ サーバー インターフェイスに変更されます。たとえば、前の図で、ポート 6 のアップストリーム リンクが切断されても、ダウンストリーム ポート 1 および 2 のリンク ステートは変わりません。ただし、アップストリーム ポート 5 のリンクも切断された場合、ダウンストリーム ポートのリンク ステートがリンクダウン ステートに変更されます。サーバー 1 およびサーバー 2 の接続については、リンクステート グループ 1 からリンクステート グループ 2 へ変更します。ダウンストリーム ポート 3 およびダウンストリーム ポート 4 は、リンクグループ 2 であるためステートを変更しません。
- リンクステート グループが設定されている場合、リンクステート トラッキングはディセーブルで、アップストリーム インターフェイスが切断され、ダウンストリーム インターフェイスのリンク ステートは変更されないままになります。サーバーはこのアップストリーム接続が切断されたことを認識せず、セカンダリ インターフェイスにフェールオーバーしません。

障害のあるダウンストリーム ポートをリンクステート グループから削除することで、ダウンストリーム インターフェイスのリンクダウン状態から復旧できます。複数のダウンストリーム インターフェイスを復旧させるには、リンクステート グループをディセーブルにします。

リンクステート トラッキングの設定方法

リンクステート トラッキングを有効にするには、リンク ステート グループを作成し、そのグループに割り当てるインターフェイスを指定します。このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **link state track** *number*
3. **interface** *interface-id*
4. **link state group** [*number*]{**upstream** | **downstream**}
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**link state track** *number* 例： `スイッチ(config)# link state track 2`	リンクステート グループを作成して、リンクステート トラッキングを有効にします。グループ番号は 1 または 2 です。デフォルトは 1 です。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/1`	設定する物理インターフェイスまたはインターフェイスの範囲を指定して、インターフェイス コンフィギュレーション モードを開始します。 有効なインターフェイスには、アクセスまたはトランクモード（IEEE 802.1q）のスイッチ ポートかルーテッド ポートが含まれます。 （注） Etherchannel インターフェイスの一部となる個々のインスターフェイスでリンクステート トラッキングを有効にしないでください。
ステップ 4	**link state group** [*number*]{**upstream** | **downstream**} 例： `スイッチ(config-if)# link state group 2 upstream`	リンクステート グループを指定し、グループ内のインターフェイスを upstream または downstream インターフェイスに設定します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

リンクステート トラッキングのモニターリング

次の表のコマンドを使用してリンクステート トラッキングのステータスを表示できます。

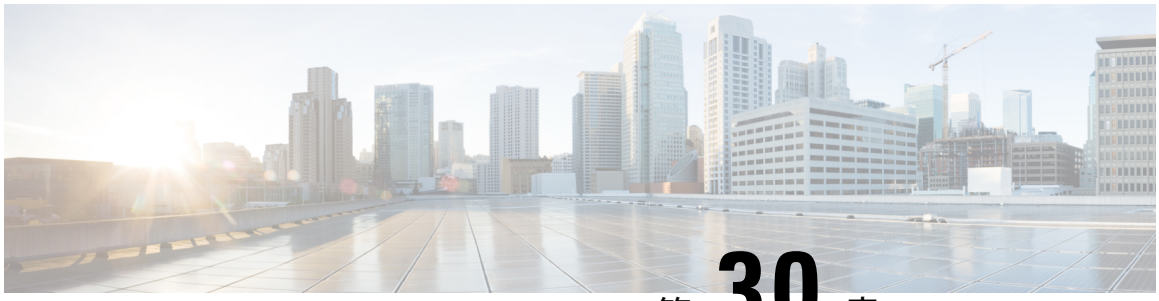
表 *56 :* リンクステート トラッキング ステータスをモニターするコマンド

コマンド	説明
show link state group [*number*] [**detail**]	リンクステート グループ情報を表示します。

リンクステート トラッキングの設定：例

次に、リンクステート グループ 1 を作成してリンクステート グループにインターフェイスを設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# link state track 1
スイッチ(config-if)# interface range gigabitethernet1/0/21-22
スイッチ(config-if)# link state group 1 upstream
スイッチ(config-if)# interface gigabitethernet1/0/1
スイッチ(config-if)# link state group 1 downstream
スイッチ(config-if)# interface gigabitethernet1/0/3
スイッチ(config-if)# link state group 1 downstream
スイッチ(config-if)# interface gigabitethernet1/0/5
スイッチ(config-if)# link state group 1 downstream
スイッチ(config-if)# end
```



第 30 章

Resilient Ethernet Protocol の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

Resilient Ethernet Protocol の概要

Resilient Ethernet Protocol（REP）はシスコ独自のプロトコルで、スパニングツリー プロトコル（STP）に代わるプロトコルとして、ネットワーク ループの制御、リンク障害の処理、コンバージェンス時間の改善を実現します。REP は、セグメントに接続されているポートのグループを制御することで、セグメントがブリッジングループを作成するのを防ぎ、セグメント内のリンク障害に応答します。REP は、より複雑なネットワークを構築するための基盤を提供し、VLAN ロード バランシングをサポートします。

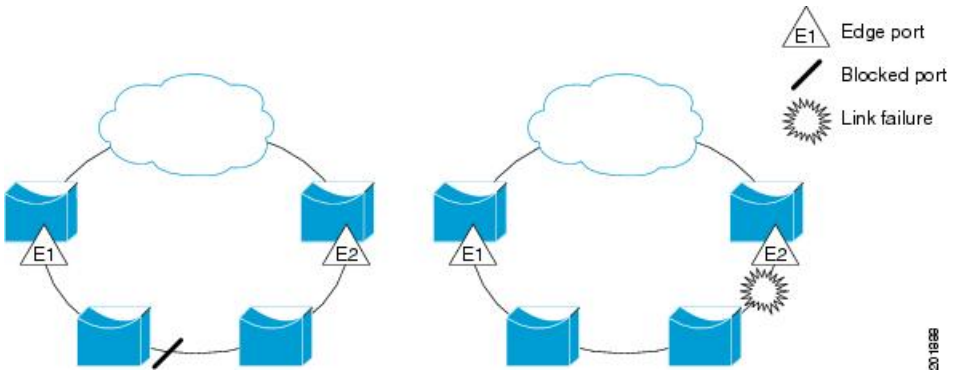
（注）

- REP は IP Base、IP Lite、および IP Services を実行している Catalyst スイッチでサポートされます。REP は LAN Base ライセンスではサポートされません。
- REP は Cisco Catalyst 3560-CX スイッチのみでサポートされています。

REP セグメントは、相互接続されたポートのチェーンで、セグメント ID が設定されます。各セグメントは、標準（非エッジ）セグメント ポートと、2 つのユーザ設定のエッジ ポートで構成されています。1 つのデバイスは同じセグメントに属するポートを複数持たず、各セグメント ポートにある外部ネイバーは 1 つだけです。セグメントは共有メディアを経由できますが、どのリンクでも同じセグメントに属することができるポートは 2 つだけです。REP はトランクのイーサネット フロー ポイント（EFP）インターフェイスでのみサポートされます。

次の図に、4 つのスイッチにまたがる 6 つのポートで構成されているセグメントの例を示します。ポート E1 および E2 がエッジ ポートとして設定されています。（左側のセグメントのように）すべてのポートが動作可能の場合、斜線で表しているように単一ポートがブロックされます。ネットワークに障害が発生した場合、ブロックされたポートがフォワーディング ステートに戻り、ネットワークの中断を最小限に抑えます。

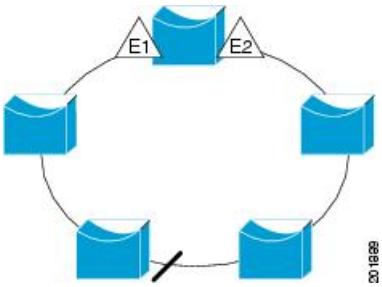
図 *51 : REP* オープン セグメント



上の図に示されたセグメントは、オープン セグメントで、2 つのエッジ ポート間は接続されていません。REP セグメントはブリッジング ループの原因とならないため、セグメント エッジを安全に任意のネットワークに接続できます。セグメント内のデバイスに接続されているすべてのホストには、エッジ ポートを通じて残りのネットワークに接続する方法が 2 つありますが、いつでもアクセス可能なのは 1 つだけです。いずれかのセグメントまたは REP セグメントのいずれかのポートに障害が発生した場合、REP はすべてのポートのブロックを解除し、他のゲートウェイ経由で接続できるようにします。

次の図に示すセグメントはリング セグメントであり、同じデバイス上に両方のエッジ ポートがあります。この設定を使用すると、セグメント内の任意の 2 デバイス間で冗長接続を形成することができます。

図 *52 : REP* リング セグメント



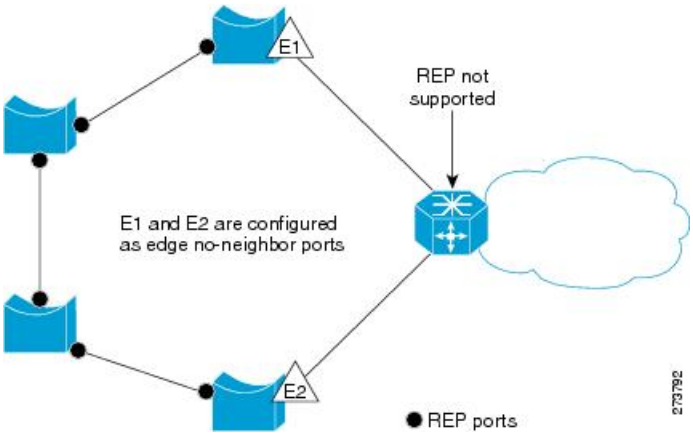
REP セグメントには、次のような特徴があります。

- セグメント内の全ポートが動作可能な場合、1 ポート（代替ポートと呼ばれる）が各 VLAN でブロック ステートとなります。VLAN ロード バランシングが設定されている場合は、セグメント内の 2 つのポートが VLAN のブロック ステートを制御します。
- セグメント内の 1 つまたは複数のポートが動作不能になると、リンク障害が発生して、すべてのポートがすべての VLAN トラフィックを転送して、接続性を確保します。
- リンク障害の場合、できるだけ早期に代替ポートのブロックが解除されます。障害リンクが復旧すると、ネットワークの中断を最小限に抑えるように VLAN 単位で論理的にブロックされたポートが選択されます。

REP セグメントに基づいて、ほとんどのネットワーク タイプを構成することができます。また REP はプライマリ エッジ ポート（セグメント内の任意のポート）で制御される VLAN ロード バランシングをサポートします。

アクセス リング トポロジでは、次の図に示すように、ネイバー スイッチで REP がサポートされない場合があります。この場合、そのスイッチ側のポート（E1 と E2）を非ネイバー エッジ ポートとして設定できます。これらのポートは、エッジ ポートのすべての特性を継承するため、他のエッジ ポートと同じように設定できます。たとえば、STP や REP のトポロジ変更通知を集約スイッチに送信するように設定することもできます。その場合、送信される STP トポロジ変更通知（TCN）は、マルチ スパニングツリー（MST）STP メッセージになります。

図 *53 :* 非ネイバー エッジ ポート



REP には次のような制限事項があります。

- 各セグメント ポートを設定する必要があります。設定を間違えると、ネットワーク内でフォワーディング ループが発生します。
- REP はセグメント内の単一障害ポートだけを管理できます。REP セグメント内の複数ポート障害の場合、ネットワークの接続が中断します。
- 冗長ネットワーク内だけに REP を設定します。冗長性のないネットワークに REP を設定すると、接続が失われます。

リンク完全性

REP は、リンク完全性の確認にエッジ ポート間でエンドツーエンド ポーリング機能を使用しません。ローカル リンク障害検出を実装しています。REP リンク ステータス レイヤ（LSL）が REP 対応ネイバーを検出して、セグメント内の接続性を確立します。ネイバーが検出されるまで、インターフェイス上ですべての VLAN がブロックされます。ネイバーが特定されたあと、REP が代替ポートとなるネイバー ポートと、トラフィックを転送するポートを決定します。

セグメント内のポートごとに、一意のポート ID が割り当てられます。ポート ID フォーマットは、スパニングツリーアルゴリズムで使用されるものと類似しており、ポート番号（ブリッジ上で一意）と、関連 MAC アドレス（ネットワーク内で一意）から構成されます。セグメント ポートが起動すると、ポートの LSL がセグメント ID およびポート ID を含むパケットの送信を開始します。ポートは、同じセグメント内のネイバーとのスリーウェイ ハンドシェイクを実行したあとで、動作可能と宣言されます。

次のような場合、セグメント ポートは動作可能になりません。

- ネイバーに同じセグメント ID がない
- 複数のネイバーに同じセグメント ID がある
- ネイバーがピアとして、ローカル ポートに確認応答しない

各ポートは、直近のネイバーと隣接関係を確立します。ネイバーとの隣接関係が確立されると、代替ポートとして機能する、セグメントのブロックされたポートを決定するようにポートが相互にネゴシエートします。その他のすべてのポートのブロックは解除されます。デフォルトでは、REP パケットはブリッジ プロトコル データ ユニットクラスの MAC アドレスに送信されます。パケットは、シスコ マルチキャスト アドレスにも送信できますが、セグメントに障害が発生した場合にブロックされたポートのアドバタイズ（BPA）メッセージの送信だけに使用されます。パケットは、REP が動作していない装置によって廃棄されます。

高速コンバージェンス

REP は、物理リンク ベースで動作し、VLAN 単位ベースでは動作しません。すべての VLAN に対して 1 つの hello メッセージしか必要ないため、プロトコル上の負荷が軽減されます。指定セグメント内の全スイッチで継続的に VLAN を作成し、REP トランク ポート上に同じ許容

VLANを設定することを推奨します。ソフトウェアでのメッセージのリレーによって発生する遅延を回避するために、REP ではいくつかのパケットを通常のマルチキャスト アドレスにフラッディングすることも可能です。これらのメッセージはハードウェア フラッド レイヤ（HFL）で動作し、REPセグメントだけではなくネットワーク全体にフラッディングされます。セグメントに属していないスイッチは、これらのメッセージをデータ トラフィックとして扱います。ドメイン全体または特定のセグメントの管理 VLAN を設定することで、これらのメッセージのフラッディングを制御することができます。

ファイバ インターフェイスのコンバージェンス復旧時間の推定値は、200 の VLAN が設定されたローカル セグメントで 50 ミリ秒から 200 ミリ秒までです。VLAN ロード バランシングのコンバージェンスは 300 ミリ秒以下です。

VLAN ロード バランシング

REP セグメント内の 1 つのエッジ ポートがプライマリ エッジ ポートとして機能し、もう一方がセカンダリ エッジ ポートとなります。セグメント内の VLAN ロード バランシングに常に参加しているのがプライマリ エッジ ポートです。REP VLAN バランシングは、設定された代替ポートでいくつかの VLAN をブロックし、プライマリ エッジ ポートでその他の全 VLAN をブロックすることで実行されます。VLAN ロード バランシングを設定する際に、次の 3 種類の方法のいずれかを使用して代替ポートを指定できます。

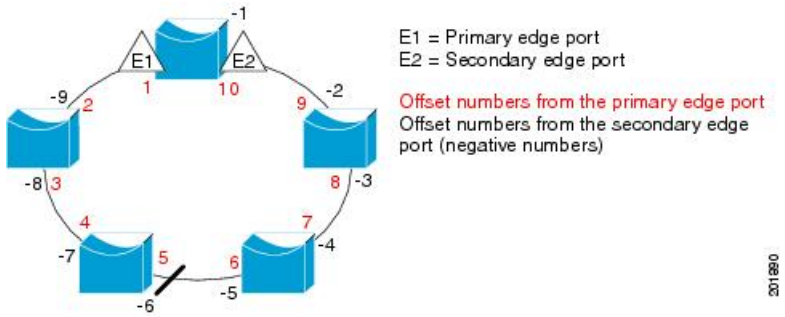
- インターフェイスにポート ID を入力します。セグメント内のポート ID を識別するには、ポートの **show interface rep detail** インターフェイス コンフィギュレーション コマンドを入力します。
- **preferred** キーワードを入力します。これにより、**rep segment** *segment-id* **preferred** インターフェイス コンフィギュレーション コマンドで優先代替ポートとしてすでに設定されているポートを選択します。
- セグメント内のポートのネイバー オフセット番号を入力します。これは、エッジ ポートのダウンストリーム ネイバー ポートを識別するものです。ネイバー オフセット番号の範囲は、-256 ～ +256 で、0 値は無効です。プライマリ エッジ ポートはオフセット番号 1 です。1 を超える正数はプライマリ エッジ ポートのダウンストリーム ネイバーを識別します。負数は、セカンダリ エッジ ポート（オフセット番号 -1）とそのダウンストリーム ネイバーを示します。

（注） プライマリ（またはセカンダリ）エッジ ポートからポートのダウンストリーム位置を識別することで、プライマリ エッジ ポートのオフセット番号を設定します。番号 1 はプライマリ エッジ ポートのオフセット番号なので、オフセット番号 1 は入力しないでください。

次の図に、E1 がプライマリ エッジ ポートで E2 がセカンダリ エッジ ポートの場合の、セグメントのネイバー オフセット番号を示します。リングの内側にある番号は、プライマリ エッジポートからのオフセット番号で、リングの外側にある番号がセカンダリエッジポー

トからのオフセット番号です。正のオフセット番号（プライマリ エッジ ポートからのダウンストリーム位置）または負のオフセット番号（セカンダリ エッジ ポートからのダウンストリーム位置）のいずれかにより、（プライマリ エッジ ポートを除く）全ポートを識別できます。E2 がプライマリ エッジ ポートになるとオフセット番号 1 となり、E1 のオフセット番号が -1 になります。

図 *54* : セグメント内のネイバー オフセット番号



REP セグメントが完了すると、すべての VLAN がブロックされます。VLAN ロード バランシングを設定する際には、次の 2 種類の方法のいずれかを使用してトリガーを設定する必要もあります。

- プライマリ エッジ ポートのあるスイッチ上で **rep preempt segment** *segment-id* 特権 EXEC コマンドを入力することで、いつでも手動で VLAN ロード バランシングをトリガーすることができます。
- **rep preempt delay** *seconds* インターフェイス コンフィギュレーション コマンドを入力すると、プリエンプション遅延時間を設定できます。リンク障害が発生して回復すると、設定されたプリエンプション期間の経過後に VLAN ロード バランシングが開始されます。設定時間が経過する前に別のポートで障害が発生した場合、遅延タイマーが再開されることに注意してください。

（注） VLAN ロード バランシングが設定されている場合、手動での介入またはリンク障害および回復によってトリガーされるまで、動作が開始されません。

VLAN ロード バランシングがトリガーされると、プライマリ エッジ ポートがメッセージを送信して、セグメント内の全インターフェイスにプリエンプションについて警告します。メッセージがセカンダリ ポートで受信されると、メッセージがネットワークに送信され、メッセージ内で指定された VLAN セットをブロックするように代替ポートに通知し、残りの VLAN をブロックするようにプライマリ エッジ ポートに通知します。

またすべての VLAN をブロックするために、セグメント内の特定ポートを設定できます。プライマリ エッジ ポートだけによって VLAN ロード バランシングが開始され、セグメントが各エンドでエッジ ポートによって終端されていない場合開始することができません。プライマリ エッジ ポートは、ローカル VLAN ロードバランシング設定を決定します。

ロード バランシングを再設定するには、プライマリ エッジ ポートを再設定します。ロードバランシング設定を変更すると、プライマリ エッジ ポートでは、**rep preempt segment** コマンドが実行されるか、ポート障害および復旧のあとで設定済みプリエンプト遅延期間が経過してから、新規設定が実行されます。エッジ ポートを通常セグメント ポートに変更しても、既存の VLAN ロードバランシング ステータスは変更されません。新規エッジ ポートを設定すると、新規トポロジ設定になる可能性があります。

スパニングツリー インタラクション

REP は、STP とも Flex Link 機能とも対話しませんが、どちらとも共存できます。セグメントに属しているポートはスパニングツリーの制御から削除されるため、セグメント ポートでは STP BPDU の送受信は行われません。したがって、STP はセグメント上で実行できません。

STP リング コンフィギュレーションから REP セグメント コンフィギュレーションに移行するには、まずリング内の単一ポートをセグメントの一部として設定し、次にセグメント数を最小限にするように隣接するポートを設定します。各セグメントには常にブロックされたポートが含まれているので、セグメントが複数になるとブロックされたポートも複数になり、接続が失われる可能性があります。セグメントがエッジポートの場所まで両方向に設定されたら、エッジ ポートを設定します。

REP ポート

REP セグメントは、障害ポート、オープン ポート、および代替ポートで構成されます。

- 標準セグメント ポートとして設定されたポートは、障害ポートとして起動します。
- ネイバーとの隣接関係が確立されると、ポートは代替ポート ステートに移行して、インターフェイス内の全 VLAN をブロックします。ブロックされたポートのネゴシエーションが実施され、セグメントが安定すると、1 つのブロックされたポートが代替ロールに留まり、他のすべてのポートがオープン ポートになります。
- リンク内で障害が発生すると、すべてのポートが障害ステートに遷移します。代替ポートは、障害通知を受信すると、すべての VLAN を転送するオープン ステートに遷移します。

通常セグメント ポートをエッジ ポートに変換しても、エッジ ポートを通常セグメント ポートに変換しても、必ずトポロジ変更が発生するわけではありません。エッジポートを通常セグメント ポートに変更する場合、設定されるまで VLAN ロード バランシングは実装されません。VLAN ロード バランシングの場合、セグメント内に 2 つのエッジ ポートを設定する必要があります。

スパニングツリー ポートとして再設定されたセグメント ポートは、スパニングツリー設定に従って再起動します。デフォルトでは、これは指定ブロッキング ポートです。PortFast が設定されていたり、STP がディセーブルの場合、ポートはフォワーディング ステートになります。

Resilient Ethernet Protocol の設定方法

セグメントは、チェーンで相互接続されているポートの集合で、セグメント ID が設定されています。REP セグメントを設定するには、REP 管理 VLAN を設定し（またはデフォルト VLAN 1 を使用し）、次にインターフェイス コンフィギュレーション モードを使用してセグメントにポートを追加します。2 つのエッジ ポートをセグメント内に設定して、デフォルトで 1 つをプライマリ エッジ ポート、もう 1 つをセカンダリ エッジ ポートにします。1 セグメント内のプライマリ エッジ ポートは 1 つだけです。別のスイッチのポートなど、セグメント内で 2 つのポートをプライマリ エッジ ポートに設定すると、REP がそのうちのいずれかを選択してセグメントのプライマリ エッジ ポートとして機能させます。必要に応じて、STCN および VLAN ロード バランシングが送信される場所を設定できます。

REP のデフォルト設定

REP はすべてのインターフェイス上でディセーブルです。イネーブルにする際に、エッジ ポートとして設定されていなければインターフェイスは通常セグメント ポートになります。

REP をイネーブルにする際に、STCN の送信タスクはディセーブルで、すべての VLAN はブロックされ、管理 VLAN は VLAN 1 になります。

VLAN ロードバランシングがイネーブルの場合、デフォルトは手動でのプリエンプションで、遅延タイマーはディセーブルになっています。VLAN ロード バランシングが設定されていない場合、手動でのプリエンプション後のデフォルト動作は、プライマリ エッジ ポートで全 VLAN がブロックとなります。

REP 設定時の注意事項

REP の設定時には、次の注意事項に従ってください。

- まず 1 ポートの設定から始めて、セグメント数とブロックされたポートの数を最小限に抑えるように隣接するポートを設定することを推奨します。
- 外部ネイバーが設定されておらずセグメント内では 3 つ以上のポートに障害が発生した場合、1 ポートがデータ パス用のフォワーディング ステートになり、設定中の接続性の維持に役立ちます。**show rep interface** コマンド出力では、このポートのポートロールは「**Fail Logical Open**」と表示され、他の障害ポートのポートロールは「**Fail No Ext Neighbor**」と表示されます。障害ポートの外部ネイバーが設定されている場合、ポートは代替ポートに移行して、代替ポート選択メカニズムに基づいて最終的にオープン ステートになるか、代替ポートのままになります。
- REP ポートは、レイヤ 2 IEEE 802.1Q またはトランク ポートのいずれかである必要があります。
- 同じ許可 VLAN のセットでセグメント内のすべてのトランク ポートを設定することを推奨します。

- 別の REP インターフェイスがブロックを解除するメッセージを送信するまで REP はすべての VLAN をブロックするため、Telnet 接続で REP を設定するときは注意してください。同じインターフェイス経由でルータにアクセスする Telnet セッションで REP をイネーブルにすると、ルータへの接続が失われることがあります。
- REP と STP または REP と Flex Link を同じセグメントやインターフェイスで実行できません。
- STP ネットワークを REP セグメントに接続する場合、接続はセグメント エッジであることを確認してください。エッジで実行されていない STP 接続は、REP セグメントでは STP が実行されないため、ブリッジング ループが発生する可能性があります。すべての STP BPDU は、REP インターフェイスで廃棄されます。
- 同じ許可 VLAN のセットでセグメント内のすべてのトランク ポートを設定する必要があります。これを行わないと、設定ミスが発生します。
- REP がスイッチの 2 つのポートでイネーブルである場合、両方のポートが通常セグメント ポートまたはエッジ ポートのいずれかである必要があります。REP ポートは以下の規則に従います。
  - スイッチ上の REP ポートの数に制限はありません。しかし、同じ REP セグメントに属することができるスイッチ上のポートは 2 つだけです。
  - セグメント内にスイッチ上の 1 ポートだけが設定されている場合、そのポートがエッジ ポートとなります。
  - 同じセグメント内に属するスイッチに 2 つのポートがある場合、両方のポートがエッジポートであるか、両方のポートが通常セグメント ポートであるか、一方が通常ポートでもう一方が非ネイバー エッジ ポートである必要があります。スイッチ上のエッジ ポートと通常セグメント ポートが同じセグメントに属することはできません。
  - スイッチ上の 2 ポートが同じセグメントに属していて、1 つがエッジ ポートとして設定され、もう 1 つが通常セグメント ポートに設定されている場合（設定ミス）、エッジ ポートは通常セグメント ポートとして扱われます。
- REP インターフェイスはブロックされた状態になり、ブロック解除できるようになるまでブロックされた状態のまま残ります。したがって、突然の切断を避けるために REP インターフェイスの状態には注意する必要があります。
- REP はネイティブ VLAN にすべての LSL PDU をタグなしフレームで送信します。シスコ マルチキャスト アドレスに送信された BPA メッセージは、管理 VLAN で送信されます。これはデフォルトで VLAN 1 です。
- ネイバーからの hello が受信されないままどのくらいの時間が経過すると REP インターフェイスがダウンするかを設定できます。**rep lsl-age-timer** value インターフェイス コンフィギュレーション コマンドを使用して、120 ～ 10000 ミリ秒の時間を設定します。LSL hello タイマーは、このエージング タイマーの値を 3 で割った値に設定されます。通常の動作では、ピア スイッチのエージング タイマーが満了になって hello メッセージが確認されるまでに LSL hello が 3 回送信されます。

- EtherChannel ポート チャネル インターフェイスでは、1000 ミリ秒未満の LSL エージング タイマー値はサポートされていません。ポート チャネルで 1000 ミリ秒未満の値を設定しようとすると、エラー メッセージが表示されてコマンドが拒否されます。

- REP ポートは、次のポート タイプのいずれかに設定できません。
  - スイッチド ポート アナライザ（SPAN）宛先ポート
  - トンネル ポート
  - アクセスポート
- REP は EtherChannel でサポートされていますが、EtherChannel に属する個別のポートではサポートされません。
- スイッチごとに最大 64 の REP セグメントを設定できます。

REP 管理 VLAN の設定

リンク障害メッセージ、およびロード バランシング時の VLAN ブロッキング通知によって作成される遅延を回避するため、REP はハードウェア フラッド レイヤ（HFL）で通常のマルチキャスト アドレスにパケットをフラッディングします。これらのメッセージは REP セグメントだけではなくネットワーク全体にフラッディングされます。管理 VLAN を設定することで、これらのメッセージのフラッディングを制御できます。

REP 管理 VLAN を設定する場合、次の注意事項に従ってください。

- 管理 VLAN を設定しない場合、デフォルトは VLAN 1 です。
- すべてのセグメントに対し 1 つの管理 VLAN をスイッチで設定できます。
- 管理 VLAN は RSPAN VLAN になりません。

REP 管理 VLAN を設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **rep admin vlan** *vlan-id*
3. **end**
4. **show interface**［*interface-id*］**rep detail**
5. **copy running-config startup config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**rep admin vlan** *vlan-id* 例： スイッチ(config)# **rep admin vlan 2**	管理 VLAN を指定します。範囲は 2 ～ 4094 です。 管理 VLAN をデフォルトの 1 に設定するには、**no rep admin vlan** グローバル コンフィギュレーション コマンドを入力します。
ステップ 3	**end** 例： スイッチ(config)# **end**	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 4	**show interface**［*interface-id*］**rep detail** 例： スイッチ# **show interface gigabitethernet1/1 rep detail**	（任意）REP インターフェイスの設定を検証します。
ステップ 5	**copy running-config startup config** 例： スイッチ# **copy running-config startup config**	（任意）スイッチ スタートアップ コンフィギュレーション ファイルに設定を保存します。

REP インターフェイスの設定

REP を設定する場合、各セグメントインターフェイスで REP をイネーブルにして、セグメント ID を指定します。このタスクは必須で、他の REP 設定の前に実行する必要があります。また、各セグメントにプライマリおよびセカンダリ エッジ ポートを設定する必要があります。それ以外の手順はすべてオプションです。

インターフェイスで REP をイネーブルにし、設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode trunk**
5. **rep segment** *segment-id* [**edge**［**no-neighbor**］[**primary**]] [**preferred**]
6. **rep stcn** {**interface** *interface id* | **segment** *id-list* | **stp**}
7. **rep block port** {**id** *port-id* | *neighbor-offset* | **preferred**} **vlan** {*vlan-list* | **all**}
8. **rep preempt delay** *seconds*
9. **rep lsl-age-timer** *value*
10. **end**
11. **show interface**［*interface-id*］**rep** [**detail**]

12. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ# **interface gigabitethernet1/1**	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは物理レイヤ2インターフェイスまたはポート チャネル（論理インターフェイス）に設定できます。
ステップ 4	**switchport mode trunk** 例： スイッチ# **switchport mode trunk**	インターフェイスをレイヤ 2 トランク ポートとして設定します。
ステップ 5	**rep segment** *segment-id* [**edge** [**no-neighbor**] [**primary**]] [**preferred**] 例： スイッチ# **rep segment 1 edge no-neighbor primary**	インターフェイス上で REP をイネーブルにして、セグメント番号を特定します。指定できるセグメント ID の範囲は 1 ～ 1024 です。 （注） 各セグメントに 1 つのプライマリ エッジ ポートを含めて、2 つのエッジ ポートを設定する必要があります。 これらの任意のキーワードは利用可能です。 • （任意）**edge**：エッジ ポートとしてポートを設定します。各セグメントにあるエッジ ポートは2つだけです。**primary** キーワードなしで **edge** キーワードを入力すると、ポートがセカンダリエッジポートとして設定されます。 • （任意）**primary**：プライマリエッジポート（VLAN ロードバランシングを設定できるポート）としてポートを設定します。

	コマンドまたはアクション	目的
		• （任意）**no-neighbor**：エッジポートとして外部 REP ネイバーを使用せずにポートを設定します。ポートはエッジ ポートのすべてのプロパティを継承し、エッジ ポートの場合と同様にプロパティを設定できます。 （注） 各セグメントにあるプライマリエッジポートは1つだけですが、2つの異なるスイッチにエッジポートを設定して**primary**キーワードを両方のスイッチに入力しても、その設定は有効です。ただし、REP ではセグメント プライマリ エッジ ポートとして 1 つのポートだけが選択されます。**show rep topology** 特権 EXEC コマンドを入力すると、セグメントのプライマリエッジポートを特定できます。 • （任意）**preferred**：ポートが優先代替ポートであるか、VLAN ロードバランシングの優先ポートであるかを示します。 （注） ポートを優先に設定しても、代替ポートになるとは限りません。同等に可能性のあるポートよりやや可能性が高くなるだけです。通常、前に障害が発生したポートが、代替ポートとなります。
ステップ 6	**rep stcn** {**interface** *interface id* | **segment** *id-list* | **stp**} 例： スイッチ# **rep stcn segment 25-50**	（任意）STCN を送信するようにエッジ ポートを設定します。 • **interface** *interface-id*：物理インターフェイスまたはポート チャネルを指定して、STCN を受け取ります。 • **segment** *id-list*：STCN を受け取る 1 つ以上のセグメントを特定します。有効な範囲は 1 ～ 1024 です。 • **stp**：STCN を STP ネットワークに送信します。 （注） STCN を STP ネットワークに送信するために **rep stcn stp** コマンドを設定する場合は、スパニング ツリー（MST）モードがネイバーなしのエッジ ノード上に必要です。

	コマンドまたはアクション	目的
ステップ 7	**rep block port** {**id** *port-id* | *neighbor-offset* | **preferred**} **vlan** {*vlan-list* | **all**} 例： `スイッチ# rep block port id 0009001818D68700 vlan 1-100`	（任意）プライマリエッジポートに VLAN ロードバランシングを設定して、3 つの方法のいずれかを使用して REP 代替ポートを特定し（**id** *port-id*、*neighbor_offset*、**preferred**）、代替ポートでブロックされるように VLAN を設定します。 • **id** *port-id*：ポート ID で代替ポートを特定します。セグメント内の各ポートにポート ID が自動的に生成されます。**show interface** *type number* **rep** [**detail**] 特権 EXEC コマンドを入力し、インターフェイスポート ID を表示できます。 • *neighbor_offset*：エッジ ポートからのダウンストリーム ネイバーとして代替ポートを特定するための番号。有効範囲は -256 ～ 256 で、負数はセカンダリ エッジ ポートからのダウンストリーム ネイバーを示します。**0** の値が無効です。**-1** を入力して、セカンダリエッジポートを代替ポートとして識別します。 （注） プライマリエッジポート（オフセット番号 1）に **rep block port** コマンドを入力するので、代替ポートを特定するのにオフセット値 1 を入力できません。 • **preferred**：すでに VLAN ロード バランシングの優先代替ポートとして指定されている通常セグメント ポートを選択します。 • **vlan** *vlan-list*：1 つの VLAN または VLAN の範囲をブロックします。 • **vlan all**：すべての VLAN をブロックします。 （注） REP プライマリ エッジ ポート上にだけこのコマンドを入力します。
ステップ 8	**rep preempt delay** *seconds* 例： `スイッチ# rep preempt delay 100`	（任意）プリエンプト遅延時間を設定します。 • リンク障害が発生して復旧した後に、VLAN ロード バランシングを自動的にトリガーするには、このコマンドを使用します。 • 遅延時間の範囲は 15 ～ 300 秒です。デフォルトは、遅延時間のない手動によるプリエンプションです。

	コマンドまたはアクション	目的
		（注） REP プライマリ エッジ ポート上にだけこのコマンドを入力します。
ステップ 9	**rep lsl-age-timer** *value* 例： スイッチ# **rep lsl-age-timer 2000**	（任意）ネイバーからの hello が受信されないままどのくらいの時間（ミリ秒）が経過すると REP インターフェイスがダウンするかを設定します。 指定できる範囲は 120 ～ 10000 ミリ秒（40 ミリ秒単位）です。デフォルト値は 5000 ミリ秒（5 秒）です。 （注） • EtherChannel ポート チャネル インターフェイスでは、1000 ミリ秒未満の LSL エージング タイマー値はサポートされていません。 • リンクのフラップを避けるため、リンクの両方のポートに同じ LSL エージが設定されている必要があります。
ステップ 10	**end** 例： スイッチ(config)# **end**	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 11	**show interface** ［*interface-id*］ **rep** [**detail**] 例： スイッチ(config)# **show interface gigabitethernet1/1 rep detail**	（任意）REP インターフェイスの設定を表示します。
ステップ 12	**copy running-config startup-config** 例： スイッチ(config)# **copy running-config startup-config**	（任意）スイッチスタートアップコンフィギュレーション ファイルに設定を保存します。

VLAN ロード バランシングの手動によるプリエンプションの設定

プライマリエッジポートで **rep preempt delay** *seconds* インターフェイス コンフィギュレーション コマンドを入力しないで、プリエンプション時間遅延を設定する場合、デフォルトではセグメントで VLAN ロードバランシングを手動でトリガーします。手動で VLAN ロード バランシングをプリエンプトする前に、他のすべてのセグメント設定が完了しているかどうか確認してください。**rep preempt delay segment** *segment-id* コマンドを入力すると、プリエンプションによってネットワークが中断する可能性があるため、コマンド実行前に確認メッセージが表示されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **rep preempt segment** *segment-id*
4. **show rep topology segment** *segment-id*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**rep preempt segment** *segment-id* 例： スイッチ# **rep preempt segment 100** The command will cause a momentary traffic disruption. Do you still want to continue? [confirm]	手動により、セグメント上の VLAN ロード バランシングをトリガーします。 実行前にコマンドを確認する必要があります。
ステップ 4	**show rep topology segment** *segment-id* 例： スイッチ# **show rep topology segment 100**	（任意）REP トポロジの情報を表示します。
ステップ 5	**end** 例： スイッチ# **end**	特権 EXEC モードを終了します。

REP の SNMP トラップ設定

REP 固有のトラップを送信して、簡易ネットワーク管理プロトコル（SNMP）サーバにリンクの動作状態の変更およびすべてのポート役割の変更を通知するようにルータを設定できます。

手順の概要

1. **configure terminal**
2. **snmp mib rep trap-rate** *value*
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**snmp mib rep trap-rate** *value* 例： `スイッチ(config)# snmp mib rep trap-rate 500`	スイッチで REP トラップの送信をイネーブルにして、1 秒あたりのトラップの送信数を設定します。 • 1 秒あたりのトラップの送信数を入力します。範囲は 0 ～ 1000 です。デフォルトは 0（制限なし、発生するたびにトラップが送信される）です。
ステップ 3	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show running-config** 例： `スイッチ# show running-config`	（任意）実行コンフィギュレーションを表示します。これを使用して REP トラップ コンフィギュレーションを検証できます。
ステップ 5	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）スイッチ スタートアップ コンフィギュレーション ファイルに設定を保存します。

Resilient Ethernet Protocol 設定のモニタリング

このトピックのコマンドを使用して、REP インターフェイスと REP トポロジの詳細を表示できます。

- **show interface**［*interface-id*］**rep**［**detail**］

特定のインターフェイスまたはすべてのインターフェイスの REP の設定とステータスを表示します。

- （任意）**detail**：インターフェイス固有の REP 情報を表示します。

例：

```
Device# show interfaces TenGigabitEthernet4/1 rep detail

TenGigabitEthernet4/1 REP enabled
Segment-id: 3 (Primary Edge)
PortID: 03010015FA66FF80
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 02040015FA66FF804050
Port Role: Open
Blocked VLAN: <empty>
Admin-vlan: 1
Preempt Delay Timer: disabled
Configured Load-balancing Block Port: none
Configured Load-balancing Block VLAN: none
STCN Propagate to: none
LSL PDU rx: 999, tx: 652
HFL PDU rx: 0, tx: 0
BPA TLV rx: 500, tx: 4
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 6, tx: 5
EPA-COMMAND TLV rx: 0, tx: 0
EPA-INFO TLV rx: 135, tx: 136
```

- **show rep topology**［**segment** *segment-id*］［**archive** ］［**detail**］

  セグメント内のプライマリおよびセカンダリエッジポートを含む、1 セグメントまたは全セグメントの REP トポロジ情報を表示します。

  - （任意）**archive**：最後の安定したトポロジを表示します。

（注） アーカイブのトポロジは、スイッチをリロードすると保持されません。

  - （任意）**detail**：詳細なアーカイブ情報を表示します。

例：

```
Device# show rep topology

REP Segment 1
BridgeName       PortName   Edge Role
---------------- ---------- ---- ----
10.64.106.63     Te5/4      Pri  Open
10.64.106.228    Te3/4           Open
10.64.106.228    Te3/3           Open
10.64.106.67     Te4/3           Open
10.64.106.67     Te4/4           Alt
10.64.106.63     Te4/4      Sec  Open
```

```
REP Segment 3
BridgeName       PortName   Edge Role
---------------- ---------- ---- ----
10.64.106.63     Gi50/1     Pri  Open
SVT_3400_2       Gi0/3           Open
SVT_3400_2       Gi0/4           Open
10.64.106.68     Gi40/2          Open
10.64.106.68     Gi40/1          Open
10.64.106.63     Gi50/2     Sec  Alt
```

Resilient Ethernet Protocol の設定例

ここでは、次の設定例について説明します。

例：REP 管理 VLAN の設定

次に、管理 VLAN を VLAN 100 として設定して、REP インターフェイスの 1 つに **show interface rep detail** コマンドを入力して設定を確認する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# rep admin vlan 100
スイッチ(config)# end
スイッチ# show interface gigabitethernet1/1 rep detail

GigabitEthernet1/1 REP enabled
Segment-id: 2 (Edge)
PortID: 00010019E7144680
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 0002001121A2D5800E4D
Port Role: Open
Blocked Vlan: <empty>
Admin-vlan: 100
Preempt Delay Timer: disabled
LSL Ageout Timer: 5000 ms
Configured Load-balancing Block Port: none
Configured Load-balancing Block VLAN: none
STCN Propagate to: none
LSL PDU rx: 3322, tx: 1722
HFL PDU rx: 32, tx: 5
BPA TLV rx: 16849, tx: 508
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 118, tx: 118
EPA-COMMAND TLV rx: 0, tx: 0
EPA-INFO TLV rx: 4214, tx: 4190
```

次に、セグメントごとに管理 VLAN を作成する例を示します。ここでは、VLAN 2 は REP セグメント 2 でのみ管理 VLAN として設定されます。設定されていない残りのすべてのセグメントは、デフォルトで VLAN 1 が管理 VLAN となります。

```
スイッチ# configure terminal
スイッチ(config)# rep admin vlan 2 segment 2
スイッチ(config)# end
```

例：REP インターフェイスの設定

次に、インターフェイスをセグメント 1 のプライマリ エッジ ポートに設定し、STCN をセグメント 2 ～ 5 に送信し、代替ポートをポート ID 0009001818D68700 のポートとして設定して、セグメント ポート障害および回復後の 60 秒のプリエンプション遅延後にすべての VLAN をブロックする例を示します。このインターフェイスは、ネイバーからの hello が受信されないまま 6000 ミリ秒が経過するとダウンするように設定されています。

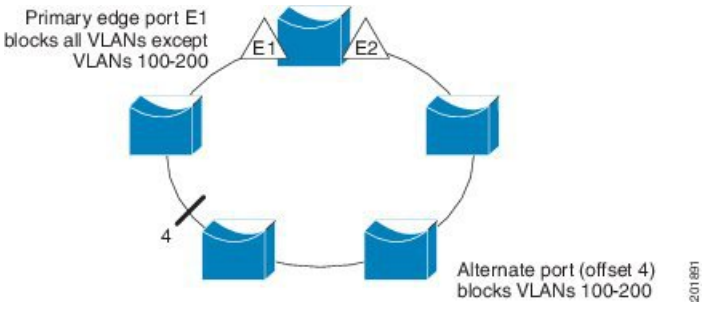
```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep stcn segment 2-5
Switch (conf-if)# rep block port 0009001818D68700 vlan all
Switch (conf-if)# rep preempt delay 60
Switch (conf-if)# rep lsl-age-timer 6000
Switch (conf-if)# end
```

次に、インターフェイスに外部 REP ネイバーがない場合の同じ設定の例を示します。

```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge no-neighbor primary
Switch (conf-if)# rep stcn segment 2-5
Switch (conf-if)# rep block port 0009001818D68700 vlan all
Switch (conf-if)# rep preempt delay 60
Switch (conf-if)# rep lsl-age-timer 6000
Switch (conf-if)# end
```

次に、図 5 のように VLAN ブロッキング コンフィギュレーションを設定する例を示します。代替ポートは、ネイバー オフセット番号 4 のネイバーです。手動プリエンプションのあと、VLAN 100 ～ 200 はこのポートでブロックされ、その他すべての VLAN はプライマリ エッジ ポート E1（ギガビット イーサネット ポート 1/1）でブロックされます。

図 *55 : VLAN* ブロッキングの例




```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep block port 4 vlan 100-200
Switch (conf-if)# end
```

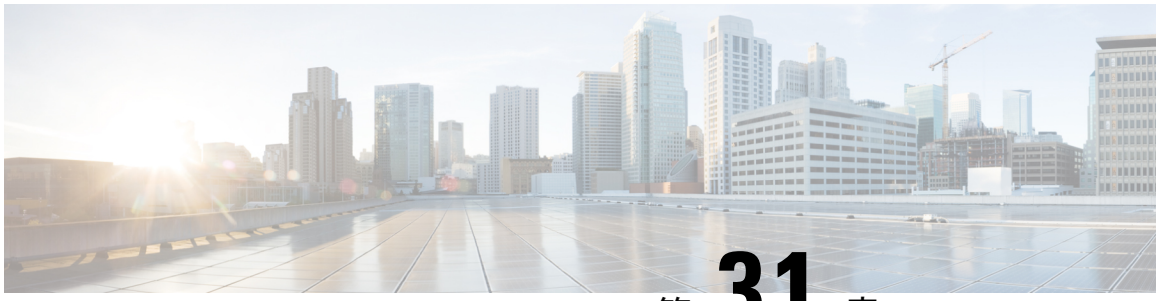
Resilient Ethernet Protocol の機能情報

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

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

表 *57 : Resilient Ethernet Protocol* の機能情報

機能名	リリース	機能情報
Resilient Ethernet Protocol	Cisco IOS リリース 15.2(6)E1	この機能が導入されました。 この機能は、Cisco IOS リリース 15.2(6)E1 で Cisco Catalyst 2960-L シリーズ スイッチと Cisco Catalyst 2960-X シリーズ スイッチでサポートされます。



第 31 章

Flex Link および MAC アドレス テーブル移動更新機能の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

Flex Link および MAC アドレス テーブル移動更新設定の制約事項

- Flex Link はレイヤ 2 ポートおよびポート チャネルだけでサポートされます。
- 最大 16 のバックアップ リンクを設定できます。

- アクティブ リンクには、Flex Link バックアップ リンクを 1 つだけ設定できます。バックアップ リンクは、アクティブ インターフェイスとは異なるインターフェイスにする必要があります。
- インターフェイスが所属できる Flex Link ペアは 1 つだけです。インターフェイスは、1 つだけのアクティブ リンクのバックアップ リンクにすることができます。アクティブ リンクは、別の Flex Link ペアに属することができません。
- どちらのリンクも、EtherChannel に属するポートには設定できません。ただし、2 つのポート チャネル（EtherChannel 論理インターフェイス）を Flex Link として設定でき、ポート チャネルおよび物理インターフェイスを Flex Link として設定して、ポート チャネルか物理インターフェイスのどちらかをアクティブ リンクにすることができます。
- バックアップ リンクはアクティブ リンクと同じタイプ（ギガビット イーサネットまたはポート チャネル）にする必要はありません。ただし、スタンバイ リンクがトラフィック転送を開始した場合にループが発生したり動作が変更したりしないように、両方の Flex Link を同様の特性で設定する必要があります。
- Flex Link ポートでは STP がディセーブルになります。ポート上にある VLAN が STP 用に設定されている場合でも、Flex Link ポートは STP に参加しません。STP がイネーブルでない場合は、設定されているトポロジでループが発生しないようにしてください。

Flex Link および MAC アドレス テーブル移動更新に関する情報

Flex Link

Flex Link は、レイヤ 2 インターフェイス（device ポートまたはポート チャネル）のペアで、一方のインターフェイスが他方のインターフェイスのバックアップとして機能するように設定されます。この機能は、スパニングツリー プロトコル（STP）の代替ソリューションです。ユーザーは、STP をディセーブルにしても、基本的リンク冗長性を保つことができます。Flex Link は、通常、ユーザーがdeviceでSTPを実行したくない場合に、サービス プロバイダまたは企業ネットワークで設定されます。deviceが STP を実行中の場合は、STP がすでにリンク レベルの冗長性またはバックアップを提供しているため、Flex Link は不要です。

別のレイヤ 2 インターフェイスを Flex Link またはバックアップ リンクとして割り当てることで、1 つのレイヤ 2 インターフェイス（アクティブ リンク）に Flex Link を設定します。devicesでは、Flex Link を、同じdeviceまたはスタックの別のdevice上で使用できます。リンクの 1 つがアップでトラフィックを転送しているときは、もう一方のリンクがスタンバイ モードで、このリンクがシャット ダウンした場合にトラフィックの転送を開始できるように準備しています。どの時点でも、1 つのインターフェイスのみがリンクアップ ステートでトラフィックを転送しています。プライマリ リンクがシャットダウンされると、スタンバイ リンクがトラフィックの転送を開始します。アクティブ リンクがアップに戻った場合はスタンバイ モードになり、

トラフィックが転送されません。STP は Flex Link インターフェイス上ではディセーブル化されています。

Flex Link の設定

次の図で、device A のポート 1 と 2 はアップリンクスイッチ B と C に接続されています。それらは Flex Link として設定されているため、インターフェイスのうち 1 つだけがトラフィックを転送し、その他はスタンバイモードになります。ポート 1 がアクティブ リンクになる場合、ポート 1 とスイッチ B との間でトラフィックの転送を開始し、ポート 2（バックアップ リンク）とスイッチ C との間のリンクでは、トラフィックは転送されません。ポート 1 がダウンすると、ポート 2 がアップ状態になってスイッチ C へのトラフィックの転送を開始します。ポート 1 が再びアップ状態に戻ってもスタンバイ モードになり、トラフィックを転送しません。ポート 2 がトラフィック転送を続けます。

また、トラフィックを転送する優先ポートを指定して、プリエンプション機能を設定できます。たとえば、プリエンプション モードと Flex Link ペアを設定できます。図のシナリオでは、ポート 1 がバックアップとなって、ポート 2 より帯域幅が大きい場合、ポート 1 は 60 秒後にパケットの転送を開始します。ポート 2 がスタンバイとなります。これを行うには、**switchport backup interface preemption mode bandwidth** および **switchport backup interface preemption delay** インターフェイス コンフィギュレーション コマンドを入力します。

プライマリ（転送）リンクがダウンすると、トラップによってネットワーク管理ステーションが通知を受けます。スタンバイ リンクがダウンすると、トラップによってユーザーが通知を受けます。

Flex Link はレイヤ 2 ポートおよびポート チャネルだけでサポートされ、VLAN またはレイヤ 3 ポートではサポートされません。

VLAN Flex Link ロード バランシングおよびサポート

VLAN Flex Link ロード バランシングにより、ユーザーは相互排他的な VLAN のトラフィックを両方のポートで同時に転送するように Flex Link ペアを設定できます。たとえば、Flex Link ポートが 1 ～ 100 の VLAN に対して設定されている場合、最初の 50 の VLAN のトラフィックを 1 つのポートで転送し、残りの VLAN のトラフィックをもう一方のポートで転送できます。どちらかのポートで障害が発生した場合には、もう一方のアクティブ ポートがすべてのトラフィックを転送します。障害が発生したポートが元に戻ると、優先 VLAN のトラフィックの転送を再開します。冗長性を提供する以外に、この Flex Link のペアはロード バランシングに使用できます。Flex Link VLAN ロード バランシングによってアップリンク devicesが制約を受けることはありません。

Flex Link フェールオーバーによるマルチキャスト高速コンバージェンス

Flex Link マルチキャスト高速コンバージェンスにより、Flex Link 障害発生後のマルチキャスト トラフィック コンバージェンス時間が短縮されます。マルチキャスト高速コンバージェン

スはmrouterポートとしてのバックアップ リンクの学習、IGMP レポートの生成、およびIGMP レポートのリークを組み合わせて実行されます。

その他の Flex Link ポートを mrouter ポートとして学習

通常のマルチキャスト ネットワークでは、個々の VLAN について 1 つのクエリアが選定されます。ネットワーク エッジに展開されたdeviceには、クエリーを受信するいずれかの Flex Link ポートが存在します。Flex Link ポートは常に、転送状態になります。

クエリーを受信するポートが、deviceのmrouterポートとして追加されます。mrouterポートは、deviceが学習したすべてのマルチキャスト グループの 1 つとして認識されます。切り替えの後、クエリーは別のFlex Linkポートによって受信されます。この別のFlex Linkポートはmrouterポートとして認識されるようになります。切り替えの後、マルチキャスト トラフィックは別の Flex Link ポートを介して流れます。トラフィック コンバージェンスを高速化するために、いずれかの Flex Link ポートが mrouter ポートとして学習されると、両方の Flex Link ポートが mrouter ポートとして認識されます。いずれの Flex Link ポートも常に、マルチキャスト グループの一部として扱われます。

通常の動作モードではいずれのFlex Linkポートもグループの一部として認識されますが、バックアップポートを通過するトラフィックはすべてブロックされます。mrouterポートとしてバックアップ ポートを追加しても、通常のマルチキャスト データ フローが影響を受けることはありません。切り替えが生じると、バックアップポートのブロックが解除され、トラフィックが流れるようになります。この場合、バックアップポートのブロックが解除されるとただちに、アップストリーム データが流れ始めます。

生成する、IGMP レポートを

切り替えの後、バックアップ リンクがアップ状態になると、アップストリームでの新しいディストリビューション device でのマルチキャスト データの転送は開始されません。これは、ブロックされた Flex Link ポートに接続されているアップストリーム ルータのポートが、マルチキャスト グループの一部として認識されないからです。マルチキャスト グループのレポートは、バックアップ リンクがブロックされているため、ダウンストリーム deviceで転送されませんでした。このポートのデータは、マルチキャストグループが学習されるまで流れません。マルチキャスト グループの学習は、レポートを受信した後にだけ行われます。

レポートは、一般クエリーが受信されると、ホストより送信されます。一般クエリーは、通常のシナリオであれば 60 秒以内に送信されます。バックアップ リンクが転送を開始し、マルチキャストデータの高速コンバージェンスを達成できるようになると、ダウンストリーム deviceが一般クエリーを待つことなく、ただちにこのポート上のすべての学習済みグループに対し、プロキシ レポートを送信します。

リークする、IGMP レポートを

マルチキャスト トラフィック コンバージェンスを最小限の損失で達成できるように、Flex Linkのアクティブ リンクがダウンする前に冗長データ パスを設定しておく必要があります。これは、Flex Link バックアップ リンクで IGMP レポート パケットだけをリークさせることで行えます。こうしてリークさせたIGMPレポートメッセージがアップストリームのディストリビューション ルータで処理されるため、マルチキャスト データのトラフィックはバックアップ イン

ターフェイスに転送されます。バックアップインターフェイスの着信トラフィックはすべてアクセスdeviceの入り口部分でドロップされるため、ホストが重複したマルチキャストトラフィックを受信することはありません。Flex Link のアクティブ リンクに障害が発生した場合、ただちにアクセス deviceがバックアップ リンクからのトラフィックを受け入れ始めます。このスキームの唯一の欠点は、ディストリビューション devices間のリンク、およびディストリビューションとアクセス devicesの間のバックアップ リンクで帯域幅が大幅に消費される点です。この機能はデフォルトでは無効に設定されていて、**switchport backup interface** *interface-id* **multicast fast-convergence** コマンドを使用することにより設定できます。

切り替え時にこの機能がイネーブルになっている場合、deviceで転送ポートに設定されたバックアップ ポート上でプロキシ レポートは生成されません。

MAC アドレス テーブル移動更新

MAC アドレス テーブル移動更新機能により、プライマリ（転送）リンクがダウンしてスタンバイ リンクがトラフィックの転送を開始したときに、deviceで高速双方向コンバージェンスが提供されます。

Flex Link の VLAN ロード バランシング設定時の注意事項

- Flex Link VLAN ロード バランシングでは、バックアップ インターフェイス上で優先される VLAN を選択する必要があります。
- 同じ Flex Link ペアに対して、プリエンプション メカニズムと VLAN ロード バランシングを設定することはできません。

MAC アドレス テーブル移動更新設定時の注意事項

- アクセス deviceでこの機能のイネーブル化と設定を行うと、MAC アドレス テーブル移動更新を送信（*send*）できます。
- MACアドレステーブル移動更新メッセージを取得（*get*）する場合、この機能をアップリンク devicesでイネーブルにして設定します。

デフォルトの Flex Link および MAC アドレス テーブル移動更新の設定

- Flex Link は設定されておらず、バックアップ インターフェイスは定義されていません。
- プリエンプション モードはオフです。
- プリエンプション遅延は 35 秒です。
- MAC アドレス テーブル移動更新機能は、 device上で設定されません。

Flex Link および MAC アドレス テーブル移動更新機能の設定方法

Flex Link の設定

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(conf)# interface gigabitethernet1/0/1`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは物理レイヤ 2 インターフェイスまたはポート チャネル（論理インターフェイス）に設定できます。
ステップ 3	**switchport backup interface** *interface-id* 例： `スイッチ(conf-if)# switchport backup interface gigabitethernet1/0/2`	物理レイヤ 2 インターフェイス（ポート チャネル）をインターフェイスがある Flex Link ペアの一部として設定します。1 つのリンクがトラフィックを転送している場合、もう一方のインターフェイスはスタンバイ モードです。
ステップ 4	**end** 例： `スイッチ(conf-if)# end`	特権 EXEC モードに戻ります。

Flex Link ペアのプリエンプション方式の設定

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**]
5. **switchport backup interface** *interface-id* **preemption delay** *delay-time*
6. **end**
7. **show interface** [*interface-id*] **switchport backup**
8. **copy running-config startup config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(conf)# interface gigabitethernet1/0/1`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは物理レイヤ 2 インターフェイスまたはポート チャネル（論理インターフェイス）に設定できます。
ステップ 3	**switchport backup interface** *interface-id* 例： `スイッチ(conf-if)# switchport backup interface gigabitethernet1/0/2`	物理レイヤ 2 インターフェイス（ポート チャネル）をインターフェイスがある Flex Link ペアの一部として設定します。1 つのリンクがトラフィックを転送している場合、もう一方のインターフェイスはスタンバイ モードです。
ステップ 4	**switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**] 例： `スイッチ(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption mode forced`	Flex Link インターフェイス ペアのプリエンプション メカニズムおよび遅延を設定します。次のプリエンプション モードを設定することができます。 • **forced**：（任意）アクティブ インターフェイスはバックアップを常にプリエンプトします。 • **bandwidth**：（任意）より大きい帯域幅のインターフェイスが常にアクティブ インターフェイスとして動作します。 • **off**：（任意）アクティブからバックアップへのプリエンプトは発生しません。

	コマンドまたはアクション	目的
ステップ 5	**switchport backup interface** *interface-id* **preemption delay** *delay-time* 例： `スイッチ(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption delay 50`	ポートが他のポートより先に使用されるまでの遅延時間を設定します。 （注） 遅延時間の設定は、forced モードおよび bandwidth モードでのみ有効です。
ステップ 6	**end** 例： `スイッチ(conf-if)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show interface** [*interface-id*] **switchport backup** 例： `スイッチ# show interface gigabitethernet1/0/2 switchport backup`	設定を確認します。
ステップ 8	**copy running-config startup config** 例： `スイッチ# copy running-config startup config`	（任意）device スタートアップ コンフィギュレーション ファイルに設定を保存します。

Flex Link の VLAN ロード バランシングの設定

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id* **prefer vlan** *vlan-range*
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**interface** *interface-id* 例： `スイッチ（config)# interface gigabitethernet2/0/6`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは物理レイヤ 2 インターフェイスまたはポート チャネル（論理インターフェイス）に設定できます。
ステップ 3	**switchport backup interface** *interface-id* **prefer vlan** *vlan-range* 例： `スイッチ（config-if)# switchport backup interface gigabitethernet2/0/8 prefer vlan 2`	物理レイヤ 2 インターフェイス（またはポート チャネル）を、インターフェイスを装備した Flex Link ペアの一部として設定し、インターフェイス上の VLAN を指定します。VLAN ID の範囲は 1 ～ 4094 です。
ステップ 4	**end** 例： `スイッチ（config-if)# end`	特権 EXEC モードに戻ります。

MAC アドレス テーブル移動更新の設定

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. 次のいずれかを使用します。
   - **switchport backup interface** *interface-id*
   - **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id*
4. **end**
5. **mac address-table move update transmit**
6. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**interface** *interface-id* 例： スイッチ#**interface gigabitethernet1/0/1**	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは物理レイヤ 2 インターフェイスまたはポート チャネル（論理インターフェイス）に設定できます。
ステップ 3	次のいずれかを使用します。 • **switchport backup interface** *interface-id* • **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id* 例： スイッチ(config-if)# **switchport backup interface gigabitethernet0/2 mmu primary vlan 2**	物理レイヤ 2 インターフェイス（またはポート チャネル）を、インターフェイスを装備した Flex Link ペアの一部として設定します。MAC アドレス テーブル移動更新 VLAN はインターフェイスで最も低い VLAN ID です。 物理レイヤ 2 インターフェイス（ポート チャネル）を設定し、MAC アドレス テーブル移動更新の送信に使用される VLAN ID をインターフェイスで指定します。 1 つのリンクがトラフィックを転送している場合、もう一方のインターフェイスはスタンバイ モードです。
ステップ 4	**end** 例： スイッチ(config-if)# **end**	グローバル コンフィギュレーション モードに戻ります。
ステップ 5	**mac address-table move update transmit** 例： スイッチ(config)# **mac address-table move update transmit**	プライマリ リンクがダウンし、スタンバイ リンクを介してdeviceがトラフィックの転送を開始すると、アクセス deviceで、ネットワークの他のdevicesに MAC アドレス テーブル移動更新を送信できます。 MMU パケットが MAC テーブルを更新するように、device でコマンド **mac address-table move update** を入力します。プライマリ リンクが復帰すると、MAC テーブルは再収束する必要があり、このコマンドによって MMU が送信され、動作が確立されます。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

MAC アドレス テーブル移動更新メッセージの取得および処理用のデバイス設定

手順の概要

1. **configure terminal**
2. **mac address-table move update receive**
3. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mac address-table move update receive** 例： `スイッチ (config)# mac address-table move update receive`	deviceで MAC アドレス テーブル移動更新の取得と処理を可能にします。
ステップ 3	**end** 例： `スイッチ (config)# end`	特権 EXEC モードに戻ります。

Flex Link、マルチキャスト高速コンバージェンス、および MAC アドレス テーブル移動更新の監視

コマンド	目的
show interface [*interface-id*] **switchport backup**	インターフェイス用に設定された Flex Link バックアップ インターフェイス、または設定されたすべての Flex Link と、各アクティブ インターフェイスおよびバックアップ インターフェイスの状態（アップまたはスタンバイ モード）を表示します。

コマンド	目的
show ip igmp profile address-table move update *profile-id*	特定の IGMP プロファイルまたはdevice上で定義されているすべての IGMP プロファイルを表示します。
show mac address-table move update	device上に MAC アドレステーブル移動移動を表示します。

Flex Link の設定例

Flex Link の設定：例

この例では、バックアップインターフェイスでインターフェイスを設定した後に、設定を確認する方法を示します。

```
スイッチ# show interface switchport backup

Switch Backup Interface Pairs:
Active Interface Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/0/1 GigabitEthernet1/0/2 Active Up/Backup Standby
```

この例では、バックアップ インターフェイス ペアにプリエンプション モードを強制として設定した後に、設定を確認する方法を示します。

```
スイッチ# show interface switchport backup detail

Switch Backup Interface Pairs:

Active Interface Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/0/211 GigabitEthernet1/0/2 Active Up/Backup Standby
Interface Pair : Gi1/0/1, Gi1/0/2
Preemption Mode : forced
Preemption Delay : 50 seconds
Bandwidth : 100000 Kbit (Gi1/0/1), 100000 Kbit (Gi1/0/2)
Mac Address Move Update Vlan : auto
```

Flex Link における VLAN ロード バランシングの設定：例

次の例では、deviceに VLAN 1 ～ 50、60、および 100 ～ 120 を設定する例を示します。

```
スイッチ(config)# interface gigabitethernet 2/0/6
スイッチ(config-if)# switchport backup interface gigabitethernet 2/0/8 prefer vlan
60,100-120
```

両方のインターフェイスが起動しているとき、Gi2/0/8 は VLAN 60 および 100 ～ 120 のトラフィックを転送し、Gi2/0/6 は VLAN 1 ～ 50 のトラフィックを転送します。

```
スイッチ# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Up/Backup Standby

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

Flex Link インターフェイスがダウンすると（LINK_DOWN）、このインターフェイスで優先される VLAN は、Flex Link ペアのピア インターフェイスに移動します。この例では、インターフェイス Gi2/0/6 がダウンして、Gi2/0/8 が Flex Link ペアのすべての VLAN を引き継ぎます。

```
スイッチ# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Down/Backup Up

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

FlexLink インターフェイスがアップになると、このインターフェイスで優先される VLAN はピア インターフェイスでブロックされ、アップしたインターフェイスでフォワーディング ステートに移動します。次に、インターフェイス Gi2/0/6 が起動すると、このインターフェイスの優先 VLAN は、ピア インターフェイス Gi2/0/8 ではブロックされ、Gi2/0/6 で転送されます。

```
スイッチ# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Up/Backup Standby

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120

スイッチ# show interfaces switchport backup detail

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
FastEthernet1/0/3       FastEthernet1/0/4       Active Down/Backup Up

Vlans Preferred on Active Interface: 1-2,5-4094
Vlans Preferred on Backup Interface: 3-4
Preemption Mode  : off
```

```
Bandwidth : 10000 Kbit (Fa1/0/3), 100000 Kbit (Fa1/0/4)
Mac Address Move Update Vlan : auto
```

MAC アドレス テーブル移動更新の設定：例

この例では、MAC アドレス テーブル移動更新を送信するためアクセス deviceを設定した後に設定を確認する方法を示します。

```
スイッチ# show mac address-table move update

Switch-ID : 010b.4630.1780
Dst mac-address : 0180.c200.0010
Vlans/Macs supported : 1023/8320
Default/Current settings: Rcv Off/On, Xmt Off/On
Max packets per min : Rcv 40, Xmt 60
Rcv packet count : 5
Rcv conforming packet count : 5
Rcv invalid packet count : 0
Rcv packet count this min : 0
Rcv threshold exceed count : 0
Rcv last sequence# this min : 0
Rcv last interface : Po2
Rcv last src-mac-address : 000b.462d.c502
Rcv last switch-ID : 0403.fd6a.8700
Xmt packet count : 0
Xmt packet count this min : 0
Xmt threshold exceed count : 0
Xmt pak buf unavail cnt : 0
Xmt last interface : None
```

Flex Link フェールオーバーによるマルチキャスト高速コンバージェンスの設定：例

次に、Flex Link を GigabitEthernet1/0/11 および GigabitEthernet1/0/12 に設定したときに他の Flex Link ポートを mrouter ポートとして学習する設定例と、**show interfaces switchport backup** コマンドの出力を示します。

```
スイッチ# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
スイッチ(config)# interface GigabitEthernet1/0/11
スイッチ(config-if)# switchport trunk encapsulation dot1q
スイッチ(config-if)# switchport mode trunk
スイッチ(config-if)# switchport backup interface GigabitEthernet1/0/12
スイッチ(config-if)# exit
スイッチ(config)# interface GigabitEthernet1/0/12
スイッチ(config-if)# switchport trunk encapsulation dot1q
スイッチ(config-if)# switchport mode trunk
スイッチ(config-if)# end
スイッチ# show interfaces switchport backup detail
Switch Backup Interface Pairs:
Active Interface Backup Interface State
```

```
GigabitEthernet1/0/11 GigabitEthernet1/0/12 Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : Off
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

この出力は、GigabitEthernet1/0/11 を介してdeviceに到達するクエリーのある、VLAN 1 および 401 のクエリアを示します。

```
スイッチ# show ip igmp snooping querier

Vlan  IP Address  IGMP Version  Port
-----------------------------------------------------------
1     10.0.0.10    v2      Gi1/0/11
401   41.41.41.1  v2      Gi1/0/11
```

この例では、VLAN 1 および VLAN 401 に対する **show ip igmp snooping mrouter** コマンドの出力を示します。

```
スイッチ# show ip igmp snooping mrouter

Vlan   ports
----   -----
1      Gi1/0/11(dynamic), Gi1/0/12(dynamic)
401    Gi1/0/11(dynamic), Gi1/0/12(dynamic)
```

同様に、両方の Flex Link ポートが学習されたグループに属しています。次の例では、GigabitEthernet2/0/11 は VLAN 1 のレシーバ/ホストであり、2 つのマルチキャスト グループに関連しています。

```
スイッチ# show ip igmp snooping groups

Vlan   Group    Type   Version    Port List
------------------------------------------------------------------------
1   228.1.5.1    igmp   v2    Gi1/0/11, Gi1/0/12, Gi2/0/11
1   228.1.5.2    igmp   v2    Gi1/0/11, Gi1/0/12, Gi2/0/11
```

ホストが一般クエリーに応答するときに、deviceはすべてのマルチキャスト ルータ ポートに関するこのレポートを転送します。次の例では、ホストがグループ 228.1.5.1 のレポートを送信するとき、バックアップ ポート GigabitEthernet1/0/12 はブロックされているので、レポートは GigabitEthernet1/0/11 でだけ送信されます。アクティブ リンク GigabitEthernet1/0/11 がダウンすると、バックアップ ポート GigabitEthernet1/0/12 が転送を開始します。

このポートが転送を開始すると、ただちにdeviceがホストに代わり、228.1.5.1 と 228.1.5.2 のグループにプロキシ レポートを送信します。アップストリーム ルータはグループを学習し、マルチキャスト データの転送を開始します。これは、Flex Link のデフォルトの動作です。この動作は、ユーザーが **switchport backup interface gigabitEthernet 1/0/12 multicast fast-convergence** コマンドを使用して高速コンバージェンスを設定した場合に変更されます。次に、この機能をオンにする例を示します。

```
スイッチ# configure terminal
```

```
Enter configuration commands, one per line. End with CNTL/Z.
スイッチ(config)# interface gigabitEthernet 1/0/11
スイッチ(config-if)# switchport backup interface gigabitEthernet 1/0/12 multicast
fast-convergence
スイッチ(config-if)# exit
スイッチ# show interfaces switchport backup detail

Switch Backup Interface Pairs:
Active       Interface       Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/0/11  GigabitEthernet1/0/12  Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : On
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

この出力は、GigabitEthernet1/0/11 を介してdeviceに到達するクエリーのある、VLAN 1 および 401 のクエリアを示します。

```
スイッチ# show ip igmp snooping querier

Vlan  IP Address  IGMP Version  Port
-----------------------------------------------------------
1     10.0.0.10   v2      Gi1/0/11
401   41.41.41.1  v2      Gi1/0/11
```

次に VLAN 1 と 401 に対する **show ip igmp snooping mrouter** コマンドの出力を示します。

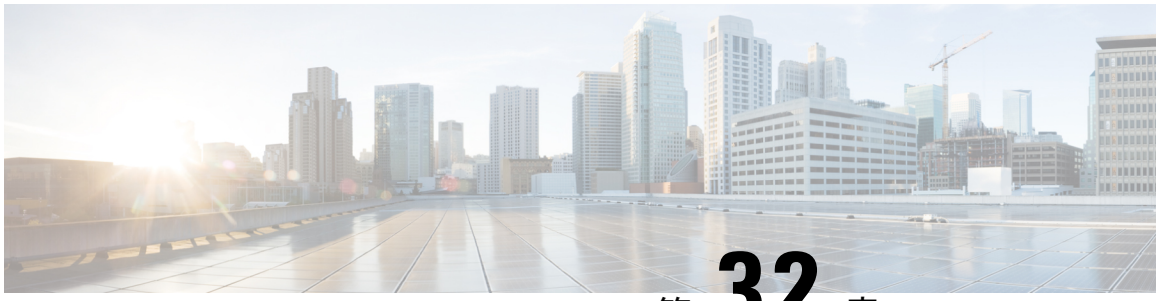
```
スイッチ# show ip igmp snooping mrouter

Vlan    ports
----    -----
1    Gi1/0/11(dynamic), Gi1/0/12(dynamic)
401    Gi1/0/11(dynamic), Gi1/0/12(dynamic)
```

同様に、両方の Flex Link ポートが学習されたグループに属しています。次の例では、GigabitEthernet2/0/11 は VLAN 1 のレシーバ/ホストであり、2 つのマルチキャスト グループに関連しています。

```
スイッチ# show ip igmp snooping groups

Vlan   Group    Type   Version    Port List
------------------------------------------------------------------------
1   228.1.5.1    igmp   v2    Gi1/0/11, Gi1/0/12, Gi2/0/11
1   228.1.5.2    igmp   v2    Gi1/0/11, Gi1/0/12, Gi2/0/11
```

一般クエリーに対してあるホストが応答すると必ず、deviceがすべての mrouter ポートに関するこのレポートを転送します。コマンドラインポートを使用してこの機能をオンにすると、レポートは、GigabitEthernet1/0/11 上のdeviceによって転送されるときにバックアップ ポート GigabitEthernet1/0/12 にも送信されます。アップストリーム ルータはグループを学習し、マルチキャスト データの転送を開始します。GigabitEthernet1/0/12 はブロックされているので、このデータは入力でドロップされます。アクティブ リンク GigabitEthernet1/0/11 がダウンすると、バックアップ ポート GigabitEthernet1/0/12 が転送を開始します。マルチキャスト データはアッ

プストリーム ルータによりすでに転送されているため、いずれのプロキシ レポートも送信する必要がありません。レポートをバックアップ ポートにリークすると冗長マルチキャスト パスが設定され、マルチキャスト トラフィック コンバージェンス用の時間が最小限になります。



第 32 章

単方向リンク検出の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

UDLD 設定の制約事項

次に、単方向リンク検出（UDLD）設定の制約事項を示します。

- UDLD 対応ポートが別のdeviceの UDLD 非対応ポートに接続されている場合、このポートは単一方向リンクを検出できません。
- モード（通常またはアグレッシブ）を設定する場合、リンクの両側に同じモードを設定します。

⚠

注意 ループ ガードは、ポイントツーポイント リンクでのみサポートされます。リンクの各終端には、STP を実行するデバイスを直接接続することを推奨します。

UDLD について

UniDirectional Link Detection（UDLD）は、光ファイバまたはツイストペア イーサネット ケーブルを通して接続されたデバイスからケーブルの物理設定をモニタリングしたり、単一方向リンクの存在を検出できるようにするためのレイヤ2プロトコルです。このプロトコルが単一方向リンクを正常に識別してディセーブルにするには、接続されたすべてのデバイスで UDLD プロトコルがサポートされている必要があります。UDLDは単一方向リンクを検出すると、影響を受けるポートをディセーブルにして警報を発信します。単一方向リンクは、スパニングツリー トポロジ ループをはじめ、さまざまな問題を引き起こす可能性があります。

動作モード

UDLD しています。通常（デフォルト）とアグレッシブです。通常モードのUDLDは、光ファイバ接続におけるポートの誤った接続による単一方向リンクを検出できます。アグレッシブ モードの UDLD は、光ファイバ リンクおよびツイストペア リンク上の片方向トラフィックと、光ファイバ リンク上のポートの誤った接続による単一方向リンクも検出できます。

通常およびアグレッシブの両モードの UDLD は、レイヤ 1 のメカニズムを使用して、リンクの物理ステータスを学習します。レイヤ1では、物理的シグナリングおよび障害検出は、自動ネゴシエーションによって処理されます。UDLD は、ネイバー ID の検出、誤って接続されたポートのシャットダウンなど、自動ネゴシエーションでは実行不可能な処理を実行します。自動ネゴシエーションと UDLD の両方をイネーブルにすると、レイヤ 1 と 2 の検出機能が連動し、物理的および論理的な単一方向接続、および他のプロトコルの誤動作を防止します。

ローカルデバイスが送信したトラフィックをネイバーが受信するにもかかわらず、ネイバーから送信されたトラフィックをローカルデバイスが受信しない場合に、単一方向リンクが発生します。

通常モード

通常モードの UDLD は、光ファイバ ポートの光ファイバが誤って接続されている場合に単一方向リンクを検出しますが、レイヤ1メカニズムは、この誤った接続を検出しません。ポートが正しく接続されていてもトラフィックが片方向である場合、単一方向リンクを検出するはずのレイヤ1メカニズムがこの状況を検出できないため、UDLDは単一方向リンクを検出できません。この場合、論理リンクは不確定と見なされ、UDLD はポートをディセーブルにしません。

UDLD が通常モードのときに、ペアの一方の光ファイバが切断されており、自動ネゴシエーションがアクティブであると、レイヤ1メカニズムがリンクの物理的な問題を検出するため、リンクは稼働状態でなくなります。この場合は、UDLDは何のアクションも行わず、論理リンクは不確定と見なされます。

アグレッシブモード

アグレッシブ モードでは、UDLD はこれまでの検出方法で単一方向リンクを検出します。アグレッシブ モードの UDLD は、2 つのデバイス間の障害発生が許されないポイントツーポイ

ントリンクの単一方向リンクも検出できます。また、次のいずれかの問題が発生している場合に、単一方向リンクも検出できます。

- 光ファイバ リンクまたはツイストペア リンクで、ポートの 1 つがトラフィックを送受信できない。
- 光ファイバ リンクまたはツイストペア リンクで、ポートの 1 つがダウンし、残りのインターフェイスが稼働している。
- ケーブルのうち 1 本の光ファイバが切断されている。

これらの場合、UDLD は影響を受けたポートをディセーブルにします。

ポイントツーポイントリンクでは、UDLD helloパケットをハートビートと見なすことができ、ハートビートがあればリンクは正常です。逆に、ハートビートがないということは、双方向リンクを再確立できない限り、リンクをシャットダウンする必要があることを意味しています。

レイヤ1の観点からケーブルの両方の光ファイバが正常な状態であれば、アグレッシブ モードの UDLD はそれらの光ファイバが正しく接続されているかどうか、およびトラフィックが正しいネイバー間で双方向に流れているかどうかを検出します。自動ネゴシエーションはレイヤ 1 で動作するため、このチェックは自動ネゴシエーションでは実行できません。

単一方向の検出方法

UDLD は、2 つの方法で動作します。

- ネイバー データベース メンテナンス
- イベントドリブン検出およびエコー

ネイバー データベース メンテナンス

UDLD は、アクティブな各ポート上で hello パケット（別名アドバタイズまたはプローブ）を定期的に送信して、他の UDLD 対応ネイバーに関して学習し、各デバイスがネイバーに関する情報を常に維持できるようにします。

deviceが hello メッセージを受信すると、エージング タイム（ホールド タイムまたは存続可能時間）が経過するまで、情報をキャッシュします。古いキャッシュ エントリの期限が切れる前に、deviceが新しい hello メッセージを受信すると、deviceが古いエントリを新しいエントリで置き換えます。

UDLDの実行中にポートがディセーブルになったり、ポート上でUDLDがディセーブルになったり、またはdeviceをリセットした場合、UDLD は設定変更の影響を受けるポートの既存のキャッシュ エントリをすべてクリアします。UDLDは、ステータス変更の影響を受けるキャッシュの一部をフラッシュするよう、ネイバーに通知するメッセージを1つまたは複数送信します。このメッセージは、キャッシュを継続的に同期するためのものです。

イベントドリブン検出およびエコー

UDLD は検出動作としてエコーを利用します。UDLD デバイスが新しいネイバーを学習するか、または同期していないネイバーから再同期要求を受信すると、接続の UDLD デバイス側の検出ウィンドウを再起動して、エコー メッセージを返送します。この動作はすべての UDLD ネイバーに対して同様に行われるため、エコー送信側では返信エコーを受信するように待機します。

検出ウィンドウが終了し、有効な応答メッセージが受信されなかった場合、リンクは、UDLD モードに応じてシャットダウンされることがあります。UDLD が通常モードにある場合、リンクは不確定と見なされ、シャットダウンされない場合があります。UDLD がアグレッシブ モードにある場合は、リンクは単一方向と見なされ、ポートはディセーブルになります。

UDLD リセット オプション

インターフェイスが UDLD でディセーブル化された場合、次のオプションの 1 つを使用して UDLD をリセットできます。

- **udld reset** インターフェイス コンフィギュレーション コマンドです。
- **no shutdown** インターフェイス コンフィギュレーション コマンドに続いて **shutdown** インターフェイス コンフィギュレーション コマンドを入力すると、ディセーブル化されたポートを再起動できます。
- **no udld** {**aggressive** | **enable**} グローバル コンフィギュレーション コマンドの後に **udld** {**aggressive** | **enable**} グローバル コンフィギュレーション コマンドが続くと、無効なポートが再度イネーブルになります。
- **no udld port** インターフェイス コンフィギュレーション コマンドに続いて **udld port** [**aggressive**] インターフェイス コンフィギュレーション コマンドを入力すると、無効なファイバー オプティック ポートがイネーブルになります。
- **errdisable recovery cause udld** グローバル コンフィギュレーション コマンドを使用すると、UDLD の errdisable ステートから自動回復するタイマーをイネーブルにできます。さらに、**errdisable recovery interval** *interval* グローバル コンフィギュレーション コマンドでは、udld errdisable ステートから回復する時間を指定します。

UDLD のデフォルト設定

表 *58 : UDLD* のデフォルト設定

機能	デフォルト設定
UDLD グローバル イネーブル ステート	グローバルにディセーブル
ポート別の UDLD イネーブル ステート（光ファイバ メディア用）	すべてのイーサネット光ファイバ ポート上でディセーブル

機能	デフォルト設定
ポート別の UDLD イネーブル ステート（ツイストペア（銅製）メディア用）	すべてのイーサネット 10/100 および 1000BASE-TX ポート上でディセーブル
UDLD アグレッシブ モード	ディセーブル

UDLD の設定方法

UDLD のグローバルなイネーブル化

アグレッシブ モードまたは通常モードで UDLD をイネーブルにし、device上のすべての光ファイバ ポートに設定可能なメッセージ タイマーを設定するには、次の手順に従います。

手順の概要

1. **configure terminal**
2. **udld** {**aggressive** | **enable** | **message time** *message-timer-interval*}
3. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**udld** {**aggressive**	**enable**	**message time** *message-timer-interval*} 例： `スイッチ(config)# udld enable` `message time 10`	UDLD モードの動作を指定します。 • **aggressive**：すべての光ファイバ ポートにおいて、アグレッシブ モードで UDLD をイネーブルにします。 • **enable**：device上のすべての光ファイバ ポート上で、UDLD を通常モードでイネーブルにします。UDLD はデフォルトでディセーブルです。 個々のインターフェイスの設定は、**udld enable** グローバル コンフィギュレーション コマンドの設定を上書きします。 • **message time** *message-timer-interval*：アドバタイズメント フェーズにあり、双方向リンクが検出されたポートでの UDLD プローブ メッセージ

	コマンドまたはアクション	目的
		の時間間隔を設定します。有効な範囲は 1 ～ 90 秒です。デフォルト値は 15 です。 （注） このコマンドが作用するのは、光ファイバ ポートだけです。他のポートタイプで UDLD をイネーブルにする場合は、**udld** インターフェイス コンフィギュレーション コマンドを使用します。 UDLD をディセーブルにするには、このコマンドの **no** 形式を使用します。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

インターフェイスでの UDLD のイネーブル化

アグレッシブ モードまたは通常モードをイネーブルにする、またはポート上で UDLD をディセーブルにするには、次の手順に従います。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **udld port** [**aggressive**]
4. **end**

手順の詳細

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

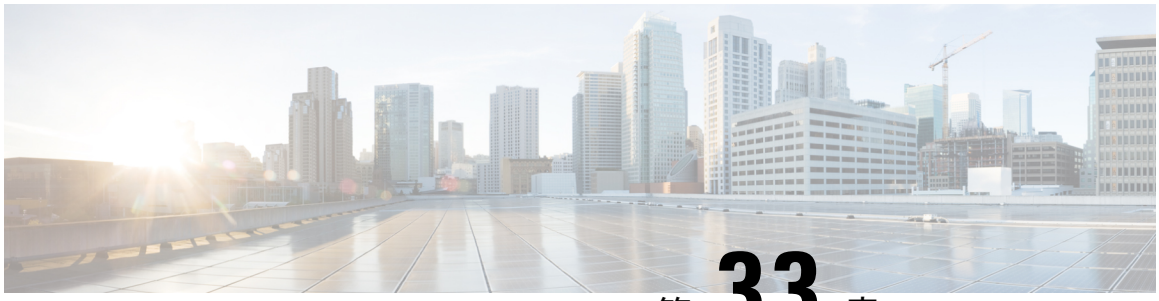
	コマンドまたはアクション	目的
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	UDLD 用にイネーブルにするポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**udld port** [**aggressive**] 例： `スイッチ(config-if)# udld port aggressive`	UDLD はデフォルトでディセーブルです。 • **udld port**：指定されたポート上で、UDLD を通常モードでイネーブルにします。 • **udld port aggressive**：（任意）指定されたインターフェイスにおいて、アグレッシブ モードで UDLD をイネーブルにします。 （注） 特定の光ファイバポート上でUDLDをディセーブルにする場合は、 **no udld port** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

UDLD のモニタおよびメンテナンス

コマンド	目的
show udld [*interface-id* | **neighbors**]	指定されたポートまたはすべてのポートの UDLD ステータスを表示します。

第 V 部

スタック マネージャおよびハイ アベイラビリティ



第 33 章

HSRP および VRRP の設定

HSRP の設定

この章では、ホットスタンバイ ルータ プロトコル（HSRP）を使用する方法について説明します。これによって、IP トラフィック ルーティングに冗長性を提供し、個々のルータのアベイラビリティに依存しないルーティングを実現します。

レイヤ 2 モードの HSRP のバージョンを使用すると、クラスタ コマンド スイッチが故障した場合、クラスタ管理を引き継ぐ冗長コマンド スイッチを設定することもできます。

（注） HSRP および VRRP 機能は Cisco Catalyst 3560-CXス イッチでのみサポートされます。

HSRP の設定に関する情報

HSRP の概要

HSRP は、デフォルト ゲートウェイ IP アドレスが設定された IEEE 802 LAN 上の IP ホストにファーストホップ冗長性を確保することでネットワークのアベイラビリティを高めるシスコの標準方式です。HSRP を使用すると、特定のルータのアベイラビリティに依存せずIP トラフィックをルーティングできます。また、一連のルータ インターフェイスを組み合わせることで、1 台の仮想ルータ、または LAN 上のホストへのデフォルト ゲートウェイのように機能させることができます。ネットワークまたはセグメント上に HSRP を設定すると、仮想 MAC（メディア アクセス コントロール）アドレス、および設定されたルータ グループ間で共有される IP アドレスを使用できるようになりHSRP が設定された複数のルータは、仮想ルータの MAC アドレスおよび IP ネットワーク アドレスを使用できるようになります。仮想ルータは、実際には存在しません。仮想ルータは、相互にバックアップ機能を提供するように設定されている複数のルータの共通のターゲットを表します。1 台のルータがアクティブなルータとして、もう 1 台のルータがスタンバイ ルータとして選択されます。スタンバイ ルータは、指定されたアク

ティブルータが故障した場合に、グループのMACアドレスおよびIPアドレスを制御するルータです。

（注） HSRP グループ内のルータには、ルーテッド ポート、スイッチ仮想インターフェイス（SVI）など、HSRP をサポートする任意のルータ インターフェイスを指定できます。

HSRPは、ネットワーク上のホストからのIPトラフィックに冗長性を提供することで、ネットワークのアベイラビリティを高めます。アクティブ ルータは、ルータ インターフェイスのグループ内でパケットのルーティングを実行するために選択されたルータです。スタンバイ ルータは、アクティブ ルータが故障した場合、または事前に設定した条件が満たされた場合に、ルーティング作業を引き継ぐルータです。

HSRPは、ホストがルータ ディスカバリ プロトコルをサポートしておらず、選択されたルータのリロードや電源故障時に新しいルータに切り替えることができない場合に有効です。HSRPをネットワーク セグメントに設定すると、HSRPは仮想MACアドレスとIPアドレスを1つずつ提供します。このアドレスは、HSRPが動作するルータ インターフェイス グループ内のルータ インターフェイス間で共有できます。プロトコルによってアクティブ ルータとして選択されたルータは、グループの MAC アドレス宛てのパケットを受信し、ルーティングします。n台のルータでHSRPが稼働している場合、n+1個のIPアドレスおよびMACアドレスが割り当てられます。

指定されたアクティブ ルータの故障を HSRP が検出すると、選択されているスタンバイ ルータがホットスタンバイ グループの MAC アドレスおよび IP アドレスの制御を引き継ぎます。この時点で新しいスタンバイ ルータも選択されます。HSRPが稼働しているデバイスは、マルチキャスト UDP ベースの hello パケットを送受信することにより、ルータ障害の検出、アクティブ ルータおよびスタンバイ ルータの指定を行います。インターフェイスに HSRP が設定されている場合、そのインターフェイスではインターネット制御メッセージプロトコル（ICMP）のリダイレクト メッセージが自動的にイネーブルになっています。

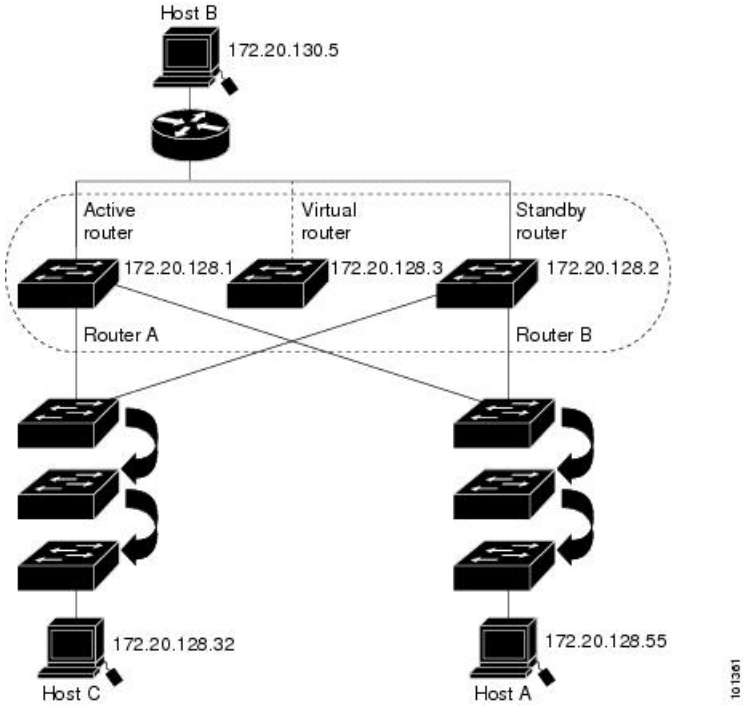
レイヤ 3 で動作するスイッチおよびスイッチ スタック間で複数のホット スタンバイ グループを設定すると、冗長ルータをさらに活用できます。

そのためには、インターフェイスに設定するホットスタンバイ コマンド グループごとにグループ番号を指定します。たとえば、スイッチ1のインターフェイスをアクティブルータ、スイッチ2のインターフェイスをスタンバイ ルータとして設定できます。また、スイッチ2の別のインターフェイスをアクティブ ルータ、スイッチ 1 の別のインターフェイスをスタンバイ ルータとして設定することもできます。

次の図に、HSRP 用に設定されたネットワークのセグメントを示します。各ルータには、仮想ルータの MAC アドレスおよび IP ネットワーク アドレスが設定されています。ルータ A の IP アドレスをネットワーク上のホストに設定する代わりに、デフォルトルータとして仮想ルータの IP アドレスを設定します。ホスト C からホスト B にパケットが送信される場合、ホスト C は仮想ルータの MAC アドレスにパケットを送信します。何らかの理由により、ルータ A がパケットの転送を停止すると、ルータ B が仮想 IP アドレスおよび仮想 MAC アドレスに応答してアクティブ ルータとなり、アクティブ ルータの作業を行います。ホスト C は引き続き仮想ルータの IP アドレスを使用し、ホスト B 宛のパケットをアドレッシングします。ルータ B は

そのパケットを受信し、ホスト B に送信します。ルータ B は HSRP の機能を使用し、ルータ A が動作を再開するまで、ホスト B のセグメント上のユーザーと通信する必要があるホスト C のセグメント上のユーザーに連続的にサービスを提供します。また、ホスト A セグメントとホスト B の間で、引き続き通常のパケット処理機能を実行します。

図 *56 : HSRP* の一般的な構成



HSRP のバージョン

以降のスイッチでサポートされている Hot Standby Router Protocol（HSRP）のバージョンは次のとおりです。

スイッチでは、次の HSRP バージョンがサポートされます。

- HSRPv1：HSRP のバージョン 1（デフォルトのバージョン）。次の機能があります。
  - HSRP グループ番号は 0 ～ 255 まで使用できます。
  - HSRPv1 は 224.0.0.2 のマルチキャスト アドレスを使用して hello パケットを送信しますが、これは Cisco Group Management Protocol（CGMP）の脱退処理と競合します。HSRPv1 と CGMP は相互に排他的なため、同時には使用できません。
- HSRPv2：HSRP のバージョン 2。このバージョンには次の機能があります。
  - HSRPv2 は 224.0.0.102 のマルチキャスト アドレスを使用して hello パケットを送信します。HSRPv2 と CGMP 脱退処理は相互に排他的ではありません。同時に使用できます。
  - HSRPv2 のパケット形式は、HSRPv1 とは異なります。

HSRPv1 を実行しているスイッチは、ルータの送信元 MAC アドレスが仮想 MAC アドレスのため、hello パケットを送信した物理的なルータを特定できません。

HSRPv2 のパケット形式は、HSRPv1 とは異なります。HSRPv2 パケットは、パケットを送信した物理ルータの MAC アドレスを格納できる 6 バイトの識別子フィールドを持った、Type Length Value（TLV）形式を使用します。

HSRPv1 を実行しているインターフェイスが HSRPv2 パケットを取得した場合、このタイプフィールドは無視されます。

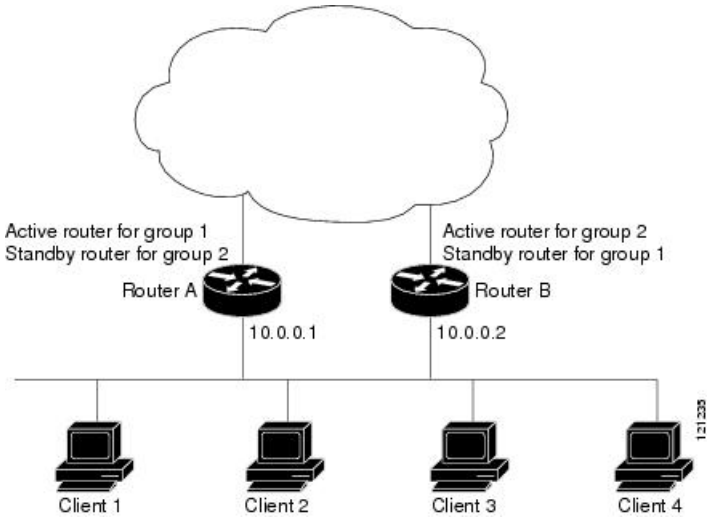
MHSRP

スイッチは、Multiple HSRP（MHSRP）をサポートします。MHSRP は HSRP の拡張版で、複数の HSRP グループ間でのロード シェアリングが可能です。ホスト ネットワークからサーバー ネットワークまで、ロードバランシングを実現して複数のスタンバイ グループ（およびパス）を使用するために、MHSRP を設定できます。

下の図では、半分のクライアントがルータ A に設定されており、もう半分はルータ B に設定されています。ルータ A およびルータ B の設定により、合計 2 つの HSRP グループが確立されています。グループ 1 では、ルータ A に最高のプライオリティが割り当てられているので、ルータ A がデフォルトのアクティブ ルータになり、ルータ B がスタンバイ ルータとなります。グループ 2 では、ルータ B に最も高いプライオリティが割り当てられているため、ルータ B がデフォルトのアクティブ ルータであり、ルータ A がスタンバイ ルータです。通常の運用では、2 つのルータが IP トラフィック負荷を分散します。いずれかのルータが使用できなくなると、もう一方のルータがアクティブになり、使用できないルータのパケット転送機能を引き継ぎます。

（注） MHSRP では、ルータに障害が発生して正常に戻った場合にプリエンプションによりロード シェアリングを復元するために、**standby preempt** インターフェイス コンフィギュレーション コマンドを HSRP インターフェイスで入力する必要があります。

図 *57 : MHSRP* ロード シェアリング



SSO HSRP

SSO HSRP は、冗長なルート プロセッサ（RP）を装備したデバイスがステートフル スイッチオーバー（SSO）冗長モード用に設定されているときの HSRP の動作を変更します。ある RP がアクティブで、もう一方の RP がスタンバイになっているとき、アクティブ RP に障害が発生すると、SSO は処理を引き継ぐスタンバイ RP をイネーブルにします。

この機能を使用すると、HSRP の SSO 情報がスタンバイ RP に同期されるため、HSRP 仮想 IP アドレスを使用して送信されるトラフィックをスイッチオーバー中も引き続き転送できるほか、データの損失やパスの変更も発生しません。さらに、HSRP アクティブ デバイスの両方の RP に障害が発生しても、スタンバイ状態の HSRP デバイスが HSRP アクティブ デバイスとして処理を引き継ぎます。

この機能は、動作の冗長モードが SSO に設定されている場合にデフォルトでイネーブルになっています。

HSRP の設定方法

HSRP のデフォルト設定

表 *59 : HSRP* のデフォルト設定

機能	デフォルト設定
HSRP バージョン	バージョン 1
HSRP グループ	未設定
スタンバイ グループ番号	0

機能	デフォルト設定
スタンバイ MAC アドレス	0000.0c07.acXX に指定されたシステム。XX は、HSRP グループ番号
スタンバイ プライオリティ	100
スタンバイ遅延	0（遅延なし）
スタンバイでのインターフェイス プライオリティの追跡	10
スタンバイ hello 時間	3 秒
スタンバイ ホールドタイム	10 秒

HSRP 設定時の注意事項

- HSRPv2 および HSRPv1 は相互に排他的です。HSRPv2 は、同じインターフェイス上で HSRPv1 と一緒には動作しません（その逆も同様）。
- 以下の手順では、次に示すレイヤ 3 インターフェイスの 1 つを指定する必要があります。
  - ルーテッドポート：インターフェイス コンフィギュレーション モードで **no switchport** コマンドを入力することにより、レイヤ 3 ポートとして設定された物理ポート。
  - SVI：グローバル コンフィギュレーション モードで **interface vlan** *vlan_id* を使用して作成された VLAN インターフェイス。デフォルトではレイヤ 3 インターフェイスです。
  - レイヤ 3 モードの Etherchannel ポートチャネル：グローバル コンフィギュレーション モードで **interface port-channel** *port-channel-number* を使用し、イーサネット インターフェイスをチャネルグループにバインドして作成されたポートチャネル論理インターフェイス。
- すべてのレイヤ 3 インターフェイスに IP アドレスを割り当てる必要があります。
- インターフェイスの HSRP バージョンを変更する場合、HSRP グループは新しい MAC アドレスを持つことになるため、リセットされます。

HSRP のイネーブル化

standby ip インターフェイス コンフィギュレーション コマンドは、設定されているインターフェイスで HSRP をアクティブにします。IP アドレスを指定した場合は、IP アドレスがホットスタンバイ グループの指定アドレスとして使用されます。IP アドレスを指定しなかった場合は、スタンバイ機能によってアドレスが学習されます。指定アドレスを使用し、LAN 上に

少なくとも 1 つのレイヤ 3 ポートを設定する必要があります。IP アドレスを設定すると、常に、現在使用されている別の指定アドレスが、設定した IP アドレスに変更されます。

standby ip コマンドがインターフェイス上で有効にされており、プロキシ ARP が有効な場合、インターフェイスのホットスタンバイ状態がアクティブになると、プロキシ ARP 要求に対する応答は、ホットスタンバイグループの MAC アドレスを使用して実行されます。インターフェイスが別のステートの場合、プロキシ ARP の応答は抑制されます。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **standby version** { **1** | **2** }
4. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
5. **end**
6. **show standby** [*interface-id* [*group*]]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch(config)# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Switch(config)# interface gigabitethernet1/0/1`	インターフェイス コンフィギュレーション モードを開始し、HSRP をイネーブルにするレイヤ 3 インターフェイスを入力します。
ステップ 3	**standby version** { **1** | **2** } 例： `Switch(config-if)# standby version 1`	（任意）インターフェイスに HSRP バージョンを設定します。 • 1：HSRPv1 を選択します。 • 2：HSRPv2 を選択します。 このコマンドを入力しない場合、またはキーワードを指定しない場合、インターフェイスはデフォルトの HSRP バージョンである HSRPv1 を実行します。
ステップ 4	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] 例： `Switch(config-if)# standby 1 ip`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成（またはイネーブルに）します。 • （任意）group-number：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場

	コマンドまたはアクション	目的
		合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）ip-address：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IP アドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 5	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります
ステップ 6	**show standby** [*interface-id* [*group*]] 例： `Switch # show standby`	スタンバイ グループの設定を確認します。
ステップ 7	**copy running-config startup-config** 例： `Switch# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

HSRP のプライオリティの設定

standby priority, **standby preempt**、および **standby track** インターフェイス コンフィギュレーション コマンドはいずれも、アクティブ ルータとスタンバイ ルータを検索するための特性、および新しいアクティブ ルータが処理を引き継いだ場合の動作を設定するために使用できます。

HSRP プライオリティを設定する場合の注意事項は、次のとおりです。

- プライオリティを割り当てておくと、アクティブ ルータおよびスタンバイ ルータを選択できます。プリエンプションがイネーブルの場合は、プライオリティが最高のルータがアクティブルータになります。プライオリティが等しい場合は、現在アクティブなルータに変更はありません。
- 最大の値（1 ～ 255）が、最高のプライオリティ（アクティブ ルータになる確率が最も高い）を表します。
- プライオリティ、プリエンプト、またはその両方を設定するときは、少なくとも 1 つのキーワード（**priority**、**preempt**、または両方）を指定する必要があります。
- インターフェイスが **standby track** コマンドによって設定されている場合、ルータ上の別のインターフェイスがダウンすると、デバイスのプライオリティが動的に変更されることもあります。
- **standby track** インターフェイス コンフィギュレーション コマンドを実行すると、ルータのホットスタンバイ プライオリティとインターフェイスのアベイラビリティが関連付けられます。この機能は、HSRP 用に設定されていないインターフェイスを追跡する場合に有効です。追跡対象のインターフェイスが故障すると、トラッキングが設定されているデバイスのホットスタンバイ プライオリティが 10 減少します。追跡対象でないインターフェイスの場合は、そのステートが変わっても、設定済みデバイスのホットスタンバイプライオリティは変わりません。ホットスタンバイ用に設定されたインターフェイスごとに、追跡するインターフェイスのリストを個別に設定できます。
- **standby track** *interface-priority* インターフェイス コンフィギュレーション コマンドを実行すると、追跡対象のインターフェイスがダウンした場合のホットスタンバイ優先順位の減少幅を指定できます。インターフェイスが稼働状態に戻ると、プライオリティは同じ分だけ増加します。
- *interface-priority* 値が設定されている場合に、複数の追跡対象インターフェイスがダウンすると、設定済みプライオリティの減少幅が累積されます。プライオリティ値が設定されていない追跡対象インターフェイスが故障した場合、デフォルトの減少幅は 10 です。この値は累積されません。
- インターフェイスに対してルーティングを最初にイネーブルにした時点で、完全なルーティングテーブルは存在しません。このインターフェイスがプリエンプトに設定されている場合はアクティブルータになりますが、十分なルーティング処理はできません。この問題を解決するには、ルータがルーティングテーブルを更新できるように遅延時間を設定します。

インターフェイスに HSRP プライオリティ特性を設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **standby** ［*group-number*］ **priority***priority*
4. **standby**［*group-number*］**preempt**［**delay**［**minimum***seconds*］［**reload***seconds*］［**sync***seconds*］］
5. **standby** ［*group-number*］ **track** *type number*［*interface-priority*］
6. **end**
7. **show running-config**

8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch # configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Switch(config)# interface gigabitethernet1/0/1`	インターフェイス コンフィギュレーション モードを開始し、プライオリティを設定する HSRP インターフェイスを入力します。
ステップ 3	**standby** ［*group-number*］ **priority***priority* 例： `Switch(config-if)# standby 120 priority 50`	アクティブ ルータを選択するときに使用される **priority** 値を設定します。指定できる範囲は 1 ～ 255 です。デフォルト プライオリティは 100 です。最大の値が、最高のプライオリティを表します。 • （任意）group-number：コマンドが適用されるグループ番号です。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 4	**standby**［*group-number*］**preempt**［**delay**［**minimum***seconds*］［**reload***seconds*］［**sync***seconds*］］ 例： `Switch(config-if)# standby 1 preempt delay 300`	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ ルータよりも高い場合は、アクティブ ルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）

	コマンドまたはアクション	目的
		で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 5	**standby** ［*group-number*］ **track** *type number* ［*interface-priority*］ 例： `Switch(config-if)# standby track interface gigabitethernet1/1/1`	他のインターフェイスを追跡するようにインターフェイスを設定します。この設定により、他のインターフェイスの 1 つがダウンした場合は、そのデバイスのホットスタンバイ プライオリティが減少します。 • （任意）group-number：コマンドが適用されるグループ番号です。 • type：追跡対象のインターフェイスタイプを（インターフェイス番号とともに）入力します。 • number：追跡対象のインターフェイス番号を（インターフェイスタイプとともに）入力します。 • （任意）interface-priority：インターフェイスがダウンした場合、または稼働状態に戻った場合に、ルータのホットスタンバイ プライオリティを減少または増加させる幅を入力します。デフォルト値は 10 です。
ステップ 6	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config**	スタンバイ グループの設定を確認します。
ステップ 8	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MHSRP の設定

MHSRP およびロード バランシングをイネーブルにするには、MHSRP の項の *MHSRP* ロード シェアリングの図に示したように、グループのアクティブ ルータとして 2 つのルータを設定し、スタンバイ ルータとして仮想ルータを設定します。ルータに障害が発生して正常に戻った場合、プリエンプションを発生させてロードバランシングを復元するために、**standby preempt** インターフェイス コンフィギュレーション コマンドをそれぞれの HSRP インターフェイスで入力する必要があります。

ルータ A はグループ 1 のアクティブ ルータとして、ルータ B はグループ 2 のアクティブ ルータとして設定されています。ルータ A の HSRP インターフェイスの IP アドレスは 10.0.0.1、グ

ループ 1 のスタンバイ プライオリティは 110（デフォルトは 100）です。ルータ B の HSRP インターフェイスの IP アドレスは 10.0.0.2、グループ 2 のスタンバイ プライオリティは 110 です。

グループ 1 は仮想 IP アドレス 10.0.0.3 を使用し、グループ 2 は仮想 IP アドレス 10.0.0.4 を使用します。

ルータ A の設定

手順の概要

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch # configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type number* 例： `Switch (config)# interface gigabitethernet1/0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**no switchport** 例： `Switch (config)# no switchport`	レイヤ 2 モードになっているインターフェイスを、レイヤ 3 設定用にレイヤ 3 モードに切り替えます。
ステップ 4	**ip address** *ip-address mask* 例： `Switch (config-if)# ip address 10.0.0.1 255.255.255.0`	インターフェイスの IP アドレスを指定します。

	コマンドまたはアクション	目的
ステップ 5	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] 例： `Switch (config-if)# standby 1 ip 10.0.0.3`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IP アドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 6	**standby** ［*group-number*］ **priority** *priority* 例： `Switch(config-if)# standby 1 priority 110`	アクティブ ルータを選択するときに使用される **priority** 値を設定します。指定できる範囲は 1 ～ 255 です。デフォルト プライオリティは 100 です。最大の値が、最高のプライオリティを表します。 • （任意）*group-number*：コマンドが適用されるグループ番号です。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 7	**standby** ［*group-number*］ **preempt** ［**delay** ［**minimum** *seconds*］ ［**reload** *seconds*］ ［**sync** *seconds*］］ 例： `Switch(config-if)# standby 1 preempt delay 300`	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ ルータよりも高い場合は、アクティブ ルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定でき

	コマンドまたはアクション	目的
		る範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 8	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] 例： `Switch (config-if)# standby 2 ip 10.0.0.4`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IP アドレスが比較され、IP アドレスが大きいルータがアクティブ ルー

	コマンドまたはアクション	目的
		タ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 9	**standby** ［*group-number*］ **preempt** ［**delay** ［**minimum** *seconds*］ ［**reload** *seconds*］ ［**sync** *seconds*］］ 例： `Switch(config-if)# standby 2 preempt delay 300`	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ ルータよりも高い場合は、アクティブ ルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 10	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 11	**show running-config**	スタンバイ グループの設定を確認します。
ステップ 12	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ルータ B の設定

手順の概要

1. **configure terminal**

2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch # configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *type number* 例： `Switch (config)# interface gigabitethernet1/0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**no switchport** 例： `Switch (config)# no switchport`	レイヤ 2 モードになっているインターフェイスを、レイヤ 3 設定用にレイヤ 3 モードに切り替えます。
ステップ 4	**ip address** *ip-address mask* 例： `Switch (config-if)# 10.0.0.2 255.255.255.0`	インターフェイスの IP アドレスを指定します。
ステップ 5	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] 例： `Switch (config-if)# standby 1 ip 10.0.0.3`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定しま

	コマンドまたはアクション	目的
		す。少なくとも1つのインターフェイスに対して仮想IPアドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IPアドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 6	**standby** ［*group-number*］ **priority** *priority* 例： `Switch(config-if)# standby 1 priority 110`	アクティブ ルータを選択するときに使用される **priority** 値を設定します。指定できる範囲は 1 ～ 255 です。デフォルト プライオリティは 100 です。最大の値が、最高のプライオリティを表します。 • （任意）*group-number*：コマンドが適用されるグループ番号です。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 7	**standby** ［*group-number*］ **preempt** ［**delay** ［**minimum** *seconds*］ ［**reload** *seconds*］ ［**sync** *seconds*］］ 例： `Switch(config-if)# standby 1 preempt delay 300`	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ ルータよりも高い場合は、アクティブ ルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ロー

	コマンドまたはアクション	目的
		カル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 8	**standby** ［*group-number*］ **ip** ［*ip-address* ［**secondary**］］ 例： `Switch (config-if)# standby 2 ip 10.0.0.4`	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成します。 • （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を指定します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。ルータがセカンダリ ルータとスタンバイ ルータのいずれにも指定されず、かつプライオリティも設定されていない場合は、プライマリ IP アドレスが比較され、IP アドレスが大きいルータがアクティブ ルータ、IP アドレスが 2 番めに大きいルータがスタンバイ ルータになります。
ステップ 9	**standby** ［*group-number*］ **preempt** ［**delay** ［**minimum** *seconds*］ ［**reload** *seconds*］ ［**sync** *seconds*］］ 例： `Switch(config-if)# standby 2 preempt delay 300`	ルータを **preempt** に設定し、ローカル ルータのプライオリティがアクティブ ルータよりも高い場合は、アクティブ ルータとなります。 • （任意）group-number：コマンドが適用されるグループ番号です。 • （任意）**delay minimum**：ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォル

	コマンドまたはアクション	目的
		トは 0 です（引き継ぐ前の遅延はありません）。 • （任意）**delay reload**：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600（1 時間）で、デフォルトは 0 です（リロードの後、引き継ぐ前の遅延はありません）。 • （任意）**delay sync**：IP 冗長性クライアントが応答できるように（ok または wait 応答）、ローカル ルータがアクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。指定できる範囲は 0 ～ 3600 秒（1 時間）で、デフォルトは 0 です（引き継ぐ前の遅延はありません）。 デフォルト値に戻すには、このコマンドの **no** 形式を使用します。
ステップ 10	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 11	**show running-config**	スタンバイ グループの設定を確認します。
ステップ 12	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

HSRP 認証およびタイマーの設定

HSRP 認証ストリングを設定したり、hello タイム インターバルやホールドタイムを変更することもできます。

これらの属性を設定する場合の注意事項は次のとおりです。

- 認証ストリングはすべての HSRP メッセージで暗号化されずに送信されます。相互運用できるように、接続されたすべてのルータおよびアクセス サーバーに同じ認証ストリングを設定する必要があります。認証ストリングが一致しないと、HSRP によって設定された他のルータから、指定されたホットスタンバイ IP アドレスおよびタイマー値を学習できません。
- スタンバイ タイマー値が設定されていないルータまたはアクセス サーバーは、アクティブ ルータまたはスタンバイ ルータからタイマー値を学習できます。アクティブ ルータに設定されたタイマーは、常に他のタイマー設定よりも優先されます。
- ホットスタンバイ グループのすべてのルータで、同じタイマー値を使用する必要があります。通常、*holdtime* は *hellotime* の 3 倍以上です。

インターフェイスにHSRPの認証とタイマーを設定するには、特権EXECモードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **standby** [*group-number*] **authentication** *string*
4. **standby** [*group-number*] **timers** *hellotime holdtime*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch # configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Switch(config) # interface gigabitethernet1/0/1`	インターフェイス コンフィギュレーション モードを開始し、プライオリティを設定する HSRP インターフェイスを入力します。
ステップ 3	**standby** [*group-number*] **authentication** *string* 例： `Switch(config-if) # standby 1 authentication word`	（任意）**authentication** *string*：すべての HSRP メッセージで伝達されるストリングを入力します。認証ストリングには 8 文字までを指定できます。デフォルトのストリングは **cisco** です。 （任意）*group-number*：コマンドが適用されるグループ番号です。
ステップ 4	**standby** [*group-number*] **timers** *hellotime holdtime* 例： `Switch(config-if) # standby 1 timers 5 15`	（任意）helloパケット間隔、およびアクティブ ルータのダウンを他のルータが宣言するまでの時間を設定します。 • *group-number*：コマンドが適用されるグループ番号です。 • *hellotime*：連続するhelloパケット間のインターバルを秒単位で設定します。範囲は、1 ～ 255 秒です。デフォルトは 3 です。 • *holdtime*：ローカル ルータがリロードの後アクティブ ルータの役割を引き継ぐまでの時間を、指定された秒数だけ延期します。範囲は 0 ～ 3600 秒（1 時間）です。デフォルトは 0 です

	コマンドまたはアクション	目的
		（リロードの後、引き継ぐ前の遅延はありません）。
ステップ 5	**end** 例： `Switch(config-if) # end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config**	スタンバイ グループの設定を確認します。
ステップ 7	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ICMP リダイレクト メッセージの HSRP サポートのイネーブル化

HSRP が設定されたインターフェイスでは、ICMP リダイレクト メッセージが自動的にイネーブルになります。ICMP は、エラーをレポートするためのメッセージ パケットや IP 処理に関連する他の情報を提供する、ネットワーク層インターネット プロトコルです。ICMP には、ホストへのエラー パケットの方向付けや送信などの診断機能があります。この機能は、HSRP を介した発信 ICMP リダイレクト メッセージをフィルタリングします。HSRP では、ネクスト ホップ IP アドレスが HSRP 仮想 IP アドレスに変更される可能性があります。詳細については、『Cisco IOS IP Configuration Guide, Release 12.4』を参照してください。

HSRP グループおよびクラスタリングの設定

デバイスが HSRP スタンバイ ルーティングに参加し、クラスタリングがイネーブルの場合は、同じスタンバイ グループを使用して、コマンド スイッチの冗長性および HSRP の冗長性を確保できます。同じ HSRP スタンバイ グループをイネーブルにし、コマンド スイッチおよびルーティングの冗長性を確保するには、**cluster standby-group** *HSRP-group-name* [**routing-redundancy**] グローバル コンフィギュレーション コマンドを使用します。**routing-redundancy** キーワードを指定せずに同じ HSRP スタンバイ グループ名でクラスタを作成すると、そのグループに対する HSRP スタンバイ ルーティングはディセーブルになります。

HSRP のトラブルシューティング

次の表で説明されている状況のいずれかが発生した場合、以下のメッセージが表示されます。

```
%FHRP group not consistent with already configured groups on the switch stack - virtual
MAC reservation failed
```

表 *60 : HSRP* のトラブルシューティング

状況	アクション（Action）
32 個を超える HSRP グループ インスタンスを設定する。	最大 32 個のグループ インスタンスに設定されるように HSRP グループを削除します。

HSRP の確認

HSRP コンフィギュレーションの確認

HSRP 設定を表示するには、次の特権 EXEC モードで次のコマンドを使用します。

show standby ［*interface-id*［*group*］］［**brief**］［**detail**］

スイッチ全体、特定のインターフェイス、HSRP グループ、またはインターフェイスの HSRP グループに関する HSRP 情報を表示できます。HSRP 情報の概要または詳細のいずれを表示するかを指定することもできます。デフォルトの表示は **detail** です。多数の HSRP グループがある場合に、修飾子を指定しないで **show standby** コマンドを使用すると、正確に表示されないことがあります。

例

```
Switch #show standby
VLAN1 - Group 1
Local state is Standby, priority 105, may preempt
Hellotime 3 holdtime 10
Next hello sent in 00:00:02.182
Hot standby IP address is 172.20.128.3 configured
Active router is 172.20.128.1 expires in 00:00:09
Standby router is local
Standby virtual mac address is 0000.0c07.ac01
Name is bbb

VLAN1 - Group 100
Local state is Standby, priority 105, may preempt
Hellotime 3 holdtime 10
Next hello sent in 00:00:02.262
Hot standby IP address is 172.20.138.51 configured
Active router is 172.20.128.1 expires in 00:00:09
Active router is local
Standby router is unknown expired
Standby virtual mac address is 0000.0c07.ac64
Name is test
```

HSRP の設定例

HSRP のイネーブル化：例

次に、インターフェイスのグループ 1 で HSRP をアクティブにする例を示します。ホットスタンバイ グループで使用される IP アドレスは、HSRP を使用して学習されます。

（注） これは、HSRP をイネーブルにするために必要な最小限の手順です。その他の設定は任意です。

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
```

```
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# end
Switch # show standby
```

HSRP のプライオリティの設定：例

次に、ポートをアクティブにして、IP アドレスおよびプライオリティ 120（デフォルト値よりも高いプライオリティ）を設定して、アクティブ ルータになるまで 300 秒（5 分間）待機する例を示します。

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby ip 172.20.128.3
Switch(config-if)# standby priority 120 preempt delay 300
Switch(config-if)# end
Switch # show standby
```

MHSRP の設定：例

次に、*MHSRP* ロード シェアリングの図で示した MHSRP 設定をイネーブルにする例を示します。

ルータ A の設定

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.1 255.255.255.0
Switch(config-if)# standby ip 10.0.0.3
Switch(config-if)# standby 1 priority 110
Switch(config-if)#  standby 1 preempt
Switch(config-if)# standby 2 ip 10.0.0.4
Switch(config-if)#  standby 2 preempt
Switch(config-if)# end
```

ルータ B の設定

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.2 255.255.255.0
Switch(config-if)# standby ip 10.0.0.3
Switch(config-if)#  standby 1 preempt
Switch(config-if)# standby 2 ip 10.0.0.4
Switch(config-if)# standby 1 priority 110
Switch(config-if)#  standby 2 preempt
Switch(config-if)# end
```

HSRP 認証およびタイマーの設定：例

次に、グループ 1 のホットスタンバイ ルータを相互運用させるために必要な認証ストリングとして、word を設定する例を示します。

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 authentication word
Switch(config-if)# end
```

次に、hello パケット間隔が 5 秒、ルータがダウンしたと見なされるまでの時間が 15 秒となるように、スタンバイ グループ 1 のタイマーを設定する例を示します。

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# standby 1 timers 5 15
Switch(config-if)# end
```

HSRP グループおよびクラスタリングの設定：例

次に、スタンバイ グループ my_hsrp をクラスタにバインドし、同じ HSRP グループをイネーブルにしてコマンド スイッチおよびルータの冗長性に使用する例を示します。このコマンドを実行できるのは、コマンド スイッチに対してだけです。スタンバイ グループの名前または番号が存在しない場合、またはスイッチがクラスタ メンバー スイッチである場合は、エラー メッセージが表示されます。

```
Switch # configure terminal
Switch(config) # cluster standby-group my_hsrp routing-redundancy
Switch(config-if)# end
```

VRRP の概要

VRRP の設定

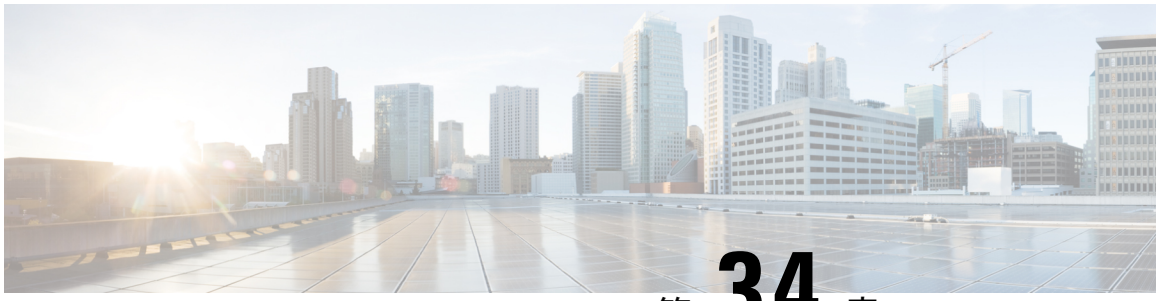
Virtual Router Redundancy Protocol（VRRP）は、ルータのグループを使用して単一の仮想ルータを形成し、冗長性を実現する選択プロトコルです。VRRP の設定では、1 つのルータが仮想ルータプライマリとして選択され、他のルータは障害発生時のバックアップとして機能します。LAN クライアントは、デフォルト ゲートウェイとして仮想ルータを使用して設定でき、マルチアクセス リンク上の複数のルータが同じ仮想 IP アドレスを使用できるようにします。ルータのグループを表す仮想ルータは、VRRP グループを形成します。

HSRP も VRRP も、同じ機能を実行します。デバイスまたはスタックに、IETF 標準 VRRP を設定するか、シスコのより強力な HSRP プロトコルを設定するかを選択できます。

VRRP の制約事項

- スイッチの VRRP 実装は、RFC 2787 で指定された MIB をサポートしません。

- スイッチの VRRP 実装は、テキストベースの認証だけをサポートします。



第 34 章

サービス レベル契約の設定

この章では、スイッチで Cisco IOS IP サービス レベル契約（SLA）を使用する方法について説明します。

特に明記しないかぎり、スイッチという用語はスタンドアロン スイッチまたはスイッチ スタックを意味します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

SLA の制約事項

ここでは、SLA の制約事項を示します。

次に示すのは、IP SLA ネットワーク パフォーマンス測定の制約事項です。

- deviceは、ゲートキーパー登録遅延動作測定を使用した Voice over IP（VoIP）サービス レベルはサポートしていません。

- Cisco IOS デバイスだけが宛先 IP SLA Responder の送信元になります。
- 他社製のデバイスに IP SLA Responder を設定することはできません。また、Cisco IOS IP SLA はこれらのデバイス固有のサービスに対してだけ動作パケットを送信できます。

SLA について

Cisco IOS IP サービス レベル契約（SLA）

Cisco IOS IP SLA はネットワークにデータを送信し、複数のネットワーク ロケーション間あるいは複数のネットワーク パス内のパフォーマンスを測定します。Cisco IOS IP SLA は、ネットワーク データおよび IP サービスをシミュレーションし、ネットワーク パフォーマンス情報をリアル タイムで収集します。Cisco IOS IP SLA は、Cisco IOS デバイス間のトラフィックまたは Cisco IOS デバイスからネットワーク アプリケーション サーバーのようなリモート IP デバイスへのトラフィックを生成し、分析します。さまざまな Cisco IOS IP SLA 動作で評価を実行し、トラブルシューティング、問題分析、ネットワーク トポロジの設計に使用します。

Cisco IOS IP SLA 動作に応じてシスコ デバイスのネットワーク パフォーマンス統計情報がモニタリングされ、コマンドライン インターフェイス（CLI）MIB および簡易ネットワーク管理プロトコル（SNMP）MIB に格納されます。IP SLA パケットには設定可能な IP レイヤおよびアプリケーション層のオプションがあります。たとえば、発信元および宛先 IP アドレス、ユーザー データグラム プロトコル（UDP）/TCP ポート番号、タイプ オブ サービス（ToS）バイト（DiffServ コードポイント（DSCP）および IP プレフィックス ビットを含む）、VPN ルーティング/転送インスタンス（VRF）、URL Web アドレスなどが設定できます。

Cisco IP SLA はレイヤ 2 転送に依存していないので、異なるネットワーク間にエンドツーエンド動作を設定してエンドユーザーが経験しそうなメトリックを最大限に反映させることができます。IP SLA は、次のパフォーマンス メトリックを収集して分析します。

- 遅延（往復および一方向）
- ジッター（方向性あり）
- パケット損失（方向性あり）
- パケット シーケンス（パケット順序）
- パス（ホップ単位）
- 接続（方向性あり）
- サーバーまたは Web サイトのダウンロード時間

Cisco IP SLA は SNMP によるアクセスが可能なので、Cisco Prime Internetwork Performance Monitor（IPM）やサードパーティ製パフォーマンス管理製品などのパフォーマンス モニタリング アプリケーションでも使用できます。

IP SLA を使用すると、次の利点が得られます。

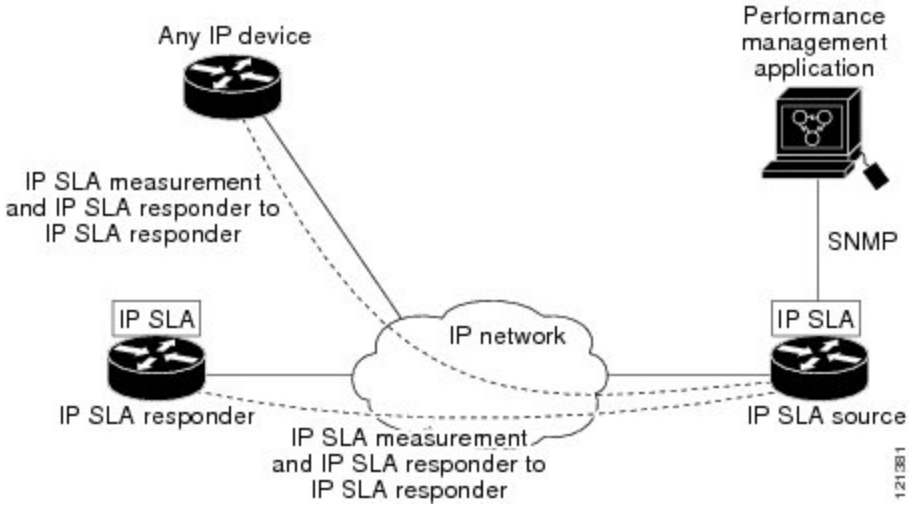
- SLA モニタリング、評価、検証。
- ネットワーク パフォーマンス モニタリング。
  - ネットワークのジッター、遅延、パケット損失の測定。
  - 連続的で信頼性のある予測可能な測定。
- IP サービス ネットワーク ヘルス アセスメントにより、既存の QoS が新しい IP サービスに適していることを確認できる。
- 端末間のネットワーク アベイラビリティをモニタリングして、ネットワーク リソースをあらかじめ検証し接続をテストできる（たとえば、ビジネス上の重要なデータを保存する NFS サーバーのネットワーク アベイラビリティをリモート サイトから確認できる）。
- 問題をすぐに認識し、トラブルシューティングにかかる時間を短縮できる一貫性のある信頼性の高い測定によるネットワーク動作のトラブルシューティング。
- マルチプロトコル ラベル スイッチング（MPLS）パフォーマンス モニターリングとネットワークの検証を行う（deviceが MPLS をサポートする場合）。

Cisco IOS IP SLA でのネットワーク パフォーマンスの測定

IP SLA を使用して、プローブを物理的に配置せずに、コア、分散、エッジといったネットワーク内の任意のエリア間のパフォーマンスを監視することができます。2 つのネットワーク デバイス間のネットワーク パフォーマンスは、生成トラフィックで測定します。

図 *58 : Cisco IOS IP SLA* 動作

次の図に、送信元デバイスが宛先デバイスに生成パケットを送信するときに IP SLA が開始される手順を示します。宛先デバイスがパケットを受信すると、IP SLA 動作の種類によって、送信元のタイム スタンプ情報に応じてパフォーマンス メトリックを算出します。IP SLA 動作は、特定のプロトコル（UDP など）を使用してネットワークの送信元から宛先へのネットワーク測定を行います。



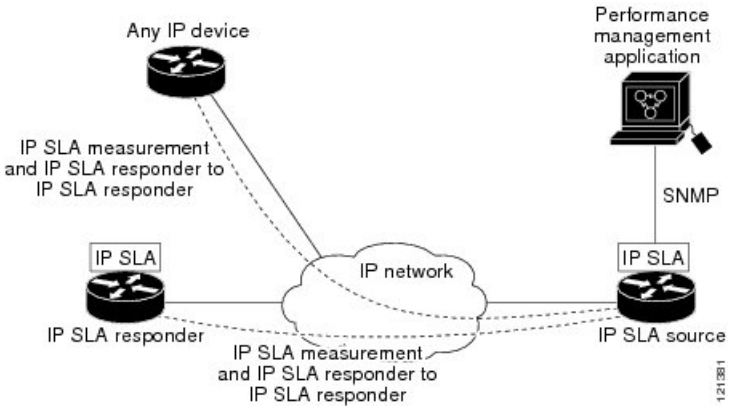
IP SLA レスポンダおよび IP SLA 制御プロトコル

IP SLA レスポンダは宛先 Cisco デバイスに組み込まれたコンポーネントで、システムが IP SLA 要求パケットを予想して応答します。Responderは専用プローブなしで正確な測定を行います。レスポンダは、受信および応答するポートが通知されるメカニズムを Cisco IOS IP SLA コントロール プロトコルを通じて実現します。

（注） IP SLA レスポンダはレスポンダ設定可能なdeviceである Cisco IOS レイヤ 2 にすることもできます。レスポンダは、IP SLA 機能を全面的にサポートする必要はありません。

次の図は、IP ネットワーク内での Cisco IOS IP SLA レスポンダの配置場所を示します。レスポンダは、IP SLA 動作から送信されたコントロール プロトコル メッセージを指定されたポートで受信します。コントロール メッセージを受信したら、指定された UDP または TCP ポートを指定された時間だけ有効にします。この間に、レスポンダは要求を受け付け、応答します。レスポンダは、IP SLA パケットに応答した後または指定の時間が経過したら ポートを無効にします。セキュリティの向上のために、コントロール メッセージでは MD5 認証が利用できます。

図 *59 : Cisco IOS IP SLA* 動作



すべての IP SLA 動作に対して宛先デバイスのレスポンダをイネーブルにする必要はありません。たとえば、宛先ルータが提供しているサービス（Telnet や HTTP など）は Responder では必要ありません。

IP SLA の応答時間の計算

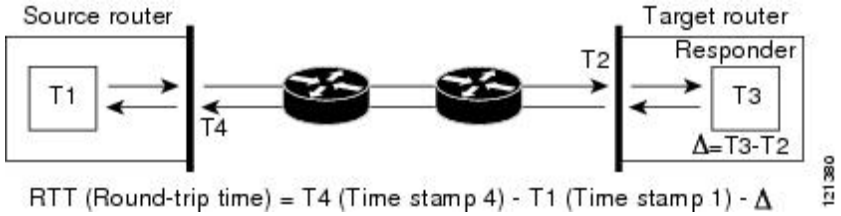
スイッチ、コントローラ、ルータは、他の高優先度プロセスがあるために、着信パケットの処理に数十ミリ秒かかることがあります。この遅延により応答時間が影響を受けます。テストパケットの応答が処理待ちのキューに入っていることもあるからです。この場合、応答時間は正しいネットワーク遅延を反映しません。IP SLA はソース デバイスとターゲット デバイス（レスポンダが使用されている場合）の処理遅延を最小化し、正しいラウンドトリップ時間（RTT）

を識別します。IP SLA テスト パケットは、タイム スタンプによって処理遅延を最小化します。

IP SLA レスポンダが有効の場合、パケットが割り込みレベルでインターフェイスに着信したときおよびパケットが出て行くときにターゲット デバイスでタイム スタンプを付け、処理時間は含めません。タイム スタンプはサブミリ秒単位で構成されます。

図 *60 : Cisco IOS IP SLA* レスポンダ タイム スタンプ

次の図に、レスポンダの動作を示します。RTT を算出するためのタイム スタンプが 4 つ付けられます。ターゲット ルータでレスポンダ機能がイネーブルの場合、タイム スタンプ 3（TS3）からタイム スタンプ 2（TS2）を引いてテスト パケットの処理にかかった時間を求め、デルタ（Δ）で表します。次に全体の RTT からこのデルタの値を引きます。IP SLA により、この方法はソース ルータにも適用されます。その場合、着信タイム スタンプ 4（TS4）が割り込みレベルで付けられ、より正確な結果を得ることができます。



この他にも、ターゲット デバイスに 2 つのタイム スタンプがあれば一方向遅延、ジッター、方向性を持つパケット損失がトラッキングできるという利点があります。大半のネットワーク動作は非同期なので、このような統計情報があるのは問題です。ただし一方向遅延測定を取り込むには、ソース ルータとターゲット ルータの両方にネットワーク タイム プロトコル（NTP）を設定し、両方のルータを同じくロック ソースに同期させる必要があります。一方向ジッター測定にはクロック同期は不要です。

IP SLA 動作のスケジューリング

IP SLA 動作を設定する場合、統計情報の取り込みとエラー情報の収集から開始するように動作をスケジューリングする必要があります。スケジューリングは、すぐに動作を開始する、または特定の月、日、時刻に開始するように設定できます。また、*pending* オプションを使用して、あとで動作を開始するように設定することもできます。pending オプションは動作の内部状態に関するもので、SNMP で表示できます。トリガーを待機する反応（しきい値）動作の場合も pending オプションを使用します。1 度に 1 つの IP SLA 動作をスケジューリングしたり、グループの動作をスケジューリングすることもできます。

Cisco IOS CLI または CISCO RTTMON-MIB で 1 つのコマンドを使用して、複数の IP SLA 動作をスケジューリングできます。等間隔で動作を実行するようにスケジューリングすると、IP SLA モニタリング トラフィックの数を制御できます。IP SLA 動作をこのように分散させるとCPU 使用率を最小限に抑え、ネットワーク スケーラビリティを向上させることができます。

IP SLA 複数動作のスケジューリング機能の詳細については、『*Cisco IOS IP SLA Configuration Guide*』の「IP SLAs—Multiple Operation Scheduling」の章を参照してください。

IP SLA 動作のしきい値のモニタリング

サービス レベル契約モニタリングを正しくサポートするには、違反が発生した場合にすぐに通知されるメカニズムにする必要があります。IP SLA は次のような場合にイベントによってトリガーされる SNMP トラップを送信できます。

- 接続の損失
- タイムアウト
- RTT しきい値
- 平均ジッターしきい値
- 一方向パケット損失
- 一方向ジッター
- 一方向平均オピニオン評点（MOS）
- 一方向遅延

IP SLA しきい値違反が発生した場合も、あとで分析するために別の IP SLA 動作がトリガーされます。たとえば、回数を増やしたり、Internet Control Message Protocol（ICMP）パス エコーや ICMP パス ジッター動作を開始してトラブルシューティングを行うことができます。

ICMP エコー

ICMP エコー動作は、シスコ デバイスと IP を使用するその他のデバイス間のエンドツーエンド応答時間を測定します。応答時間は、ICMP エコー要求メッセージを宛先に送信し、ICMP エコー応答を受信するのにかかる時間を測定して算出されます。多くのお客様は、IP SLA ICMP ベース動作、社内 ping テスト、またはこの応答所要時間を測定するために ping ベース専用プローブを使用します。IP SLA ICMP エコー動作は、ICMP ping テストと同じ仕様に準拠しており、どちらの方法でも同じ応答所要時間になります。

UDP ジッター

ジッターとは、パケット間遅延の差異を説明する簡単な用語です。複数のパケットが送信元から宛先まで 10 ミリ秒の間隔で継続的に送信される場合、宛先は 10 ミリ秒間隔で受信します（ネットワークが正常に動作している場合）。しかし、ネットワークに遅延がある場合（キューイングや代替ルートを通じた到着など）、パケットの着信の間隔が 10 ミリ秒を超える場合や 10 ミリ秒未満になる場合があります。正のジッター値は、パケットが 10 ミリ秒を超える間隔で到着することを示します。負のジッター値は、パケットが 10 ミリ秒未満の間隔で到着することを示します。パケットの到着が 12 ミリ秒間隔の場合、正のジッター値は 2 ミリ秒です。8 ミリ秒間隔で到着する場合、負のジッター値は 2 ミリ秒です。遅延による影響を受けやすいネットワークの場合、正のジッターは望ましくありません。ジッター値 0 が理想的です。

ジッターのモニタリング以外にも、IP SLA UDP ジッター動作を多目的データ収集動作に使用できます。IP SLA によって生成されるパケットは、データを送受信するパケットを含めて、

送信元および動作ターゲットからシーケンス情報とタイム スタンプを伝送します。このデータに基づいて、UDP ジッター動作は次を測定します。

- 方向別ジッター（送信元から宛先へ、宛先から送信元へ）
- 方向別パケット損失
- 方向別遅延（一方向遅延）
- ラウンドトリップ遅延（平均 RTT）

データを送受信するパスが異なる場合もあるので（非同期）、方向別データを使用すればネットワークで発生している輻輳や他の問題の場所を簡単に突き止めることができます。

UDP ジッター動作では合成（シミュレーション）UDP トラフィックを生成し、送信元ルータからターゲット ルータに多数の UDP パケットを送信します。その際の各パケットのサイズ、パケット同士の間隔、送信間隔は決められています。デフォルトでは、10 バイトのペイロード サイズのパケット フレームを 10 ミリ秒で 10 個生成し、60 秒間隔で送信します。これらのパラメータは、提供する IP サービスを最適にシミュレートするように設定できます。

一方向遅延を正確に測定する場合、（NTP によって提供される）送信元デバイスとターゲット デバイス間のクロック同期が必要です。一方向ジッターおよびパケット損失を測定する場合は、クロック同期は不要です。送信元デバイスとターゲットデバイスの間でクロックが同期していない場合、一方向ジッターとパケット損失のデータは戻されますが、UDP ジッター動作による一方向遅延測定は 0 の値が戻ります。

IP SLA 動作の設定方法

ここでは、利用可能なすべての動作の設定情報について説明されているわけではありません。設定情報の詳細については『*Cisco IOS IP SLAs Configuration Guide*』を参照してください。ここでは、応答側の設定、UDP ジッター動作の設定（応答側が必要）、ICMP エコー動作の設定（応答側が不要）などの動作例を説明します。他の動作の設定の詳細については、『*Cisco IOS IP SLAs Configuration Guide*』を参照してください。

デフォルト設定

IP SLA 動作は設定されていません。

設定時の注意事項

IP SLA のコマンドについては、『*Cisco IOS IP SLA Command Reference, Release 12.4T*』のコマンド リファレンスを参照してください。

説明と設定手順の詳細については、『*Cisco IOS IP SLAs Configuration Guide, Release 12.4TL*』を参照してください。

ガイドに記載されている IP SLA コマンドまたは動作の中にはdeviceでサポートされないものもあります。deviceでは、UDP ジッター、UDP エコー、HTTP、TCP 接続、ICMP エコー、ICMP パス エコー、ICMP パス ジッター、FTP、DNS、DHCP を使用する IP サービス レベル分析がサポートされます。また、複数動作スケジューリングおよび事前に設定されたしきい値のモニタリングもサポートされます。ゲートキーパー登録遅延動作測定を使用した Voice over IP（VoIP）サービス レベルはサポートしていません。

IP SLA アプリケーションを設定する前に、**show ip sla application** 特権 EXEC コマンドを使用してソフトウェアイメージで動作タイプがサポートされていることを確認してください。コマンド出力例は次のとおりです。

```
スイッチ# show ip sla application

        IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:
        icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http
        dns, udpJitter, dhcp, ftp, udpApp, wspApp

Supported Features:
        IPSLAs Event Publisher

IP SLAs low memory water mark: 33299323
Estimated system max number of entries: 24389

Estimated number of configurable operations: 24389
Number of Entries configured  : 0
Number of active Entries      : 0
Number of pending Entries     : 0
Number of inactive Entries    : 0
Time of last change in whole IP SLAs: *13:04:37.668 UTC Wed Dec 19 2012
```

IP SLA レスポンダの設定

IP SLA レスポンダは、Cisco IOS ソフトウェアベース デバイスだけで利用可能です。これには、IP SLA 機能をフルにサポートしていない一部のレイヤ 2 devicesも含まれます。

ターゲット デバイス（動作ターゲット）上の IP SLA 応答側を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number* 例： スイッチ(config)# **ip sla responder udp-echo 172.29.139.134 5000**	deviceを IP SLA レスポンダとして設定します。 キーワードの意味は次のとおりです。 • **tcp-connect**：Responder の TCP 接続動作をイネーブルにします。 • **udp-echo**：レスポンダのユーザー データグラム プロトコル（UDP）エコー動作またはジッター動作をイネーブルにします。 • **ipaddress** *ip-address*：宛先 IP アドレスを入力します。 • **port** *port-number*：宛先ポート番号を入力します。 （注） IP アドレスとポート番号は、IP SLA 動作のソース デバイスに設定した IP アドレスおよびポート番号と一致している必要があります。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。

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

IP SLA ネットワーク パフォーマンス測定の実装

device上でIP SLA ネットワーク パフォーマンス測定を実施するには、次の手順を実行します。

始める前に

show ip sla application 特権 EXEC コマンドを使用して、ソフトウェアイメージで目的の動作タイプがサポートされていることを確認してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla** *operation-number*
4. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
5. **frequency** *seconds*
6. **threshold** *milliseconds*
7. **exit**
8. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**ip sla** *operation-number* 例： スイッチ(config)# **ip sla 10**	IP SLA 動作を作成し、IP SLA コンフィギュレーション モードを開始します。
ステップ 4	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*] 例： スイッチ(config-ip-sla)# **udp-jitter 172.29.139.134 5000**	IP SLA 動作を目的の動作タイプとして設定して（例では UDP ジッター動作が使用されています）、そのコンフィギュレーションモードを開始します（例では UDP ジッター コンフィギュレーション モードが使用されています）。 • *destination-ip-address* | *destination-hostname*：宛先 IP アドレスまたはホスト名を指定します。 • *destination-port*：宛先ポート番号を 1 ～ 65535 の範囲で指定します。 • （任意）**source-ip** {*ip-address* | *hostname*}：送信元 IP アドレスまたはホスト名を指定します。送信元 IP アドレスまたはホスト名が指定されていない場合、IP SLA では、宛先に最も近い IP アドレスが選択されます。 • （任意）**source-port** *port-number*：送信元ポート番号を 1 ～ 65535 の範囲で指定します。ポート番号を指定しない場合、IP SLA は利用可能なポートを選択します。 • （任意）**control**：IP SLA 制御メッセージの IP SLA レスポンダへの送信を有効または無効にします。デフォルトでは、IP SLA 制御メッセージは宛先デバイスに送信され、IP SLA レスポンダとの接続が確立されます。 • （任意）**num-packets** *number-of-packets*：生成するパケット数を入力します。指定できる範囲は 1 ～ 6000 です。デフォルトは 10 です。 • （任意）**interval** *inter-packet-interval*：パケットの送信間隔をミリ秒で入力します。指定できる範囲は 1 ～ 6000 です。デフォルトは 20 ミリ秒です。

	コマンドまたはアクション	目的
ステップ 5	**frequency** *seconds* 例： `スイッチ(config-ip-sla-jitter)# frequency 45`	（任意）SLA 動作のオプションを設定します。次の例では、指定された IP SLA 動作が繰り返されるレートを設定します。指定できる範囲は 1 ～ 604800 秒で、デフォルトは 60 秒です。
ステップ 6	**threshold** *milliseconds* 例： `スイッチ(config-ip-sla-jitter)# threshold 200`	（任意）しきい値条件を設定します。次の例では、指定された IP SLA 動作のしきい値が 200 に設定されます。有効な範囲は 0 ～ 60000 ミリ秒です。
ステップ 7	**exit** 例： `スイッチ(config-ip-sla-jitter)# exit`	SLA 動作コンフィギュレーション モード（この例では UDP ジッター コンフィギュレーション モード）を終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 8	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] 例： `スイッチ(config)# ip sla schedule 10 start-time now life forever`	個々の IP SLA 動作のスケジューリング パラメータを設定します。 • *operation-number*：RTR エントリ番号を入力します。 • （任意）**life**：動作の実行を無制限（**forever**）に指定するか、特定の秒数（*seconds*）を指定します。有効な範囲は 0 ～ 2147483647 です。デフォルトは 3600 秒（1 時間）です。 • （任意）**start-time**：情報の収集を開始する時刻を入力します。 特定の時刻に開始する場合は、時、分、秒（24 時間表記）、月日を入力します。月を入力しない場合、当月がデフォルト設定です。 **pending** と入力すれば、開始時刻を指定するまでは情報を収集しません。 **now** と入力すれば、ただちに動作を開始します。 **after** *hh:mm:ss* と入力すれば、指定した時刻の経過後に動作を開始します。 • （任意）**ageout** *seconds*：情報を収集していないとき、メモリの動作を保存する秒数を指定します。指定できる範囲は 0 ～ 2073600 秒です。

	コマンドまたはアクション	目的
		デフォルトは 0 秒（いつまでも保存する）です。 • （任意）**recurring**：毎日、動作を自動的に実行します。
ステップ 9	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

UDP ジッター コンフィギュレーション

次に、UDP ジッター IP SLA 動作の設定例を示します。

```
スイッチ(config)# ip sla 10
スイッチ(config-ip-sla)# udp-jitter 172.29.139.134 5000
スイッチ(config-ip-sla-jitter)# frequency 30
スイッチ(config-ip-sla-jitter)# exit
スイッチ(config)# ip sla schedule 5 start-time now life forever
スイッチ(config)# end
スイッチ# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 10.0.0.10/10.0.0.1
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
```

```
Control Packets: enabled
Schedule:
    Operation frequency (seconds): 30
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
Enhanced History:
```

UDP ジッター動作を使用した IP サービス レベルの分析

送信元デバイス上の UDP ジッター作を設定するには、次の手順を実行します。

始める前に

送信元デバイス上で UDP ジッター動作を設定するには、ターゲット デバイス（動作ターゲット）で、IP SLA レスポンダをイネーブルにする必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla** *operation-number*
4. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
5. **frequency** *seconds*
6. **exit**
7. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sla** *operation-number* 例： `Device(config)# ip sla 10`	IP SLA 動作を作成し、IP SLA コンフィギュレーション モードを開始します。
ステップ 4	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*] 例： `Device(config-ip-sla)# udp-jitter 172.29.139.134 5000 source-ip 172.29.139.140 source-port 4000`	IP SLA 動作を UDP ジッター動作として設定し、UDP ジッター コンフィギュレーション モードを開始します。 • *destination-ip-address* | *destination-hostname*：宛先 IP アドレスまたはホスト名を指定します。 • *destination-port*：宛先ポート番号を 1 ～ 65535 の範囲で指定します。 • （任意）**source-ip** {*ip-address* | *hostname*}：送信元 IP アドレスまたはホスト名を指定します。送信元 IP アドレスまたはホスト名が指定されていない場合、IP SLA では、宛先に最も近い IP アドレスが選択されます。 • （任意）**source-port** *port-number*：送信元ポート番号を 1 ～ 65535 の範囲で指定します。ポート番号を指定しない場合、IP SLA は利用可能なポートを選択します。 （注） **udp-jitter** コマンドで送信元ポートが設定されていない場合、UDP は制御パケット用のポートをランダムに選択します。UDP が予約済みポート 1967 を選択した場合、IP SLA レスポンダによる CPU 使用率が高くなる可能性があります。 • （任意）**control**：IP SLA 制御メッセージの IP SLA レスポンダへの送信を有効または無効にします。デフォルトでは、IP SLA 制御メッセージは宛先デバイスに送信され、IP SLA レスポンダとの接続が確立されます。

	コマンドまたはアクション	目的
		• （任意）**num-packets** *number-of-packets*：生成するパケット数を入力します。指定できる範囲は 1 ～ 6000 です。デフォルトは 10 です。 • （任意）**interval** *inter-packet-interval*：パケットの送信間隔をミリ秒単位で入力します。指定できる範囲は 1 ～ 6000 です。デフォルトは 20 ミリ秒です。
ステップ 5	**frequency** *seconds* 例： `Device(config-ip-sla-jitter)# frequency 45`	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。指定できる範囲は 1 ～ 604800 秒で、デフォルトは 60 秒です。
ステップ 6	**exit** 例： `Device(config-ip-sla-jitter)# exit`	UDP ジッター コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 7	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] 例： `Device(config)# ip sla schedule 10 start-time now life forever`	個々の IP SLA 動作のスケジューリング パラメータを設定します。 • *operation-number*：RTR エントリ番号を入力します。 • （任意）**life**：動作の実行を無制限（**forever**）に設定するか、特定の秒数（*seconds*）を設定します。有効な範囲は 0 ～ 2147483647 です。デフォルトは 3600 秒（1 時間）です。 • （任意）**start-time**：情報の収集を開始する時刻を入力します。 特定の時刻に開始する場合は、時、分、秒（24 時間表記）、月日を入力します。月を入力しない場合、当月がデフォルト設定です。 **pending** と入力すれば、開始時刻を指定するまでは情報を収集しません。 **now** と入力すれば、ただちに動作を開始します。 **after** *hh:mm:ss* と入力すれば、指定した時刻の経過後に動作を開始します。

	コマンドまたはアクション	目的
		• （任意）**ageout** *seconds*：情報を収集していないときに、メモリの動作を保存する秒数を指定します。指定できる範囲は 0 ～ 2073600 秒です。デフォルトは0秒（いつまでも保存する）です。 • （任意）**recurring**：毎日、動作を自動的に実行するように設定します。
ステップ 8	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 9	**show running-config** 例： `Device# show running-config`	入力を確認します。
ステップ 10	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

UDP ジッター IP SLA 動作の設定

次に、UDP ジッター IP SLA 動作の設定例を示します。

```
Device(config)# ip sla 10
Device(config-ip-sla)# udp-jitter 172.29.139.134 5000 source-ip 172.29.139.140 source-port
 4000
Device(config-ip-sla-jitter)# frequency 30
Device(config-ip-sla-jitter)# exit
Device(config)# ip sla schedule 10 start-time now life forever
Device(config)# end
Device# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 10.0.0.10/10.0.0.1
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
```

```
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
    Operation frequency (seconds): 30
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
Enhanced History:
```

ICMP エコー動作を使用した IP サービス レベルの分析

送信元デバイス上の ICMP エコー動作を設定するには、次の手順を実行します。

始める前に

この動作では、IP SLA レスポンダ側を有効にしておく必要はありません。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla** *operation-number*
4. **icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*]
5. **frequency** *seconds*
6. **exit**
7. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sla** *operation-number* 例： スイッチ(config)# **ip sla 10**	IP SLA 動作を作成し、IP SLA コンフィギュレーション モードを開始します。
ステップ 4	**icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*] 例： スイッチ(config-ip-sla)# **icmp-echo 172.29.139.134**	IP SLA 動作を ICMP エコー動作として設定し、ICMP エコー コンフィギュレーション モードを開始します。 • *destination-ip-address* | *destination-hostname*：宛先 IP アドレスまたはホスト名を指定します。 •（任意）**source-ip** {*ip-address* | *hostname*}：送信元 IP アドレスまたはホスト名を指定します。送信元 IP アドレスまたはホスト名が指定されていない場合、IP SLA では、宛先に最も近い IP アドレスが選択されます。 •（任意）**source-interface** *interface-id*：動作に対する送信元インターフェイスを指定します。
ステップ 5	**frequency** *seconds* 例： スイッチ(config-ip-sla-echo)# **frequency 30**	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。指定できる範囲は 1 ～ 604800 秒で、デフォルトは 60 秒です。
ステップ 6	**exit** 例： スイッチ(config-ip-sla-echo)# **exit**	UDP エコー コンフィギュレーション モードを終了します。続いて、グローバル コンフィギュレーション モードに戻ります。
ステップ 7	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]	個々の IP SLA 動作のスケジューリング パラメータを設定します。

	コマンドまたはアクション	目的
	例： スイッチ(config)# **ip sla schedule 5 start-time now life forever**	• *operation-number*：RTR エントリ番号を入力します。 • （任意）**life**：動作の実行を無制限（**forever**）に指定するか、特定の秒数（*seconds*）を指定します。有効な範囲は 0 ～ 2147483647 です。デフォルトは 3600 秒（1 時間）です。 • （任意）**start-time**：情報の収集を開始する時刻を入力します。 特定の時刻に開始する場合は、時、分、秒（24 時間表記）、月日を入力します。月を入力しない場合、当月がデフォルト設定です。 **pending** と入力すれば、開始時刻を指定するまでは情報を収集しません。 **now** と入力すれば、ただちに動作を開始します。 **after** *hh:mm:ss* と入力すると、指定した時刻の経過後に動作を開始します。 • （任意）**ageout** *seconds*：情報を収集していないとき、メモリの動作を保存する秒数を指定します。指定できる範囲は 0 ～ 2073600 秒です。デフォルトは 0 秒（いつまでも保存する）です。 • （任意）**recurring**：毎日、動作を自動的に実行します。
ステップ 8	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 9	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 10	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

ICMP エコー IP SLA 動作の設定

次に、ICMP エコー IP SLA 動作の設定例を示します。

```
スイッチ(config)# ip sla 12
スイッチ(config-ip-sla)# icmp-echo 172.29.139.134
スイッチ(config-ip-sla-echo)# frequency 30
スイッチ(config-ip-sla-echo)# exit
スイッチ(config)# ip sla schedule 5 start-time now life forever
スイッチ(config)# end
スイッチ# show ip sla configuration 22
IP SLAs, Infrastructure Engine-II.

Entry number: 12
Owner:
Tag:
Type of operation to perform: echo
Target address: 2.2.2.2
Source address: 0.0.0.0
Request size (ARR data portion): 28
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Schedule:
    Operation frequency (seconds): 60
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
History Statistics:
    Number of history Lives kept: 0
    Number of history Buckets kept: 15
    History Filter Type: None
Enhanced History:
```

IP SLA 動作のモニタリング

次の表で、IP SLA 動作の設定と結果を表示するために使用するコマンドについて説明します。

表 *61 : IP SLA* 動作のモニタリング

show ip sla application	Cisco IOS IP SLA のグローバル情報を表示します。
show ip sla authentication	IP SLA 認証情報を表示します。
show ip sla configuration [*entry-number*]	すべての IP SLA 動作または特定の IP SLA 動作に関する、デフォルト値をすべて含めた設定値を表示します。
show ip sla enhanced-history {**collection-statistics** | **distribution statistics**} [*entry-number*]	収集した履歴バケットの拡張履歴統計情報、あるいはすべての IP SLA 動作または特定の IP SLA 動作に関する分散統計情報を表示します。
show ip sla ethernet-monitor configuration [*entry-number*]	IP SLA 自動イーサネット設定を表示します。
show ip sla group schedule [*schedule-entry-number*]	IP SLA グループ スケジューリング設定と個別情報を表示します。
show ip sla history [*entry-number* | **full** | **tabular**]	すべての IP SLA 動作について収集した履歴を表示します。
show ip sla mpls-lsp-monitor {**collection-statistics** | **configuration** | **ldp operational-state** | **scan-queue** | **summary** [*entry-number*] | **neighbors**}	MPLS ラベル スイッチド パス（LSP）ヘルス モニター動作を表示します。
show ip sla reaction-configuration [*entry-number*]	すべての IP SLA 動作または特定の IP SLA 動作に関する、予防的しきい値のモニタリングの設定を表示します。
show ip sla reaction-trigger [*entry-number*]	すべての IP SLA 動作または特定の IP SLA 動作に関する反応トリガー情報を表示します。
show ip sla responder	IP SLA レスポンダ側の情報を表示します。
show ip sla statistics [*entry-number* | **aggregated** | **details**]	動作ステータスおよび統計情報の現在値または合計値を表示します。

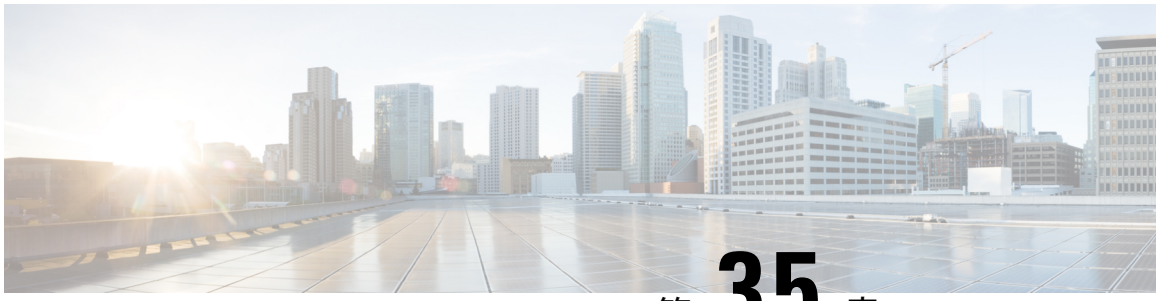
IP SLA 動作のモニタリングの例

次の例は、アプリケーションごとのすべての IP SLA を示しています。

```
スイッチ# show ip sla application

        IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:
        icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http
        dns, udpJitter, dhcp, ftp, udpApp, wspApp

Supported Features:
        IPSLAs Event Publisher

IP SLAs low memory water mark: 33299323
Estimated system max number of entries: 24389

Estimated number of configurable operations: 24389
Number of Entries configured  : 0
Number of active Entries      : 0
Number of pending Entries     : 0
Number of inactive Entries    : 0
Time of last change in whole IP SLAs: *13:04:37.668 UTC Wed Dec 19 2012
```

次の例は、すべての IP SLA ディストリビューション統計情報を示しています。

```
スイッチ# show ip sla enhanced-history distribution-statistics

Point by point Enhanced History
Entry    = Entry Number
Int      = Aggregation Interval
BucI     = Bucket Index
StartT   = Aggregation Start Time
Pth      = Path index
Hop      = Hop in path index
Comps    = Operations completed
OvrTh    = Operations completed over thresholds
SumCmp   = Sum of RTT (milliseconds)
SumCmp2L = Sum of RTT squared low 32 bits (milliseconds)
SumCmp2H = Sum of RTT squared high 32 bits (milliseconds)
TMax     = RTT maximum (milliseconds)
TMin     = RTT minimum (milliseconds)

Entry Int BucI StartT     Pth Hop Comps OvrTh SumCmp     SumCmp2L   SumCmp2H   T
Max    TMin
```



第 35 章

拡張オブジェクト トラッキングの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

拡張オブジェクト トラッキングに関する情報

拡張オブジェクト トラッキングの概要

拡張オブジェクト トラッキング機能が導入される前は、ホットスタンバイ ルータ プロトコル（HSRP）に単純なトラッキング メカニズムが内蔵されていますた。このメカニズムでは、インターフェイスのラインプロトコルのステートしか追跡することができませんでした。インターフェイスのラインプロトコル ステートがダウンになった場合、ルータの HSRP 優先度は削減され、より高い優先度のもう 1 つの HSRP ルータがアクティブになることができます。

拡張オブジェクト トラッキング機能は、HSRP からトラッキング メカニズムを分離させて、独立したトラッキング プロセスを別途生成します。これにより、HSRP 以外のプロセスがこのト

ラッキングプロセスを使用できます。この機能を使用すると、インターフェイスのラインプロトコルのステートに加えて他のオブジェクトも追跡できます。

HSRP、仮想ルータ冗長プロトコル（VRRP）、Gateway Load Balancing Protoco（GLBP）などのクライアント プロセスで、トラッキング オブジェクトに対する興味を登録し、追跡対象オブジェクトの状態が変化したときに通知を受け取るようにすることができます。

各追跡対象オブジェクトには、トラッキング コマンドライン インターフェイス（CLI）で指定される一意の番号があります。クライアントプロセスは、この番号を使用して特定のオブジェクトを追跡します。トラッキングプロセスは、追跡対象オブジェクトに値の変化がないかどうかを定期的にポーリングし、（アップまたはダウン値など）変化があれば登録されているクライアントプロセスに通知します。ただちに通知する場合と、指定された時間遅延後に通知する場合があります。同じオブジェクトを複数のクライアントが追跡して、オブジェクトのステートが変化した場合に、それぞれが異なるアクションを実行できます。

複数のオブジェクトを組み合わせて1つのリストにして追跡することもできます。このリストの状態判定には、重みしきい値またはパーセンテージを使用します。オブジェクトの組み合わせには、ブールロジックを使用できます。「AND」ブール関数を使用する追跡リストの場合、リスト内の各オブジェクトがアップ ステートでないと追跡対象オブジェクトはアップになりません。「OR」ブール関数を使用する追跡リストの場合、リスト内の 1 つのオブジェクトだけがアップ ステートであれば追跡対象オブジェクトはアップになります。

インターフェイス ラインプロトコルまたは IP ルーティング ステートのトラッキング

インターフェイス ラインプロトコル ステートまたはインターフェイス IP ルーティング ステートのいずれかを追跡できます。IP ルーティング ステートを追跡する場合、オブジェクトをアップするには次の 3 つの条件が必要です。

- インターフェイス上で IP ルーティングがイネーブル、かつアクティブになっている。
- インターフェイス ラインプロトコル ステートが使用可能な状態（アップ）にある。
- 既知のインターフェイス IP アドレスを使用している。

この 3 つの条件がすべて合致しないと、IP ルーティング ステートはダウンになります。

追跡リスト

オブジェクトの追跡リストは、ブール式、重みしきい値、またはパーセントしきい値を使用して設定できます。トラッキング対象リストには1つまたは複数のオブジェクトが含まれます。オブジェクトは存在していないと追跡リストに追加できません。

- 設定にブール式による演算を指定する場合は、「AND」または「OR」演算子を使用します。
- 追跡リストのステートを重みしきい値で判定する場合は、追跡リスト内の各オブジェクトに重み番号を割り当てます。追跡リストのステートは、このしきい値に合致したかどうか

で判定されます。各オブジェクトのステートは、すべてのオブジェクトの重みの合計と各オブジェクトのしきい値の重みを比較して判定されます。

- 追跡リストをパーセントしきい値で判定する場合は、追跡リスト内のすべてのオブジェクトにパーセントしきい値を割り当てます。各オブジェクトのステートは、各オブジェクトに割り当てたパーセンテージとリストを比較して判定されます。

他の特性のトラッキング

拡張オブジェクト トラッキングを使用して他の特性を追跡することもできます。

- **track ip route reachability** グローバル コンフィギュレーション コマンドを使用すると、IP ルートの到達可能性を追跡できます。
- **track ip route metric threshold** グローバル コンフィギュレーション コマンドを使用すると、ルートがしきい値を超えているか下回っているかを確認できます。
- **track resolution** グローバル コンフィギュレーション コマンドを使用すると、ルーティング プロトコルのメトリック解決のデフォルト値を変更できます。
- **track timer tracking** コンフィギュレーション コマンドを使用すると、トラッキング対象オブジェクトを定期的にポーリングするようにトラッキング プロセスを設定できます。

拡張オブジェクトトラッキング設定を確認する場合は、 **show track** 特権 EXEC コマンドを使用してください。

IP SLA オブジェクト トラッキング

Cisco IOS IP サービス レベル契約（SLA）は、ネットワーク パフォーマンスの測定と診断を行うツールです。ネットワーク パフォーマンスを測定するためのトラフィック生成には、アクティブ モニタリングが使用されます。Cisco IP SLA 動作は、ネットワークのトラブルシューティングや設計、分析に使用できるリアルタイム メトリックを収集します。

IP SLA 動作のオブジェクト トラッキングを活用すると、クライアントは IP SLA オブジェクトの出力を追跡して、その情報をアクションのトリガーに使用できます。各 IP SLA 動作は、OK または OverThreshold のような簡易ネットワーク管理プロトコル（SNMP）動作の戻りコード値を保持しているため、トラッキング プロセス側で解釈できます。ステートと到達可能性という IP SLA 動作の 2 つの側面をトラッキングできます。ステートの場合、戻りコードが OK のとき、トラック ステートがアップします。リターン コードが OK ではないとき、トラック ステートはダウンします。到達可能性の場合、戻りコードが OK または OverThreshold のとき、到達可能性がアップします。リターン コードが OK ではないとき、到達可能性はダウンします。

スタティック ルート オブジェクト トラッキング

拡張オブジェクトトラッキングを使用したスタティック ルーティング サポートにより、device で ICMP ping を使用して、設定済みのスタティック ルートまたは DHCP ルートがダウンしてい

ることを認識できます。トラッキングを有効にしている場合、システムはルート ステートを追跡し、ステートの変化をクライアントに通知できます。スタティック ルート オブジェクト トラッキングは、プライマリ ゲートウェイへの接続状態をモニターするために、Cisco IP SLA を使用して ICMP ping を生成します。

拡張オブジェクト トラッキングの設定方法

インターフェイスでのライン ステート プロトコルまたは IP ルーティング ステートのトラッキングの設定

インターフェイスのラインプロトコル ステートまたは IP ルーティング ステートを追跡するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **track** *object-number***interface** *interface-id***line-protocol**
4. **delay** { *object-number***up***seconds*[**down***seconds*]|[**up***seconds*]**down***seconds*}
5. **exit**
6. **track** *object-number***interface** *interface-id***ip routing**
7. **delay** { *object-number***up***seconds*[**down***seconds*]|[**up***seconds*]**down***seconds*}
8. **end**
9. **show track***object-number*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *object-number***interface** *interface-id***line-protocol** 例：	（任意）インターフェイスのラインプロトコル ステートを追跡するための追跡リストを作成し、ト

	コマンドまたはアクション	目的
	`スイッチ(config)# track 33 interface gigabitethernet 1/0/1 line-protocol`	ラッキング コンフィギュレーション モードを開始します。 • object-number：追跡対象オブジェクトの番号です。指定できる範囲は 1 ～ 500 です。 • **interface** interface-id は、追跡されるインターフェイスです。
ステップ 4	**delay** { *object-number***up***seconds*[**down***seconds*][**up***seconds*]**down***seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 5	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**track** *object-number***interface** *interface-id***ip routing** 例： `スイッチ(config)# track 33 interface gigabitethernet 1/0/1 ip routing`	（任意）インターフェイスの IP ルーティング ステートを追跡するための追跡リストを作成し、トラッキング コンフィギュレーション モードを開始します。IP ルート追跡では、ルーティング テーブル内の IP ルートおよびインターフェイスの IP パケット ルーティング機能を追跡します。 • object-number：追跡対象オブジェクトの番号です。指定できる範囲は 1 ～ 500 です。 • **interface** interface-id は、追跡されるインターフェイスです。
ステップ 7	**delay** { *object-number***up***seconds*[**down***seconds*][**up***seconds*]**down***seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 8	**end**	特権 EXEC モードに戻ります。
ステップ 9	**show track***object-number*	指定したオブジェクトが追跡されているかどうかを確認します。

追跡リストの設定

重みしきい値による追跡リストの設定

重みしきい値による追跡を行うには、複数オブジェクトを含んだ追跡リストを作成し、重みをしきい値として使用することを指定したあと、各オブジェクトに重み値を設定します。各オブ

ジェクトのステートは、アップであるすべてのオブジェクトの重み合計と各オブジェクトのしきい値の重みを比較して判定されます。

重みしきい値のリストには、「NOT」ブール演算子を使用できません。

重みしきい値を使用してオブジェクトの追跡リストを作成し、各オブジェクトに重み値を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **track** *track-number***list threshold** {**weight**}
4. **object** *object-number*[**weight***weight-number*]
5. **threshold weight** {**up***number*|[**down***number*]}
6. **delay** { **up***seconds*[**down***seconds*]|[**up***seconds*]**down***seconds*}
7. **end**
8. **show track***object-number*
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *track-number***list threshold** {**weight**} 例： スイッチ**(config)# track 4 list threshold weight**	トラッキング対象リスト オブジェクトを設定し、トラッキング コンフィギュレーション モードを開始します。指定できる track-number の範囲は 1 ～ 500 です。 • **threshold—**追跡リストのステートがしきい値に基づくことを指定します。 • **weight—** しきい値が重みに基づくことを指定します。
ステップ 4	**object** *object-number*[**weight***weight-number*] 例：	追跡対象のオブジェクトを指定します。指定できる範囲は 1 ～ 500 です。任意の **weight***weight-number* に

	コマンドまたはアクション	目的
	スイッチ**(config)# object 2 weight 15**	は、オブジェクトのしきい値の重みを指定します。範囲は 1 ～ 255 です。 （注） オブジェクトは存在していないと追跡リストに追加できません。
ステップ 5	**threshold weight** {**up***number*|[**down***number*]} 例： スイッチ**(config-track)# threshold weight up 30 down 10**	（任意）重みしきい値を指定します。 • **up***number*：範囲は 1 ～ 255 です。 • **down***number*：（任意）範囲は **up***number* で選択した数値によって異なります。**up***number* を 25 に設定すると、down number の範囲は 0 ～ 24 になります。
ステップ 6	**delay** {**up***seconds*[**down***seconds*]|[**up***seconds*]**down***seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 7	**end**	特権 EXEC モードに戻ります。
ステップ 8	**show track***object-number*	指定したオブジェクトが追跡されているかどうかを確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

パーセントしきい値による追跡リストの設定

パーセントしきい値による追跡を行うには、複数オブジェクトを含んだ追跡リストを作成し、パーセンテージをしきい値として使用することを指定したあと、リスト内のすべてのオブジェクトにパーセンテージを指定します。リストのステートは、各オブジェクトに割り当てたパーセンテージとリストを比較して判定されます。

パーセントしきい値のリストには、「NOT」ブール演算子を使用できません。

パーセントしきい値を使用してオブジェクトの追跡リストを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **track** *track-number***list threshold** {**percentage**}

4. **object** *object-number*
5. **threshold percentage** {**up***number*|[**down***number*]}
6. **delay** { **up***seconds*[**down***seconds*]|[**up***seconds*]**down***seconds*}
7. **end**
8. **show track***object-number*
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *track-number***list threshold** {**percentage**} 例： スイッチ**(config)# track 4 list threshold percentage**	トラッキング対象リスト オブジェクトを設定し、トラッキング コンフィギュレーション モードを開始します。指定できる track-number の範囲は 1 ～ 500 です。 • **threshold—**追跡リストのステートがしきい値に基づくことを指定します。 • **percentage—** しきい値がパーセンテージに基づくことを指定します。
ステップ 4	**object** *object-number* 例： スイッチ**(config)# object 1**	追跡対象のオブジェクトを指定します。指定できる範囲は 1 ～ 500 です。 （注） オブジェクトは存在していないと追跡リストに追加できません。
ステップ 5	**threshold percentage** {**up***number*|[**down***number*]} 例： スイッチ**(config)# threshold percentage up 51 down 10**	（任意）パーセントしきい値を指定します。 • **up***number*：範囲は 1 ～ 100 です。 • **down***number*：（任意）範囲は **up***number* で選択した数値によって異なります。**up***number* を 25 に設定すると、down number の範囲は 0 ～ 24 になります。

	コマンドまたはアクション	目的
ステップ 6	**delay** { **up***seconds*[**down***seconds*]|[**up***seconds*]**down***seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 7	**end**	特権 EXEC モードに戻ります。
ステップ 8	**show track***object-number*	指定したオブジェクトが追跡されているかどうかを確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

HSRP オブジェクト トラッキングの設定

特定のオブジェクトを追跡し、そのオブジェクトのステートに基づいてHSRPプライオリティを変更できるようにスタンバイ HSRP グループを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **track** *object-number*{**interface** *interface-id*{**line-protocol**|**ip routing**}|**ip route***ip address/prefix-length*{**metric threshold**|**reachability**}**list**{**boolean**{**and**|**or**}}|{**threshold**{**weight**|**percentage**}}}
4. **exit**
5. **interface** { *interface-id*
6. **standby**[*group-number*]**ip**[*ip-address***secondary**]]
7. **standby**[*group-number*]**track**[*object-number*[**decrement** *priority-decrement*]]
8. **end**
9. **show standby**
10. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *object-number*{**interface** *interface-id*{**line-protocol**|**ip routing**}|**ip route***ip address/prefix-length*{**metric threshold**|**reachability**}|**list**{**boolean**{**and**|**or**}|{**threshold**{**weight***percentage*}}}	（任意）設定されたステートを追跡するための追跡リストを作成し、トラッキング コンフィギュレーション モードを開始します。 • object-number：追跡対象オブジェクトの番号です。指定できる範囲は 1 ～ 500 です。 • 追跡するインターフェイスを指定するには、**interface** *interface-id* を入力します。 • インターフェイス ライン プロトコルの状態を追跡するには **line-protocol** を入力します。また、インターフェイス IP ルーティングの状態を追跡するには、**ip routing** を入力します。 • IP ルートの状態を追跡するには、**ip route***ip-address/prefix-length* を入力します。 • しきい値メトリックを追跡する場合は **metric threshold**、ルートが到達可能かどうかを追跡するには **reachability** を入力します。 デフォルトの up しきい値は 254、デフォルトの down しきい値は 255 です。 • リスト内の一連のオブジェクトを追跡するには、**list** を入力します。 （注） 追跡するインターフェイスごとにこの手順を繰り返してください。
ステップ 4	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 5	**interface** { *interface-id*	インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	**standby**[*group-number*]**ip**[*ip-address***secondary**]]	HSRP グループの番号および仮想 IP アドレスを使用して、HSRP グループを作成（またはイネーブルに）します。

	コマンドまたはアクション	目的
		• （任意）*group-number*：HSRP をイネーブルにするインターフェイスのグループ番号を入力します。指定できる範囲は 0 ～ 255 です。デフォルトは 0 です。HSRP グループが 1 つしかない場合は、グループ番号を入力する必要はありません。 • （1 つのインターフェイスで必須、それ以外は任意）*ip-address*：ホットスタンバイ ルータ インターフェイスの仮想 IP アドレスを指定します。少なくとも 1 つのインターフェイスに対して仮想 IP アドレスを入力する必要があります。他のインターフェイスは、その仮想 IP アドレスを学習します。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。このキーワードが省略された場合、設定されたアドレスはプライマリ IP アドレスになります。
ステップ 7	**standby**[*group-number*]**track**[*object-number*[**decrement** *priority-decrement*]]	特定のオブジェクトを追跡し、そのオブジェクト ステートに基づいてホットスタンバイ プライオリティを変更できるように HSRP を設定します。 • （任意）*group-number*：追跡が適用されるグループ番号を入力します。 • *object-number*：追跡対象のオブジェクト番号を入力します。指定できる範囲は 1 ～ 500 で、デフォルトは 1 です。 • （任意）**secondary**：IP アドレスがセカンダリ ホット スタンバイ ルータ インターフェイスであることを指定します。このキーワードが省略された場合、設定されたアドレスはプライマリ IP アドレスになります。 • （任意）**decrement***priority-decrement*：追跡対象のオブジェクトがダウンになった場合（またはアップに戻った場合）に、ルータのホットスタンバイの優先順位を減少（または増加）させる幅を指定します。指定できる範囲は 1 ～ 255 で、デフォルトは 10 です。
ステップ 8	**end**	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 9	**show standby**	スタンバイ ルータの IP アドレスおよび追跡ステートを確認します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IP SLA オブジェクト トラッキングの設定

IP SLA 動作のステートまたは IP SLA IP ホストの到達可能性を追跡するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **track** *object-number* **ip sla** *operation-number* {**state | reachability**}
4. **delay** { **up***seconds*[**down***seconds*]|[**up***seconds*]**down***seconds*}
5. **end**
6. **show track***object-number*
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**track** *object-number* **ip sla** *operation-number* {**state	reachability**} 例： スイッチ**(config)# track 2 ip sla 123 state**	トラッキング コンフィギュレーション モードを開始し、IP SLA 動作のステートを追跡します。 • *object-number* の範囲は 1 ～ 500 です。

	コマンドまたはアクション	目的
		• *operation-number* の範囲は 1 ～ 2147483647 です。
ステップ 4	**delay** { **up***seconds*[**down***seconds*]|[**up***seconds*]**down***seconds*}	（任意）追跡対象オブジェクトのステート変更の通信を遅延させる時間（秒）を指定します。指定できる範囲は 1 ～ 180 秒です。
ステップ 5	**end**	特権 EXEC モードに戻ります。
ステップ 6	**show track***object-number*	指定したオブジェクトが追跡されているかどうかを確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

スタティック ルート オブジェクト トラッキングの設定

スタティック ルーティング用のプライマリ インターフェイスの設定

スタティック ルーティングのプライマリ インターフェイスを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface***interface-id*
4. **description***string*
5. **ip address***ip-address mask*[**secondary**]
6. **exit**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface***interface-id*	プライマリまたはセカンダリ インターフェイスを選択し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**description***string*	インターフェイスに説明を追加します。
ステップ 5	**ip address***ip-address mask*[**secondary**]	インターフェイスのプライマリまたはセカンダリ IP アドレスを設定します。
ステップ 6	**exit**	グローバル コンフィギュレーション モードに戻ります。

DHCP のプライマリ インターフェイスの設定

DHCP のプライマリ インターフェイスを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface***interface-id*
4. **description***string*
5. **ip dhcp client route track***number*
6. **exit**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface***interface-id*	プライマリまたはセカンダリ インターフェイスを選択し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**description***string*	インターフェイスに説明を追加します。
ステップ 5	**ip dhcp client route track***number*	DHCP クライアントを設定し、追加されたルートを指定の追跡番号に関連付けます。有効な数値は 1 ～ 500 です。
ステップ 6	**exit**	グローバル コンフィギュレーション モードに戻ります。

IP SLA モニタリング エージェントの設定

プライマリ インターフェイスおよびエージェント状態をモニターするトラック オブジェクトを使用して、IP アドレスの ping を実行するように IP SLA エージェントを設定することができます。

Cisco IP SLA でネットワーク モニタリングを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla***operation number*
4. **icmp-echo**{ *destination ip-address*|*destination hostname*[**source - ipaddr**{*ip-address*|*hostname***source-interface***interface-id*]
5. **timeout***milliseconds*
6. **frequency***seconds*
7. **threshold***milliseconds*
8. **exit**
9. **ip sla schedule** *operation-number*[**life** {**forever**|*seconds*}]**start-time***time*|**pending**|**now**|**after***time*]**ageout***seconds*][**recurring**]
10. **track** *object-number***rtr** *operation-number***statereachability**
11. **end**
12. **show track***object-number*
13. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sla***operation number*	Cisco IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードを開始します。
ステップ 4	**icmp-echo**{ *destination ip-address*|*destination hostname*[**source - ipaddr**{*ip-address*|*hostname***source-interface***interface-id*]	Cisco IP SLA エンドツーエンド ICMP エコー応答時間動作を設定し、IP SLA ICMP エコー コンフィギュレーション モードを開始します。
ステップ 5	**timeout***milliseconds*	要求パケットの応答に対する動作の待機時間を設定します。
ステップ 6	**frequency***seconds*	動作がネットワークに送信される頻度を設定します。
ステップ 7	**threshold***milliseconds*	反応イベントを生成し、その動作の履歴情報を保存するしきい値（ヒステリシス）の上限を設定します。
ステップ 8	**exit**	IP SLA ICMP エコー コンフィギュレーション モードを終了します。
ステップ 9	**ip sla schedule** *operation-number*[**life** {**forever**|*seconds*}]**start-time***time***pending**|**now**|**after***time*][**ageout***seconds*][**recurring**] 例： スイッチ(config)# track 2 200 state	単一の IP SLA 動作のスケジューリング パラメータを設定します。 • *object-number* の範囲は 1 ～ 500 です。 • *operation-number* の範囲は 1 ～ 2147483647 です。
ステップ 10	**track** *object-number***rtr** *operation-number***statereachability**	Cisco IOS IP SLA 動作の状態を追跡し、トラッキング コンフィギュレーション モードを開始します。
ステップ 11	**end**	特権 EXEC モードに戻ります。
ステップ 12	**show track***object-number*	指定したオブジェクトが追跡されているかどうかを確認します。
ステップ 13	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

ルーティング ポリシーおよびデフォルト ルートの設定

オブジェクト トラッキングを使用してバックアップ スタティック ルーティングのルーティング ポリシーを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **access-list***access-list-number*
4. **route-map***map tag*[**permit**|**deny**][*sequence-number*]
5. **match ip address**{*access-list number*[**permit**|**deny**][*sequence-number*]
6. **set ip next-hop dynamic dhcp**
7. **set interface***interface-id*
8. **exit**
9. **ip local policy route-map***map tag*
10. **ip route***prefix mask*{*ip address*|*interface-id*[*ip address*]}[*distance*][*name*][**permanent**|**track***track-number*][*tag tag*]
11. **end**
12. **show ip route track table**
13. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**access-list***access-list-number*	拡張 IP アクセス リストを定義します。オプションの文字を設定します。

	コマンドまたはアクション	目的
ステップ 4	**route-map***map tag*[**permit**|**deny**][*sequence-number*]	ルートマップ コンフィギュレーション モードを開始し、特定のルーティングから別のルーティングへの再配信ルートの条件を定義します。
ステップ 5	**match ip address**{*access-list number*[**permit**|**deny**][*sequence-number*]	標準または拡張アクセス リストに許可された宛先ネットワーク番号アドレスを持つルートを配信し、パケットのポリシー ルーティングを実行します。複数の番号または名前を入力できます。
ステップ 6	**set ip next-hop dynamic dhcp**	DHCP ネットワーク専用。DHCP クライアントが学んだ最新のゲートウェイへのネクスト ホップを設定します。
ステップ 7	**set interface***interface-id*	スタティック ルーティング ネットワーク専用。ポリシー ルーティングのルート マップ一致条件をパスした出力パケットの送信場所を指定します。
ステップ 8	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 9	**ip local policy route-map***map tag*	ルート マップを特定し、ローカル ポリシー ルーティングに使用します。
ステップ 10	**ip route***prefix mask*{*ip address*|*interface-id*[*ip address*]}[*distance*][*name*][**permanent**|**track***track-number*][*tag tag*]	スタティック ルーティング ネットワーク専用。スタティック ルートを確立します。**track***track-number* を入力し、設定したトラックオブジェクトがアップの場合に限り、静的ルートがインストールされるように指定します。
ステップ 11	**end**	特権 EXEC モードに戻ります。
ステップ 12	**show ip route track table**	IP ルート トラック テーブルの情報を表示します。
ステップ 13	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

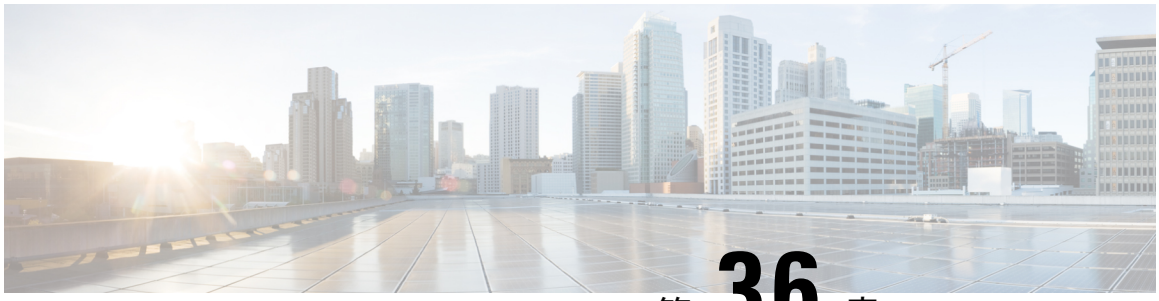
拡張オブジェクト トラッキングのモニタリング

下の表に示す特権 EXEC コマンドまたはユーザー EXEC コマンドを使用して、拡張オブジェクトの追跡情報を表示します。

。

表 *62 :* 追跡情報を表示するコマンド

コマンド	目的
show ip route track table	IP ルート トラック テーブルの情報を表示します。
show track [*object-number*]	すべての追跡リストまたは指定リストの情報を表示します。
show track brief	すべてのインターフェイスまたは指定されたインターフェイスに対する VTP のステータスおよび設定を表示します。
show track interface [brief]	追跡対象のインターフェイス オブジェクトに関する情報を表示します。
show track ip [*object-number*][**brief**]**route**	追跡対象 IP ルート オブジェクトの情報を表示します。
show track resolution	追跡対象パラメータの解像度を表示します。
show track timer	追跡対象のポーリング インターバル タイマーを表示します。



第 36 章

スイッチ スタックの管理

スイッチ スタックの前提条件

スイッチ スタックの制約事項

Catalyst 3560cx の水平スタックに関する制約事項はありません。

スイッチ スタックに関する情報

水平スタック構成

水平スタック構成には、10G SFP+ アップリンク ポートおよび MGig ポートをサポートする Catalyst 3560CX シリーズ スイッチを含めることができます。SFP+ を使用して、MGig ポートをさまざまなロケーションに配置されたボックスに光ケーブルと銅ケーブルで接続し、小型ボックスが異なるフロアやビルに配置されたスタックを構成することができます。必要に応じてハーフ リンクまたはフル リングを形成し、残りのアップリンク ポートは引き続きネットワーク ポートとして動作させることができます。

ネットワーク ポートをスタック ポートに変換しても、そのポートはスイッチを次にリロードするまでは引き続きネットワーク ポートとして機能するため、現在の実行コンフィギュレー

ションに影響はありません。その特定のネットワーク ポートのすべての現在の設定は、ポートがスタック ポートとして起動するとスイッチのリロード後に失われます。

スタック ポートをネットワーク ポートに戻す場合、スイッチが次にリロードされるまでは引き続きスタック ポートとして機能します。スイッチのリロード後、ポートはデフォルト設定を適用したネットワーク ポートとして起動されます。

（注） アップリンク ポートがスタック ポートとして機能している間、それらの特定のアップリンク インターフェイス（Te1/0/1 など）は他のすべてのネットワーク ポートとは異なり、show コマンドでリストされることはありません。また、コンフィギュレーション コマンドで使用することもできません。スイッチがリロードされてポートがネットワーク ポートに変換されてからでなければ、それらのポートは使用できません。

表 *63 :* 水平スタック構成をサポートする *C3560CX* スイッチ

製品 ID	アクセス ポート	アップリンク	スタック可能なポート
WS-C3560CX-12PD-S	12 GE	2GE + 2SFP+	2 つの 10G アップリンク
WS-C3560CX-8XPD-S	6 GE	2 つのマルチギガと 2 つのSFP+	1 つのマルチギガと 1 つの 10G アップリンク、2 つのマルチギガ、または 2 つの 10G アップリンク

スイッチ スタックのメンバーシップ

スイッチ スタックは、スタック ポートを使用して接続された最大 8 台のスタック メンバから構成されます。スイッチスタックには、必ず 1 個のアクティブスイッチがあります。

スタンドアロンデバイスは、アクティブスイッチとしても動作するスタックメンバーを1つだけ持つデバイススタックです。スタンドアロンデバイスをもう 1 つの同じものと接続して、2 つのスタックメンバーで構成され、一方がアクティブスイッチであるスタックを構築できます。スタンドアロンデバイスを既存のデバイススタックに接続して、スタックメンバーシップを増やすことができます。

図 *61 : 2* 台のスタンドアロン スイッチからのスイッチ スタックの構築

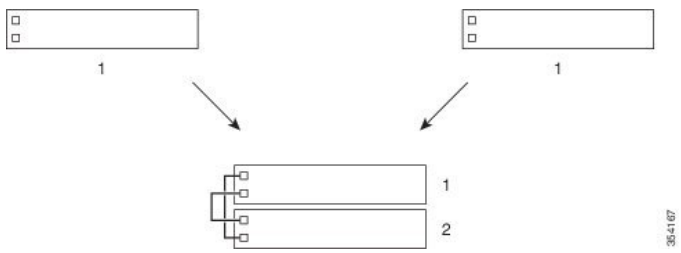
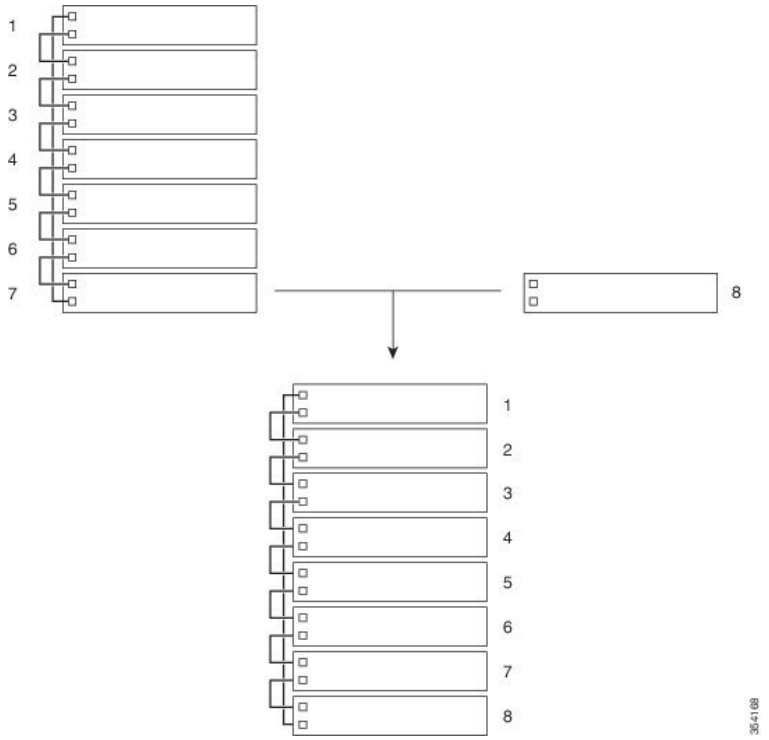


図 *62 :* スタンドアロン スイッチのスイッチ スタックへの追加



スイッチ スタック メンバーシップの変更

スタック メンバを同一のモデルと交換した場合、新たなスイッチ（プロビジョニングされるスイッチとも呼びます）は交換されたスイッチと同じメンバ番号を使用すると、交換されたスイッチとまったく同じ設定で機能します。

アクティブ スイッチを削除したり、電源の入ったスタンドアロン スイッチまたはスイッチ スタックを追加したりしないかぎり、メンバーシップの変更中も、スイッチ スタックの動作は間断なく継続されます。

- 電源が入っているスイッチを追加すると（マージ）、マージ中のスイッチスタックの各アクティブスタックは、その中から1台のアクティブスタックを選択します。再選択されたアクティブスタックは役割と設定を保持し、そのスタックメンバーも同様に保持します。それ以前のアクティブスタックを含め残りのすべてのスイッチは、リロードされ、スタックメンバーとしてスイッチスタックに参加します。また、スタックメンバー番号を使用可能な最小の番号に変更し、再選択されたアクティブスタックのスタック設定を使用します。

- 電源が入った状態のスタック メンバを取り外すと、スイッチ スタックが、それぞれ同じ設定を持つ2つ以上のスイッチ スタックに分割（パーティション化）されます。これにより、以下の現象が発生する可能性があります。

  - ネットワーク内での IP アドレスの競合。スイッチ スタックを分離されたままにしておきたい場合は、新しく作成されたスイッチ スタックの IP アドレス（複数の場合あり）を変更してください。

- スタック内の 2 つのメンバー間の MAC アドレスの競合。**stack-mac update force** コマンドを使用して、この競合を解消できます。

（注） スイッチスタックに追加または削除するスイッチの電源がオフであることを確認します。

スタック メンバーを追加または削除したら、スイッチ スタックがすべての帯域幅で動作していることを確認します。スタック モード LED が点灯するまで、スタック メンバの Mode ボタンを押します。スタック内のすべてのスイッチでは、右側の最後の 2 つのポート LED がグリーンに点灯します。スイッチ モデルに応じて、右側の最後の 2 つのポートは 10 ギガビット イーサネット ポートまたは Small Form-Factor Pluggable（SFP）モジュール ポート（10/100/1000 ポート）になります。スイッチの一方または両方の LED がグリーンでない場合、スタックは全帯域幅で稼働していません。

新しいスタック メンバーが既存のスイッチ スタックに追加される場合、スタック コンバージェンスに最大 4 秒かかることがあります。

スタックを分割しないで、電源が入ったスタック メンバを取り外す場合、次の手順を実行します。

- 新規に作成されたスイッチ スタックのスイッチの電源をオフにします。
- それをそのスタック ポートを介して元のスイッチ スタックに再接続します。
- スイッチの電源を入れます。

スタック メンバー番号

スタック メンバ番号（1 ～ 8）は、スイッチ スタック内の各メンバを識別します。また、メンバー番号によって、スタック メンバーが使用するインターフェイス レベルの設定が決定します。**show switch** EXEC コマンドを使用すると、スタックメンバー番号を表示できます。

新しい初期設定状態（デバイススタックに参加していない、またはスタックメンバー番号が手動で割り当てられていない）デバイスは、デフォルトスタックメンバー番号 1 で出荷されます。そのデバイスがデバイススタックに参加すると、そのデフォルトスタックメンバー番号がスタック内で使用可能な最小メンバー番号に変更されます。

同じスタック内のスタックメンバーが同じスタックメンバー番号を持つことはできません。スタンドアロンデバイスを含むすべてのスタックメンバーは、番号が手動で変更されるまで、または、その番号がスタック内の他のメンバーによってすでに使用されていないかぎり、独自のメンバー番号を保持します。

- **switch***current-stack-member-number* **renumber** *new-stack-member-number* コマンドを使用して手動でスタックメンバー番号を変更した場合は、その番号がスタック内の他のメンバーに未割り当てなときにだけ、スタックメンバーのリセット後（または **reload slot** *stack-member-number* 特権 EXEC コマンドの使用後）に新番号が有効となります。スタッ

クメンバー番号を変更するもう 1 つの方法は、device_NUMBER 環境変数を変更することです。

番号がスタック内の他のメンバーによって使用されている場合は、デバイスがスタック内で使用可能な最小番号を選択します。

手動でスタック メンバーの番号を変更し、新たなメンバー番号にインターフェイス レベルの設定が関連付けられていない場合は、スタック メンバーをデフォルト設定にリセットします。

プロビジョニングされたデバイス上では、**switch** *current-stack-member-number* **renumber** *new-stack-member-number* コマンドを使用できません。使用すると、コマンドは拒否されます。

- スタックメンバーを別のデバイススタックに移動した場合、そのスタックメンバーは、自分の番号がスタック内の他のメンバーによって使用されていない場合にだけ、その番号を保持します。その番号が使用されている場合は、デバイスがスタック内で使用可能な最小番号を選択します。
- デバイススタックをマージした場合は、新しいアクティブデバイスのデバイススタックに参加しているデバイスがスタック内で使用可能な最小番号を選択します。

ハードウェア インストレーション ガイドに記載されているように、デバイスポート LED をスタックモードで使用すれば、各スタックメンバーのスタックメンバー番号を目視で確認できます。

デフォルトモードでは、アクティブスイッチのスタック LED だけが緑色に点滅します。ただし、[MODE] ボタンを **[Stack]** オプションまでスクロールすると、すべてのスタック メンバのスタック LED が緑色に点灯します。

[モード（Mode）] ボタンが [スタック（Stack）] オプションまでスクロールすると、各スタックメンバーのスイッチ番号が、そのスイッチの最初の 5 つのポートの LED で表示されます。スイッチ番号は、すべてのスタック メンバで、バイナリ形式で表示されます。スイッチでは、オレンジ色の LED は値 0、緑の LED は値 1 を示します。

スイッチ番号 5（バイナリ 00101）の例：

最初の 5 つのLEDは、スイッチ番号 5 のスタック メンバ上で次のように点灯します。

- ポート 1：オレンジ
- ポート 2：オレンジ
- ポート 3：緑
- ポート 4：オレンジ
- ポート 5：緑

同様に、スイッチ番号に基づき、すべてのスタック メンバーで、最初の 5 つの LED がオレンジ色か緑色に点灯します。

（注）
- 水平スタック ポートを相手側の通常のネットワーク ポートに接続した場合、相手側から受信した SDP パケットがないと、スタック ポートの送受信は 30 秒以内に無効になります。
- スタック ポートはダウンしませんが、送受信だけ無効になります。次に示すログ メッセージがコンソールに表示されます。ピア側のネットワーク ポートがスタック ポートに変換されると、このスタック ポートの送受信が有効になります。

```
%STACKMGR-4-HSTACK_LINK_CONFIG: Verify peer stack port setting for
hstack StackPort-1 switch 5 (hostname-switchnumber)
```

スタック メンバーのプライオリティ値

スタックメンバーのプライオリティ値が高いほど、アクティブスイッチとして選択され、自分のスタックメンバー番号を保持できる可能性が高くなります。プライオリティ値は 1 ～ 15 の範囲で指定できます。デフォルトのプライオリティ値は 1 です。**show switch** EXEC コマンドを使用すると、スタックメンバーのプライオリティ値を表示できます。

（注） アクティブデバイスにしたいデバイスには、最高のプライオリティ値を割り当てることをお勧めします。これにより、再選択が発生したときにそのデバイスがアクティブデバイスとして選択されます。

スタックメンバーのプライオリティ値を変更するには、**switch***stack-member-number* **priority** *new priority-value* コマンドを使用します。詳細については、「スタック メンバー プライオリティ値の設定」のセクションを参照してください。

新しいプライオリティ値はすぐに有効となりますが、現在のアクティブデバイスには影響しません。新たなプライオリティ値は、現在のアクティブデバイスまたはデバイススタックのリセット時に、どのスタックメンバーが新たなアクティブデバイスとして選択されるかを決定する場合に影響を及ぼします。

スイッチ スタック ブリッジ ID と MAC アドレス

アクティブスイッチの MAC アドレスによって、スタック MAC アドレスが決定されます。

スタックが初期化した場合は、アクティブスイッチの MAC アドレスによって、ネットワーク内のスタックを識別するブリッジ ID が決定されます。

アクティブスイッチが変わると、新たなアクティブスイッチの MAC アドレスによって、新たなブリッジ ID とスタック MAC アドレスが決まります。

スイッチスタック全体をリロードする場合、スイッチスタックがアクティブスイッチの MAC アドレスを使用します。

スイッチ スタック上の永続的 MAC アドレス

また、スタック MAC アドレスが新しいアクティブスイッチ MAC アドレスに変更されないように、スタック MAC の永続性を設定することもできます。

アクティブ スイッチとスタンバイ スイッチの選択と再選択

アクティブスイッチは、次の要因のいずれか1つに基づいて、示されている順に選択または再選択されます。

1. 現在のアクティブスイッチであるスイッチ。
2. 最高のスタック メンバ プライオリティ値を持つスイッチ

（注） アクティブスイッチにしたいスイッチには、最高のプライオリティ値を割り当てることをお勧めします。それにより、アクティブスイッチの再選択時に、そのスイッチが再びアクティブスイッチとして選択されます。

3. MAC アドレスが最小のスイッチ

スイッチ スタックのコンフィギュレーション ファイル

コンフィギュレーション ファイルには、次の設定情報が格納されています。

- すべてのスタック メンバーに適用される IP 設定、STP 設定、VLAN 設定、SNMP 設定などのシステム レベル（グローバル）のコンフィギュレーション設定
- スタック メンバーのインターフェイス固有のコンフィギュレーション設定：各スタック メンバーに固有

（注） 実行コンフィギュレーションをスタートアップ コンフィギュレーションに保存せずにアクティブスイッチを交換した場合は、アクティブスイッチのインターフェイス固有の設定が保存されます。

スイッチスタックに参加している新しい初期設定のままのデバイスは、そのスイッチスタックのシステムレベルの設定を使用します。デバイスが電源をオンにする前に別のスイッチスタックに移動された場合、そのデバイスは保存されたコンフィギュレーションファイルを失って、新しいスイッチスタックのシステムレベルの設定を使用します。デバイスが新しいスイッチスタックに参加する前にスタンドアロンデバイスとして電源をオンにされた場合は、スタックがリロードされます。スタックがリロードされると、新しいデバイスがデバイスになって、そのコンフィギュレーションを保持し、他のスタックメンバーのコンフィギュレーションファイルを上書きする可能性があります。

各スタック メンバーのインターフェイス固有のコンフィギュレーションには、スタック メンバー番号が関連付けられます。スタック メンバーは、番号が手動で変更された場合、または同じスイッチ スタック内の他のメンバーによってすでに使用されている場合以外は、自分の番号を保持します。スタック メンバーの番号を変更した場合は、そのスタック メンバーのリセット後に新しい番号が有効になります。

- そのメンバー番号に対応するインターフェイス固有のコンフィギュレーションが存在しない場合は、スタック メンバーはデフォルトのインターフェイス固有のコンフィギュレーションを使用します。
- そのメンバー番号に対応するインターフェイス固有のコンフィギュレーションが存在する場合は、スタック メンバーはそのメンバー番号に関連付けられたインターフェイス固有のコンフィギュレーションを使用します。

故障したメンバーを同一のモデルに交換すると、交換後のメンバーが、自動的に、故障したデバイスと同じインターフェイス固有のコンフィギュレーションを使用します。インターフェイス設定を再設定する必要はありません。交換後のデバイス（プロビジョニングされたデバイスとも呼ばれる）には、故障したデバイスと同じスタックメンバー番号を割り当てる必要があります。

スタンドアロンデバイスのコンフィギュレーションの場合と同様に、スタック コンフィギュレーションをバックアップして復元します。

スタック メンバーを割り当てるためのオフライン設定

オフライン設定機能を使用すると、新しいスイッチがスイッチ スタックに参加する前に、スイッチに割り当て（設定を割り当て）できます。現在スタックに属していないスイッチに関連付けられたスタック メンバー番号、スイッチ タイプ、およびインターフェイスを設定できます。スイッチ スタックで作成した設定を割り当てられた設定と呼びます。スイッチ スタックに追加され、この設定を受信するスイッチを割り当てられたスイッチと呼びます。

switch*stack-member-number* **provision** *type* グローバル コンフィギュレーション コマンドにより、手動で設定を作成しプロビジョニングします。*stack-member-number* は、スタックに追加する前に、プロビジョニングされたスイッチ上で変更する必要があり、スイッチ スタック上の新しいスイッチ用に作成したスタック メンバー番号と一致する必要があります。割り当てられた設定内のスイッチ タイプは新しく追加したスイッチのスイッチ タイプと一致する必要があります。スイッチ スタックにスイッチを追加する場合に、割り当てられた設定が存在しないときは、割り当てられる設定が自動的に作成されます。

プロビジョニングされたスイッチに関連付けられているインターフェイスを設定すると、スイッチ スタックがその設定を受け入れ、実行コンフィギュレーションにその情報が表示されます。ただし、スイッチがアクティブでないため、インターフェイス上の設定が機能しないうえ、割り当てられたスイッチに関連付けられたインターフェイスが特定の機能の表示には現れません。たとえば、プロビジョニングされたスイッチに関連付けられている VLAN 設定情報は、スイッチスタック上の **show vlan** ユーザー EXEC コマンド出力に表示されません。

スイッチ スタックは、割り当てられたスイッチがスタックに属するかどうかに関係なく、実行コンフィギュレーションに割り当てられた設定を保持します。**copy running-config startup-config**

特権 EXEC コマンドを入力すると、プロビジョニングされた設定をスタートアップ コンフィギュレーション ファイルに保存できます。スタートアップ コンフィギュレーション ファイルでは、割り当てられたスイッチがスタックに属するかどうかに関係なく、スイッチ スタックは保存した情報をリロードして使用できます。

割り当てられたスイッチのスイッチ スタックへの追加による影響

プロビジョニングされたデバイスをスイッチ スタックに追加すると、スタックはプロビジョニングされた設定かデフォルト設定のどちらかを適用します。下の表に、スイッチ スタックが、プロビジョニングされた設定とプロビジョニングされたスイッチを比較するときに発生するイベントを示します。

表 *64* : プロビジョニングされた設定とプロビジョニングされたスイッチの比較結果

シナリオ		結果
スタック メンバー番号とデバイス タイプが一致する場合。	**1.** プロビジョニングされたスイッチのスタック メンバ番号と、スタックのプロビジョニングされた設定のスタック メンバ番号が一致する場合、かつ **2.** プロビジョニングされたスイッチのデバイス タイプと、スタック上でプロビジョニングされた設定内のデバイス タイプが一致する場合。	スイッチ スタックは、プロビジョニングされた設定をプロビジョニングされたスイッチに適用し、スタックに追加します。
スタック メンバー番号は一致するが、デバイス タイプが一致しない場合。	**1.** プロビジョニングされたスイッチのスタック メンバ番号と、スタックのプロビジョニングされた設定のスタック メンバ番号が一致する場合、ただし **2.** プロビジョニングされたスイッチのデバイス タイプと、スタック上でプロビジョニングされた設定内のデバイス タイプが一致しない場合。	スイッチ スタックは、デフォルト設定をプロビジョニングされたスイッチに適用し、スタックに追加します。 プロビジョニングされた設定は、新しい情報を反映するために変更されます。

シナリオ		結果
プロビジョニングされた設定でスタック メンバ番号が検出されない		スイッチ スタックは、デフォルト設定をプロビジョニングされたスイッチに適用し、スタックに追加します。 プロビジョニングされた設定は、新しい情報を反映するために変更されます。
プロビジョニングされたスイッチのスタック メンバ番号が、プロビジョニングされた設定で検出されない		スイッチ スタックは、デフォルト設定をプロビジョニングされたスイッチに適用し、スタックに追加します。

プロビジョニングされた設定で指定されたタイプとは異なるプロビジョニングされたスイッチを、電源が切られたスイッチスタックに追加して電力を供給すると、スイッチスタックがスタートアップ コンフィギュレーション ファイル内の（現在は不正な）**switch** *stack-member-number* **provision** *type* グローバル コンフィギュレーション コマンドを拒否します。ただし、スタックの初期化中は、スタートアップ コンフィギュレーション ファイルのデフォルトでないインターフェイス コンフィギュレーション情報が、（間違ったタイプの可能性がある）割り当てられたインターフェイス向けに実行されます。実際のデバイス タイプと前にプロビジョニングされたスイッチ タイプの違いによって、拒否されるコマンドと、受け入れられるコマンドがあります。

（注） スイッチ スタックに新しいデバイスのプロビジョニングされた設定が含まれていない場合は、デバイスがデフォルトのインターフェイス設定でスタックに参加します。その後、スイッチスタックが、新しいデバイスと一致する **switch** *stack-member-number* **provision** *type* グローバル コンフィギュレーション コマンドで、その実行コンフィギュレーションに追加されます。設定情報については、「スイッチ スタックへの新しいメンバーのプロビジョニング」のセクションを参照してください。

スイッチ スタックの割り当てられたスイッチの交換による影響

スイッチ スタック内の割り当てられたスイッチに障害が発生し、スタックから削除して別のデバイスと交換すると、スタックが割り当てられた設定またはデフォルト設定をそのスイッチに適用します。スイッチ スタックが割り当てられた設定と割り当てられたスイッチを比較するときに発生するイベントは、割り当てられたスイッチをスタックに追加するときに発生するものと同じです。

割り当てられたスイッチのスイッチ スタックからの削除による影響

割り当てられたスイッチをスイッチ スタックから削除すると、削除されたスタック メンバーに関連付けられた設定は、割り当てられた情報として実行コンフィギュレーション内に残ります。設定を完全に削除するには、**no switch** *stack-member-number* **provision** グローバル コンフィギュレーション コマンドを使用します。

スタック プロトコル バージョン

各ソフトウェア イメージには、スタック プロトコル バージョンが含まれます。スタック プロトコル バージョンには、メジャー バージョン番号とマイナー バージョン番号があります（たとえば、1.4 の場合、1 がメジャー バージョン番号、4 がマイナー バージョン番号になります）。両方のバージョン番号によって、スタック メンバー間の互換性レベルが決定します。

Cisco IOS ソフトウェア バージョンが同じスイッチは、スタック プロトコル バージョンも同じです。このようなスイッチは完全に互換可能で、すべての機能がスイッチ スタック全体に亘って適切に動作します。アクティブスイッチと Cisco IOS ソフトウェアバージョンが同じデバイスは、すぐにスイッチスタックに参加できます。

非互換性が混合する場合は、完全な機能を備えたスタック メンバーが、特定のスタック メンバーとの非互換が生じていることを示すシステム メッセージを生成します。アクティブスイッチがこのメッセージをすべてのスタックメンバーに送信します。

詳細については、「スイッチ間のメジャーバージョン番号の非互換性」の手順と「スイッチ間のマイナー バージョン番号の非互換性」の手順を参照してください。

スタック可能なスイッチ間のメイン スタック プロトコル バージョン番号の非互換性

多くの場合、異なる Cisco IOS ソフトウェア バージョンのデバイスは、スタック プロトコル バージョンも異なります。メジャーバージョン番号が異なるデバイスは非互換で、同じスイッチ スタック内には存在できません。

スタック可能なスイッチ間のマイナー スタック プロトコル バージョン番号の非互換性

自動アップグレード

自動アップグレード機能の目的は、スイッチを互換性のあるソフトウェアイメージにアップグレードしてスイッチ スタックに参加できるようにすることです。

新しいスイッチがスイッチ スタックに参加しようとすると、各スタック メンバーがそれ自体と新しいスイッチの互換性チェックを実行します。各スタックメンバーは、アクティブスタックに互換性チェックの結果を送信し、その結果に基づいてスイッチがスイッチスタックに参加できるかどうかが判断されます。新しいスイッチ上のソフトウェアがスイッチ スタックと互換性がない場合は、新しいスイッチがバージョン不一致（VM）モードに入ります。

既存のスイッチスタックで自動アップグレード機能がイネーブルになっている場合は、アクティブスタックが、自動的に、互換性のあるスタックメンバー上で実行されているものと同じ

ソフトウェアイメージで新しいスイッチをアップグレードします。自動アップグレードは、一致しないソフトウェアが検出された数分後に起動します。

自動アップグレードには自動コピー プロセスと自動抽出プロセスが含まれます。

- 自動コピーは、スタック メンバー上で実行しているソフトウェアイメージを新しいスイッチに自動的にコピーして、そのスイッチをアップグレードします。また、自動コピーは、自動アップグレードがイネーブルになっている場合、新しいスイッチ上に十分なフラッシュ メモリが存在する場合、およびスイッチ スタック上で実行しているソフトウェア イメージが新しいスイッチに適合する場合に実行されます。

（注） VMモードのスイッチでは、すべてのリリース済みのソフトウェアが稼働するとは限りません。たとえば、新しいスイッチ ハードウェアは以前のバージョンのソフトウェアでは認識されません。

自動アップグレードプロセスが完了すると、新しいスイッチがリロードして、完全に機能するメンバーとしてスタックに参加します。リロード時に両方のスタック ケーブルが接続されていれば、スイッチスタックが2つのリング上で動作するため、ネットワークのダウンタイムが発生しません。

スイッチ スタックの管理接続

スイッチスタックおよびスタック メンバ インターフェイスは、アクティブスイッチを経由して管理します。CLI、SNMP、およびサポートされているネットワーク管理アプリケーションを使用できます。個別のデバイスごとにスタックメンバーを管理することはできません。

特定のスタック メンバーへの接続

特定のスタック メンバ ポートを設定する場合は、CLI コマンドインターフェイス表記にスタック メンバ番号を含めてください。

IP アドレスによるスイッチ スタックへの接続

スイッチ スタックは、単一 IP アドレスを介して管理されます。IP アドレスは、システムレベル設定であり、アクティブスタックやその他のスタックメンバー固有ではありません。スタックからアクティブスタックまたはその他のスタックメンバーを削除しても IP 接続があれば、そのまま同じ IP アドレスを使用してスタックを管理できます。

（注） スイッチ スタックからスタック メンバーを削除した場合、各スタック メンバーは自身の IP アドレスを保持します。したがって、ネットワーク内で同じ IP アドレスを持つ 2 つのデバイスが競合するのを避けるため、スイッチスタックから削除したアクティブスタックの IP アドレスを変更しておきます。

スイッチ スタック設定の関連情報については、「スイッチ スタックのコンフィギュレーション ファイル」のセクションを参照してください。

コンソール ポートによるスイッチ スタックへの接続

1つまたは複数のスタックメンバーのコンソールポートを経由して、端末またはPCをアクティブスイッチに接続することで、アクティブスイッチに接続できます。

スタックメンバーのコンソールポートを使用すると、192.168.0.1/24 サブネットの IP アドレスで VTY セッションが作成されます。

アクティブスイッチへの複数の CLI セッションを使用する場合は注意が必要です。1 つのセッションで入力したコマンドは、別のセッションには表示されません。そのため、コマンドを入力したセッションを識別できなくなることがあります。

スイッチ スタックを管理する場合は、1 つの CLI セッションだけを使用することを推奨します。

スイッチ スタックの設定方法

スタック ポートとしてのネットワーク ポートの設定

10G ネットワーク ポートと multigig ポートの両方をスタック ポートとして設定したり、1 つのポートをスタック ポートとして設定したり、別のポートをネットワーク ポートとして保持することができます。

手順の概要

1. **enable**
2. **configure terminal**
3. **switch** *switch-number* **hstack-port** *stack-port*
4. **end**
5. **show switch horizontal-stack-ports**
6. **copy running-config startup-config**
7. **reload**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**switch** *switch-number* **hstack-port** *stack-port* 例： `スイッチ(config)# switch 1 hstack-port 1 TenGigabitEthernet 1/0/1`	ネットワーク ポートをスタック ポートに設定します。 （注） 設定した後、ネットワーク ポートがスタック ポートになるようにスイッチを再起動します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show switch horizontal-stack-ports** 例： `スイッチ# show switch hstack-ports`	ネットワーク ポートとスタック ポートの動作ステータスを確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。
ステップ 7	**reload** 例： `スイッチ# reload`	設定をリロードします。

次のタスク

スタックポートをネットワークポートに変換するには、 **no switch** *switch-number* **hstack-port** *stack-port* コマンドを実行します。

```
スイッチ(config)# no switch 1 hstack-port 1 TenGigabitEthernet 1/0/1
スイッチ# copy running-config startup-config
スイッチ# reload
```

（注） 設定した後、スイッチを再起動してスタック ポートをネットワーク ポートに変換します。

（注） スタック ポートからネットワーク ポートへの変換、またはその逆の変換のCLIは、NVGEN処理されません。書き込み消去リロードでは、スタック モードのスイッチはスタンドアロンに変換されず、スタック ポートをネットワーク ポートに手動で変換する必要があります。

永続的 MAC アドレス機能のイネーブル化

（注） この機能を設定するためにコマンドを入力すると、設定の結果を記述した警告メッセージが表示されます。この機能は慎重に使用してください。古いアクティブスイッチのMACアドレスを同じドメイン内で使用すると、トラフィックが失われることがあります。

永続 MAC アドレスをイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **stack-mac persistent timer** [**0** | *time-value*]
4. **end**
5. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**stack-mac persistent timer** [**0** | *time-value*] 例： `Device(config)# stack-mac persistent timer 7`	スタックのアクティブスイッチが変更された後、スタック MAC アドレスが新しい ac の MAC アドレスに変更されるまでの遅延時間をイネーブルにします。この間に以前のアクティブ スイッチがスタックに再加入した場合、スタックはその MAC アドレスをスタック MAC アドレスとして使用します。 時間は 0 ～ 60 分の範囲で指定できます。 • 約 4 分というデフォルトの遅延を設定するには、値を指定しないでコマンドを入力します。必ず値を入力することを推奨します。 値を指定しないでコマンドを入力すると、実行コンフィギュレーションファイルには、遅延時間は明示タイマー値 4 分として書き込まれます。 • 現在のアクティブスイッチの MAC アドレスを無期限に使用し続けるには、**0** を入力します。 スタック MAC アドレスを現在のアクティブスイッチの MAC アドレスにただちに変更するための **no stack-mac persistent timer** コマンドを入力するまで、前のアクティブスイッチのスタック MAC アドレスが使用されます。 • スタック MAC アドレスが新しいアクティブ スイッチの MAC アドレスに変更されるまでの時間を設定するには、*time-value* に 1 ～ 60 分の範囲内の値を入力します。 設定された時間が過ぎるまで、または **no stack-mac persistent timer** コマンドを入力するまで、以前のアクティブスイッチのスタック MAC アドレスが使用されます。 （注） 新しいアクティブスイッチが引き継いだ後、時間切れになる前に **no stack-mac persistent timer** コマンドを入力した場合、スイッチスタックは現在のアクティブスイッチの MAC アドレスに移行します。
ステップ 4	**end** 例：	特権 EXEC モードに戻ります。

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

次のタスク

永続的 MAC アドレス機能をディセーブルにするには、**no stack-mac persistent timer** グローバル コンフィギュレーション コマンドを使用します。

スタック メンバー番号の割り当て

このオプションタスクは、アクティブスタックのみから使用できます。

メンバー番号をスタック メンバーに割り当てるには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **switch** *current-stack-member-number* **renumber** *new-stack-member-number*
4. **end**
5. **reload slot** *stack-member-number*
6. **show switch**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**switch** *current-stack-member-number* **renumber** *new-stack-member-number* 例： スイッチ(config)# **switch 3 renumber 4**	スタック メンバの現在のスタック メンバ番号と新たなスタック メンバ番号を指定します。指定できる範囲は 1 ～ 8 です。 **show switch** ユーザー EXEC コマンドを使用すると、現在のスタックメンバー番号を表示できます。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**reload slot** *stack-member-number* 例： スイッチ# **reload slot 4**	スタック メンバをリセットします。
ステップ 6	**show switch** 例： **show**スイッチ	スタック メンバ番号を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

スタック メンバー プライオリティ値の設定

このオプションタスクは、アクティブスタックのみから使用できます。

プライオリティ値をスタック メンバーに割り当てるには、次の手順を実行します。

手順の概要

1. **enable**
2. **switch** *stack-member-number* **priority** *new-priority-number*
3. **show switch** *stack-member-number*
4. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ **enable**	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**switch** *stack-member-number* **priority** *new-priority-number* 例： スイッチ# **switch 3 priority 2**	スタック メンバのスタック メンバ番号と、新しいプライオリティを指定します。スタック メンバ番号の有効範囲は 1 ～ 8 です。プライオリティ値の範囲は 1 ～ 15 です。 **show switch** ユーザー EXEC コマンドを使用して、現在のプライオリティ値を表示できます。 新しいプライオリティ値はすぐに有効となりますが、現在のアクティブスタックには影響しません。新たなプライオリティ値は、現在のアクティブスタックまたはスイッチスタックのリセット時に、どのスタックメンバーが新たなアクティブスタックとして選択されるかを決定する場合に影響を及ぼします。
ステップ 3	**show switch** *stack-member-number* 例： スイッチ# **show switch**	スタック メンバープライオリティ値を確認します。
ステップ 4	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

スイッチ スタックへの新しいメンバーのプロビジョニング

このオプション タスクは、アクティブ スイッチのみから使用できます。

手順の概要

1. **show switch**
2. **configure terminal**
3. **switch** *stack-member-number* **provision** *type*
4. **end**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**show switch** 例： スイッチ# **show switch**	スイッチ スタックに関する要約情報を表示します。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**switch** *stack-member-number* **provision** *type* 例： スイッチ(config)# **switch 3 provision WS-xxxx**	事前に設定されたスイッチのスタック メンバー番号を指定します。デフォルトでは、スイッチはプロビジョニングされません。 *Stack-member-number* の範囲は 1 ～ 8 です。スイッチ スタック内でまだ使用されていないスタック メンバー番号を指定します。ステップ 1 を参照してください。 *Type* には、コマンドライン ヘルプ ストリングに示されたサポート対象のスイッチのモデル番号を入力します。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

プロビジョニングされたスイッチ情報の削除

開始する前に、スタックから割り当てられたスイッチを削除する必要があります。このオプションタスクは、アクティブスタックのみから使用できます。

手順の概要

1. **configure terminal**
2. **no switch** *stack-member-number* **provision**
3. **end**

4. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**no switch** *stack-member-number* **provision** 例： スイッチ(config)# **no switch 3 provision**	指定されたメンバーの割り当て情報を削除します。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

例

次のように設定されたスタック内の割り当てられたスイッチを削除する場合：

- スタックは 4 つのメンバーを持つ
- スタックメンバー 1 がアクティブスタックである
- スタック メンバー 3 が割り当てられたスイッチである

さらに、割り当てられた情報を削除し、エラーメッセージを受信しないようにするには、スタックメンバー 3 の電源を切り、スタックメンバー 3 とそれが接続されているスイッチとの間のケーブルを抜き、そのケーブルを別のメンバー間に再接続して、**no switch** *stack-member-number* **provision** グローバル コンフィギュレーション コマンドを入力します。

スイッチ スタックのトラブルシューティング

スタック ポートの一時的なディセーブル化

スタックポートがフラッピングしていることが原因で、スタックリングが不安定になるためにポートをディセーブルにするには、**switch** *stack-member-number* **stack port** *port-number* **disable** 特権 EXEC コマンドを入力します。ポートを再びイネーブルにするには、**switch** *stack-member-number* **stack port** *port-number* **enable** コマンドを入力します。

（注） **switch** *stack-member-number* **stack port** *port-number* **disable** コマンドを使用するときは注意してください。スタック ポートをディセーブルにすると、スタックは半分の帯域幅で稼働します。

スタック ポートを通じてすべてのメンバーが接続されており、準備完了状態であれば、スタックはフルリング状態です。

次の現象が発生すると、スタックが部分リング状態になります。

- すべてのメンバがスタック ポートを通じて接続されたが、一部が ready ステートではない。
- スタック ポートを通じて接続されていないメンバーがある。

手順の概要

1. **switch** *stack-member-number* **stack port** *port-number* **disable**
2. **switch** *stack-member-number* **stack port** *port-number* **enable**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**switch** *stack-member-number* **stack port** *port-number* **disable** 例： `スイッチ# switch 2 stack port 1 disable`	指定されたポートをディセーブルにします。
ステップ 2	**switch** *stack-member-number* **stack port** *port-number* **enable** 例： `スイッチ# switch 2 stack port 1 enable`	スタック ポートを再びイネーブルにします。

スタックがフルリング状態のときにスタック ポートをディセーブルにしようとする場合は、1つのスタック ポートしかディセーブルにすることができません。次のメッセージが表示されます。

```
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
```

スタックが部分リング状態のときにスタック ポートをディセーブルにしようとしても、そのポートをディセーブルにすることができません。次のメッセージが表示されます。

```
Disabling stack port not allowed with current stack configuration.
```

他のメンバーの起動中のスタック ポートの再イネーブル化

スイッチ 1 のポート 1 がスイッチ 4 のポート 2 に接続されています。ポート 1 でフラッピングが発生した場合は、**switch 1 stack port 1 disable** 特権 EXEC コマンドを使用してポート 1 をディセーブルにできます。 スイッチ 1 のポート 1 がディセーブルになっており、スイッチ 1 の電源がまだオンになっている状態でスタック ポートを再びイネーブルにするには、次の手順を実行します。

ステップ 1 スイッチ 1 のポート 1 とスイッチ 4 のポート 2 の間のスタック ケーブルを取り外します。

ステップ 2 スタックからスイッチ 4 を取り外します。

ステップ 3 スイッチを追加してスイッチ 4 を交換し、スイッチ番号 4 を割り当てます。

ステップ 4 スイッチ 1 のポート 1 とスイッチ 4（交換後のスイッチ）のポート 2 の間のケーブルを再接続します。

ステップ 5 スイッチ間のリンクを再びイネーブルにします。**switch 1 stack port 1 enable** 特権 EXEC コマンドを入力して、スイッチ 1 のポート 1 をイネーブルにします。

ステップ 6 スイッチ 4 の電源を入れます。

⚠

注意 スイッチ 1 のポート 1 をイネーブルにする前にスイッチ 4 の電源を入れると、スイッチのいずれかがリロードされる場合があります。

スイッチ 4 の電源を最初に入れた場合は、リンクを確立するために、**switch 1 stack port 1 enable** および **switch 4 stack port 2 enable** 特権 EXEC コマンドの入力が必要になる場合があります。

デバイス スタックのモニターリング

表 *65 :* スタック情報を表示するコマンド

コマンド	説明
show switch	割り当てられたスイッチやバージョン不一致モードのスイッチのステータスなど、スタックに関するサマリー情報を表示します。
show switch *stack-member-number*	特定のメンバーに関する情報を表示します。
show switch detail	スタックに関する詳細情報を表示します。
show switch neighbors	スタック ネイバーを表示します。
show switch stack-ports	スタックのポート情報を表示します。

スイッチ スタックの設定例

スイッチ スタックの設定のシナリオ

これらのスイッチ スタック設定シナリオのほとんどが、少なくとも 2 つのdeviceが ポート経由で接続されていることを前提とします。

表 *66 :* 設定シナリオ

シナリオ		結果
既存のアクティブスイッチによって明確に決定されるアクティブスイッチ選択	ポート経由で 2 つの電源の入ったスイッチ スタックを接続します。	2 つのアクティブスイッチのうち 1 つだけが新しいアクティブスイッチになります。

シナリオ		結果
スタックメンバーのプライオリティ値によって明確に決定されるアクティブスイッチ選択	1. ポート経由で2つのスイッチを接続します。 2. **switch***stack-member-number* **priority** *new-priority-number*global configuration コマンドを使用して、一方のスタックメンバーにより高いメンバープライオリティ値を設定します。 3. 両方のスタックメンバーを同時に再起動します。	より高いプライオリティ値を持つスタックメンバーがアクティブスイッチに選択されます。
コンフィギュレーション ファイルによって明確に決定されるアクティブスイッチ選択	両方のスタック メンバーが同じプライオリティ値を持つものと仮定します。 1. 一方つのスタック メンバーがデフォルトのコンフィギュレーションを持ち、他方のスタック メンバーが保存済み（デフォルトでない）のコンフィギュレーション ファイルを持つことを確認します。 2. 両方のスタックメンバーを同時に再起動します。	保存済みのコンフィギュレーション ファイルを持つスタックメンバーがアクティブスイッチに選択されます。
MAC アドレスによって明確に決定されるアクティブスイッチ選択	両方のスタック メンバーが同じプライオリティ値、コンフィギュレーションファイル、フィーチャセットを持っていると仮定して、両方のスタックメンバーを同時に再起動します。	MAC アドレスが小さい方のスタックメンバーがアクティブスイッチに選択されます。

シナリオ		結果
スタック メンバー番号の競合	一方のスタック メンバーが他方のスタック メンバーより高いプライオリティ値を持つものと仮定します。 **1.** 両方のスタック メンバーが同じスタック メンバー番号を持つように確認します。必要に応じて、**switch** *current-stack-member-number* **renumber** *new-stack-member-number* global configuration コマンドを使用します。 **2.** 両方のスタック メンバーを同時に再起動します。	より高いプライオリティ値を持つスタック メンバーが、自分のスタック メンバー番号を保持します。もう一方のスタック メンバーは、新たなスタック メンバー番号を持ちます。
スタック メンバーの追加	**1.** 新しいスイッチの電源を切ります。 **2.** ポート経由で、新しいスイッチを電源の入ったスイッチ スタックに接続します。 **3.** 新しいスイッチの電源を入れます。	アクティブスイッチは保持されます。新たなスイッチがスイッチ スタックに追加されます。
アクティブスイッチの障害	アクティブスイッチを取り外します（または電源をオフにします）。	残りのスタックメンバーの 1 つが新しいアクティブスイッチになります。スタック内の他のすべてのスタック メンバーは、スタック メンバーのままで、再起動はされません。
8 台を超えるスタック メンバーの追加	**1.** ポート経由で、9台のdeviceを接続します。 **2.** すべてのdeviceの電源をオンにします。	2 つのdeviceがアクティブスイッチになります。1つのアクティブスイッチには、8つのスタックメンバーがあります。もう一方のアクティブスイッチはスタンドアロンdeviceとして維持されます。 アクティブスイッチのdeviceとそれぞれのアクティブスイッチに属しているdeviceを識別するには、device上の Mode ボタンとポート LED を使用します。

永続的 MAC アドレス機能のイネーブル化：例

次に、永続的 MAC アドレス機能に 7 分の遅延時間を設定し、設定を確認する例を示します。

```
スイッチ(config)# stack-mac persistent timer 7
  WARNING: The stack continues to use the base MAC of the old Master
  WARNING: as the stack MAC after a master switchover until the MAC
  WARNING: persistency timer expires. During this time the Network
  WARNING: Administrators must make sure that the old stack-mac does
  WARNING: not appear elsewhere in this network domain. If it does,
  WARNING: user traffic may be blackholed.
  スイッチ(config)# end
  スイッチ# show switch
  Switch/Stack Mac Address : 0016.4727.a900
  Mac persistency wait time: 7 mins
                                                H/W   Current
  Switch#  Role   Mac Address     Priority Version  State
  -----------------------------------------------------------
  *1       Master 0016.4727.a900     1      P2B     Ready
```

スイッチ スタックへの新しいメンバーの割り当て：例

次に、スタック メンバー番号 2 が設定されたスイッチをスイッチ スタックに割り当てる例を示します。**show running-config** コマンドの出力は、プロビジョニングされたスイッチに関連付けられたインターフェイスを示します。

スタック ポートへのネットワーク ポートの設定：例

次の例は、ネットワーク ポートをスタック ポートに変換する方法を示しています。

```
スイッチ> enable
スイッチ#configure terminal
スイッチ(config)#switch 1 hstack-port 1 TenGigabitEthernet 1/0/1
Do you want to continue?[confirm]
New port setting will be effective after next reload


スイッチ(config)#switch 1 hstack-port 2 TenGigabitEthernet 1/0/2
Do you want to continue?[confirm]
New port setting will be effective after next reload
```

次の出力例は、ネットワーク ポートからスタック ポートへのリロード前のポートのステータスを示します。

```
スイッチ#show switch hstack-ports
 Horizontal stack port status :
 Te Ports   Stack Port   Operational Status   Next Reload Status   Media Type
 --------- ------------ -------------------- -------------------- --------------
 Te1/0/1      1           N/W Port             Stack Port            Fiber
 Te1/0/2      2           N/W Port             Stack Port            Fiber
```

次の出力例は、ネットワーク ポートからスタック ポートへのリロード後のポートのステータスを示します。

```
スイッチ#show switch hstack-ports
 Horizontal stack port status :
 Te Ports   Stack Port   Operational Status   Next Reload Status   Media Type
 --------- ------------ -------------------- -------------------- --------------
 Te1/0/1      1           Stack Port           Stack Port          Fiber
 Te1/0/2      2           Stack Port           Stack Port          Fiber
```

次の例に、スタック ポートをネットワーク ポートに戻す方法を示します。

```
スイッチ> enable
スイッチ#configure terminal
スイッチ(config)#no switch 1 hstack-port 1
Do you want to continue?[confirm]
New port setting will be effective after next reload
```

次の出力例は、スタック ポートからネットワーク ポートへのリロード前のポートのステータスを示します。

```
スイッチ#show switch hstack-ports
 Horizontal stack port status :
 Te Ports   Stack Port   Operational Status   Next Reload Status   Media Type
 --------- ------------ -------------------- -------------------- --------------
 Te1/0/1      1           Stack Port           N/W Port            Fiber
 Te1/0/2      2           Stack Port           Stack Port          Fiber
```

次の出力例は、スタック ポートからネットワーク ポートへのリロード後のポートのステータスを示します。

```
スイッチ#show switch hstack-ports
 Horizontal stack port status :
 Te Ports   Stack Port   Operational Status   Next Reload Status   Media Type
 --------- ------------ -------------------- -------------------- --------------
 Te1/0/1      1           N/W Port             N/W Port            Fiber
 Te1/0/2      2           Stack Port           Stack Port          Fiber
```

次の出力例は、水平スタック ポートのステータスを示します。

```
スイッチ# show switch hstack-ports
 Horizontal stack port status :
 Te Ports   Stack Port   Operational Status   Next Reload Status   Media Type
 --------- ------------ -------------------- -------------------- --------------
 Te1/0/1      1            Stack Port         Stack Port          Fiber
 Te1/0/2      2            Stack Port         Stack Port          Fiber
 Te2/0/1      1            Stack Port         Stack Port          Fiber
 Te2/0/2      2            Stack Port         Stack Port          Fiber
 Te3/0/1      1            Stack Port         Stack Port          Copper
 Te3/0/2      NA           N/W Port           N/W Port            Copper
 Te3/0/3      2            Stack Port         Stack Port          Fiber
 Te3/0/4      NA           N/W Port           N/W Port            Fiber
 Te4/0/1      NA           N/W Port           N/W Port            Copper
 Te4/0/2      1            Stack Port         Stack Port          Copper
 Te4/0/3      2            Stack Port         Stack Port          Fiber
 Te4/0/4      NA           N/W Port           N/W Port            Fiber
 Te5/0/1      1            Stack Port         Stack Port          Fiber
 Te5/0/2      2            Stack Port         Stack Port          Fiber
 Te6/0/1      1            Stack Port         Stack Port          Fiber
 Te6/0/2      2            Stack Port         Stack Port          Fiber
 Te7/0/1      1            Stack Port         Stack Port          Copper
 Te7/0/2      NA           N/W Port           N/W Port            Copper
 Te7/0/3      2            Stack Port         Stack Port          Fiber
 Te7/0/4      NA           N/W Port           N/W Port            Fiber
 Te8/0/1      NA           N/W Port           N/W Port            Copper
 Te8/0/2      1            Stack Port         Stack Port          Copper
```

```
Te8/0/3      2           Stack Port        N/W Port          Fiber
Te8/0/4      NA          N/W Port          N/W Port          Fiber
```

スイッチ スタックに関する追加情報

関連資料

関連項目	マニュアルタイトル
スイッチ スタックのケーブル配線と電源供給。	

エラー メッセージ デコーダ

説明	リンク
このリリースのシステムエラーメッセージを調査し解決するために、エラー メッセージ デコーダ ツールを使用します。	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

標準および RFC

標準/RFC	タイトル
なし	—

MIB

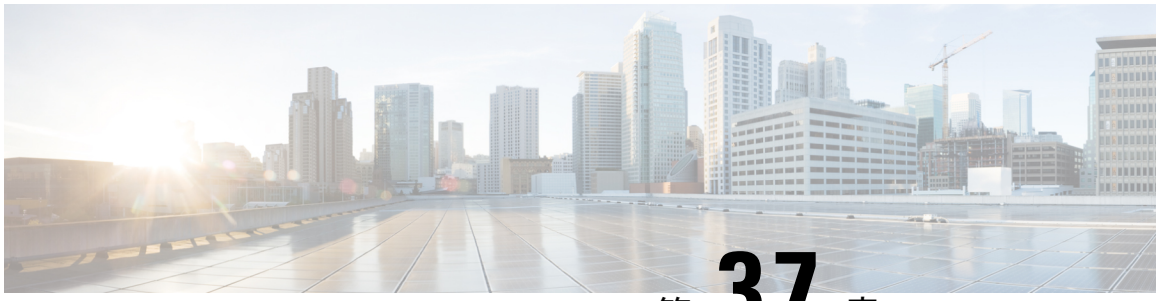
MIB	MIB のリンク
本リリースでサポートするすべての MIB	選択したプラットフォーム、Cisco IOS リリース、およびに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

シスコのテクニカル サポート

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

第 VI 部

ネットワーク管理



第 37 章

Cisco IOS Configuration Engine の設定

Configuration Engine を設定するための前提条件

- ユーザが接続している Configuration Engine インスタンスの名前を取得します。
- CNS は、イベント バスとコンフィギュレーション サーバーの両方を使用してデバイスに設定を提供するので、設定済みのdeviceごとに ConfigID と DeviceID の両方を定義する必要があります。
- **cns config partial** グローバル コンフィギュレーション コマンドを使用して設定されたすべての devices は、イベントバスにアクセスする必要があります。したがって、（device を起源とする）DeviceID が、Configuration Engine 内の対応する device 定義の DeviceID と一致する必要があります。ユーザが接続しているイベントバスのホスト名を把握する必要があります。

Configuration Engine の設定に関する制約事項

- コンフィギュレーション サーバーの 1 つのインスタンスでは、設定済みの 2 つのdevicesが同じ ConfigID 値を共有できません。
- イベント バスの 1 つのインスタンスでは、設定済みの 2 つのdevicesが同じ DeviceID 値を共有できません。

Configuration Engine の設定について

Cisco Configuration Engine ソフトウェア

Cisco Configuration Engine は、ネットワーク管理ユーティリティ ソフトウェアで、ネットワーク デバイスおよびサービスの配置と管理を自動化するためのコンフィギュレーション サービスとして機能します。各 Cisco Configuration Engine は、シスコ デバイス（devicesとルータ）のグループとデバイスが提供するサービスを管理し設定を保存して、必要に応じて配信します。Cisco Configuration Engine は、デバイス固有のコンフィギュレーション変更を生成してデバイスに送信し、コンフィギュレーション変更を実行して結果をログに記録することにより、初期設定とコンフィギュレーションの更新を自動化します。

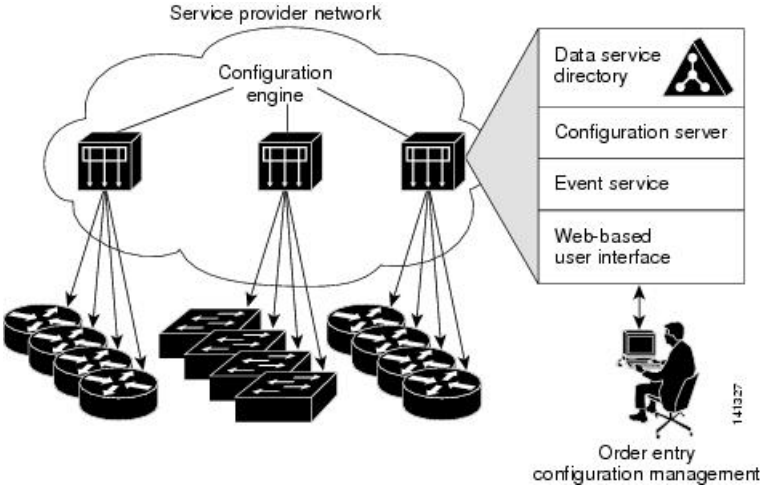
Cisco Configuration Engine は、スタンドアロン モードとサーバ モードをサポートし、次の Cisco Networking Service（CNS）コンポーネントがあります。

- コンフィギュレーション サービス
  - Web サーバ
  - ファイル マネージャ
  - ネームスペース マッピング サーバ
- イベント サービス（イベント ゲートウェイ）
- データ サービス ディレクトリ（データ モデルおよびスキーマ）

（注） Cisco Configuration Engine のサポートは、今後のリリースで廃止されます。『Cisco Plug and Play Feature Guide』に説明されている構成を使用してください。

スタンドアロン モードでは、内部に組み込まれたディレクトリ サービスがサポートされます。このモードでは、外部ディレクトリまたはその他のデータ ストアは必要ありません。サーバ モードでは、ユーザが定義した外部ディレクトリの使用がサポートされます。

図 *63 : Cisco Configuration Engine* のアーキテクチャの概要



コンフィギュレーション サービス

コンフィギュレーション サービスは、Cisco Configuration Engine の中核コンポーネントです。device上にある Cisco IOS CNS エージェントと連携して動作するコンフィギュレーション サーバーで構成されています。コンフィギュレーション サービスは、初期設定と論理グループによる大規模な再設定のために、デバイスとサービスの設定をdeviceに配信します。Devicesはネットワーク上で初めて起動するときに、コンフィギュレーション サービスから初期設定を受信します。

コンフィギュレーション サービスは CNS イベント サービスを使用して設定変更イベントを送受信し、成功および失敗の通知を送信します。

コンフィギュレーション サーバは Web サーバであり、コンフィギュレーション テンプレートと組み込み型ディレクトリ（スタンドアロン モード）またはリモート ディレクトリ（サーバ モード）に保存されているデバイス固有の設定情報を使用します。

コンフィギュレーションテンプレートは、CLI（コマンドライン インターフェイス）コマンド形式で静的な設定情報を含んだテキスト ファイルです。テンプレートでは、変数は、Lightweight Directory Access Protocol（LDAP）URL を使用して指定します。この URL はディレクトリに保存されているデバイス固有の設定情報を参照します。

Cisco IOS エージェントは受信したコンフィギュレーション ファイルの構文をチェックし、イベントを発行して構文チェックが成功または失敗したかを表示します。コンフィギュレーション エージェントは設定をただちに適用することも、あるいは同期化イベントをコンフィギュレーション サーバから受信するまで適用を遅らせることもできます。

イベント サービス

Cisco Configuration Engine は、設定イベントの受信および生成にイベント サービスを使用します。イベント サービスはイベント エージェント、イベント ゲートウェイから構成されます。

イベント エージェントはdevice上にあり、deviceと Cisco Configuration Engine のイベント ゲートウェイ間の通信を容易にします。

イベント サービスは、非常に有効なパブリッシュ サブスクライブ通信方式です。イベント サービスは、サブジェクトベースのアドレス指定を使用して、メッセージを宛先に送信します。サブジェクトベースのアドレス表記法では、メッセージおよび宛先には簡単で均一なネームスペースを定義します。

名前空間マッパー

Cisco Configuration Engine はネームスペース マッパー（NSM）を備えています。これは、アプリケーション、デバイスまたはグループ ID、およびイベントに基づいてデバイスの論理グループを管理するための検索サービスを提供します。

Cisco IOS デバイスは、たとえば cisco.cns.config.load といった、Cisco IOS ソフトウェアで設定されたサブジェクト名と一致するイベント サブジェクト名のみを認識します。ネームスペース マッピング サービスを使用すると、希望する命名規則を使用することでイベントを指定できます。サブジェクト名でデータ ストアにデータを入力した場合、NSM はイベント サブジェクト名ストリングを、Cisco IOS が認識するものに変更します。

サブスクライバの場合、一意のデバイス ID とイベントが指定されると、ネームスペース マッピング サービスは、サブスクライブ対象のイベント セットを返します。同様にパブリッシャの場合、一意のグループ ID、デバイス ID、およびイベントが指定されると、マッピング サービスは、パブリッシュ対象のイベント セットを返します。

Cisco Networking Service ID およびデバイスのホスト名

Cisco Configuration Engine は、設定対象の各deviceに一意の識別子が関連付けられていることを前提としています。一意の識別子は複数の同義語を持つことができますが、各同義語は特定のネームスペース内で一意です。イベント サービスは、ネームスペースの内容を使用してメッセージのサブジェクトベース アドレス指定を行います。

Cisco Configuration Engine は、イベント バス用とコンフィギュレーション サーバ用の 2 つの名前空間を交差します。コンフィギュレーション サーバのネームスペースでは、*ConfigID* という用語がデバイスの一意な識別子です。イベント バスのネームスペースでは、*DeviceID* という用語がデバイスの CNS 一意識別子です。

ConfigID

設定対象のdeviceはそれぞれ固有の ConfigID を持ちます。これは Cisco Configuration Engine ディレクトリからdevice CLI 属性の対応するセットを取得するためのキーとなります。deviceで定義された ConfigID は、Cisco Configuration Engine 上の対応するdevice定義の ConfigID と一致する必要があります。

ConfigID は起動時に固定され、device ホスト名を再設定した場合でもデバイスを再起動するまで変更できません。

DeviceID

イベントバスに参加している設定済みのdeviceごとに一意のDeviceIDがあります。これはdeviceの送信元アドレスに似ているので、deviceをバス上の特定の宛先として指定できます。

DeviceID の発信元は、deviceの Cisco IOS ホスト名によって定義されます。ただし、DeviceID 変数およびその使用は、deviceに隣接するイベント ゲートウェイ内にあります。

イベントバス上の Cisco IOS の論理上の終点は、イベント ゲートウェイに組み込まれ、それがdeviceの代わりにプロキシとして動作します。イベント ゲートウェイはイベント バスに対して、deviceおよび対応する DeviceID を表示します。

deviceは、イベント ゲートウェイとの接続が成功するとすぐに、そのホスト名をイベント ゲートウェイに宣言します。接続が確立されるたびに、イベント ゲートウェイは DeviceID 値を Cisco IOS ホスト名に組み合わせます。イベント ゲートウェイは、deviceと接続している間、この DeviceID 値を保持します。

ホスト名および DeviceID

DeviceID は、イベント ゲートウェイと接続したときに固定され、device ホスト名を再設定した場合でも変更されません。

deviceでdevice ホスト名を変更するとき、DeviceID を更新する唯一の方法は、deviceとイベント ゲートウェイ間の接続を切断することです。DeviceID 更新の手順については、以下の「関連項目」を参照してください。

接続が再確立されると、deviceは変更したホスト名をイベント ゲートウェイに送信します。イベント ゲートウェイは DeviceID を新しい値に再定義します。

⚠

注意 Cisco Configuration Engine ユーザー インターフェイスを使用するときは、最初に DeviceID フィールドを、device が前ではなく後に取得するホスト名値に設定する必要があります。Cisco IOS CNS エージェント用に設定を再初期化する必要があります。そのようにしないと、後続の部分的なコンフィギュレーション コマンド操作で誤動作が発生する可能性があります。

ホスト名、DeviceID、および ConfigID

スタンドアロン モードでは、ホスト名の値をdeviceに設定すると、コンフィギュレーション サーバーはイベントをホスト名に送信する場合、そのホスト名を DeviceID として使用します。ホスト名が設定されていない場合、イベントはデバイスの cn=<*value*>で送信されます。

サーバモードでは、ホスト名は使用されません。このモードでは、バス上のイベント送信には常に一意の DeviceID 属性が使用されます。この属性が設定されていない場合は device を更新できません。

Cisco Configuration Engine で **Setup** を実行する場合、これらの属性および関連する属性（タグ値のペア）を設定します。

Cisco IOS CNS エージェント

CNSイベントエージェント機能によって、deviceはイベントバス上でイベントにパブリッシュおよびサブスクライブを行い、Cisco IOS CNS エージェントと連携できます。device Cisco IOS ソフトウェアに組み込まれているこれらのエージェントでは、deviceを接続して、自動的に設定できます。

初期設定

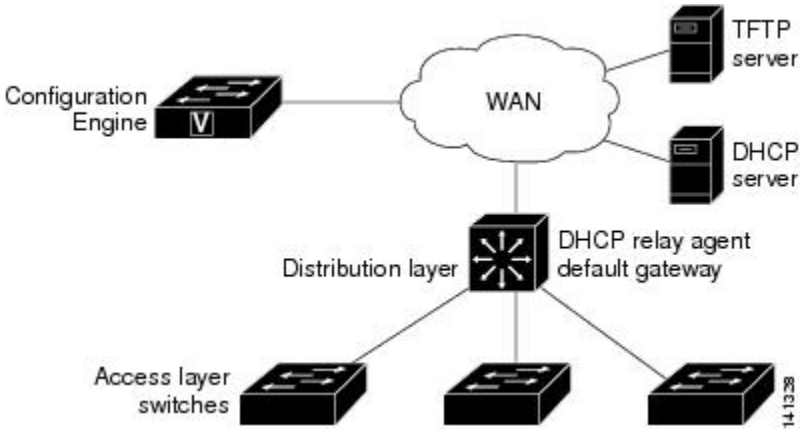
deviceが最初に起動すると、ネットワークで Dynamic Host Configuration Protocol（DHCP）要求をブロードキャストすることで IP アドレスを取得しようとします。サブネット上には DHCP サーバーがないものと想定し、ディストリビューション deviceは DHCP リレー エージェントとして動作し、要求を DHCP サーバーに転送します。DHCP サーバーは要求を受信すると、新しいdeviceに IP アドレスを割り当て、Trivial File Transfer Protocol（TFTP）サーバーのインターネット プロトコル（IP）アドレス、ブートストラップ コンフィギュレーション ファイルへのパス、デフォルト ゲートウェイの IP アドレスを、DHCP リレー エージェントに対するユニキャスト応答に組み入れます。DHCP リレー エージェントは、この応答をdeviceに転送します。

deviceは、割り当てられた IP アドレスを自動的にインターフェイス VLAN 1（デフォルト）に設定し、TFTP サーバーからブートストラップ コンフィギュレーション ファイルをダウンロードします。ブートストラップ コンフィギュレーション ファイルが正常にダウンロードされると、deviceはそのファイルを実行コンフィギュレーションにロードします。

Cisco IOS CNS エージェントは、該当する ConfigID および EventID を使用して Configuration Engine との通信を開始します。Configuration Engine はこの ConfigID をテンプレートにマッピングして、deviceに完全なコンフィギュレーション ファイルをダウンロードします。

次の図に、DHCPベースの自動設定を使用して初期ブートストラップ コンフィギュレーション ファイルを取得するためのネットワーク構成例を示します。

図 *64 :* 初期設定



差分（部分的）設定

ネットワークが稼働すると、Cisco IOS CNS エージェントを使用して新しいサービスを追加できます。差分（部分）設定は、deviceに送信できます。実際の設定を、イベント ペイロードとしてイベント ゲートウェイを介して（プッシュ処理）送信するか、deviceにプル オペレーションを開始させる信号イベントとして送信できます。

deviceは、適用する前に設定の構文をチェックできます。構文が正しい場合は、deviceは差分設定を適用し、コンフィギュレーション サーバーに成功を信号で伝えるイベントを発行します。deviceが差分設定を適用しない場合、エラー ステータスを示すイベントを発行します。deviceが差分設定を適用した場合、不揮発性RAM（NVRAM）に書き込むか、または書き込むように指示されるまで待つことができます。

コンフィギュレーションの同期

deviceは、設定を受信した場合、書き込み信号イベントの受信時に設定の適用を遅らせることができます。書き込み信号イベントは、更新された設定を NVRAM に保存しないようにdeviceに指示します。deviceは更新された設定を実行コンフィギュレーションとして使用します。これによりdeviceの設定は、次のリブート時の使用のために NVRAM に設定を保存する前に、他のネットワーク アクティビティと同期化されます。

自動 CNS 設定

deviceの自動CNS設定をイネーブルにするには、まずこのトピックに示す前提条件を完了する必要があります。条件設定を完了したらdeviceの電源を入れます。**setup** プロンプトでは何も入力しません。deviceが初期設定を開始します。コンフィギュレーション ファイル全体がdeviceにロードされると作業は完了です。

初期設定中の動作については、「関連項目」を参照してください。

表 *67*: 自動設定イネーブル化の条件

デバイス	必要な設定
アクセス device	出荷時の設定（コンフィギュレーション ファイルなし）
ディストリビューション device	• IP ヘルパー アドレス • DHCP リレー エージェントをイネーブルにする[2] • IPルーティング（デフォルト ゲートウェイとして使用する場合）
DHCP サーバ	• IP アドレスの割り当て • TFTP サーバの IP アドレス • TFTP サーバのブートストラップ コンフィギュレーション ファイルへのパス • デフォルト ゲートウェイの IP アドレス

デバイス	必要な設定
TFTP サーバ	• deviceと Configuration Engine との通信を可能にする CNS コンフィギュレーション コマンドを含むブートストラップ コンフィギュレーション ファイル • （デフォルトのホスト名の代わりに）device MAC アドレスまたはシリアル番号のいずれかを使用して ConfigID および EventID を生成するように設定されたdevice • にコンフィギュレーション ファイルをプッシュするように設定された CNS イベント エージェント device
CNS Configuration Engine	デバイス タイプ別の 1 つまたは複数のテンプレートで、テンプレートにデバイスの ConfigID がマッピングされています。

[2] DHCP リレーは、DHCP サーバがクライアントとは異なるサブネット上にある場合にのみ必要です。

Configuration Engine の設定方法

CNS イベント エージェントのイネーブル化

（注） device上で CNS イベント エージェントをイネーブルにしてから、CNS 設定エージェントをイネーブルにする必要があります。

device上で CNS イベントエージェントをイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **cns event** {*hostname* | *ip-address*} [*port-number*] [[**keepalive** *seconds retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*] | **backup**]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**cns event** {*hostname* | *ip-address*} [*port-number*] [[**keepalive** *seconds retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*] | **backup**] 例： スイッチ(config)# **cns event 10.180.1.27 keepalive 120 10**	イベントエージェントをイネーブルにして、ゲートウェイ パラメータを入力します。 • {*hostname* | *ip-address*} に、イベント ゲートウェイのホスト名またはIPアドレスを入力します。 • （任意）*port number* に、イベント ゲートウェイのポート番号を入力します。デフォルトのポート番号は 11011 です。 • （任意）**keepalive** *seconds* に、deviceがキープアライブメッセージを送信する間隔を入力します。*retry-count* に、キープアライブ メッセージへの応答がない場合に接続を終了するまでのdeviceのメッセージ送信回数を入力します。デフォルト値はいずれも 0 です。 • （任意）**failover-time** *seconds* に、バックアップ ゲートウェイが確立された後にdeviceがプライマリ ゲートウェイ ルートを待つ時間を入力します。 • （任意）**reconnect-time** *time* に、deviceがイベントゲートウェイに再接続しようとする前の最大時間間隔を入力します。 • （任意）バックアップゲートウェイであることを示す場合は、**backup** を入力します（省略した場合は、プライマリ ゲートウェイになります）。

	コマンドまたはアクション	目的
		（注） **encrypt** および **clock-timeout** *time* キーワードは、コマンドラインのヘルプストリングに表示されますが、サポートされていません。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

イベントエージェントに関する情報を確認するには、**show cns event connections** コマンドを特権 EXEC モードで使用します。

CNS イベントエージェントをディセーブルにするには、**no cns event** { *ip-address* | *hostname* } グローバル コンフィギュレーション コマンドを使用します。

Cisco IOS CNS エージェントのイネーブル化

device上で Cisco IOS CNS エージェントをイネーブルにするには、次の手順を実行します。

始める前に

このエージェントをイネーブルにする前に、deviceで CNS イベント エージェントをイネーブルにする必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **cns config initial** {*hostname* | *ip-address*} [*port-number*]
4. **cns config partial** {*hostname* | *ip-address*} [*port-number*]

5. **end**
6. **show running-config**
7. **copy running-config startup-config**
8. Cisco IOS CNS エージェントを、deviceで開始します。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**cns config initial** {*hostname* | *ip-address*} [*port-number*] 例： `スイッチ(config)# cns config initial 10.180.1.27 10`	Cisco IOS CNS エージェントをイネーブルにし、コンフィギュレーション サーバー パラメータを入力します。 • {*hostname* | *ip-address*} に、コンフィギュレーション サーバーのホスト名または IP アドレスを入力します。 • （任意）*port number* に、コンフィギュレーション サーバーのポート番号を入力します。 このコマンドがCisco IOS CNSエージェントをイネーブルにして、deviceで初期設定を開始します。
ステップ 4	**cns config partial** {*hostname* | *ip-address*} [*port-number*] 例： `スイッチ(config)# cns config partial 10.180.1.27 10`	Cisco IOS CNS エージェントをイネーブルにし、コンフィギュレーション サーバー パラメータを入力します。 • {*hostname* | *ip-address*} に、コンフィギュレーション サーバーのホスト名または IP アドレスを入力します。 • （任意）*port number* に、コンフィギュレーション サーバーのポート番号を入力します。 Cisco IOS CNS エージェントをイネーブルにして、deviceで部分的設定を開始します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。
ステップ 8	Cisco IOS CNS エージェントを、deviceで開始します。	

次のタスク

リモートで差分設定をdeviceに送信するために、Cisco Configuration Engine を使用できるようになりました。

Cisco IOS CNS エージェントの初期設定のイネーブル化

device上で、CNS コンフィギュレーション エージェントをイネーブルにして初期設定を開始するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **cns template connect** *name*
4. **cli** *config-text*
5. 別の CNS 接続テンプレートを設定する場合は、ステップ 3 ～ 4 を繰り返します。
6. **exit**
7. **cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]
8. **discover** { **controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}
9. **template** 名前 [*... name*]
10. ステップ 8 ～ 9 を繰り返し、CNS 接続プロファイルにさらに多くのインターフェイス パラメータと CNS 接続テンプレートを指定します。

11. **exit**
12. **hostname** *name*
13. **ip route** *network-number*
14. **cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]
15. **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]
16. **cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]
17. **end**
18. **show running-config**
19. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**cns template connect** *name* 例： `スイッチ(config)# cns template connect template-dhcp`	CNSテンプレート接続コンフィギュレーションモードを開始して、CNS 接続テンプレートの名前を指定します。
ステップ 4	**cli** *config-text* 例： `スイッチ(config-tmpl-conn)# cli ip address dhcp`	CNS 接続テンプレートにコマンドラインを入力します。テンプレート内の各コマンドラインにこの手順を繰り返します。
ステップ 5	別の CNS 接続テンプレートを設定する場合は、ステップ 3 ～ 4 を繰り返します。	
ステップ 6	**exit** 例： `スイッチ(config)# exit`	グローバル コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
ステップ 7	**cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*] 例： `スイッチ(config)# cns connect dhcp`	CNS接続コンフィギュレーションモードを開始し、CNS 接続プロファイルの名前を指定し、プロファイル パラメータを定義します。deviceは CNS 接続プロファイルを使用して Configuration Engine に接続します。 • CNS接続プロファイルの *name* を入力します。 •（任意）**retries** *number* に、接続のリトライ回数を入力します。指定できる範囲は 1 ～ 30 です。デフォルト値は 3 です。 •（任意）**retry-interval** *seconds* に、Configuration Engine への連続する接続の試行間隔を入力します。指定できる範囲は1～40秒です。デフォルトは 10 秒です。 •（任意）**sleep** *seconds* に、最初の接続試行を実行するまで待機する時間を入力します。指定できる範囲は 0 ～ 250 秒です。デフォルト値は 0 です。 •（任意）**timeout** *seconds* に、接続試行が終了するまでの時間を入力します。値の範囲は 10 ～ 2000 秒です。デフォルト値は 120 です。
ステップ 8	**discover** { **controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*} 例： `スイッチ(config-cns-conn)# discover interface gigabitethernet`	CNS 接続プロファイル内のインターフェイス パラメータを入力します。 • **controller** *controller-type* に、コントローラタイプを入力します。 • **dlci** に、アクティブなデータリンク接続識別子（DLCI）を入力します。 （任意）**subinterface** *subinterface-number* に、アクティブなDLCIの検索に使用するポイントツーポイント サブインターフェイス番号を指定します。 • **interface** [*interface-type*] に、インターフェイスのタイプを入力します。 • **line** *line-type* に、回線タイプを入力します。

	コマンドまたはアクション	目的
ステップ 9	**template** 名前 [... *name*] 例： `スイッチ(config-cns-conn)# template template-dhcp`	deviceの設定に適用する CNS 接続プロファイル内の CNS 接続テンプレートのリストを指定します。複数のテンプレートを指定できます。
ステップ 10	ステップ 8 ～ 9 を繰り返し、CNS 接続プロファイルにさらに多くのインターフェイス パラメータと CNS 接続テンプレートを指定します。	
ステップ 11	**exit** 例： `スイッチ(config-cns-conn)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 12	**hostname** *name* 例： `スイッチ(config)# hostname device1`	deviceのホスト名を入力します。
ステップ 13	**ip route** *network-number* 例： `Remoteスイッチ(config)# ip route 172.28.129.22 255.255.255.255 11.11.11.1`	（任意）IP アドレスが *network-number* の Configuration Engine へのスタティック ルートを確立します。
ステップ 14	**cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] 例： `Remoteスイッチ(config)# cns id GigabitEthernet0/1 ipaddress`	（任意）Configuration Engine が使用する一意の EventID または ConfigID を設定します。このコマンドを入力する場合は、**cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**] コマンドを入力しないでください。 • *interface num* に、インターフェイスのタイプを入力します。たとえば、ethernet、group-async、loopback、virtual-template を入力します。この設定では、一意の ID を定義するためにどのインターフェイスから IP アドレスまたは MAC アドレスを取得するかを指定します。 • {**dns-reverse** | **ipaddress** | **mac-address**} では、一意の ID として使用する値がホスト名の場合は **dns-reverse**、IP アドレスの場合は **ipaddress**、MAC アドレスの場合は **mac-address** を入力します。

	コマンドまたはアクション	目的
		• （任意）ID をdeviceの識別に使用する event-id 値になるように設定するには、**event** を入力します。 • （任意）ID をdeviceの識別に使用する image-id 値になるように設定するには、**image** を入力します。 （注） **event** と **image** キーワードの両方を省略した場合は、deviceの識別には image-id 値が使用されます。
ステップ 15	**cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**] 例： `Remoteスイッチ(config)# cns id hostname`	（任意）Configuration Engine が使用する一意の EventID または ConfigID を設定します。このコマンドを入力する場合は、**cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] コマンドを入力しないでください。 • { **hardware-serial** | **hostname** | **string** *string* | **udi** } には、deviceのシリアル番号を一意の ID として設定する場合は **hardware-serial** を入力し、deviceのホスト名を一意の ID として選択する場合は **hostname**（デフォルト）を入力します。また、**string** *string* が一意の ID の場合は任意のテキスト文字列を入力し、一意のデバイス識別子（UDI）を一意の ID として設定する場合は **udi** を入力します。
ステップ 16	**cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**] 例： `Remoteスイッチ(config)# cns config initial 10.1.1.1 no-persist`	Cisco IOS エージェントをイネーブルにして、初期設定を開始します。 • {*hostname* | *ip-address*} に、コンフィギュレーション サーバーのホスト名または IP アドレスを入力します。 • （任意）*port number* に、コンフィギュレーションサーバーのポート番号を入力します。デフォルトのポート番号は 80 です。 • （任意）設定が完了した際の設定の成功、失敗、または警告のメッセージ用に **event** をイネーブルにします。 • （任意）**cns config initial** グローバル コンフィギュレーション コマンドの入力結果によってプルされた設定の NVRAM への自動書き込み

	コマンドまたはアクション	目的
		を抑制するには、**no-persist** をイネーブルにします。**no-persist** キーワードを入力しない場合、**cns config initial** コマンドを使用すると、その結果の設定が自動的に NVRAM に書き込まれます。 • （任意）**page** *page* に、初期設定の Web ページを入力します。デフォルトは /Config/config/asp です。 • （任意）送信元 IP アドレスに使用するには、**source** *ip-address* を入力します。 • （任意）このパラメータを使用したときの構文をチェックするには、**syntax-check** をイネーブルにします。 （注） **encrypt**、**status** *url*、および **inventory** キーワードは、コマンドラインのヘルプストリングに表示されますが、サポートされていません。
ステップ 17	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 18	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 19	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

コンフィギュレーション エージェントに関する情報を確認するには、**show cns config connections** コマンドを特権 EXEC モードで使用します。

Cisco IOS エージェントをディセーブルにするには、**no cns config initial** { *ip-address* | *hostname* } グローバル コンフィギュレーション コマンドを使用します。

DeviceID の更新

device上でホスト名を変更するときに DeviceID を更新するには、次の手順を実行します。

手順の概要

1. **enable**
2. **show cns config connections**
3. CNS イベント エージェントがイベント ゲートウェイに正しく接続されていることを確認します。
4. **show cns event connections**
5. ステップ 4 の出力に基づいて、次に示す現在接続されている接続に関する情報を記録します。この手順の以降のステップで IP アドレスとポート番号を使用します。
6. **configure terminal**
7. **no cns event** *ip-address port-number*
8. **cns event** *ip-address port-number*
9. **end**
10. **show cns event connections** からの出力を調べて、deviceとイベント接続間の接続が再確立されていることを確認します。
11. **show running-config**
12. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show cns config connections** 例： `スイッチ# show cns config connections`	CNS イベント エージェントがゲートウェイに接続しているか、接続されているか、またはアクティブか、およびイベント エージェントに使用されているゲートウェイ、その IP アドレス、およびポート番号を表示します。
ステップ 3	CNS イベント エージェントがイベント ゲートウェイに正しく接続されていることを確認します。	次の点について、**show cns config connections** の出力調べます。 • 接続がアクティブになっている。

	コマンドまたはアクション	目的
		• 接続で現在設定されているdeviceホスト名を使用している。DeviceID はこれらの手順を使用して、新しいホスト名の設定に対応するように更新されます。
ステップ 4	**show cns event connections** 例： スイッチ# **show cns event connections**	deviceのイベント接続情報を表示します。
ステップ 5	ステップ 4 の出力に基づいて、次に示す現在接続されている接続に関する情報を記録します。この手順の以降のステップで IP アドレスとポート番号を使用します。	
ステップ 6	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 7	**no cns event** *ip-address port-number* 例： スイッチ(config)# **no cns event 172.28.129.22 2012**	このコマンドで、ステップ 5 で記録した IP アドレスとポート番号を指定します。 このコマンドで、deviceとイベント ゲートウェイ間の接続が解除されます。最初に接続を解除し、次にこの接続を再確立して、DeviceID を更新する必要があります。
ステップ 8	**cns event** *ip-address port-number* 例： スイッチ(config)# **cns event 172.28.129.22 2012**	このコマンドで、ステップ 5 で記録した IP アドレスとポート番号を指定します。 このコマンドで、deviceとイベント ゲートウェイ間の接続が再確立されます。
ステップ 9	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 10	**show cns event connections** からの出力を調べて、deviceとイベント接続間の接続が再確立されていることを確認します。	
ステップ 11	**show running-config** 例：	入力を確認します。

	コマンドまたはアクション	目的
	スイッチ# **show running-config**	
ステップ 12	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

Cisco IOS CNS エージェントの部分的設定のイネーブル化

device上で Cisco IOS CNS エージェントをイネーブルにして部分設定を開始するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*] 例： スイッチ(config)# **cns config partial 172.28.129.22 2013**	コンフィギュレーションエージェントをイネーブルにし、部分設定を開始します。 • {*ip-address* | *hostname*} に、コンフィギュレーション サーバーの IP アドレスまたはホスト名を入力します。

	コマンドまたはアクション	目的
		• （任意）*port number* に、コンフィギュレーション サーバーのポート番号を入力します。デフォルトのポート番号は 80 です。 • （任意）送信元 IP アドレスに使用するには、**source** *ip-address* を入力します。 （注） **encrypt** キーワードは、コマンドラインのヘルプ ストリングには表示されますが、サポートされていません。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

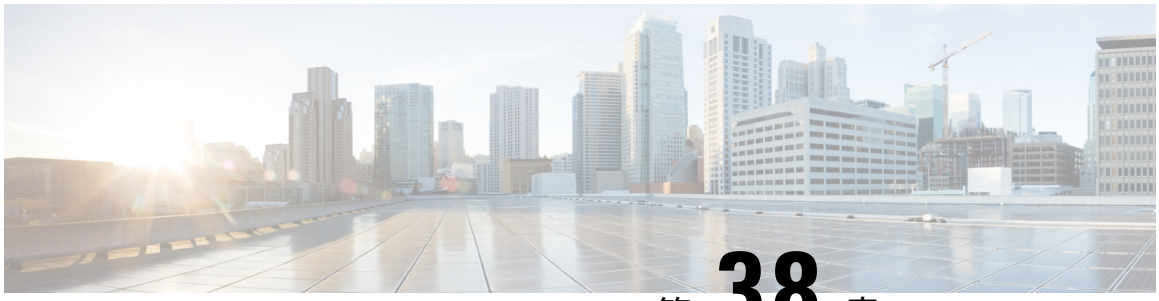
コンフィギュレーション エージェントに関する情報を確認するには、**show cns config stats** または **show cns config outstanding** コマンドを特権 EXEC モードで使用します。

Cisco IOS エージェントをディセーブルにするには、**no cns config partial** { *ip-address* | *hostname* } グローバル コンフィギュレーション コマンドを使用します。部分設定を取り消すには、**cns config cancel** グローバル コンフィギュレーション コマンドを使用します。

CNS 設定のモニタリング

表 *68 : CNS show* コマンド

コマンド	目的
show cns config connections スイッチ# **show cns config connections**	CNS Cisco IOS CNS エージェントの接続のステータスを表示します。
show cns config outstanding スイッチ# **show cns config outstanding**	開始されたがまだ終了していない差分（部分）CNS設定に関する情報を表示します。
show cns config stats スイッチ# **show cns config stats**	Cisco IOS CNS エージェントに関する統計情報を表示します。
show cns event connections スイッチ# **show cns event connections**	CNSイベントエージェントの接続のステータスを表示します。
show cns event gateway スイッチ# **show cns event gateway**	deviceのイベント ゲートウェイ情報を表示します。
show cns event stats スイッチ# **show cns event stats**	CNSイベントエージェントに関する統計情報を表示します。
show cns event subject スイッチ# **show cns event subject**	アプリケーションによってサブスクライブされたイベントエージェントのサブジェクト一覧を表示します。



第 38 章

Cisco Discovery Protocol の設定

Cisco Discovery Protocol は、シスコデバイス上で動作し、ネットワーキング アプリケーションが直接接続された付近のデバイスに関して学習できるようにする、メディア独立型かつネットワーク独立型のレイヤ 2 プロトコルです。このプロトコルによってシスコデバイスが検出されてその設定状態が特定され、異なるネットワーク層プロトコルを使用するシステムが相互に学習できるようになることで、デバイスの管理が容易になります。

このモジュールでは、Cisco Discovery Protocol バージョン 2 とその SNMP での動作について説明します。

CDP に関する情報

Cisco Discovery Protocol の概要

Cisco Discovery Protocol は、すべてのシスコデバイス（ルータ、ブリッジ、アクセスサーバ、コントローラ、およびスイッチ）のレイヤ 2（データリンク層）で動作するデバイス検出プロトコルです。ネットワーク管理アプリケーションは CDP を使用することにより、既知のデバイスのネイバーであるシスコデバイスを検出することができます。また、下位レイヤのトランスペアレント プロトコルが稼働しているネイバーデバイスのデバイスタイプや、SNMP エージェントアドレスを学習することもできます。この機能によって、アプリケーションからネイバー デバイスに SNMP クエリーを送信できます。

Cisco Discovery Protocol は、サブネットワーク アクセス プロトコル（SNAP）をサポートしているすべてのメディアで動作します。Cisco Discovery Protocol はデータリンク層でのみ動作するため、異なるネットワーク層プロトコルをサポートする 2 つのシステムで互いの情報を学習できます。

Cisco Discovery Protocol が設定された各デバイスはマルチキャストアドレスに定期的にメッセージを送信して、SNMP メッセージを受信可能なアドレスを 1 つまたは複数アドバタイズします。このアドバタイズには、受信側デバイスで Cisco Discovery Protocol 情報を廃棄せずに保持

する時間を表す存続可能時間、つまりホールドタイム情報も含まれます。各デバイスは他のデバイスから送信されたメッセージも待ち受けて、ネイバー デバイスについて学習します。

Cisco Discovery Protocol はdevice上で Network Assistant をイネーブルにすることで、ネットワークをグラフィカルに表示できます。deviceは Cisco Discovery Protocol を使用してクラスタ候補を検出し、クラスタメンバ、およびコマンドdeviceから最大 3 台（デフォルト）離れたクラスタ対応の他のデバイスについての情報を維持します。

- Cisco Discovery Protocol は、deviceと直接通信する接続されたエンドポイントを識別します。
- ネイバー デバイスのレポートが重複しないように、1 つの有線deviceだけがロケーション情報をレポートします。
- 有線deviceとエンドポイントは、ロケーションの送信と受信の両方を行います。

Cisco Discovery Protocol のデフォルト設定

次の表に、Cisco Discovery Protocol のデフォルト設定を示します。

機能	デフォルト設定
Cisco Discovery Protocol グローバル状態	イネーブル
Cisco Discovery Protocol インターフェイス状態	イネーブル
Cisco Discovery Protocol タイマー（パケット更新頻度）	60 秒
Cisco Discovery Protocol 保持時間（廃棄前）	180 秒
Cisco Discovery Protocol バージョン 2 アドバタイズメント	有効

CDP の設定方法

Cisco Discovery Protocol の特性の設定

次の Cisco Discovery Protocol の特性を設定できます。

- Cisco Discovery Protocol アップデートの頻度
- 破棄するまで情報を保持する時間の長さ
- バージョン 2 アドバタイズメントを送信するかどうか

（注） ステップ 3 ～ 5 はすべて任意であり、どの順番で実行してもかまいません。

次の手順に従って、Cisco Discovery Protocol の特性を設定します。

手順の概要

1. **enable**
2. **configure terminal**
3. **cdp timer** *seconds*
4. **cdp holdtime** *seconds*
5. **cdp advertise-v2**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**cdp timer** *seconds* 例： `スイッチ(config)# cdp timer 20`	（任意）Cisco Discovery Protocol 更新の送信頻度を秒単位で設定します。 指定できる範囲は 5 ～ 254 です。デフォルトは 60 秒です。
ステップ 4	**cdp holdtime** *seconds* 例： `スイッチ(config)# cdp holdtime 60`	（任意）受信デバイスがこのデバイスから送信された情報を破棄せずに保持する時間を指定します。 指定できる範囲は 10 ～ 255 秒です。デフォルトは 180 秒です。
ステップ 5	**cdp advertise-v2** 例： `スイッチ(config)# cdp advertise-v2`	（任意）バージョン 2 アドバタイズを送信するように Cisco Discovery Protocol を設定します。 これは、デフォルトの状態です。

	コマンドまたはアクション	目的
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

デフォルト設定に戻すには、Cisco Discovery Protocol コマンドの **no** 形式を使用します。

Cisco Discovery Protocol のディセーブル化

Cisco Discovery Protocol はデフォルトでイネーブルになっています。

（注） デバイス クラスタと他のシスコデバイス（Cisco IP Phone など）は、Cisco Discovery Protocol メッセージを定期的に交換します。Cisco Discovery Protocol をディセーブルにすると、クラスタ検出が中断され、デバイスの接続が切断されます。

Cisco Discovery Protocol デバイス検出機能をディセーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no cdp run** 例： スイッチ(config)# **no cdp run**	Cisco Discovery Protocol を無効にします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

Cisco Discovery Protocol を使用するには、再度有効にする必要があります。

Cisco Discovery Protocol の有効化

Cisco Discovery Protocol はデフォルトでイネーブルになっています。

（注） デバイスクラスタと他のシスコデバイス（Cisco IP Phone など）は、Cisco Discovery Protocol メッセージを定期的に交換します。Cisco Discovery Protocol をディセーブルにすると、クラスタ検出が中断され、デバイスの接続が切断されます。

ディセーブルになっている Cisco Discovery Protocol をイネーブルにするには、次の手順を実行します。

始める前に

Cisco Discovery Protocol がディセーブルになっていないと、イネーブルにはできません。

手順の概要

1. **enable**
2. **configure terminal**
3. **cdp run**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**cdp run** 例： `スイッチ(config)# cdp run`	Cisco Discovery Protocol がディセーブルになっている場合にイネーブルにします。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

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

次のタスク

Cisco Discovery Protocol がイネーブルになっていることを表示するには、 **show run all** コマンドを使用します。**show run** を入力しただけでは、Cisco Discovery Protocol がイネーブルになっていることが表示されない場合があります。

インターフェイス上で Cisco Discovery Protocol をディセーブルにします。

Cisco Discovery Protocol は、Cisco Discovery Protocol 情報を送受信するために、サポートされているすべてのインターフェイスでデフォルトで有効になっています。

（注） デバイスクラスタと他のシスコデバイス（Cisco IP Phone など）は、Cisco Discovery Protocol メッセージを定期的に交換します。Cisco Discovery Protocol をディセーブルにすると、クラスタ検出が中断され、デバイスの接続が切断されます。

（注） Discovery Protocol バイパスはサポートされていないため、ポートが err-disabled ステートになる場合があります。

ポートで Cisco Discovery Protocol をディセーブルにするには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no cdp enable**
5. **end**

6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	Cisco Discovery Protocol をディセーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**no cdp enable** 例： `スイッチ(config-if)# no cdp enable`	ステップ 3 で指定したインターフェイス上で Cisco Discovery Protocol をディセーブルにします。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

インターフェイス上での Cisco Discovery Protocol のイネーブル化

Cisco Discovery Protocol は、Cisco Discovery Protocol 情報を送受信するために、サポートされているすべてのインターフェイスでデフォルトで有効になっています。

（注） デバイス クラスタと他のシスコデバイス（Cisco IP Phone など）は、Cisco Discovery Protocol メッセージを定期的に交換します。Cisco Discovery Protocol をディセーブルにすると、クラスタ検出が中断され、デバイスの接続が切断されます。

（注） Discovery Protocol バイパスはサポートされていないため、ポートが err-disabled ステートになる場合があります。

ポートでディセーブルになっている Cisco Discovery Protocol をイネーブルにするには、次の手順を実行します。

始める前に

Cisco Discovery Protocol をイネーブルにしようとしているポートでは、Cisco Discovery Protocol がディセーブルになっている必要があります。そうでないと、イネーブルにできません。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **cdp enable**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

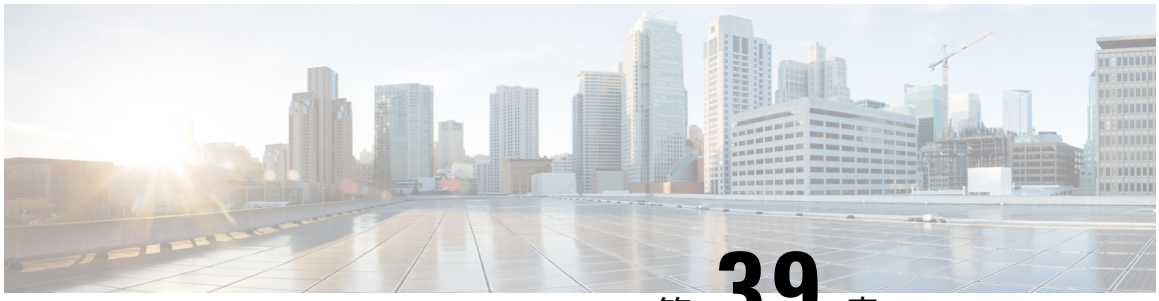
	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	Cisco Discovery Protocol をイネーブルにするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**cdp enable** 例： スイッチ(config-if)# **cdp enable**	ディセーブルになっているインターフェイスで Cisco Discovery Protocol をイネーブルにします。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

Cisco Discovery Protocol のモニタリングとメンテナンス

表 *69 : Cisco Discovery Protocol* 情報を表示するためのコマンド

コマンド	説明
clear cdp counters	トラフィック カウンタを 0 にリセットします。
clear cdp table	ネイバーに関する情報の Cisco Discovery Protocol テーブルを削除します。
show cdp	送信間隔、送信したパケットの保持時間などのグローバル情報を表示します。

コマンド	説明
show cdp entry *entry-name* ［**version**］［**protocol**］	特定のネイバーに関する情報を表示します。 アスタリスク（*）を入力して、すべてのCisco Discovery Protocol ネイバーを表示することも、情報が必要なネイバーの名前を入力することもできます。 また、指定されたネイバー上でイネーブルになっているプロトコルの情報や、デバイス上で稼働しているソフトウェアのバージョン情報が表示されるように、表示内容を制限することもできます。
show cdp interface［*interface-id*］	Cisco Discovery Protocol がイネーブルになっているインターフェイスに関する情報を表示します。 必要なインターフェイスの情報だけを表示できます。
show cdp neighbors［*interface-id*］［*detail*］	装置タイプ、インターフェイス タイプ、インターフェイス番号、保持時間の設定値、機能、プラットフォーム、ポート ID を含めたネイバー情報を表示します。 特定のインターフェイスに関するネイバー情報だけを表示したり、詳細表示にするため表示内容を拡張したりできます。
show cdp traffic	Cisco Discovery Protocol カウンタ（送信済み/受信済みパケット数とチェックサム エラー数を含む）を表示します。



第 39 章

簡易ネットワーク管理プロトコルの設定

SNMP の前提条件

サポートされている SNMP バージョン

このソフトウェア リリースは、次の SNMP バージョンをサポートしています。

- SNMPv1：RFC1157 に規定された SNMP（完全インターネット標準）。
- SNMPv2C は、SNMPv2Classic のバルク検索機能を残し、エラー処理を改善したうえで、SNMPv2Classic のパーティ ベースの管理およびセキュリティ フレームワークをコミュニティ ストリング ベースの管理フレームワークに置き換えたものです。次の機能があります。
  - SNMPv2：RFC 1902 ～ 1907 に規定された SNMP バージョン 2（ドラフト版インターネット標準）
  - SNMPv2C：RFC 1901 に規定された SNMPv2 のコミュニティ ストリング ベースの管理フレームワーク（試験版インターネット プロトコル）
- SNMPv3：SNMP のバージョン 3 は、RFC 2273 ～ 2275 に規定されている相互運用可能な標準ベース プロトコルです。SNMPv3 は、ネットワーク上のパケットを認証、暗号化することでデバイスへのアクセスに対するセキュリティを提供します。SNMPv3 は、次のセキュリティ機能を備えています。
  - メッセージの完全性：パケットが伝送中に改ざんされないようにします。
  - 認証：有効な送信元からのメッセージであるかどうかを判別します。

- 暗号化：パッケージの内容をミキシングし、許可されていない送信元に内容が読まれることを防止します。

（注） 暗号化を選択するには、**priv** キーワードを入力します。

SNMPv1 と SNMPv2C は、ともにコミュニティベース形式のセキュリティを使用します。エージェントの MIB にアクセスできるマネージャのコミュニティが、IP アドレス アクセス コントロール リストおよびパスワードによって定義されます。

SNMPv2C にはバルク検索機能が組み込まれ、より詳細なエラー メッセージを管理ステーションに報告します。バルク検索機能は、テーブルや大量の情報を検索し、必要な往復回数を削減します。SNMPv2C ではエラー処理機能が改善され、さまざまなエラーを区別するための拡張エラー コードが使用されています。これらのエラーは、SNMPv1 では単一のエラー コードで報告されます。SNMPv2 では、エラー リターン コードでエラー タイプが報告されるようになりました。

SNMPv3 では、セキュリティ モデルとセキュリティ レベルの両方が提供されています。セキュリティ モデルは、ユーザとユーザが属しているグループ用に設定された認証方式です。セキュリティ レベルとは、セキュリティ モデル内で許可されるセキュリティのレベルです。セキュリティ レベルとセキュリティ モデルの組み合わせにより、SNMP パケットを扱うときに使用するセキュリティ方式が決まります。使用可能なセキュリティ モデルは、SNMPv1、SNMPv2C、および SNMPv3 です。

次の表では、この特性を識別し、セキュリティ モデルとセキュリティ レベルの異なる組み合わせを比較します。

表 *70 : SNMP* セキュリティ モデルおよびセキュリティ レベル

モデル	レベル	認証	暗号化	結果
SNMPv1	noAuthNoPriv	コミュニティ ストリング	未対応	コミュニティ ストリングの照合を使用して認証します。
SNMPv2C	noAuthNoPriv	コミュニティ ストリング	未対応	コミュニティ ストリングの照合を使用して認証します。
SNMPv3	noAuthNoPriv	Username	未対応	ユーザ名の照合を使用して認証します。

モデル	レベル	認証	暗号化	結果
SNMPv3	authNoPriv	Message Digest 5（MD5）または Secure Hash Algorithm（SHA）	なし	HMAC-MD5 アルゴリズムまたは HMAC-SHA アルゴリズムに基づいて認証します。
SNMPv3	authPriv	MD5 または SHA	データ暗号規格（DES）または Advanced Encryption Standard（AES）	HMAC-MD5 アルゴリズムまたは HMAC-SHA アルゴリズムに基づいて認証します。 次の暗号化アルゴリズムで、User-based Security Model（USM）を指定できます。 • CBC-DES（DES-56）規格に基づく認証に加えた DES 56 ビット暗号化 • 3DES 168 ビット暗号化 • AES 128 ビット暗号化、192 ビット暗号化、または 256 ビット暗号化

管理ステーションでサポートされている SNMP バージョンを使用するには、SNMP エージェントを設定する必要があります。エージェントは複数のマネージャと通信できるため、SNMPv1、SNMPv2C、および SNMPv3 を使用する通信をサポートするようにソフトウェアを設定できます。

SNMP の制約事項

バージョンの制約事項

- SNMPv1 は informs をサポートしていません。

SNMP に関する情報

SNMP の概要

SNMP は、マネージャとエージェント間の通信のメッセージ フォーマットを提供するアプリケーションレイヤプロトコルです。SNMP システムは、SNMP マネージャ、SNMP エージェント、および管理情報ベース（MIB）で構成されます。SNMP マネージャは、Cisco Prime Infrastructure などのネットワーク管理システム（NMS）に統合できます。エージェントおよび MIB は、deviceに常駐します。device 上で SNMP を設定するには、マネージャとエージェント間の関係を定義します。

SNMP エージェントは MIB 変数を格納し、SNMP マネージャはこの変数の値を要求または変更できます。マネージャはエージェントから値を取得したり、エージェントに値を格納したりできます。エージェントは、デバイス パラメータやネットワーク データの保存場所である MIB から値を収集します。また、エージェントはマネージャのデータ取得またはデータ設定の要求に応答できます。

エージェントは非送信請求トラップをマネージャに送信できます。トラップは、ネットワーク上のある状態を SNMP マネージャに通知するメッセージです。トラップは不正なユーザ認証、再起動、リンク ステータス（アップまたはダウン）、MAC アドレス追跡、TCP 接続の終了、ネイバーとの接続の切断などの重要なイベントの発生を意味する場合があります。

SNMP マネージャ機能

SNMP マネージャは、MIB 情報を使用して、次の表に示す動作を実行します。

表 *71 : SNMP* の動作

動作	説明
get-request	特定の変数から値を取得します。
get-next-request	テーブル内の変数から値を取得します。[3]
get-bulk-request[4]	テーブルの複数の行など、通常はサイズの小さい多数のデータ ブロックに分割して送信する必要がある巨大なデータ ブロックを取得します。
get-response	NMS から送信される get-request、get-next-request、および set-request に対して応答します。
set-request	特定の変数に値を格納します。
trap	SNMP エージェントから SNMP マネージャに送られる、イベントの発生を伝える非送信請求メッセージです。

[3] この動作では、SNMP マネージャに正確な変数名を認識させる必要はありません。テーブル内を順に検索して、必要な変数を検出します。

[4] get-bulk コマンドを使用できるのは、SNMPv2 以上に限られます。

SNMP エージェント機能

SNMP エージェントは、次のようにして SNMP マネージャ要求に応答します。

- MIB 変数の取得：SNMP エージェントは NMS からの要求に応答して、この機能を開始します。エージェントは要求された MIB 変数の値を取得し、この値を使用して NMS に応答します。
- MIB 変数の設定：SNMP エージェントは NMS からのメッセージに応答して、この機能を開始します。SNMP エージェントは、MIB 変数の値を NMS から要求された値に変更します。

エージェントで重要なイベントが発生したことを NMS に通知するために、SNMP エージェントは非送信請求トラップ メッセージも送信します。トラップ条件の例には、ポートまたはモジュールがアップまたはダウン状態になった場合、スパニングツリー トポロジが変更された場合、認証に失敗した場合などがあります。

SNMP コミュニティ ストリング

SNMP コミュニティ ストリングは、MIB オブジェクトへのアクセスを認証し、組み込みパスワードとして機能します。NMS が device にアクセスするには、NMS 上のコミュニティ ストリング定義が device 上の 3 つのコミュニティ ストリング定義の少なくとも 1 つと一致しなければなりません。

コミュニティ ストリングの属性は、次のいずれかです。

- 読み取り専用（RO）：コミュニティ ストリングを除き MIB 内のすべてのオブジェクトに、許可された管理ステーションに対する読み取りアクセス権を与えますが、書き込みアクセスは許可しません。
- 読み取り-書き込み（RW）：MIB 内のすべてのオブジェクトに、許可された管理ステーションに対する読み取りおよび書き込みアクセス権を与えますが、コミュニティストリングへのアクセスは許可しません。
- クラスタを作成すると、コマンド deviceがメンバ devicesと SNMP アプリケーション間のメッセージ交換を管理します。Network Assistant ソフトウェアは、コマンド device上で最初に設定された RW および RO コミュニティ ストリングにメンバ device番号（@esN、N はdevice番号）を追加し、これらのストリングをメンバ devicesに伝播します。

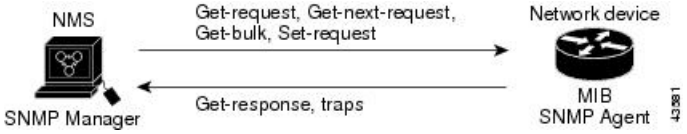
SNMP MIB 変数アクセス

NMS の例として、Cisco Prime Infrastructure ネットワーク管理ソフトウェアがあります。Cisco Prime Infrastructure ソフトウェアは、device MIB 変数を使用して装置変数を設定し、ネットワーク上の装置をポーリングして特定の情報を取得します。ポーリング結果は、グラフ形式で表示されます。この結果を解析して、インターネットワーキング関連の問題のトラブルシューティ

ング、ネットワーク パフォーマンスの改善、デバイス設定の確認、トラフィック負荷のモニタなどを行うことができます。

次の図に示すように、SNMP エージェントは MIB からデータを収集します。エージェントは SNMP マネージャに対し、トラップ（特定イベントの通知）を送信でき、SNMP マネージャはトラップを受信して処理します。トラップは、ネットワーク上で発生した不正なユーザ認証、再起動、リンク ステータス（アップまたはダウン）、MAC アドレス トラッキングなどの状況を SNMP マネージャに通知します。SNMP エージェントはさらに、SNMP マネージャから *get-request*、*get-next-request*、および *set-request* 形式で送信される MIB 関連のクエリに応答します。

図 *65 : SNMP* ネットワーク



SNMP 通知

SNMP を使用すると、特定のイベントが発生した場合に、deviceから SNMP マネージャに通知を送信できます。SNMP 通知は、トラップまたは情報要求として送信できます。コマンド構文では、トラップまたは情報を選択するオプションがコマンドにない限り、キーワード traps はトラップ、情報、またはその両方を表します。**snmp-server host** コマンドを使用して、トラップまたは情報として SNMP 通知を送信するかどうかを指定します。

（注） SNMPv1 は informs をサポートしていません。

トラップは信頼性に欠けます。受信側はトラップを受信しても確認応答を送信しないので、トラップが受信されたかどうかが送信側にわからないからです。情報要求の場合、受信した SNMP マネージャは SNMP 応答プロトコル データ ユニット（PDU）でメッセージを確認します。送信側が応答を受信しなかった場合は、再び情報要求を送信できます。再送信できるので、情報の方がトラップより意図した宛先に届く可能性が高くなります。

情報の方がトラップより信頼性が高いのは、deviceおよびネットワークのリソースを多く消費するという特性にも理由があります。送信と同時に廃棄されるトラップと異なり、情報要求は応答を受信するまで、または要求がタイムアウトになるまで、メモリ内に保持されます。トラップの送信は1回限りですが、情報は数回にわたって再送信つまり再試行が可能です。再送信の回数が増えるとトラフィックが増加し、ネットワークのオーバーヘッドが高くなる原因にもなります。したがって、トラップにするか情報にするかは、信頼性を取るかリソースを取るかという選択になります。SNMP マネージャですべての通知を受信することが重要な場合は、情報要求を使用してください。ネットワークまたはdevice のメモリ上のトラフィックが問題になる場合で、なおかつ通知が不要な場合は、トラップを使用してください。

SNMP ifIndex MIB オブジェクト値

NMS の IF-MIB は、物理インターフェイスまたは論理インターフェイスを識別する、ゼロより大きい一意の値である interface index（ifIndex）オブジェクト値の生成および割り当てを行います。deviceの再起動またはdevice ソフトウェアのアップグレード時に、deviceは、インターフェイスにこれと同じ値を使用します。たとえば、deviceのポート 2 に 10003 という ifIndex 値が割り当てられていると、deviceの再起動後も同じ値が使用されます。

deviceは、次の表内のいずれかの値を使用して、インターフェイスに ifIndex 値を割り当てます。

表 *72 : ifIndex* 値

インターフェイスタイプ	**ifIndex 範囲**
SVI[5]	1 ～ 4999
EtherChannel	5001 ～ 5048
トンネル	5078 ～ 5142
タイプとポート番号に基づく物理（ギガビットイーサネットまたはSFP[6] モジュール インターフェイスなど）	10000 ～ 14500
ヌル	14501
ループバックおよびトンネル	24567+

[5] SVI = スイッチ仮想インターフェイス
[6] SFP = Small Form-Factor Pluggable

SNMP のデフォルト設定

機能	デフォルト設定
SNMP エージェント	ディセーブル[7]
SNMP トラップ レシーバ	未設定
SNMP トラップ	TCP接続のトラップ（tty）以外は、イネーブルではありません。
SNMP バージョン	バージョン キーワードがない場合、デフォルトはバージョン 1 になります。
SNMPv3 認証	キーワードを入力しなかった場合、セキュリティレベルはデフォルトで **noauth**（noAuthNoPriv）になります。
SNMP 通知タイプ	タイプが指定されていない場合、すべての通知が送信されます。

[7] これは、deviceが起動し、スタートアップ コンフィギュレーションに **snmp-server** グローバル コンフィギュレーション コマンドが設定されていない場合のデフォルトです。

SNMP 設定時の注意事項

device が起動し、device のスタートアップ コンフィギュレーションに少なくとも 1 つの **snmp-server** グローバル コンフィギュレーション コマンドが設定されている場合、SNMP エージェントは有効になります。

SNMP グループは、SNMP ユーザを SNMP ビューに対応付けるテーブルです。SNMP ユーザは、SNMP グループのメンバです。SNMP ホストは、SNMP トラップ動作の受信側です。SNMP エンジン *ID* は、ローカルまたはリモート SNMP エンジンの名前です。

SNMP グループを設定するときには、次の注意事項に従ってください。

- SNMP グループを設定するときは、通知ビューを指定しません。**snmp-server host** グローバル コンフィギュレーション コマンドがユーザの通知ビューを自動生成し、そのユーザに関連付けられているグループに追加します。グループの通知ビューを変更すると、そのグループに対応付けられたすべてのユーザが影響を受けます。
- リモート ユーザを設定する場合は、ユーザが存在するデバイスのリモート SNMP エージェントに対応する IP アドレスまたはポート番号を指定します。
- 特定のエージェントのリモートユーザを設定する前に、**snmp-server engineID** グローバル コンフィギュレーション コマンドを **remote** オプションとともに使用して、SNMP エンジン ID を設定してください。リモート エージェントの SNMP エンジン ID およびユーザ パスワードを使用して認証およびプライバシーダイジェストが算出されます。先にリモート エンジン ID を設定しておかないと、コンフィギュレーション コマンドがエラーになります。
- SNMP 情報を設定するときには、プロキシ要求または情報の送信先となるリモート エージェントの SNMP エンジン ID を SNMP データベースに設定しておく必要があります。
- ローカルユーザーがリモートホストと関連付けられていない場合、device は **auth**（authNoPriv）および **priv**（authPriv）の認証レベルの情報を送信しません。
- SNMP エンジン ID の値を変更すると、重大な影響が生じます。（コマンド ラインで入力された）ユーザのパスワードは、パスワードおよびローカル エンジン ID に基づいて、MD5 または SHA セキュリティ ダイジェストに変換されます。コマンドラインのパスワードは、RFC 2274 の規定に従って廃棄されます。このようにパスワードが廃棄されるため、エンジン ID 値を変更した場合は、SNMPv3 ユーザのセキュリティダイジェストが無効となり、**snmp-server user** *username* グローバル コンフィギュレーション コマンドを使用して、SNMP ユーザを再設定する必要があります。エンジン ID を変更した場合は、同様の制限によってコミュニティ ストリングも再設定する必要があります。

SNMP の設定方法

SNMP エージェントのディセーブル化

no snmp-server グローバル コンフィギュレーション コマンドは、デバイス上で実行している SNMP エージェントのすべてのバージョン（バージョン 1、バージョン 2C、バージョン 3）をディセーブルにします。入力した最初の **snmp-server** グローバル コンフィギュレーション コマンドによって、SNMP エージェントのすべてのバージョンを再度イネーブルにします。特に SNMP をイネーブルにするために指定された Cisco IOS コマンドはありません。

SNMP エージェントをディセーブルにするには、次の手順を実行します。

始める前に

SNMP エージェントをディセーブルにする前にイネーブルにする必要があります。デバイス上で入力した最初の **snmp-server** グローバル コンフィギュレーション コマンドによって SNMP エージェントがイネーブルになります。

手順の概要

1. **enable**
2. **configure terminal**
3. **no snmp-server**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no snmp-server** 例：	SNMP エージェント動作をディセーブルにします。

	コマンドまたはアクション	目的
	スイッチ(config)# **no snmp-server**	
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

コミュニティ ストリングの設定

SNMPマネージャとエージェントの関係を定義するには、SNMPコミュニティストリングを使用します。コミュニティ ストリングは、device上のエージェントへのアクセスを許可する、パスワードと同様の役割を果たします。ストリングに対応する次の特性を1つまたは複数指定することもできます。

- コミュニティ ストリングを使用してエージェントにアクセスできる SNMP マネージャの IP アドレスのアクセス リスト
- 指定のコミュニティにアクセスできるすべての MIB オブジェクトのサブセットを定義する MIB ビュー
- コミュニティにアクセスできる MIB オブジェクトの読み書き権限または読み取り専用権限

device上でコミュニティ ストリングを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server community** *string*［**view** *view-name*］［**ro** | **rw**］［*access-list-number*］
4. **access-list** *access-list-number* {**deny** | **permit**} *source* ［*source-wildcard*］
5. **end**
6. **show running-config**

7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server community** *string*［**view** *view-name*］［**ro**	**rw**］［*access-list-number*］ 例： スイッチ(config)# **snmp-server community comaccess ro 4**	コミュニティ ストリングを設定します。 （注） コンテキスト情報を区切るには @ 記号を使用します。このコマンドの設定時に SNMP コミュニティ ストリングの一部として @ 記号を使用しないでください。 • *string* には、パスワードと同様に機能し、SNMP プロトコルへのアクセスを許可するストリングを指定します。任意の長さのコミュニティ ストリングを 1 つまたは複数設定できます。 •（任意）**view** には、コミュニティがアクセスできるビューレコードを指定します。 •（任意）許可された管理ステーションで MIB オブジェクトを取得する場合は読み取り専用（**ro**）、許可された管理ステーションで MIB オブジェクトを取得および変更する場合は読み書き（**rw**）を指定します。デフォルトでは、コミュニティ ストリングはすべてのオブジェクトに対する読み取り専用アクセスを許可します。 •（任意）*access-list-number* には、1 ～ 99 および 1300 ～ 1999 の標準 IP アクセス リスト番号を入力します。

	コマンドまたはアクション	目的
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： `スイッチ(config)# access-list 4 deny any`	（任意）ステップ 3 で標準 IP アクセス リスト番号を指定してリストを作成した場合は、必要に応じてコマンドを繰り返します。 • *access-list-number* には、ステップ 3 で指定したアクセス リスト番号を入力します。 • **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 • *source* には、コミュニティ ストリングを使用してエージェントにアクセスできる SNMP マネージャの IP アドレスを入力します。 • （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在することに注意してください。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

SNMP コミュニティのアクセスをディセーブルにするには、そのコミュニティのコミュニティ ストリングをヌル ストリングに設定します（コミュニティ ストリングに値を入力しないでください）。

特定のコミュニティストリングを削除するには、**no snmp-server** グローバル コンフィギュレーション コマンドを使用します。

deviceのローカルまたはリモート SNMP サーバー エンジンを表す識別名（エンジン ID）を指定できます。SNMP ユーザーを SNMP ビューにマッピングする、SNMP サーバー グループを設定し、新規ユーザーを SNMP グループに追加できます。

SNMP グループおよびユーザの設定

deviceのローカルまたはリモート SNMP サーバー エンジンを表す識別名（エンジン ID）を指定できます。SNMP ユーザーを SNMP ビューにマッピングする、SNMP サーバー グループを設定し、新規ユーザーを SNMP グループに追加できます。

device上の SNMP グループとユーザーを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}
4. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
5. **snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*] } {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } [*priv* {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*} 例： `スイッチ(config)# snmp-server engineID local 1234`	SNMP のローカル コピーまたはリモート コピーに名前を設定します。 • *engineid-string* は、SNMP のコピー名を指定する 24 文字の ID ストリングです。後続ゼロが含まれる場合は、24 文字のエンジン ID すべてを指定する必要はありません。指定するのは、エンジン ID のうちゼロのみが続く箇所を除いた部分だけです。手順例では、123400000000000000000000 のエンジン ID を設定します。 • **remote** を指定した場合、SNMP のリモートコピーが置かれているデバイスの *ip-address* を指定し、任意でリモートデバイスのユーザデータグラム プロトコル（UDP）ポートを指定します。デフォルトは 162 です。
ステップ 4	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] 例： `スイッチ(config)# snmp-server group public v2c access lmnop`	リモート デバイス上で新しい SNMP グループを設定します。 *group-name* には、グループの名前を指定します。 次のいずれかのセキュリティ モデルを指定します。 • **v1** は、最も安全性の低いセキュリティ モデルです。 • **v2c** は、2 番目に安全性の低いセキュリティ モデルです。標準の 2 倍の幅で情報および整数を伝送できます。 • **v3**最も安全な場合には、次の認証レベルの 1 つを選択する必要があります。 **auth**：Message Digest 5（MD5）およびセキュア ハッシュ アルゴリズム（SHA）によるパケット認証を可能にします。 **noauth**：noAuthNoPriv セキュリティ レベルを可能にします。キーワードを指定しなかった場合、これがデフォルトです。 **priv**：データ暗号規格（DES）によるパケット暗号化（プライバシーともいう）を可能にします。

	コマンドまたはアクション	目的
		（任意）**read** *readview* とともに、エージェントの内容を表示できるビュー名を表す文字列（64 文字以内）を入力します。 （任意）**write** *writeview* とともに、データを入力し、エージェントの内容を表示できるビュー名を表す文字列（64 文字以内）を入力します。 （任意）**notify** *notifyview* とともに、通知、情報、またはトラップを指定するビュー名を表す文字列（64 文字以内）を入力します。 （任意）**access** *access-list* とともに、アクセスリスト名の文字列（64 文字以内）を入力します。
ステップ 5	**snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*] } {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } [*priv* {**des** | **3des** | **aes** {**128** | **192** | **256**} } *priv-password*] 例： `スイッチ(config)# snmp-server user Pat public v2c`	SNMP グループに対して新規ユーザを追加します。 *username* は、エージェントに接続するホスト上のユーザ名です。 *group-name* は、ユーザが関連付けられているグループの名前です。 **remote** を入力して、ユーザが所属するリモート SNMP エンティティおよびそのエンティティのホスト名または IP アドレスとともに、任意で UDP ポート番号を指定します。デフォルトは 162 です。 SNMP バージョン番号（**v1**、**v2c**、または **v3**）を入力します。**v3** を入力する場合は、次のオプションを追加します。 • **encrypted** は、パスワードを暗号化形式で表示するように指定します。このキーワードは、**v3** キーワードが指定されている場合にのみ使用できます。 • **auth** では、認証レベルを設定します。HMAC-MD5-96（**md5**）または HMAC-SHA-96（**sha**）認証レベルを指定できます。また、*auth-password* でパスワードの文字列を指定する必要があります（最大 64 文字）。 **v3** を入力すると、次のキーワードを使用して（64 文字以内）、プライベート（**priv**）暗号化アルゴリズムおよびパスワード文字列 *priv-password* を設定することもできます。

	コマンドまたはアクション	目的
		• **priv** は、ユーザベース セキュリティ モデル（USM）を指定します。 • **des** 56 ビット DES アルゴリズムを使用する場合に指定します。 • **3des** 168 ビット DES アルゴリズムを使用する場合に指定します。 • **aes** DES アルゴリズムを使用する場合に指定します。128 ビット暗号化、192 ビット暗号化、または 256 ビット暗号化のいずれかを選択する必要があります。 （任意）**access** *access-list* とともに、アクセスリスト名の文字列（64 文字以内）を入力します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

SNMP 通知の設定

トラップマネージャは、トラップを受信して処理する管理ステーションです。トラップは、特定のイベントが発生したときにdeviceが生成するシステム アラートです。デフォルトでは、トラップ マネージャは定義されず、トラップは送信されません。この Cisco IOS Release が稼働しているDevicesでは、トラップ マネージャを無制限に設定できます。

（注） コマンド構文で **traps** というワードを使用するコマンドは多数あります。トラップまたは情報を選択するオプションがコマンドにない限り、キーワード **traps** はトラップ、情報のいずれか、またはその両方を表します。**snmp-server host** グローバル コンフィギュレーション コマンドを使用して、トラップまたは情報として SNMP 通知を送信するかどうかを指定します。

snmp-server host グローバル コンフィギュレーション コマンドと組み合わせて使用すると、次の表に示す通知タイプを特定のホストで受信できます。これらのトラップの一部または全部をイネーブルにして、これを受信するようにトラップ マネージャを設定できます。

表 *73*：デバイスの通知タイプ

通知タイプのキーワード	説明
bridge	STP ブリッジ MIB トラップを生成します。
cluster	クラスタ設定が変更された場合に、トラップを生成します。
config	SNMP 設定が変更された場合に、トラップを生成します。
copy-config	SNMP コピー設定が変更された場合に、トラップを生成します。
cpu threshold	CPU に関連したトラップをイネーブルにします。
entity	SNMP エンティティが変更された場合に、トラップを生成します。
envmon	環境モニタ トラップを生成します。ファン（fan）、シャットダウン（shutdown）、ステータス（status）、電源（supply）、温度（temperature）の環境トラップのいずれかまたはすべてをイネーブルにできます。
errdisable	ポート VLAN が errdisable ステートになった場合に、トラップを生成します。1 分あたりの最大トラップ速度も設定できます。指定できる範囲は 0 ～ 10000 です。デフォルトは 0 で、レート制限がないという意味です。
flash	SNMP FLASH 通知を生成します。device スタックでは、オプションとして、フラッシュの追加または削除に関する通知を有効にできます。このようにすると、スタックから deviceを削除するか、またはスタックにスイッチを追加した場合に（物理的な取り外し、電源の再投入、またはリロードの場合に）、トラップが発行されます。
fru-ctrl	エンティティ現場交換可能ユニット（FRU）制御トラップを生成します。device スタックでは、このトラップはスタックにおけるdeviceの挿入/取り外しを意味します。

通知タイプのキーワード	説明
hsrp	ホットスタンバイ ルータ プロトコル（HSRP）が変更された場合に、トラップを生成します。
ipmulticast	IP マルチキャスト ルーティングが変更された場合に、トラップを生成します。
ipsla	SNMP IP サービスレベル契約（SLA）のトラップを生成します。
mac-notification	MAC アドレス通知のトラップを生成します。
msdp	Multicast Source Discovery Protocol（MSDP）が変更された場合に、トラップを生成します。
ospf	Open Shortest Path First（OSPF）が変更された場合に、トラップを生成します。シスコ固有、エラー、リンクステートアドバタイズ、レート制限、再送信、ステート変更に関するトラップを任意にイネーブルにできます。
pim	Protocol-Independent Multicast（PIM）が変更された場合に、トラップを生成します。無効な PIM メッセージ、ネイバー変更、およびランデブー ポイント（RP）マッピングの変更に関するトラップを任意にイネーブルにできます。
port-security	SNMP ポートセキュリティ トラップを生成します。1 秒あたりの最大トラップ速度も設定できます。指定できる範囲は 0 ～ 1000 秒です。デフォルトは 0 秒で、レート制限がないという意味です。 （注） 通知タイプ **port-security** を使用してトラップを設定する際に、まずポートセキュリティトラップを設定して、次に以下のポートセキュリティ トラップ レートを設定します。 **1. snmp-server enable traps port-security** **2. snmp-server enable traps port-security trap-rate** *rate*
snmp	認証、コールドスタート、ウォームスタート、リンクアップ、またはリンク ダウンについて、SNMP タイプ通知のトラップを生成します。
storm-control	SNMP ストーム制御のトラップを生成します。1 分あたりの最大トラップ速度も設定できます。指定できる範囲は 0 ～ 1000 です。デフォルトは 0 に設定されています（制限なしの状態では、発生ごとにトラップが送信されます）。
stpx	SNMP STP 拡張 MIB トラップを生成します。
syslog	SNMP の Syslog トラップを生成します。

通知タイプのキーワード	説明
tty	TCP 接続のトラップを生成します。このトラップは、デフォルトでイネーブルに設定されています。
vlan-membership	SNMP VLAN メンバーシップが変更された場合に、トラップを生成します。
vlancreate	SNMP VLAN 作成トラップを生成します。
vlandelete	SNMP VLAN 削除トラップを生成します。
vtp	VLAN トランキング プロトコル（VTP）が変更された場合に、トラップを生成します。

ホストにトラップまたは情報を送信するようにdeviceを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server engineID remote** *ip-address engineid-string*
4. **snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*] } {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] }
5. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**} } [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
6. **snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**} }] *community-string* [*notification-type*]
7. **snmp-server enable traps** *notification-types*
8. **snmp-server trap-source** *interface-id*
9. **snmp-server queue-length** *length*
10. **snmp-server trap-timeout** *seconds*
11. **end**
12. **show running-config**
13. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**snmp-server engineID remote** *ip-address engineid-string* 例： スイッチ(config)# **snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b**	リモート ホストのエンジン ID を指定します。
ステップ 4	**snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*] } {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } 例： スイッチ(config)# **snmp-server user Pat public v2c**	SNMP ユーザを設定し、ステップ 3 で作成したリモート ホストに関連付けます。 （注） アドレスに対応するリモート ユーザを設定するには、先にリモート ホストのエンジン ID を設定しておく必要があります。このようにしないと、エラー メッセージが表示され、コマンドが実行されません。
ステップ 5	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**} } [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] 例： スイッチ(config)# **snmp-server group public v2c access lmnop**	SNMP グループを設定します。
ステップ 6	**snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**} }] *community-string* [*notification-type*] 例： スイッチ(config)# **snmp-server host 203.0.113.1 comaccess snmp**	SNMP トラップ動作の受信先を指定します。 *host-addr* には、ホスト（対象となる受信側）の名前またはインターネット アドレスを指定します。 （任意）SNMP トラップをホストに送信するには、**traps**（デフォルト）を指定します。 （任意）SNMP 情報をホストに送信するには、**informs** を指定します。 （任意）SNMP **version**（**1** 、**2c** 、または **3**）を指定します。SNMPv1 は informs をサポートしていません。 （任意）バージョン 3 の場合、認証レベルとして **auth**、**noauth**、または **priv** を選択します。 （注） **priv** キーワードは、暗号化ソフトウェア イメージがインストールされている場合のみ指定できます。 *community-string* には、**version 1** または **version 2c** が指定されている場合、通知動作で送信される、パ

	コマンドまたはアクション	目的
		スワードに類似したコミュニティストリングを入力します。**version 3** が指定されている場合は、SNMPv3 のユーザ名を入力します。 コンテキスト情報を区切るには @ 記号を使用します。このコマンドの設定時に SNMP コミュニティストリングの一部として @ 記号を使用しないでください。 （任意）*notification-type* には、上の表に記載されているキーワードを使用します。タイプが指定されていない場合、すべての通知が送信されます。
ステップ 7	**snmp-server enable traps** *notification-types* 例： `スイッチ(config)# snmp-server enable traps snmp`	deviceでのトラップまたはインフォームの送信をイネーブルにし、送信する通知の種類を指定します。通知タイプの一覧については、上の表を参照するか、と入力してください。 **snmp-server enable traps ?** 複数のトラップタイプをイネーブルにするには、トラップタイプごとに **snmp-server enable traps** コマンドを個別に入力する必要があります。 （注） 通知タイプ **port-security** を使用してトラップを設定する際に、まずポートセキュリティトラップを設定して、次に以下のポート セキュリティ トラップ レートを設定します。 1. **snmp-server enable traps port-security** 2. **snmp-server enable traps port-security trap-rate** *rate*
ステップ 8	**snmp-server trap-source** *interface-id* 例： `スイッチ(config)# snmp-server trap-source gigabitethernet 1/0/1`	（任意）送信元インターフェイスを指定します。このインターフェイスによってトラップ メッセージの IP アドレスが提供されます。情報の送信元 IP アドレスも、このコマンドで設定します。
ステップ 9	**snmp-server queue-length** *length* 例： `スイッチ(config)# snmp-server queue-length 20`	（任意）各トラップ ホストのメッセージ キューの長さを指定します。指定できる値の範囲は 1 ～ 5000 です。デフォルトは 10 です。
ステップ 10	**snmp-server trap-timeout** *seconds* 例： `スイッチ(config)# snmp-server trap-timeout 60`	（任意）トラップ メッセージを再送信する頻度を指定します。指定できる範囲は 1 ～ 1000 です。デフォルトは 30 秒です。

	コマンドまたはアクション	目的
ステップ 11	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 12	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 13	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

snmp-server host コマンドでは、通知を受信するホストを指定します。**snmp-server enable traps** コマンドによって、指定された通知方式（トラップおよび情報）がグローバルにイネーブルになります。ホストが情報を受信できるようにするには、そのホストに対応する **snmp-server host informs** コマンドを設定し、**snmp-server enable traps** コマンドを使用して情報をグローバルにイネーブルにする必要があります。

指定したホストがトラップを受信しないようにするには、 **no snmp-server host** *host* グローバル コンフィギュレーション コマンドを使用します。キーワードを指定しないで **no snmp-server host** コマンドを使用すると、ホストへのトラップはディセーブルになりますが、情報はディセーブルになりません。情報をディセーブルにするには、**no snmp-server host informs** グローバル コンフィギュレーション コマンドを使用します。特定のトラップタイプをディセーブルにするには、**no snmp-server enable traps** *notification-types* グローバル コンフィギュレーション コマンドを使用します。

エージェント コンタクトおよびロケーションの設定

SNMP エージェントのシステム接点およびロケーションを設定して、コンフィギュレーション ファイルからこれらの記述にアクセスできるようにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server contact** *text*
4. **snmp-server location** *text*

5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server contact** *text* 例： スイッチ(config)# **snmp-server contact Dial System Operator at beeper 21555**	システムの連絡先文字列を設定します。
ステップ 4	**snmp-server location** *text* 例： スイッチ(config)# **snmp-server location Building 3/Room 222**	システムの場所を表す文字列を設定します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SNMP を通して使用する TFTP サーバの制限

SNMP を介したコンフィギュレーション ファイルの保存とロードに使用する TFTP サーバを、アクセス リストで指定されたサーバに限定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server tftp-server-list** *access-list-number*
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server tftp-server-list** *access-list-number* 例： `スイッチ(config)# snmp-server tftp-server-list 44`	SNMP を介したコンフィギュレーション ファイルのコピーに使用する TFTP サーバを、アクセス リストのサーバに限定します。 *access-list-number* には、1 ～ 99 および 1300 ～ 1999 の標準 IP アクセス リスト番号を入力します。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： `スイッチ(config)# access-list 44 permit 10.1.1.2`	標準アクセス リストを作成し、コマンドを必要な回数だけ実行します。 *access-list-number* には、ステップ 3 で指定したアクセス リスト番号を入力します。 **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。 *source* には、deviceにアクセスできる TFTP サーバーの IP アドレスを入力します。

	コマンドまたはアクション	目的
		（任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

SNMP ステータスのモニタリング

不正なコミュニティ ストリング エントリ、エラー、要求変数の数など、SNMP の入出力統計情報を表示するには、**show snmp** 特権 EXEC コマンドを使用します。また、次の表にリストされたその他の特権 EXEC コマンドを使用して、SNMP 情報を表示することもできます。

表 *74 : SNMP* 情報を表示するためのコマンド

コマンド	目的
show snmp	SNMP 統計情報を表示します。
	デバイスに設定されているローカル SNMP エンジンおよびすべてのリモート エンジンに関する情報を表示します。
show snmp group	ネットワーク上の各 SNMP グループに関する情報を表示します。
show snmp pending	保留中の SNMP 要求の情報を表示します。

コマンド	目的
show snmp sessions	現在のSNMPセッションの情報を表示します。
show snmp user	SNMP ユーザ テーブルの各 SNMP ユーザ名に関する情報を表示します。 （注） このコマンドは、**auth** | **noauth** | **priv** モードのSNMPv3設定情報を表示する際に使用する必要があります。この情報は、**show running-config** の出力には表示されません。

SNMP の例

次に、SNMP の全バージョンをイネーブルにする例を示します。この設定では、任意の SNMP マネージャがコミュニティ ストリング *public* を使用して、読み取り専用権限ですべてのオブジェクトにアクセスできます。この設定では、device はトラップを送信しません。

```
スイッチ(config)# snmp-server community public
```

次に、任意の SNMP マネージャがコミュニティ ストリング *public* を使用して、読み取り専用権限ですべてのオブジェクトにアクセスする例を示します。deviceはさらに、SNMPv1 を使用してホスト 192.180.1.111 および 192.180.1.33 に、SNMPv2C を使用してホスト 192.180.1.27 に VTP トラップを送信します。コミュニティ ストリング *public* は、トラップとともに送信されます。

```
スイッチ(config)# snmp-server community public
スイッチ(config)# snmp-server enable traps vtp
スイッチ(config)# snmp-server host 192.180.1.27 version 2c public
スイッチ(config)# snmp-server host 192.180.1.111 version 1 public
スイッチ(config)# snmp-server host 192.180.1.33 public
```

次に、*comaccess* コミュニティ ストリングを使用するアクセス リスト 4 のメンバに、すべてのオブジェクトへの読み取り専用アクセスを許可する例を示します。その他のSNMPマネージャは、どのオブジェクトにもアクセスできません。SNMP 認証障害トラップは、SNMPv2C がコミュニティ ストリング *public* を使用してホスト *cisco.com* に送信します。

```
スイッチ(config)# snmp-server community comaccess ro 4
スイッチ(config)# snmp-server enable traps snmp authentication
スイッチ(config)# snmp-server host cisco.com version 2c public
```

次に、エンティティ MIB トラップをホスト *cisco.com* に送信する例を示します。コミュニティ ストリングは制限されます。1 行目で、device はすでにイネーブルになっているトラップ以外に、エンティティMIBトラップを送信できるようになります。2行目はこれらのトラップの宛先を指定し、ホスト *cisco.com* に対する以前の **snmp-server** ホストコマンドを無効にします。

```
スイッチ(config)# snmp-server enable traps entity
スイッチ(config)# snmp-server host cisco.com restricted entity
```

次に、コミュニティ ストリング *public* を使用して、すべてのトラップをホスト *myhost.cisco.com* に送信するようにdeviceをイネーブルにする例を示します。

```
スイッチ(config)# snmp-server enable traps
スイッチ(config)# snmp-server host myhost.cisco.com public
```

次に、ユーザとリモートホストを関連付けて、ユーザがグローバル コンフィギュレーション モードの際に **auth**（authNoPriv）認証レベルで情報を送信する例を示します。

```
スイッチ(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
スイッチ(config)# snmp-server group authgroup v3 auth
スイッチ(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5
mypassword
スイッチ(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
スイッチ(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
スイッチ(config)# snmp-server enable traps
スイッチ(config)# snmp-server inform retries 0
```

第 40 章

SPAN および RSPAN の設定

SPAN および RSPAN の前提条件

SPAN

- SPAN トラフィックを特定の VLAN に制限するには、**filter vlan** キーワードを使用します。トランク ポートをモニターしている場合、このキーワードで指定された VLAN 上のトラフィックのみがモニターされます。デフォルトでは、トランク ポート上のすべての VLAN がモニターされます。

RSPAN

- RSPAN VLAN を設定してから、RSPAN 送信元または宛先セッションを設定することを推奨します。

SPAN および RSPAN の制約事項

SPAN

SPAN の制約事項は次のとおりです。

- 各 device で 66 のセッションを設定できます。最大 つの送信元セッションを設定できます。残りのセッションは、RSPAN 宛先セッションとして設定できます。送信元セッショ

ンは、ローカル SPAN セッションまたは RSPAN 送信元セッションのどちらかになります。

- SPAN 送信元の場合は、セッションごとに、単一のポートまたは VLAN、一連のポートまたはVLAN、一定範囲のポートまたはVLANのトラフィックを監視できます。1つのSPANセッションに、送信元ポートおよび送信元 VLAN を混在させることはできません。
- 宛先ポートを送信元ポートにすることはできません。同様に、送信元ポートを宛先ポートにすることもできません。
- 同じ宛先ポートで 2 つの SPAN セッションを設定することはできません。
- device ポートを SPAN 宛先ポートとして設定すると、通常のdevice ポートではなくなります。SPAN 宛先ポートを通過するのは、監視対象トラフィックのみになります。
- SPAN コンフィギュレーション コマンドを入力しても、前に設定した SPAN パラメータは削除されません。設定されている SPAN パラメータを削除するには、**no monitor session**{*session_number* | **all** | **local** | **remote**} グローバル コンフィギュレーション コマンドを入力する必要があります。
- ローカル SPAN では、**encapsulation replicate** キーワードが指定されている場合、SPAN 宛先ポートを経由する発信パケットは元のカプセル化ヘッダー（タグなし、ISL、または IEEE 802.1Q）を伝送します。このキーワードが指定されていない場合、パケットはネイティブ形式で送信されます。
- 無効のポートを送信元ポートまたは宛先ポートとして設定することはできますが、SPAN 機能が開始されるのは、宛先ポートと少なくとも 1 つの送信元ポートまたは送信元 VLAN が有効になってからです。
- 単一の SPAN セッションに、送信元 VLAN とフィルタ VLAN を混在させることはできません。

SPAN セッションのトラフィック監視には次の制約事項があります。

- ポートまたは VLAN を送信元にできますが、同じセッション内に送信元ポートと送信元 VLAN を混在させることはできません。
- Wireshark は、出力スパンがアクティブな場合は出力パケットをキャプチャしません。
- 同じdeviceまたはdevice スタック内で、ローカル SPAN と RSPAN の送信元セッションの両方を実行できます。device または device スタックは、合計 66 の送信元および RSPAN 宛先セッションをサポートします。
- 別個のまたは重複する SPAN 送信元ポートと VLAN のセットによって、SPAN または RSPAN 送信元セッションを 2 つ個別に設定できます。スイッチド ポートおよびルーテッド ポートはいずれも SPAN 送信元および宛先として設定できます。
- 1 つの SPAN セッションに複数の宛先ポートを設定できますが、1 つのdevice スタックあたりに設定できる宛先ポートは最大で 64 個です。

- SPAN セッションがdeviceの通常の動作を妨げることはありません。ただし、10 Mbps のポートで 100 Mbps のポートをトラフィック監視するなど、オーバーサブスクライブの SPAN 宛先は、パケットのドロップまたは消失を招くことがあります。
- SPAN または RSPAN が有効の場合、監視中の各パケットは 2 回送信されます（1 回は標準トラフィックとして、もう 1 回は監視されたパケットとして）。多数のポートまたは VLAN を監視すると、大量のネットワーク トラフィックが生成されることがあります。
- ディセーブルのポート上に SPAN セッションを設定することはできますが、そのセッション用に宛先ポートと少なくとも 1 つの送信元ポートまたは VLAN をイネーブルにしない限り、SPAN セッションはアクティブになりません。
- deviceは、単一セッション内でのローカル SPAN と RSPAN の併用をサポートしません。
  - RSPAN 送信元セッションにローカル宛先ポートを設定できません。
  - RSPAN 宛先セッションにローカル送信元ポートを設定できません。
  - 同じdeviceまたはdevice スタック上で、同じ RSPAN VLAN を使用する RSPAN 宛先セッションおよび RSPAN 送信元セッションを実行できません。

RSPAN

RSPAN の制約事項は次のとおりです。

- RSPAN は、BPDU パケット監視または他のレイヤ 2 device プロトコルをサポートしません。
- RSPAN VLAN はトランク ポートにのみ設定されており、アクセス ポートには設定されていません。不要なトラフィックが RSPAN VLAN に発生しないようにするために、参加しているすべてのdevicesで VLAN RSPAN 機能がサポートされていることを確認してください。
- 送信元トランク ポートにアクティブな RSPAN VLAN が設定されている場合、RSPAN VLAN はポートベース RSPAN セッションの送信元として含まれます。また、RSPAN VLAN を SPAN セッションの送信元に設定することもできます。ただし、deviceはスパンされたトラフィックをモニターしないため、deviceの RSPAN 送信元セッションの宛先として識別された RSPAN VLAN では、パケットの出力スパニングがサポートされません。
- CDP パケットは、ハードウェアの制限により、RSPAN が設定された VLAN では転送されません。これは、スイッチに接続されたデバイス上で RSPAN VLAN を伝送するすべてのインターフェイスの CDP をディセーブルにすることで回避できます。
- VTP および VTP プルーニングをイネーブルにすると、トランク内で RSPAN トラフィックがプルーニングされ、1005 以下の VLAN ID に関して、ネットワークで不必要な RSPAN トラフィックのフラッディングが防止されます。
- RSPAN を使用するには、スイッチが LAN Base イメージを実行している必要があります。

SPAN および RSPAN について

SPAN および RSPAN

ポートまたは VLAN を通過するネットワーク トラフィックを解析するには、SPAN または RSPAN を使用して、そのdevice上、またはネットワーク アナライザやその他のモニター デバイス、あるいはセキュリティ デバイスに接続されている別のdevice上のポートにトラフィックのコピーを送信します。SPAN は送信元ポート上または送信元 VLAN 上で受信、送信、または送受信されたトラフィックを宛先ポートにコピー（ミラーリング）して、解析します。SPAN は送信元ポートまたは VLAN 上のネットワーク トラフィックのスイッチングには影響しません。宛先ポートは SPAN 専用にする必要があります。SPAN または RSPAN セッションに必要なトラフィック以外、宛先ポートがトラフィックを受信したり転送したりすることはありません。

SPAN を使用してモニターできるのは、送信元ポートを出入りするトラフィックまたは送信元 VLAN に出入りするトラフィックだけです。送信元 VLAN にルーティングされたトラフィックはモニターできません。たとえば、着信トラフィックをモニターしている場合、別の VLAN から送信元 VLAN にルーティングされているトラフィックはモニターできません。ただし、送信元 VLAN で受信し、別の VLAN にルーティングされるトラフィックは、モニターできます。

ネットワーク セキュリティ デバイスからトラフィックを注入する場合、SPAN または RSPAN 宛先ポートを使用できます。たとえば、Cisco 侵入検知システム（IDS）センサー装置を宛先ポートに接続した場合、IDS デバイスは TCP リセット パケットを送信して、疑わしい攻撃者の TCP セッションを停止させることができます。

ローカル SPAN

ローカル SPAN は 1 つのdevice内の SPAN セッション全体をサポートします。すべての送信元ポートまたは送信元 VLAN、および宛先ポートは、同じdeviceまたはdevice スタック内にあります。ローカル SPAN は、任意の VLAN 上の 1 つまたは複数の送信元ポートからのトラフィック、あるいは 1 つまたは複数の VLAN からのトラフィックを解析するために宛先ポートへコピーします。

図 *66 :* 単一デバイスでのローカル *SPAN* の設定例

ポート 5（送信元ポート）上のすべてのトラフィックがポート 10（宛先ポート）にミラーリングされます。ポート 10 のネットワーク アナライザは、ポート 5 に物理的には接続されていま

せんが、ポート 5 からのすべてのネットワーク トラフィックを受信します。

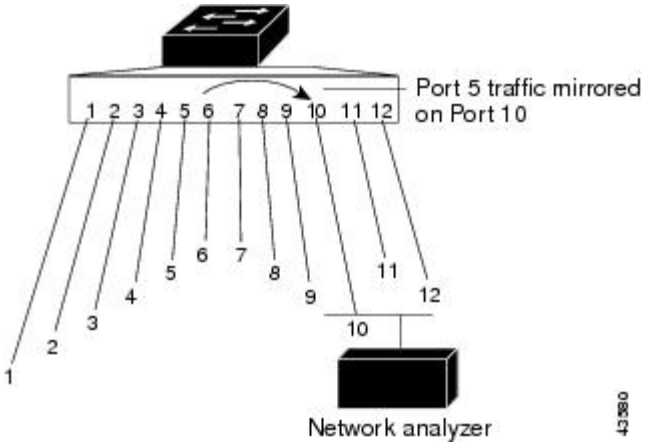
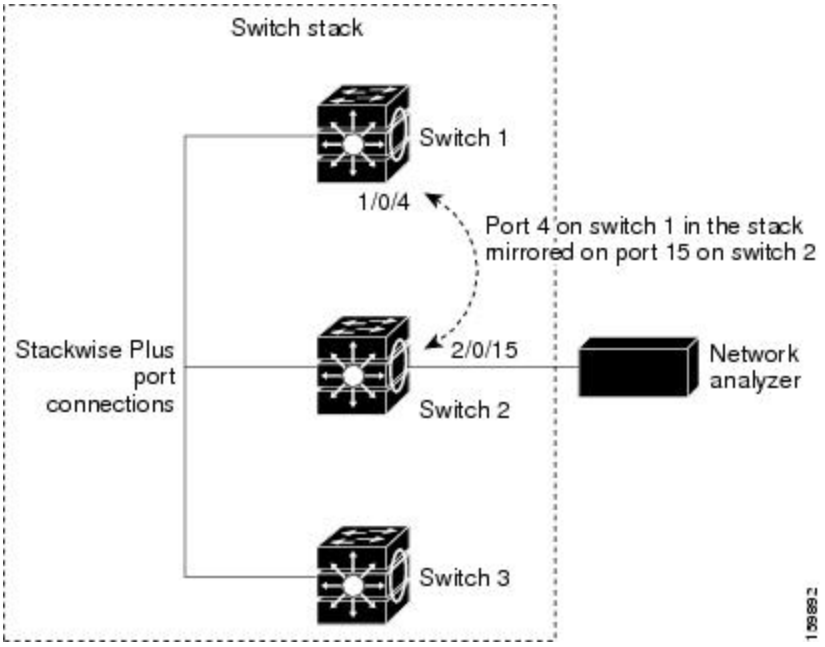


図 *67 : デバイス スタックでのローカル SPAN の設定例*

これは、device スタック内のローカル SPAN の例です。送信元ポートと宛先ポートは異なるスタック メンバにあります。



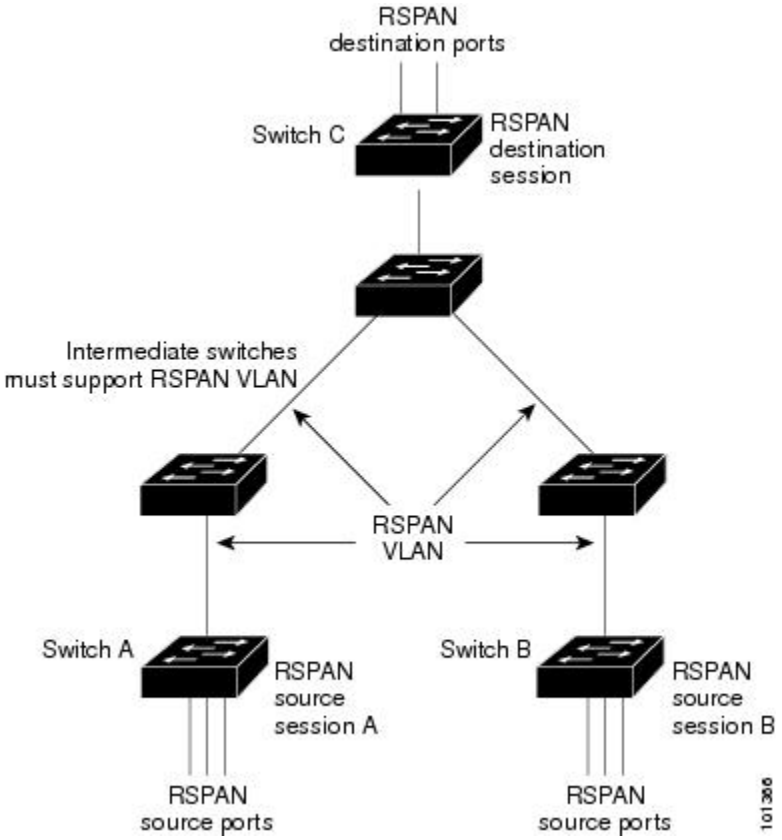
リモート SPAN

RSPANは、異なるdevices（または異なるdeviceスタック）上の送信元ポート、送信元VLAN、および宛先ポートをサポートしているので、ネットワーク上で複数のdevicesをリモート監視できます。

図 *68 : RSPAN の設定例*

下の図に デバイス A と デバイス B の送信元ポートを示します。各 RSPAN セッションのトラフィックは、ユーザーが指定した RSPAN VLAN 上で伝送されます。この RSPAN VLAN は、

参加しているすべての devices の RSPAN セッション専用です。送信元ポートまたは VLAN からの RSPAN トラフィックは RSPAN VLAN にコピーされ、RSPAN VLAN を伝送するトランクポートを介して、RSPAN VLAN を監視する宛先セッションに転送されます。各 RSPAN 送信元 deviceには、ポートまたは VLAN のいずれかが RSPAN 送信元として必要です。図中のデバイス C のように、宛先は常に物理ポートになります。



SPAN と RSPAN の概念および用語

SPAN セッション

SPAN セッション（ローカルまたはリモート）を使用すると、1 つまたは複数のポート上、あるいは1つまたは複数の VLAN 上でトラフィックをモニターし、そのモニターしたトラフィックを 1 つまたは複数の宛先ポートに送信できます。

ローカル SPAN セッションは、宛先ポートと送信元ポートまたは送信元 VLAN（すべて単一のネットワーク デバイス上にある）を結び付けたものです。ローカル SPAN には、個別の送信元および宛先のセッションはありません。ローカル SPAN セッションはユーザーが指定した入力および出力のパケット セットを収集し、SPAN データ ストリームを形成して、宛先ポートに転送します。

RSPAN は少なくとも 1 つの RSPAN 送信元セッション、1 つの RSPAN VLAN、および少なくとも 1 つの RSPAN 宛先セッションで構成されています。RSPAN 送信元セッションと RSPAN 宛先セッションは、異なるネットワーク デバイス上に別々に設定します。デバイスに RSPAN

送信元セッションを設定するには、一連の送信元ポートまたは送信元 VLAN を RSPAN VLAN に関連付けます。このセッションの出力は、RSPAN VLAN に送信される SPAN パケットのストリームです。別のデバイスにRSPAN宛先セッションを設定するには、宛先ポートをRSPAN VLANに関連付けます。宛先セッションはRSPAN VLANトラフィックをすべて収集し、RSPAN宛先ポートに送信します。

RSPAN 送信元セッションは、パケット ストリームが転送される点を除き、ローカル SPAN セッションに非常に似ています。RSPAN 送信元セッションでは、SPAN パケットに RSPAN VLAN ID ラベルが再設定され、通常のトランク ポートを介して宛先deviceに転送されます。

RSPAN宛先セッションはRSPAN VLAN上で受信されたすべてのパケットを取得し、VLANのタギングを除去し、宛先ポートに送ります。セッションは、（レイヤ2制御パケットを除く）すべての RSPAN VLAN パケットのコピーを分析のためにユーザーに提供します。

SPAN セッションでのトラフィックのモニターには、次のような制約があります。

- ポートまたは VLAN を送信元にできますが、同じセッション内に送信元ポートと送信元 VLAN を混在させることはできません。
- 同じdeviceまたはdevice スタック内で、ローカル SPAN と RSPAN の送信元セッションの両方を実行できます。device または device スタックは、合計 66 の送信元および RSPAN 宛先セッションをサポートします。
- 別個のまたは重複する SPAN 送信元ポートと VLAN のセットによって、SPAN または RSPAN 送信元セッションを 2 つ個別に設定できます。スイッチド ポートおよびルーテッド ポートはいずれも SPAN 送信元および宛先として設定できます。
- 1 つの SPAN セッションに複数の宛先ポートを設定できますが、1 つのdevice スタックあたりに設定できる宛先ポートは最大で 64 個です。
- SPAN セッションがdeviceの通常の動作を妨げることはありません。ただし、10 Mbps のポートで 100 Mbps のポートをトラフィック監視するなど、オーバーサブスクライブの SPAN 宛先は、パケットのドロップまたは消失を招くことがあります。
- SPAN または RSPAN が有効の場合、監視中の各パケットは 2 回送信されます（1 回は標準トラフィックとして、もう 1 回は監視されたパケットとして）。したがって、多数のポートまたは VLAN をモニターすると、大量のネットワーク トラフィックが生成されることがあります。
- ディセーブルのポート上にSPANセッションを設定することはできますが、そのセッション用に宛先ポートと少なくとも 1 つの送信元ポートまたは VLAN をイネーブルにしない限り、SPAN セッションはアクティブになりません。
- deviceは、単一セッション内でのローカル SPAN と RSPAN の併用をサポートしません。
  - RSPAN 送信元セッションにローカル宛先ポートを設定できません。
  - RSPAN 宛先セッションにローカル送信元ポートを設定できません。
  - 同じdeviceまたはdevice スタック上で、同じ RSPAN VLAN を使用する RSPAN 宛先セッションおよび RSPAN 送信元セッションを実行できません。

モニタ対象トラフィック

SPAN セッションは、次のトラフィック タイプを監視できます。

- 受信（Rx）SPAN：受信（または入力）SPANは、deviceが変更または処理を行う前に、送信元インターフェイスまたは VLAN が受信したすべてのパケットをできるだけ多くモニターリングします。送信元が受信した各パケットのコピーがその SPANセッションに対応する宛先ポートに送られます。

  Diffserv コード ポイント（DSCP）の変更など、ルーティングや Quality of Service（QoS）が原因で変更されたパケットは、変更される前にコピーされます。

  受信処理中にパケットをドロップする可能性のある機能は、入力 SPANには影響を与えません。宛先ポートは、実際の着信パケットがドロップされた場合でも、パケットのコピーを受信します。パケットをドロップする可能性のある機能は、標準および拡張 IP 入力アクセス コントロール リスト（ACL）、入力 QoS ポリシング、VLAN ACL、および出力 QoS ポリシングです。

- 送信（Tx）SPAN：送信（または出力）SPANは、deviceによる変更または処理がすべて実行されたあとに、送信元インターフェイスから送信されたすべてのパケットをできる限り多くモニターリングします。送信元が送信した各パケットのコピーがその SPANセッションに対応する宛先ポートに送られます。コピーはパケットの変更後に用意されます。

  ルーティングが原因で変更されたパケット（存続可能時間（TTL）、MACアドレス、QoS値の変更など）は、宛先ポートで（変更されて）コピーされます。

  送信処理中にパケットをドロップする可能性のある機能は、SPAN 用の複製コピーにも影響します。これらの機能には、標準および拡張 IP 出力 ACL、出力 QoS ポリシングがあります。

- 両方：SPAN セッションで、受信パケットと送信パケットの両方について、ポートまたは VLAN をモニタすることもできます。これはデフォルトです。

したがって、カプセル化レプリケーションがイネーブルにされたローカルSPANセッションでは、タグなし、および IEEE 802.1Q タグ付きパケットが宛先ポートに混在することがあります。

デバイスの輻輳により、入力送信元ポート、出力送信元ポート、またはSPAN宛先ポートでパケットがドロップされることがあります。一般に、これらの特性は互いに無関係です。次に例を示します。

- パケットは通常どおり転送されますが、SPAN 宛先ポートのオーバーサブスクライブが原因でモニタされないことがあります。
- 入力パケットが標準転送されないにもかかわらず、SPAN 宛先ポートに着信することがあります。
- deviceの輻輳が原因でドロップされた出力パケットは、出力 SPAN からもドロップされます。

SPAN の設定によっては、同一送信元のパケットのコピーが複数、SPAN 宛先ポートに送信されます。たとえば、ポート A での RX モニター用とポート B での TX モニター用に双方向（RX

と TX）SPAN セッションが設定されているとします。パケットがポート A から device に入ってポート B にスイッチされると、着信パケットも発信パケットも宛先ポートに送信されます。このため、両方のパケットは同じものになります。レイヤ 3 書き換えが行われた場合には、パケット変更のため異なるパケットになります。

送信元ポート

送信元ポート（別名モニター側ポート）は、ネットワーク トラフィック分析のために監視するスイッチド ポートまたはルーテッド ポートです。

1 つのローカル SPAN セッションまたは RSPAN 送信元セッションでは、送信元ポートまたは VLAN のトラフィックを単一方向または双方向でモニターできます。

deviceは、任意の数の送信元ポート（deviceで利用可能なポートの最大数まで）と任意の数の送信元 VLAN（サポートされている VLAN の最大数まで）をサポートしています。

ただし、deviceが送信元ポートまたは VLAN でサポートするセッション数には上限（4 つ。deviceが Catalyst 2960-S スイッチのスタック内にある場合は 2 つ）（ローカルまたは RSPAN）があります。単一のセッションにポートおよび VLAN を混在させることはできません。

送信元ポートの特性は、次のとおりです。

- 複数の SPAN セッションでモニターできます。
- モニターする方向（入力、出力、または両方）を指定して、各送信元ポートを設定できます。
- すべてのポート タイプ（EtherChannel、ギガビット イーサネットなど）が可能です。
- EtherChannel 送信元の場合は、EtherChannel 全体で、または物理ポートがポート チャネルに含まれている場合は物理ポート上で個別に、トラフィックをモニターできます。
- アクセス ポート、トランク ポート、ルーテッド ポート、または音声 VLAN ポートに指定できます。
- 宛先ポートにすることはできません。
- 送信元ポートは同じ VLAN にあっても異なる VLAN にあってもかまいません。
- 単一セッション内で複数の送信元ポートをモニターすることが可能です。

送信元 VLAN

VLAN ベースの SPAN（VSPAN）では、1 つまたは複数の VLAN のネットワーク トラフィックをモニターできます。VSPAN 内の SPAN または RSPAN 送信元インターフェイスが VLAN ID となり、トラフィックはその VLAN のすべてのポートでモニターされます。

VSPAN には次の特性があります。

- 送信元 VLAN 内のすべてのアクティブ ポートは送信元ポートとして含まれ、単一方向または双方向でモニターできます。
- 指定されたポートでは、モニター対象の VLAN 上のトラフィックのみが宛先ポートに送信されます。

- 宛先ポートが送信元 VLAN に所属する場合は、送信元リストから除外され、モニターされません。
- ポートが送信元 VLAN に追加または削除されると、これらのポートで受信された送信元 VLAN のトラフィックは、モニター中の送信元に追加または削除されます。
- VLAN 送信元と同じセッション内のフィルタ VLAN を使用することはできません。
- モニターできるのは、イーサネット VLAN だけです。

VLAN フィルタリング

トランク ポートを送信元ポートとしてモニターする場合、デフォルトでは、トランク上でアクティブなすべての VLAN がモニターされます。VLAN フィルタリングを使用して、トランク送信元ポートでの SPAN トラフィックのモニター対象を特定の VLAN に制限できます。

- VLAN フィルタリングが適用されるのは、トランク ポートまたは音声 VLAN ポートのみです。
- VLAN フィルタリングはポートベース セッションにのみ適用され、VLAN 送信元によるセッションでは使用できません。
- VLAN フィルタ リストが指定されている場合、トランク ポートまたは音声 VLAN アクセス ポートではリスト内の該当 VLAN のみがモニターされます。
- 他のポート タイプから着信する SPAN トラフィックは、VLAN フィルタリングの影響を受けません。つまり、すべての VLAN を他のポートで使用できます。
- VLAN フィルタリング機能は、宛先 SPAN ポートに転送されたトラフィックにのみ作用し、通常のトラフィックのスイッチングには影響を与えません。

宛先ポート

各ローカル SPAN セッションまたは RSPAN 宛先セッションには、送信元ポートおよび VLAN からのトラフィックのコピーを受信し、SPAN パケットをユーザー（通常はネットワーク アナライザ）に送信する宛先ポート（別名モニター側ポート）が必要です。

宛先ポートの特性は、次のとおりです。

- ローカル SPAN セッションの場合、宛先ポートは送信元ポートと同じdeviceまたはdevice スタックに存在している必要があります。RSPAN セッションの場合は、RSPAN 宛先セッションを含むdevice上にあります。RSPAN 送信元セッションのみを実行するdeviceまたは device スタックには、宛先ポートはありません。
- ポートを SPAN 宛先ポートとして設定すると、元のポート設定が上書きされます。SPAN 宛先設定を削除すると、ポートは以前の設定に戻ります。ポートが SPAN 宛先ポートとして機能している間にポートの設定が変更されると、SPAN 宛先設定が削除されるまで、変更は有効になりません。

- ポートがEtherChannelグループに含まれていた場合、そのポートが宛先ポートとして設定されている間、グループから削除されます。削除されたポートがルーテッドポートであった場合、このポートはルーテッド ポートでなくなります。
- 任意のイーサネット物理ポートにできます。
- セキュア ポートにすることはできません。
- 送信元ポートにすることはできません。
- 一度に 1 つの SPAN セッションにしか参加できません（ある SPAN セッションの宛先ポートは、別の SPAN セッションの宛先ポートになることはできません）。
- アクティブな場合、着信トラフィックはディセーブルになります。ポートは SPAN セッションに必要なトラフィック以外は送信しません。宛先ポートでは着信トラフィックを学習したり、転送したりしません。
- 入力トラフィック転送がネットワーク セキュリティ デバイスでイネーブルの場合、宛先ポートはレイヤ 2 でトラフィックを転送します。
- レイヤ 2 プロトコル（STP、VTP、CDP、DTP、PAgP）のいずれにも参加しません。
- 任意の SPAN セッションの送信元 VLAN に所属する宛先ポートは、送信元リストから除外され、モニターされません。
- deviceまたはdevice スタックの宛先ポートの最大数は 64 です。

ローカル SPAN および RSPAN 宛先ポートは、VLAN タギングおよびカプセル化で次のように動作が異なります。

- ローカル SPAN では、宛先ポートに **encapsulation replicate** キーワードが指定されている場合、各パケットに元のカプセル化が使用されます（タグなし、ISL、または IEEE 802.1Q）。これらのキーワードが指定されていない場合、パケットはタグなしフォーマットになります。したがって、**encapsulation replicate** がイネーブルになっているローカル SPAN セッションの出力に、タグなし、ISL、または IEEE 802.1Q タグ付きパケットが混在することがあります。
- RSPAN の場合は、元の VLAN ID は RSPAN VLAN ID で上書きされるため失われます。したがって、宛先ポート上のすべてのパケットはタグなしになります。

RSPAN VLAN

RSPAN VLAN は、RSPAN の送信元セッションと宛先セッション間で SPAN トラフィックを伝送します。RSPAN VLAN には、次の特性があります。

- RSPAN VLAN 内のすべてのトラフィックは、常にフラッディングされます。
- RSPAN VLAN では MAC アドレスは学習されません。
- RSPAN VLAN トラフィックが流れるのは、トランク ポート上のみです。

- RSPAN VLAN は、**remote-span** VLAN コンフィギュレーション モード コマンドを使用して、VLAN コンフィギュレーション モードで設定する必要があります。
- STP は RSPAN VLAN トランク上で実行できますが、SPAN 宛先ポート上では実行できません。
- RSPAN VLAN を、プライベート VLAN のプライマリまたはセカンダリ VLAN にはできません。

VLAN トランキング プロトコル（VTP）に対して可視である VLAN 1 ～ 1005 の場合、VLAN ID および対応する RSPAN 特性は VTP によって伝播されます。拡張 VLAN 範囲（1006 ～ 4094）内の RSPAN VLAN ID を割り当てる場合は、すべての中間devicesを手動で設定する必要があります。

通常は、ネットワークに複数の RSPAN VLAN を配置し、それぞれの RSPAN VLAN でネットワーク全体の RSPAN セッションを定義します。つまり、ネットワーク内の任意の場所にある複数の RSPAN 送信元セッションで、パケットを RSPAN セッションに送信できます。また、ネットワーク全体に対して複数の RSPAN 宛先セッションを設定し、同じ RSPAN VLAN をモニタしたり、ユーザにトラフィックを送信したりできます。セッションは RSPAN VLAN ID によって区別されます。

SPAN および RSPAN と他の機能の相互作用

SPAN は次の機能と相互に作用します。

- ルーティング：SPAN はルーテッド トラフィックを監視しません。VSPAN が監視するのはdeviceに出入りするトラフィックに限られ、VLAN 間でルーティングされるトラフィックは監視しません。たとえば、VLAN が受信モニタされ、deviceが別の VLAN から監視対象VLANにトラフィックをルーティングする場合、そのトラフィックは監視されず、SPAN 宛先ポートで受信されません。
- STP：SPAN または RSPAN セッションがアクティブな間、宛先ポートは STP に参加しません。SPAN または RSPAN セッションが無効になると、宛先ポートは STP に参加できます。送信元ポートでは、SPAN は STP ステータスに影響を与えません。STP は RSPAN VLAN を伝送するトランク ポート上でアクティブにできます。
- CDP：SPAN セッションがアクティブな間、SPAN 宛先ポートは CDP に参加しません。SPAN セッションがディセーブルになると、ポートは再び CDP に参加します。
- VTP：VTP を使用すると、devices間で RSPAN VLAN のプルーニングが可能です。
- VLAN およびトランキング：送信元ポート、または宛先ポートの VLAN メンバーシップまたはトランクの設定値を、いつでも変更できます。ただし、宛先ポートの VLAN メンバーシップまたはトランクの設定値に対する変更が有効になるのは、SPAN 宛先設定を削除してからです。送信元ポートの VLAN メンバーシップまたはトランクの設定値に対する変更は、ただちに有効になり、対応する SPAN セッションが変更に応じて自動的に調整されます。
- EtherChannel：EtherChannel グループを送信元ポートとして設定することができます。グループが SPAN 送信元として設定されている場合、グループ全体が監視されます。

監視対象の EtherChannel グループに物理ポートを追加すると、SPAN 送信元ポート リストに新しいポートが追加されます。監視対象の EtherChannel グループからポートを削除すると、送信元ポート リストからそのポートが自動的に削除されます。

EtherChannel グループに所属する物理ポートを SPAN 送信元ポートとして設定し、引き続き EtherChannel の一部とすることができます。この場合、この物理ポートは EtherChannel に参加しているため、そのポートからのデータは監視されます。ただし、EtherChannel グループに含まれる物理ポートを SPAN 宛先として設定した場合、その物理ポートはグループから削除されます。SPAN セッションからそのポートが削除されると、EtherChannel グループに再加入します。EtherChannel グループから削除されたポートは、グループ メンバのままですが、inactive または suspended ステートになります。

EtherChannel グループに含まれる物理ポートが宛先ポートであり、その EtherChannel グループが送信元の場合、ポートは EtherChannel グループおよび監視対象ポート リストから削除されます。

- マルチキャスト トラフィックを監視できます。出力ポートおよび入力ポートの監視では、未編集のパケットが 1 つだけ SPAN 宛先ポートに送信されます。マルチキャスト パケットの送信回数は反映されません。
- プライベート VLAN ポートは、SPAN 宛先ポートには設定できません。
- セキュア ポートを SPAN 宛先ポートにすることはできません。

  SPAN セッションでは、入力転送が宛先ポートで有効の場合、出力を監視しているポートでポート セキュリティを有効にしないでください。RSPAN 送信元セッションでは、出力を監視しているポートでポート セキュリティを有効にしないでください。
- IEEE 802.1x ポートは SPAN 送信元ポートにできます。SPAN 宛先ポート上で IEEE 802.1x を有効にできますが、SPAN 宛先としてこのポートを削除するまで、IEEE 802.1x は無効に設定されます。

  SPAN セッションでは、入力転送が宛先ポートで有効の場合、出力を監視しているポートで IEEE 802.1x を有効にしないでください。RSPAN 送信元セッションでは、出力を監視しているポートで IEEE 802.1x を有効にしないでください。

フローベースの SPAN

送信元ポートで監視されるトラフィックにアクセス コントロール リスト（ACL）を適用するフローベース SPAN（FSPAN）またはフローベース RSPAN（FRSPAN）を使用して、SPAN または RSPAN で監視するネットワーク トラフィックのタイプを制御できます。FSPAN ACL は、IPv4、IPv6、および監視される非 IP トラフィックをフィルタリングするように設定できます。

インターフェイスを通して ACL を SPAN セッションに適用します。ACL は SPAN セッション内のすべてのインターフェイスで監視されるすべてのトラフィックに適用されます。この ACL によって許可されるパケットは、SPAN 宛先ポートにコピーされます。ほかのパケットは SPAN 宛先ポートにコピーされません。

元のトラフィックは継続して転送され、接続している任意のポート、VLAN、およびルータ ACL が適用されます。FSPAN ACL は転送の決定に影響を与えることはありません。同様に、ポート、VLAN、およびルータ ACL は、トラフィックのモニタリングに影響を与えません。セキュリティ入力 ACL がパケットを拒否したために転送されない場合でも、FSPAN ACL が許可すると、パケットは SPAN 宛先ポートにコピーされます。しかし、セキュリティ出力 ACL がパケットを拒否したために転送されない場合、パケットはSPAN宛先ポートにコピーされません。ただし、セキュリティ出力 ACL がパケットの送信を許可した場合だけ、パケットは、FSPAN ACL が許可した場合 SPAN 宛先ポートにコピーされます。これは RSPAN セッションについてもあてはまります。

（注） FSPAN セッションを設定するときは、既存の SPAN セッションを削除し、FSPAN セッションを設定してから、SPAN セッションを再設定してください。

SPAN セッションには、次の 3 つのタイプの FSPAN ACL を接続できます。

- IPv4 FSPAN ACL：IPv4 パケットだけをフィルタリングします。
- IPv6 FSPAN ACL：IPv6 パケットだけをフィルタリングします。
- MAC FSPAN ACL：IP パケットだけをフィルタリングします。

セキュリティ ACL は、device上の FSPAN ACL よりも高いプライオリティをもっています。FSPAN ACL が適用され、その後ハードウェア メモリに収まらないセキュリティ ACL を追加する場合、適用された FSPAN ACL は、セキュリティ ACL のスペースを確保するためにメモリから削除されます。この処理（アンローディングと呼ばれる）は、システム メッセージにより通知されます。メモリ内に常駐するスペースが確保できたら、device上のハードウェア メモリに FSPAN ACL が追加されます。この処理（リローディングと呼ばれる）は、システム メッセージにより通知されます。IPv4、IPv6、およびMAC FSPAN ACL は、別個にアンロードまたはリロードできます。

スタックに設定された VLAN ベースの FSPAN セッションが 1 つまたは複数のdevices上のハードウェア メモリに収まらない場合、セッションはこれらのdevices上でアンロードされたものとして処理され、deviceでの FSPAN ACL およびソーシングのためのトラフィックは、SPAN 宛先ポートにコピーされません。FSPAN ACL は継続して正しく適用され、トラフィックはFSPAN ACL がハードウェア メモリに収まるdevicesの SPAN 宛先ポートにコピーされます。

空の FSPAN ACL が接続されると、一部のハードウェア機能により、その ACL の SPAN 宛先ポートにすべてのトラフィックがコピーされます。十分なハードウェア リソースが使用できない場合、空の FSPAN ACL もアンロードされる可能性があります。

IPv4 および MAC FSPAN ACL は、すべてのフィーチャ セットでサポートされています。IPv6 FSPAN ACL は、拡張 IP Services フィーチャ セットでだけサポートされています。

SPAN および RSPAN のデフォルト設定

表 *75 : SPAN* および *RSPAN* のデフォルト設定

機能	デフォルト設定
SPAN のステート（SPAN および RSPAN）	ディセーブル
モニタする送信元ポート トラフィック	受信トラフィックと送信トラフィックの両方（**both**）
カプセル化タイプ（宛先ポート）	ネイティブ形式（タグなしパケット）
入力転送（宛先ポート）	ディセーブル
VLAN フィルタリング	送信元ポートとして使用されるトランク インターフェイス上では、すべての VLAN がモニタリングされます。
RSPAN VLAN	未設定

設定時の注意事項

SPAN 設定時の注意事項

- SPAN セッションから送信元ポート、宛先ポート、または VLAN を削除する場合は、**no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} グローバル コンフィギュレーション コマンドまたは **no monitor session** *session_number* **destination interface** *interface-id* グローバル コンフィギュレーション コマンドを使用します。宛先インターフェイスの場合、このコマンドの **no** 形式を使用すると、**encapsulation** オプションは無視されます。
- トランクポート上のすべての VLAN をモニターするには、**no monitor session** *session_number* **filter** グローバル コンフィギュレーション コマンドを使用します。

RSPAN 設定時の注意事項

- すべての SPAN 設定時の注意事項が RSPAN に適用されます。
- RSPAN VLAN には特性があるので、RSPAN VLAN として使用するためにネットワーク上の VLAN をいくつか確保し、それらの VLAN にはアクセス ポートを割り当てないでおく必要があります。
- RSPAN トラフィックに出力 ACL を適用して、特定のパケットを選択的にフィルタリングまたはモニターできます。RSPAN 送信元devices内の RSPAN VLAN 上で、これらの ACL を指定します。

- RSPAN を設定する場合は、送信元ポートおよび宛先ポートをネットワーク内の複数の devicesに分散させることができます。
- RSPAN VLAN 上のアクセス ポート（音声 VLAN ポートを含む）は、非アクティブ ステートになります。
- 次の条件を満たす限り、任意の VLAN を RSPAN VLAN として設定できます。
  - すべてのdevicesで、RSPAN セッションに同じ RSPAN VLAN が使用されている。
  - 参加しているすべてのdevicesで RSPAN がサポートされている。

FSPAN および FRSPAN 設定時の注意事項

- 少なくとも 1 つの FSPAN ACL が接続されている場合、FSPAN はイネーブルになります。
- SPAN セッションに空ではない FSPAN ACL を少なくとも 1 つ接続し、ほかの 1 つまたは複数の FSPAN ACL を接続しなかった場合（たとえば、空ではない IPv4 ACL を接続し、IPv6 と MAC ACL を接続しなかった場合）、FSPAN は、接続されていない ACL によってフィルタリングされたと思われるトラフィックをブロックします。したがって、このトラフィックは監視されません。

SPAN および RSPAN の設定方法

ローカル SPAN セッションの作成

SPAN セッションを作成し、送信元（監視対象）ポートまたは VLAN、および宛先（監視側）ポートを指定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** { **interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** { **interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} 例： `スイッチ(config)# no monitor session all`	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 4 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカルセッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 4	**monitor session** *session_number* **source** { **interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] 例： `スイッチ(config)# monitor session 1 source interface gigabitethernet1/0/1`	SPANセッションおよび送信元ポート（監視対象ポート）を指定します。 • *session_number* の範囲は、1 ～ 4 です。 • *interface-id* には、モニタリングする送信元ポートを指定します。有効なインターフェイスには、物理インターフェイスおよびポートチャネル論理インターフェイス（**port-channel** *port-channel-number*）があります。有効なポートチャネル番号は 1 ～ 6 です。 • *vlan-id* には、監視する送信元 VLAN を指定します。指定できる範囲は 1 ～ 4094 です（RSPAN VLAN は除く）。 （注） 1 つのセッションに、一連のコマンドで定義された複数の送信元（ポートまたは VLAN）を含めることができます。ただし、1 つのセッション内では送信元ポートと送信元 VLAN を併用できません。

	コマンドまたはアクション	目的
		• （任意）**[,|-]** には、一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。 • （任意）**both	rx	tx**：監視するトラフィックの方向を指定します。トラフィックの方向を指定しなかった場合、送信元インターフェイスは送信トラフィックと受信トラフィックの両方を送信します。 • **both**：受信トラフィックと送信トラフィックの両方を監視します。 • **rx**：受信トラフィックをモニターします。 • **tx**：送信トラフィックをモニターします。 （注） **monitor session** *session_number***source** コマンドを複数回使用すると、複数の送信元ポートを設定できます。
ステップ 5	**monitor session** *session_number* **destination** { **interface** *interface-id* **[,**	-] [**encapsulation replicate**]} 例： `スイッチ(config)# monitor session 1 destination interface gigabitethernet1/0/2 encapsulation replicate`	SPANセッションおよび宛先ポート（モニター側ポート）を指定します。設定変更が有効になると、ポートのLEDがオレンジ色に変わります。LEDはSPAN宛先の設定を削除した後にのみ、元の状態（緑色）に戻ります。 （注） ローカルSPANの場合は、送信元および宛先インターフェイスに同じセッション番号を使用する必要があります。 • *session_number* には、ステップ 4 で入力したセッション番号を指定します。 • *interface-id* には、宛先ポートを指定します。宛先インターフェイスには物理ポートを指定する必要があります。EtherChannel や VLAN は指定できません。 • （任意）**[,	-]** には、一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。

	コマンドまたはアクション	目的
		（任意）**encapsulation replicate** には、宛先インターフェイスが送信元インターフェイスのカプセル化方式を複製することを指定します。選択しない場合のデフォルトは、ネイティブ形式（タグなし）でのパケットの送信です。 （注） **monitor session** *session_number* **destination** コマンドを複数回使用すると、複数の送信元ポートを設定できます。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ローカル SPAN セッションの作成および着信トラフィックの設定

SPAN セッションを作成し、さらに送信元ポートまたは VLAN および宛先ポートを指定した後、宛先ポートでネットワーク セキュリティ デバイス（Cisco IDS センサー装置等）用に着信トラフィックをイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** { **interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** { **interface** *interface-id* [**,** | **-**] [**encapsulation replicate**[**ingress** { **dot1q vlan** *vlan-id* **| untagged vlan** *vlan-id* **| vlan** *vlan-id*}]}
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} 例： スイッチ(config)# **no monitor session all**	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 4 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカルセッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 4	**monitor session** *session_number* **source** { **interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] 例： スイッチ(config)# **monitor session 2 source gigabitethernet0/1 rx**	SPANセッションおよび送信元ポート（監視対象ポート）を指定します。
ステップ 5	**monitor session** *session_number* **destination** { **interface** *interface-id* [**,** | **-**] [**encapsulation replicate**[**ingress** { **dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]} 例： スイッチ(config)# **monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate ingress dot1q vlan 6**	SPAN セッション、宛先ポート、パケットカプセル化、および入力 VLAN とカプセル化を指定します。 • *session_number* には、ステップ 4 で入力したセッション番号を指定します。 • *interface-id* には、宛先ポートを指定します。宛先インターフェイスには物理ポートを指定する必要があります。EtherChannel や VLAN は指定できません。 • （任意）[**,** | **-**]：一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマまたはハイフンの前後にスペースを 1 つずつ入力します。

	コマンドまたはアクション	目的
		• （任意）**encapsulation replicate**：宛先インターフェイスが送信元インターフェイスのカプセル化方式を複製することを指定します。選択しない場合のデフォルトは、ネイティブ形式（タグなし）でのパケットの送信です。 • **ingress**：宛先ポートでの着信トラフィックの転送をイネーブルにして、カプセル化タイプを指定します。 • **dot1q vlan** *vlan-id*：デフォルトの VLAN として指定した VLAN で、IEEE 802.1Q でカプセル化された着信パケットを受け入れます。 • **untagged vlan** *vlan-id* または **vlan** *vlan vlan-id*：デフォルトの VLAN として指定した VLAN で、タグなしでカプセル化された着信パケットを受け入れます。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

フィルタリングする VLAN の指定

SPAN 送信元トラフィックを特定の VLAN に制限するには、次の手順を実行します。

手順の概要

1. **enable**

2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source interface** *interface-id*
5. **monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]
6. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} 例： `スイッチ(config)# no monitor session all`	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 66 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカル セッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 4	**monitor session** *session_number* **source interface** *interface-id* 例： `スイッチ(config)# monitor session 2 source interface gigabitethernet1/0/2 rx`	送信元ポート（モニター対象ポート）と SPAN セッションの特性を指定します。 • *session_number* の範囲は、1 ～ 66 です。 • *interface-id* には、モニタリングする送信元ポートを指定します。指定したインターフェイスは、あらかじめトランク ポートとして設定しておく必要があります。

	コマンドまたはアクション	目的
ステップ 5	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] 例： `スイッチ(config)# monitor session 2 filter vlan 1 - 5 , 9`	SPAN 送信元トラフィックを特定の VLAN に制限します。 • *session_number* には、ステップ 4 で指定したセッション番号を入力します。 • *vlan-id* に指定できる範囲は 1 ～ 4094 です。 •（任意）カンマ（**,**）を使用して一連の VLAN を指定するか、ハイフン（**-**）を使用して VLAN 範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。
ステップ 6	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} 例： `スイッチ(config)# monitor session 2 destination interface gigabitethernet1/0/1`	SPANセッションおよび宛先ポート（モニター側ポート）を指定します。 • *session_number* には、ステップ 4 で入力したセッション番号を指定します。 • *interface-id* には、宛先ポートを指定します。宛先インターフェイスには物理ポートを指定する必要があります。EtherChannel や VLAN は指定できません。 •（任意）[**,**|**-**] には、一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。 •（任意）**encapsulation replicate** には、宛先インターフェイスが送信元インターフェイスのカプセル化方式を複製することを指定します。選択しない場合のデフォルトは、ネイティブ形式（タグなし）でのパケットの送信です。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。

	コマンドまたはアクション	目的
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

RSPAN VLAN としての VLAN の設定

新しい VLAN を作成し、RSPAN セッション用の RSPAN VLAN になるように設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan** *vlan-id*
4. **remote-span**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vlan** *vlan-id* 例： スイッチ(config)# **vlan 100**	VLAN ID を入力して VLAN を作成するか、または既存の VLAN の VLAN ID を入力して、VLAN コンフィギュレーション モードを開始します。指定できる範囲は 2 ～ 1001 または 1006 ～ 4094 です。 RSPAN VLAN を VLAN 1（デフォルト VLAN）または VLAN ID 1002 ～ 1005（トークンリングおよび FDDI VLAN 専用）にすることはできません。

	コマンドまたはアクション	目的
ステップ 4	**remote-span** 例： スイッチ(config-vlan)# **remote-span**	VLAN を RSPAN VLAN として設定します。
ステップ 5	**end** 例： スイッチ(config-vlan)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

RSPAN に参加するすべてのdevicesに RSPAN VLAN を作成する必要があります。RSPAN VLAN ID が標準範囲（1005 未満）であり、VTP がネットワーク内でイネーブルである場合は、1 つのdeviceに RSPAN VLAN を作成し、VTP がこの RSPAN VLAN を VTP ドメイン内の他のdevices に伝播するように設定できます。拡張範囲 VLAN（1005 を超える ID）の場合、送信元と宛先の両方のdevices、および中間devicesに RSPAN VLAN を設定する必要があります。

VTP プルーニングを使用して、RSPAN トラフィックが効率的に流れるようにするか、または RSPAN トラフィックの伝送が不要なすべてのトランクから、RSPAN VLAN を手動で削除します。

VLAN からリモート SPAN 特性を削除して、標準 VLAN に戻すように変換するには、**no remote-span** VLAN コンフィギュレーション コマンドを使用します。

SPAN セッションから送信元ポートまたは VLAN を削除するには、**no monitor session***session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} グローバル コンフィギュレーション コマンドを使用します。セッションから RSPAN VLAN を削除するには、**no monitor session** *session_number* **destination remote vlan** *vlan-id* コマンドを使用します。

RSPAN 送信元セッションの作成

RSPAN送信元セッションを作成および開始し、モニター対象の送信元および宛先RSPAN VLAN を指定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no monitor session** {*session_number*	**all**	**local**	**remote**} 例： `スイッチ(config)# no monitor session 1`	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 66 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカルセッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 4	**monitor session** *session_number* **source** {**interface** *interface-id*	**vlan** *vlan-id*} [**,**	**-**] [**both**	**rx**	**tx**] 例： `スイッチ(config)# monitor session 1 source interface gigabitethernet1/0/1 tx`	RSPAN セッションおよび送信元ポート（モニター対象ポート）を指定します。 • *session_number* の範囲は、1 ～ 66 です。 • RSPAN セッションの送信元ポートまたは送信元 VLAN を入力します。 • *interface-id* には、モニタリングする送信元ポートを指定します。有効なインターフェイスには、物理インターフェイスおよび

	コマンドまたはアクション	目的
		ポートチャネル論理インターフェイス（**port-channel** *port-channel-number*）があります。有効なポートチャネル番号は 1 ～ 48 です。 • *vlan-id* には、モニターする送信元 VLAN を指定します。指定できる範囲は 1 ～ 4094 です（RSPAN VLAN は除く）。 1 つのセッションに、一連のコマンドで定義された複数の送信元（ポートまたは VLAN）を含めることができます。ただし、1 つのセッション内で送信元ポートと送信元 VLAN を併用することはできません。 • （任意）**[,** | **-]**：一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。 • （任意）**both** | **rx** | **tx**：監視するトラフィックの方向を指定します。トラフィックの方向を指定しなかった場合、送信元インターフェイスは送信トラフィックと受信トラフィックの両方を送信します。 • **both**：受信トラフィックと送信トラフィックの両方を監視します。 • **rx**：受信トラフィックをモニターします。 • **tx**：送信トラフィックをモニターします。
ステップ 5	**monitor session** *session_number* **destination remote vlan** *vlan-id* 例： `スイッチ(config)# monitor session 1 destination remote vlan 100`	RSPAN セッション、宛先 RSPAN VLAN、および宛先ポート グループを指定します。 • *session_number* には、ステップ 4 で指定した番号を入力します。 • *vlan-id* には、モニタリングする送信元 RSPAN VLAN を指定します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

フィルタリングする VLAN の指定

RSPAN 送信元トラフィックを特定の VLAN に制限するように RSPAN 送信元セッションを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source interface** *interface-id*
5. **monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]
6. **monitor session** *session_number* **destination remote vlan** *vlan-id*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} 例： `スイッチ(config)# no monitor session 2`	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 66 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカルセッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 4	**monitor session** *session_number* **source interface** *interface-id* 例： `スイッチ(config)# monitor session 2 source interface gigabitethernet1/0/2 rx`	送信元ポート（モニター対象ポート）と SPAN セッションの特性を指定します。 • *session_number* の範囲は、1 ～ 66 です。 • *interface-id* には、モニタリングする送信元ポートを指定します。指定したインターフェイスは、あらかじめトランクポートとして設定しておく必要があります。
ステップ 5	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] 例： `スイッチ(config)# monitor session 2 filter vlan 1 - 5 , 9`	SPAN 送信元トラフィックを特定の VLAN に制限します。 • *session_number* には、ステップ 4 で指定したセッション番号を入力します。 • *vlan-id* に指定できる範囲は 1 ～ 4094 です。 • （任意）**,**|**-** カンマ（**,**）を使用して一連の VLAN を指定するか、ハイフン（**-**）を使用して VLAN 範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。
ステップ 6	**monitor session** *session_number* **destination remote vlan** *vlan-id* 例： `スイッチ(config)# monitor session 2 destination remote vlan 902`	RSPAN セッションおよび宛先リモート VLAN（RSPAN VLAN）を指定します。 • *session_number* には、ステップ 4 で指定したセッション番号を入力します。 • *vlan-id* には、宛先ポートにモニタ対象トラフィックを伝送する RSPAN VLAN を指定します。
ステップ 7	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	
ステップ 8	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

RSPAN 宛先セッションの作成

RSPAN 宛先セッションは、別のdeviceまたはdevice スタック（送信元セッションが設定されていないdeviceまたはdevice スタック）に設定します。

このdevice上で RSPAN VLAN を定義し、RSPAN 宛先セッションを作成し、送信元 RSPAN VLAN および宛先ポートを指定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan** *vlan-id*
4. **remote-span**
5. **exit**
6. **no monitor session** {*session_number* | **all** | **local** | **remote**}
7. **monitor session** *session_number* **source remote vlan** *vlan-id*
8. **monitor session** *session_number* **destination interface** *interface-id*
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vlan** *vlan-id* 例： スイッチ(config)# **vlan 901**	送信元deviceで作成された RSPAN VLAN の VLAN ID を指定し、VLAN コンフィギュレーション モードを開始します。 両方のdevicesが VTP に参加し、RSPAN VLAN ID が 2 ～ 1005 である場合は、VTP ネットワークを介して RSPAN VLAN ID が伝播されるため、ステップ 3 ～ 5 は不要です。
ステップ 4	**remote-span** 例： スイッチ(config-vlan)# **remote-span**	VLAN を RSPAN VLAN として識別します。
ステップ 5	**exit** 例： スイッチ(config-vlan)# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**no monitor session** {*session_number* | **all** | **local** | **remote**} 例： スイッチ(config)# **no monitor session 1**	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 66 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカル セッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 7	**monitor session** *session_number* **source remote vlan** *vlan-id* 例： スイッチ(config)# **monitor session 1 source remote vlan 901**	RSPAN セッションと送信元 RSPAN VLAN を指定します。 • *session_number* の範囲は、1 ～ 66 です。 • *vlan-id* には、モニタリングする送信元 RSPAN VLAN を指定します。

	コマンドまたはアクション	目的
ステップ 8	**monitor session** *session_number* **destination interface** *interface-id* 例： `スイッチ(config)# monitor session 1 destination interface gigabitethernet2/0/1`	RSPAN セッションと宛先インターフェイスを指定します。 • *session_number* には、ステップ 7 で指定した番号を入力します。 RSPAN 宛先セッションでは、送信元 RSPAN VLAN および宛先ポートに同じセッション番号を使用する必要があります。 • *interface-id* には、宛先インターフェイスを指定します。宛先インターフェイスは物理インターフェイスでなければなりません。 • **encapsulation replicate** はコマンドラインのヘルプストリングに表示されますが、RSPAN ではサポートされていません。元の VLAN ID は RSPAN VLAN ID によって上書きされ、宛先ポート上のすべてのパケットはタグなしになります。
ステップ 9	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

RSPAN 宛先セッションの作成および着信トラフィックの設定

RSPAN 宛先セッションを作成し、送信元 RSPAN VLAN および宛先ポートを指定し、宛先ポートでネットワーク セキュリティ デバイス（Cisco IDS センサー装置等）用に着信トラフィックをイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source remote vlan** *vlan-id*
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** { **dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no monitor session** {*session_number*	**all**	**local**	**remote**} 例： `スイッチ(config)# no monitor session 2`	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 66 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカルセッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 4	**monitor session** *session_number* **source remote vlan** *vlan-id* 例： `スイッチ(config)# monitor session 2 source remote vlan 901`	RSPAN セッションと送信元 RSPAN VLAN を指定します。 • *session_number* の範囲は、1 ～ 66 です。 • *vlan-id* には、モニタリングする送信元 RSPAN VLAN を指定します。

	コマンドまたはアクション	目的
ステップ 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** { **dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* **| vlan** *vlan-id*}]} 例： `スイッチ(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6`	SPAN セッション、宛先ポート、パケット カプセル化、および着信 VLAN とカプセル化を指定します。 • *session_number* には、ステップ 5 で指定した番号を入力します。 RSPAN 宛先セッションでは、送信元 RSPAN VLAN および宛先ポートに同じセッション番号を使用する必要があります。 • *interface-id* には、宛先インターフェイスを指定します。宛先インターフェイスは物理インターフェイスでなければなりません。 • **encapsulation replicate** はコマンドラインのヘルプストリングに表示されますが、RSPAN ではサポートされていません。元の VLAN ID は RSPAN VLAN ID によって上書きされ、宛先ポート上のすべてのパケットはタグなしになります。 •（任意）[**,** | **-**] には、一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。 • 宛先ポートでの着信トラフィックの転送をイネーブルにして、カプセル化タイプを指定するには、**ingress** を追加のキーワードと一緒に入力します。 • **dot1q vlan** *vlan-id*：デフォルトの VLAN として指定した VLAN で、IEEE 802.1Q でカプセル化された着信パケットを転送します。 • **untagged vlan** *vlan-id* または **vlan** *vlan-id*：デフォルトの VLAN として指定した VLAN で、タグなしでカプセル化された着信パケットを転送します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

FSPAN セッションの設定

SPAN セッションを作成し、送信元（監視対象）ポートまたは VLAN、および宛先（モニタ側）ポートを指定し、セッションに FSPAN を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** { **interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
6. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} 例： スイッチ(config)# **no monitor session 2**	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 66 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカルセッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 4	**monitor session** *session_number* **source** { **interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] 例： スイッチ(config)# **monitor session 2 source interface gigabitethernet1/0/1**	SPAN セッションおよび送信元ポート（モニター対象ポート）を指定します。 • *session_number* の範囲は、1 ～ 66 です。 • *interface-id* には、モニタリングする送信元ポートを指定します。有効なインターフェイスには、物理インターフェイスおよびポートチャネル論理インターフェイス（**port-channel** *port-channel-number*）があります。有効なポートチャネル番号は 1 ～ 48 です。 • *vlan-id* には、監視する送信元 VLAN を指定します。指定できる範囲は 1 ～ 4094 です（RSPAN VLAN は除く）。 （注） 1 つのセッションに、一連のコマンドで定義された複数の送信元（ポートまたは VLAN）を含めることができます。ただし、1 つのセッション内では送信元ポートと送信元 VLAN を併用できません。 • （任意）[**,** | **-**]：一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。 • （任意）[**both** | **rx** | **tx**]：モニタリングするトラフィックの方向を指定します。トラフィックの方向を指定しなかった場合、SPAN は送信トラフィックと受信トラフィックの両方をモニターします。

	コマンドまたはアクション	目的
		• **both**：送信トラフィックと受信トラフィックの両方を監視します。これはデフォルトです。 • **rx**：受信トラフィックをモニターします。 • **tx**：送信トラフィックをモニターします。 （注） **monitor session** *session_number***source** コマンドを複数回使用すると、複数の送信元ポートを設定できます。
ステップ 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} 例： スイッチ(config)# **monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate**	SPANセッションおよび宛先ポート（モニター側ポート）を指定します。 • *session_number* には、ステップ 4 で入力したセッション番号を指定します。 • **destination** では、次のパラメータを指定します。 • *interface-id* には、宛先ポートを指定します。宛先インターフェイスには物理ポートを指定する必要があります。EtherChannel や VLAN は指定できません。 • （任意）**[,|-]** には、一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。 • （任意）**encapsulation replicate** には、宛先インターフェイスが送信元インターフェイスのカプセル化方式を複製することを指定します。選択しない場合のデフォルトは、ネイティブ形式（タグなし）でのパケットの送信です。 （注） ローカル SPAN の場合は、送信元および宛先インターフェイスに同じセッション番号を使用する必要があります。 **monitor session** *session_number* **destination** コマンドを複数回使用すると、複数の送信元ポートを設定できます。

	コマンドまたはアクション	目的
ステップ 6	**monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*} 例： `スイッチ(config)# monitor session 2 filter ipv6 access-group 4`	SPAN セッション、フィルタリングするパケットのタイプ、および FSPAN セッションで使用する ACL を指定します。 • *session_number* には、ステップ 4 で入力したセッション番号を指定します。 • *access-list-number* には、トラフィックのフィルタリングに使用したい ACL 番号を指定します。 • *name* には、トラフィックのフィルタリングに使用する ACL の名前を指定します。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

FRSPAN セッションの設定

RSPAN 送信元セッションを開始し、監視対象の送信元および宛先 RSPAN VLAN を指定し、セッションに FRSPAN を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** { **interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **vlan** *vlan-id*
7. **remote-span**

8. **exit**
9. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} 例： `スイッチ(config)# no monitor session 2`	セッションに対する既存の SPAN 設定を削除します。 • *session_number* の範囲は、1 ～ 66 です。 • **all**：すべての SPAN セッションを削除します。 • **local**：すべてのローカル セッションを削除します。 • **remote**：すべてのリモート SPAN セッションを削除します。
ステップ 4	**monitor session** *session_number* **source** { **interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] 例： `スイッチ(config)# monitor session 2 source interface gigabitethernet1/0/1`	SPAN セッションおよび送信元ポート（モニター対象ポート）を指定します。 • *session_number* の範囲は、1 ～ 66 です。 • *interface-id* には、モニタリングする送信元ポートを指定します。有効なインターフェイスには、物理インターフェイスおよびポートチャネル論理インターフェイス（**port-channel** *port-channel-number*）があります。有効なポートチャネル番号は 1 ～ 48 です。 • *vlan-id* には、監視する送信元 VLAN を指定します。指定できる範囲は 1 ～ 4094 です（RSPAN VLAN は除く）。

	コマンドまたはアクション	目的
		（注） 1 つのセッションに、一連のコマンドで定義された複数の送信元（ポートまたは VLAN）を含めることができます。ただし、1 つのセッション内では送信元ポートと送信元 VLAN を併用できません。 • （任意）[**,**|**-**]：一連のインターフェイスまたはインターフェイスの範囲を指定します。カンマの前後およびハイフンの前後にスペースを 1 つずつ入力します。 • （任意）[**both** | **rx** | **tx**]：モニタリングするトラフィックの方向を指定します。トラフィックの方向を指定しなかった場合、SPAN は送信トラフィックと受信トラフィックの両方をモニターします。 • **both**：送信トラフィックと受信トラフィックの両方をモニターします。これはデフォルトです。 • **rx**：受信トラフィックをモニターします。 • **tx**：送信トラフィックをモニターします。 （注） **monitor session** *session_number***source** コマンドを複数回使用すると、複数の送信元ポートを設定できます。
ステップ 5	**monitor session** *session_number* **destination remote vlan** *vlan-id* 例： `スイッチ(config)# monitor session 2 destination remote vlan 5`	RSPAN セッションと宛先 RSPAN VLAN を指定します。 • *session_number* には、ステップ 4 で指定した番号を入力します。 • *vlan-id* には、モニタリングする宛先 RSPAN VLAN を指定します。
ステップ 6	**vlan** *vlan-id* 例： `スイッチ(config)# vlan 10`	VLAN コンフィギュレーション モードを開始します。*vlan-id* には、モニタリングする送信元 RSPAN VLAN を指定します。
ステップ 7	**remote-span** 例：	ステップ 5 で指定した VLAN が RSPAN VLAN の一部であることを指定します。

	コマンドまたはアクション	目的
	スイッチ(config-vlan)# **remote-span**	
ステップ 8	**exit** 例： スイッチ(config-vlan)# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 9	**monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*} 例： スイッチ(config)# **monitor session 2 filter ip access-group 7**	RSPAN セッション、フィルタリングするパケットのタイプ、および FRSPAN セッションで使用する ACL を指定します。 • *session_number* には、ステップ 4 で入力したセッション番号を指定します。 • *access-list-number* には、トラフィックのフィルタリングに使用したいACL番号を指定します。 • *name* には、トラフィックのフィルタリングに使用する ACL の名前を指定します。
ステップ 10	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 11	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 12	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SPAN および RSPAN 動作のモニタリング

次の表で、SPAN および RSPAN 動作の設定と結果を表示して動作をモニタするために使用するコマンドについて説明します。

表 *76 : SPAN* および *RSPAN* 動作のモニタリング

コマンド	目的
show monitor	現在の SPAN、RSPAN、FSPAN、または FRSPAN 設定を表示します。

SPAN および RSPAN の設定例

例：ローカル SPAN の設定

次に、SPAN セッション 1 を設定し、宛先ポートへ向けた送信元ポートのトラフィックをモニタする例を示します。最初に、セッション 1 の既存の SPAN 設定を削除し、カプセル化方式を維持しながら、双方向トラフィックを送信元ポート GigabitEthernet 1 から宛先ポート GigabitEthernet 2 にミラーリングします。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# no monitor session 1
スイッチ(config)# monitor session 1 source interface gigabitethernet1/0/1
スイッチ(config)# monitor session 1 destination interface gigabitethernet1/0/2
encapsulation replicate
スイッチ(config)# end
```

次に、SPAN セッション 1 の SPAN 送信元としてのポート 1 を削除する例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# no monitor session 1 source interface gigabitethernet1/0/1
スイッチ(config)# end
```

次に、双方向モニタが設定されていたポート 1 で、受信トラフィックのモニタをディセーブルにする例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# no monitor session 1 source interface gigabitethernet1/0/1 rx
```

ポート 1 で受信するトラフィックのモニタはディセーブルになりますが、このポートから送信されるトラフィックは引き続きモニタされます。

次に、SPAN セッション 2 内の既存の設定を削除し、VLAN 1 ～ 3 に属するすべてのポートで受信トラフィックをモニタするように SPAN セッション 2 を設定し、モニタされたトラフィックを宛先ポート GigabitEthernet 2 に送信する例を示します。さらに、この設定は VLAN 10 に属するすべてのポートですべてのトラフィックをモニタするよう変更されます。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# no monitor session 2
スイッチ(config)# monitor session 2 source vlan 1 - 3 rx
スイッチ(config)# monitor session 2 destination interface gigabitethernet1/0/2
スイッチ(config)# monitor session 2 source vlan 10
スイッチ(config)# end
```

次に、SPAN セッション 2 の既存の設定を削除し、ギガビット イーサネット ソース送信元ポート 1 上で受信されるトラフィックをモニタするように SPAN セッション 2 を設定し、そのトラフィックを送信元ポートと同じ出力カプセル化方式の宛先ギガビット イーサネット ポート 2 に送信し、デフォルト入力 VLAN として VLAN 6 を使用した入力転送をイネーブルにする例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# no monitor session 2
スイッチ(config)# monitor session 2 source gigabitethernet0/1 rx
スイッチ(config)# monitor session 2 destination interface gigabitethernet0/2 encapsulation
 replicate ingress vlan 6
スイッチ(config)# end
```

次に、SPAN セッション 2 の既存の設定を削除し、トランク ポート GigabitEthernet 2 で受信されたトラフィックをモニターするように SPAN セッション 2 を設定し、VLAN 1 ～ 5 および 9 に対してのみトラフィックを宛先ポート GigabitEthernet 1 に送信する例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# no monitor session 2
スイッチ(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
スイッチ(config)# monitor session 2 filter vlan 1 - 5 , 9
スイッチ(config)# monitor session 2 destination interface gigabitethernet1/0/1
スイッチ(config)# end
```

例：RSPAN VLAN の作成

この例は、RSPAN VLAN 901 の作成方法を示しています。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# vlan 901
スイッチ(config-vlan)# remote span
スイッチ(config-vlan)# end
```

次に、セッション 1 に対応する既存の RSPAN 設定を削除し、複数の送信元インターフェイスをモニタするように RSPAN セッション 1 を設定し、さらに宛先を RSPAN VLAN 901 に設定する例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# no monitor session 1
スイッチ(config)# monitor session 1 source interface gigabitethernet1/0/1 tx
スイッチ(config)# monitor session 1 source interface gigabitethernet1/0/2 rx
スイッチ(config)# monitor session 1 source interface port-channel 2
スイッチ(config)# monitor session 1 destination remote vlan 901
スイッチ(config)# end
```

次に、RSPAN セッション 2 の既存の設定を削除し、トランク ポート 2 で受信されるトラフィックをモニタするように RSPAN セッション 2 を設定し、VLAN 1 ～ 5 および 9 に対してのみトラフィックを宛先 RSPAN VLAN 902 に送信する例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# no monitor session 2
スイッチ(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
スイッチ(config)# monitor session 2 filter vlan 1 - 5 , 9
スイッチ(config)# monitor session 2 destination remote vlan 902
スイッチ(config)# end
```

次に、送信元リモート VLAN として VLAN 901、宛先インターフェイスとしてポート 1 を設定する例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# monitor session 1 source remote vlan 901
スイッチ(config)# monitor session 1 destination interface gigabitethernet2/0/1
スイッチ(config)# end
```

次に、RSPAN セッション 2 で送信元リモート VLAN として VLAN 901 を設定し、送信元ポート GigabitEthernet 2 を宛先インターフェイスとして設定し、VLAN 6 をデフォルトの受信 VLAN として着信トラフィックの転送をイネーブルにする例を示します。

```
スイッチ> enable
スイッチ# configure terminal
スイッチ(config)# monitor session 2 source remote vlan 901
スイッチ(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress
vlan 6
スイッチ(config)# end
```

第 41 章

RMON の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

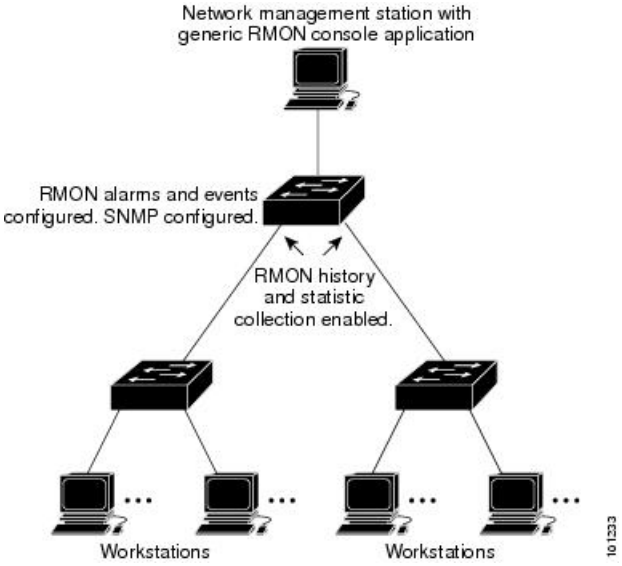
RMON について

RMON の概要

RMON とは Internet Engineering Task Force（IETF）の標準モニタリング仕様の 1 つで、RMON 準拠のコンソール システムとネットワーク プローブ間で交換可能な一連の統計情報と機能を定義します。RMONによって、総合的なネットワーク障害診断、プランニング、パフォーマンス チューニングに関する情報が得られます。

次の図に、deviceでの RMON 機能と Simple Network Management Protocol（SNMP）エージェントの構成例を示します。この例では、接続されているすべての LAN セグメント上のすべての devices間のすべてのトラフィックをモニターします。

図 *69 :* リモート モニタリングの例



deviceは次の RMON グループ（RFC 1757 で規定）をサポートしています。

- 統計情報（RMON グループ 1）：インターフェイス上のイーサネットの統計情報（device タイプとサポートされているインターフェイスに応じた、ファストイーサネットやギガビットイーサネット統計情報など）を収集します。
- 履歴（RMON グループ 2）：指定されたポーリング間隔で、イーサネットポート上の統計情報（deviceタイプとサポートされているインターフェイスに応じた、ファストイーサネットやギガビットイーサネット統計情報など）の履歴グループを収集します。
- アラーム（RMON グループ 3）：指定された期間、特定の管理情報ベース（MIB）オブジェクトをモニタリングし、指定された値（上限しきい値）でアラームを発生し、別の値（下限しきい値）でアラームをリセットします。アラームはイベントと組み合わせて使用できます。アラームがイベントを発生させ、イベントによってログエントリまたはSNMP トラップが生成されるようにできます。
- イベント（RMON グループ 9）：アラームによってイベントが発生した際のアクションを指定します。アクションは、ログ エントリまたは SNMP トラップを生成できます。

このソフトウェアリリースがサポートするdevicesは、RMONデータの処理にハードウェアカウンタを使用するので、モニターが効率的になり、処理能力はほとんど必要ありません。

（注） 64 ビット カウンタは、RMON アラームではサポートされていません。

RMON の設定方法

RMON のデフォルト設定

RMONは、デフォルトではディセーブルに設定されています。アラームまたはイベントは設定されていません。

RMON アラームおよびイベントの設定

始める前に

スイッチを RMON 対応として設定するには、コマンドライン インターフェイス（CLI）または SNMP 準拠のネットワーク管理ステーションを使用します。

（注） 64 ビット カウンタは、RMON アラームではサポートされていません。

RMON アラームおよびイベントをイネーブルにするには、次の手順を実行します。

- ネットワーク管理ステーション（NMS）上で汎用 RMON コンソール アプリケーションを使用し、RMON のネットワーク管理機能を利用することを推奨します。
- RMON MIB オブジェクトにアクセスするために、スイッチ上で SNMP を設定することも必要です。

手順の概要

1. **enable**
2. **configure terminal**
3. **rmon alarm** {*number variable interval* **absolute** | **delta** } **rising-threshold***value [event-number]* **falling-threshold** *value [event-number]* [**owner***string*]
4. **rmon event** *number* [**description** *string*] [**log**] [**owner** *string*] [**trap** *community*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**rmon alarm** {*number variable interval* **absolute** | **delta** } **rising-threshold***value [event-number]* **falling-threshold** *value [event-number]* [**owner***string*] 例： `Switch(config)# rmon alarm 10 ifEntry.20.1 20 delta rising-threshold 15 1 falling-threshold 0 owner jjohnson`	MIB オブジェクトにアラームを設定します。 *number* には、アラーム番号を指定します。指定できる範囲は 1 ～ 65535 です。 *variable* には、モニタ対象の MIB オブジェクトを指定します。 *interval* には、アラームが MIB 変数をモニタする時間を秒数で指定します。値の範囲は 1 ～ 4294967295 秒です。 各 MIB 変数を直接テストする場合は、**absolute** キーワードを指定します。MIB 変数のサンプル間の変動をテストする場合は、**delta** キーワードを指定します。 *value* には、アラームを発生させる値およびアラームがリセットされる値を指定します。**rising threshold** および **falling threshold** の値の範囲は -2147483648 ～ 2147483647 です。 （任意）*event-number* には、上限および下限しきい値が限度を超えた場合に発生させるイベントの番号を指定します。 （任意）**owner** *string* には、アラームの所有者を指定します。
ステップ 4	**rmon event** *number* [**description** *string*] [**log**] [**owner** *string*] [**trap** *community*] 例： `スイッチ(config)# rmon event 1 log trap eventtrap description "High ifOutErrors" owner jjones`	RMON イベント テーブルで RMON イベント番号に関連付けられたイベントを追加します。 *number* には、イベント番号を割り当てます。指定できる範囲は 1 ～ 65535 です。 （任意）**description** *string* には、イベントの説明を指定します。 （任意）イベント発生時に RMON ログエントリを生成する場合は、**log** キーワードを使用します。 （任意）**owner** *string* には、イベントの所有者を指定します。 （任意）**trap** *community* には、このトラップに使用する SNMP コミュニティストリングを入力します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

アラームをディセーブルにするには、設定した各アラームに対して、**no rmon alarm** *number* グローバル コンフィギュレーション コマンドを使用します。設定したすべてのアラームを一度にディセーブルにすることはできません。イベントをディセーブルにするには、**no rmon event** *number* グローバル コンフィギュレーション コマンドを使用します。

インターフェイス上でのグループ履歴統計情報の収集

インターフェイス上でグループ履歴統計情報を収集するには、次の手順を実行します。この手順は任意です。

始める前に

収集情報を表示するには、最初にRMONアラームおよびイベントを設定する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **rmon collection history** *index* [**buckets** *bucket-number*] [**interval** *seconds*] [**owner** *ownername*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet2/0/1**	履歴を収集するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**rmon collection history** *index* [**buckets** *bucket-number*] [**interval** *seconds*] [**owner** *ownername*] 例：	指定したバケット数と期間での履歴収集をイネーブルにします。 *index* には、RMON 統計グループを指定します。指定できる範囲は 1 ～ 65535 です。 （任意）**buckets** *bucket-number* には、RMON 統計グループ履歴収集に必要な最大バケット数を指定します。指定できる範囲は 1 ～ 65535 です。デフォルトのバケット数は 50 です。 （任意）**interval** *seconds* には、ポーリングサイクルを秒数で指定します。指定できる範囲は 1 ～ 3600 です。デフォルトは 1,800 秒です。 （任意）**owner** *ownername* には、RMON 統計グループの所有者名を入力します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。

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

次のタスク

履歴収集をディセーブルにするには、**no rmon collection history** *index* インターフェイス コンフィギュレーション コマンドを使用します。

インターフェイス上でのイーサネット グループ統計情報の収集

インターフェイス上でグループ イーサネット統計情報を収集するには、次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **rmon collection stats** *index* [**owner** *ownername*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet2/0/1**	統計情報を収集するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**rmon collection stats** *index* [**owner** *ownername*] 例： `スイッチ(config-if)# rmon collection stats 2 owner root`	インターフェイスの RMON 統計情報収集をイネーブルにします。 *index* には、RMON 統計グループを指定します。有効な範囲は 1 ～ 65535 です。 （任意）**owner** *ownername* には、RMON 統計グループの所有者名を入力します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

イーサネット統計グループの収集をディセーブルにするには、**no rmon collection stats** *index* インターフェイス コンフィギュレーション コマンドを使用します。

RMON ステータスのモニタリング

表 *77 : RMON* ステータスを表示するコマンド

コマンド	目的
show rmon	汎用 RMON 統計情報を表示します。
show rmon alarms	RMON アラーム テーブルを表示します。
show rmon events	RMON イベント テーブルを表示します。
show rmon history	RMON 履歴テーブルを表示します。
show rmon statistics	RMON 統計情報テーブルを表示します。

第 42 章

Embedded Event Manager の設定

Embedded Event Manager について

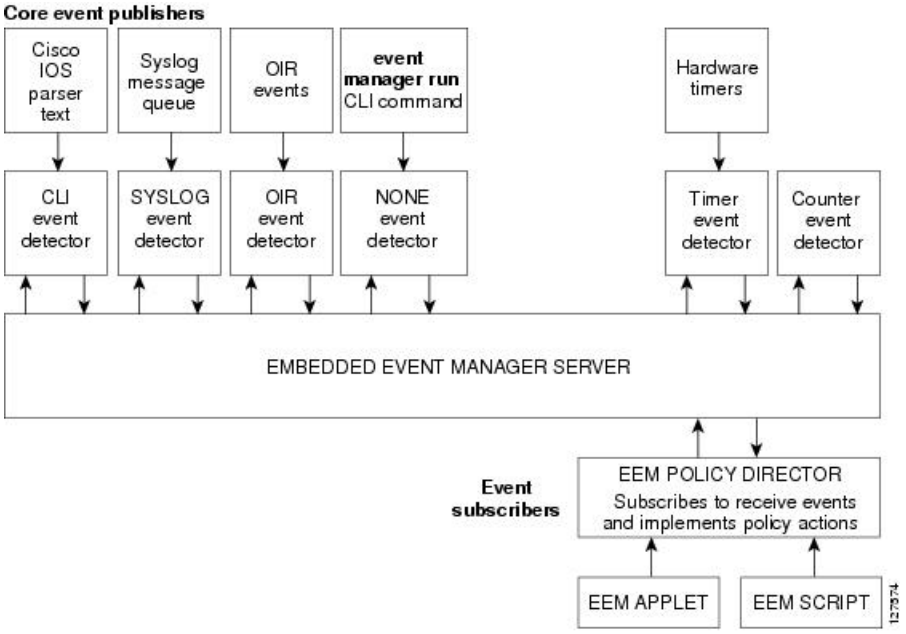
Embedded Event Manager の概要

Embedded Event Manager（EEM）は、Cisco IOS デバイス内でイベント検出および回復のために配布されカスタマイズされたアプローチです。EEM はイベントを監視する機能を提供します。また、監視されたイベントが発生するかしきい値に達した場合に情報を得たり、是正措置を行ったり、または他の EEM 処理を実行したりする機能も提供します。EEM ポリシーは、イベントおよびイベントが発生した場合に行う処理を定義します。

EEM はキー システムのイベントを監視し、セット ポリシーを通してイベントに作用します。このポリシーはプログラムされたスクリプトで、これを使用して、発生した特定の一連のイベントに基づいて処理を呼び出すように、スクリプトをカスタマイズできます。このスクリプトは、カスタム Syslog または簡易ネットワーク管理プロトコル（SNMP）トラップの生成、CLI（コマンドライン インターフェイス）コマンドの呼び出し、フェールオーバーの強制などの処理を生成します。スイッチからすべてのイベント管理を管理できるわけではなく、何らかの問題によって、スイッチと外部ネットワーク管理デバイス間の通信に障害が発生することがあるため、EEM のイベント管理機能は役立ちます。スイッチをリブートすることなく自動回復処理が行われる場合、ネットワークのアベイラビリティは向上します。

次に、EEM サーバ、コア イベント パブリッシャ（イベント検出器）、およびイベント サブスクライバ（ポリシー）の関係の例を示します。イベント パブリッシャはイベントを選別し、イベント サブスクライバによって提供されたイベント仕様と一致するイベントがいつ発生するかを決定します。イベントが発生すると、イベント検出器が EEM サーバに通知します。次に、システムの現在の状態と特定のイベントに対してポリシーで指定された処理に基づいて、EEM ポリシーが回復を実行します。

図 *70 : Embedded Event Manager* コア イベント検出器



（注） EEM をサポートするのは、Cisco Catalyst 3560-CX スイッチのみです。

EEM をサポートするのは、IP Base ライセンスおよび IP Services ライセンスを実行する Catalyst スイッチのみです。

Embedded Event Manager のアクション

イベントに応答して次の処理が発生します。

- 名前付きカウンタの修正。
- アプリケーション特有のイベントのパブリッシュ。
- SNMP トラップの生成。
- 優先化された syslog メッセージの生成。
- Cisco IOS ソフトウェアのリロード。
- スイッチ スタックのリロード。
- マスター切り替え時のマスター スイッチのリロード。この場合、新しいマスター スイッチが選択されます。

Embedded Event Manager ポリシー

EEM はイベントを監視して情報を提供するか、または監視されたイベントが発生するかしきい値に達した場合に是正措置を行うことができます。EEM ポリシーは、イベントおよびイベントが発生した場合に行う処理を定義するエンティティです。

EEM ポリシーにはアプレットとスクリプトの 2 つのタイプがあります。アプレットは、CLI 設定内で定義される簡易なポリシーです。イベントの選別基準とイベントが発生した場合に行う処理を定義する簡易な方法です。スクリプトは、ASCII エディタを使用して、ネットワーキング デバイス上で定義されます。スクリプト（バイトコード（.tbc）とテキスト（.tcl）スクリプトで作成できます）は、次に、ネットワーキング デバイスにコピーされ、EEM によって登録されます。さらに、1 つの .tcl ファイルに複数のイベントを登録できます。

EEM を使用して、EEM ポリシー ツール コマンド言語（TCL）スクリプトを使用する独自のポリシーを記述して実行します。マスター スイッチで TCL スクリプトを設定すると、ファイルはメンバー スイッチに自動的に送信されます。マスター スイッチが変わった場合に TCL スクリプト ポリシーが機能し続けるように、メンバー スイッチでユーザ定義の TCL スクリプトが使用できる必要があります。

キーワード拡張という形のシスコの TCL 機能拡張は、EEM ポリシーの開発を容易にします。これらのキーワードは、検出されたイベント、その後の処理、ユーティリティ情報、カウンタ値、およびシステム情報を識別します。

Embedded Event Manager の環境変数

EEM は EEM ポリシーで環境変数を使用します。この環境変数は、CLI コマンドおよび **event manager environment** コマンドを実行して、EEM ポリシー Tool Command Language（TCL）スクリプトで定義します。

- ユーザ定義の変数：ユーザ定義のポリシーに対して、ユーザにより定義されます。
- シスコ定義の変数：特定のサンプル ポリシーに対してシスコにより定義されます。
- シスコ組み込み変数（EEM アプレットで利用可能）：シスコにより定義され、読み取り専用または読み取りと書き込みに設定できます。読み取り専用変数は、アプレットが実行を開始する前に、システムによって設定されます。1 つの読み取りと書き込み変数 *_exit_status* により、同期イベントからトリガーされるポリシーの終了ステータスを設定できます。

シスコ定義の環境変数とシスコ システム定義の環境変数は、特定の 1 つのイベント ディテクタまたはすべてのイベントディテクタに適用されます。ユーザ定義の環境変数またはサンプル ポリシーでシスコにより定義される環境変数は、event manager environment グローバル コンフィギュレーション コマンドを使用して設定されます。ポリシーを登録する前に、変数を EEM ポリシーに定義する必要があります。

Embedded Event Manager 3.2

Embedded Event Manager 3.2 では次のイベント ディテクタがサポートされています。

- ネイバー探索：ネイバー探索イベント検出器によって、次の場合に自動ネイバー検出に応答するポリシーをパブリッシュできます。
  - Cisco Discovery Protocol（CDP）のキャッシュ エントリが追加、削除、または更新された場合。
  - リンク層検出プロトコル（LLDP）キャッシュ エントリが追加、削除、または更新された場合。
  - インターフェイスのリンク ステータスが変更された場合。
  - インターフェイスのライン ステータスが変更された場合。
- ID：ID イベント検出器は、AAA の許可および認証が成功した場合、障害が発生した場合、またはポート上で通常のユーザ トラフィックの送信が許可された後にイベントを生成します。
- Mac-Address-Table：Mac-Address-Table イベント検出器は、MAC アドレスが MAC アドレス テーブルで学習された場合にイベントを生成します。

（注） Mac-Address-Table イベント検出器は、スイッチ プラットフォームでだけサポートされており、MAC アドレスが学習されたレイヤ 2 インターフェイスだけで使用できます。レイヤ 3 インターフェイスはアドレスを学習せず、ルータは通常、学習された MAC アドレスを EFM に通知するために必要な MAC アドレス テーブル インフラストラクチャをサポートしません。

EEM 3.2 では、新しいイベント検出器で動作するアプレットをサポートするための CLI コマンドも導入されています。

Embedded Event Manager の設定方法

Embedded Event Manager アプレットの登録と定義

EEM にアプレットを登録し、**event applet** および **action applet** コンフィギュレーション コマンドを使用して EEM アプレットを定義するには、特権 EXEC モードで次の手順を実行します。

（注） EEM アプレットでは、1 つのイベント アプレット コマンドしか使用できません。複数の処理アプレット コマンドが使用できます。**no event** および **no action** コマンドを指定しない場合、コンフィギュレーション モードを終了すると、アプレットは削除されます。

手順の概要

1. **configure terminal**
2. **event manager applet***applet-name*

3. **event snmp oid** *oid-value* **get-type** {**exact**|**next**} **entry-op** { **eq**|**ge**|**gt**|**le**|**lt**|**ne**} **entry-val** *entry-val* [**exit-comb** {**or**|**and**}] [**exit-op**{**eq**|**ge**|**gt**|**le**|**lt**|**nc**}] [**exit-val** *exit-val*] [**exit-time** *exit-time-val*] **poll interval** *poll-int-val*
4. **action label syslog** [**priority** *priority-level*] **msg** *msg-text*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**event manager applet***applet-name*	EEM でアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 3	**event snmp oid** *oid-value* **get-type** {**exact**|**next**} **entry-op** { **eq**|**ge**|**gt**|**le**|**lt**|**ne**} **entry-val** *entry-val* [**exit-comb** {**or**|**and**}] [**exit-op**{**eq**|**ge**|**gt**|**le**|**lt**|**nc**}] [**exit-val** *exit-val*] [**exit-time** *exit-time-val*] **poll interval** *poll-int-val*	EEM アプレットを実行する要因となるイベント基準を指定します。 （任意）終了基準。終了基準を指定しない場合、イベント モニタリングがすぐに再イネーブル化されます。
ステップ 4	**action label syslog** [**priority** *priority-level*] **msg** *msg-text*	EEM アプレットがトリガーされたときの処理を指定します。この処理を繰り返して、アプレットに他の CLI コマンドを追加します。 • （任意）プライオリティ キーワードは、Syslog メッセージのプライオリティ レベルを指定します。選択した場合、プライオリティ レベル引数を定義する必要があります。 • *msg-text* の場合、引数は文字テキスト、環境変数、またはこの 2 つを組み合わせたものになります。
ステップ 5	**end**	アプレット コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、SNMP オブジェクト ID によって指定されたフィールドの 1 つが、定義されたしきい値を超えた場合の EEM での出力例を示します。

```
スイッチ(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
 lt entry-val 5120000 poll-interval 10
```

次に、EEM イベントに応答して行われる処理の例を示します。

```
スイッチ(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current
 available memory is $_snmp_oid_val bytes"
スイッチ (config-applet)# action 2.0 force-switchover
```

Embedded Event Manager TCL スクリプトの登録と定義

EEM で TCL スクリプトを登録し、TCL スクリプトとポリシー コマンドを定義するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **show event manager environment** [**all** | *variable-name*]
3. **configure terminal**
4. **event manager environment variable-name string**
5. **event manager policy policy-file-name** [**type system**] [**trap**]
6. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**show event manager environment** [**all** | *variable-name*]	（任意）**show event manager environment** コマンドは、EEM 環境変数の名前と値を表示します。 （任意）**all** キーワードは、EEM 環境変数を表示します。 （任意）*variable-name* 引数は、指定された環境変数に関する情報を表示します。
ステップ 3	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**event manager environment variable-name string**	指定された EEM 環境変数の値を設定します。要求されたすべての環境変数でこのステップを繰り返します。
ステップ 5	**event manager policy policy-file-name** [**type system**] [**trap**]	ポリシー内で定義された指定イベントが発生した場合に、EEMポリシーを実行するよう、定義します。
ステップ 6	**exit**	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、show event manager environment コマンドの出力例を示します。

```
スイッチ# show event manager environment all
No.   Name                Value
1     _cron_entry         0-59/2 0-23/1 * * 0-6
2     _show_cmd           show ver
3     _syslog_pattern     .*UPDOWN.*Ethernet1/0.*
```

次に、ソフトウェアによって割り当てられた CRON タイマー環境変数を毎日の毎時間、毎分、毎秒に設定する方法を示します。

```
スイッチ (config)# event manager environment_cron_entry 0-59/2 0-23/1 * * 0-6
```

次に、システム ポリシーとして登録された tm_cli_cmd.tcl という名前の EEM ポリシーの例を示します。システム ポリシーは Cisco IOS イメージの一部です。ユーザ定義の TCL スクリプトは、最初にフラッシュ メモリにコピーする必要があります。

```
スイッチ (config)# event manager policy tm_cli_cmd.tcl type system
```

Embedded Event Manager のモニタリング

Embedded Event Manager 情報の表示

表 *78 : EEM* 情報を表示するためのコマンド

コマンド	目的
show event manager environment[**all**| *variable-name*]	すべての EEM 環境変数の名前および値を表示します。

EEM 登録済みポリシーや EEM 履歴データなど、EEM に関する情報の表示については、『Cisco IOS Network Management Command Reference』を参照してください。

Embedded Event Manager の設定例

例：SNMP 通知の生成

次に、SNMP オブジェクト ID によって指定されたフィールドの 1 つが定義されたしきい値を超えた場合の EEM での出力例を示します。

```
スイッチ(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
 lt entry-val 5120000 poll-interval 10
```

例：EEM イベントへの応答

次に、EEM イベントに応答して行われる処理の例を示します。

```
スイッチ(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current
 available memory is $_snmp_oid_val bytes"
スイッチ(config-applet)# action 2.0 force-switchover
```

例：EEM 環境変数の表示

次に、show event manager environment コマンドの出力例を示します。

```
スイッチ# show event manager environment all
No.   Name                      Value
1     _cron_entry               0-59/2 0-23/1 * * 0-6
2     _show_cmd                 show ver
3     _syslog_pattern           .*UPDOWN.*Ethernet1/0.*
4     _config_cmd1 interface    Ethernet1/0
5     _config_cmd2              no shut
```

次に、ソフトウェアによって割り当てられたCRONタイマー環境変数を毎日の毎時間、毎分、毎秒に設定する方法を示します。

```
スイッチ(config)# event manager environment_cron_entry 0-59/2 0-23/1 * * 0-6
```

次に、システム ポリシーとして登録された tm_cli_cmd.tcl という名前の EEM ポリシーの例を示します。システム ポリシーは Cisco IOS イメージの一部です。ユーザ定義の TCL スクリプトは、最初にフラッシュ メモリにコピーする必要があります。

```
スイッチ(config)# event manager policy tm_cli_cmd.tcl type system
```

第 43 章

Flexible NetFlow の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

NetFlow Lite の前提条件

NetFlow Lite モニターを接続するために、次の 2 つのターゲットがサポートされています。

- ポート：EtherChannel などの論理インターフェイスではなく、物理インターフェイスのみでサポートされるモニター接続。物理インターフェイスは、ルーテッド ポートまたはスイッチド ポートです。
- VLAN：モニター接続は、レイヤ 2 VLAN ではなく、VLAN インターフェイス（SVI）のみでサポートされます。

NetFlow Lite の制約事項

NetFlow Liteの制限事項は次のとおりです。

- モニタの制約事項：
  - モニター接続は、入力方向に限りサポートされます。
  - エクスポータはインターフェイスごとに複数サポートされますが、モニターはインターフェイスごとに 1 つサポートされます。
  - モニターでは永続キャッシュと標準キャッシュのみサポートされます。即時キャッシュはサポートされません。
  - モニター パラメータがインターフェイスまたは VLAN に適用される場合は、それらのモニター パラメータは変更できません。
  - ポートおよびVLANの両方がモニターに接続されている場合、ポートに着信するトラフィックについて VLAN モニターはポート モニターを上書きします。
  - フローモニタータイプとトラフィック タイプ（タイプとは、IPv4、IPv6、およびデータ リンクを意味します）は、作成するフローで同じである必要があります。
  - deviceでは、インターフェイスにIPおよびポートベースのモニターを同時に接続できません。48 ポートdeviceは最大 48 台のモニター（IP またはポートベース）をサポートし、256 SVI は最大 256 台のモニター（IP またはポートベース）を設定できます。
  - **show flow monitor** *flow_name***cache** コマンドを実行すると、スイッチはそれ以前のスイッチ ソフトウェア バージョン（Catalyst 2960-S）からのキャッシュ情報を、すべてのフィールドにゼロが入力された状態で表示します。これらのフィールドはスイッチに適用できないため、無視します。
- サンプラーの制限事項：
  - サンプリングされた NetFlow のみがサポートされます。
  - ポートと VLAN の両方について、deviceでは合計 4 つのサンプラーのみ（ランダムまたは確定）がサポートされます。
  - 両方のモードのサンプリング最小レートは、32 個のフローの中から 1 つで、両方のモードのサンプリング最大レートは 1022 個のフローから 1 つです。
  - サンプラーをインターフェイスに接続している間、サンプラーをモニタと関連付けておく必要があります。これを行わないと、コマンドは拒否されます。このタスクを実行するには、**ip flow monitor** *monitor_name* sampler s*ampler_name* **input** インターフェイス コンフィギュレーション コマンドを使用します。
  - 確定サンプラーを使用してモニタを接続する場合は、同じサンプラーを使用するすべての接続で、4 個の使用可能なサンプラーの中から 1 つの新しいフリー サンプラーを

スイッチ（ハードウェア）から使用します。サンプラーによるモニターの接続は4つまで許容されます。

ランダムサンプラーを使用してモニターを接続する場合は、最初の接続のみがスイッチ（ハードウェア）からの新しいサンプラーを使用します。同じサンプラーを使用する残りのすべての接続は、同じサンプラーを共有します。

この動作のため、確定サンプラーを使用する場合は、サンプリングレートとdeviceが送信した内容を比較することによって、サンプリングされたフローの正確な数を常に確認できます。同じランダム サンプラーを複数のインターフェイスで使用する場合は、任意のインターフェイスからのフローを常にサンプリングし、他のインターフェイスからのフローは常にスキップすることができます。

- ネットワーク フローおよび統計情報はライン レートで収集されます。
- ACL ベースの NetFlow はサポートされていません。
- NetFlow バージョン 9 のみが *export-protocol* コマンド オプションを使用した Flexible NetFlow エクスポータでサポートされます。NetFlow バージョン 5 を設定した場合、このバージョンは受け入れられますが、現在、NetFlow バージョン 5 のエクスポート機能は利用できず、サポートもされていません。
- スイッチは同種スタック構成をサポートしますが、混合スタック構成はサポートしません。

NetFlow Lite について

NetFlow Lite の概要

NetFlow Lite ではフローを使用して、アカウンティング、ネットワーク モニターリング、およびネットワーク プランニングに関連する統計情報を提供します。

フローは送信元インターフェイスに届く単方向のパケット ストリームで、キーの値は同じです。キーは、パケット内のフィールドを識別する値です。フローを作成するには、フロー レコードを使用して、フロー固有のキーを定義します。

deviceは、ネットワーク異常とセキュリティ問題の高度な検出をイネーブルにする NetFlow Lite 機能をサポートします。NetFlow Lite により、大量の定義済みフィールドの集合からキーを選択して、特定のアプリケーションに最適なフロー レコードを定義できます。

1 つのフローと見なされるパケットでは、すべてのキー値が一致している必要があります。フローは、設定したエクスポート レコード バージョンに基づいて、関係のある他のフィールドを集めることもあります。フローは NetFlow Lite キャッシュに格納されます。

エクスポータを使用してNetFlow Liteがフローのために収集するデータをエクスポートし、NetFlow Lite コレクタなどのリモート システムにこのデータをエクスポートできます。NetFlow Lite コレクタは、IPv4 アドレスを使用できます。

モニターを使用してフローのために収集するデータのサイズを定義します。モニターで、フロー レコードおよびエクスポータを NetFlow Lite キャッシュ情報と結合します。

Cisco IOS XE 16.12.1 リリース以降、Flexible NetFlow 上の送信元グループタグ（SGT）および宛先グループタグ（DGT）フィールドは、IPv6 トラフィックでサポートされます。

Flexible NetFlow のコンポーネント

Flexible NetFlow は、いくつかのバリエーションで一緒に使用して、トラフィック分析およびデータ エクスポートに使用できるコンポーネントで構成されます。Flexible NetFlow のユーザー定義のフロー レコードおよびコンポーネントの構造では、最小限の数のコンフィギュレーション コマンドで、ネットワーキング デバイスでのトラフィック分析およびデータ エクスポートのためのさまざまなコンフィギュレーションの作成が容易になります。各フローモニターに、フロー レコード、フロー エクスポータ、およびキャッシュ タイプの固有の組み合わせを設定できます。フロー エクスポータの宛先 IP アドレスなどのパラメータを変更する場合、フロー エクスポータを使用するすべてのフローモニターに対して自動的に変更されます。同じフローモニターを複数のフローサンプラと組み合わると、さまざまなインターフェイス上でさまざな速度の同じタイプのネットワーク トラフィックをサンプリングできます。ここでは、Flexible NetFlow コンポーネントのその他の情報を提供します。

フロー レコード

Flexible NetFlow では、キー フィールドと非キー フィールドの組み合わせをレコードと呼びます。Flexible NetFlow のレコードは Flexible NetFlow フロー モニターに割り当てられ、フローデータの格納に使用されるキャッシュが定義されます。

フロー レコードでは、フロー内のパケットを識別するために Flexible NetFlow で使用するキーとともに、Flexible NetFlow がフローについて収集する他の関連フィールドを定義します。キーと関連フィールドを任意の組み合わせで指定して、フロー レコードを定義できます。device は、幅広いキーセットをサポートします。フローレコードでは、フロー単位で収集するカウンタのタイプも定義します。64 ビットのパケットまたはバイトカウンタを設定できます。device は、フロー レコードの作成時に、デフォルトとして次の match フィールドをイネーブルにします。

- match datalink：レイヤ 2 属性
- match ipv4：IPv4 属性
- match ipv6：IPv6 属性
- match transport：トランスポート層フィールド
- match wireless：ワイヤレス フィールド

NetFlow の事前定義済みのレコード

Flexible NetFlow には事前定義済みのレコードがいくつか含まれ、それを使用してネットワーク トラフィックの監視を開始できます。事前定義済みのレコードは、Flexible NetFlow を迅速に導入するために役立ち、ユーザー定義のフロー レコードよりも簡単に使用できます。ネット

ワーク モニタリングのニーズを満たす定義済みのレコードのリストから選択できます。Flexible NetFlowが改良されると、一般的なユーザー定義のフロー レコードを事前定義済みレコードとして使用でき、簡単に導入できるようになります。

事前定義済みレコードにより、エクスポートされるデータのために既存のNetFlow コレクタ コンフィギュレーションとの下位互換性が確保されます。事前定義済みレコードは、それぞれ固有の key および nonkey フィールドの組み合わせを持ち、ルータで Flexible NetFlow をカスタマイズしなくても、ネットワーク内のさまざまなタイプのトラフィックを監視する、内蔵機能を提供します。

2 つの事前定義済みレコード（NetFlow original と NetFlow IPv4/IPv6 original output）は機能的に同等で、以前の（入力）NetFlow、および以前の NetFlow の 出力 NetFlow アカウンティング機能をそれぞれエミュレートします。その他の Flexible NetFlow の事前定義済みレコードのいくつかは、以前の NetFlow で利用できる集約キャッシュ方式に基づきます。以前の NetFlow で利用できる集約キャッシュ方式に基づく Flexible NetFlow の事前定義済みレコードでは、集約を実行しません。代わりに、事前定義済みレコードによって各フローが個別に追跡されます。

ユーザー定義レコード

Flexible NetFlow では、key および nonkey フィールドを指定し、実際の要件に合わせてデータ収集をカスタマイズすることで、Flexible NetFlow フロー モニター キャッシュ用の独自のレコードを定義できます。Flexible NetFlow フロー モニター キャッシュに対して独自のレコードを定義する場合、ユーザー定義レコードと呼ばれます。nonkeyフィールドの値は、フロー内のトラフィックに関する追加情報を提供するためにフローに追加されます。nonkeyフィールドの値の変更によって新しいフローが作成されることはありません。ほとんどの場合、nonkeyフィールドの値はフロー内の最初のパケットからのみ取得されます。FlexibleNetFlowを使用すると、nonkeyフィールドとして、フロー内のバイト数やパケット数などのカウンター値をキャプチャできます。

Flexible NetFlow では、ヘッダーおよびパケット セクションのタイプに新しいバージョン 9 エクスポート フォーマット フィールド タイプが追加されます。Flexible NetFlow は NetFlow コレクタに、対応するバージョン 9 エクスポート テンプレート フィールドで設定されたセクション サイズを通知します。ペイロード セクションには、対応する長さフィールドがあり、収集されるセクションの実際のサイズを収集するために使用できます。

NetFlow Lite match パラメータ

フロー レコードの次のキー フィールドを照合できます。

- IPv4 または IPv6 宛先アドレス
- Datalink フィールド（送信元および宛先 MAC アドレス、ならびに MAC EtherType（ネットワーク プロトコルのタイプ））。
- アプリケーションのタイプ（ICMP、IGMP、または TCP トラフィック）を識別するトランスポート フィールドの送信元および宛先ポート。

次の表で、NetFlow Lite の match パラメータについて説明します。フロー レコードごとに、次の match パラメータを 1 つ以上設定する必要があります。

表 *79 : match* パラメータ

コマンド	目的
match datalink {**ethertype** | **mac** {**destination address input** | **source address input**}}	データ リンクまたはレイヤ 2 フィールドとの一致を指定します。次のコマンド オプションが使用可能です。 • **ethertype**：パケットの ethertype と一致します。 • **mac**：入力時のパケットの送信元または宛先 MAC アドレスと一致します。 （注） データリンク フロー モニタがインターフェイスまたはVLANに割り当てられている場合、非IPv6または非IPv4トラフィック用のフローだけが作成されます。
match ipv4 {**destination** {**address**} | **protocol** | **source** {**address**} | **tos**}	IPv4 フィールドとの一致を指定します。次のコマンド オプションが使用可能です。 • **destination**：IPv4 宛先アドレス ベースのフィールドと一致します。 • **protocol**：IPv4プロトコルと一致します。 • **source**：IPv4 送信元アドレス ベースのフィールドと一致します。 • **tos**：IPv4 タイプ オブ サービス フィールドと一致します。
match ipv6 {**destination** {**address**} | **flow-label** | **protocol** | **source** {**address**} }	IPv6 フィールドとの一致を指定します。次のコマンド オプションが使用可能です。 • **destination**：IPv6 宛先アドレス ベースのフィールドと一致します。 • **flow-label**：IPv6 フローラベル フィールドと一致します。 • **protocol**：IPv6 ペイロード プロトコル フィールドと一致します。 • **source**：IPv6 送信元アドレス ベースのフィールドと一致します。

コマンド	目的
match transport {**destination-port** | **source-port**}	トランスポート層フィールドとの一致を指定します。次のコマンド オプションが使用可能です。 • **destination-port**：転送先ポートと一致します。 • **source-port**：転送元ポートと一致します。
	フロー レコードのキー フィールドとしてSSIDのワイヤレス ネットワークの使用を指定します。

NetFlow Lite collect パラメータ

フロー レコードの次のキー フィールドを収集できます。

- 合計バイト数、エクスポータによって送信されるまたはフローまたはパケット（exporter）、または 64 ビット カウンタのバイト数またはパケット数（long）。
- 最初のパケットの送信時間または最新（最後）のパケットが見つかった時間からのシステム稼働時間に基づくタイムスタンプ。
- 入力インターフェイスのSNMPインデックス。サービスモジュールに着信するトラフィックのインターフェイスは、スイッチの転送キャッシュに基づいています。このフィールドは、一般にデータ リンク、IPv4 および IPv6 アドレスとともに使用され、直接接続されたホストの実際のファースト ホップのインターフェイスを提供します。
  - 値 0 は、インターフェイス情報がキャッシュにないことを意味します。
  - 一部の NetFlow コレクタでは、フロー レコードにこの情報が必要です。

次の表で、NetFlow Liteの collect パラメータについて説明します。

表 *80 : collect* パラメータ

コマンド	目的
collect counter {**bytes** {**long** | **permanent** } | **packets** { **long** | **permanent**}}	カウンタ フィールドの合計バイト数と合計パケット数を収集します。
collect flow {**sampler**}	フロー サンプラー識別子（ID）を収集します。
collect interface {**input**}	入力インターフェイスからフィールドを収集します。

コマンド	目的
collect timestamp sys-uptime {**first** | **last**}	最初のパケットが確認された時刻、または最新のパケットが最後に確認された時刻のフィールドを収集します（ミリ秒）。
collect transport tcp flags	次の転送 TCP フラグを収集します。 • **ack**：TCP 確認応答フラグ • **cwr**：TCP 輻輳ウィンドウ縮小フラグ • **ece**：TCP ECN エコー フラグ • **fin**：TCP 終了フラグ • **psh**：TCP プッシュ フラグ • **rst**：TCP リセット フラグ • **syn**：TCP 同期フラグ • **urg**：TCP 緊急フラグ
	ワイヤレス クライアントが関連付けられているアクセス ポイントの MAC アドレスを収集します。

フロー エクスポータ

フローエクスポータでは、フロー モニタ キャッシュ内のデータをリモートシステム（たとえば、分析および保管のためにNetFlowコレクタを実行するサーバ）にエクスポートします。フローエクスポータは、コンフィギュレーションで別のエンティティとして作成されます。フローエクスポータは、フローモニタにデータエクスポート機能を提供するためにフローモニタに割り当てられます。複数のフローエクスポータを作成して、1 つまたは複数のフローモニタに適用すると、いくつかのエクスポート先を指定することができます。1つのフローエクスポータを作成し、いくつかのフローモニタに適用することができます。

NetFlow データ エクスポート フォーマットのバージョン 9

NetFlow の基本出力はフロー レコードです。NetFlow が改良され、フロー レコードのいくつかのフォーマットが向上しました。NetFlow エクスポート フォーマットの最新の進化は、バージョン 9 と呼ばれます。NetFlow Version 9 エクスポート フォーマットの識別機能は、テンプレートがベースとなります。テンプレートは、レコードフォーマットの設計を拡張可能なものにします。NetFlowサービスが将来拡張されても、基本フロー レコードフォーマットを変更し続ける必要がありません。テンプレートを使用すると、次のいくつかの利点があります。

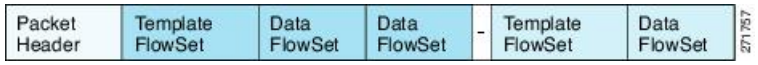
- NetFlow のコレクタを提供したり、サービスを表示したりするアプリケーションを作成するサードパーティ ビジネス パートナーは、新規の NetFlow 機能が追加されるたびにアプ

リケーションを再コンパイルする必要はありません。代わりに、既知のテンプレート フォーマットを記述する外部のデータ ファイルを使用することができます。

- 新規機能は、現在の導入環境を損ねることなく、NetFlow に迅速に追加できます。
- バージョン 9 フォーマットは新しいプロトコルや開発中のプロトコルに適応できるため、NetFlow はこれらのプロトコルに対して「将来的に対応」します。

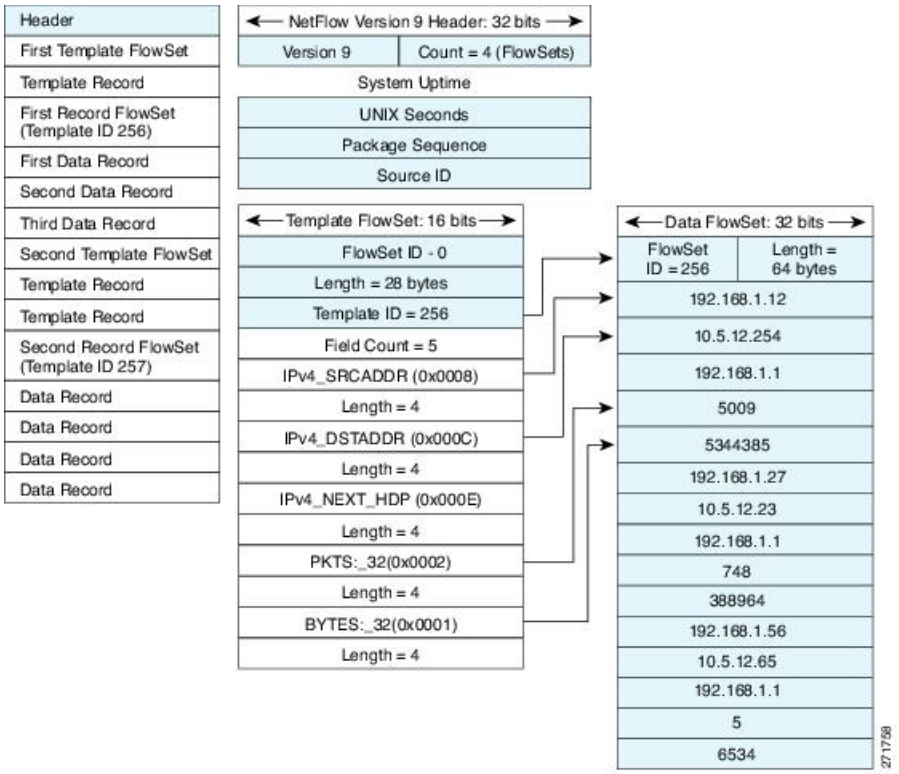
バージョン 9 のエクスポート フォーマットは、パケット ヘッダーとそれに続く 1 つ以上のテンプレート フロー セットまたはデータ フロー セットで構成されています。テンプレート フローセットでは、将来のデータ フローセットに表示されるフィールドの説明が提供されます。このようなデータ フロー セットは、後で同じエクスポート パケットまたは後続のエクスポート パケットで発生する可能性があります。テンプレート フロー セットおよびデータ フロー セットは、次の図に示すように、単一のエクスポート パケットに混在させることができます。

図 *71* : バージョン *9* エクスポート パケット



NetFlow Version 9 では、送信されるデータを NetFlow コレクタが理解できるように、テンプレート データを定期的にエクスポートします。また、テンプレートのデータ フロー セットもエクスポートします。Flexible NetFlow の主な利点は、ユーザーがフロー レコードを設定すると、バージョン 9 テンプレートに効率的に変換され、コレクタに転送されることです。下の図に、ヘッダー、テンプレート フロー セットおよびデータ フロー セットを含めて、NetFlow Version 9 エクスポート フォーマットの詳細な例を示します。

図 *72 : NetFlow* バージョン *9* エクスポート フォーマットの詳細例



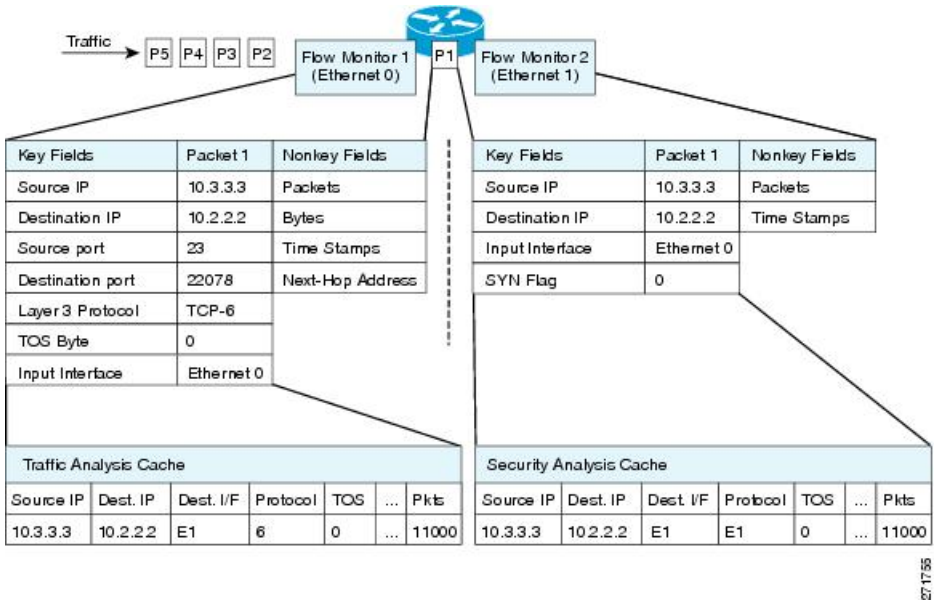
フロー モニター

フロー モニターは Flexible NetFlow のネットワーク トラフィックの監視を実行するコンポーネントで、インターフェイスに適用されます。

フロー データはネットワーク トラフィックから収集され、フロー レコードの key フィールドおよび nonkey フィールドに基づいて監視プロセス中にフロー モニター キャッシュに追加されます。

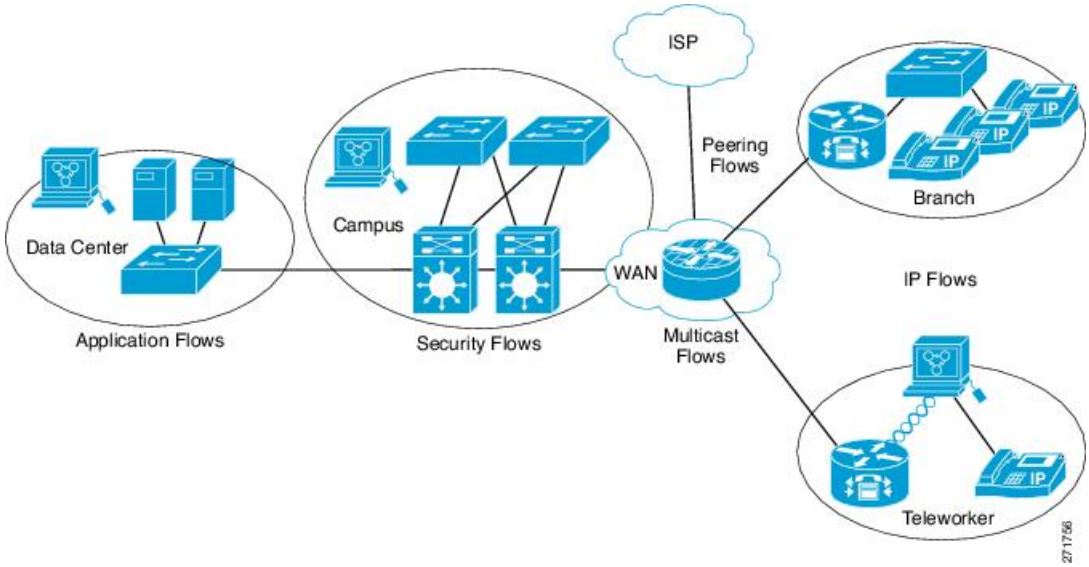
Flexible NetFlow は、同じトラフィックのさまざまなタイプの分析を実行するために使用できます。下の図では、入力インターフェイス上の標準トラフィック分析のために設計されたレコードと、出力インターフェイス上のセキュリティ分析のために設計されたレコードを使用してパケット 1 が分析されます。

図 *73 : 2* つのフロー モニターを使用した同じトラフィックの分析例



下の図に、カスタム レコードを使用して複数のタイプのフロー モニターを適用するより複雑な方法の例を示します。

図 *74 :* カスタム レコードでの複数のタイプのフロー モニターの複雑な使用例



標準

デフォルトのキャッシュ タイプは「normal」です。このモードでは、キャッシュ内のエントリが timeout active 設定と timeout inactive 設定に従って期限切れになります。キャッシュ エントリは、期限切れになるとキャッシュから削除され、設定されている何らかのエクスポータによってエクスポートされます。

フロー サンプラー

フローサンプラーは、ルータのコンフィギュレーションで別のコンポーネントとして作成されます。フローサンプラーは、分析用に選択されるパケット数を制限することで、NetFlow Lite を実行しているデバイス上の負荷を削減するために使用されます。

フロー サンプリングでは、ルータのパフォーマンスに対するモニタリング精度が交換されます。サンプラーをフローモニターに適用すると、フローモニターが分析する必要のあるパケット数が減少するため、ルータでフローモニターを実行するためのオーバーヘッド負荷が低下します。フロー モニターで分析されるパケット数が減少すると、フロー モニターのキャッシュに格納される情報の精度が、それに応じて低下します。

ip flow monitor コマンドを使用してインターフェイスに適用される場合、サンプラーはフローモニターと組み合わせて使用されます。

デフォルト設定

次の表に、device の NetFlow Lite デフォルト設定を示します。

表 *81 : NetFlow Lite* のデフォルト設定

設定	デフォルト
フロー アクティブ タイムアウト	1800 秒 （注） この設定のデフォルト値は特定の NetFlow Lite 設定では高すぎる場合があります。低い値（180 秒または 300 秒）への変更を検討してください。
フロー タイムアウトの非アクティブ化	イネーブル、30 秒
フロー アップデート タイムアウト	1800 秒
デフォルト キャッシュ サイズ	16640 エントリ

Flexible NetFlow の設定方法

Flexible Netflow を設定するには、次の一般的な手順に従います。

1. フローにキー フィールドおよび非キー フィールドを指定して、フロー レコードを作成します。
2. プロトコルを指定して任意のフローエクスポータを作成し、宛先ポート、宛先、およびその他のパラメータを転送します。
3. フロー レコードおよびフロー エクスポータに基づいて、フロー モニターを作成します。

4. 任意のサンプラーを作成します。
5. レイヤ 2 ポート、レイヤ 3 ポート、または VLAN にフロー モニターを適用します。

フロー レコードの作成

フロー レコードを作成し、照合するキー、および収集するフィールドをフロー内に追加できます。

手順の概要

1. **configure terminal**
2. **flow record** *name*
3. **description** *string*
4. **match** *type*
5. **collect** *type*
6. **end**
7. **show flow record** [**name** *record-name*]
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**flow record** *name* 例： `スイッチ(config)# flow record test` `スイッチ(config-flow-record)#`	フロー レコードを作成し、フロー レコード コンフィギュレーション モードを開始します。
ステップ 3	**description** *string* 例： `スイッチ(config-flow-record)# description Ipv4Flow`	（任意）最大 63 文字で、このフローの説明を指定します。
ステップ 4	**match** *type* 例： `スイッチ(config-flow-record)# match ipv4 source`	一致キーを指定します。

	コマンドまたはアクション	目的
	address スイッチ(config-flow-record)# **match ipv4 destination address** スイッチ(config-flow-record)# **match flow direction**	
ステップ 5	**collect** *type* 例： スイッチ(config-flow-record)# **collect counter bytes layer2 long** スイッチ(config-flow-record)# **collect counter bytes long** スイッチ(config-flow-record)# **collect timestamp absolute first** スイッチ(config-flow-record)# **collect transport tcp flags** スイッチ(config-flow-record)# **collect interface output**	コレクション フィールドを指定します。 （注） フローレコードの collect フィールドとしての **collect interface output** がフローモニターにある場合は、スイッチの宛先アドレスに基づいて出力インターフェイスが検出されます。そのため、他のフロー モニタの場合は、次の設定が必要です。 • ipv4 フローモニターの場合は、「**match ip destination address**」を設定します • ipv6 フローモニターの場合は、「**match ipv6 destination address**」を設定します • データリンクフローモニターの場合は、「**match datalink mac output**」を設定します 次のアドレスのいずれかにフローが作成された場合、**collect interface output** フィールドに **NULL** の値が返されます。 • L3 ブロードキャスト • L2 ブロードキャスト • L3 マルチキャスト • L2 マルチキャスト • L2 の不明な宛先。
ステップ 6	**end** 例： スイッチ(config-flow-record)# **end**	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 7	**show flow record** [**name** *record-name*] 例： スイッチ **show flow record test**	（任意）NetFlow のフロー レコード情報を表示します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

エクスポート フォーマット、プロトコル、宛先、およびその他のパラメータを指定することによって、任意でフロー エクスポータを定義します。

フロー エクスポータの作成

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

（注） フロー エクスポータごとに、1 つの宛先のみがサポートされます。複数の宛先にデータをエクスポートする場合は、複数のフロー エクスポータを設定してフロー モニターに割り当てる必要があります。

IPv4 アドレスを使用して宛先にエクスポートできます。

手順の概要

1. **configure terminal**
2. **flow exporter** *name*
3. **description** *string*
4. **destination** {*ipv4-address*}
5. **dscp** *value*
6. **transport udp** *number*
7. **ttl** *seconds*
8. **export-protocol** {**netflow-v9**}
9. **end**
10. **show flow exporter** [**name** *record-name*]
11. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**flow exporter** *name* 例： スイッチ(config)# **flow exporter ExportTest**	フロー エクスポータを作成し、フロー エクスポータ コンフィギュレーション モードを開始します。
ステップ 3	**description** *string* 例： スイッチ(config-flow-exporter)# **description ExportV9**	（任意）最大 63 文字で、このフローの説明を指定します。
ステップ 4	**destination** {*ipv4-address*} 例： スイッチ(config-flow-exporter)# **destination 192.0.2.1** (IPv4 destination)	このエクスポータに IPv4 宛先アドレスまたはホスト名を設定します。
ステップ 5	**dscp** *value* 例： スイッチ(config-flow-exporter)# **dscp 0**	（任意）DSCP（DiffServ コードポイント）値を指定します。範囲は 0 ～ 63 です。デフォルトは 0 です。
ステップ 6	**transport udp** *number* 例： スイッチ(config-flow-exporter)# **transport udp 200**	（任意）NetFlow コレクタに到達するために使用する UDP ポートを指定します。
ステップ 7	**ttl** *seconds* 例： スイッチ(config-flow-exporter)# **ttl 210**	（任意）エクスポータによって送信されるデータグラムの存続可能時間（TTL）値を設定します。範囲は 1 ～ 255 秒です。デフォルトは 255 です。

	コマンドまたはアクション	目的
ステップ 8	**export-protocol** {**netflow-v9**} 例： `スイッチ(config-flow-exporter)# export-protocol netflow-v9`	エクスポータで使用される NetFlow エクスポート プロトコルのバージョンを指定します。
ステップ 9	**end** 例： `スイッチ(config-flow-record)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show flow exporter** [**name** *record-name*] 例： `スイッチ# show flow exporter ExportTest`	（任意）NetFlow のフロー エクスポータ情報を表示します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

フロー レコードおよびフロー エクスポータに基づいて、フロー モニターを定義します。

フロー モニターの作成

フロー モニタを作成して、フロー レコードおよびフロー エクスポータと関連付けることができます。

手順の概要

1. **configure terminal**
2. **flow monitor** *name*
3. **description** *string*
4. **exporter** *name*
5. **record** *name*
6. **cache** { **timeout** {**active** | **inactive**} *seconds* | **type normal** }
7. **end**
8. **show flow monitor** [**name** *record-name*]
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**flow monitor** *name* 例： スイッチ(config)# **flow monitor MonitorTest** スイッチ (config-flow-monitor)#	フローモニタを作成し、フローモニタ コンフィギュレーション モードを開始します。
ステップ 3	**description** *string* 例： スイッチ(config-flow-monitor)# **description Ipv4Monitor**	（任意）最大 63 文字で、このフローの説明を指定します。
ステップ 4	**exporter** *name* 例： スイッチ(config-flow-monitor)# **exporter ExportTest**	フロー エクスポータとこのフロー モニタを関連付けます。
ステップ 5	**record** *name* 例： スイッチ(config-flow-monitor)# **record test**	フロー レコードを指定したフロー モニタと関連付けます。
ステップ 6	**cache** { **timeout** {**active** | **inactive**} *seconds* | **type normal** } 例： スイッチ(config-flow-monitor)# **cache timeout active 15000**	指定したフロー モニタとフロー キャッシュを関連付けます。
ステップ 7	**end** 例： スイッチ(config-flow-monitor)# **end**	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	**show flow monitor** [**name** *record-name*] 例： スイッチ **show flow monitor name MonitorTest**	（任意）NetFlow のフロー モニタ情報を表示します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

レイヤ 2 インターフェイス、レイヤ 3 インターフェイス、または VLAN にフロー モニタを適用します。

サンプラーの作成

サンプラーを作成し、フローの NetFlow サンプリング レートを定義できます。

手順の概要

1. **configure terminal**
2. **sampler** *name*
3. **description** *string*
4. **mode** {**deterministic** {*m* - *n*} | **random** {*m* - *n*}}
5. **end**
6. **show sampler** [*name*]
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**sampler** *name* 例：	サンプラーを作成し、サンプラー コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ(config)# **sampler SampleTest**	
ステップ 3	**description** *string* 例： スイッチ(config-flow-sampler)# **description samples**	（任意）最大 63 文字で、このフローの説明を指定します。
ステップ 4	**mode** {**deterministic** {*m* - *n*} | **random** {*m* - *n*}} 例： スイッチ(config-flow-sampler)# **mode random 1 out-of 1022**	ランダム サンプル モードを定義します。 インターフェイスに対してランダムサンプラーまたは確定的サンプラーのいずれも設定できます。*n* パケット ウィンドウから *m* 個のパケットを選択します。ウィンドウ サイズには、32～1022 の範囲のパケットを選択します。 インターフェイスにサンプラーを設定する際は、次の点に注意してください。 • 確定的サンプラー（s1 など）を使用してモニターを接続する場合、同じサンプラーs1との接続ごとにdevice（ハードウェア）から 4 つの使用可能なサンプラーのうちの新しい空きサンプラーの 1 つを使用します。したがって、サンプラーとモニターの接続は、4 つを超えて行うことができません。 • これとは逆に、ランダムサンプラー（たとえば、この場合もs1など）を使用してモニターを接続する場合、最初の接続だけが device（ハードウェア）の新しいサンプラーを使用します。同じサンプラーs1を使用するすべての接続のうちの残りは同じサンプラーを共有します。 • この動作により、確定的サンプラーを使用する際は、サンプリング レートとdeviceが何を送信するかを比較して、適切な数のフローがサンプリングされているかを常に確認することができます。複数のインターフェイスに同じランダムサンプラーを使用している場合は、インターフェイスからのフローを常にサンプリングすることができ、他のインターフェイスからのフローは常にスキップできます。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config-flow-sampler)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show sampler** [*name*] 例： スイッチ **show sample SampleTest**	（任意）NetFlow サンプラに関する情報を表示します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

ソースインターフェイス、または SVI にフローモニターを適用します。

インターフェイスへのフローの適用

フロー モニターおよびオプションのサンプラーをインターフェイスに適用できます。

手順の概要

1. **configure terminal**
2. **interface** *type*
3. {**ip flow monitor** | **ipv6 flow monitor**}*name* [| **sampler** *name*] {**input**}
4. **end**
5. **show flow interface** [*interface-type number*]
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**interface** *type* 例： `スイッチ(config)# interface GigabitEthernet1/0/1`	インターフェイス コンフィギュレーション モードを開始し、インターフェイスを設定します。
ステップ 3	{**ip flow monitor** | **ipv6 flow monitor**}*name* [| **sampler** *name*] {**input**} 例： `スイッチ(config-if)# ip flow monitor MonitorTest input`	入力または出力パケットに対応するインターフェイスに、IPv4 または IPv6 フロー モニター、およびオプションのサンプラーを関連付けます。 入力と出力の両方向でインターフェイスに複数のモニターを関連付けることができます。
ステップ 4	**end** 例： `スイッチ(config-flow-monitor)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show flow interface** [*interface-type number*] 例： `スイッチ# show flow interface`	（任意）インターフェイスの NetFlow 情報を表示します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VLAN 上でのブリッジ型 NetFlow の設定

フロー モニターおよびオプションのサンプラーを VLAN に適用できます。

手順の概要

1. **configure terminal**
2. **vlan** [**configuration**] *vlan-id*
3. **ip flow monitor** *monitor name* [**sampler** *sampler name*] { **input** }
4. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vlan** [**configuration**] *vlan-id* 例： スイッチ(config)# **vlan configuration 30** スイッチ(config-vlan-config)#	VLAN または VLAN コンフィギュレーション モードを開始します。
ステップ 3	**ip flow monitor** *monitor name* [**sampler** *sampler name*] { **input** } 例： スイッチ(config-vlan-config)# **ip flow monitor MonitorTest input**	入力パケットに対応する VLAN に、フローモニターおよびオプションのサンプラーを関連付けます。
ステップ 4	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

レイヤ 2 NetFlow の設定

NetFlow Lite レコード内でレイヤ 2 キーを定義できます。このレコードを使用して、レイヤ 2 インターフェイスのフローをキャプチャできます。

手順の概要

1. **configure terminal**
2. **flow record** *name*
3. **match datalink**{**ethertype** | **mac**{**destination** {**address input**} | **source**{**address input**}}}
4. **end**
5. **show flow record** [*name*]
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**flow record** *name* 例： `スイッチ(config)# flow record L2_record` `スイッチ(config-flow-record)#`	フロー レコード コンフィギュレーション モードを開始します。
ステップ 3	**match datalink**{**ethertype** | **mac**{**destination** {**address input**} | **source**{**address input**}}} 例： `スイッチ(config-flow-record)# match datalink mac source address input` `スイッチ(config-flow-record)# match datalink mac destination address input`	レイヤ 2 属性をキーとして指定します。この例では、入力時のパケットの送信元および宛先の MAC アドレスがキーです。 （注） データリンク フロー モニターがインターフェイスまたは VLAN レコードに割り当てられている場合、非 IPv4 または非 IPv6 トラフィック用のフローだけが作成されます。
ステップ 4	**end** 例： `スイッチ(config-flow-record)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show flow record** [*name*] 例： `スイッチ# show flow record`	（任意）インターフェイスの NetFlow 情報を表示します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

Flexible NetFlow の監視

次の表にあるコマンドを使用して、Flexible NetFlow をモニタリングできます。

表 *82 : Flexible NetFlow* のモニタリング コマンド

コマンド	目的
show flow exporter [**broker** | **export-ids** | **name** | *name* | **statistics** | **templates**]	NetFlow のフロー エクスポータ情報と統計情報を表示します。
show flow exporter [**name** *exporter-name*]	NetFlow のフロー エクスポータ情報と統計情報を表示します。
show flow interface	NetFlowインターフェイスに関する情報を表示します。
	NetFlow のフロー モニター情報と統計情報を表示します。
show flow monitor statistics	フロー モニターの統計情報を表示します。
	指定された形式でフローモニターのキャッシュの内容を表示します。
show flow record [**name** *record-name*]	NetFlowのフロー レコード情報を表示します。
show sampler [**broker** | **name** | *name*]	NetFlow サンプラーに関する情報を表示します。

設定例 NetFlow Lite

例：フローの設定

（注） フローを設定する場合、フロー レコードで定義されたプロトコル、送信元ポート、宛先ポート、最初と最後のタイムスタンプ、パケットおよびバイト カウンタが必要です。これらがないと、「Warning: Cannot set protocol distribution with this Flow Record. Require protocol, source and destination ports, first and last timestamps and packet and bytes counters.」というエラー メッセージが表示されます。

フローを作成し、そのフローをインターフェイスに適用する例を示します。

```
スイッチ# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

スイッチ(config)# flow exporter export1
スイッチ(config-flow-exporter)# destination 10.0.101.254
スイッチ(config-flow-exporter)# transport udp 2055
スイッチ(config-flow-exporter)# template data timeout 60
スイッチ(config-flow-exporter)# exit
スイッチ(config)# flow record record1
スイッチ(config-flow-record)# match ipv4 source address
スイッチ(config-flow-record)# match ipv4 destination address
スイッチ(config-flow-record)# match ipv4 protocol
スイッチ(config-flow-record)# match transport source-port
スイッチ(config-flow-record)# match transport destination-port
スイッチ(config-flow-record)# collect counter bytes long
スイッチ(config-flow-record)# collect counter packets long
スイッチ(config-flow-record)# collect timestamp sys-uptime first
スイッチ(config-flow-record)# collect timestamp sys-uptime last
スイッチ(config-flow-record)# exit
スイッチ(config)# sampler SampleTest
スイッチ(config-sampler)# mode random 1 out-of 100
スイッチ(config-sampler)# exit
スイッチ(config)# flow monitor monitor1
スイッチ(config-flow-monitor)# cache timeout active 300
スイッチ(config-flow-monitor)# cache timeout inactive 120
スイッチ(config-flow-monitor)# record record1
スイッチ(config-flow-monitor)# exporter export1
スイッチ(config-flow-monitor)# exit
スイッチ(config)# interface GigabitEthernet1/0/1
スイッチ(config-if)# ip flow monitor monitor1 sampler SampleTest input
スイッチ(config-if)# end
```

Flexible NetFlow の機能情報

リリース	変更内容
Cisco IOS Release 15.2(3)E	この機能が導入されました。
Cisco IOS XE Gibraltar 16.12.1	IPv6 トラフィックについて、FNF の SGT フィールドと DGT フィールドのサポートが導入されました。

第 44 章

Web Cache Communication Protocol を使用したキャッシュ サービスの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

WCCP の前提条件

スイッチで WCCP を設定する前に、次の設定要件に従ってください。

- 同じサービス グループ内のアプリケーション エンジンおよびスイッチは、WCCP 対応のスイッチに直接接続された同一サブネットワーク内に存在する必要があります。
- クライアント、アプリケーション エンジン、およびレイヤ 3 インターフェイスとしてのサーバ（ルーテッド ポートおよびスイッチ仮想インターフェイス（SVI））に接続されたスイッチ インターフェイスを設定します。WCCP パケットのリダイレクトが機能するた

めには、サーバ、アプリケーションエンジン、およびクライアントが、異なるサブネット上に存在する必要があります。

- 各アプリケーション エンジンに 1 つのマルチキャスト アドレスを設定するときは、予約されていないマルチキャスト アドレスだけを使用します。
- WCCP エントリおよび PBR エントリは、同じ TCAM リージョンを使用します。WCCP は、PBR（アクセス、ルーティング、デュアル IPv4/v6 ルーティング）をサポートするテンプレート上でだけサポートされます。
- TCAM エントリを WCCP エントリの追加に使用できない場合、パケットはリダイレクトされず、標準ルーティング テーブルを使用して転送されます。
- 使用可能な PBR ラベルの数は、WCCP 入力方法でイネーブルになるインターフェイスが増えるにつれて減っていきます。サービス グループをサポートする各インターフェイスでは、ラベルが 1 つ消費されます。WCCP ラベルは PBR ラベルから取得されます。PBR と WCCP 間で使用可能なラベルを監視および管理する必要があります。ラベルが使用できないと、スイッチはサービス グループを追加できなくなります。ただし、別のインターフェイスに同じ一連のサービス グループがある場合、新しいラベルは必要にならず、グループをインターフェイスに追加できます。
- スタック メンバー スイッチで設定されたルーティング最大伝送単位（MTU）サイズは、クライアント MTU サイズより長い必要があります。アプリケーション エンジンに接続されたポートで設定された MAC レイヤ MTU サイズは、GRE トンネル ヘッダー バイトを考慮する必要があります。

WCCP に関する制約事項

サポートされない WCCP 機能

次の WCCP 機能は、このソフトウェア リリースでサポートされていません。

- **ip wccp redirect out** インターフェイス コンフィギュレーション コマンドを使用して設定された発信インターフェイスでのパケットのリダイレクト
- パケット リダイレクトの GRE 転送方式
- GRE リダイレクトおよび GRE リターン
- Cisco Catalyst 3650-CX スイッチでは、パケット損失を回避するために、カスタマーエッジ（CE）に接続された 1 ギガバイトのポートでフロー制御インターフェイス コンフィギュレーション コマンドを使用する必要があります。
- GRE 上での WCCP
- ロードバランシング用のハッシュ割り当て方式
- WCCP の SNMP サポート

- ハードウェアでのハッシュ割り当てマスク割り当てのみを使用したロードバランスの実行
- フラグメント化されたパケットのリダイレクト。これは、セキュリティ機能です。

一般的な制約事項

- サービス グループの最大数：8 入力および 8 出力。
- 同じスイッチ インターフェイス上では、WCCP と VPN ルーティングおよび転送（VRF）を設定できません。
- 同じスイッチ インターフェイス上では、WCCP および PBR を設定できません。
- 同じスイッチ インターフェイス上では、WCCP およびプライベート VLAN（PVLAN）を設定できません。
- **ip wccp redirect exclude in** コマンドは、出力 WCCP 方式から入力パケットを除外できるようにします。これは、CE へのインターフェイスでは必要ではありません。
- キャッシュ エンジンが使用できない場合は、一致するパケットはドロップされます。これは、クローズ グループのサポートです。VRF 認識 WCCP のサポート、IPv6 WCCP のサポートはありません。
- デバイスを **ip wccp check services all** コマンドで設定すると、リダイレクト ACL がパケットと一致しなかった場合、次のプライオリティのサービスグループと照合されます。

WCCP に関する情報

WCCP の概要

（注） この機能を使用するには、デバイス上で IP Services フィーチャー セットが稼働している必要があります。

WCCP をサポートするのは、Cisco Catalyst 3560-CX スイッチのみです。

WCCP はシスコが開発したコンテンツ ルーティング技術です。WCCP を使用すると広域アプリケーション エンジン（以降、アプリケーション エンジンと呼ぶ）をネットワーク インフラストラクチャに統合できます。アプリケーションエンジンは、頻繁にアクセスのあるコンテンツを透過的に格納し、その同じコンテンツへの要求を満たし、サーバから繰り返し伝送されることを防ぎます。アプリケーションエンジンは、コンテンツ配信を加速させ、最大限のスケーラビリティとコンテンツの可用性を実現します。サービスプロバイダーネットワークのアクセス ポイント（POP）で、WCCP およびアプリケーション エンジン ソリューションを展開できます。エンタープライズ ネットワークでは、地域サイトまたは小規模ブランチ オフィスで WCCP およびアプリケーション エンジン ソリューションを展開できます。

WCCPおよびシスコのキャッシュ エンジン（またはWCCPが稼働している他のアプリケーション エンジン）は、ネットワークでのトラフィック パターンをローカライズし、コンテンツ要求がローカルで実現されるようにします。

WCCP により、サポート対象のシスコ ルータおよびdevicesは、コンテンツ要求を透過的にリダイレクトできます。透過リダイレクトを使用すると、ユーザは使用しているブラウザがWebプロキシを使用するように設定する必要がありません。代わりに、ターゲット URL を使用してコンテンツを要求でき、その要求は自動的にアプリケーション エンジンにリダイレクトされます。透過という用語は、エンドユーザが、自分の要求したファイル（Webページなど）が、もとの指定したサーバからではなくアプリケーション エンジンから送信されるのを知らないという意味です。

アプリケーション エンジンが要求を受け取ると、自身のローカル キャッシュからサービスしようとします。要求された情報が存在しない場合、アプリケーション エンジンは別個の要求をエンド サーバに送信し、要求された情報を取得します。取得した情報は、アプリケーション エンジンが要求元のクライアントに転送するとともに、その後の要求に応えるため、情報をキャッシュします。

WCCPでは、アプリケーション エンジン クラスタ（一連のアプリケーション エンジン）は、、複数のルータまたはdevicesにサービスできます。

WCCP メッセージ交換

次の一連のイベントは、WCCP メッセージ交換について説明します。

1. アプリケーション エンジンは、WCCP を使用して IP アドレスを WCCP 対応deviceに送信し、Here I am メッセージを通して自己の存在を伝えます。deviceおよびアプリケーション エンジンは、UDP ポート 2048 に基づき、制御チャネルを介して互いに通信します。
2. WCCP対応deviceは、アプリケーション エンジンのIP情報を使用してクラスタ ビュー（クラスタ内のアプリケーション エンジンのリスト）を作成します。このビューが、I see you メッセージでクラスタ内の各アプリケーション エンジンに送信すると、本質的にすべてのアプリケーション エンジンが互いの存在を認識するようになります。クラスタのメンバーシップが一定時間同じままになった後で、安定したビューが確立されます。
3. 安定したビューが確立されると、クラスタ内の低い IP アドレスを持つアプリケーション エンジンが指定アプリケーション エンジンとして選択されます。

WCCP ネゴシエーション

WCCP プロトコル メッセージを交換する際、指定アプリケーション エンジンおよび WCCP 対応deviceは次の項目をネゴシエートします。

- 転送方式（deviceがパケットをアプリケーション エンジンに転送するときに使用される方式）。deviceは、パケット宛先 MAC アドレスをターゲット アプリケーション エンジン MACアドレスに置き換えて、レイヤ2ヘッダーを書き換えます。次にスイッチは、パケットをアプリケーション エンジンに転送します。この転送方式では、ターゲット アプリケーション エンジンがレイヤ 2 でdeviceに直接接続されている必要があります。

- 割り当て方式（パケットをクラスタ内のアプリケーションエンジン間に配信するときに使用される方式）。deviceは宛先 IP アドレス、送信元 IP アドレス、宛先レイヤ 4 ポート、および送信元レイヤ4ポートの一部のビットを使用して、リダイレクトされたパケットを受け取るアプリケーション エンジンを判別します。
- パケット戻し方式（パケットをアプリケーション エンジンから通常の転送用deviceに戻すときに使用される方式）。アプリケーションエンジンがパケットを拒否し、パケット戻し機能を起動するのには以下の理由があります。
  - アプリケーション エンジンが過負荷となり、パケットにサービスする余裕がない。
  - アプリケーション エンジンがサーバからエラー メッセージ（プロトコル エラーや認証エラーなど）を受け取り、ダイナミック クライアント バイパス機能を使用している。バイパスは、クライアントがアプリケーションエンジンをバイパスし、サーバに直接接続できるようにします。

アプリケーション エンジンはパケットを WCCP 対応deviceに戻し、アプリケーション エンジンが存在しないかのようにサーバに転送します。アプリケーションエンジンは、再接続試行を代行受信しません。このようにして、アプリケーションエンジンは効率的にアプリケーションエンジンへのパケットのリダイレクトをキャンセルし、バイパス フローを作成します。戻し方式がレイヤ2書き換えである場合、パケットはハードウェア内でターゲット サーバに転送されます。サーバが情報に応答しているとき、deviceは通常のレイヤ 3 転送を使用して、情報を要求しているクライアントに戻します。

MD5 セキュリティ

WCCP は各プロトコル メッセージでオプションのセキュリティ コンポーネントを提供し、deviceとアプリケーションエンジン間のメッセージでMD5 認証をdeviceが使用できるようにします。（deviceの認証がイネーブルになっているとき）MD5 で認証されないメッセージは、deviceによって廃棄されます。パスワード文字列は、MD5 値と組み合され、deviceとアプリケーション エンジン間の接続のセキュリティを確立します。各アプリケーション エンジンで同じパスワードを設定する必要があります。

パケットのリダイレクトおよびサービス グループ

WCCP を設定して、FTP、プロキシ Web キャッシュ処理、音声およびビデオアプリケーションなど、リダイレクト用トラフィックを分類できます。この分類はサービス グループと呼ばれ、プロトコル タイプ（TCP または UDP）およびレイヤ 4 送信元ポート番号と宛先ポート番号に基づきます。サービス グループは、TCP ポート 80 を意味する、Web キャッシュなどの Well-known 名または 0 ～ 99 のサービス番号のいずれかで識別されます。サービス グループは、プロトコルおよびレイヤ4ポート番号にマッピングするように設定され、独立して確立および維持されます。WCCP は、アプリケーション エンジンに加入して分類基準を動的に提供するダイナミック サービス グループを許可します。

deviceまたはdeviceスタックでは最大8つまでのサービスグループを、およびサービスグループにつき 32 までのキャッシュエンジンを設定できます。WCCP のグループ定義には、サービス

グループのプライオリティがあります。WCCPは、プライオリティを使用して、deviceハードウェアのサービス グループを設定します。たとえば、サービス グループ1はプライオリティ100で、宛先ポート80を探していて、サービス グループ2はプライオリティ50で、送信元ポート80を探している場合、送信元および宛先ポート80の着信パケットは、サービス グループ1を使用して転送されます。これは、サービス グループ1の方がプライオリティが高いためです。

WCCPは各サービス グループのアプリケーション エンジンのクラスタをサポートします。リダイレクトされたトラフィックは、アプリケーション エンジンの1つに送信可能です。deviceは、サービス グループのクラスタ内のアプリケーション エンジン間で、トラフィックのロードバランシングのマスク割り当て方式をサポートします。

WCCPがdevice上で設定された後、deviceはクライアントから受信したすべてのサービス グループ パケットをアプリケーション エンジンに転送します。ただし、次のパケットはリダイレクトされません。

- アプリケーションエンジンから発信され、サーバに宛てられたパケット
- アプリケーション エンジンから発信され、クライアントに宛てられたパケット
- アプリケーションエンジンにより返送または拒否されたパケットこれらのパケットはサーバに送信されます。

プロトコル メッセージの送受信用に、サービス グループにつき1つのマルチキャスト アドレスを設定できます。マルチキャスト アドレスが1つの場合、アプリケーション エンジンは通知を1つのアドレスに送信することになり、たとえば225.0.0.0など、サービス グループのすべてのルータにカバレッジを提供します。ルータを動的に追加および削除する場合、1つのマルチキャスト アドレスを使用することで、コンフィギュレーションが簡単になります。これは、特にWCCPネットワークのすべてのデバイスのアドレスを入力する必要がないためです。

ルータ グループ リストを使用すれば、アプリケーション エンジンから受け取ったプロトコル パケットを検証できます。グループ リストのアドレスに一致するパケットは処理され、グループ リスト アドレスに一致しないパケットはドロップされます。

特定クライアント、サーバ、またはクライアントとサーバのペアのキャッシングをディセーブルにするには、WCCP リダイレクト アクセス コントロール リスト（ACL）を使用します。リダイレクトACLに一致しないパケットはキャッシュをバイパスし、通常通りに転送されます。

WCCP パケットがリダイレクトされる前、deviceはインターフェイス上に設定されているすべての着信機能に関連したACLをテストし、パケットがACL内のエントリと一致するかどうかによって、パケットの転送を許可または拒否します。

（注） WCCP リダイレクト リストでは、許可と拒否の両方の ACL エントリがサポートされます。

パケットがリダイレクトされると、リダイレクトされたインターフェイスに関連付けられた出力ACLがパケットに適用されます。元のポートに関連付けられたACLは、リダイレクトされたインターフェイス上で必須出力 ACL を特に設定しない限り適用されません。

WCCP の設定方法

WCCP のデフォルト設定

機能	デフォルト設定
WCCP イネーブル ステート	WCCP サービスはディセーブルです。
プロトコル バージョン	WCCPv2
インターフェイス上で受信したトラフィックのリダイレクト	ディセーブル

キャッシュ サービスのイネーブル化

WCCP パケット リダイレクトが機能するために、クライアントに接続されたdevice インターフェイスが着信パケットをリダイレクトするように設定する必要があります。

この手順では、ルーテッドポートでこれらの機能を設定する方法を示します。これらの機能をSVI で設定するには、手順に従った設定例を参照してください。

キャッシュ サービスをイネーブルにしたり、マルチキャスト グループ アドレスまたはグループ リストを設定したり、ルーテッド インターフェイスを設定したり、クライアントから受信した着信パケットをアプリケーション エンジンにリダイレクトしたり、マルチキャスト アドレスを受信するようにインターフェイスをイネーブルにしたり、パスワードを設定したりするには、次の手順を実行します。この手順は必須です。

始める前に

SDM テンプレートを設定し、デバイスをリブートします。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip wccp** {**web-cache** | *service-number*} [**group-address** *groupaddress*] [**group-list** *access-list*] [**redirect-list** *access-list*] [**password** *encryption-number password*]
4. **interface** *interface-id*
5. **no switchport**
6. **ip address** *ip-address subnet-mask*
7. **no shutdown**
8. **exit**
9. **interface** *interface-id*
10. **no switchport**
11. **ip address** *ip-address subnet-mask*
12. **no shutdown**

13. **ip wccp** {**web-cache** | *service-number*} **redirect in**
14. **ip wccp** {**web-cache** | *service-number*} **group-listen**
15. **exit**
16. **end**
17. **show running-config**
18. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip wccp** {**web-cache** | *service-number*} [**group-address** *groupaddress*] [**group-list** *access-list*] [**redirect-list** *access-list*] [**password** *encryption-number password*] 例： スイッチ(config)# **ip wccp web-cache**	キャッシュ サービスをイネーブルにし、アプリケーション エンジンで定義されたダイナミック サービスに対応するサービス番号を指定します。デフォルトでは、この機能はディセーブルになっています。 （任意）**group-address** *groupaddress* には、devices およびアプリケーションエンジンがサービスグループに加入するときに使用するマルチキャスト グループ アドレスを指定します。 （任意） **group-list** *access-list* には、マルチキャスト グループ アドレスが使用されない場合、サービスグループに加入しているアプリケーションエンジンに対応する有効な IP アドレスのリストを指定します。 （任意）**redirect-list** *access-list* には、特定ホストのリダイレクトサービスまたはホストから特定パケットを指定します。 （任意）**password** *encryption-number password* には、暗号化番号を指定します。指定できる範囲は 0 ～ 7 です。暗号化しない場合は 0、独自の場合は 7 を使用します。7 文字以内でパスワード名を指定します。device は、パスワードと MD5 認証値を組み合わせて、device とアプリケーション エンジンと

	コマンドまたはアクション	目的
		の接続にセキュリティを確保します。デフォルトでは、パスワードは設定されておらず、認証も実行されていません。 各アプリケーション エンジンで同じパスワードを設定する必要があります。 認証がイネーブルになっている場合、deviceは認証されないメッセージを廃棄します。
ステップ 4	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/1`	アプリケーション エンジンまたはサーバに接続されたインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**no switchport** 例： `スイッチ(config-if)# no switchport`	レイヤ 3 モードを開始します。
ステップ 6	**ip address** *ip-address subnet-mask* 例： `スイッチ(config-if)# ip address 172.20.10.30 255.255.255.0`	IP アドレスとサブネット マスクを設定します。
ステップ 7	**no shutdown** 例： `スイッチ(config-if)# no shutdown`	インターフェイスをイネーブルにします。
ステップ 8	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。各アプリケーション エンジンおよびサーバにステップ 4 ～ 8 を繰り返します。
ステップ 9	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/2`	クライアントに接続するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 10	**no switchport** 例： `スイッチ(config-if)# no switchport`	レイヤ 3 モードを開始します。

	コマンドまたはアクション	目的
ステップ 11	**ip address** *ip-address subnet-mask* 例： `スイッチ(config-if)# ip address 175.20.20.10 255.255.255.0`	IP アドレスとサブネット マスクを設定します。
ステップ 12	**no shutdown** 例： `スイッチ(config-if)# no shutdown`	インターフェイスをイネーブルにします。
ステップ 13	**ip wccp** {**web-cache** | *service-number*} **redirect in** 例： `スイッチ(config-if)# ip wccp web-cache redirect in`	クライアントから受信したパケットをアプリケーション エンジンにリダイレクトします。クライアントに接続されているインターフェイス上でイネーブルにします。
ステップ 14	**ip wccp** {**web-cache** | *service-number*} **group-listen** 例： `スイッチ(config-if)# ip wccp web-cache group-listen`	（任意）マルチキャスト グループ アドレスを使用するとき、**group-listen** キーワードはインターフェイスをイネーブルにしてマルチキャストアドレスをリッスンします。アプリケーション エンジンに接続されているインターフェイス上でイネーブルにします。
ステップ 15	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。各クライアントにステップ 9 ～ 15 を繰り返します。
ステップ 16	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 17	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 18	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

設定例

次に、ルーテッド インターフェイスを設定し、マルチキャスト グループ アドレスとリダイレクト アクセス リストでキャッシュ サービスをイネーブルにする例を示します。ギガビット イーサネット ポート 1 はアプリケーション エンジンに接続され、IP アドレス 172.20.10.30 のルーテッド ポートとして設定され、再イネーブル化されています。ギガビット イーサネット ポート 2 はインターネット経由でサーバに接続され、IP アドレス 175.20.20.10 のルーテッド ポートとして設定され、再イネーブル化されています。ギガビット イーサネット ポート 3 ～ 6 はクライアントに接続され、IP アドレス 175.20.30.20、175.20.40.30、175.20.50.40、および 175.20.60.50 のルーテッド ポートとして設定されています。deviceはマルチキャスト トラフィックを受信し、クライアント インターフェイスから受信したパケットをアプリケーション エンジンにリダイレクトします。

```
スイッチ# configure terminal
スイッチ(config)# ip wccp web-cache group-address 224.1.1.100 redirect list 12
スイッチ(config)# access-list 12 permit host 10.1.1.1
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 172.20.10.30 255.255.255.0
スイッチ(config-if)# no shutdown
スイッチ(config-if)# ip wccp web-cache group-listen
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 175.20.20.10 255.255.255.0
スイッチ(config-if)# no shutdown
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/3
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 175.20.30.20 255.255.255.0
スイッチ(config-if)# no shutdown
スイッチ(config-if)# ip wccp web-cache redirect in
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/4
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 175.20.40.30 255.255.255.0
スイッチ(config-if)# no shutdown
スイッチ(config-if)# ip wccp web-cache redirect in
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/5
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 175.20.50.40 255.255.255.0
スイッチ(config-if)# no shutdown
スイッチ(config-if)# ip wccp web-cache redirect in
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/6
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 175.20.60.50 255.255.255.0
スイッチ(config-if)# no shutdown
```

```
スイッチ(config-if)# ip wccp web-cache redirect in
スイッチ(config-if)# exit
```

次に、SVIを設定し、マルチキャスト グループ リストでキャッシュ サービスをイネーブルにする例を示します。VLAN 299 は、IP アドレス 175.20.20.10 で作成および設定されています。ギガビット イーサネットのポート 1 をインターネット経由でサーバに接続し、VLAN 299 のアクセス ポートとして設定します。VLAN 300 は、IP アドレス 172.20.10.30 で作成および設定されています。ギガビット イーサネット ポート 2 はアプリケーション エンジンに接続され、VLAN 300 のアクセス ポートとして設定されています。VLAN 301 を作成し、IP アドレス 175.20.30.50 に設定します。クライアントに接続されているファスト イーサネット ポート 3 ～ 6 は、VLAN 301 のアクセス ポートとして設定されています。deviceは、クライアント インターフェイスから受信したパケットをアプリケーション エンジンにリダイレクトします。

（注） WCCP リダイレクト リストでは、許可と拒否の両方の ACL エントリがサポートされます。

```
スイッチ# configure terminal
スイッチ(config)# ip wccp web-cache group-list 15
スイッチ(config)# access-list 15 permit host 171.69.198.102
スイッチ(config)# access-list 15 permit host 171.69.198.104
スイッチ(config)# access-list 15 permit host 171.69.198.106
スイッチ(config)# vlan 299
スイッチ(config-vlan)# exit
スイッチ(config)# interface vlan 299
スイッチ(config-if)# ip address 175.20.20.10 255.255.255.0
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# switchport mode access
スイッチ(config-if)# switchport access vlan 299
スイッチ(config)# vlan 300
スイッチ(config-vlan)# exit
スイッチ(config)# interface vlan 300
スイッチ(config-if)# ip address 171.69.198.100 255.255.255.0
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# switchport mode access
スイッチ(config-if)# switchport access vlan 300
スイッチ(config-if)# exit
スイッチ(config)# vlan 301
スイッチ(config-vlan)# exit
スイッチ(config)# interface vlan 301
スイッチ(config-if)# ip address 175.20.30.20 255.255.255.0
スイッチ(config-if)# ip wccp web-cache redirect in
スイッチ(config-if)# exit
スイッチ(config)# interface range gigabitethernet1/0/3 - 6
スイッチ(config-if-range)# switchport mode access
スイッチ(config-if-range)# switchport access vlan 301
スイッチ(config-if-range)# exit
```

次のタスク

キャッシュサービスをディセーブルにするには、**no ip wccp web-cache** グローバル コンフィギュレーション コマンドを使用します。着信パケットリダイレクトをディセーブルにするには、**no ip wccp web-cache redirect in** インターフェイス コンフィギュレーション コマンドを使用します。この手順を完了した後、ネットワークでアプリケーション エンジンを設定します。

第 VII 部

QoS

第 45 章

QoS の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報と注意事項については、ご使用のプラットフォームとソフトウェアリリースに対応したリリースノートを参照してください。

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

QoS の前提条件

標準 QoS を設定する前に、次の事項を十分に理解しておく必要があります。

- 使用するアプリケーションのタイプおよびネットワークのトラフィックパターン
- トラフィックの特性およびネットワークのニーズ。たとえば、ネットワークのトラフィックがバーストであるかどうか。音声およびビデオストリーム用の帯域幅確保の必要性
- ネットワークの帯域幅要件および速度
- ネットワーク上の輻輳発生箇所

QoS ACL の注意事項

アクセス コントロール リスト（ACL）を使用して QoS 設定する場合は、次のガイドラインに従ってください。

- IP フラグメントと設定されている IP 拡張 ACL を照合することによって、QoS を実施することはできません。IP フラグメントはベストエフォート型として送信されます。IP フラグメントは IP ヘッダーのフィールドで示されます。
- 1 つのクラス マップごとに、使用できる ACL は 1 つだけであり、使用できる **match** クラスマップ コンフィギュレーション コマンドは 1 つだけです。ACL には、フィールドとパケットの内容を照合する ACE を複数指定できます。
- ポリシー マップの信頼ステートメントには、1 つの ACL 行につき複数のハードウェア エントリが必要になります。入力サービス ポリシー マップの ACL に信頼ステートメントが含まれている場合、アクセス リストが大きくなりすぎて使用可能な QoS ハードウェア メモリに収容できない可能性があり、ポリシーマップをポートに適用したときにエラーになることがあります。QoS ACL の行数はできる限り少なくする必要があります。

ポリシングの注意事項

- 複数の物理ポートを制御するポート ASIC デバイスは、256 個のポリサー（255 個のユーザー設定可能なポリサーと 1 個のシステムの内部使用向けに予約されたポリサー）をサポートします。ポートごとにサポートされるユーザー設定可能なポリサーの最大数は 63 です。ポリサーはソフトウェアによってオンデマンドで割り振られ、ハードウェアおよび ASIC の限界によって制約されます。

  ポートごとにポリサーを確保することはできません。ポートがいずれかのポリサーに割り当てられる保証はありません。
- 入力ポートでは 1 つのパケットに適用できるポリサーは 1 つだけです。設定できるのは、平均レートパラメータおよび認定バーストパラメータだけです。
- QoS 対応として設定されているポートを介して受信したすべてのトラフィックは、そのポートに結合されたポリシーマップに基づいて分類、ポリシング、およびマーキングが行われます。QoS が設定されたトランク ポートでは、そのポートを通じて受信されるすべての VLAN 内トラフィックは、ポートに付加されたポリシー マップに従って分類、ポリシング、およびマーキングが行われます。
- スイッチ上で EtherChannel ポートが設定されている場合、EtherChannel を形成する個々の物理ポートに QoS の分類、ポリシング、マッピング、およびキューイングを設定する必要があります。また、QoS の設定を EtherChannel のすべてのポートで照合するかどうかを決定する必要があります。
- 既存の QoS ポリシーのポリシー マップを変更する必要がある場合は、最初にすべてのインターフェイスからポリシーマップを削除し、その後ポリシーマップを変更またはコピーします。変更が終了したら、変更したポリシーマップをインターフェイスに適用します。

最初にすべてのインターフェイスからポリシー マップを削除しなかった場合、CPU 使用率が高くなり、コンソールが長期間停止する可能性があります。

一般的な QoS の注意事項

一般的な QoS の注意事項を次に示します。

- QoS を設定できるのは物理ポートだけです。VLAN のレベルでは QoS はサポートされていません。
- スイッチで受信された制御トラフィック（スパニングツリー ブリッジ プロトコル データ ユニット（BPDU）やルーティング アップデート パケットなど）には、入力 QoS 処理がすべて行われます。
- キュー設定を変更すると、データが失われることがあります。したがって、トラフィックが最小のときに設定を変更するようにしてください。

QoS の制約事項

以下は、QoS の制約事項を示しています。

- 次の機能を使用するには、スイッチが LAN Base イメージを実行している必要があります。スタック構成、DSCP、自動 QoS、信頼境界、ポリシング、マーキング、マッピング テーブル、および重み付けテール ドロップ。
- 入力キューイングはサポートされません。
- スイッチには 4 つのデフォルトの出力キューをサポートし、さらに 4 つの出力キューを追加して合計 8 つをイネーブルにするオプションがあります。このオプションは、LAN Base イメージを実行しているスタンドアロン スイッチにのみ使用できます。
- 設定で次の機能を実行する場合は、**mls qos srr-queue output queues 8** コマンドを使用して 8 つの出力キューをイネーブルにしないことを推奨します。
  - Auto-QoS
  - Auto SmartPort
  - EnergyWise

  スイッチでは、8 つの出力キューを単一の設定でイネーブルにしてこれらの機能を実行することはできません。
- QoS を設定できるのは物理ポートのみです。VLAN-based QoS はサポートされません。分類、キューイングおよびスケジューリングのような QoS が設定できます。また、ポートにポリシー マップも適用できます。物理ポートに QoS を設定した場合は、非階層型のポリシー マップをポートに適用します。
- スイッチが LAN Lite イメージを実行している場合、次のことが可能になります。

- ACL を設定する。ただし、それを物理インターフェイスに接続することはできません。ACL を VLAN インターフェイスに接続して CPU へのトラフィックをフィルタリングします。
- インターフェイス レベルで cos 信頼だけを有効にする。
- インターフェイス レベルで SRR シェーピングおよび共有を有効にする。
- インターフェイス レベルでプライオリティ キューイングを有効にする。
- **mls qos rewrite ip dscp**を有効または無効にします。

- 次の QoS 機能を使用するには、スイッチが LAN Base イメージを実行している必要があります。
  - ポリシー マップ
  - ポリシングおよびマーキング
  - マッピング テーブル
  - WTD

QoS の概要

QoS の実装

ネットワークは通常、ベストエフォート型の配信方式で動作します。したがって、すべてのトラフィックに等しいプライオリティが与えられ、適度なタイミングで配信される可能性はどのトラフィックでも同等です。輻輳が発生すると、すべてのトラフィックが等しくドロップされます。

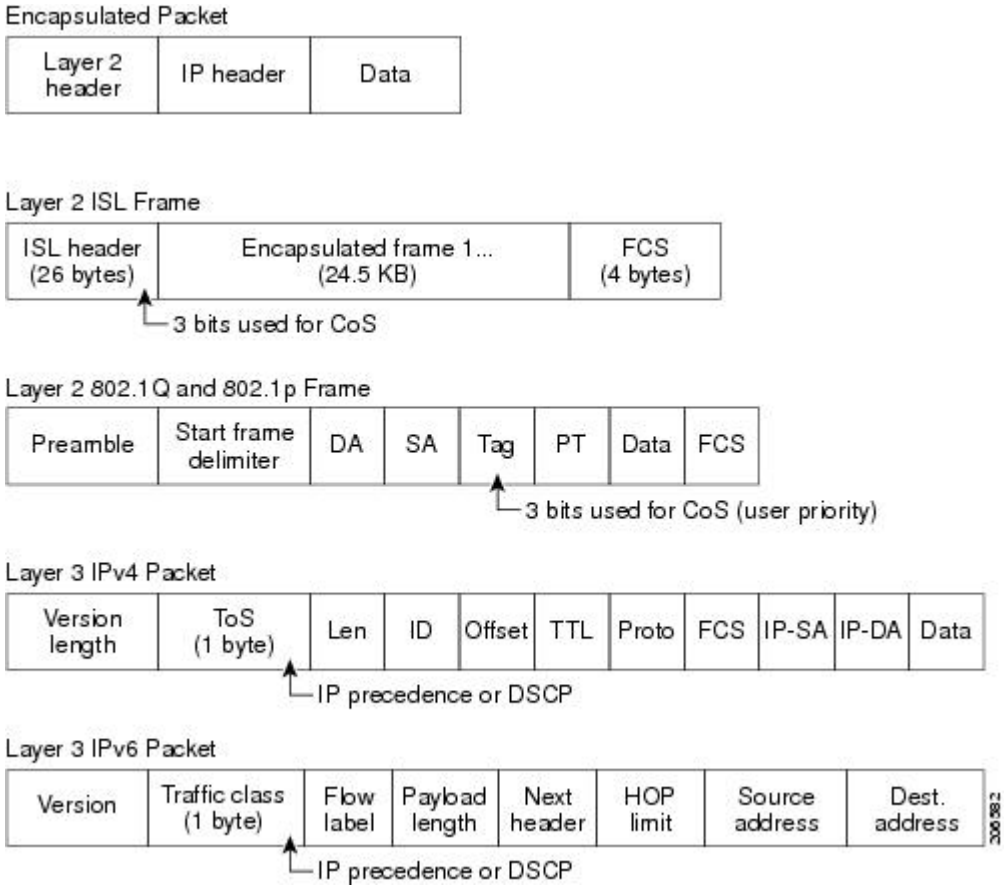
QoS 機能を設定すると、特定のネットワーク トラフィックを選択し、相対的な重要性に応じてそのトラフィックに優先度を指定し、輻輳管理および輻輳回避技術を使用して、優先処理を実行できます。ネットワークに QoS を実装すると、ネットワーク パフォーマンスがさらに予測しやすくなり、帯域幅をより効率的に利用できるようになります。

QoS は、インターネット技術特別調査委員会（IETF）の規格である Differentiated Services（Diff-Serv）アーキテクチャに基づいて実装されます。このアーキテクチャでは、ネットワークに入るときに各パケットを分類することが規定されています。

この分類は IP パケット ヘッダーに格納され、推奨されない IP タイプ オブ サービス（ToS）フィールドの 6 ビットを使用して、分類（クラス）情報として伝達されます。分類情報をレイヤ 2 フレームでも伝達できます。

図 *75:* フレームおよびパケットにおける *QoS* 分類レイヤ

次の図にレイヤ 2 フレームまたはレイヤ 3 パケットの特殊ビットを示します。



レイヤ 2 フレームのプライオリティ ビット

レイヤ 2 の ISL（スイッチ間リンク）フレーム ヘッダーには、下位 3 ビットで IEEE 802.1p サービス クラス（CoS）値を伝達する 1 バイトのユーザ フィールドがあります。レイヤ 2 ISL トランクとして設定されたポートでは、すべてのトラフィックが ISL フレームに収められます。

レイヤ 2 802.1Q フレーム ヘッダーには、2 バイトのタグ制御情報フィールドがあり、上位 3 ビット（ユーザ プライオリティ ビット）で CoS 値が伝達されます。レイヤ 2 802.1Q トランクとして設定されたポートでは、ネイティブ Virtual LAN（VLAN）のトラフィックを除くすべてのトラフィックが 802.1Q フレームに収められます。

他のフレーム タイプでレイヤ 2 CoS 値を伝達することはできません。

レイヤ 2 CoS 値の範囲は、0（ロー プライオリティ）～ 7（ハイ プライオリティ）です。

レイヤ 3 パケットのプライオリティ ビット

レイヤ 3 IP パケットは、IP precedence 値または Diffserv コード ポイント（DSCP）値のいずれかを伝送できます。DSCP 値は IP precedence 値と下位互換性があるので、QoS ではどちらの値も使用できます。

IP precedence 値の範囲は 0 ～ 7 です。DSCP 値の範囲は 0 ～ 63 です。

分類を使用したエンドツーエンドの QoS ソリューション

インターネットにアクセスするすべてのスイッチおよびルータはクラス情報に基づいて、同じクラス情報が与えられているパケットは同じ扱いで転送を処理し、異なるクラス情報のパケットはそれぞれ異なる扱いをします。パケットのクラス情報は、設定されているポリシー、パケットの詳細な検証、またはその両方に基づいて、エンドホストが割り当てるか、または伝送中にスイッチまたはルータで割り当てることができます。パケットの詳細な検証は、コア スイッチおよびルータの負荷が重くならないように、ネットワークのエッジ付近で行います。

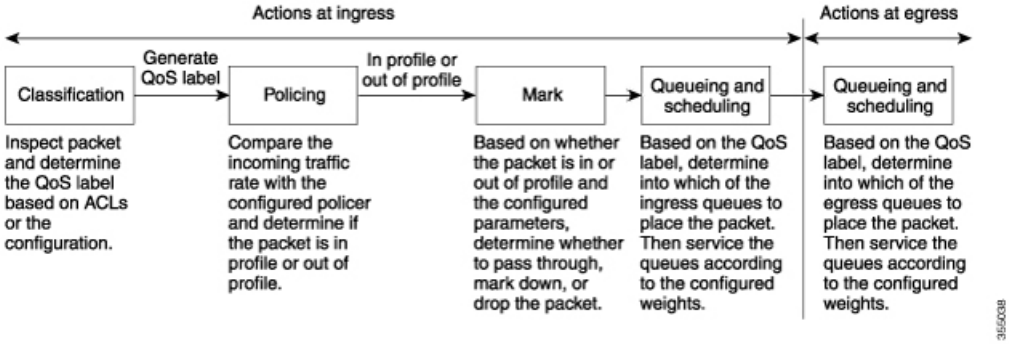
パス上のスイッチおよびルータは、クラス情報を使用して、個々のトラフィック クラスに割り当てるリソースの量を制限できます。Diff-Serv アーキテクチャでトラフィックを処理するときの、各デバイスの動作をホップ単位動作といいます。パス上のすべてのデバイスに一貫性のあるホップ単位動作をさせることによって、エンドツーエンドの QoS ソリューションを構築できます。

ネットワーク上で QoS を実装する作業は、インターネットワーキング デバイスが提供する QoS 機能、ネットワークのトラフィック タイプおよびパターン、さらには着信および発信トラフィックに求める制御のきめ細かさによって、簡単にも複雑にもなります。

QoS 基本モデル

QoS を実装するには、スイッチ上でパケットまたはフローを相互に区別し（分類）、パケットがスイッチを通過するときに所定の QoS を表すラベルを割り当て、設定されたリソース使用率制限にパケットを適合させ（ポリシングおよびマーキング）、リソース競合が発生する状況に応じて異なる処理（キューイングおよびスケジューリング）を行う必要があります。また、スイッチから送信されたトラフィックが特定のトラフィック プロファイルを満たすようにする必要もあります（シェーピング）。

図 *76 : QoS* 基本有線モデル



入力ポートでのアクション

入力ポートでのアクションには、トラフィックの分類、ポリシング、マーキング、およびスケジューリングがあります。

- パケットと QoS ラベルを関連付けて、パケットごとに異なるパスを分類します。スイッチはパケット内の CoS または DSCP を QoS ラベルにマッピングして、トラフィックの種類を区別します。生成された QoS ラベルは、このパケットでこれ以降に実行されるすべての QoS アクションを識別します。
- ポリシングでは、着信トラフィックのレートを設定済みポリサーと比較して、パケットが適合か不適合かを判別します。ポリサーは、トラフィック フローで消費される帯域幅を制限します。その判別結果がマーカーに渡されます。
- マーキングでは、パケットが不適合の場合の対処法に関して、ポリサーおよび設定情報を検討し、パケットの扱い（パケットを変更しないで通過させるか、パケットの QoS ラベルをマークダウンするか、またはパケットをドロップするか）を決定します。

（注） キューイングおよびスケジューリングは、スイッチの出力でのみサポートされ、入力ではサポートされません。

出力ポートでのアクション

出力ポートでのアクションには、キューイングおよびスケジューリングがあります。

- 4 つの出力キューのどれを使用するかを選択する前に、キューイングでは、QoS パケット ラベルおよび対応する DSCP または CoS 値を評価します。複数の入力ポートが 1 つの出力ポートに同時にデータを送信すると輻輳が発生することがあるため、WTD を使用してトラフィック クラスを区別し、QoS ラベルに基づいてパケットに別々のしきい値を適用します。しきい値を超過している場合、パケットはドロップされます。

- スケジューリングでは、設定されている SRR の共有重みまたはシェーピング重みに基づいて、4 つの出力キューを処理します。キューの 1 つ（キュー 1）は、他のキューの処理前に空になるまで処理される緊急キューにできます。

分類の概要

分類とは、パケットのフィールドを検証して、トラフィックの種類を区別するプロセスです。QoS がスイッチ上でグローバルにイネーブルになっている場合のみ、分類はイネーブルです。デフォルトでは、QoS はグローバルにディセーブルになっているため、分類は実行されません。

分類中に、スイッチは検索処理を実行し、パケットに QoS ラベルを割り当てます。QoS ラベルは、パケットに対して実行するすべての QoS アクション、およびパケットの送信元キューを識別します。

QoS ラベルは、パケット内の DSCP または CoS 値に基づいて、パケットに実行されるキューイングおよびスケジューリング アクションを決定します。QoS ラベルは信頼設定およびパケット タイプに従ってマッピングされます（分類フローチャートを参照）。

着信トラフィックの分類に、フレームまたはパケットのどのフィールドを使用するかは、ユーザ側で指定します。

Non-IP のトラフィック分類

次の表は、QoS 設定の非 IP トラフィックの分類オプションを示しています。

表 *83 :* 非 *IP* トラフィックの分類

Non-IP のトラフィック分類	説明
CoS 値の信頼	着信フレーム内の CoS 値を信頼し（CoS を信頼するようにポートを設定）、設定可能な CoS/DSCP マップを使用してパケットの DSCP 値を生成します。 レイヤ 2 の ISL フレームヘッダーは、1 バイトのユーザフィールドの下位 3 ビットで CoS 値を伝達します。 レイヤ 2 802.1Q フレームのヘッダーは、タグ制御情報フィールドの上位 3 ビットで CoS 値を伝達します。CoS 値の範囲は、0（ロー プライオリティ）～7（ハイ プライオリティ）です。

Non-IP のトラフィック分類	説明
DSCP を信頼するか、または IP precedence 値を信頼します。	着信フレームの DSCP または IP precedence 値を信頼します。これらの設定は、非 IP トラフィックの場合は無意味です。これらのいずれかの方法で設定されているポートに非 IP トラフィックが着信した場合は、CoS 値が割り当てられ、CoS/DSCP マップから内部 DSCP 値が生成されます。スイッチは内部 DSCP値を使用して、トラフィックのプライオリティを表示する CoS 値を生成します。
設定されたレイヤ 2 の MAC ACL に基づいた分類	設定されたレイヤ 2 の MAC アクセス コントロール リスト（ACL）に基づいて分類を実行します。レイヤ 2 の MAC ACL は、MAC 送信元アドレス、MAC 宛先アドレス、およびその他のフィールドを調べることができます。ACL が設定されていない場合、パケットには DSCP および CoS 値として 0 が割り当てられ、トラフィックがベストエフォート型であることを意味します。ACL が設定されている場合は、ポリシーマップ アクションによって、着信フレームに割り当てられる DSCP または CoS 値が指定されます。

分類されたパケットは、ポリシングおよびマーキングの各段階に送られます。

IP のトラフィック分類

次の表は、QoS 設定の IP トラフィック分類オプションを示します。

表 *84 : IP* のトラフィック分類

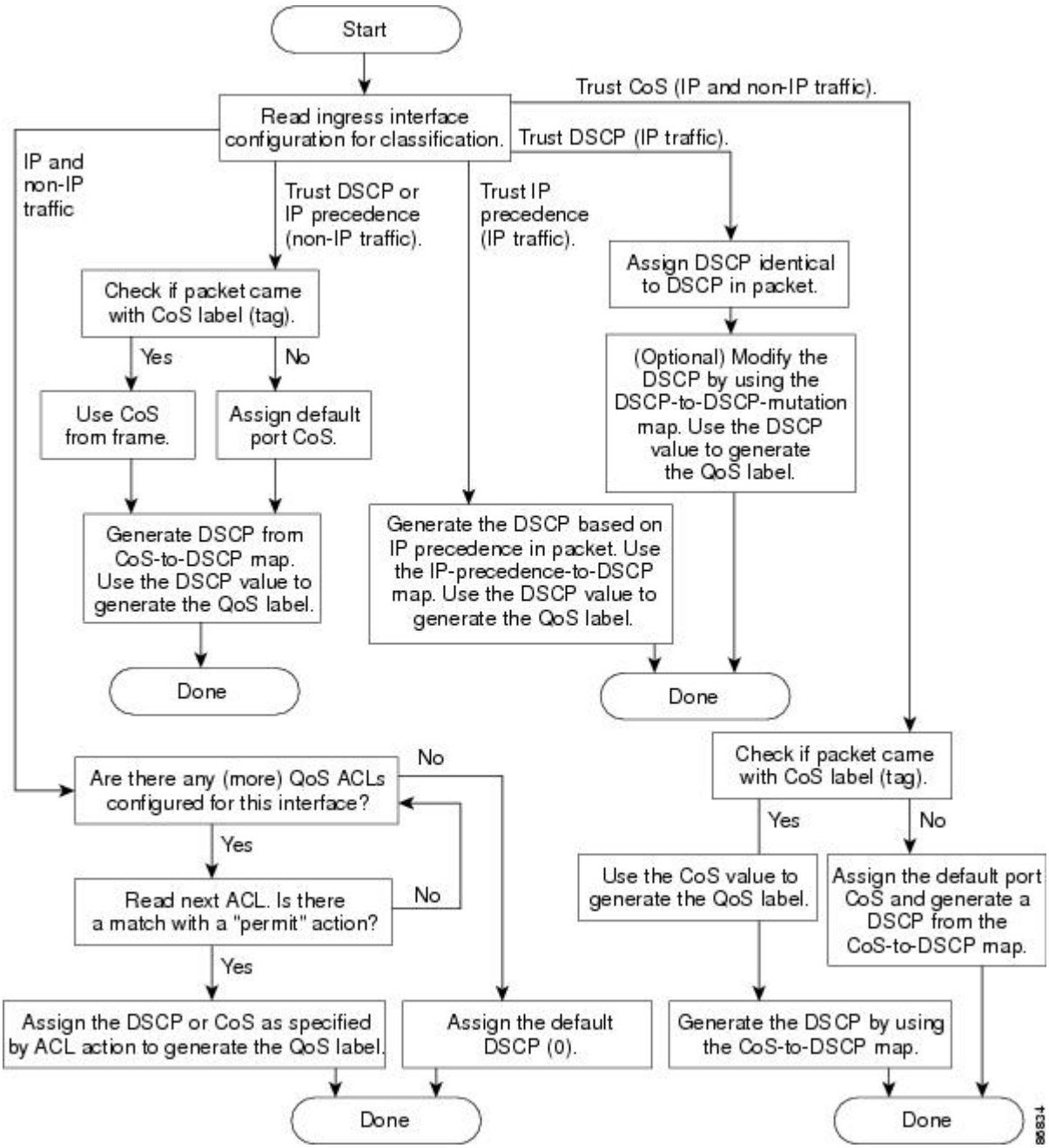
IP のトラフィック分類	説明
DSCP 値の信頼	着信パケットの DSCP 値を信頼し（DSCP を信頼するようにポートを設定し）、同じ DSCP 値をパケットに割り当てます。IETF は、1 バイトの ToS フィールドの上位 6 ビットを DSCP として定義しています。特定の DSCP 値が表すプライオリティは、設定可能です。DSCP 値の範囲は 0 ～ 63 です。 また IPv6 DSCP に基づいて IP トラフィックを分類することもできます。 2 つの QoS 管理ドメインの境界上にあるポートの場合は、設定可能な DSCP/DSCP 変換マップを使用して、DSCP を別の値に変更できます。

IP のトラフィック分類	説明
IP precedence 値の信頼	着信パケットの IP precedence 値を信頼し（IP precedence を信頼するようにポートを設定し）、設定可能な IP precedence/DSCP マップを使用してパケットの DSCP 値を生成します。IP バージョン 4 仕様では、1 バイトの ToS フィールドの上位 3 ビットが IP precedence として定義されています。IP precedence 値の範囲は 0（ロー プライオリティ）～ 7（ハイ プライオリティ）です。 また IPv6 precedence に基づいて IP トラフィックを分類することもできます。
CoS 値の信頼	着信パケットに CoS 値がある場合には、その CoS 値を信頼し、CoS/DSCP マップを使用してパケットの DSCP 値を生成します。CoS 値が存在しない場合は、デフォルトのポート CoS 値を使用します。
IP 標準または拡張 ACL	設定された IP 標準 ACL または IP 拡張 ACL（IP ヘッダーの各フィールドを調べる）に基づいて、分類を実行します。ACL が設定されていない場合、パケットには DSCP および CoS 値として 0 が割り当てられ、トラフィックがベストエフォート型であることを意味します。ACL が設定されている場合は、ポリシーマップ アクションによって、着信フレームに割り当てられる DSCP または CoS 値が指定されます。
設定された CoS の上書き	着信パケットに設定された CoS を上書きし、デフォルトのポート CoS 値を適用します。IPv6 パケットの場合、DSCP 値は CoS/DSCP マップとポートのデフォルトの CoS を使用して書き換えられます。これは、IPv4 と IPv6 の両方のトラフィックに対して実行できます。

分類されたパケットは、ポリシングおよびマーキングの各段階に送られます。

分類フローチャート

図 *77* : 分類フローチャート



アクセス コントロール リスト

IP 標準 ACL、IP 拡張 ACL、またはレイヤ 2 MAC ACL を使用すると、同じ特性を備えたパケット グループ（クラス）を定義できます。また IPv6 ACL に基づいて IP トラフィックを分類することもできます。

QoSのコンテキストでは、アクセス コントロールエントリ（ACE）の許可および拒否アクションの意味が、セキュリティ ACL の場合とは異なります。

- 許可アクションとの一致が検出されると（最初の一致の原則）、指定のQoS関連アクションが実行されます。
- 拒否アクションと一致した場合は、処理中の ACL がスキップされ、次の ACL が処理されます。

（注） 拒否アクションは Cisco IOS リリース 3.7.4E 以降のリリースでサポートされます。

- 許可アクションとの一致が検出されないまま、すべての ACE の検証が終了した場合、そのパケットでは QoS 処理は実行されず、によってベストエフォート型サービスが実行されます。
- ポートに複数の ACL が設定されている場合に、許可アクションを含む最初の ACL とパケットの一致が見つかると、それ以降の検索処理は中止され、QoS処理が開始されます。

（注） アクセス リストを作成するときは、アクセス リストの末尾に暗黙の拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。

ACL でトラフィック クラスを定義した後で、そのトラフィック クラスにポリシーを結合できます。ポリシーにはそれぞれにアクションを指定した複数のクラスを含めることができます。ポリシーには、特定の集約としてクラスを分類する（DSCP を割り当てるなど）コマンドまたはクラスのレート制限を実施するコマンドを含めることができます。このポリシーを特定のポートに結合すると、そのポートでポリシーが有効になります。

IP ACL を実装して IP トラフィックを分類する場合は、**access-list** グローバル コンフィギュレーション コマンドを使用します。レイヤ 2 MAC ACL を実装して非 IP トラフィックを分類する場合は、**mac access-list extended** グローバル コンフィギュレーション コマンドを使用します。

クラス マップおよびポリシー マップに基づく分類

ポリシー マップを使用するには、スイッチが LAN Base イメージを実行している必要があります。

クラス マップは、特定のトラフィック フロー（またはクラス）に名前を付けて、他のすべてのトラフィックと区別するためのメカニズムです。クラスマップでは、さらに細かく分類するために、特定のトラフィック フローと照合する条件を定義します。この条件には、ACL で定義されたアクセス グループとの照合、または DSCP 値や IP precedence 値の特定のリストとの照合を含めることができます。複数のトラフィック タイプを分類する場合は、別のクラスマッ

プを作成し、異なる名前を使用できます。パケットをクラスマップ条件と照合した後で、ポリシー マップを使用してさらに分類します。

ポリシー マップでは、作用対象のトラフィック クラスを指定します。トラフィック クラスのCoS、DSCP、または IP precedence 値を信頼するアクションや、トラフィック クラスに特定のDSCP または IP precedence 値を設定するアクション、またはトラフィック帯域幅の制限やトラフィックが不適合な場合の対処法を指定するアクションなどを指定できます。ポリシーマップを効率的に機能させるには、ポートにポリシー マップを結合する必要があります。

クラスマップを作成するには、**class-map** グローバル コンフィギュレーション コマンドまたは**class** ポリシー マップ コンフィギュレーション コマンドを使用します。多数のポート間でマップを共有する場合には、**class-map** コマンドを使用する必要があります。**class-map** コマンドを入力すると、クラス マップ コンフィギュレーション モードが開始されます。このモードで、**match** クラス マップ コンフィギュレーション コマンドを使用して、トラフィックの一致条件を定義します。

class class-default ポリシーマップ コンフィギュレーション コマンドを使用して、デフォルトクラスを設定できます。分類されていないトラフィック（ポリシーマップで設定された他のトラフィック クラスで指定されているトラフィック）は、デフォルト トラフィックとして処理されます。

ポリシーマップは、**policy-map** グローバル コンフィギュレーション コマンドを使用して作成し、名前を付けます。このコマンドを入力すると、ポリシー マップ コンフィギュレーションモードが開始されます。このモードでは、**class**、**trust**、または **set** ポリシー マップ コンフィギュレーション コマンドおよびポリシーマップ クラス コンフィギュレーション コマンドを使用して、特定のトラフィッククラスに対して実行するアクションを指定します。

ポリシーマップには、ポリサー、トラフィックの帯域幅制限、および制限を超えた場合のアクションを定義する **police** および **police aggregate** ポリシー マップ クラスコンフィギュレーション コマンドを含めることもできます。

ポリシーマップを有効にするには、 **service-policy** インターフェイス コンフィギュレーション コマンドを使用してポートにマップを結合します。

ポリシングおよびマーキングの概要

パケットを分類し、DSCP または CoS に基づいて QoS ラベルを割り当てたあとで、ポリシングおよびマーキング プロセスを開始できます。

ポリシングには、トラフィックの帯域幅限度を指定するポリサーの作成が伴います。制限を超えるパケットは、「アウトオブプロファイル」または「不適合」になります。各ポリサーはパケットごとに、パケットが適合か不適合かを判別し、パケットに対するアクションを指定します。これらのアクションはマーカーによって実行されます。パケットを変更しないで通過させるアクション、パケットをドロップするアクション、またはパケットに割り当てられた DSCP 値を変更（マークダウン）してパケットの通過を許可するアクションなどがあります。設定可能なポリシング済み DSCP マップを使用すると、パケットに新しい DSCP ベース QoS ラベルが設定されます。マークダウンされたパケットは、元の QoS ラベルと同じキューを使用して、フロー内のパケットの順番が崩れないようにします。

（注） すべてのトラフィックは、ブリッジングされるかルーティングされるかに関係なく、ポリサーの影響を受けます（ポリサーが設定されている場合）。その結果、ブリッジングされたパケットは、ポリシングまたはマーキングが行われたときにドロップされたり、DSCP または CoS フィールドが変更されたりすることがあります。

ポリシングは物理ポートに対して設定できます。ポリシーマップおよびポリシングアクションを設定した後で、**service-policy** インターフェイス コンフィギュレーション コマンドを使用して、ポリシーをポートに接続します。

物理ポートのポリシング

物理ポートのポリシー マップでは、次のポリサー タイプを作成できます。

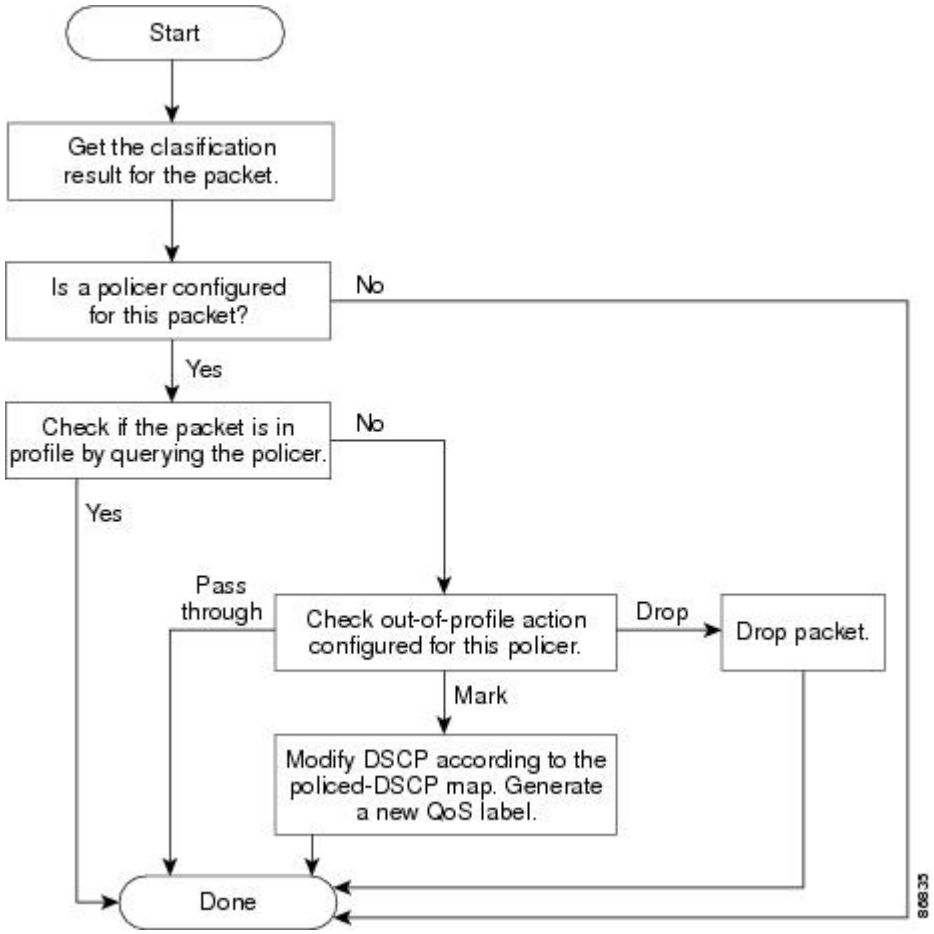
- Individual：QoS はポリサーに指定された帯域幅限度を、一致したトラフィック クラスごとに別々に適用します。このタイプのポリサーは、**police** ポリシーマップ クラス コンフィギュレーション コマンドを使用して、ポリシーマップ内で設定します。
- Aggregate：QoS はポリサーで指定された帯域幅限度を、一致したすべてのトラフィック フローに累積的に適用します。このタイプのポリサーは、**police aggregate** ポリシー マップ クラス コンフィギュレーション コマンドを使用して、ポリシーマップ内で集約ポリサー名を指定することにより設定します。ポリサーの帯域幅制限を指定するには、**mls qos aggregate-policer** グローバル コンフィギュレーション コマンドを使用します。このようにして、集約ポリサーはポリシー マップ内にある複数のトラフィック クラスで共有されます。

ポリシングはトークンバケットアルゴリズムを使用します。各フレームがスイッチに着信すると、バケットにトークンが追加されます。バケットにはホールがあり、平均トラフィック レートとして指定されたレート（ビット/秒）で送信されます。バケットにトークンが追加されるたびに、スイッチは、バケット内に十分なスペースがあるかを確認します。十分なスペースがなければ、パケットは不適合とマーキングされ、指定されたポリサーアクション（ドロップまたはマークダウン）が実行されます。

バケットが満たされる速度は、バケット深度（burst-byte）、トークンが削除されるレート（rate-bps）、および平均レートを上回るバースト期間によって決まります。バケットのサイズによってバースト長に上限が設定され、バックツーバックで送信できるフレーム数が制限されます。バースト期間が短い場合、バケットはオーバーフローせず、トラフィック フローに何のアクションも実行されません。ただし、バースト期間が長く、レートが高い場合、バケットはオーバーフローし、そのバーストのフレームに対してポリシングアクションが実行されます。

バケットの深さ（バケットがオーバーフローするまでの許容最大バースト）を設定するには、**police** ポリシーマップ クラス コンフィギュレーション コマンドの burst-byte オプションまたは **mls qos aggregate-policer** グローバル コンフィギュレーション コマンドを使用します。トークンがバケットから削除される速度（平均速度）を設定するには、**police** ポリシーマップ クラス コンフィギュレーション コマンドの *rate-bps* オプションまたは **mls qos aggregate-policer** グローバル コンフィギュレーション コマンドを使用します。

図 *78 :* 物理ポートのポリシングおよびマーキング フローチャート



マッピング テーブルの概要

QoS を処理している間、すべてのトラフィック（非 IP トラフィックを含む）のプライオリティは、分類段階で取得された DSCP または CoS 値に基づいて、QoS ラベルで表されます。

次の表は、QoS 処理とマッピング テーブルについて説明しています。

表 *85 : QoS* 処理およびマッピング テーブル

QoS 処理段階	マッピング テーブルの使用
分類	分類段階で、QoS は設定可能なマッピング テーブルを使用して、受信された CoS、DSCP、または IP precedence 値から、対応する DSCP または CoS 値を取得します。これらのマップには、CoS/DSCP マップや IP precedence/DSCP マップなどがあります。 これらのマップを設定するには、**mls qos map cos-dscp** および **mls qos map ip-prec-dscp** グローバル コンフィギュレーション コマンドを使用します。 DSCP 信頼状態で設定された入力ポートの DSCP 値が QoS ドメイン間で異なる場合は、2 つの QoS ドメイン間の境界にあるポートに、設定可能な DSCP/DSCP 変換マップを適用できます。 このマップを設定するには、**mls qos map dscp-mutation** グローバル コンフィギュレーション コマンドを使用します。
ポリシング	ポリシング段階で、QoS は IP パケットまたは非 IP パケットに別の DSCP 値を割り当てることができます（パケットが不適合で、マークダウン値がポリサーによって指定されている場合）。この設定可能なマップは、ポリシング済み DSCP マップといいます。 このマップを設定するには、**mls qos map policed-dscp** グローバル コンフィギュレーション コマンドを使用します。
プレスケジュール	トラフィックがスケジューリング段階に達する前に、QoS は QoS ラベルに従って、出力キューにパケットを格納します。QoS ラベルはパケット内の DSCP または CoS 値に基づいており、DSCP 出力キューしきい値マップまたは CoS 出力キューしきい値マップを使用してキューを選択します。出力のキューに加えて、QOS ラベルは WTD しきい値も識別します。 これらのマップを設定するには、**mls qos srr-queue** { **output**} **dscp-map** および **mls qos srr-queue** { **output**} **cos-map** グローバル コンフィギュレーション コマンドを使用します。

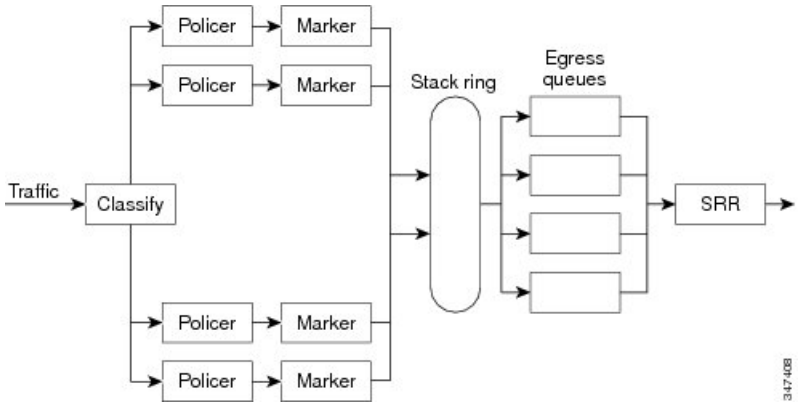
CoS/DSCP、DSCP/CoS、および IP precedence/DSCP マップのデフォルト値は、使用しているネットワークに適する場合と適さない場合があります。

デフォルトの DSCP/DSCP 変換マップおよびデフォルトのポリシング済み DSCP マップは、空のマップです。これらのマップでは、着信した DSCP 値が同じ DSCP 値にマッピングされます。DSCP/DSCP 変換マップは、特定のポートに適用できる唯一のマップです。その他のすべてのマップはスイッチ全体に適用されます。

キューイングおよびスケジューリングの概要

スイッチは、輻輳を防ぐために特定の場所にキューがあります。

図 *79 :* スイッチの出力キューの位置



（注） スイッチはデフォルトで 4 つの出力キューをサポートしますが、合計 8 つの出力キューを有効にするオプションがあります。8 出力キューの設定はスタンドアロン スイッチでのみサポートされます。

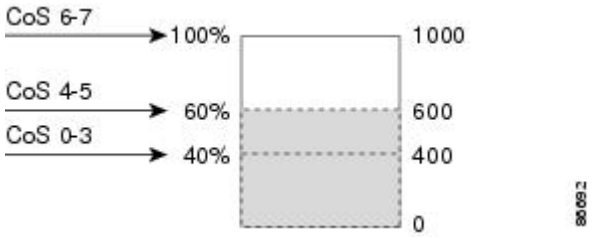
重み付けテール ドロップ

フレームが特定のキューにキューイングされると、WTD はフレームに割り当てられた QoS ラベルを使用して、それぞれ異なるしきい値を適用します。この QoS ラベルのしきい値を超えると（宛先キューの空きスペースがフレーム サイズより小さくなると）、フレームはドロップされます。

各キューには 3 つのしきい値があります。QoS ラベルは、3 つのしきい値のうちのどれがフレームの影響を受けるかを決定します。3つのしきい値のうち、2つは設定可能（明示的）で、1 つは設定不可能（暗示的）です。

図 *80 : WTD* およびキューの動作

次の図は、サイズが1000フレームであるキューでのWTDの動作の例を示しています。ドロップ割合は次のように設定されています。40%（400フレーム）、60%（600フレーム）、および100%（1000 フレーム）です。これらのパーセンテージは、40% しきい値の場合は最大 400 フレーム、60% しきい値の場合は最大 600 フレーム、100% しきい値の場合は最大 1000 フレーム



をキューイングできるという意味です。

この例では、CoS 値 6 および 7 は他の CoS 値よりも重要度が高く、100% ドロップしきい値に割り当てられます（キューフル ステート）。CoS 値 4 および 5 は 60% しきい値に、CoS 値 0 ～ 3 は 40% しきい値に割り当てられます。

600 個のフレームが格納されているキューに、新しいフレームが着信したとします。このフレームの CoS 値は 4 および 5 で、60% のしきい値が適用されます。このフレームがキューに追加されると、しきい値を超過するため、フレームは廃棄されます。

SRR のシェーピングおよび共有

出力キューでは、SRR を共有またはシェーピング用に設定できます。

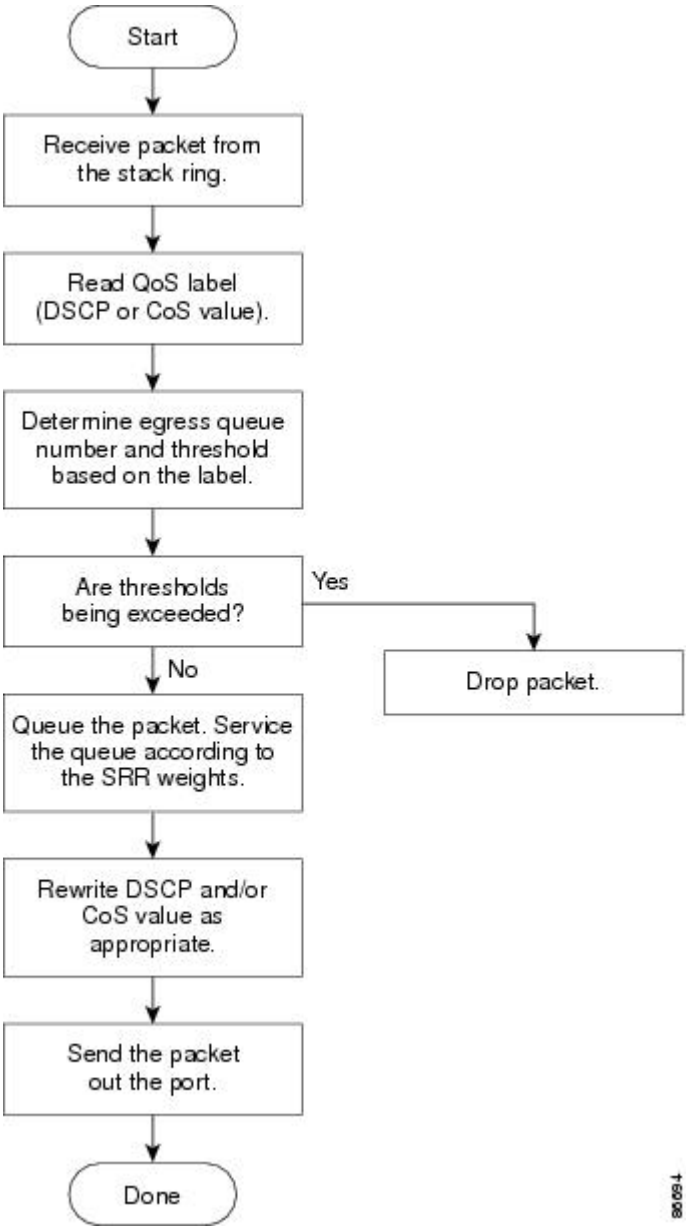
シェーピング モードでは、出力キューの帯域幅割合が保証され、この値にレートが制限されます。リンクがアイドルの場合でも、シェーピングされたトラフィックは割り当てられた帯域幅を超えて使用できません。シェーピングを使用すると、時間あたりのトラフィック フローがより均一になり、バースト トラフィックの最高時と最低時を削減します。シェーピングの場合は、各重みの絶対値を使用して、キューに使用可能な帯域幅が計算されます。

共有モードでは、設定された重みによりキュー間で帯域幅が共有されます。このレベルでは帯域幅は保証されていますが、このレベルに限定されていません。たとえば、特定のキューが空であり、リンクを共有する必要がない場合、残りのキューは未使用の帯域幅を使用して、共有できます。共有の場合、キューからパケットを取り出す頻度は重みの比率によって制御されます。重みの絶対値は関係ありません。シェーピングおよび共有は、インターフェイスごとに設定されます。各インターフェイスは、一意に設定できます。

出力キューでのキューイングおよびスケジューリング

次の図は、スイッチの出力ポートのキューイングおよびスケジューリングのフローチャートを示しています。

図 *81* : スイッチの出力ポートのキューイングおよびスケジューリング フロー チャート



（注） 緊急キューがイネーブルの場合、SRR によって空になるまで処理されてから、他の 3 つのキューが処理されます。

出力緊急キュー

各ポートは、そのうち1つ（キュー1）を出力緊急キューにできる、4つの出力キューをサポートしています。これらのキューはキューセットに割り当てられます。スイッチに存在するすべ

てのトラフィックは、パケットに割り当てられた QoS ラベルに基づいて、これらの 4 つのキューのいずれかを通過し、しきい値の影響を受けます。

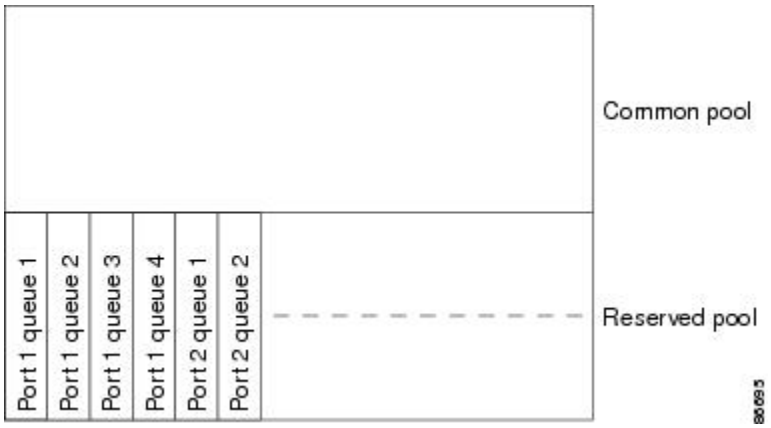
（注） 緊急キューがイネーブルの場合、SRR によって空になるまで処理されてから、他の 3 つのキューが処理されます。

出力キューのバッファ割り当て

次の図は、出力キューのバッファを示しています。

図 *82* : 出力キューのバッファ割り当て

バッファ スペースは共通プールと専用プールで構成されます。スイッチはバッファ割り当て方式を使用して、出力キューごとに最小バッファ サイズを確保します。これにより、いずれかのキューまたはポートがすべてのバッファを消費して、その他のキューのバッファが不足することがなくなり、要求元のキューにバッファ スペースを割り当てるかどうかが制御されます。スイッチは、ターゲット キューが予約量を超えるバッファを消費していないかどうか（アンダーリミット）、その最大バッファをすべて消費したかどうか（オーバーリミット）、共通のプールが空（空きバッファがない）か空でない（空きバッファ）かを判断します。キューがオーバーリミットでない場合は、スイッチは予約済みプールまたは共通のプール（空でない場合）からバッファ スペースを割り当てることができます。共通のプールに空きバッファがない場合や、キューがオーバーリミットの場合、スイッチはフレームをドロップします。



バッファおよびメモリの割り当て

バッファのアベイラビリティの保証、ドロップしきい値の設定、およびキューセットの最大メモリ割り当ての設定を行うには、**mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold* グローバル コンフィギュレーション コマンドを使用します。各しきい値はキューに割り当てられたメモリの割合です。このパーセント値を指定するには、**mls qos queue-set output** *qset-id* **buffers** *allocation1 ... allocation4* グローバル コンフィギュレーション コマンドを使用します。割り当てられたすべてのバッファの合計が専用プールになります。残りのバッファは共通プールの一部になります。

バッファ割り当てを行うと、ハイプライオリティ トラフィックを確実にバッファに格納できます。たとえば、バッファ スペースが 400 の場合、バッファ スペースの 70% をキュー 1 に割り当てて、10% をキュー 2 ～ 4 に割り当てることができます。キュー 1 には 280 バッファが割り当てられ、キュー 2 ～ 4 にはそれぞれ 40 バッファが割り当てられます。

割り当てられたバッファをキューセット内の特定のキュー用に確保するよう保証できます。たとえば、キュー用として 100 バッファがある場合、50%（50 バッファ）を確保できます。残りの 50 バッファは共通プールに戻されます。また、最大しきい値を設定することにより、いっぱいになったキューが確保量を超えるバッファを取得できるようにすることもできます。共通プールが空でない場合、必要なバッファを共通プールから割り当てることができます。

（注） スイッチはデフォルトで 4 つの出力キューをサポートしますが、合計 8 つの出力キューを有効にするオプションがあります。8 つの出力キューをすべて有効にするには、**mls qos srr-queue output queues 8** グローバル コンフィギュレーション コマンドを使用します。8 出力キューがイネーブルになったら、8 つすべてのキューのしきい値およびバッファを設定できます。8 出力キューの設定はスタンドアロン スイッチでのみサポートされます。

キューおよび WTD しきい値

スイッチを通過する各パケットをキューおよびしきい値に割り当てることができます。

特に、出力キューには DSCP または CoS 値、しきい値 ID には DSCP または CoS 値をそれぞれマッピングします。**mls qos srr-queue output dscp-map queue** *queue-id* {*dscp1...dscp8* | **threshold** *threshold-id dscp1...dscp8*} または **mls qos srr-queue output cos-map queue** *queue-id* {*cos1...cos8* | **threshold** *threshold-id cos1...cos8*} グローバル コンフィギュレーション コマンドを使用します。DSCP 出力キューしきい値マップおよび CoS 出力キューしきい値マップを表示するには、**show mls qos maps** 特権 EXEC コマンドを使用します。

キューは WTD を使用して、トラフィック クラスごとに異なるドロップ割合をサポートします。各キューには 3 つのドロップしきい値があります。そのうちの 2 つは設定可能（明示的）な WTD しきい値で、もう 1 つはキューフル ステートに設定済みの設定不可能（暗示的）なしきい値です。しきい値 ID 1 および ID 2 用の 2 つの WTD しきい値割合を割り当てます。しきい値 ID 3 のドロップしきい値は、キューフル ステートに設定済みで、変更できません。キューセットにポートをマッピングするには、**queue-set qset-id** インターフェイス コンフィギュレーション コマンドを使用します。WTD しきい値の割合を変更するには、キューセット設定を変更します。

（注） スイッチはデフォルトで 4 つの出力キューをサポートしますが、合計 8 つの出力キューを有効にするオプションがあります。8 つの出力キューをすべて有効にするには、**mls qos srr-queue output queues 8** グローバル コンフィギュレーション コマンドを使用します。8 出力キューがイネーブルになったら、8 つすべてのキューのしきい値およびバッファを設定できます。8 出力キューの設定はスタンドアロン スイッチでのみサポートされます。

シェーピング モードまたは共有モード

SRR は、シェーピング モードまたは共有モードでキューセットを処理します。キューセットにポートをマッピングするには、**queue-set** *qset-id* インターフェイス コンフィギュレーション コマンドを使用します。

共有重みまたはシェーピング重みをポートに割り当てるには、**srr-queue bandwidth share** *weight1 weight2 weight3 weight4* または **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* インターフェイス コンフィギュレーション コマンドを使用します。

バッファ割り当てと SRR 重み比率を組み合わせることにより、パケットがドロップされる前にバッファに格納して送信できるデータ量が制御されます。重みの比率は、SRR スケジューラが各キューからパケットを送信する頻度の比率です。

緊急キューがイネーブルでない限り、4 つのキューはすべて SRR に参加し、この場合、1 番めの帯域幅重みは無視されて比率計算に使用されません。緊急キューはプライオリティ キューであり、他のキューのサービスが提供される前に空になるまでサービスを提供します。緊急キューを有効にするには、**priority-queue out** インターフェイス コンフィギュレーション コマンドを使用します。

ここに記載されたコマンドを組み合わせると、特定の DSCP または CoS を持つパケットを特定のキューに格納したり、大きなキュー サイズを割り当てたり、キューをより頻繁に処理したり、プライオリティが低いパケットがドロップされるようにキューのしきい値を調整したりして、トラフィックのプライオリティを設定できます。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。出力キューについて十分理解したうえで、この設定がユーザの QoS ソリューションを満たさないと判断した場合に限り、設定を変更してください。

（注） スイッチはデフォルトで 4 つの出力キューをサポートしますが、合計 8 つの出力キューを有効にするオプションがあります。8 つの出力キューをすべて有効にするには、**mls qos srr-queue output queues 8** グローバル コンフィギュレーション コマンドを使用します。8 出力キューが有効になると、8 つすべてのキューのしきい値、バッファ、帯域幅の共有重みおよび帯域幅シェーピング重みを設定できます。8 出力キューの設定はスタンドアロン スイッチでのみサポートされます。

パケットの変更

QoS を設定するには、パケットの分類、ポリシング、キューイングを行います。QoS を提供するプロセス中に次のパケットの変更が発生することがあります。

- IP パケットおよび非 IP パケットの分類では、受信パケットの DSCP または CoS に基づいて、パケットに QoS ラベルが割り当てられます。ただし、この段階ではパケットは変更されません。割り当てられた DSCP または CoS 値の指定のみがパケットとともに伝達されます。

- ポリシング中は、IP および非 IP パケットに別の DSCP を割り当てることができます（これらのパケットが不適合で、ポリサーがマークダウン DSCP を指定している場合）。この場合も、パケット内の DSCP は変更されず、マークダウン値の指定がパケットとともに伝達されます。IP パケットの場合は、この後の段階でパケットが変更されます。非 IP パケットの場合は、DSCP が CoS に変換され、キューイングおよびスケジューリングの決定に使用されます。
- フレームに割り当てられた QoS ラベル、および選択された変換マップに応じて、フレームの DSCP および CoS 値が書き換えられます。テーブル マップを設定しない場合、および着信フレームの DSCP を信頼するようにポートが設定されている場合、フレーム内の DSCP 値は変更されませんが、CoS は、DSCP/CoS マップに基づいて書き換えられます。着信フレームの CoS を信頼するようにポートが設定されていて、着信フレームが IP パケットの場合、フレーム内の CoS 値は変更されないで、CoS/DSCP マップに従って DSCP が変更されることがあります。

  入力変換が行われると、選択された新しい DSCP 値に応じて DSCP が書き換えられます。ポリシー マップの設定アクションによっても、DSCP が書き換えられます。

標準 QoS のデフォルト設定

標準 QoS はデフォルトで無効になっています。

QoS が無効の場合は、パケットが変更されないため、信頼できるポートまたは信頼できないポートといった概念はありません。パケット内の CoS、DSCP、および IP precedence 値は変更されません。

トラフィックは Pass-Through モードでスイッチングされます。パケットは書き換えられることなくスイッチングされ、ポリシングなしのベスト エフォートに分類されます。

mls qos グローバル コンフィギュレーション コマンドを使用して QoS を有効にし、その他のすべての QoS 設定がデフォルトである場合、トラフィックはポリシングを伴わないベストエフォート型として分類されます（DSCP および CoS 値は 0 に設定されます）。ポリシー マップは設定されません。すべてのポート上のデフォルト ポートの信頼性は、信頼性なし（untrusted）の状態です。

（注） Cisco IOS リリース 15.2(1)E 以降、IPv6 QoS は、lanbase-routing テンプレートを使用して LAN ベースライセンスを実行しているスイッチでサポートされます。

出力キューのデフォルト設定

次の表は、出力キューのデフォルト設定について説明しています。

（注） スイッチはデフォルトで 4 つの出力キューをサポートしますが、合計 8 つの出力キューを有効にするオプションがあります。8 つの出力キューをすべて有効にするには、**mls qos srr-queue output queues 8** グローバル コンフィギュレーション コマンドを使用します。8 出力キューがイネーブルになったら、8 つすべてのキューのしきい値およびバッファを設定できます。8 出力キューの設定はスタンドアロン スイッチでのみサポートされます。

次の表は、QoS がイネーブルの場合の各キュー セットに対するデフォルトの出力キューを示しています。すべてのポートはキューセット 1 にマッピングされます。ポートの帯域幅限度は 100% に設定され、レートは制限されません。SRR シェーピング重み（絶対）機能では、ゼロのシェーピング重みはキューが共有モードで動作していることを示しています。SRR 共有重み機能では、帯域幅の 4 分の 1 が各キューに割り当てられます。

表 *86*：出力キューのデフォルト設定

機能	キュー 1	キュー 2	キュー 3	キュー 4
バッファ割り当て	25%	25%	25%	25%
WTD ドロップしきい値 1	100%	200%	100%	100%
WTD ドロップしきい値 2	100%	200%	100%	100%
予約済みしきい値	50%	50%	50%	50%
最大しきい値	400%	400%	400%	400%
SRR シェーピング重み（絶対）	25	0	0	0
SRR 共有重み	25	25	25	25

次の表は、QoS がイネーブルの場合のデフォルトの CoS 出力キューしきい値マップを示しています。

表 *87*：デフォルトの *CoS* 出力キューしきい値マップ

CoS 値	キュー ID-しきい値 ID
0、1	2 - 1
2、3	3 - 1
4	4 - 1
5	1 - 1

CoS 値	キュー ID-しきい値 ID
6、7	4 - 1

次の表は、QoS がイネーブルの場合のデフォルトの DSCP 出力キューしきい値マップを示しています。

表 *88 :* デフォルトの *DSCP* 出力キューしきい値マップ

DSCP 値	キュー ID-しきい値 ID
0 ～ 15	2 - 1
16 ～ 31	3 - 1
32 ～ 39	4 - 1
40 ～ 47	1 - 1
48 ～ 63	4 - 1

次の表に、**mls qos srr-queue output queues 8** コマンドを使用して 8 出力キュー設定が有効になっている場合のデフォルトの出力キューの設定を示します。

表 *89 : 8* 出力キューのデフォルト設定

機能	キュー 1	キュー 2	キュー 3	キュー 4	キュー 5	キュー 6	キュー 7	キュー 8
バッファ割り当て	10	30	10	10	10	10	10	10
WTD ドロップしきい値 1	100	1600	100	100	100	100	100	100
WTD ドロップしきい値 2	100	2000	100	100	100	100	100	100
予約済みしきい値	100	100	100	100	100	100	100	100
最大しきい値	400	2400	400	400	400	400	400	400
SRR シェーピング重み	25	0	0	0	0	0	0	0

機能	キュー 1	キュー 2	キュー 3	キュー 4	キュー 5	キュー 6	キュー 7	キュー 8
SRR 共有重み	25	25	25	25	25	25	25	25

次の表に、**mls qos srr-queue output queues 8** コマンドを使用して 8 出力キュー コンフィギュレーションが有効になっており QoS が有効な場合の、デフォルトの CoS 出力キューしきい値マップを示します。

表 *90 :* デフォルトの *CoS* 出力 *8* キューしきい値マップ

CoS	出力キュー	しきい値 ID	4 出力キュー マッピング
0	2	1	2
1	3	1	2
2	4	1	3
3	5	1	3
4	6	1	4
5	1	1	1
6	7	1	4
7	8	1	4

次の表に、**mls qos srr-queue output queues 8** コマンドを使用して 8 出力キュー コンフィギュレーションが有効になっており QoS が有効な場合の、デフォルトの DSCP 出力キューしきい値マップを示します。

表 *91 :* デフォルトの *DSCP* 出力 *8* キューしきい値マップ

DSCP	出力キュー	しきい値 ID	4 出力キュー マッピング
0 ～ 7	2	1	2
8 ～ 15	3	1	2
16 ～ 23	4	1	3
24 ～ 31	5	1	3
32 ～ 39	6	1	4
40 ～ 47	1	1	1
48 ～ 55	7	1	4

DSCP	出力キュー	しきい値 ID	4 出力キュー マッピング
56 ～ 63	8	1	4

マッピング テーブルのデフォルト設定

デフォルトの DSCP/DSCP 変換マップは、着信 DSCP 値を同じ DSCP 値にマッピングするヌルマップです。

デフォルトのポリシング済み DSCP マップは、着信 DSCP 値を同じ DSCP 値にマッピングする（マークダウンしない）空のマップです。

DSCP マップ

デフォルトの CoS/DSCP マップ

DSCP 透過モードを無効にすると、DSCP 値は次の表に従って CoS から抽出されます。これらの値が使用しているネットワークに適さない場合は、値を変更する必要があります。

（注） DSCP 透過モードはデフォルトでは無効になっています。これがイネーブルになっている場合（**no mls qos rewrite ip dscp** インターフェイス コンフィギュレーション コマンド）、DSCP の書き換えは実行されません。

表 *92 :* デフォルトの *CoS/DSCP* マップ

CoS 値	DSCP 値
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

デフォルトの IP Precedence/DSCP マップ

着信パケットの IP precedence 値を、QoS がトラフィックのプライオリティを表すために内部使用する DSCP 値にマッピングするには、IP precedence/DSCP マップを使用します。次の表は、

デフォルトの IP Precedence/DSCP マップを示しています。これらの値が使用しているネットワークに適さない場合は、値を変更する必要があります。

表 *93 :* デフォルトの *IP Precedence/DSCP* マップ

IP precedence 値	DSCP 値
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

デフォルトの DSCP/CoS マップ

4 つの出力キューのうち 1 つを選択するために使用される CoS 値を生成するには、DSCP/CoS マップを使用します。次の表に、デフォルトの DSCP/CoS マップを示します。これらの値が使用しているネットワークに適さない場合は、値を変更する必要があります。

表 *94 :* デフォルトの *DSCP/CoS* マップ

DSCP 値	CoS 値
0 ～ 7	0
8 ～ 15	1
16 ～ 23	2
24 ～ 31	3
32 ～ 39	4
40 ～ 47	5
48 ～ 55	6
56 ～ 63	7

QoS の設定方法

QoS のグローバルなイネーブル化

デフォルトでは、QoS はスイッチ上でディセーブルに設定されています。

QoS をイネーブルにするために次の手順が必要です。

手順の概要

1. **configure terminal**
2. **mls qos**
3. **end**
4. **show mls qos**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mls qos** 例： `スイッチ(config)# mls qos`	QoS をグローバルにイネーブルにします。 QoS は、次の関連トピックのセクションで説明されているデフォルト設定で動作します。 （注） QoS をディセーブルにするには、 **no mls qos** グローバル コンフィギュレーション コマンドを使用します。
ステップ 3	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show mls qos** 例： `スイッチ# show mls qos`	QoS の設定を確認します。

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

物理ポートでの VLAN ベースの QoS のイネーブル化

デフォルトでは、VLAN ベースの QoS はスイッチにあるすべての物理ポートでディセーブルです。スイッチ ポートで VLAN ベースの QoS をイネーブルにできます。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **mls qos vlan-based**
4. **end**
5. **show mls qos interface** *interface-id*
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	物理ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**mls qos vlan-based** 例： スイッチ(config-if)# **mls qos vlan-based**	ポートで VLAN ベースの QoS をイネーブルにします。 （注） 物理ポートで VLAN ベースの QoS をディセーブルにする場合は、**no mls qos vlan-based** インターフェイス コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos interface** *interface-id* 例： スイッチ# **show mls qos interface gigabitethernet 1/0/1**	VLAN ベースの QoS が物理ポートでイネーブルかどうかを確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

QoS ポリシーの設定

QoS ポリシーを設定するには、次のタスクが必要です。

- トラフィックのクラスへの分類
- 各トラフィック クラスに適用するポリシーの設定
- ポートへのポリシーの付加

ここでは、トラフィックを分類、ポリシング、マーキングする方法について説明します。ネットワーク設定に応じて、この項のモジュールの 1 つ以上を実行します。

ACL を使用したトラフィックの分類

IPv4 標準 ACLS、IPv4 拡張 ACL または IPv6 ACL を使用して IP トラフィックを分類できます。

非 IP トラフィックの分類はレイヤ 2 MAC ACL でできます。

IPv4 トラフィック用の IP 標準 ACL の作成

始める前に

この作業を実行する前に、QoS 設定のために使用するアクセス リストを決定します。

手順の概要

1. **configure terminal**

2. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] 例： スイッチ(config)# **access-list 1 permit 192.2.255.0 10.1.1.255**	IP 標準 ACL を作成し、必要な回数だけコマンドを繰り返します。 • *access-list-number* には、アクセス リスト番号を入力します。有効範囲は 1 ～ 99 および 1300 ～ 1999 です。 • **permit** キーワードを使用すると、条件が一致した場合に特定のトラフィックタイプを許可します。**deny** キーワードを使用すると、条件が一致した場合に特定のトラフィックタイプを拒否します。 • *source* には、パケットの送信元となるネットワークまたはホストを指定します。**any** キーワードは 0.0.0.0 255.255.255.255 の省略形として使用できます。 •（任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。 アクセス リストを作成するときは、アクセス リストの末尾に暗黙の拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。 （注） アクセスリストを削除するには、**no access-list** *access-list-number* グローバル コンフィギュレーション コマンドを入力します。

	コマンドまたはアクション	目的
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**show access-lists** 例： スイッチ# **show access-lists**	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv4 トラフィック用の IP 拡張 ACL の作成

始める前に

この作業を実行する前に、QoS 設定のために使用するアクセス リストを決定します。

手順の概要

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**access-list** *access-list-number* {**deny**	**permit**} *protocol source source-wildcard destination destination-wildcard* 例：	IP 拡張 ACL を作成し、必要な回数だけコマンドを繰り返します。

	コマンドまたはアクション	目的
	スイッチ(config)# **access-list 100 permit ip any any dscp 32**	• *access-list-number* には、アクセス リスト番号を入力します。有効範囲は 100 ～ 199 および 2000 ～ 2699 です。 • **permit** キーワードを使用すると、条件が一致した場合に特定のトラフィックタイプを許可します。**deny** キーワードを使用すると、条件が一致した場合に特定のトラフィックタイプを拒否します。 • *protocol* には、IP プロトコルの名前または番号を入力します。疑問符（?）を使用すると、使用できるプロトコル キーワードのリストが表示されます。 • *source* には、パケットの送信元となるネットワークまたはホストを指定します。ネットワークまたはホストを指定するには、ドット付き 10 進表記を使用したり、*source* 0.0.0.0 *source-wildcard* 255.255.255.255 の短縮形として **any** キーワードを使用したり、*source* 0.0.0.0 を表す **host** キーワードを使用します。 • *source-wildcard* では、無視するビット位置に 1 を入力することによって、ワイルドカードビットを指定します。ワイルドカードを指定するには、ドット付き 10 進表記を使用したり、*source* 0.0.0.0 *source-wildcard* 255.255.255.255 の短縮形として **any** キーワードを使用したり、*source* 0.0.0.0 を表す **host** キーワードを使用します。 • *destination* には、パケットの宛先となるネットワークまたはホストを指定します。*destination* および *destination-wildcard* には、*source* および *source-wildcard* での説明と同じオプションを使用できます。 アクセス リストを作成する際は、アクセス リストの末尾に暗黙の deny ステートメントがデフォルトで存在し、ACL の終わりに到達するまで一致が見つからなかったすべてのパケットに適用されることに注意してください。

	コマンドまたはアクション	目的
		（注） アクセスリストを削除するには、**no access-list** *access-list-number* グローバル コンフィギュレーション コマンドを入力します。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**show access-lists** 例： スイッチ# **show access-lists**	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 トラフィック用の IPv6 ACL の作成

始める前に

この作業を実行する前に、QoS 設定のために使用するアクセス リストを決定します。

手順の概要

1. **configure terminal**
2. **ipv6 access-list** *access-list-name*
3. {**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix*/ *prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input]** [**routing**] [**sequence** *value*] [**time-range** *name*]
4. **end**
5. **show ipv6 access-list**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**ipv6 access-list** *access-list-name* 例： スイッチ(config)# **ipv6 access-list ipv6_Name_ACL**	IPv6 ACL を定義し、IPv6 アクセス リスト コンフィギュレーション モードを開始します。 アクセスリスト名にはスペースまたは引用符を含めることはできません。また、数字で開始することもできません。 （注） アクセスリストを削除するには、**no ipv6 access-list** *access-list-number* グローバル コンフィギュレーション コマンドを入力します。
ステップ 3	{**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*] 例： スイッチ(config-ipv6-acl)# **permit ip host 10::1 host 11::2 host**	条件が一致した場合にパケットを拒否する場合は **deny** を指定し、許可する場合は **permit** を指定します。次に、条件について説明します。 *protocol* には、インターネットプロトコルの名前または番号を入力します。**ahp**、**esp**、**icmp**、**ipv6**、**pcp**、**stcp**、**tcp**、**udp**、または IPv6 プロトコル番号を表す 0 ～ 255 の整数を使用できます。 • *source-ipv6-prefix/prefix-length* または *destination-ipv6-prefix/ prefix-length* は、拒否条件または許可条件を設定する送信元または宛先 IPv6 ネットワークあるいはネットワーク クラスで、コロン区切りの 16 ビット値を使用した 16 進形式で指定します（RFC 2373 を参照）。 • IPv6 プレフィックス ::/0 の短縮形として、**any** を入力します。 • **host** *source-ipv6-address* または *destination-ipv6-address* には、拒否条件または許可条件を設定する送信元または宛先 IPv6 ホストアドレスを入力します。アドレスはコロン区切りの 16 ビット値を使用した 16 進形式で指定します。 •（任意）*operator* には、指定のプロトコルの送信元ポートまたは宛先ポートを比較するオペランドを指定します。オペランドには、**lt**（より小さい）、**gt**（より大きい）、**eq**（等しい）、**neq**（等しくない）、および **range**（包含範囲）があります。

	コマンドまたはアクション	目的
		source-ipv6-prefix/prefix-length 引数のあとの operator は、送信元ポートに一致する必要があります。*destination-ipv6- prefix/prefix-length* 引数のあとの operator は、宛先ポートに一致する必要があります。 • （任意）*port-number* は、0 ～ 65535 の 10 進数または TCP あるいは UDP ポートの名前です。TCP ポート名を使用できるのは、TCP のフィルタリング時だけです。UDP ポート名を使用できるのは、UDP のフィルタリング時だけです。 • （任意）**dscp** *value* を入力して、各 IPv6 パケットヘッダーの Traffic Class フィールド内のトラフィッククラス値と DiffServ コードポイント値を照合します。指定できる範囲は 0 ～ 63 です。 • （任意）**fragments** を入力して、先頭ではないフラグメントを確認します。このキーワードが表示されるのは、プロトコルが IPv6 の場合だけです。 • （任意）**log** を指定すると、エントリと一致するパケットに関するログメッセージがコンソールに送信されます。**log-input** を指定すると、ログエントリに入力インターフェイスが追加されます。ロギングはルータ ACL でだけサポートされます。 • （任意）**routing** を入力して、IPv6 パケットのルーティングを指定します。 • （任意）**sequence** *value* を入力して、アクセスリスト ステートメントのシーケンス番号を指定します。指定できる範囲は 1 ～ 4294967295 です。 • （任意）**time-range** *name* を入力して、拒否または許可ステートメントに適用される時間の範囲を指定します。
ステップ 4	**end** 例： `スイッチ(config-ipv6-acl)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 5	**show ipv6 access-list** 例： スイッチ# **show ipv6 access-list**	アクセス リストの設定を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

非 IP トラフィック用のレイヤ 2 MAC ACL の作成

始める前に

この作業を実行する前に、レイヤ 2 の MAC アクセス リストが QoS 設定に必要であることを決定します。

手順の概要

1. **configure terminal**
2. **mac access-list extended** *name*
3. {**permit** | **deny**} { **host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]
4. **end**
5. **show access-lists** [*access-list-number* | *access-list-name*]
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mac access-list extended** *name* 例： スイッチ(config)# **mac access-list extended maclist1**	リストの名前を指定することによって、レイヤ 2 MAC ACL を作成します。 このコマンドを入力すると、拡張 MAC ACL コンフィギュレーション モードに切り替わります。

	コマンドまたはアクション	目的
		（注） アクセスリストを削除するには、 **no mac access-list extended** *access-list-name* グローバル コンフィギュレーション コマンドを入力します。
ステップ 3	{**permit** | **deny**} { **host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*] 例： `スイッチ(config-ext-mac1) # permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0` `スイッチ(config-ext-mac1) # permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp`	条件が一致した場合に許可または拒否するトラフィック タイプを指定します。必要な回数だけコマンドを入力します。 • *src-MAC-addr* には、パケットの送信元となるホストの MAC アドレスを指定します。MAC アドレスを指定するには、16 進表記（H.H.H）を使用したり、*source* 0.0.0、*source-wildcard* ffff.ffff.ffff の短縮形として **any** キーワードを使用したり、*source* 0.0.0 を表す **host** キーワードを使用します。 • *mask* では、無視するビット位置に 1 を入力することによって、ワイルドカード ビットを指定します。 • *dst-MAC-addr* には、パケットの宛先となるホストの MAC アドレスを指定します。MAC アドレスを指定するには、16 進表記（H.H.H）を使用したり、*source* 0.0.0、*source-wildcard* ffff.ffff.ffff の短縮形として **any** キーワードを使用したり、*source* 0.0.0 を表す **host** キーワードを使用します。 • （任意）*type mask* には、Ethernet II または SNAP でカプセル化されたパケットの Ethertype 番号を指定して、パケットのプロトコルを識別します。*type* の範囲は 0 ～ 65535 です。通常は 16 進数で指定します。*mask* では、一致をテストする前に Ethertype に適用される *don't care* ビットを入力します。 アクセス リストを作成するときは、アクセス リストの末尾に暗黙の拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。
ステップ 4	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config-ext-mac1)# **end**	
ステップ 5	**show access-lists** [*access-list-number* | *access-list-name*] 例： スイッチ# **show access-lists**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

クラス マップによるトラフィックの分類

個々のトラフィックフロー（またはクラス）を他のすべてのトラフィックから分離して名前を付けるには、**class-map** グローバル コンフィギュレーション コマンドを使用します。クラス マップでは、さらに細かく分類するために、特定のトラフィック フローと照合する条件を定義します。match ステートメントには、ACL、IP precedence 値、DSCP 値などの条件を指定できます。一致条件は、クラス マップ コンフィギュレーション モードの中で match ステートメントを 1 つ入力することによって定義します。

（注） **class** ポリシーマップ コンフィギュレーション コマンドを使用することによって、ポリシーマップの作成時にクラスマップを作成することもできます。

手順の概要

1. **configure terminal**
2. 次のいずれかを使用します。
   - **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
   - **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*]
   - **ipv6 access-list** *access-list-name* {**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix*/ *prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]
   - **mac access-list extended** *name* {**permit** | **deny**} { **host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **match** { **access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}

5. **end**
6. **show class-map**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	次のいずれかを使用します。 • **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] • **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] • **ipv6 access-list** *access-list-name* {**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*] • **mac access-list extended** *name* {**permit** | **deny**} { **host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*] 例： `スイッチ(config)# access-list 103 permit ip any any dscp 10`	必要な回数だけコマンドを繰り返し、IP標準または IP 拡張 ACL、IP トラフィック用の IPv6 ACL、または非 IP トラフィック用のレイヤ 2 MAC ACL を作成します。 アクセス リストを作成するときは、アクセス リストの末尾に暗黙の拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。
ステップ 3	**class-map** [**match-all** | **match-any**] *class-map-name* 例： `スイッチ(config)# class-map class1`	クラスマップを作成し、クラスマップ コンフィギュレーション モードを開始します。 デフォルトでは、クラスマップは定義されていません。 • （任意）このクラスマップ配下のすべての一致ステートメントの論理 AND を実行するには、**match-all** キーワードを使用します。この場合は、クラスマップ内のすべての一致条件と一致する必要があります。

	コマンドまたはアクション	目的
		• （任意）このクラスマップ配下のすべての一致ステートメントの論理 OR を実行するには、**match-any** キーワードを使用します。この場合は、1 つまたは複数の一致条件と一致する必要があります。 • *class-map-name* には、クラス マップ名を指定します。 **match-all** または **match-any** キーワードのいずれも指定しない場合は、**match-all** がデフォルトです。 （注） 既存のクラスマップを削除するには、**no class-map** [**match-all** | **match-any**] *class-map-name* グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**match** { **access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*} 例： `スイッチ(config-cmap)# match ip dscp 10 11 12`	トラフィックを分類するための一致条件を定義します。 デフォルトでは、一致条件は定義されていません。 クラスマップごとにサポートされる一致条件は 1 つだけです。また、クラスマップごとにサポートされる ACL は 1 つだけです。 • **access-group** *acl-index-or-name* には、ステップ 2 で作成した ACL の番号または名前を指定します。 • IPv6 トラフィックを **match access-group** コマンドでフィルタリングするには、ステップ 2 の手順で IPv6 ACL を作成します。 • **ip dscp** *dscp-list* には、着信パケットと照合する IP DSCP 値を 8 つまで入力します。各値はスペースで区切ります。指定できる範囲は 0 ～ 63 です。 • **ip precedence** *ip-precedence-list* には、着信パケットと照合する IP precedence 値を 8 つまで入力します。各値はスペースで区切ります。指定できる範囲は 0 ～ 7 です。 （注） 一致条件を削除するには、 **no match** { **access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} クラスマップ コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config-cmap)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show class-map** 例： スイッチ# **show class-map**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

クラス マップの使用と IPv6 トラフィックのフィルタリングによるトラフィックの分類

プライマリ一致基準を IPv4 トラフィックに対してのみ適用するには **match protocol** コマンドで **ip** キーワードを使用します。プライマリ一致基準を IPv6 トラフィックに対してのみ適用するには **match protocol** コマンドで **ipv6** キーワードを使用します。

手順の概要

1. **configure terminal**
2. **class-map** {**match-all**} *class-map-name*
3. **match protocol**[*ip* |*ipv6*]
4. **match** {**ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}
5. **end**
6. **show class-map**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**class-map** {**match-all**} *class-map-name* 例：	クラスマップを作成し、クラスマップ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ(config)# **class-map cm-1**	デフォルトでは、クラスマップは定義されていません。 **match protocol** コマンドを使用すると、**match-all** キーワードのみがサポートされます。 • *class-map-name* には、クラス マップ名を指定します。 **match-all** または **match-any** キーワードのいずれも指定しない場合は、**match-all** がデフォルトです。 （注） 既存のクラスマップを削除するには、**no class-map** [**match-all** | **match-any**] *class-map-name* グローバル コンフィギュレーション コマンドを使用します。
ステップ 3	**match protocol**[*ip* |*ipv6*] 例： スイッチ(config-cmap)# **match protocol ip**	（任意）クラス マップを適用する IP プロトコルを指定します。 • IPv4 トラフィックを指定するには引数 *ip*、IPv6 トラフィックを指定するには *ipv6* をそれぞれ指定します。 • **match protocol** コマンドを使用すると、**class-map** コマンドで **match-all** キーワードのみがサポートされます。
ステップ 4	**match** {**ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*} 例： スイッチ(config-cmap)# **match ip dscp 10**	トラフィックを分類するための一致条件を定義します。 デフォルトでは、一致条件は定義されていません。 • **ip dscp** *dscp-list* には、着信パケットと照合する IP DSCP 値を 8 つまで入力します。各値はスペースで区切ります。指定できる範囲は 0 ～ 63 です。 • **ip precedence** *ip-precedence-list* には、着信パケットと照合する IP precedence 値を 8 つまで入力します。各値はスペースで区切ります。指定できる範囲は 0 ～ 7 です。 （注） 一致条件を削除するには、no match {**access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} クラスマップ コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config-cmap)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show class-map** 例： スイッチ# **show class-map**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ポリシーマップによる物理ポートのトラフィックの分類、ポリシング、およびマーキング

作用対象となるトラフィック クラスを指定するポリシー マップを、物理ポート上に設定できます。トラフィック クラスの CoS 値、DSCP 値、または IP precedence 値を信頼するアクション、トラフィック クラスに特定の DSCP 値または IP precedence 値を設定するアクション、および一致する各トラフィック クラスにトラフィック帯域幅限度を指定するアクション（ポリサー）や、トラフィックが不適合な場合の対処法を指定するアクション（マーキング）などを指定できます。

ポリシー マップには、次の特性もあります。

- 1 つのポリシーマップに、それぞれ異なる一致条件とポリサーを指定した複数のクラス ステートメントを指定できます。
- ポリシー マップには、事前に定義されたデフォルトのトラフィック クラスを含めることができます。デフォルトのトラフィック クラスはマップの末尾に明示的に配置されます。
- 1 つのポートから受信されたトラフィック タイプごとに、別々のポリシー マップ クラスを設定できます。

物理ポートでポリシー マップを設定する場合には、次の注意事項に従ってください。

- 入力ポートごとに付加できるポリシー マップは、1 つだけです。
- **mls qos map ip-prec-dscp** *dscp1...dscp8* グローバル コンフィギュレーション コマンドを使用して IP-precedence/DSCP マップを設定する場合、その設定は IP precedence 値を信頼するよう設定されている入力インターフェイス上のパケットにのみ影響を与えます。ポリシー マップでは、 **set ip precedence** *new-precedence* ポリシー マップ クラス コンフィギュレー

ション コマンドを使用してパケット IP precedence 値に新規の値を設定すると、出力 DSCP 値は IP precedence/DSCP マップからは影響を受けません。出力 DSCP 値を入力値とは異なる値に設定する場合、**set dscp new-dscp** ポリシー マップ クラス コンフィギュレーション コマンドを使用します。

- **set ip dscp** コマンドを入力または使用した場合 、はこのコマンドをコンフィギュレーション内で **set dscp** に変更します。
- **set ip precedence** または **set precedence** ポリシーマップ クラス コンフィギュレーション コマンドを使用すると、パケット IP precedence 値を変更できます。 コンフィギュレーションではこの設定は set ip precedence として表示されます。
- ポリシーマップとポート信頼状態は、両方とも物理インターフェイス上で有効にすることができます。ポリシー マップは、ポート信頼状態の前に適用されます。
- **class class-default** ポリシーマップ コンフィギュレーション コマンドを使用してデフォルトのトラフィッククラスを設定すると、未分類トラフィック（トラフィッククラスで指定された一致基準に一致しないトラフィック）はデフォルトのトラフィッククラス（class-default）として処理されます。

手順の概要

1. **configure terminal**
2. **class-map** [**match-all** | **match-any**] *class-map-name*
3. **policy-map** *policy-map-name*
4. **class** [*class-map-name* | **class-default**]
5. **trust**[**cos** |**dscp** |**ip-precedence**]
6. **set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*}
7. **police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]
8. **exit**
9. **exit**
10. **interface** *interface-id*
11. **service-policy input** *policy-map-name*
12. **end**
13. **show policy-map** [*policy-map-name* [**class** *class-map-name*]]
14. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**class-map** [**match-all** | **match-any**] *class-map-name* 例： `スイッチ(config)# class-map ipclass1`	クラスマップを作成し、クラスマップ コンフィギュレーション モードを開始します。 デフォルトでは、クラス マップは定義されていません。 • （任意）このクラスマップ配下のすべての一致ステートメントの論理 AND を実行するには、**match-all** キーワードを使用します。この場合は、クラス マップ内のすべての一致条件と一致する必要があります。 • （任意）このクラスマップ配下のすべての一致ステートメントの論理 OR を実行するには、**match-any** キーワードを使用します。この場合は、1 つまたは複数の一致条件と一致する必要があります。 • *class-map-name* には、クラス マップ名を指定します。 **match-all** または **match-any** キーワードのいずれも指定しない場合は、**match-all** がデフォルトです。
ステップ 3	**policy-map** *policy-map-name* 例： `スイッチ(config-cmap)# policy-map flowit`	ポリシーマップ名を入力することによってポリシーマップを作成し、ポリシーマップ コンフィギュレーション モードを開始します。 デフォルトでは、ポリシー マップは定義されていません。 ポリシー マップのデフォルトの動作では、パケットが IP パケットの場合は DSCP が 0 に、パケットがタグ付きの場合は CoS が 0 に設定されます。ポリシングは実行されません。 （注） 既存のポリシーマップを削除するには、**no policy-map** *policy-map-name* グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**class** [*class-map-name* | **class-default**] 例： `スイッチ(config-pmap)# class ipclass1`	トラフィックの分類を定義し、ポリシーマップ クラスコンフィギュレーションモードを開始します。 デフォルトでは、ポリシー マップ クラス マップは定義されていません。

	コマンドまたはアクション	目的
		すでに **class-map** グローバル コンフィギュレーション コマンドを使用してトラフィック クラスが定義されている場合は、このコマンドで *class-map-name* にその名前を指定します。 **class-default** トラフィック クラスは定義済みで、どのポリシーにも追加できます。このトラフィック クラスは、常にポリシー マップの最後に配置されます。暗黙の **match any** が **class-default** クラスに含まれている場合、他のトラフィック クラスと一致しないパケットはすべて **class-default** と一致します。 （注） 既存のクラスマップを削除するには、**no class** *class-map-name* ポリシーマップ コンフィギュレーション コマンドを使用します。
ステップ 5	**trust**[**cos** |**dscp** |**ip-precedence**] 例： `スイッチ(config-pmap-c)# trust dscp`	CoS ベースまたは DSCP ベースの QoS ラベルを生成するために QoS が使用する信頼ステートを設定します。 このコマンドと **set** コマンドは、同じポリシーマップ内で相互に排他的になります。**trust** コマンドを入力する場合は 、ステップ 6 に進みます。 デフォルトでは、ポートは trusted ではありません。キーワードを指定せずにコマンドを入力した場合、デフォルトは **dscp** です。 キーワードの意味は次のとおりです。 • **cos**：QoS は受信した CoS 値やデフォルトのポート CoS 値、および CoS/DSCP マップを使用して、DSCP 値を抽出します。 • **dscp**：QoS は入力パケットの DSCP 値を使用して、DSCP 値を抽出します。タグ付きの非 IP パケットの場合、QoS は受信した CoS 値を使用して DSCP 値を抽出します。タグなしの非 IP パケットの場合、QoS はデフォルトのポート CoS 値を使用して DSCP 値を抽出します。いずれの場合も、DSCP 値は CoS/DSCP マップから抽出されます。 • **ip-precedence**：QoS は入力パケットの IP precedence 値および IP precedence/DSCP マップを使用して、DSCP 値を抽出します。タグ付きの非 IP パケットの場合、QoS は受信した CoS

	コマンドまたはアクション	目的
		値を使用してDSCP値を抽出します。タグなしの非 IP パケットの場合、QoS はデフォルトのポート CoS 値を使用して DSCP 値を抽出します。いずれの場合も、DSCP 値は CoS/DSCP マップから抽出されます。 （注） untrusted ステートに戻すには、**no trust** ポリシーマップ コンフィギュレーション コマンドを使用します。
ステップ 6	**set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*} 例： `スイッチ(config-pmap-c)# set dscp 45`	パケットに新しい値を設定することによって、IP トラフィックを分類します。 • **dscp** *new-dscp* には、分類されたトラフィックに割り当てる新しいDSCP値を入力します。指定できる範囲は 0 ～ 63 です。 • **ip precedence** *new-precedence* には、分類されたトラフィックに割り当てる新しいIP precedence値を入力します。指定できる範囲は 0 ～ 7 です。 （注） 割り当てられたDSCPまたはIP precedence値を削除するには、**no set** {**dscp** *new-dscp* | **ip** precedence *new-precedence*} ポリシーマップ コンフィギュレーション コマンドを使用します。
ステップ 7	**police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}] 例： `スイッチ(config-pmap-c)# police 100000 80000 drop`	分類したトラフィックにポリサーを定義します。 デフォルトでは、ポリサーは定義されていません。 • *rate-bps* には、平均トラフィック レートをビット/秒（bps）で指定します。指定できる範囲は 8000 ～ 10000000000 です • *burst-byte* には、標準バースト サイズをバイト数で指定します。指定できる範囲は 8000 ～ 1000000 です。 • （任意）レートを超過した場合に実行するアクションを指定します。パケットをドロップするには、**exceed-action drop** キーワードを使用します。（ポリシング済みDSCPマップを使用して）DSCP値をマークダウンし、パケットを送信するには、**exceed-action policed-dscp-transmit** キーワードを使用します。

	コマンドまたはアクション	目的
		（注） 既存のポリサーを削除するには、**no police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}] ポリシーマップ コンフィギュレーション コマンドを使用します。
ステップ 8	**exit** 例： `スイッチ(config-pmap-c)# exit`	ポリシー マップ コンフィギュレーション モードに戻ります。
ステップ 9	**exit** 例： `スイッチ(config-pmap)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 10	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/1`	ポリシー マップを適用するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。 有効なインターフェイスには、物理ポートが含まれます。
ステップ 11	**service-policy input** *policy-map-name* 例： `スイッチ(config-if)# service-policy input flowit`	ポリシー マップ名を指定し、入力ポートに適用します。 サポートされるポリシー マップは、入力ポートに 1 つだけです。 （注） ポリシーマップとポートの関連付けを解除するには、 **no service-policy***input policy-map-name* インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 12	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 13	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] 例：	入力を確認します。

	コマンドまたはアクション	目的
	スイッチ# **show policy-map**	
ステップ 14	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

集約ポリサーによるトラフィックの分類、ポリシング、およびマーキング

集約ポリサーを使用すると、同じポリシー マップ内の複数のトラフィック クラスで共有されるポリサーを作成できます。ただし、集約ポリサーを複数の異なるポリシーマップまたはポートにわたって使用することはできません。

集約ポリサーは、物理ポートの非階層型ポリシー マップにだけ設定できます。

手順の概要

1. **configure terminal**
2. **mls qos aggregate-policer** *aggregate-policer-name rate-bps burst-byte* **exceed-action** {**drop** | **policed-dscp-transmit**}
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **policy-map** *policy-map-name*
5. **class** [*class-map-name* | **class-default**]
6. **police aggregate** *aggregate-policer-name*
7. **exit**
8. **interface** *interface-id*
9. **service-policy input** *policy-map-name*
10. **end**
11. **show mls qos aggregate-policer** [*aggregate-policer-name*]
12. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mls qos aggregate-policer** *aggregate-policer-name rate-bps burst-byte* **exceed-action** {**drop**	**policed-dscp-transmit**}	同じポリシー マップ内の複数のトラフィック クラスに適用できるポリサーパラメータを定義します。

	コマンドまたはアクション	目的
	例： スイッチ(config)# **mls qos aggregate-police transmit1 48000 8000 exceed-action policed-dscp-transmit**	デフォルトでは、集約ポリサーは定義されていません。 • *aggregate-policer-name* には、集約ポリサーの名前を指定します。 • *rate-bps* には、平均トラフィック レートをビット/秒（bps）で指定します。指定できる範囲は 8000 ～ 10000000000 です • *burst-byte* には、標準バースト サイズをバイト数で指定します。指定できる範囲は 8000 ～ 1000000 です。 • レートを超過した場合に実行するアクションを指定します。パケットをドロップするには、**exceed-action drop** キーワードを使用します。（ポリシング済み DSCP マップを使用して）DSCP 値をマークダウンし、パケットを送信するには、**exceed-action policed-dscp-transmit** キーワードを使用します。
ステップ 3	**class-map** [**match-all** | **match-any**] *class-map-name* 例： スイッチ(config)# **class-map ipclass1**	必要に応じて、トラフィックを分類するクラスマップを作成します。
ステップ 4	**policy-map** *policy-map-name* 例： スイッチ(config-cmap)# **policy-map aggflow1**	ポリシーマップ名を入力することによってポリシーマップを作成し、ポリシーマップ コンフィギュレーション モードを開始します。
ステップ 5	**class** [*class-map-name* | **class-default**] 例： スイッチ(config-cmap-p)# **class ipclass1**	トラフィックの分類を定義し、ポリシーマップ クラス コンフィギュレーション モードを開始します。
ステップ 6	**police aggregate** *aggregate-policer-name* 例： スイッチ(configure-cmap-p)# **police aggregate transmit1**	同じポリシー マップ内の複数のクラスに集約ポリサーを適用します。 *aggregate-policer-name* には、ステップ 2 で指定した名前を入力します。 指定された集約ポリサーをポリシーマップから削除するには、**no police aggregate** *aggregate-policer-name*

	コマンドまたはアクション	目的
		ポリシー マップ コンフィギュレーション コマンドを使用します。集約ポリサーおよびそのパラメータを削除するには、**no mls qos aggregate-policer** *aggregate-policer-name* グローバル コンフィギュレーション コマンドを使用します。
ステップ 7	**exit** 例： `スイッチ(configure-cmap-p)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 8	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/1`	ポリシー マップを適用するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。 有効なインターフェイスには、物理ポートが含まれます。
ステップ 9	**service-policy input** *policy-map-name* 例： `スイッチ(config-if)# service-policy input aggflow1`	ポリシー マップ名を指定し、入力ポートに適用します。 サポートされるポリシー マップは、入力ポートに 1 つだけです。
ステップ 10	**end** 例： `スイッチ(configure-if)# end`	特権 EXEC モードに戻ります。
ステップ 11	**show mls qos aggregate-policer** [*aggregate-policer-name*] 例： `スイッチ# show mls qos aggregate-policer transmit1`	入力を確認します。
ステップ 12	**copy running-config startup-config** 例： `スイッチ# copy-running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

DSCP マップの設定

CoS/DSCP マップの設定

CoS/DSCP マップを使用して、着信パケットの CoS 値を、QoS がトラフィックのプライオリティを表すために内部使用する DSCP 値にマッピングします。

CoS/DSCP マップを変更するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **mls qos map cos-dscp** *dscp1...dscp8*
3. **end**
4. **show mls qos maps cos-dscp**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mls qos map cos-dscp** *dscp1...dscp8* 例： スイッチ(config)# **mls qos map cos-dscp 10 15 20 25 30 35 40 45**	CoS/DSCP マップを変更します。 *dscp1...dscp8* には、CoS 値 0 ～ 7 に対応する 8 つの DSCP 値を入力します。各 DSCP 値はスペースで区切ります。 DSCP の範囲は 0 ～ 63 です。 （注） デフォルトのマップに戻すには、 **no mls qos cos-dscp** グローバル コンフィギュレーション コマンドを使用します。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**show mls qos maps cos-dscp** 例：	入力を確認します。

	コマンドまたはアクション	目的
	スイッチ# **show mls qos maps cos-dscp**	
ステップ 5	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IP precedence/DSCP マップの設定

着信パケットの IP precedence 値を、QoS がトラフィックのプライオリティを表すために内部使用する DSCP 値にマッピングするには、IP precedence/DSCP マップを使用します。

IP precedence/DSCP マップを変更するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **mls qos map ip-prec-dscp** *dscp1...dscp8*
3. **end**
4. **show mls qos maps ip-prec-dscp**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mls qos map ip-prec-dscp** *dscp1...dscp8* 例： スイッチ(config)# **mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45**	IP precedence/DSCP マップを変更します。 *dscp1...dscp8* には、IP precedence 値 0 ～ 7 に対応する 8 つの DSCP 値を入力します。各 DSCP 値はスペースで区切ります。 DSCP の範囲は 0 ～ 63 です。 （注） デフォルトのマップに戻すには、**no mls qos ip-prec-dscp** グローバル コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**show mls qos maps ip-prec-dscp** 例： スイッチ# **show mls qos maps ip-prec-dscp**	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ポリシング済み DSCP マップの設定

ポリシングおよびマーキング アクションによって得られる新しい値に DSCP 値をマークダウンするには、ポリシング済み DSCP マップを使用します。

デフォルトのポリシング設定 DSCP マップは、着信 DSCP 値を同じ DSCP 値にマッピングするヌル マップです。

ポリシング済み DSCP マップを変更するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **mls qos map policed-dscp** *dscp-list* **to** *mark-down-dscp*
3. **end**
4. **show mls qos maps policed-dscp**
5. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**mls qos map policed-dscp** *dscp-list* **to** *mark-down-dscp* 例： スイッチ(config)# **mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0**	ポリシング済み DSCP マップを変更します。 • *dscp-list* には、最大 8 つの DSCP 値をスペースで区切って入力します。さらに、**to** キーワードを入力します。 • *mark-down-dscp* には、対応するポリシング設定（マークダウンされた）DSCP値を入力します。 （注） デフォルトのマップに戻すには、**no mls qos policed-dscp** グローバル コンフィギュレーション コマンドを使用します。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**show mls qos maps policed-dscp** 例： スイッチ(config)# **show mls qos maps policed-dscp**	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： スイッチ#	（任意）コンフィギュレーション ファイルに設定を保存します。

DSCP/CoS マップの設定

4 つの出力キューのうち 1 つを選択するために使用される CoS 値を生成するには、DSCP/CoS マップを使用します。

特権 EXEC モードで開始し、次の手順に従って DSCP/CoS マップを修正します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **mls qos map dscp-cos** *dscp-list* **to** *cos*
3. **end**
4. **show mls qos maps dscp-to-cos**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mls qos map dscp-cos** *dscp-list* **to** *cos* 例： スイッチ# **mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0**	DSCP/CoS マップを変更します。 • *dscp-list* には、最大 8 つの DSCP 値をスペースで区切って入力します。さらに、**to** キーワードを入力します。 • *cos* には、DSCP 値と対応する CoS 値を入力します。 DSCP の範囲は 0 ～ 63、CoS の範囲は 0 ～ 7 です。 （注） デフォルトのマップに戻すには、**no mls qos dscp-cos** グローバル コンフィギュレーション コマンドを使用します。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**show mls qos maps dscp-to-cos** 例： スイッチ# **show mls qos maps dscp-to-cos**	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

DSCP/DSCP 変換マップの設定

2 つの QoS ドメインで異なる DSCP 定義が使用されている場合は、一方のドメインの一連の DSCP 値を変換して、もう一方のドメインの定義に一致させる DSCP/DSCP 変換マップを使用

します。DSCP/DSCP 変換マップは、QoS 管理ドメインの境界にある受信ポートに適用します（入力変換）。

入力変換により、パケットの DSCP 値が新しい DSCP 値で上書きされ、QoS はこの新しい値をパケットに適用します。は、新しい DSCP 値とともにそのパケットをポートへ送出します。

1 つの入力ポートに複数の DSCP/DSCP 変換マップを設定できます。デフォルトの DSCP/DSCP 変換マップは、着信 DSCP 値を同じ DSCP 値にマッピングするヌル マップです。

DSCP/DSCP 変換マップを変更するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp*
3. **interface** *interface-id*
4. **mls qos trust dscp**
5. **mls qos dscp-mutation** *dscp-mutation-name*
6. **end**
7. **show mls qos maps dscp-mutation**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp* 例： `スイッチ(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0`	DSCP/DSCP 変換マップを変更します。 • *dscp-mutation-name* には、変換マップ名を入力します。新しい名前を指定することにより、複数のマップを作成できます。 • *in-dscp* には、最大 8 つの DSCP 値をスペースで区切って入力します。さらに、**to** キーワードを入力します。 • *out-dscp* には、1 つの DSCP 値を入力します。 DSCP の範囲は 0 ～ 63 です。 （注） デフォルトのマップに戻すには、**no mls qos dscp-mutation** *dscp-mutation-name* グローバル コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/1`	マップを適用するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。 有効なインターフェイスには、物理ポートが含まれます。
ステップ 4	**mls qos trust dscp** 例： `スイッチ(config-if)# mls qos trust dscp`	DSCP trusted ポートとして入力ポートを設定します。デフォルトでは、ポートは trusted ではありません。
ステップ 5	**mls qos dscp-mutation** *dscp-mutation-name* 例： `スイッチ(config-if)# mls qos dscp-mutation mutation1`	指定された DSCP trusted 入力ポートにマップを適用します。 *dscp-mutation-name* には、ステップ 2 で指定した変換マップ名を入力します。
ステップ 6	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show mls qos maps dscp-mutation** 例： `スイッチ# show mls qos maps dscp-mutation`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy-running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

出力キューの特性の設定

ネットワークおよび QoS ソリューションの複雑さに応じて、次のモジュールで示す作業をすべて実行しなければならない場合があります。次の特性を決定する必要があります。

- DSCP 値または CoS 値によって各キューおよびしきい値 ID にマッピングされるパケット
- キューセット（ポートごとの 4 つの出力キュー）に適用されるドロップしきい値の割合、およびトラフィック タイプに必要なメモリの確保量および最大メモリ

- キュー セットに割り当てる固定バッファ スペースの量
- ポートの帯域幅に関するレート制限の必要性
- 出力キューの処理頻度、および使用する技術（シェーピング、共有、または両方）

設定時の注意事項

緊急キューがイネーブルにされているとき、または SRR の重みに基づいて出力キューのサービスが提供されるときには、次の注意事項に従ってください。

- 出力緊急キューがイネーブルにされている場合は、キュー 1 に対して SRR のシェーピングおよび共有された重みが無効にされます。
- 出力緊急キューがディセーブルにされており、SRR のシェーピングおよび共有された重みが設定されている場合は、キュー 1 に対して shaped モードは shared モードを無効にし、SRR はこのキューに shaped モードでサービスを提供します。
- 出力緊急キューがディセーブルで、SRR シェーピング重みが設定されていない場合、SRR はこのキューを共有モードで処理します。

出力キューセットに対するバッファ スペースの割り当ておよび WTD しきい値の設定

バッファのアベイラビリティの保証、WTD しきい値の設定、およびキューセットの最大メモリ割り当ての設定を行うには、**mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold* グローバル コンフィギュレーション コマンドを使用します。

各しきい値はキューに割り当てられたバッファの割合です。このパーセント値を指定するには、**mls qos queue-set output** *qset-id* **buffers** *allocation1 ... allocation4* グローバル コンフィギュレーション コマンドを使用します。キューは WTD を使用して、トラフィック クラスごとに異なるドロップ割合をサポートします。

（注） スイッチはデフォルトで 4 つの出力キューをサポートしますが、合計 8 つの出力キューを有効にするオプションがあります。8 つの出力キューをすべて有効にするには、**mls qos srr-queue output queues 8** グローバル コンフィギュレーション コマンドを使用します。8 出力キューが有効になると、8 つすべてのキューのしきい値、バッファ、帯域幅の共有重みおよび帯域幅シェーピング重みを設定できます。8 出力キューの設定はスタンドアロン スイッチでのみサポートされます。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。出力キューについて十分理解したうえで、この設定がユーザの QoS ソリューションを満たさないと判断した場合に限り、設定を変更してください。

キューセットのメモリ割り当てとドロップしきい値を設定するには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **mls qos srr-queue output queues 8**
3. **mls qos queue-set output** *qset id* **buffers** *allocation1...allocation8*
4. **mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold*
5. **interface** *interface-id*
6. **queue-set** *qset-id*
7. **end**
8. **show mls qos interface** [*interface-id*] **buffers**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mls qos srr-queue output queues 8** 例： `スイッチ(config)# mls qos srr-queue output queues 8`	（任意）スイッチはデフォルトで 4 つの出力キューをサポートしますが、合計 8 つの出力キューを有効にすることができます。4 つの追加出力キューを有効にするには、オプションの **mls qos srr-queue output queues 8** コマンドを使用します。 8 つのキュー サポートが有効になると、4 つの追加キューの設定に進むことができます。追加のキューパラメータをサポートするように、既存の出力キュー設定コマンドが変更されます。 （注） 8 つのキューを有効にするオプションは、スタンドアロン スイッチのみで使用できます。
ステップ 3	**mls qos queue-set output** *qset id* **buffers** *allocation1...allocation8* 例： `スイッチ(config)# mls qos queue-set output 2 buffers 40 20 20 20 10 10 10 10`	バッファをキューセットに割り当てます。 デフォルトでは、すべての割り当て値は4つのキューに均等にマッピングされます（25、25、25、25）。各キューがバッファ スペースの 1/4 を持ちます。8 つの出力キューを設定すると、デフォルトで、合計バッファ スペースの 30 % がキュー 2 に割り当てら

	コマンドまたはアクション	目的
		れ、キュー1、3、4、5、6、7、および8にそれぞれ10 % が割り当てられます。 上記のステップ 2 で説明したように、8 つの出力キューを有効にした場合は次が適用されます。 • *qset-id* には、キューセットの ID を入力します。指定できる範囲は1～2です。各ポートはキューセットに属し、ポート単位で出力キュー 4 つの特性すべてを定義します。 • *allocation1 ... allocation8* には、キューセット内のキューごとに 1 つずつ、合計 8 つのパーセンテージを指定します。*allocation1*、*allocation3*、および *allocation4* ~ *allocation8* の範囲は 0 ~ 99 です。*allocation2* の範囲は 1 ～ 100 です（CPU バッファを含める）。 トラフィックの重要度に応じてバッファを割り当てます。たとえば、最高プライオリティのトラフィックを持つキューには多くの割合のバッファを与えます。 （注） デフォルトの設定に戻すには、**no mls qos queue-set output** *qset-id* **buffers** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold* 例： `スイッチ(config)# mls qos queue-set output 2 threshold 2 40 60 100 200`	WTD しきい値を設定し、バッファのアベイラビリティを保証し、キューセット（ポートごとに 4 つの出力キュー）の最大メモリ割り当てを設定します。 デフォルトでは、キュー 1、3、および 4 の WTD は 100% に設定されています。キュー2 の WTD は 200% に設定されています。キュー 1、2、3、および 4 の専用は 50% に設定されています。すべてのキューの最大しきい値はデフォルトで 400% に設定されています。 上記のステップ 2 で説明したように、8 つの出力キューを有効にした場合は次が適用されます。 • *qset-id* には、ステップ 2 で指定したキューセットの ID を入力します。指定できる範囲は 1 ～ 2 です。 • *queue-id* には、コマンドの実行対象となるキューセット内の特定のキューを入力します。queue-id

	コマンドまたはアクション	目的
		の範囲は、デフォルトでは1～4、8つのキューが有効になっている場合は 1 ～ 8 です。 • *drop-threshold1 drop-threshold2* には、キューの割り当てメモリのパーセンテージとして表される 2 つの WTD しきい値を指定します。指定できる範囲は 1 ～ 3200% です。 • *reserved-threshold* には、割り当てメモリのパーセンテージとして表されるキューに保証（確保）されるメモリ サイズを入力します。指定できる範囲は 1 ～ 100% です。 • *maximum-threshold* を指定すると、いっぱいになったキューが確保量を超えるバッファを取得できるようになります。この値は、共通プールが空でない場合に、パケットがドロップされるまでキューが使用できるメモリの最大値です。指定できる範囲は 1 ～ 3200% です。 （注） デフォルトの WTD しきい値の割合に戻すには、**no mls qos queue-set output** *qset-id* **threshold** [*queue-id*] グローバル コンフィギュレーション コマンドを使用します。
ステップ 5	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/1`	発信トラフィックのポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	**queue-set** *qset-id* 例： `スイッチ(config-id)# queue-set 2`	キューセットにポートをマッピングします。 *qset-id* には、ステップ 2 で指定したキューセットの ID を入力します。指定できる範囲は 1 ～ 2 です。デフォルトは 1 です。
ステップ 7	**end** 例： `スイッチ(config-id)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show mls qos interface** [*interface-id*] **buffers** 例：	入力を確認します。

	コマンドまたはアクション	目的
	スイッチ# **show mls qos interface buffers**	
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。 デフォルトの設定に戻すには、**no mls qos queue-set output***qset-id***buffers** グローバル コンフィギュレーション コマンドを使用します。デフォルトの WTD しきい値の割合に戻すには、**no mls qos queue-set output** *qset-id* **threshold** [[*queue-id*]] グローバル コンフィギュレーション コマンドを使用します。

出力キューおよびしきい値 ID への DSCP または CoS 値のマッピング

トラフィックに優先度を設定するには、特定の DSCP または CoS を持つパケットを特定のキューに格納し、より低い優先度を持つパケットがドロップされるようにキューのしきい値を調整します。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。デフォルト設定の変更が必要となるのは、出力キューについて完全に理解している場合、およびデフォルトの設定がご使用の QoS ソリューションを満たしていない場合だけです。

DSCP または CoS 値を出力キューおよび ID にマッピングするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. 次のいずれかを使用します。
   - **mls qos srr-queue output dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8*
   - **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8*
3. **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8*
4. **end**
5. **show mls qos maps**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	次のいずれかを使用します。 • **mls qos srr-queue output dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8* • **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* 例： スイッチ(config)# **mls qos srr-queue output dscp-map queue 1 threshold 2 10 11**	DSCP または CoS 値を出力キューおよびしきい値 ID にマッピングします。 デフォルトでは、DSCP 値 0 ～ 15 はキュー 2 およびしきい値 1 に、DSCP 値 16 ～ 31 はキュー 3 およびしきい値 1 に、DSCP 値 32 ～ 39 および 48 ～ 63 はキュー 4 およびしきい値 1 に、DSCP 値 40 ～ 47 はキュー 1 およびしきい値 1 にマッピングされます。 デフォルトでは、CoS 値 0 および 1 はキュー 2 およびしきい値 1 に、CoS 値 2 および 3 はキュー 3 およびしきい値 1 に、CoS 値 4、6、および 7 はキュー 4 およびしきい値 1 に、CoS 値 5 はキュー 1 およびしきい値 1 にマッピングされます。 • *queue-id* で指定できる範囲は 1 ～ 4 です。 （注） **mls qos srr-queue output queues 8** グローバル コンフィギュレーション コマンドを使用して 8 つの出力キューを有効にした場合、*queue-id* の範囲は 1 ～ 8 になります。 • *threshold-id* で指定できる範囲は 1 ～ 3 です。しきい値 3 のドロップしきい値（%）は事前に定義されています。パーセンテージはキューがいっぱいの状態に対して設定されます。 • *dscp1...dscp8* には、各値をスペースで区切って、最大 8 の値を入力します。指定できる範囲は 0 ～ 63 です。 • *cos1...cos8* には、各値をスペースで区切って、最大 8 の値を入力します。指定できる範囲は 0 ～ 7 です。 （注） デフォルトの DSCP 出力キューしきい値マップまたはデフォルトの CoS 出力キューしきい値マップに戻すには、**no mls qos srr-queue output dscp-map** または **no mls qos srr-queue output cos-map** グローバル コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 3	**mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* 例： `スイッチ(config)# mls qos srr-queue output cos-map queue 3 threshold 1 2 3`	CoS 値を出力キューおよびしきい値 ID にマッピングします。 デフォルトでは、CoS 値 0 および 1 はキュー 2 およびしきい値 1 に、CoS 値 2 および 3 はキュー 3 およびしきい値 1 に、CoS 値 4、6、および 7 はキュー 4 およびしきい値 1 に、CoS 値 5 はキュー 1 およびしきい値 1 にマッピングされます。 • *queue-id* で指定できる範囲は 1 ～ 4 です。 • *threshold-id* で指定できる範囲は 1 ～ 3 です。しきい値 3 のドロップしきい値（%）は事前に定義されています。パーセンテージはキューがいっぱいの状態に対して設定されます。 • *cos1...cos8* には、各値をスペースで区切って、最大 8 の値を入力します。指定できる範囲は 0 ～ 7 です。 （注） デフォルトの CoS 出力キューしきい値マップを返すには、**no mls qos srr-queue output cos-map** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos maps** 例： `スイッチ# show mls qos maps`	入力を確認します。 DSCP 出力キューしきい値マップは、表形式で表示されます。d1 列は DSCP 値の最上位桁、d2 行は DSCP 値の最下位桁を示します。d1 および d2 値の交点がキュー ID およびしきい値 ID です。たとえば、キュー 2 およびしきい値 1（02-01）のようになります。 CoS 出力キューしきい値マップでは、先頭行に CoS 値、2 番めの行に対応するキュー ID およびしきい値 ID が示されます。たとえば、キュー 2 およびしきい値 2（2-2）のようになります。
ステップ 6	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy-running-config startup-config**	デフォルトの DSCP 出力キューしきい値マップまたはデフォルトの CoS 出力キューしきい値マップに戻すには、**no mls qos srr-queue output dscp-map** または **no mls qos srr-queue output cos-map** グローバル コンフィギュレーション コマンドを使用します。

出力キューでの SRR シェーピング重みの設定

各キューに割り当てられる使用可能な帯域幅の量を指定できます。重みの比率は、SRR スケジューラが各キューからパケットを送信する頻度の比率です。

出力キューにシェーピング重み、共有重み、またはその両方を設定できます。バースト性のあるトラフィックをスムーズにする、または長期にわたって出力をスムーズにする場合に、シェーピングを使用します。

ポートにマッピングされた4つの出力キューにシェーピング重みを割り当てて、帯域幅のシェーピングをイネーブルにするには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4*
4. **end**
5. **show mls qos interface** *interface-id* **queueing**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet2/0/1**	発信トラフィックのポートを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* 例： スイッチ(config-if)# **srr-queue bandwidth shape 8 0 0 0**	出力キューにSRR重みを割り当てます。デフォルトでは、weight1 は 25、weight2、weight3、および weight4 は 0 に設定されています。これらのキューは共有モードです。 *weight1 weight2 weight3 weight4* には、シェーピングされるポートの割合を制御する重みを入力します。このキューのシェーピング帯域幅は、インバース比率（1/weight）によって制御されます。各値はスペースで区切ります。指定できる範囲は 0 ～ 65535 です。 重み 0 を設定した場合は、対応するキューが共有モードで動作します。**srr-queue bandwidth shape** コマンドで指定された重みは無視され、**srr-queue bandwidth share** インターフェイス コンフィギュレーション コマンドで設定されたキューの重みが有効になります。シェーピングおよび共有の両方に対して同じキューセットのキューを設定した場合は、必ず番号が最も小さいキューにシェーピングを設定してください。 シェーピング モードは、共有モードを無効にします。 デフォルトの設定に戻す場合は、 **no srr-queue bandwidth shape** インターフェイス コンフィギュレーション コマンドを使用します。 （注） **mls qos srr-queue output queues 8** グローバル コンフィギュレーション コマンドを使用して 8 個の出力キューを有効にした場合、合計 8 個のキューに SRR 重みを割り当てることができます。
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos interface** *interface-id* **queueing** 例： スイッチ# **show mls qos interface interface-id queuing**	入力を確認します。

	コマンドまたはアクション	目的
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。 デフォルトの設定に戻す場合は、**no srr-queue bandwidth shape** インターフェイス コンフィギュレーション コマンドを使用します。

出力キューでの SRR 共有重みの設定

共有モードでは、設定された重みによりキュー間で帯域幅が共有されます。このレベルでは帯域幅は保証されていますが、このレベルに限定されていません。たとえば、特定のキューが空であり、リンクを共有する必要がない場合、残りのキューは未使用の帯域幅を使用して、共有ができます。共有の場合、キューからパケットを取り出す頻度は重みの比率によって制御されます。重みの絶対値は関係ありません。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。出力キューについて十分理解したうえで、この設定がユーザの QoS ソリューションを満たさないと判断した場合に限り、設定を変更してください。

ポートにマッピングされた 4 つの出力キューに共有重みを割り当てて、帯域幅の共有をイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth share** *weight1 weight2 weight3 weight4*
4. **end**
5. **show mls qos interface** *interface-id* **queueing**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例：	発信トラフィックのポートを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ(config)# **interface gigabitethernet2/0/1**	
ステップ 3	**srr-queue bandwidth share** *weight1 weight2 weight3 weight4* 例： スイッチ(config-id)# **srr-queue bandwidth share 1 2 3 4**	出力キューにSRR重みを割り当てます。デフォルトでは、4つの重みがすべて25です（各キューに帯域幅の 1/4 が割り当てられています）。 *weight1 weight2 weight3 weight4* には、SRR スケジューラがパケットを送信する頻度の比率を制御する重みを入力します。各値はスペースで区切ります。指定できる範囲は 1 ～ 255 です。 デフォルトの設定に戻す場合は、 **no srr-queue bandwidth share** インターフェイス コンフィギュレーション コマンドを使用します。 （注） **mls qos srr-queue output queues 8** グローバル コンフィギュレーション コマンドを使用して 8 個の出力キューを有効にした場合、合計 8 個のキューに SRR 重みを割り当てることができます。
ステップ 4	**end** 例： スイッチ(config-id)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos interface** *interface-id* **queueing** 例： スイッチ# **show mls qos interface interface_id queuing**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy-running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。 デフォルトの設定に戻す場合は、**no srr-queue bandwidth share** インターフェイス コンフィギュレーション コマンドを使用します。

出力緊急キューの設定

出力緊急キューにパケットを入れることにより、特定のパケットのプライオリティを他のすべてのパケットより高く設定できます。SRR は、このキューが空になるまで処理してから他のキューを処理します。

出力緊急キューをイネーブルにするには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **mls qos**
3. **interface** *interface-id*
4. **priority-queue out**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mls qos** 例： `スイッチ(config)# mls qos`	スイッチの QoS をイネーブルにします。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/1`	出力ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**priority-queue out** 例： `スイッチ(config-if)# priority-queue out`	デフォルトでディセーブルに設定されている出力緊急キューをイネーブルにします。 このコマンドを設定すると、SRR に参加するキューは 1 つ少なくなるため、SRR 重みおよびキュー サイズの比率が影響を受けます。これは、**srr-queue bandwidth shape** または **srr-queue bandwidth share**

	コマンドまたはアクション	目的
		コマンド内の *weight1* が無視される（比率計算に使用されない）ことを意味します。 （注） 出力緊急キューをディセーブルにするには、**no priority-queue out** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。 出力緊急キューをディセーブルにするには、**no priority-queue out** インターフェイス コンフィギュレーション コマンドを使用します。

出力インターフェイスの帯域幅の制限

出力ポートの帯域幅は制限できます。たとえば、カスタマーが高速リンクの一部しか費用を負担しない場合は、帯域幅をその量に制限できます。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。出力キューについて十分理解したうえで、この設定がユーザの QoS ソリューションを満たさないと判断した場合に限り、設定を変更してください。

出力ポートの帯域幅を制限するには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth limit** *weight1*

4. **end**
5. **show mls qos interface** [*interface-id*] **queueing**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/1`	レートを制限するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**srr-queue bandwidth limit** *weight1* 例： `スイッチ(config-if)# srr-queue bandwidth limit 80`	ポートの上限となるポート速度の割合を指定します。指定できる範囲は 10 ～ 90 です。 デフォルトでは、ポートのレートは制限されず、100% に設定されています。 （注） デフォルトの設定に戻す場合は、**no srr-queue bandwidth limit** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos interface** [*interface-id*] **queueing** 例： `スイッチ# show mls qos interface interface_id queueing`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy-running-config startup-config**	デフォルトの設定に戻す場合は、 **no srr-queue bandwidth limit** インターフェイス コンフィギュレーション コマンドを使用します。

標準 QoS のモニタリング

表 *95:* スイッチ上で標準 *QoS* をモニタリングするためのコマンド

コマンド	説明
show mls qos	グローバル QoS コンフィギュレーション情報を表示します。
show mls qos aggregate-policer [*aggregate-policer-name*]	集約ポリサーの設定を表示します。
show mls qos interface [*interface-id*] [**buffers** | **policers** | **queueing** | **statistics**]	バッファ割り当て、ポリサーが設定されているポート、キューイング方式、入出力統計情報など、ポート レベルの QoS 情報が表示されます。
show mls qos maps [**cos-dscp** | |**cos-output-q** | **dscp-cos** | |**dscp-mutation** *dscp-mutation-name* | **dscp-output-q** | **ip-prec-dscp** | **policed-dscp**]	QoS のマッピング情報を表示します。
show mls qos queue-set [*qset-id*]	出力キューの QoS 設定を表示します。
show running-config | include rewrite	DSCP 透過性設定を表示します。

QoS の設定例

例：DSCP 信頼状態へのポートの設定および DSCP/DSCP 変換マップの変更

次に、ポートが DSCP を信頼する状態に設定し、着信した DSCP 値 10 ～ 13 が DSCP 値 30 にマッピングされるように DSCP/DSCP 変換マップ（*gi1/0/2-mutation*）を変更する例を示します。

```
スイッチ(config)# mls qos map dscp-mutation gigabitethernet1/0/2-mutation
10 11 12 13 to 30
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# mls qos trust dscp
スイッチ(config-if)# mls qos dscp-mutation gigabitethernet1/0/2-mutation
```

```
スイッチ(config-if)# end
```

例：ACL によるトラフィックの分類

次に、指定された 3 つのネットワーク上のホストだけにアクセスを許可する例を示します。ネットワーク アドレスのホスト部分にワイルドカード ビットが適用されます。アクセス リストのステートメントと一致しない送信元アドレスのホストはすべて拒否されます。

```
スイッチ(config)# access-list 1 permit 192.5.255.0 0.0.0.255
スイッチ(config)# access-list 1 permit 128.88.0.0 0.0.255.255
スイッチ(config)# access-list 1 permit 36.0.0.0 0.0.0.255
! (Note: all other access implicitly denied)
```

次に、任意の送信元から、DSCP 値が 32 に設定されている任意の宛先への IP トラフィックを許可する ACL を作成する例を示します。

```
スイッチ(config)# access-list 100 permit ip any any dscp 32
```

次に、10.1.1.1 の送信元ホストから 10.1.1.2 の宛先ホストへの IP トラフィック（precedence 値は 5）を許可する ACL を作成する例を示します。

```
スイッチ(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

次に、任意の送信元からアドレス 224.0.0.2 の宛先グループへの PIM トラフィック（DSCP 値は 32）を許可する ACL を作成する例を示します。

```
スイッチ(config)# access-list 102 permit pim any 224.0.0.2 dscp 32
```

次に、任意の送信元から、DSCP 値が 32 に設定されている任意の宛先への IPv6 トラフィックを許可する ACL を作成する例を示します。

```
スイッチ(config)# ipv6 access-list 100 permit ip any any dscp 32
```

次に、10.1.1.1 の送信元ホストから 10.1.1.2 の宛先ホストへの IPv6 トラフィック（precedence 値は 5）を許可する ACL を作成する例を示します。

```
スイッチ(config)# ipv6 access-list ipv6_Name_ACL permit ip host 10::1 host 10.1.1.2
precedence 5
```

次に、2 つの許可（permit）ステートメントを指定したレイヤ 2 の MAC ACL を作成する例を示します。最初のステートメントでは、MAC アドレスが 0001.0000.0001 であるホストから、MAC アドレスが 0002.0000.0001 であるホストへのトラフィックが許可されます。2 番めのステートメントでは、MAC アドレスが 0001.0000.0002 であるホストから、MAC アドレスが 0002.0000.0002 であるホストへの、Ethertype が XNS-IDP のトラフィックのみが許可されます。

```
スイッチ(config)# mac access-list extended maclist1
スイッチ(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
スイッチ(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
! (Note: all other access implicitly denied)
```

例：クラス マップによるトラフィックの分類

次に、*class1* というクラス マップの設定例を示します。*class1* にはアクセス リスト 103 という一致条件が 1 つ設定されています。このクラスマップによって、任意のホストから任意の宛先へのトラフィック（DSCP 値は 10）が許可されます。

```
スイッチ(config)# access-list 103 permit ip any any dscp 10
スイッチ(config)# class-map class1
スイッチ(config-cmap)# match access-group 103
スイッチ(config-cmap)# end
スイッチ#
```

次に、DSCP 値が 10、11、および 12 である着信トラフィックと照合する、*class2* という名前のクラス マップを作成する例を示します。

```
スイッチ(config)# class-map class2
スイッチ(config-cmap)# match ip dscp 10 11 12
スイッチ(config-cmap)# end
スイッチ#
```

次に、IP precedence 値が 5、6、および 7 である着信トラフィックと照合する、*class3* という名前のクラス マップを作成する例を示します。

```
スイッチ(config)# class-map class3
スイッチ(config-cmap)# match ip precedence 5 6 7
スイッチ(config-cmap)# end
スイッチ#
```

次に、IP DSCP および IPv6 と照合するクラス マップを設定する例を示します。

```
スイッチ(config)# Class-map cm-1
スイッチ(config-cmap)# match ip dscp 10
スイッチ(config-cmap)# match protocol ipv6
スイッチ(config-cmap)# exit
スイッチ(config)# Class-map cm-2
スイッチ(config-cmap)# match ip dscp 20
スイッチ(config-cmap)# match protocol ip
スイッチ(config-cmap)# exit
スイッチ(config)# Policy-map pm1
スイッチ(config-pmap)# class cm-1
スイッチ(config-pmap-c)# set dscp 4
スイッチ(config-pmap-c)# exit
```

```
スイッチ(config-pmap)# class cm-2
スイッチ(config-pmap-c)# set dscp 6
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
スイッチ(config)# interface G1/0/1
スイッチ(config-if)# service-policy input pm1
```

次に、IPv4 トラフィックと IPv6 トラフィックの両方に適用するクラス マップを設定する例を示します。

```
スイッチ(config)# ip access-list 101 permit ip any any
スイッチ(config)# ipv6 access-list ipv6-any permit ip any any
スイッチ(config)# Class-map cm-1
スイッチ(config-cmap)# match access-group 101
スイッチ(config-cmap)# exit
スイッチ(config)# class-map cm-2
スイッチ(config-cmap)# match access-group name ipv6-any
スイッチ(config-cmap)# exit
スイッチ(config)# Policy-map pm1
スイッチ(config-pmap)# class cm-1
スイッチ(config-pmap-c)# set dscp 4
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class cm-2
スイッチ(config-pmap-c)# set dscp 6
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
スイッチ(config)# interface G0/1
スイッチ(config-if)# switch mode access
スイッチ(config-if)# service-policy input pm1
```

例：ポリシーマップを使用した物理ポートのトラフィックの分類、ポリシング、およびマーキング

次に、ポリシー マップを作成し、入力ポートに結合する例を示します。この設定では、IP 標準ACLでネットワーク 10.1.0.0からのトラフィックを許可します。この分類にトラフィックが一致した場合、着信パケットの DSCP 値が信頼されます。一致したトラフィックが平均トラフィック レート（48000 bps）、および標準バースト サイズ（8000 バイト）を超過している場合は、（ポリシング済み DSCP マップに基づいて）DSCP はマークダウンされて、送信されます。

```
スイッチ(config)# access-list 1 permit 10.1.0.0 0.0.255.255
スイッチ(config)# class-map ipclass1
スイッチ(config-cmap)# match access-group 1
スイッチ(config-cmap)# exit
スイッチ(config)# policy-map flow1t
スイッチ(config-pmap)# class ipclass1
スイッチ(config-pmap-c)# trust dscp
スイッチ(config-pmap-c)# police 1000000 8000 exceed-action policed-dscp-transmit
```

```
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
スイッチ(config)# interface gigabitethernet2/0/1
スイッチ(config-if)# service-policy input flow1t
```

次に、2 つの許可ステートメントを指定してレイヤ 2 MAC ACL を作成し、入力ポートに結合する例を示します。最初の許可ステートメントでは、MAC アドレスが 0001.0000.0001 であるホストから、MACアドレスが0002.0000.0001であるホストへのトラフィックが許可されます。2 番めの許可ステートメントでは、MAC アドレスが 0001.0000.0002 であるホストから、MAC アドレスが 0002.0000.0002 であるホストへの、Ethertype が XNS-IDP のトラフィックのみが許可されます。

```
スイッチ(config)# mac access-list extended maclist1
スイッチ(config-ext-mac)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
スイッチ(config-ext-mac)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
スイッチ(config-ext-mac)# exit
スイッチ(config)# mac access-list extended maclist2
スイッチ(config-ext-mac)# permit 0001.0000.0003 0.0.0 0002.0000.0003 0.0.0
スイッチ(config-ext-mac)# permit 0001.0000.0004 0.0.0 0002.0000.0004 0.0.0 aarp
スイッチ(config-ext-mac)# exit
スイッチ(config)# class-map macclass1
スイッチ(config-cmap)# match access-group maclist1
スイッチ(config-cmap)# exit
スイッチ(config)# policy-map macpolicy1
スイッチ(config-pmap)# class macclass1
スイッチ(config-pmap-c)# set dscp 63
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class macclass2 maclist2
スイッチ(config-pmap-c)# set dscp 45
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# mls qos trust cos
スイッチ(config-if)# service-policy input macpolicy1
```

次に、分類されていないトラフィックに適用されるデフォルト クラスを使用して、IPv4 と IPv6 の両方のトラフィックに適用されるクラス マップを作成する例を示します。

```
スイッチ(config)# ip access-list 101 permit ip any any
スイッチ(config)# ipv6 access-list ipv6-any permit ip any any
スイッチ(config)# class-map cm-1
スイッチ(config-cmap)# match access-group 101
スイッチ(config-cmap)# exit
スイッチ(config)# class-map cm-2
スイッチ(config-cmap)# match access-group name ipv6-any
スイッチ(config-cmap)# exit
スイッチ(config)# policy-map pm1
スイッチ(config-pmap)# class cm-1
スイッチ(config-pmap-c)# set dscp 4
スイッチ(config-pmap-c)# exit
```

```
スイッチ(config-pmap)# class cm-2
スイッチ(config-pmap-c)# set dscp 6
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class class-default
スイッチ(config-pmap-c)# set dscp 10
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
スイッチ(config)# interface G0/1
スイッチ(config-if)# switch mode access
スイッチ(config-if)# service-policy input pm1
```

例：階層型ポリシー マップによる SVI のトラフィックの分類、ポリシング、およびマーキング

次に、階層型のポリシー マップの作成方法を示します。

```
Switch> enable
スイッチ#  configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
スイッチ(config)# access-list 101 permit ip any any
スイッチ(config)# class-map cm-1
スイッチ(config-cmap)# match access 101
スイッチ(config-cmap)# exit
スイッチ(config)# exit
スイッチ#
スイッチ#
```

次に、SVI に新しいマップを割り当てる例を示します。

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# class-map cm-interface-1
スイッチ(config-cmap)# match input gigabitethernet3/0/1 - gigabitethernet3/0/2
スイッチ(config-cmap)# exit
スイッチ(config)# policy-map port-plcmap
スイッチ(config-pmap)# class cm-interface-1
スイッチ(config-pmap-c)# police 900000 9000 exc policed-dscp-transmit
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
スイッチ(config)# policy-map vlan-plcmap
スイッチ(config-pmap)# class cm-1
スイッチ(config-pmap-c)# set dscp 7
スイッチ(config-pmap-c)# service-policy port-plcmap-1
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class cm-2
スイッチ(config-pmap-c)# service-policy port-plcmap-1
スイッチ(config-pmap-c)# set dscp 10
スイッチ(config-pmap)# exit
スイッチ(config-pmap)# class cm-3
```

```
スイッチ(config-pmap-c)# service-policy port-plcmap-2
スイッチ(config-pmap-c)# set dscp 20
スイッチ(config-pmap)# exit
スイッチ(config-pmap)# class cm-4
スイッチ(config-pmap-c)# trust dscp
スイッチ(config-pmap)# exit
スイッチ(config)# interface vlan 10
スイッチ(config-if)# service-policy input vlan-plcmap
スイッチ(config-if)# exit
スイッチ(config)# exit
スイッチ#
```

次の例では、子レベルのポリシーマップがクラス下に添付されるタイミング、そのクラスのアクションが指定される必要があるタイミングを示します。

```
スイッチ(config)# policy-map vlan-plcmap
スイッチ(config-pmap)# class cm-5
スイッチ(config-pmap-c)# set dscp 7
スイッチ(config-pmap-c)# service-policy port-plcmap-1
```

次に、IP DSCP および IPv6 と照合するクラス マップを設定する例を示します。

```
スイッチ(config)# class-map cm-1
スイッチ(config-cmap)# match ip dscp 10
スイッチ(config-cmap)# match protocol ipv6
スイッチ(config-cmap)# exit
スイッチ(config)# class-map cm-2
スイッチ(config-cmap)# match ip dscp 20
スイッチ(config-cmap)# match protocol ip
スイッチ(config-cmap)# exit
スイッチ(config)# policy-map pm1
スイッチ(config-pmap)# class cm-1
スイッチ(config-pmap-c)# set dscp 4
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class cm-2
スイッチ(config-pmap-c)# set dscp 6
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
スイッチ(config)# interface G1/0/1
スイッチ(config-if)# service-policy input pm1
```

次に、デフォルト トラフィック クラスをポリシー マップに設定する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# class-map cm-3
スイッチ(config-cmap)# match ip dscp 30
スイッチ(config-cmap)# match protocol ipv6
スイッチ(config-cmap)# exit
スイッチ(config)# class-map cm-4
スイッチ(config-cmap)# match ip dscp 40
```

```
スイッチ(config-cmap)# match protocol ip
スイッチ(config-cmap)# exit
スイッチ(config)# policy-map pm3
スイッチ(config-pmap)# class class-default
スイッチ(config-pmap)# set dscp 10
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class cm-3
スイッチ(config-pmap-c) set dscp 4
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class cm-4
スイッチ(config-pmap-c)# trust cos
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
```

次に、class-default が最初に設定されていても、ポリシーマップ pm3 の最後にデフォルト トラフィック クラスが自動的に配置される例を示します。

```
スイッチ# show policy-map pm3
  Policy Map pm3
    Class cm-3
      set dscp 4
    Class cm-4
      trust cos
    Class class-default
      police 8000 80000 exceed-action drop
スイッチ#
```

例：集約ポリサーによるトラフィックの分類、ポリシング、およびマーキング

次に、集約ポリサーを作成して、ポリシーマップ内の複数のクラスに結合する例を示します。この設定では、IP ACL はネットワーク 10.1.0.0 およびホスト 11.3.1.1 からのトラフィックを許可します。ネットワーク 10.1.0.0 から着信するトラフィックの場合は、着信パケットの DSCP が信頼されます。ホスト 11.3.1.1 から着信するトラフィックの場合、パケットの DSCP は 56 に変更されます。ネットワーク 10.1.0.0 およびホスト 11.3.1.1 からのトラフィック レートには、ポリシングが設定されます。トラフィックが平均レート（48000 bps）、および標準バースト サイズ（8000 バイト）を超過している場合は、（ポリシング済み DSCP マップに基づいて）DSCP がマークダウンされて、送信されます。ポリシーマップは入力ポートに結合されます。

```
スイッチ(config)# access-list 1 permit 10.1.0.0 0.0.255.255
スイッチ(config)# access-list 2 permit 11.3.1.1
スイッチ(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action
policed-dscp-transmit
スイッチ(config)# class-map ipclass1
スイッチ(config-cmap)# match access-group 1
スイッチ(config-cmap)# exit
スイッチ(config)# class-map ipclass2
スイッチ(config-cmap)# match access-group 2
スイッチ(config-cmap)# exit
```

```
スイッチ(config)# policy-map aggflow1
スイッチ(config-pmap)# class ipclass1
スイッチ(config-pmap-c)# trust dscp
スイッチ(config-pmap-c)# police aggregate transmit1
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class ipclass2
スイッチ(config-pmap-c)# set dscp 56
スイッチ(config-pmap-c)# police aggregate transmit1
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# class class-default
スイッチ(config-pmap-c)# set dscp 10
スイッチ(config-pmap-c)# exit
スイッチ(config-pmap)# exit
スイッチ(config)# interface gigabitethernet2/0/1
スイッチ(config-if)# service-policy input aggflow1
スイッチ(config-if)# exit
```

例：DSCP マップの設定

次に、CoS/DSCP マップを変更して表示する例を示します。

```
スイッチ(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45
スイッチ(config)# end
スイッチ# show mls qos maps cos-dscp

Cos-dscp map:
        cos:   0  1  2  3  4  5  6  7
     --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

次に、IP precedence/DSCP マップを変更して表示する例を示します。

```
スイッチ(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45
スイッチ(config)# end
スイッチ# show mls qos maps ip-prec-dscp

IpPrecedence-dscp map:
     ipprec:   0  1  2  3  4  5  6  7
     --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

次に、DSCP 50 ～ 57 を、マークダウンされる DSCP 値 0 にマッピングする例を示します。

```
スイッチ(config)# mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0
スイッチ(config)# end
スイッチ# show mls qos maps policed-dscp
Policed-dscp map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 01 02 03 04 05 06 07 08 09
      1 :    10 11 12 13 14 15 16 17 18 19
      2 :    20 21 22 23 24 25 26 27 28 29
```

```
 3 :    30 31 32 33 34 35 36 37 38 39
 4 :    40 41 42 43 44 45 46 47 48 49
 5 :    00 00 00 00 00 00 00 00 58 59
 6 :    60 61 62 63
```

（注） このポリシング済み DSCP マップでは、マークダウンされる DSCP 値が表形式で示されています。d1 列は元の DSCP の最上位桁、d2 行は元の DSCP の最下位桁を示します。d1 と d2 の交点にある値が、マークダウンされる値です。たとえば、元の DSCP 値が 53 の場合、マークダウンされる DSCP 値は 0 です。

次に、DSCP 値 0、8、16、24、32、40、48、および 50 を CoS 値 0 にマッピングして、マップを表示する例を示します。

```
スイッチ(config)# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0
スイッチ(config)# end
スイッチ# show mls qos maps dscp-cos
Dscp-cos map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 00 01
      1 :    01 01 01 01 01 01 00 02 02 02
      2 :    02 02 02 02 00 03 03 03 03 03
      3 :    03 03 00 04 04 04 04 04 04 04
      4 :    00 05 05 05 05 05 05 05 00 06
      5 :    00 06 06 06 06 06 07 07 07 07
      6 :    07 07 07 07
```

（注） 上記の DSCP/CoS マップでは、CoS 値が表形式で示されています。d1 列は DSCP の最上位桁、d2 行は DSCP の最下位桁を示します。d1 と d2 の交点にある値が CoS 値です。たとえば、この DSCP/CoS マップでは、DSCP 値が 08 の場合、対応する CoS 値は 0 です。

次の例では、DSCP/DSCP 変換マップを定義する方法を示します。明示的に設定されていないすべてのエントリは変更されません（空のマップで指定された値のままです）。

```
スイッチ(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0
スイッチ(config)# mls qos map dscp-mutation mutation1 8 9 10 11 12 13 to 10
スイッチ(config)# mls qos map dscp-mutation mutation1 20 21 22 to 20
スイッチ(config)# mls qos map dscp-mutation mutation1 30 31 32 33 34 to 30
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# mls qos trust dscp
スイッチ(config-if)# mls qos dscp-mutation mutation1
スイッチ(config-if)# end
スイッチ# show mls qos maps dscp-mutation mutation1
Dscp-dscp mutation map:
   mutation1:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 10 10
```

```
1 :    10 10 10 10 14 15 16 17 18 19
2 :    20 20 20 23 24 25 26 27 28 29
3 :    30 30 30 30 30 35 36 37 38 39
4 :    40 41 42 43 44 45 46 47 48 49
5 :    50 51 52 53 54 55 56 57 58 59
6 :    60 61 62 63
```

（注） 上記の DSCP/DSCP 変換マップでは、変換される値が表形式で示されています。d1 列は元の DSCP の最上位桁、d2 行は元の DSCP の最下位桁を示します。d1 と d2 の交点の値が、変換される値です。たとえば、DSCP 値が 12 の場合、対応する変換される値は 10 です。

例：出力キューの特性の設定

次に、DSCP 値 10 および 11 を出力キュー 1 およびしきい値 2 にマッピングする例を示します。

```
スイッチ(config)# mls qos srr-queue output dscp-map queue 1 threshold 2 10 11
```

次に、キュー 1 に帯域幅のシェーピングを設定する例を示します。キュー 2、3、4 の重み比が 0 に設定されているので、これらのキューは共有モードで動作します。キュー 1 の帯域幅の重みは 1/8（12.5%）です。

```
スイッチ(config)# interface gigabitethernet2/0/1
スイッチ(config-if)# srr-queue bandwidth shape 8 0 0 0
```

次の例では、出力ポートで稼働する SRR スケジューラの重み比を設定する方法を示します。4 つのキューが使用され、共有モードで各キューに割り当てられる帯域幅の比率は、キュー 1、2、3、および 4 に対して 1/(1 + 2 + 3 + 4)、2/(1 + 2 + 3 + 4)、3/(1 + 2 + 3 + 4)、および 4/(1 + 2 + 3 + 4) になります（それぞれ、10、20、30、および 40%）。キュー 4 はキュー 1 の帯域幅の 4 倍、キュー 2 の帯域幅の 2 倍、キュー 3 の帯域幅の 1 と 1/3 倍であることを示します。

```
スイッチ(config)# interface gigabitethernet2/0/1
スイッチ(config-if)# srr-queue bandwidth share 1 2 3 4
```

次の例では、SRR の重みが設定されている場合、出力緊急キューをイネーブルにする方法を示します。出力緊急キューは、設定された SRR ウェイトを上書きします。

```
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# srr-queue bandwidth shape 25 0 0 0
スイッチ(config-if)# srr-queue bandwidth share 30 20 25 25
スイッチ(config-if)# priority-queue out
スイッチ(config-if)# end
```

次に、ポートの帯域幅を 80% に制限する例を示します。

```
スイッチ(config)# interface gigabitethernet2/0/1
スイッチ(config-if)# srr-queue bandwidth limit 80
```

このコマンドを80%に設定すると、ポートは該当期間の20%はアイドルになります。回線レートは接続速度の 80%（800 Mbps）に低下します。ただし、ハードウェアはライン レートを 6% 単位で調整しているため、この値は厳密ではありません。

次の作業

QoS 設定でこれらの自動機能を使用できるかどうかについては、自動 QoS のマニュアルを参照してください。

第 46 章

自動 QoS の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

自動 QoS の前提条件

標準 QoS または自動 QoS を設定する前に、次の事項を十分に理解しておく必要があります。

- 使用するアプリケーションのタイプおよびネットワークのトラフィックパターン
- トラフィックの特性およびネットワークのニーズ。バースト性の高いトラフィックかどうかの判別。音声およびビデオストリーム用の帯域幅確保の必要性
- ネットワークの帯域幅要件および速度
- ネットワーク上の輻輳発生箇所

自動 QoS の設定に関する情報

自動 QoS の概要

自動 QoS 機能を使用して、QoS 機能の配置を容易にできます。自動 QoS は、ネットワーク設計を確認し、スイッチがさまざまなトラフィック フローに優先度を指定できるように QoS 設定をイネーブルにします。自動 QoS は、デフォルト（ディセーブル）の QoS 動作を使用せずに、出力キューを使用します。スイッチはパケットの内容やサイズに関係なく、各パケットにベストエフォート型のサービスを提供し、単一キューからパケットを送信します。

自動 QoS をイネーブルにすると、トラフィック タイプおよび入力パケット ラベルに基づいてトラフィックを自動的に分類します。スイッチは分類した結果を使用して適切な出力キューを選択します。

自動 QoS コマンドを使用して、次のシスコ デバイスに接続しているポートを識別できます。

- Cisco IP Phone
- Cisco SoftPhone アプリケーションを実行しているデバイス
- Cisco TelePresence
- Cisco IP Camera
- Cisco Digital Media Player

また、auto-QoS コマンドを使用してアップリンクを介して信頼のおけるトラフィックを受信するポートを指定します。自動 QoS は次の機能を実行します。

- 条件付きで信頼できるインターフェイスによる自動 QoS デバイスの有無の検出
- QoS 分類の設定
- 出力キューの設定

生成された自動 QoS 設定

デフォルトでは、自動 QoS はすべてのポートでディセーブルです。パケットは変更されません。つまり、パケットの CoS 値、DSCP 値、および IP precedence 値は変更されません。

インターフェイスの最初のポートで自動 QoS 機能をイネーブルにすると、次のようになります。

- 入力パケットラベルを使用して、トラフィックの分類、パケットラベルの割り当て、入力キューと出力キューの設定が行われます。
- QoS はグローバルにイネーブル（**mls qos** グローバル コンフィギュレーション コマンド）になり、他のグローバル コンフィギュレーション コマンドが自動的に生成されます（例：グローバルな自動 QoS 設定 （1078 ページ） を参照）。

- スイッチで信頼境界の機能がイネーブルになり、サポートされているデバイスを検出するために Cisco Discovery Protocol（CDP）が使用されます。
- パケットがプロファイル内にあるかプロファイル外にあるかを判断するためにポリシングが使用され、パケット上のアクションが指定されます。

VOIP デバイスの詳細

以下のアクティビティは、これらの自動 QoS コマンドをポート上で実行する場合に発生します。

- Cisco IP Phone に接続されたネットワークエッジのポートで **auto qos voip cisco-phone** コマンドを入力すると、スイッチは信頼境界機能をイネーブルにします。パケットに 24、26、または 46 という DSCP 値がない場合、またはパケットがプロファイル外にある場合、スイッチは DSCP 値を 0 に変更します。Cisco IP Phone が存在しない場合、入力分類はパケットの QoS ラベルを信用しないように設定されます。ポリシングは、スイッチが信頼境界機能をイネーブルにする前に、ポリシーマップの分類に一致するトラフィックに適用されます。
- **auto qos voip cisco-softphone** インターフェイス コンフィギュレーション コマンドを、Cisco SoftPhone を稼働するデバイスに接続されたネットワークのエッジのポートに入力すると、スイッチはポリシングを使用して、パケットがプロファイル内にあるかプロファイル外にあるかを判断し、パケット上のアクションを指定します。パケットに 24、26、または 46 という DSCP 値がない場合、またはパケットがプロファイル外にある場合、スイッチは DSCP 値を 0 に変更します。
- ネットワーク内部に接続されたポート上で **auto qos voip trust** インターフェイス コンフィギュレーション コマンドを入力すると、非ルーテッド ポートの場合は入力パケット内の CoS 値、ルーテッドポートの場合は入力パケット内の DSCP 値が信頼されます（前提条件は、トラフィックがすでに他のエッジデバイスによって分類されていることです）。

表 *96 :* トラフィック タイプ、パケット ラベル、およびキュー

	VoIP データ トラフィック	VoIP コントロール トラフィック	ルーティング プロトコル トラフィック	STP BPDU トラフィック	リアルタイム ビデオ トラフィック	その他すべてのトラフィック	
DSCP の値	46	24、26	48	56	34	–	
CoS 値	5	3	6	7	3	–	
CoS から出力キューへのマッピング	4、5（キュー 1）	2、3、6、7（キュー 2）			0（キュー 3）	2（キュー 3）	0、1（キュー 4）

スイッチは、次の表の設定値に従ってポートの出力キューを設定します。次の表に、出力キューに対して生成された自動 QoS の設定を示します。

表 *97 :* 出力キューに対する *auto-QoS* の設定

出力キュー	出力キュー	キュー番号	キュー ウェイト（帯域幅）	ギガビット対応ポートのキュー（バッファ）サイズ	**10/100** イーサネットポートのキュー（バッファ）サイズ
プライオリティ	**1**	**4、5**	**最大 100%**	**25%**	**15%**
SRR 共有	2	2、3、6、7	10%	25%	25%
SRR 共有	3	0	60%	25%	40%
SRR 共有	4	1	20%	25%	20%

- **auto qos voip cisco-phone**、**auto qos voip cisco-softphone** または **auto qos voip trust** インターフェイス コンフィギュレーション コマンドを使用して Auto-QoS をイネーブルにすると、スイッチはトラフィックタイプと入力パケットラベルに基づいて自動的に QoS 設定を生成し、例：グローバルな自動 QoS 設定 （1078 ページ） に示されるコマンドをポートに適用します。

ビデオ、信頼、および分類用の拡張自動 QoS

自動 QoS は、ビデオをサポートするように拡張されました。ここでは、Cisco TelePresence System と Cisco IP Camera からのトラフィックを分類して信頼する自動設定が生成されます。

自動 QoS 設定の移行

レガシー自動 QoS から拡張自動 QoS への自動 QoS 設定の移行は、次の場合に発生します。

- スイッチが 12.2(55)SE イメージで起動されます。QoS はディセーブルです。

  インターフェイス上のいずれかのビデオまたは音声の信頼設定によって、拡張自動 QoS コマンドが自動的に生成されます。

- スイッチが QoS でイネーブルになっている場合（次のガイドラインが適用されます）。
  - 音声デバイスで条件付き信頼用にインターフェイスを設定すると、レガシー自動 QoS VoIP 設定だけが生成されます。
  - ビデオ デバイスで条件付き信頼用にインターフェイスを設定すると、拡張自動 QoS VoIP 設定が生成されます。
  - 新しいインターフェイスの自動 QoS コマンドに基づいて分類または条件き信頼でインターフェイスを設定すると、拡張自動 QoS 設定が生成されます。

- **auto qos srnd4** グローバル コンフィギュレーション コマンドがイネーブルの際に、新しいデバイスを接続すると自動 QoS の移行が発生する場合。

（注） レガシー自動 QoS で以前に設定したインターフェイスが拡張自動 QoS に移行すると、新しいグローバル QoS コマンドに合わせて音声コマンドと設定が更新されます。

拡張自動 QoS からレガシー自動 QoS への自動 QoS 設定の移行が行われるのは、インターフェイスから既存の自動 QoS 設定をすべてディセーブルにした場合だけです。

自動 QoS 設定時の注意事項

自動 QoS を設定する前に、次の事項を確認してください。

- auto-QoS をイネーブルにした後、名前に *AutoQoS* を含むポリシー マップやを変更しないでください。ポリシーマップやを変更する必要がある場合、そのコピーを作成し、コピーしたポリシー マップやを変更します。生成したポリシー マップではなくこの新しいポリシー マップを使用するには、生成したポリシー マップをインターフェイスから削除し、新しいポリシー マップをインターフェイスに適用します。
- auto-QoS のデフォルトを利用するには、auto-QoS をイネーブルにしてから、その他の QoS コマンドを設定する必要があります。必要に応じて QoS 設定を微調整できますが、自動 QoS が完了した後にのみ調整することを推奨します。
- 自動 QoS は、スタティックアクセス、ダイナミックアクセス、音声 VLAN アクセス、およびトランクポートでイネーブルにできます。
- デフォルトでは、CDP 機能はすべてのポート上でイネーブルです。自動 QoS が適切に動作するために、CDP をディセーブルにしないでください。

自動 QoS VoIP に関する考慮事項

自動 QoS VoIP を設定する前に、次の事項を確認してください。

- 自動 QoS は、非ルーテッド ポートおよびルーテッド ポートで Cisco IP Phone に VoIP のスイッチを設定します。また、自動 QoS は Cisco SoftPhone アプリケーションを稼働するデバイスの VoIP 用にスイッチを設定します。

（注） Cisco SoftPhone を稼働するデバイスが非ルーテッド ポートまたはルーテッド ポートに接続されている場合、スイッチはポート単位で Cisco SoftPhone アプリケーション 1 つのみをサポートします。

- ルーテッド ポートで Cisco IP Phone の自動 QoS をイネーブルにすると、スタティック IP アドレスを IP Phone に割り当てます。

- このリリースは、Cisco IP SoftPhone Version 1.3(3) 以降のみをサポートします。
- 接続される装置は Cisco Call Manager バージョン 4 以降を使用する必要があります。

拡張された自動 QoS に関する考慮事項

自動 QoS は、ビデオをサポートするように拡張されました。ここでは、Cisco TelePresence System と Cisco IP Camera からのトラフィックを分類して信頼する自動設定が生成されます。

拡張自動 QoS を設定する前に、次の事項を確認してください。

- **auto qos srnd4** グローバル コンフィギュレーション コマンドは、拡張自動 QoS 設定の結果として生成されます。

実行コンフィギュレーションでの自動 QoS の影響

自動 QoS がイネーブルになると、**auto qos** インターフェイス コンフィギュレーション コマンドおよび生成されたグローバル コンフィギュレーションが実行コンフィギュレーションに追加されます。

スイッチは、自動 QoS が生成したコマンドを、CLI から入力したように適用します。既存のユーザー設定では、生成されたコマンドの適用に失敗することがあります。また、生成されたコマンドで既存の設定が上書きされることもあります。これらのアクションが警告なしで発生する可能性があります。生成されたコマンドがすべて正常に適用された場合、上書きされなかったユーザー入力の設定は実行コンフィギュレーション内に残ります。上書きされたユーザー入力の設定は、現在の設定をメモリに保存せずに、スイッチをリロードすると復元できます。生成コマンドが適用されなかった場合、以前の実行コンフィギュレーションが復元されます。

自動 QoS の設定方法

自動 QoS の設定

自動 QoS のイネーブル化

QoS パフォーマンスを最適化するには、ネットワーク内のすべてのデバイスで自動 QoS をイネーブルにします。

手順の概要

1. **configure terminal**
2. **interface***interface-id*
3. 次のいずれかを使用します。
   - **auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**}

- **auto qos video** {**cts** | **ip-camera** | **media-player**}
- **auto qos classify** [**police**]
- **auto qos trust** {**cos** | **dscp**}

4. **exit**
5. **interface***interface-id*
6. **auto qos trust**
7. **end**
8. **show auto qos interface***interface-id*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface***interface-id* 例： `スイッチ(config)# interface gigabitethernet 3/0/1`	ビデオデバイスに接続されたポートか、またはネットワーク内部の別の信頼できるスイッチまたはルータに接続されたアップリンク ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかを使用します。 • **auto qos voip** {**cisco-phone**	**cisco-softphone**	**trust**} • **auto qos video** {**cts**	**ip-camera**	**media-player**} • **auto qos classify** [**police**] • **auto qos trust** {**cos**	**dscp**} 例： `スイッチ(config-if)# auto qos trust dscp`	VoIP 用の自動 QoS を有効にします。 • **cisco-phone**：ポートが Cisco IP Phone に接続されている場合、着信パケットの QoS ラベルは電話が検出された場合のみ信頼されます。 • **cisco-softphone**：ポートが Cisco SoftPhone 機能を実行するデバイスに接続されています。 • **trust**：アップリンク ポートが信頼性のあるスイッチまたはルータに接続されていて、入力パケットの VoIP トラフィック分類が信頼されています。 ビデオ デバイス用の自動 QoS を有効にします。 • **cts**：Cisco Telepresence System に接続されているポート。 • **ip-camera**：Cisco ビデオ監視カメラに接続されているポート。 • **media-player**：CDP 対応 Cisco Digital Media Player に接続されているポート。

	コマンドまたはアクション	目的
		着信パケットの QoS ラベルが信頼されるのは、システムが検知される場合に限ります。 分類用の自動 QoS を有効にします。 • **police**：QoS ポリシー マップを定義し、それらをポートに適用してポリシングを設定します（ポートベースの QoS）。 信頼できるインターフェイス用の自動 QoS を有効にします。 • **cos**：サービス クラス • **dscp**：DiffServ コード ポイント。 • <cr>：信頼インターフェイス。
ステップ 4	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 5	**interface***interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/1`	信頼できるスイッチまたはルータに接続されていると識別されたスイッチ ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	**auto qos trust** 例： `スイッチ(config-if)# auto qos trust`	ポートで自動 QoS を有効にし、そのポートが信頼できるルータまたはスイッチに接続されるように指定します。
ステップ 7	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show auto qos interface***interface-id* 例： `スイッチ# show auto qos interface gigabitethernet 2/0/1`	入力を確認します。 このコマンドは、自動 QoS がイネーブルであるインターフェイス上の自動 QoS コマンドを表示します。自動 QoS 設定およびユーザーの変更を表示するに

	コマンドまたはアクション	目的
		は、**show running-config** 特権 EXEC コマンドを使用します。

自動 QoS に関するトラブルシューティング

自動 QoS のトラブルシューティングを行うには、**debug auto qos** 特権 EXEC コマンドを使用します。詳細については、このリリースに対応するコマンドリファレンスにある **debug auto qos** コマンドを参照してください。

ポートで自動 QoS を無効にするには、**auto qos** インターフェイス コンフィギュレーション コマンドの **no** 形式（**no auto qos voip** など）を使用します。このポートに対して、auto-QoS によって生成されたインターフェイス コンフィギュレーション コマンドだけが削除されます。auto-QoS をイネーブルにした最後のポートで、**no auto qos voip** コマンドを入力すると、auto-QoS によって生成されたグローバル コンフィギュレーション コマンドが残っている場合でも、auto-QoS はディセーブルと見なされます（グローバル コンフィギュレーションによって影響を受ける他のポートでのトラフィックの中断を避けるため）。

自動 QoS の監視

表 *98 :* 自動 *QoS* の監視用コマンド

コマンド	説明
show auto qos [**interface** [*interface-type*]]	最初の自動 QoS 設定を表示します。 **show auto qos** コマンド出力と **show running-config** コマンド出力を比較してユーザー定義の QoS 設定を比較できます。
show mls qos aggregate policer *policer_name*	自動 QoS によって影響されるかもしれない QoS 集約ポリシング設定に関する情報を表示します。
show mls qos interface [*interface-type* | **buffers** | **policers** | **queueing** | **statistics**]	自動 QoS によって影響されるかもしれない QoS インターフェイス設定に関する情報を表示します。
show mls qos maps [**cos-dscp** | **cos-output-q** | **dscp-cos** | **dscp-mutation** | **dscp-output-q** | **ip-prec-dscp** | **policed-dscp**]	自動 QoS によって影響されるかもしれない QoS マップ設定に関する情報を表示します。
show mls qos queue-set *queue-set ID*	自動 QoS によって影響されるかもしれない QoS キューセット設定に関する情報を表示します。

コマンド	説明
show mls qos stack-port buffers	自動 QoS によって影響されるかもしれない QoS スタック ポート バッファ設定に関する情報を表示します。
show mls qos stack-qset	自動 QoS によって影響されるかもしれない QoS スタック キューセット設定に関する情報を表示します。
show running-config	自動 QoS によって影響されるかもしれない QoS 設定に関する情報を表示します。 **show auto qos** コマンド出力と **show running-config** コマンド出力を比較してユーザー定義の QoS 設定を比較できます。

自動 QoS の設定例

例：グローバルな自動 QoS 設定

次の表は、自動 QoS および拡張自動 QoS に対してスイッチによって自動的に生成されたコマンドを説明しています。

表 *99 :* 生成された自動 *QoS* 設定

説明	自動的に生成されるコマンド {voip}	自動的に生成される拡張コマンド {Video|Trust|Classify}
スイッチが自動的に標準 QoS をイネーブルにして Cos/DSCP マップ（着信パケットの CoS 値の DSCP 値へのマッピング）を設定します。	スイッチ(config)# **mls qos** スイッチ(config)# **mls qos map cos-dscp** **0 8 16 26 32 46 48 56**	スイッチ(config)# **mls qos** スイッチ(config)# **mls qos map cos-dscp** **0 8 16 24 32 46 48 56**

説明	自動的に生成されるコマンド **{voip}**	自動的に生成される拡張コマンド **{Video|Trust|Classify}**
スイッチが、自動的に CoS 値を出力キューおよびしきい値 ID にマッピングします。	スイッチ(config)# **no mls qos srr-queue output cos-map** スイッチ(config)# **mls qos srr-queue output cos-map queue 1 threshold 3 5** スイッチ(config)# **mls qos srr-queue output cos-map queue 2 threshold 3 3 6 7** スイッチ(config)# **mls qos srr-queue output cos-map queue 3 threshold 3 2 4** スイッチ(config)# **mls qos srr-queue output cos-map queue 4 threshold 2 1** スイッチ(config)# **mls qos srr-queue output cos-map queue 4 threshold 3 0**	スイッチ(config)# **no mls qos srr-queue output cos-map** スイッチ(config)# **mls qos srr-queue output cos-map queue 1 threshold 3 4 5** スイッチ(config)# **mls qos srr-queue output cos-map queue 2 threshold 3 6 7** スイッチ(config)# **mls qos srr-queue output cos-map queue 2 threshold 1 2** スイッチ(config)# **mls qos srr-queue output cos-map queue 2 threshold 2 3** スイッチ(config)# **mls qos srr-queue output cos-map queue 3 threshold 3 0** スイッチ(config)# **mls qos srr-queue output cos-map queue 4 threshold 3 1**

説明	自動的に生成されるコマンド {voip}	自動的に生成される拡張コマンド {Video|Trust|Classify}
スイッチが、自動的に DSCP 値を出力キューおよびしきい値 ID にマッピングします。		スイッチ(config)# **no mls qos srr-queue output dscp-map** スイッチ(config)# **mls qos srr-queue output dscp-map queue 1 threshold 3 32** **33 40 41 42 43 44 45 46 47** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 1 16** **17 18 19 20 21 22 23** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 1 26** **27 28 29 30 31 34 35 36 37 38 39** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 2 24** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 48** **49 50 51 52 53 54 55 56** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 57** **58 59 60 61 62 63** スイッチ(config)# **mls qos srr-queue output dscp-map queue 3 threshold 3 0** **1 2 3 4 5 6 7** スイッチ(config)# **mls qos srr-queue output dscp-map queue 4 threshold 1 8** **9 11 13 15** スイッチ(config)# **mls qos srr-queue output dscp-map queue 4 threshold 2 10** **12 14**

説明	自動的に生成されるコマンド {voip}	自動的に生成される拡張コマンド {Video|Trust|Classify}
	スイッチ(config)# **no mls qos srr-queue output dscp-map** スイッチ(config)# **mls qos srr-queue output dscp-map queue 1 threshold 3 40 41 42 43 44 45 46 47** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 24 25 26 27 28 29 30 31** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 48 49 50 51 52 53 54 55** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 56 57 58 59 60 61 62 63** スイッチ(config)# **mls qos srr-queue output dscp-map queue 3 threshold 3 16 17 18 19 20 21 22 23** スイッチ(config)# **mls qos srr-queue output dscp-map queue 3 threshold 3 32 33 34 35 36 37 38 39** スイッチ(config)# **mls qos srr-queue output dscp-map queue 4 threshold 1 8** スイッチ(config)# **mls qos srr-queue output dscp-map queue 4 threshold 2 9 10 11 12 13 14 15** スイッチ(config)# **mls qos srr-queue output dscp-map queue 4 threshold 3 0 1 2 3 4 5 6 7**	

説明	自動的に生成されるコマンド **{voip}**	自動的に生成される拡張コマンド **{Video|Trust|Classify}**
スイッチが自動的に出力キューのバッファ サイズを設定します。ポートにマッピングされた出力キューの帯域幅と SRR モード（シェーピングまたは共有）を設定します。	スイッチ(config)# **mls qos queue-set output 1 threshold 1 138 138 92 138** スイッチ(config)# **mls qos queue-set output 1 threshold 2 138 138 92 400** スイッチ(config)# **mls qos queue-set output 1 threshold 3 36 77 100 318** スイッチ(config)# **mls qos queue-set output 1 threshold 4 20 50 67 400** スイッチ(config)# **mls qos queue-set output 2 threshold 1 149 149 100 149** スイッチ(config)# **mls qos queue-set output 2 threshold 2 118 118 100 235** スイッチ(config)# **mls qos queue-set output 2 threshold 3 41 68 100 272** スイッチ(config)# **mls qos queue-set output 2 threshold 4 42 72 100 242** スイッチ(config)# **mls qos queue-set output 1 buffers 10 10 26 54** スイッチ(config)# **mls qos queue-set output 2 buffers 16 6 17 61** スイッチ(config-if)# **priority-queue out** スイッチ(config-if)# **srr-queue bandwidth share 10 10 60 20**	スイッチ(config)# **mls qos queue-set output 1 threshold 2 100 100 50 200** スイッチ(config)# **mls qos queue-set output 1 threshold 2 125 125 100 400** スイッチ(config)# **mls qos queue-set output 1 threshold 3 100 100 100 400** スイッチ(config)# **mls qos queue-set output 1 threshold 4 60 150 50 200** スイッチ(config)# **mls qos queue-set output 1 buffers 15 25 40 20**

例：VoIP デバイス用に生成される自動 QoS 設定

次の表は、スイッチで VoIP デバイスの自動 QoS に対して自動的に生成されるコマンドについて説明しています。

表 *100 : VoIP* デバイス用に生成される自動 *QoS* 設定

説明	自動的に生成されるコマンド（VoIP）
スイッチが自動的に標準 QoS をイネーブルにして Cos/DSCP マップ（着信パケットの CoS 値の DSCP 値へのマッピング）を設定します。	スイッチ(config)# **mls qos** スイッチ(config)# **mls qos map cos-dscp 0 8 16 26 32 46 48 56**
スイッチが、自動的に CoS 値を出力キューおよびしきい値 ID にマッピングします。	スイッチ(config)# **no mls qos srr-queue output cos-map** スイッチ(config)# **mls qos srr-queue output cos-map queue 1 threshold 3 5** スイッチ(config)# **mls qos srr-queue output cos-map queue 2 threshold 3 3 6 7** スイッチ(config)# **mls qos srr-queue output cos-map queue 3 threshold 3 2 4** スイッチ(config)# **mls qos srr-queue output cos-map queue 4 threshold 2 1** スイッチ(config)# **mls qos srr-queue output cos-map queue 4 threshold 3 0**
スイッチが、自動的に DSCP 値を出力キューおよびしきい値 ID にマッピングします。	スイッチ(config)# **no mls qos srr-queue output dscp-map** スイッチ(config)# **mls qos srr-queue output dscp-map queue 1 threshold 3 40 41 42 43 44 45 46 47** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 24 25 26 27 28 29 30 31** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 48 49 50 51 52 53 54 55** スイッチ(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 56 57 58 59 60 61 62 63** スイッチ(config)# **mls qos srr-queue output dscp-map queue 3 threshold 3 16 17 18 19 20 21 22 23** スイッチ(config)# **mls qos srr-queue output dscp-map queue 3 threshold 3 32 33 34 35 36 37 38 39** スイッチ(config)# **mls qos srr-queue output dscp-map queue 4 threshold 1 8** スイッチ(config)# **mls qos srr-queue output dscp-map queue 4 threshold 2 9 10 11 12 13 14 15** スイッチ(config)# **mls qos srr-queue output dscp-map queue 4 threshold 3 0 1 2 3 4 5 6 7**

説明	自動的に生成されるコマンド（VoIP）
スイッチが自動的に出力キューのバッファ サイズを設定します。ポートにマッピングされた出力キューの帯域幅と SRR モード（シェーピングまたは共有）を設定します。	Switchスイッチconfig)# **mls qos queue-set output 1 threshold 1 138** **138 92 138** スイッチ(config)# **mls qos queue-set output 1 threshold 2 138 138 92 400** スイッチ(config)# **mls qos queue-set output 1 threshold 3 36 77 100 318** スイッチ(config)# **mls qos queue-set output 1 threshold 4 20 50 67 400** スイッチ(config)# **mls qos queue-set output 2 threshold 1 149 149 100 149** スイッチ(config)# **mls qos queue-set output 2 threshold 2 118 118 100 235** スイッチ(config)# **mls qos queue-set output 2 threshold 3 41 68 100 272** スイッチ(config)# **mls qos queue-set output 2 threshold 4 42 72 100 242** スイッチ(config)# **mls qos queue-set output 1 buffers 10 10 26 54** スイッチ(config)# **mls qos queue-set output 2 buffers 16 6 17 61** スイッチ(config-if)# **priority-que out** スイッチ(config-if)# **srr-queue bandwidth share 10 10 60 20**

auto qos voip cisco-phone コマンドを入力すると、スイッチが自動的に信頼境界機能をイネーブルにし、CDP を使用して Cisco IP Phone の有無を検出します（以下を参照）。

```
スイッチ(config-if)# mls qos trust device cisco-phone
```

auto qos voip cisco-softphone コマンドを入力すると、スイッチが自動的にクラスマップおよびポリシーマップを作成します（以下を参照）。

```
スイッチ(config)# mls qos map policed-dscp 24 26 46 to 0
スイッチ(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
スイッチ(config-cmap)# match ip dscp ef
スイッチ(config)# class-map match-all AutoQoS-VoIP-Control-Trust
スイッチ(config-cmap)# match ip dscp cs3 af31
スイッチ(config)# policy-map AutoQoS-Police-SoftPhone
スイッチ(config-pmap)# class AutoQoS-VoIP-RTP-Trust
スイッチ(config-pmap-c)# set dscp ef
スイッチ(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AutoQoS-VoIP-Control-Trust
スイッチ(config-pmap-c)# set dscp cs3
スイッチ(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

クラス マップとポリシー マップを作成すると、スイッチは自動的にポリシー マップ（別名 *AutoQoS-Police-SoftPhone*）を、Cisco SoftPhone 機能を備えた自動 QoS がイネーブルである入力インターフェイスに適用します（以下を参照）。

```
スイッチ(config-if)# service-policy input AutoQoS-Police-SoftPhone
```

例：VoIP デバイス用に生成される自動 QoS 設定

auto qos voip cisco-phone コマンドを入力すると、スイッチが自動的に信頼境界機能をイネーブルにし、CDP を使用して Cisco IP Phone の有無を検出します。

```
スイッチ(config-if)# mls qos trust device cisco-phone
```

auto qos voip cisco-softphone コマンドを入力すると、スイッチが自動的にクラス マップおよびポリシー マップを作成します（以下を参照）。

```
スイッチ(config)# mls qos map policed-dscp 24 26 46 to 0
スイッチ(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
スイッチ(config-cmap)# match ip dscp ef
スイッチ(config)# class-map match-all AutoQoS-VoIP-Control-Trust
スイッチ(config-cmap)# match ip dscp cs3 af31
スイッチ(config)# policy-map AutoQoS-Police-SoftPhone
スイッチ(config-pmap)# class AutoQoS-VoIP-RTP-Trust
スイッチ(config-pmap-c)# set dscp ef
スイッチ(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AutoQoS-VoIP-Control-Trust
スイッチ(config-pmap-c)# set dscp cs3
スイッチ(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

クラス マップとポリシー マップを作成すると、スイッチは自動的にポリシー マップ（別名 *AutoQoS-Police-SoftPhone*）を、Cisco SoftPhone 機能を備えた自動 QoS がイネーブルである入力インターフェイスに適用します。

```
スイッチ(config-if)# service-policy input AutoQoS-Police-SoftPhone
```

auto qos voip cisco-phone コマンドを入力すると、スイッチが自動的にクラス マップおよびポリシー マップを作成します（以下を参照）。

```
スイッチ(config-if)# mls qos trust device cisco-phone
```

auto qos voip cisco-softphone コマンドを入力すると、スイッチが自動的にクラス マップおよびポリシー マップを作成します（以下を参照）。

```
スイッチ(config)# mls qos map policed-dscp 24 26 46 to 0
スイッチ(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
```

```
スイッチ(config-cmap)# match ip dscp ef
スイッチ(config)# class-map match-all AutoQoS-VoIP-Control-Trust
スイッチ(config-cmap)# match ip dscp cs3 af31
スイッチ(config)# policy-map AutoQoS-Police-CiscoPhone
スイッチ(config-pmap)# class AutoQoS-VoIP-RTP-Trust
スイッチ(config-pmap-c)# set dscp ef
スイッチ(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AutoQoS-VoIP-Control-Trust
スイッチ(config-pmap-c)# set dscp cs3
スイッチ(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

クラス マップとポリシー マップを作成すると、スイッチは自動的にポリシー マップ（別名 *AutoQoS-Police-SoftPhone*）を、Cisco SoftPhone 機能を備えた自動 QoS がイネーブルである入力インターフェイスに適用します。

```
スイッチ(config-if)# service-policy input AutoQoS-Police-SoftPhone
```

例：拡張されたビデオ、信頼、および分類デバイス用に自動 QoS で生成される設定

次の拡張自動 QoS コマンドを入力すると、スイッチは CoS/DSCP のマッピングを設定します（着信パケットの CoS 値を DSCP 値にマップします）。

- **auto qos video cts**
- **auto qos video ip-camera**
- **auto qos video media-player**
- **auto qos trust**
- **auto qos trust cos**
- **auto qos trust dscp**

次のコマンドは、上記の自動 QoS コマンドのいずれかを入力した後に開始されます。

```
スイッチ(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
```

（注） クラス マップとポリシー マップは設定されません。

auto qos classify コマンドを入力すると、スイッチが自動的にクラスマップおよびポリシーマップを作成します（以下を参照）。

```
スイッチ(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
スイッチ(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
```

```
スイッチ(config)# class-map match-all AUTOQOS_MULTIENHANCED_CONF_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENHANCED-CONF
スイッチ(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
スイッチ(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
スイッチ(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
スイッチ(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
スイッチ(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
スイッチ(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICY
スイッチ(config-pmap)# class AUTOQOS_MULTIENHANCED_CONF_CLASS
スイッチ(config-pmap-c)# set dscp af41
スイッチ(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
スイッチ(config-pmap-c)# set dscp af11
スイッチ(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
スイッチ(config-pmap-c)# set dscp af21
スイッチ(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
スイッチ(config-pmap-c)# set dscp cs1
スイッチ(config-pmap)# class AUTOQOS_SIGNALING_CLASS
スイッチ(config-pmap-c)# set dscp cs3
スイッチ(config-pmap)# class AUTOQOS_DEFAULT_CLASS
スイッチ(config-pmap-c)# set dscp default
;
スイッチ(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICY
```

auto qos classify police コマンドを入力すると、スイッチが自動的にクラス マップおよびポリシー マップを作成します（以下を参照）。

```
スイッチ(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
スイッチ(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
スイッチ(config)# class-map match-all AUTOQOS_MULTIENHANCED_CONF_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENHANCED-CONF
スイッチ(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
スイッチ(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
スイッチ(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
スイッチ(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
スイッチ(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
スイッチ(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICE-POLICY
スイッチ(config-pmap)# class AUTOQOS_MULTIENHANCED_CONF_CLASS
スイッチ(config-pmap-c)# set dscp af41
スイッチ(config-pmap-c)# police 5000000 8000 exceed-action drop
スイッチ(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
スイッチ(config-pmap-c)# set dscp af11
スイッチ(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
```

```
スイッチ(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
スイッチ(config-pmap-c)# set dscp af21
スイッチ(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
スイッチ(config-pmap-c)# set dscp cs1
スイッチ(config-pmap-c)# police 10000000 8000 exceed-action drop
スイッチ(config-pmap)# class AUTOQOS_SIGNALING_CLASS
スイッチ(config-pmap-c)# set dscp cs3
スイッチ(config-pmap-c)# police 32000 8000 exceed-action drop
スイッチ(config-pmap)# class AUTOQOS_DEFAULT_CLASS
スイッチ(config-pmap-c)# set dscp default
スイッチ(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
;
スイッチ(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICE-POLICY
```

これは、**auto qos voip cisco-phone** コマンドの拡張コンフィギュレーションです。

```
スイッチ(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
スイッチ(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
スイッチ(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
スイッチ(config-cmap)# match ip dscp ef
スイッチ(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
スイッチ(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
スイッチ(config-cmap)# match ip dscp cs3
スイッチ(config)# policy-map AUTOQOS-SRND4-CISCOPHONE-POLICY
スイッチ(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
スイッチ(config-pmap-c)# set dscp ef
スイッチ(config-pmap-c)# police 128000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
スイッチ(config-pmap-c)# set dscp cs3
スイッチ(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AUTOQOS_DEFAULT_CLASS
スイッチ(config-pmap-c)# set dscp default
スイッチ(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
;
スイッチ(config-if)# service-policy input AUTOQOS-SRND4-CISCOPHONE-POLICY
```

これは、**auto qos voip cisco-softphone** コマンドの拡張コンフィギュレーションです。

```
スイッチ(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
スイッチ(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
スイッチ(config)# class-map match-all AUTOQOS_MULTIENHANCED_CONF_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENHANCED-CONF
スイッチ(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
スイッチ(config-cmap)# match ip dscp ef
スイッチ(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
スイッチ(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
スイッチ(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
```

```
スイッチ(config-cmap)# match ip dscp cs3
スイッチ(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
スイッチ(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
スイッチ(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
スイッチ(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER

スイッチ(config)# policy-map AUTOQOS-SRND4-SOFTPHONE-POLICY
スイッチ(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
スイッチ(config-pmap-c)# set dscp ef
スイッチ(config-pmap-c)# police 128000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
スイッチ(config-pmap-c)# set dscp cs3
スイッチ(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)#class AUTOQOS_MULTIENHANCED_CONF_CLASS
スイッチ(config-pmap-c)#set dscp af41
スイッチ(config-pmap-c)# police 5000000 8000 exceed-action drop
スイッチ(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
スイッチ(config-pmap-c)# set dscp af11
スイッチ(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
スイッチ(config-pmap-c)# set dscp af21
スイッチ(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
スイッチ(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
スイッチ(config-pmap-c)# set dscp cs1
スイッチ(config-pmap-c)# police 10000000 8000 exceed-action drop
スイッチ(config-pmap)# class AUTOQOS_SIGNALING_CLASS
スイッチ(config-pmap-c)# set dscp cs3
スイッチ(config-pmap-c)# police 32000 8000 exceed-action drop
スイッチ(config-pmap)# class AUTOQOS_DEFAULT_CLASS
スイッチ(config-pmap-c)# set dscp default
;
スイッチ(config-if)# service-policy input AUTOQOS-SRND4-SOFTPHONE-POLICY
```

自動 QoS の関連情報

自動 QoS 設定で特定の QoS の変更をする必要がある場合は、QoS のマニュアルを確認してください。

第 VIII 部

ルーティング

第 47 章

IP ユニキャスト ルーティングの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

IP ユニキャスト ルーティングの設定に関する情報

このモジュールでは、スイッチで IP Version 4（IPv4）ユニキャスト ルーティングを設定する方法について説明します。

スタティック ルーティング、 で使用できます。Catalyst 3560-CX スイッチ上の IP Base フィーチャ セットおよび IP Services フィーチャ セット。Catalyst 2960-CX スイッチではスタティック ルーティングのみをサポートします。

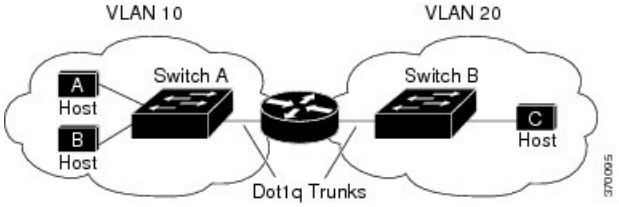
（注） IPv4 トラフィックに加えて、IP バージョン 6（IPv6）ユニキャスト ルーティングをイネーブルにし、IPv6 トラフィックを転送するようにインターフェイスを設定できます。

IP ルーティングに関する情報

一部のネットワーク環境で、VLAN（仮想 LAN）は各ネットワークまたはサブネットワークに関連付けられています。IP ネットワークで、各サブネットワークは 1 つの VLAN に対応しています。VLAN を設定すると、ブロードキャスト ドメインのサイズを制御し、ローカル トラフィックをローカル内にとどめることができます。ただし、異なる VLAN 内のネットワーク デバイスが相互に通信するには、VLAN 間でトラフィックをルーティング（VLAN 間ルーティング）するレイヤ 3 デバイス（ルータ）が必要です。VLAN 間ルーティングでは、適切な宛先 VLAN にトラフィックをルーティングするため、1 つまたは複数のルータを設定します。

図 *83 :* ルーティング トポロジの例

次の図に基本的なルーティング トポロジを示します。スイッチ A は VLAN 10 内、スイッチ B は VLAN 20 内にあります。ルータには各 VLAN のインターフェイスが備わっています。



VLAN 10 内のホスト A が VLAN 10 内のホスト B と通信する場合、ホスト A はホスト B 宛にアドレス指定されたパケットを送信します。スイッチ A はパケットをルータに送信せず、ホスト B に直接転送します。

ホスト A から VLAN 20 内のホスト C にパケットを送信する場合、スイッチ A はパケットをルータに転送し、ルータは VLAN 10 インターフェイスでトラフィックを受信します。ルータはルーティング テーブルを調べて正しい発信インターフェイスを判別し、VLAN 20 インター

フェイスを経由してパケットをスイッチ B に送信します。スイッチ B はパケットを受信し、ホスト C に転送します。

ルーティング タイプ

ルータおよびレイヤ 3 スイッチは、次の方法でパケットをルーティングできます。

- デフォルト ルーティング
- 事前にプログラミングされているトラフィックのスタティック ルートの使用
- ルーティング プロトコルによるルートの動的な計算

スイッチは、スタティック ルートとデフォルト ルートをサポートします。ルーティング プロトコルはサポートされません。

IP ルーティングの設定方法

デバイス上で、IPルーティングはデフォルトで無効となっているため、ルーティングを行う前に、IP ルーティングを有効にする必要があります。

次の手順では、次に示すレイヤ 3 インターフェイスの 1 つを指定する必要があります。

- ルーテッドポート：**no switchport** インターフェイス コンフィギュレーション コマンドを使用し、レイヤ 3 ポートとして設定された物理ポートです。
- スイッチ仮想インターフェイス（SVI）：**interface vlan** *vlan_id* グローバル コンフィギュレーション コマンドによって作成された VLAN インターフェイス。デフォルトではレイヤ 3 インターフェイスです。
- レイヤ 3 モードの Etherchannel ポートチャネル： **interface port-channel** *port-channel-number* グローバル コンフィギュレーション コマンドを使用し、イーサネット インターフェイスをチャネルグループにバインドして作成されたポートチャネル論理インターフェイスです。

（注） スイッチは、ユニキャスト ルーテッド トラフィックのトンネル インターフェイスをサポートしません。

ルーティングが発生するすべてのレイヤ3インターフェイスに、IPアドレスを割り当てる必要があります。

（注） レイヤ 3 スイッチは、各ルーテッド ポートおよび SVI に割り当てられた IP アドレスを持つことができます。

ルーティングを設定するための主な手順は次のとおりです。

- VLANインターフェイスをサポートするには、デバイスまたはスイッチ スタックでVLANを作成および設定し、レイヤ 2 インターフェイスに VLAN メンバーシップを割り当てます。詳細については、「VLAN の設定」の章を参照してください。
- レイヤ 3 インターフェイスを設定します。
- スイッチ上で IP ルーティングをイネーブルに設定します。
- レイヤ 3 インターフェイスに IP アドレスを割り当てます。
- 選択したルーティング プロトコルをスイッチ上でイネーブルにします。
- ルーティング プロトコル パラメータを設定します（任意）。

IP アドレッシングの設定方法

IP ルーティングを設定するには、レイヤ 3 ネットワーク インターフェイスに IP アドレスを割り当ててインターフェイスをイネーブルにし、IPを使用するインターフェイスを経由してホストとの通信を許可する必要があります。次の項では、さまざまな IP アドレス指定機能の設定方法について説明します。IPアドレスをインターフェイスに割り当てる手順は必須ですが、その他の手順は任意です。

- アドレス指定のデフォルト設定
- ネットワーク インターフェイスへの IP アドレスの割り当て
- アドレス解決方法の設定
- IP ルーティングがディセーブルの場合のルーティング支援機能
- ブロードキャスト パケットの処理方法の設定
- IP アドレスのモニターリングおよびメンテナンス

IP アドレス指定のデフォルト設定

表 *101 :* アドレス指定のデフォルト設定

機能	デフォルト設定
IP アドレス	未定義
ARP	ARP キャッシュに永続的なエントリはありません カプセル化：標準イーサネット形式の ARP 14400 秒（4 時間）

機能	デフォルト設定
IP ブロードキャスト アドレス	255.255.255.255（すべて 1）
IP クラスレス ルーティング	イネーブル。
IP デフォルト ゲートウェイ	ディセーブル。
IP ダイレクト ブロードキャスト	ディセーブル（すべてのIPダイレクトブロードキャストがドロップされます）
IP ドメイン	ドメイン リスト：ドメイン名は未定義 ドメイン検索：イネーブル ドメイン名：イネーブル
IP 転送プロトコル	ヘルパー アドレスが定義されているか、またはユーザー データグラム プロトコル（UDP）フラッディングが設定されている場合、デフォルトポートではUDP転送がイネーブルとなります ローカル ブロードキャスト：ディセーブル スパニングツリー プロトコル（STP）：ディセーブル ターボフラッディング：ディセーブル
IP ヘルパー アドレス	ディセーブル。
IP ホスト	ディセーブル。
ICMP Router Discovery Protocol（IRDP）	ディセーブル。 イネーブルの場合のデフォルト： • ブロードキャストIRDPアドバタイズメント • アドバタイズメント間の最大インターバル：600 秒 • アドバタイズ間の最小インターバル：最大インターバルの 0.75 倍 • プリファレンス：0
IP プロキシ ARP	イネーブル。
IP ルーティング	ディセーブル。

機能	デフォルト設定
IP サブネットゼロ	ディセーブル

ネットワーク インターフェイスへの IP アドレスの割り当て

IP アドレスは IP パケットの送信先を特定します。一部の IP アドレスは特殊な目的のために予約されていて、ホスト、サブネット、またはネットワーク アドレスには使用できません。RFC 1166 の『Internet Numbers』には IP アドレスに関する公式の説明が記載されています。

インターフェイスには、1 つのプライマリ IP アドレスを設定できます。マスクで、IP アドレス中のネットワーク番号を示すビットが識別できます。マスクを使用してネットワークをサブネット化する場合、そのマスクをサブネット マスクと呼びます。割り当てられているネットワーク番号については、インターネット サービス プロバイダにお問い合わせください。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** **例：** `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** **例：** `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* **例：** `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ3インターフェイスを指定します。
ステップ 4	**no switchport** **例：** `スイッチ(config-if)# no switchport`	レイヤ 2 コンフィギュレーション モードからインターフェイスを削除します（物理インターフェイスの場合）。
ステップ 5	**ip address** *ip-address subnet-mask* **例：** `スイッチ(config-if)# ip address 10.1.5.1 255.255.255.0`	IP アドレスおよび IP サブネット マスクを設定します。

	コマンドまたはアクション	目的
ステップ 6	**no shutdown** 例： `スイッチ(config-if)# no shutdown`	物理インターフェイスをイネーブルにします。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show ip route** 例： `スイッチ# show ip route`	入力を確認します。
ステップ 9	**show ip interface** [*interface-id*] 例： `スイッチ# show ip interface gigabitethernet 1/0/1`	入力を確認します。
ステップ 10	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

サブネット ゼロの使用

サブネット アドレスがゼロであるサブネットを作成しないでください。同じアドレスを持つネットワークおよびサブネットがある場合に問題が発生することがあります。たとえば、ネットワーク 131.108.0.0 のサブネットが 255.255.255.0 の場合、サブネット ゼロは 131.108.0.0 と記述され、ネットワーク アドレスと同じとなってしまいます。

すべてが 1 のサブネット（131.108.255.0）は使用可能です。また、IP アドレス用にサブネット スペース全体が必要な場合は、サブネット ゼロの使用をイネーブルにできます（ただし推奨できません）。

デフォルトに戻して、サブネットゼロの使用を無効にするには、**no ip subnet-zero** グローバル コンフィギュレーション コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip subnet-zero** 例： スイッチ(config)# ip subnet-zero	インターフェイス アドレスおよびルーティングのアップデート時にサブネットゼロの使用をイネーブルにします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

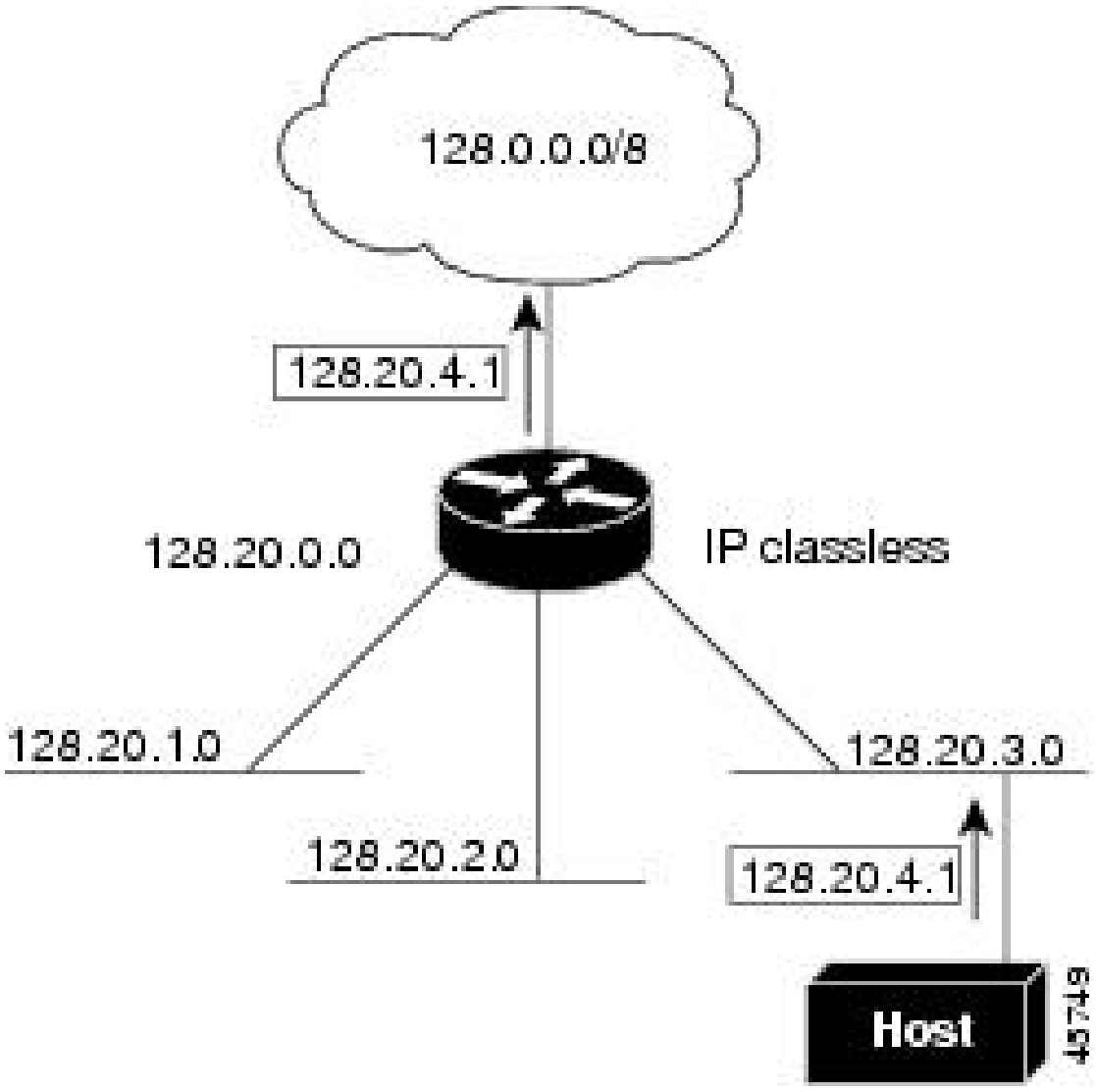
クラスレス ルーティング

ルーティングを行うように設定されたデバイスで、クラスレスルーティング動作はデフォルトで有効となっています。クラスレス ルーティングがイネーブルの場合、デフォルト ルートがないネットワークのサブネット宛てパケットをルータが受信すると、ルータは最適なスーパーネット ルートにパケットを転送します。スーパーネットは、単一の大規模アドレス空間をシ

ミュレートするために使用されるクラスCアドレス空間の連続ブロックで構成されています。スーパーネットは、クラス B アドレス空間の急速な枯渇を回避するために設計されました。

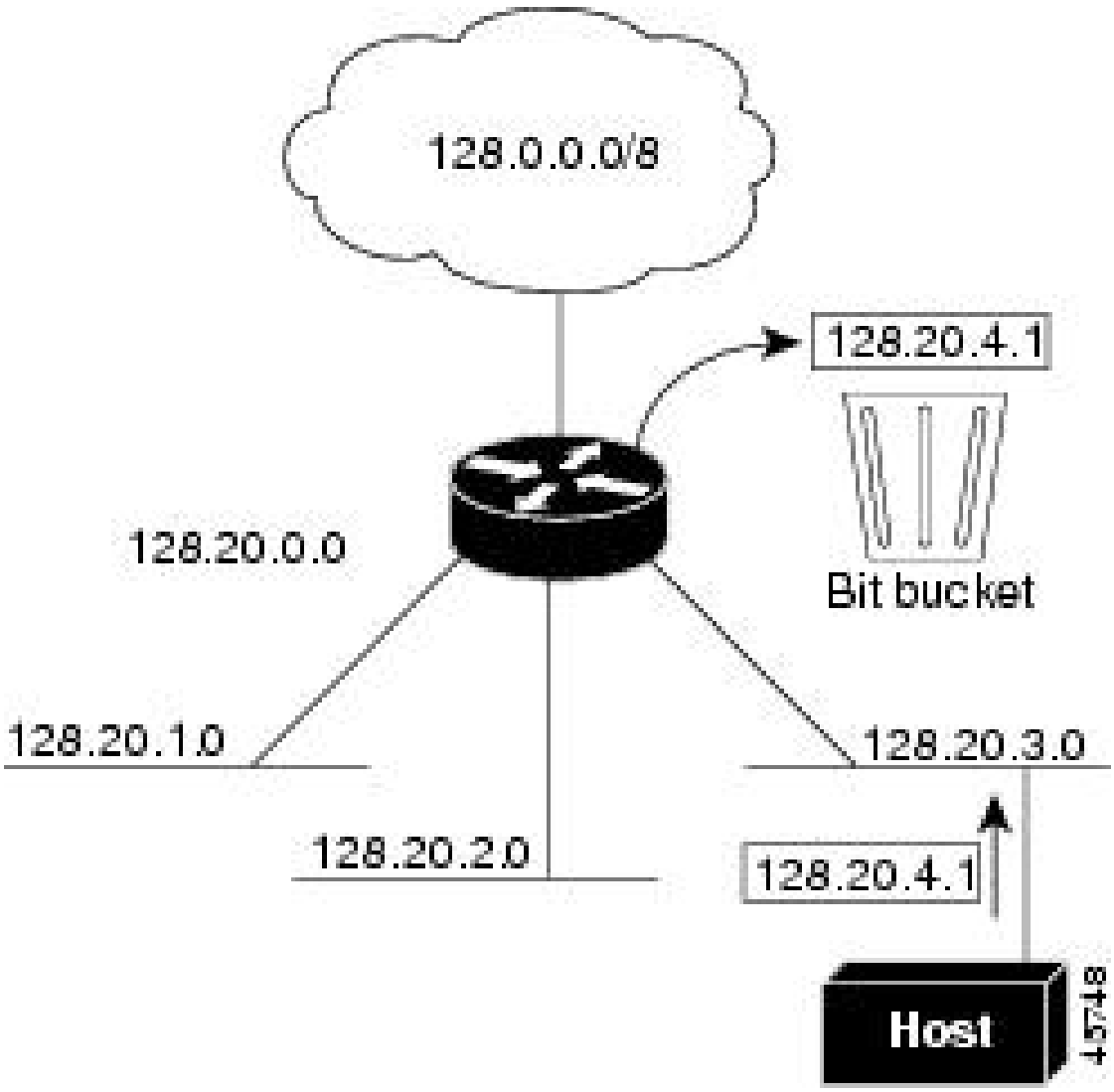
図では、クラスレスルーティングがイネーブルとなっています。ホストがパケットを128.20.4.1に送信すると、ルータはパケットを廃棄せずに、最適なスーパーネットルートに転送します。クラスレス ルーティングがディセーブルの場合、デフォルト ルートがないネットワークのサブネット宛てパケットを受信したルータは、パケットを廃棄します。

図 *84 : IP* クラスレス ルーティングがイネーブルの場合



図では、ネットワーク 128.20.0.0 のルータはサブネット 128.20.1.0、128.20.2.0、128.20.3.0 に接続されています。ホストがパケットを 128.20.4.1 に送信した場合、ネットワークのデフォルトルートが存在しないため、ルータはパケットを廃棄します。

図 *85 : IP* クラスレス ルーティングがディセーブルの場合



デバイスが認識されないサブネット宛てのパケットを最適なスーパーネットルートに転送しないようにするには、クラスレスルーティング動作を無効にします。

クラスレス ルーティングのディセーブル化

デバイスが認識されないサブネット宛てのパケットを最適なスーパーネットルートに転送しないようにするには、クラスレスルーティング動作を無効にします。

手順

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no ip classless** 例： スイッチ(config)#no ip classless	クラスレス ルーティング動作をディセーブルにします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

アドレス解決方法の設定

アドレス解決を設定するために必要な作業は次のとおりです。

アドレス解決

インターフェイス固有の IP 処理方法を制御するには、アドレス解決を行います。IP を使用するデバイスには、ローカル セグメントまたは LAN 上のデバイスを一意に定義するローカル アドレス（MAC アドレス）と、デバイスが属するネットワークを特定するネットワーク アドレスがあります。

ローカルアドレス（MAC アドレス）は、パケット ヘッダーのデータ リンク層（レイヤ 2）セクションに格納されて、データ リンク（レイヤ 2）デバイスによって読み取られるため、デー

タ リンク アドレスと呼ばれます。ソフトウェアがイーサネット上のデバイスと通信するには、デバイスの MAC アドレスを学習する必要があります。IP アドレスから MAC アドレスを学習するプロセスを、アドレス解決と呼びます。MAC アドレスから IP アドレスを学習するプロセスを、逆アドレス解決と呼びます。

デバイスでは、次の形式のアドレス解決を行うことができます。

- ARP：IP アドレスを MAC アドレスと関連付けるために使用されます。ARP は IP アドレスを入力と解釈し、対応する MAC アドレスを学習します。次に、IP アドレス/MAC アドレス アソシエーションを ARP キャッシュにストアし、すぐに取り出せるようにします。その後、IPデータグラムがリンク層フレームにカプセル化され、ネットワークを通じて送信されます。イーサネット以外の IEEE 802 ネットワークにおける IP データグラムのカプセル化、および ARP 要求や応答については、サブネットワーク アクセス プロトコル（SNAP）で規定されています。
- プロキシ ARP：ルーティング テーブルを持たないホストで、他のネットワークまたはサブネット上のホストの MAC アドレスを学習できるようにします。デバイス（ルータ）が送信者と異なるインターフェイス上のホストに宛てたARP要求を受信した場合、そのルータに他のインターフェイスを経由してそのホストに至るすべてのルートが格納されていれば、ルータは自身のローカル データ リンク アドレスを示すプロキシARPパケットを生成します。ARP要求を送信したホストはルータにパケットを送信し、ルータはパケットを目的のホストに転送します。

デバイスでは、ARP と同様の機能（ローカル MAC アドレスでなく IP アドレスを要求する点を除く）を持つ Reverse Address Resolution Protocol（RARP）を使用することもできます。RARP を使用するには、ルータインターフェイスと同じネットワーク セグメント上にRARPサーバーを設置する必要があります。サーバーを識別するには、**ip rarp-server** *address* インターフェイス コンフィギュレーション コマンドを使用します。

スタティック ARP キャッシュの定義

ARPおよび他のアドレス解決プロトコルを使用すると、IPアドレスとMACアドレス間をダイナミックにマッピングできます。ほとんどのホストではダイナミック アドレス解決がサポートされているため、通常の場合、スタティック ARP キャッシュ エントリを指定する必要はありません。静的 ARP キャッシュエントリを定義する必要がある場合は、グローバルに行うことができます。グローバルに定義すると、IP アドレスを MAC アドレスに変換するためにデバイスが使用する ARP キャッシュに永続的なエントリをインストールします。また、指定された IP アドレスに属しているかのように、デバイスが ARP 要求に応答するように指定することもできます。ARP エントリを永続的なエントリにしない場合は、ARP エントリのタイムアウト期間を指定できます。

手順

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**arp** *ip-address hardware-address type* 例： スイッチ(config)# ip 10.1.5.1 c2f3.220a.12f4 arpa	ARP キャッシュ内で IP アドレスを MAC（ハードウェア）アドレスに関連付け、次に示すカプセル化タイプのいずれかを指定します。 • **arpa**：ARPカプセル化（イーサネットインターフェイス用） • **snap**：Subnetwork Address Protocol カプセル化（トークン リングおよび FDDI インターフェイス用） • **sap**：HP の ARP タイプ
ステップ 4	**arp** *ip-address hardware-address type* [**alias**] 例： スイッチ(config)# ip 10.1.5.3 d7f3.220d.12f5 arpa alias	（任意）指定された IP アドレスがスイッチに属する場合と同じ方法で、スイッチが ARP 要求に応答するように指定します。
ステップ 5	**interface** *interface-id* 例： スイッチ(config)# interface gigabitethernet 1/0/1	インターフェイス コンフィギュレーション モードを開始し、設定するインターフェイスを指定します。
ステップ 6	**arp** *timeout seconds* 例： スイッチ(config-if)# arp 20000	（任意）ARP キャッシュ エントリがキャッシュに保持される期間を設定します。デフォルト値は 14400 秒（4 時間）です。指定できる範囲は 0 ～ 2147483 秒です。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	**show interfaces** [*interface-id*] 例： `スイッチ# show interfaces gigabitethernet 1/0/1`	すべてのインターフェイスまたは特定のインターフェイスで使用される ARP のタイプおよびタイムアウト値を確認します。
ステップ 9	**show arp** 例： `スイッチ# show arp`	ARP キャッシュの内容を表示します。
ステップ 10	**show ip arp** 例： `スイッチ# show ip arp`	ARP キャッシュの内容を表示します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ARP のカプセル化の設定

IP インターフェイスでは、イーサネット ARP カプセル化（**arpa** キーワードで表される）がデフォルトで有効に設定されています。ネットワークの必要性に応じて、カプセル化方法を SNAP に変更できます。

カプセル化タイプを無効にするには、**no arp arpa** または **no arp snap** インターフェイス コンフィギュレーション コマンドを使用します。

手順

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

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。
ステップ 4	**arp** {**arpa** | **snap**} 例： `スイッチ(config-if)# arp arpa`	ARP カプセル化方法を指定します。 • **arpa**：Address Resolution Protocol • **snap**：Subnetwork Address Protocol
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show interfaces** [*interface-id*] 例： `スイッチ# show interfaces`	すべてのインターフェイスまたは指定されたインターフェイスの ARP カプセル化設定を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

プロキシ ARP のイネーブル化

デフォルトでは、プロキシ ARP がデバイスで使用されます。ホストが他のネットワークまたはサブネット上のホストの MAC アドレスを学習できるようにするためです。

手順

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# interface gigabitethernet 1/0/2	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。
ステップ 4	**ip proxy-arp** 例： スイッチ(config-if)# ip proxy-arp	インターフェイス上でプロキシ ARP をイネーブルにします。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ip interface** [*interface-id*] 例： スイッチ# show ip interface gigabitethernet 1/0/2	指定されたインターフェイスまたはすべてのインターフェイスの設定を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IP ルーティングがディセーブルの場合のルーティング支援機能

次のメカニズムを使用することで、デバイスは、IP ルーティングが有効でない場合、別のネットワークへのルートを学習できます。

- 『Proxy ARP』
- デフォルト ゲートウェイ
- ICMP Router Discovery Protocol（IRDP）

プロシキ ARP

プロキシ ARP は、他のルートを学習する場合の最も一般的な方法です。プロキシ ARP を使用すると、ルーティング情報を持たないイーサネットホストと、他のネットワークまたはサブ

ネット上のホストとの通信が可能になります。このホストでは、すべてのホストが同じローカルイーサネット上にあり、ARP を使用して MAC アドレスを学習すると想定されています。デバイスが送信元と異なるネットワーク上にあるホストに宛てた ARP 要求を受信した場合、デバイスはそのホストへの最適なルートがあるかどうかを調べます。最適なルートがある場合、デバイスは自身のイーサネット MAC アドレスが格納された ARP 応答パケットを送信します。要求の送信元ホストはパケットをデバイスに送信し、スイッチは目的のホストにパケットを転送します。プロキシ ARP は、すべてのネットワークをローカルな場合と同様に処理し、IP アドレスごとに ARP 要求を実行します。

プロシキ ARP

プロキシ ARP は、デフォルトでイネーブルに設定されています。ディセーブル化されたプロキシ ARP をイネーブルにするには、「プロキシ ARP のイネーブル化」の項を参照してください。プロキシ ARP は、他のルータでサポートされているかぎり有効です。

デフォルト ゲートウェイ

ルートを特定するもう 1 つの方法は、デフォルト ルータ、つまりデフォルト ゲートウェイを定義する方法です。ローカルでないすべてのパケットはこのルータに送信されます。このルータは適切なルーティングを行う、または IP 制御メッセージプロトコル（ICMP）リダイレクトメッセージを返信するという方法で、ホストが使用するローカル ルータを定義します。デバイスはリダイレクトメッセージをキャッシュに格納し、各パケットをできるだけ効率的に転送します。この方法には、デフォルト ルータがダウンした場合、または使用できなくなった場合に、検出が不可能となる制限があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** **例：** `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** **例：** `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip default-gateway** *ip-address* **例：** `スイッチ(config)# ip default gateway 10.1.5.1`	デフォルト ゲートウェイ（ルータ）を設定します。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show ip redirects** 例： スイッチ# show ip redirects	設定を確認するため、デフォルト ゲートウェイ ルータのアドレスを表示します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ICMP Router Discovery Protocol

ルータディスカバリを使用すると、デバイスは ICMP Router Discovery Protocol（IRDP）を使用し、他のネットワークへのルートを動的に学習します。ホストは IRDP を使用し、ルータを特定します。クライアントとして動作しているデバイスは、ルータディスカバリパケットを生成します。ホストとして動作しているデバイスは、ルータディスカバリパケットを受信します。デバイスは Routing Information Protocol（RIP）ルーティングのアップデートを受信し、この情報を使用してルータの場所を推測することもできます。ルーティングデバイスによって送信されたルーティングテーブルは、実際にはデバイスにストアされません。どのシステムがデータを送信しているのかが記録されるだけです。IRDP を使用する利点は、プライオリティと、パケットが受信されなくなってからデバイスがダウンしていると見なされるまでの期間の両方をルータごとに指定できることです。

検出された各デバイスは、デフォルト ルータの候補となります。現在のデフォルト ルータがダウンしたと宣言された場合、または再送信が多すぎて TCP 接続がタイムアウトになりつつある場合、プライオリティが上位のルータが検出されると、最も高いプライオリティを持つ新しいルータが選択されます。

ICMP Router Discovery Protocol（IRDP）

インターフェイスで IRDP ルーティングを行う場合は、インターフェイスで IRDP 処理をイネーブルにしてください。IRDP 処理をイネーブルにすると、デフォルトのパラメータが適用されます。

これらのパラメータを変更することもできます。**maxadvertinterval** 値を変更すると、**holdtime** 値および **minadvertinterval** 値も変更されます。最初に **maxadvertinterval** 値を変更し、次に **holdtime** 値または **minadvertinterval** 値のどちらかを手動で変更することが重要です。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ3インターフェイスを指定します。
ステップ 4	**ip irdp** 例： `スイッチ(config-if)# ip irdp`	インターフェイスでIRDP処理をイネーブルにします。
ステップ 5	**ip irdp multicast** 例： `スイッチ(config-if)# ip irdp multicast`	（任意）IP ブロードキャストの代わりとして、マルチキャスト アドレス（224.0.0.1）に IRDP アドバタイズを送信します。 （注） このコマンドを使用すると、IRDP パケットをマルチキャストとして送信するサンマイクロシステムズ社の Solaris との互換性を維持できます。実装機能の中には、これらのマルチキャストを受信できないものも多くあります。このコマンドを使用する前に、エンドホストがこの機能に対応していることを確認してください。
ステップ 6	**ip irdp holdtime** *seconds* 例： `スイッチ(config-if)# ip irdp holdtime 1000`	（任意）アドバタイズが有効である IRDP 期間を設定します。デフォルトは **maxadvertinterval** 値の 3 倍です。**maxadvertinterval** 値よりも大きな値（9000 秒以下）を指定する必要があります。**maxadvertinterval**値を変更すると、この値も変更されます。

	コマンドまたはアクション	目的
ステップ 7	**ip irdp maxadvertinterval** *seconds* 例： `スイッチ(config-if)# ip irdp maxadvertinterval 650`	（任意）アドバタイズメントのIRDP最大間隔を設定します。デフォルトは 600 秒です。
ステップ 8	**ip irdp minadvertinterval** *seconds* 例： `スイッチ(config-if)# ip irdp minadvertinterval 500`	（任意）アドバタイズ間のIRDPの最小インターバルを設定します。デフォルト値は**maxadvertinterval**値の 0.75 倍です。**maxadvertinterval** を変更すると、この値も新しいデフォルト値（**maxadvertinterval** の 0.75 倍）に変更されます。
ステップ 9	**ip irdp preference** *number* 例： `スイッチ(config-if)# ip irdp preference 2`	（任意）デバイスの IRDP プリファレンス レベルを設定します。指定できる範囲は-231 ～ 231 です。デフォルトは0です。大きな値を設定すると、ルータのプリファレンス レベルも高くなります。
ステップ 10	**ip irdp address** *address* [*number*] 例： `スイッチ(config-if)# ip irdp address 10.1.10.10`	（任意）プロキシアドバタイズを行うためのIRDPアドレスとプリファレンスを設定します。
ステップ 11	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 12	**show ip irdp** 例： `スイッチ# show ip irdp`	IRDP 値を表示し、設定を確認します。
ステップ 13	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ブロードキャスト パケットの処理方法の設定

これらの方式をイネーブルにするには、次に示す作業を実行します。

- ダイレクト ブロードキャストから物理ブロードキャストへの変換のイネーブル化

- UDP ブロードキャスト パケットおよびプロトコルの転送
- IP ブロードキャスト アドレスの確立
- IP ブロードキャストのフラッディング

ブロードキャスト パケットの処理

IP インターフェイスアドレスを設定したあとで、ルーティングを有効にしたり、1 つまたは複数のルーティングプロトコルを設定したり、ネットワーク ブロードキャストへのデバイスの応答方法を設定したりできます。ブロードキャストは、物理ネットワーク上のすべてのホスト宛てのデータ パケットです。デバイスでは、2 種類のブロードキャストがサポートされています。

- ダイレクト ブロードキャスト パケット：特定のネットワークまたは一連のネットワークに送信されます。ダイレクト ブロードキャスト アドレスには、ネットワークまたはサブネット フィールドが含まれます。
- フラッディング ブロードキャスト パケット：すべてのネットワークに送信されます。

（注） **storm-control**インターフェイス コンフィギュレーション コマンドを使用して、トラフィック抑制レベルを設定し、レイヤ2インターフェイスでブロードキャスト、ユニキャスト、マルチキャストトラフィックを制限することもできます。

ルータはローカル ケーブルまでの範囲を制限して、ブロードキャスト ストームを防ぎます。ブリッジ（インテリジェントなブリッジを含む）はレイヤ 2 デバイスであるため、ブロードキャストはすべてのネットワーク セグメントに転送され、ブロードキャスト ストームを伝播します。ブロードキャスト ストーム問題を解決する最善の方法は、ネットワーク上で単一のブロードキャスト アドレス方式を使用することです。最新の IP 実装機能ではほとんどの場合、アドレスをブロードキャストアドレスとして使用するように設定できます。デバイスの場合も含めて、多くの実装機能では、ブロードキャストメッセージを転送するためのアドレス方式が複数サポートされています。

ダイレクト ブロードキャストから物理ブロードキャストへの変換のイネーブル化

デフォルトでは、IP ダイレクト ブロードキャストがドロップされるため、転送されることはありません。IP ダイレクト ブロードキャストがドロップされると、ルータが DoS 攻撃（サービス拒絶攻撃）にさらされる危険が少なくなります。

ブロードキャストが物理（MAC レイヤ）ブロードキャストになるインターフェイスでは、IP ダイレクト ブロードキャストの転送をイネーブルにできます。**ip forward-protocol** グローバル コンフィギュレーション コマンドを使用し、設定されたプロトコルだけを転送できます。

転送するブロードキャストを制御するアクセス リストを指定できます。アクセス リストを指定すると、アクセス リストで許可されている IP パケットだけが、ダイレクト ブロードキャス

トから物理ブロードキャストに変換できるようになります。アクセスリストの詳細については、「Security」のセクションの「Configuring ACLs」の章を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	インターフェイス コンフィギュレーション モードを開始し、設定するインターフェイスを指定します。
ステップ 4	**ip directed-broadcast** [*access-list-number*] 例： `スイッチ(config-if)# ip directed-broadcast 103`	インターフェイス上で、ダイレクト ブロードキャストから物理ブロードキャストへの変換をイネーブルにします。転送するブロードキャストを制御するアクセス リストを指定できます。アクセス リストを指定すると、アクセス リストで許可されている IP パケットだけが変換可能になります。
ステップ 5	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**ip forward-protocol** {**udp** [*port*] | **nd** | **sdns**} 例： `スイッチ(config)# ip forward-protocol nd`	ブロードキャスト パケットを転送するとき、ルータによって転送されるプロトコルおよびポートを指定します。 • **udp**：UPD データグラムを転送します。 port：（任意）転送される UDP サービスを制御する宛先ポートです。 • **nd**：ND データグラムを転送します。 • **sdns**：SDNS データグラムを転送します。

	コマンドまたはアクション	目的
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show ip interface** [*interface-id*] 例： スイッチ# show ip interface	指定されたインターフェイスまたはすべてのインターフェイスの設定を確認します。
ステップ 9	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

UDP ブロードキャスト パケットおよびプロトコル

ユーザー データグラム プロトコル（UDP）は IP のホスト間レイヤ プロトコルで、TCP と同様です。UDP はオーバーヘッドが少ない、コネクションレスのセッションを 2 つのエンド システム間に提供しますが、受信されたデータグラムの確認応答は行いません。場合に応じてネットワーク ホストは UDP ブロードキャストを使用し、アドレス、コンフィギュレーション、名前に関する情報を検索します。このようなホストが、サーバーを含まないネットワーク セグメント上にある場合、通常 UDP ブロードキャストは転送されません。この状況を改善するには、特定のクラスのブロードキャストをヘルパー アドレスに転送するように、ルータのインターフェイスを設定します。インターフェイスごとに、複数のヘルパーアドレスを使用できます。

UDP 宛先ポートを指定し、転送される UDP サービスを制御できます。複数の UDP プロトコルを指定することもできます。旧式のディスクレス Sun ワークステーションおよびネットワーク セキュリティ プロトコル SDNS で使用される Network Disk（ND）プロトコルも指定できます。

ヘルパー アドレスがインターフェイスに定義されている場合、デフォルトでは UDP と ND の両方の転送がイネーブルになっています。

UDP ブロードキャスト パケットおよびプロトコルの転送

UDP ブロードキャストの転送を設定するときに UDP ポートを指定しないと、ルータは BOOTP フォワーディング エージェントとして動作するように設定されます。BOOTP パケットは Dynamic Host Configuration Protocol（DHCP）情報を伝達します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# interface gigabitethernet 1/0/1	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ3インターフェイスを指定します。
ステップ 4	**ip helper-address** *address* 例： スイッチ(config-if)# ip helper address 10.1.10.1	転送をイネーブルにし、BOOTP などの UDP ブロードキャスト パケットを転送するための宛先アドレスを指定します。
ステップ 5	**exit** 例： スイッチ(config-if)# exit	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**ip forward-protocol** {**udp** [*port*] | **nd** | **sdns**} 例： スイッチ(config)# ip forward-protocol sdns	ブロードキャストパケットを転送するときに、ルータによって転送されるプロトコルを指定します。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show ip interface** [*interface-id*] 例： スイッチ# show ip interface gigabitethernet 1/0/1	指定されたインターフェイスまたはすべてのインターフェイスの設定を確認します。

	コマンドまたはアクション	目的
ステップ 9	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IP ブロードキャスト アドレスの確立

最も一般的な（デフォルトの）IP ブロードキャスト アドレスは、すべて 1 で構成されているアドレス（255.255.255.255）です。ただし、任意の形式の IP ブロードキャストアドレスを生成するようにデバイスを設定することもできます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# interface gigabitethernet 1/0/1	インターフェイス コンフィギュレーション モードを開始し、設定するインターフェイスを指定します。
ステップ 4	**ip broadcast-address** *ip-address* 例： スイッチ(config-if)# ip broadcast-address 128.1.255.255	デフォルト値と異なるブロードキャスト アドレス（128.1.255.255 など）を入力します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show ip interface** [*interface-id*] 例： スイッチ# show ip interface	指定されたインターフェイスまたはすべてのインターフェイスのブロードキャストアドレスを確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IP ブロードキャストのフラッディング

IPブロードキャストをインターネットワーク全体に、制御可能な方法でフラッディングできるようにするには、ブリッジング STP で作成されたデータベースを使用します。この機能を使用すると、ループを回避することもできます。この機能を使用できるようにするには、フラッディングが行われるインターフェイスごとにブリッジングを設定する必要があります。ブリッジングが設定されていないインターフェイス上でも、ブロードキャストを受信できます。ただし、ブリッジングが設定されていないインターフェイスでは、受信したブロードキャストが転送されません。また、異なるインターフェイスで受信されたブロードキャストを送信する場合、このインターフェイスは使用されません。

IPヘルパーアドレスのメカニズムを使用して単一のネットワーク アドレスに転送されるパケットを、フラッディングできます。各ネットワーク セグメントには、パケットのコピーが 1 つだけ送信されます。

フラッディングを行う場合、パケットは次の条件を満たす必要があります（これらの条件は、IP ヘルパー アドレスを使用してパケットを転送するときの条件と同じです）。

- パケットは MAC レベルのブロードキャストでなければなりません。
- パケットは IP レベルのブロードキャストでなければなりません。
- パケットは Trivial File Transfer Protocol（TFTP）、ドメインネームシステム（DNS）、Time、NetBIOS、ND、または BOOTP パケット、または **ip forward-protocol udp** グローバル コンフィギュレーション コマンドで指定された UDP でなければなりません。
- パケットの存続可能時間（TTL）値は 2 以上でなければなりません。

フラッディングされた UDP データグラムには、出力インターフェイスで **ip broadcast-address** インターフェイス コンフィギュレーション コマンドによって指定された宛先アドレスが表示

されます。宛先アドレスを、任意のアドレスに設定できます。このため、データグラムがネットワーク内に伝播されるにつれ、宛先アドレスが変更されることもあります。送信元アドレスは変更されません。TTL 値が減ります。

フラッディングされた UDP データグラムがインターフェイスから送信されると（場合によっては宛先アドレスが変更される）、データグラムは通常の IP 出力ルーチンに渡されます。このため、出力インターフェイスにアクセスリストがある場合、データグラムはその影響を受けます。

デバイスでは、パケットの大部分がハードウェアで転送され、デバイスの CPU を経由しません。CPUに送信されるパケットの場合は、ターボフラッディングを使用し、スパニングツリーベースの UDP フラッディングを約 4 ～ 5 倍高速化します。この機能は、ARP カプセル化用に設定されたイーサネット インターフェイスでサポートされています。

IP ブロードキャストのフラッディング

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip forward-protocol spanning-tree** 例： `スイッチ(config)# ip forward-protocol spanning-tree`	ブリッジング スパニングツリー データベースを使用し、UDPデータグラムをフラッディングします。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。

	コマンドまたはアクション	目的
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。
ステップ 7	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 8	**ip forward-protocol turbo-flood** 例： スイッチ(config)# ip forward-protocol turbo-flood	スパニングツリーデータベースを使用し、UDPデータグラムのフラッディングを高速化します。
ステップ 9	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 10	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IP アドレスのモニタリングおよびメンテナンス

特定のキャッシュ、テーブル、またはデータベースの内容が無効になっている場合、または無効である可能性がある場合は、**clear** 特権 EXEC コマンドを使用し、すべての内容を削除できます。次の表に、内容をクリアするために使用するコマンドを示します。

表 *102* : キャッシュ、テーブル、データベースをクリアするコマンド

clear arp-cache	IP ARP キャッシュおよび高速スイッチング キャッシュをクリアします。
clear host {*name* | *****}	ホスト名およびアドレス キャッシュから 1 つまたはすべてのエントリを削除します。
clear ip route {**network** [*mask*] | *****}	IP ルーティング テーブルから 1 つまたは複数のルートを削除します。

IP ルーティング テーブル、キャッシュ、データベースの内容、ノードへの到達可能性、ネットワーク内のパケットのルーティング パスなど、特定の統計情報を表示できます。次の表に、IP 統計情報を表示するために使用する特権 EXEC コマンドを示します。

表 *103* : キャッシュ、テーブル、データベースを表示するコマンド

show arp	ARP テーブル内のエントリを表示します。
show hosts	デフォルトのドメイン名、検索サービスの方式、サーバー ホスト名、およびキャッシュに格納されているホスト名とアドレスのリストを表示します。
show ip aliases	TCP ポートにマッピングされた IP アドレスを表示します（エイリアス）。
show ip arp	IP ARP キャッシュを表示します。
show ip interface [*interface-id*]	インターフェイスの IP ステータスを表示します。
show ip irdp	IRDP 値を表示します。
show ip masks *address*	ネットワーク アドレスに対して使用されるマスクおよび各マスクを使用するサブネット番号を表示します。
show ip redirects	デフォルト ゲートウェイのアドレスを表示します。
show ip route [*address* [*mask*]] | [*protocol*]	ルーティング テーブルの現在の状態を表示します。
show ip route summary	サマリー形式でルーティング テーブルの現在のステータスを表示します。

IP ユニキャスト ルーティングの設定方法

IP ユニキャスト ルーティングのイネーブル化

デフォルトで、デバイスはレイヤ 2 スイッチングモード、IP ルーティングはディセーブルとなっています。デバイスのレイヤ3機能を使用するには、IPルーティングをイネーブルにする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip routing** 例： スイッチ(config)# ip routing	IP ルーティングをイネーブルにします。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

IP ユニキャスト ルーティングのイネーブル化の例

次に、、上でスイッチ IP ルーティングを有効にする例を示します。

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# ip routing

スイッチ(config-router)# end
```

RIP に関する情報

RIP は、小規模な同種ネットワーク間で使用するために作成された Interior Gateway Protocol（IGP）です。RIP は、ブロードキャスト ユーザー データグラム プロトコル（UDP）データ パケットを使用してルーティング情報を交換するディスタンスベクトル ルーティング プロトコルです。このプロトコルは RFC 1058 に文書化されています。RIP の詳細については、『*IP Routing Fundamentals*』（Cisco Press 刊）を参照してください。

（注） RIP は Network Essentials 機能セットでサポートされています。

デバイスはRIPを使用し、30秒ごとにルーティング情報アップデート（アドバタイズメント）を送信します。180秒以上を経過しても別のルータからアップデートがルータに届かない場合、該当するルータから送られたルートは使用不能としてマークされます。240 秒後もまだ更新がない場合、ルータは更新のないルータのルーティングテーブルエントリをすべて削除します。

RIP では、各ルートの値を評価するためにホップ カウントが使用されます。ホップ カウントは、ルート内で経由されるルータ数です。直接接続されているネットワークのホップカウントは0です。ホップ カウントが16のネットワークに到達できません。このように範囲（0～15）が狭いため、RIP は大規模ネットワークには適していません。

ルータにデフォルトのネットワーク パスが設定されている場合、RIP はルータを疑似ネットワーク 0.0.0.0 にリンクするルートをアドバタイズします。0.0.0.0 ネットワークは存在しません。RIP はデフォルトのルーティング機能を実行するためのネットワークとして、このネットワークを処理します。デフォルトネットワークがRIPによって学習された場合、またはルータにラストリゾートゲートウェイがあり、RIP がデフォルトのメトリックによって設定されている場合、デバイスはデフォルトネットワークをアドバタイズします。RIP は指定されたネットワーク内のインターフェイスにアップデートを送信します。インターフェイスのネットワークを指定しなければ、RIP のアップデート中にアドバタイズされません。

RIP の設定方法

RIP のデフォルト設定

表 *104 : RIP* のデフォルト設定

機能	デフォルト設定
自動サマリー	イネーブル。
デフォルト情報送信元	ディセーブル。
デフォルト メトリック	自動メトリック変換（組み込み）
IP RIP 認証キーチェーン	認証なし 認証モード：クリア テキスト
IP RIP の起動	無効
IP スプリット ホライズン	メディアにより異なる
Neighbor	未定義
ネットワーク	指定なし
オフセット リスト	ディセーブル。
出力遅延	0 ミリ秒
タイマー基準	• 更新：30 秒 • 無効：180 秒 • ホールドダウン：180 秒 • フラッシュ：240 秒
アップデート送信元の検証	イネーブル。
バージョン	RIP バージョン 1 およびバージョン 2 パケットを受信し、バージョン 1 パケットを送信します。

基本的な RIP パラメータの設定

RIP を設定するには、ネットワークに対して RIP ルーティングを有効にします。他のパラメータを設定することもできます。デバイスでは、ネットワーク番号を設定するまでRIPコンフィギュレーション コマンドは無視されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip routing** 例： スイッチ(config)# **ip routing**	IP ルーティングを有効にします。（IP ルーティングが無効になっている場合だけ、必須です）。
ステップ 4	**router rip** 例： スイッチ(config)# **router rip**	RIP ルーティング プロセスを有効にし、ルータ コンフィギュレーション モードを開始します。
ステップ 5	**network** *network number* 例： スイッチ(config-router)# **network 12.0.0.0**	ネットワークを RIP ルーティング プロセスと関連付けます。複数の **network** コマンドを指定できます。RIP ルーティング アップデートの送受信は、これらのネットワークのインターフェイスを経由する場合だけ可能です。 （注） RIP コマンドを有効にするには、ネットワーク番号を設定する必要があります。
ステップ 6	**neighbor** *ip-address* 例： スイッチ(config-router)# **neighbor 10.2.5.1**	（任意）ルーティング情報を交換する隣接ルータを定義します。このステップを使用すると、RIP（通常はブロードキャスト プロトコル）からのルーティング アップデートが非ブロードキャスト ネットワークに到達するようになります。

	コマンドまたはアクション	目的
ステップ 7	**offset-list** [*access-list number* | *name*] {**in** | **out**} *offset* [*type number*] 例： `スイッチ(config-router)# offset-list 103 in 10`	（任意）オフセットリストをルーティングメトリックに適用し、RIPによって取得したルートへの着信および発信メトリックを増加します。アクセス リストまたはインターフェイスを使用し、オフセットリストを制限できます。
ステップ 8	**timers basic** *update invalid holddown flush* 例： `スイッチ(config-router)# timers basic 45 360 400 300`	（任意）ルーティング プロトコル タイマーを調整します。すべてのタイマーの有効範囲は 0 ～ 4294967295 秒です。 • *update*：ルーティング アップデートの送信間隔。デフォルトは 30 秒です。 • *invalid*：ルートが無効と宣言されるまでの時間。デフォルト値は 180 秒です。 • *holddown*：ルートがルーティング テーブルから削除されるまでの時間。デフォルト値は180 秒です。 • *flush*：ルーティング アップデートが延期される時間。デフォルトは 240 秒です。
ステップ 9	**version** {**1** | **2**} 例： `スイッチ(config-router)# version 2`	（任意）RIP バージョン 1 または RIP バージョン 2 のパケットだけを送受信するようにスイッチを設定します。デフォルトの場合、スイッチではバージョン 1 および 2 を受信しますが、バージョン 1 だけを送信します。インターフェイスコマンド **ip rip** {**send** | **receive**} **version 1** | **2** | **1 2**} を使用して、インターフェイスでの送受信に使用するバージョンを制御することもできます。
ステップ 10	**no auto-summary** 例： `スイッチ(config-router)# no auto-summary`	（任意）自動要約を無効にします。デフォルトでは、クラスフル ネットワーク境界を通過するときにサブプレフィックスがサマライズされます。サマライズを無効にし（RIP バージョン 2 だけ）、クラスフル ネットワーク境界にサブネットおよびホスト ルーティング情報をアドバタイズします。
ステップ 11	**output-delay** *delay* 例： `スイッチ(config-router)# output-delay 8`	（任意）送信する RIP アップデートにパケット間遅延を追加します。デフォルトでは、複数のパケットからなる RIP アップデートのパケットに、パケット間遅延が追加されません。パケットを低速なデバイスに送信する場合は、8 ～ 50 ミリ秒のパケット間遅延を追加できます。

	コマンドまたはアクション	目的
ステップ 12	**end** 例： スイッチ(config-router)# **end**	特権 EXEC モードに戻ります。
ステップ 13	**show ip protocols** 例： スイッチ# **show ip protocols**	入力を確認します。
ステップ 14	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

RIP 認証の設定

RIP バージョン 1 は認証をサポートしていません。RIP バージョン 2 のパケットを送受信する場合は、インターフェイスでRIP認証を有効にできます。インターフェイスで使用できる一連のキーは、キーチェーンによって指定されます。キーチェーンが設定されていないと、デフォルトの場合でも認証は実行されません。

RIP認証が有効であるインターフェイスでは、プレーンテキストとMD5という2つの認証モードがデバイスでサポートされます。デフォルトはプレーン テキストです。

手順

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

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するインターフェイスを指定します。
ステップ 4	**ip rip authentication key-chain** *name-of-chain* 例： `スイッチ(config-if)# ip rip authentication key-chain trees`	RIP 認証を有効にします。
ステップ 5	**ip rip authentication mode** {**text** | **md5**} 例： `スイッチ(config-if)# ip rip authentication mode md5`	プレーン テキスト認証（デフォルト）または MD5 ダイジェスト認証を使用するように、インターフェイスを設定します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

サマリー アドレスおよびスプリット ホライズン

ブロードキャストタイプの IP ネットワークに接続され、ディスタンスベクトル ルーティング プロトコルを使用するルータでは、通常ルーティング ループの発生を抑えるために、スプリット ホライズン メカニズムが使用されます。スプリット ホライズンは、ルートに関する情報の発信元であるインターフェイス上の、ルータによって、その情報がアドバタイズされないようにします。この機能を使用すると、通常の場合は複数のルータ間通信が最適化されます（特にリンクが壊れている場合）。

サマリー アドレスおよびスプリット ホライズンの設定

（注） ルートを適切にアドバタイズするため、アプリケーションがスプリット ホライズンを無効にする必要がある場合を除き、通常はこの機能を無効にしないでください。

ダイヤルアップクライアント用のネットワークアクセスサーバーで、サマライズされたローカルIPアドレスプールをアドバタイズするように、RIPが動作しているインターフェイスを設定する場合は、**ip summary-address rip** インターフェイス コンフィギュレーション コマンドを使用します。

（注） スプリット ホライズンが有効の場合、自動サマリーとインターフェイスIPサマリーアドレスはともにアドバタイズされません。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。
ステップ 4	**ip address** *ip-address subnet-mask* 例： `スイッチ(config-if)# ip address 10.1.1.10 255.255.255.0`	IP アドレスおよび IP サブネットを設定します。
ステップ 5	**ip summary-address rip** *ip-address ip-network mask* 例：	サマライズする IP アドレスおよび IP ネットワーク マスクを設定します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **ip summary-address rip 10.1.1.30 255.255.255.0**	
ステップ 6	**no ip split-horizon** 例： スイッチ(config-if)# **no ip split-horizon**	インターフェイスでスプリット ホライズンを無効にします。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show ip interface** *interface-id* 例： スイッチ# **show ip interface gigabitethernet 1/0/1**	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

スプリット ホライズンの設定

ブロードキャストタイプの IP ネットワークに接続され、ディスタンスベクトル ルーティング プロトコルを使用するルータでは、通常ルーティング ループの発生を抑えるために、スプリット ホライズン メカニズムが使用されます。スプリット ホライズンは、ルートに関する情報の発信元であるインターフェイス上の、ルータによって、その情報がアドバタイズされないようにします。この機能を使用すると、複数のルータ間通信が最適化されます（特にリンクが壊れている場合）。

（注） ルートを適切にアドバタイズするために、アプリケーションがスプリット ホライズンを無効にする必要がある場合を除き、通常この機能を無効にしないでください。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# interface **gigabitethernet 1/0/1**	インターフェイス コンフィギュレーション モードを開始し、設定するインターフェイスを指定します。
ステップ 4	**ip address** *ip-address subnet-mask* 例： スイッチ(config-if)# ip address 10.1.1.10 255.255.255.0	IP アドレスおよび IP サブネットを設定します。
ステップ 5	**no ip split-horizon** 例： スイッチ(config-if)# no ip split-horizon	インターフェイスでスプリット ホライズンを無効にします。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show ip interface** *interface-id* 例： スイッチ# show ip interface **gigabitethernet 1/0/1**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

サマリーアドレスとスプリットホライズンの構成例

次の例では、主要ネットは 10.0.0.0 です。自動サマリー アドレス 10.0.0.0 はサマリー アドレス 10.2.0.0 によって上書きされるため、10.2.0.0 はインターフェイス ギガビット イーサネット ポート 2 からアドバタイズされますが、10.0.0.0 はアドバタイズされません。この例では、インターフェイスがレイヤ 2 モード（デフォルト）の場合は、**no switchport** インターフェイス コンフィギュレーション コマンドを入力してから、**ip address** インターフェイス コンフィギュレーション コマンドを入力する必要があります。

（注） スプリットホライズンが有効である場合、（**ip summary-address rip** ルータ コンフィギュレーション コマンドによって設定される）自動サマリーとインターフェイス サマリー アドレスはともにアドバタイズされません。

```
スイッチ(config)# router rip
スイッチ(config-router)# interface gigabitethernet1/0/2
スイッチ(config-if)# ip address 10.1.5.1 255.255.255.0
スイッチ(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0
スイッチ(config-if)# no ip split-horizon
スイッチ(config-if)# exit
スイッチ(config)# router rip
スイッチ(config-router)# network 10.0.0.0
スイッチ(config-router)# neighbor 2.2.2.2 peer-group mygroup
スイッチ(config-router)# end
```

OSPF に関する情報

OSPF は IP ネットワーク専用の IGP で、IP サブネット化、および外部から取得したルーティング情報のタグ付けをサポートしています。OSPF を使用するとパケット認証も可能になり、パケットを送受信するときに IP マルチキャストが使用されます。シスコの実装では、RFC1253 の OSPF 管理情報ベース（MIB）がサポートされています。

シスコの実装は、次の主要機能を含む OSPF バージョン 2 仕様に準拠します。

- スタブ エリアの定義がサポートされています。
- 任意の IP ルーティング プロトコルによって取得されたルートは、別の IP ルーティング プロトコルに再配信されます。つまり、ドメイン内レベルで、OSPF は EIGRP および RIP によって取得したルートを取り込むことができます。OSPF ルートを RIP に伝達することもできます。
- エリア内の隣接ルータ間でのプレーン テキスト認証および MD5 認証がサポートされています。
- 設定可能なルーティング インターフェイス パラメータには、インターフェイス出力コスト、再送信インターバル、インターフェイス送信遅延、ルータ プライオリティ、ルータのデッド インターバルと hello インターバル、認証キーなどがあります。

- 仮想リンクがサポートされています。
- RFC 1587 に基づく Not-So-Stubby-Area（NSSA）がサポートされています。

通常、OSPF を使用するには、多くの内部ルータ、複数のエリアに接続された Area Border Router（ABR;エリア境界ルータ）、および自律システム境界ルータ（ASBR）間で調整する必要があります。最小設定では、すべてのデフォルト パラメータ値、エリアに割り当てられたインターフェイスが使用され、認証は行われません。環境をカスタマイズする場合は、すべてのルータの設定を調整する必要があります。

OSPF の設定方法

OSPF のデフォルト設定

表 *105 : OSPF* のデフォルト設定

機能	デフォルト設定
インターフェイス パラメータ	コスト：デフォルト コストは未定義 再送信インターバル：5 秒 送信遅延：1 秒 プライオリティ：1 hello インターバル：10 秒 デッド インターバル：hello インターバルの 4 倍 認証なし パスワードの指定なし MD5 認証はディセーブル
エリア	認証タイプ：0（認証なし） デフォルト コスト：1 範囲：ディセーブル スタブ：スタブ エリアは未定義 NSSA：NSSA エリアは未定義
自動コスト	100 Mb/s
デフォルト情報送信元	ディセーブル。イネーブルの場合、デフォルトのメトリック設定は 10 で、外部ルート タイプのデフォルトはタイプ 2 です。

機能	デフォルト設定
デフォルト メトリック	各ルーティング プロトコルに適切な、組み込みの自動メトリック変換
距離 OSPF	dist1（エリア内のすべてのルート）：110 dist2（エリア間のすべてのルート）：110 dist3（他のルーティング ドメインからのルート）：110。
OSPF データベース フィルタ	ディセーブル。すべての発信 LSA がインターフェイスにフラッディングされます。
IP OSPF 名検索	ディセーブル。
隣接関係変更ログ	イネーブル。
ネイバー	指定なし
ネイバー データベース フィルタ	ディセーブル。すべての発信 LSA はネイバーにフラッディングされます。
ネットワーク エリア	ディセーブル。
ノンストップ フォワーディング（NSF）認識	イネーブル。レイヤ 3 デバイスでは、ハードウェアやソフトウェアの変更中に、隣接する NSF 対応ルータからのパケットを転送し続けることができます。
NSF 対応	ディセーブル。 （注） デバイススタックは OSPF NSF 対応ルーティングを IPv4 に対してサポートします。
ルータ ID	OSPF ルーティング プロセスは未定義
サマリー アドレス	ディセーブル。
タイマー LSA グループのペーシング	240 秒
タイマー Shortest Path First（SPF）	spf 遅延：5 秒; spf ホールドタイム：10 秒

機能	デフォルト設定
仮想リンク	エリア ID またはルータ ID は未定義 hello インターバル：10 秒 再送信インターバル：5 秒 送信遅延：1 秒 デッド インターバル：40 秒 認証キー：キーは未定義 メッセージダイジェスト キー（MD5）：キーは未定義

ルーテッド アクセスの OSPF

Cisco IOS Release 12.2(55)SE で、IP Base イメージは OSPF for Routed Access をサポートしています。ルート制限のない複数の OSPFv2 および OSPFv3 インスタンスが必要な場合は、IP サービス イメージが必要です。また、マルチ VRF CE 機能をイネーブルにするためにも、IP サービス イメージが必要です。

OSPF for Routed Access は、特にレイヤ 3 のルーティング機能をワイヤリング クローゼットに拡張するために作成されました。

（注） OSPF for Routed Access は、動的に学習された合わせて 1000 のルートを持つ OSPFv2 インスタンスと OSPFv3 インスタンスをそれぞれ 1 つだけサポートします。IP ベース イメージは、ルーテッド アクセス用に OSPF を提供します。

ただし、これらの制限はこのリリースでは適用されません。

構内環境内の標準的なトポロジ（ハブおよびスポーク）では、すべての非ローカルトラフィックをディストリビューション レイヤに転送するディストリビューション スイッチ（ハブ）にワイヤリングクローゼット（スポーク）が接続されているため、ワイヤリング クローゼット デバイスで完全なルーティングテーブルを保持する必要はありません。OSPF for Routed Access をワイヤリングクローゼットで使用する場合、エリア間ルートおよび外部ルートに到達するためのデフォルトルートがディストリビューション デバイスによってワイヤリング クローゼット デバイスに送信される、ベストプラクティスの設計（OSPF スタブまたは完全スタブエリア構成）を使用する必要があります。

詳細については、『High Availability Campus Network Design—Routed Access Layer using EIGRP or OSPF』を参照してください。

OSPF NSF

デバイスまたはスイッチスタックは2つのレベルのノンストップ フォワーディング（NSF）をサポートしています。

OSPF NSF 認識

隣接ルータが NSF 対応である場合、レイヤ 3 デバイスでは、ルータに障害（クラッシュ）が発生してプライマリルートプロセッサ（RP）がバックアップ RP によって引き継がれる間、または処理を中断させずにソフトウェアアップグレードを行うためにプライマリ RP を手動でリロードしている間、隣接ルータからパケットを転送し続けます。

この機能をディセーブルにできません。

OSPF NSF 対応

では、前のリリースでサポートされていた OSPFv2 NSF Cisco フォーマットに加えて、OSPFv2 NSF IETF フォーマットもサポートされます。この機能の詳細については、『*NSF—OSPF (RFC 3623 OSPF Graceful Restart)*』を参照してください。

は、OSPF NSF 対応ルーティングも IPv4 に対してサポートし、スタックのアクティブスイッチ変更後のコンバージェンス向上と、トラフィック損失低減を実現します。

（注） OSPF NSF では、すべてのネイバー ネットワーク デバイスが NSF 認識である必要があります。ネットワーク セグメント上に非 NSF 認識ネイバーが検出された場合、NSF 対応ルータはそのセグメントに対する NSF 機能をディセーブルにします。すべてのデバイスが NSF 認識または NSF 対応デバイスとなっているその他のネットワーク セグメントでは、NSF 対応機能が継続して提供されます。

OSPF NSF ルーティングを有効にするには、**nsf** OSPF ルーティング コンフィギュレーション コマンドを使用します。OSPF NSF ルーティングが有効になっていることを確認するには、**show ip ospf** 特権 EXEC コマンドを使用します。

詳細については、次の URL の『*Cisco Nonstop Forwarding*』を参照してください。
http://www.cisco.com/en/US/docs/ios/ha/configuration/guide/ha-nonstp_fwdg.html

基本的な OSPF パラメータの設定

OSPF をイネーブルにするには、OSPF ルーティング プロセスを作成し、そのルーティング プロセスに関連付けられる IP アドレスの範囲を指定し、その範囲に関連付けられるエリア ID を割り当てます。

手順

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**router ospf** *process-id* 例： スイッチ(config)# **router ospf 15**	OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードを開始します。プロセス ID はローカルに割り当てられ、内部で使用される識別パラメータで、任意の正の整数を指定できます。各 OSPF ルーティング プロセスには一意の値があります。 （注） OSPF for Routed Access は、OSPFv2 インスタンスと OSPFv3 インスタンスをそれぞれ 1 つずつと、最大 1000 のダイナミックに学習されるルートをサポートします。
ステップ 3	**network** *address wildcard-mask* **area** *area-id* 例： スイッチ(config-router)# **network 10.1.1.1 255.240.0.0 area 20**	OSPF が動作するインターフェイス、およびそのインターフェイスのエリア ID を定義します。単一のコマンドにワイルドカードマスクを指定し、特定の OSPF エリアに関連付けるインターフェイスを 1 つまたは複数定義できます。エリア ID には 10 進数または IP アドレスを指定できます。
ステップ 4	**end** 例： スイッチ(config-router)#**end**	特権 EXEC モードに戻ります。
ステップ 5	**show ip protocols** 例： スイッチ# **show ip protocols**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

例：基本的な OSPF パラメータの設定

次に、OSPF ルーティング プロセスを設定し、プロセス番号 109 を割り当てる例を示します。

```
スイッチ(config)# router ospf 109
スイッチ(config-router)# network 131.108.0.0 255.255.255.0 area 24
```

OSPF インターフェイスの設定

ip ospf インターフェイス コンフィギュレーション コマンドを使用すると、インターフェイス固有の OSPF パラメータを変更できます。これらのパラメータを変更する必要はありませんが、一部のインターフェイスパラメータ（helloインターバル、デッドインターバル、認証キーなど）については、接続されたネットワーク内のすべてのルータで統一性を維持する必要があります。これらのパラメータを変更した場合は、ネットワーク内のすべてのルータの値も同様に変更してください。

（注） **ip ospf** インターフェイス コンフィギュレーション コマンドはすべてオプションです。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ3インターフェイスを指定します。
ステップ 3	**ip ospf cost** *cost* 例： `スイッチ(config-if)# ip ospf cost 8`	（任意）インターフェイスでパケットを送信するコストを明示的に指定します。
ステップ 4	**ip ospf retransmit-interval** *seconds* 例： `スイッチ(config-if)# ip ospf transmit-interval 10`	（任意）LSA 送信間隔を秒数で指定します。指定できる範囲は 1 ～ 65535 秒です。デフォルト値は 5 秒です。
ステップ 5	**ip ospf transmit-delay** *seconds* 例： `スイッチ(config-if)# ip ospf transmit-delay 2`	（任意）リンク ステート アップデート パケットを送信するまでの予測待機時間を秒数で設定します。指定できる範囲は1～65535秒です。デフォルト値は 1 秒です。
ステップ 6	**ip ospf priority** *number* 例：	（任意）ネットワークに対して、OSPF で指定されたルータを検索するときに役立つプライオリティを

	コマンドまたはアクション	目的
	`スイッチ(config-if)# ip ospf priority 5`	設定します。有効な範囲は0～255です。デフォルトは1です。
ステップ7	**ip ospf hello-interval** *seconds* 例： `スイッチ(config-if)# ip ospf hello-interval 12`	（任意）OSPFインターフェイスでhelloパケットの送信間隔を秒数で設定します。ネットワークのすべてのノードで同じ値を指定する必要があります。指定できる範囲は1～65535秒です。デフォルトは10秒です。
ステップ8	**ip ospf dead-interval** *seconds* 例： `スイッチ(config-if)# ip ospf dead-interval 8`	（任意）最後のデバイスでhelloパケットが確認されてから、OSPFルータがダウンしていることがネイバーによって宣言されるまでの時間を秒数で設定します。ネットワークのすべてのノードで同じ値を指定する必要があります。指定できる範囲は1～65535秒です。デフォルト値はhelloインターバルの4倍です。
ステップ9	**ip ospf authentication-key** *key* 例： `スイッチ(config-if)# ip ospf authentication-key password`	（任意）隣接OSPFルータで使用されるパスワードを割り当てます。パスワードには、キーボードから入力した任意の文字列（最大8バイト長）を指定できます。同じネットワーク上のすべての隣接ルータには、OSPF情報を交換するため、同じパスワードを設定する必要があります。
ステップ10	**ip ospf message-digest-key** *keyid* **md5** *key* 例： `スイッチ(config-if)# ip ospf message digest-key 16 md5 your1pass`	（任意）MDS認証をイネーブルにします。 • *keyid*：1～255のID。 • *key*：最大16バイトの英数字パスワード
ステップ11	**ip ospf database-filter all out** 例： `スイッチ(config-if)# ip ospf database-filter all out`	（任意）インターフェイスへのOSPF LSAパケットのフラッディングを阻止します。デフォルトでは、OSPFは、LSAが到着したインターフェイスを除き、同じエリア内のすべてのインターフェイスで新しいLSAをフラッドします。
ステップ12	**end** 例： `スイッチ(config)# end`	特権EXECモードに戻ります。
ステップ13	**show ip ospf interface** [*interface-name*] 例： `スイッチ# show ip ospf interface`	OSPFに関連するインターフェイス情報を表示します。

	コマンドまたはアクション	目的
ステップ 14	**show ip ospf neighbor detail** 例： `スイッチ# show ip ospf neighbor detail`	ネイバー スイッチの NSF 認証ステータスを表示します。出力には、次のいずれかが表示されます。 • *Options is 0x52* *LLS Options is 0x1 (LR)* これらの行の両方が表示される場合、ネイバー スイッチが NSF 認識です。 • *Options is 0x42*：ネイバー スイッチが NSF 認識でないことを示します。
ステップ 15	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

OSPF エリア パラメータ

複数のOSPFエリアパラメータを設定することもできます。設定できるパラメータには、エリア、スタブ エリア、およびNSSA への無許可アクセスをパスワードによって阻止する認証用パラメータがあります。スタブエリアは、外部ルートの情報が送信されないエリアです。が、代わりに、自律システム（AS）外の宛先に対するデフォルトの外部ルートが、ABR によって生成されます。NSSAではコアからそのエリアへ向かうLSAの一部がフラッディングされませんが、再配信することによって、エリア内の AS 外部ルートをインポートできます。

経路集約は、アドバタイズされたアドレスを、他のエリアでアドバタイズされる単一のサマリー ルートに統合することです。ネットワーク番号が連続する場合は、**area range** ルータ コンフィギュレーション コマンドを使用し、範囲内のすべてのネットワークを対象とするサマリールートをアドバタイズするように ABR を設定できます。

OSPF エリア パラメータの設定

始める前に

（注） OSPF **area** ルータ コンフィギュレーション コマンドはすべて任意です。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router ospf** *process-id* 例： `スイッチ(config)# router ospf 109`	OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードを開始します。
ステップ 3	**area** *area-id* **authentication** 例： `スイッチ(config-router)# area 1 authentication`	（任意）特定のエリアへの無許可アクセスに対して、パスワードベースの保護を可能にします。ID には 10 進数または IP アドレスのいずれかを指定できます。
ステップ 4	**area** *area-id* **authentication message-digest** 例： `スイッチ(config-router)# area 1 authentication message-digest`	（任意）エリアに関して MD5 認証を有効にします。
ステップ 5	**area** *area-id* **stub** [**no-summary**] 例： `スイッチ(config-router)# area 1 stub`	（任意）エリアをスタブエリアとして定義します。**no-summary** キーワードを指定すると、ABR はサマリー リンク アドバタイズメントをスタブエリアに送信できなくなります。
ステップ 6	**area** *area-id* **nssa** [**no-redistribution**] [**default-information-originate**] [**no-summary**] 例： `スイッチ(config-router)# area 1 nssa default-information-originate`	（任意）エリアを NSSA として定義します。同じエリア内のすべてのルータは、エリアが NSSA であることを認識する必要があります。次のキーワードのいずれかを選択します。 • **no-redistribution**：ルータが NSSA ABR の場合、**redistribute** コマンドを使用して、ルートを NSSA エリアでなく通常のエリアに取り込む場合に使用します。 • **default-information-originate**：LSA タイプ 7 を NSSA に取り込めるようにする場合に、ABR で選択します。 • **no-redistribution**：サマリー LSA を NSSA に送信しない場合に選択します。

	コマンドまたはアクション	目的
ステップ 7	**area** *area-id* **range** *address mask* 例： `スイッチ(config-router)# area 1 range 255.240.0.0`	（任意）単一のルートをアドバタイズするアドレス範囲を指定します。このコマンドは、ABR に対してだけ使用します。
ステップ 8	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 9	**show ip ospf** [*process-id*] 例： `スイッチ# show ip ospf`	設定を確認するため、一般的な OSPF ルーティング プロセスまたは特定のプロセス ID に関する情報を表示します。
ステップ 10	**show ip ospf** [*process-id* [*area-id*]] **database** 例： `スイッチ# show ip osfp database`	特定のルータの OSPF データベースに関連する情報のリストを表示します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

その他の OSPF パラメータ

ルータ コンフィギュレーション モードで、その他の OSPF パラメータを設定することもできます。

- ルート集約：他のプロトコルからルートを再配信すると、各ルートは外部 LSA 内で個別にアドバタイズされます。OSPF リンクステートデータベースのサイズを小さくするには、**summary-address** ルータ コンフィギュレーション コマンドを使用し、指定されたネットワークアドレスおよびマスクに含まれる、再配信されたすべてのルートを単一のルータにアドバタイズします。
- 仮想リンク：OSPF では、すべてのエリアがバックボーン エリアに接続されている必要があります。バックボーンが不連続である場合に仮想リンクを確立するには、2 つの ABR を仮想リンクのエンドポイントとして設定します。設定情報には、他の仮想エンドポイント（他の ABR）の ID、および 2 つのルータに共通する非バックボーン リンク（通過エリア）などがあります。仮想リンクをスタブ エリアから設定できません。

- デフォルトルート：OSPFルーティング ドメイン内へのルート再配信を設定すると、ルータは自動的に自律システム境界ルータ（ASBR）になります。ASBR を設定し、強制的にOSPF ルーティング ドメインにデフォルト ルートを生成できます。
- すべての OSPF **show** 特権 EXEC コマンドでの表示にドメインネームサーバー（DNS）名を使用すると、ルータ IDやネイバー IDを指定して表示する場合に比べ、ルータを簡単に特定できます。
- デフォルト メトリック：OSPF は、インターフェイスの帯域幅に従ってインターフェイスの OSPF メトリックを計算します。メトリックは、帯域幅で分割された *ref-bw* として計算されます。ここでの *ref* のデフォルト値は 10 で、帯域幅（*bw*）は **bandwidth** インターフェイス コンフィギュレーション コマンドによって指定されます。大きな帯域幅を持つ複数のリンクの場合は、大きな数値を指定し、これらのリンクのコストを区別できます。
- アドミニストレーティブディスタンスは、ルーティング情報送信元の信頼性を表す数値です。0 ～ 255 の整数を指定でき、値が大きいほど信頼性は低下します。アドミニストレーティブディスタンスが 255 の場合はルーティング情報の送信元をまったく信頼できないため、無視する必要があります。OSPF では、エリア内のルート（エリア内）、別のエリアへのルート（エリア間）、および再配信によって学習した別のルーティング ドメインからのルート（外部）の 3 つの異なるアドミニストレーティブディスタンスが使用されます。どのアドミニストレーティブ ディスタンスの値でも変更できます。
- 受動インターフェイス：イーサネット上の 2 つのデバイス間のインターフェイスは 1 つのネットワーク セグメントしか表しません。このため、OSPF が送信側インターフェイスに hello パケットを送信しないようにするには、送信側デバイスを受動インターフェイスに設定する必要があります。両方のデバイスは受信側インターフェイス宛ての hello パケットを使用することで、相互の識別を可能にします。
- ルート計算タイマー：OSPF がトポロジ変更を受信してから SPF 計算を開始するまでの遅延時間、および 2 つの SPF 計算の間のホールド タイムを設定できます。
- ネイバー変更ログ：OSPF ネイバー ステートが変更されたときに Syslog メッセージを送信するようにルータを設定し、ルータの変更を詳細に表示できます。

その他の OSPF パラメータの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router ospf** *process-id* 例：	OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`スイッチ(config)# router ospf 10`	
ステップ 3	**summary-address** *address mask* 例： `スイッチ(config)# summary-address 10.1.1.1 255.255.255.0`	（任意）1 つのサマリー ルートだけがアドバタイズされるように、再配信されたルートのアドレスおよび IP サブネット マスクを指定します。
ステップ 4	**area** *area-id* **virtual-link** *router-id* [**hello-interval** *seconds*] [**retransmit-interval** *seconds*] [**trans**] [[**authentication-key** *key*] | **message-digest-key** *keyid* **md5** *key*]] 例： `スイッチ(config)# area 2 virtual-link 192.168.255.1 hello-interval 5`	（任意）仮想リンクを確立し、パラメータを設定します。
ステップ 5	**default-information originate** [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*] 例： `スイッチ(config)# default-information originate metric 100 metric-type 1`	（任意）強制的に OSPF ルーティング ドメインにデフォルト ルートを生成するように ASBR を設定します。パラメータはすべて任意です。
ステップ 6	**ip ospf name-lookup** 例： `スイッチ(config)# ip ospf name-lookup`	（任意）DNS 名検索を設定します。デフォルトでは無効になっています。
ステップ 7	**ip auto-cost reference-bandwidth** *ref-bw* 例： `スイッチ(config)# ip auto-cost reference-bandwidth 5`	（任意）単一のルートをアドバタイズするアドレス範囲を指定します。このコマンドは、ABR に対してだけ使用します。
ステップ 8	**distance ospf** {[**inter-area** *dist1*] [**inter-area** *dist2*] [**external** *dist3*]} 例： `スイッチ(config)# distance ospf inter-area 150`	（任意）OSPF の距離の値を変更します。各タイプのルートのデフォルト距離は 110 です。有効値は 1 ～ 255 です。
ステップ 9	**passive-interface** *type number* 例：	（任意）指定されたインターフェイス経由の hello パケットの送信を抑制します。

	コマンドまたはアクション	目的
	スイッチ(config)# passive-interface gigabitethernet 1/0/6	
ステップ 10	**timers throttle spf** *spf-delay spf-holdtime spf-wait* 例： スイッチ(config)# timers throttle spf 200 100 100	（任意）ルート計算タイマーを設定します。 • *spf-delay*：SPF 計算の変更を受信する間の遅延。指定できる範囲は1～600000ミリ秒です。 • *spf-holdtime*：最初と 2 番目の SPF 計算の間の遅延。指定できる範囲は 1 ～ 600000 ミリ秒です。 • *spf-wait*：SPF計算の最大待機時間（ミリ秒）。指定できる範囲は 1 ～ 600000 ミリ秒です。
ステップ 11	**ospf log-adj-changes** 例： スイッチ(config)# ospf log-adj-changes	（任意）ネイバー ステートが変更されたとき、syslog メッセージを送信します。
ステップ 12	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 13	**show ip ospf** [*process-id* [*area-id*]] **database** 例： スイッチ# show ip ospf database	特定のルータのOSPFデータベースに関連する情報のリストを表示します。
ステップ 14	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

LSA グループ ペーシング

OSPF LSA グループ ペーシング機能を使用すると、OSPF LSA をグループ化し、リフレッシュ、チェックサム、エージング機能の同期を取って、ルータをより効率的に使用できるようになります。デフォルトでこの機能はイネーブルとなっています。デフォルトのペーシング インターバルは 4 分間です。通常は、このパラメータを変更する必要はありません。最適なグループ ペーシング インターバルは、ルータがリフレッシュ、チェックサム、エージングを行う LSA 数に反比例します。たとえば、データベース内に約 10000 個の LSA が格納されている場合は、

ペーシング インターバルを短くすると便利です。小さなデータベース（40 ～ 100 LSA）を使用する場合は、ペーシング インターバルを長くし、10 ～ 20 分に設定してください。

LSA グループ ペーシングの変更

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router ospf** *process-id* 例： `スイッチ(config)# router ospf 25`	OSPF ルーティングをイネーブルにし、ルータ コンフィギュレーション モードを開始します。
ステップ 3	**timers lsa-group-pacing** *seconds* 例： `スイッチ(config-router)# timers lsa-group-pacing 15`	LSA の グループ ペーシングを変更します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ループバック インターフェイス

OSPF は、インターフェイスに設定されている最大の IP アドレスをルータ ID として使用します。このインターフェイスがダウンした場合、または削除された場合、OSPF プロセスは新し

いルータ ID を再計算し、すべてのルーティング情報をそのルータのインターフェイスから再送信します。ループバック インターフェイスが IP アドレスによって設定されている場合、他のインターフェイスにより大きなIPアドレスがある場合でも、OSPFはこのIPアドレスをルータ ID として使用します。ループバック インターフェイスに障害は発生しないため、安定性は増大します。OSPF は他のインターフェイスよりもループバック インターフェイスを自動的に優先し、すべてのループバック インターフェイスの中で最大の IP アドレスを選択します。

ループバック インターフェイスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface loopback 0** 例： `スイッチ(config)# interface loopback 0`	ループバック インターフェイスを作成し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**ip address** address mask 例： `スイッチ(config-if)# ip address 10.1.1.5 255.255.240.0`	このインターフェイスに IP アドレスを割り当てます。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show ip interface** 例： `スイッチ# show ip interface`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

OSPF のモニタリング

IP ルーティング テーブルの内容、キャッシュの内容、およびデータベースの内容など、特定の統計情報を表示できます。

表 *106 : IP OSPF* 統計情報の表示コマンド

show ip ospf [*process-id*]	OSPF ルーティング プロセスに関する一般情報を表示します。
show ip ospf [*process-id*] **database** [**router**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**router**] [**self-originate**] **show ip ospf** [*process-id*] **database** [**router**] [**adv-router** [*ip-address*]] **show ip ospf** [*process-id*] **database** [**network**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**asbr-summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**external**] [*link-state-id*] **show ip ospf** [*process-id area-id*] **database** [**database-summary**]	OSPFデータベースに関連する情報のリストを表示します。
show ip ospf border-routes	内部の OSPF ルーティング ABR および ASBR テーブル エントリを表示します。
show ip ospf interface [*interface-name*]	OSPFに関連するインターフェイス情報を表示します。
show ip ospf neighbor [*interface-name*] [*neighbor-id*] **detail**	OSPF インターフェイス ネイバー情報を表示します。
show ip ospf virtual-links	OSPF に関連する仮想リンク情報を表示します。

EIGRP に関する情報

EIGRP は IGRP のシスコ独自の拡張バージョンです。EIGRP は IGRP と同じディスタンス ベクトル アルゴリズムおよび距離情報を使用しますが、EIGRP では収束性および動作効率が大幅に改善されています。

コンバージェンステクノロジーには、拡散更新アルゴリズム（DUAL）と呼ばれるアルゴリズムが採用されています。DUAL を使用すると、ルート計算の各段階でループが発生しなくなり、トポロジの変更に関連するすべてのデバイスを同時に同期できます。トポロジ変更の影響を受けないルータは、再計算に含まれません。

IP EIGRP を導入すると、ネットワークの幅が広がります。RIP の場合、ネットワークの最大幅は 15 ホップです。EIGRP メトリックは数千ホップをサポートするほど大きいため、ネットワークを拡張するときに問題となるのは、トランスポート レイヤのホップ カウンタだけです。IP パケットが 15 台のルータを経由し、宛先方向のネクスト ホップが EIGRP によって取得されている場合だけ、EIGRP は転送制御フィールドの値を増やします。RIP ルートを宛先へのネクスト ホップとして使用する場合、転送制御フィールドでは、通常どおり値が増加します。

EIGRP の機能

EIGRP には次の機能があります。

- 高速コンバージェンス
- 差分更新：宛先のステートが変更された場合、ルーティング テーブルの内容全体を送信する代わりに差分更新を行い、EIGRP パケットに必要な帯域幅を最小化します。
- 低い CPU 使用率：完全更新パケットを受信ごとに処理する必要がないため、CPU 使用率が低下します。
- プロトコルに依存しないネイバー探索メカニズム：このメカニズムを使用し隣接ルータに関する情報を取得します。
- 可変長サブネット マスク（VLSM）
- 任意のルート集約
- 大規模ネットワークへの対応

EIGRP コンポーネント

EIGRP には次に示す 4 つの基本コンポーネントがあります。

- ネイバー探索および回復：直接接続されたネットワーク上の他のルータに関する情報を動的に取得するために、ルータで使用されるプロセスです。また、ネイバーが到達不能または動作不能になっていることを検出するためにも使用されます。ネイバー探索および回復は、サイズの小さな hello パケットを定期的に送信することにより、わずかなオーバーヘッ

ドで実現されます。hello パケットが受信されているかぎり、Cisco ISO ソフトウェアは、ネイバーが有効に機能していると学習します。このように判別された場合、隣接ルータはルーティング情報を交換できます。

- Reliable Transport Protocol：EIGRP パケットをすべてのネイバーに確実に、順序どおりに配信します。マルチキャスト パケットとユニキャスト パケットが混在した伝送もサポートされます。EIGRP パケットには確実に送信する必要があるものと、そうでないものがあります。効率化のため、信頼性は必要時にのみ提供されます。たとえば、マルチキャスト機能があるマルチアクセス ネットワーク（イーサネットなど）では、すべてのネイバーにそれぞれ hello パケットを確実に送信する必要はありません。そのため、EIGRP は、1 つのマルチキャスト hello を送信し、パケットに確認応答が必要ないという通知をそのパケットに含めます。他のタイプのパケット（アップデートなど）の場合は、確認応答（ACK パケット）を要求します。信頼性の高い伝送であれば、ペンディング中の未確認応答パケットがある場合、マルチキャストパケットを迅速に送信できます。このため、リンク速度が変化する場合でも、コンバージェンス時間を短く保つことができます。

- DUAL 有限状態マシンには、すべてのルート計算の決定プロセスが組み込まれており、すべてのネイバーによってアドバタイズされたすべてのルートが追跡されます。DUAL は距離情報（メトリックともいう）を使用して、効率的な、ループのないパスを選択し、さらに DUAL は適切な後継ルータに基づいて、ルーティング テーブルに挿入するルートを選択します。後継ルータは、宛先への最小コスト パス（ルーティング ループに関連しないことが保証されている）を持つ、パケット転送に使用される隣接ルータです。適切な後継ルータが存在しなくても、宛先にアドバタイズするネイバーが存在する場合は再計算が行われ、この結果、新しい後継ルータが決定されます。ルートの再計算に要する時間によって、コンバージェンス時間が変わります。再計算はプロセッサに負荷がかかるため、必要でない場合は、再計算しないようにしてください。トポロジが変更されると、DUAL はフィジブル サクセサの有無を調べます。適切なフィジブル サクセサが存在する場合は、それらを探して使用し、不要な再計算を回避します。

- プロトコル依存モジュールは、ネットワーク層プロトコル固有のタスクを実行します。たとえば、IP EIGRP モジュールは、IP でカプセル化された EIGRP パケットを送受信します。また、EIGRP パケットを解析したり、DUAL に受信した新しい情報を通知したりします。EIGRP は DUAL にルーティング決定を行うように要求しますが、結果は IP ルーティング テーブルに格納されます。EIGRP は、他の IP ルーティング プロトコルによって取得したルートの再配信も行います。

（注） EIGRP をイネーブルにするには、デバイスまたはアクティブ スイッチ上で が稼働している必要があります。

EIGRP の設定方法

EIGRP ルーティング プロセスを作成するには、EIGRP をイネーブルにし、ネットワークを関連付ける必要があります。EIGRP は指定されたネットワーク内のインターフェイスにアップ

デートを送信します。インターフェイス ネットワークを指定しないと、どのEIGRPアップデートでもアドバタイズされません。

（注） ネットワーク上に IGRP 用に設定されているルータがあり、この設定を EIGRP に変更する場合は、IGRP と EIGRP の両方が設定された移行ルータを指定する必要があります。この場合は、この次の項に記載されているステップ 1 ～ 3 を実行し、さらに「スプリット ホライゾンの設定」も参照してください。ルートを自動的に再配信するには、同じ AS 番号を使用する必要があります。

EIGRP のデフォルト設定

表 *107 : EIGRP* のデフォルト設定

機能	デフォルト設定
自動サマリー	ディセーブル。
デフォルト情報	再配信中は外部ルートが許可され、EIGRP プロセス間でデフォルト情報が渡されます。
デフォルト メトリック	デフォルト メトリックなしで再配信できるのは、接続されたルートおよびインターフェイスのスタティック ルートだけです。デフォルト メトリックは次のとおりです。 • 帯域幅：0 以上の kb/s • 遅延（10 マイクロ秒）：0 または 39.1 ナノ秒の倍数である任意の正の数値 • 信頼性：0 ～ 255 の任意の数値（255 の場合は信頼性が 100%） • 負荷：0 ～ 255 の数値で表される有効帯域幅（255 の場合は 100% の負荷） • MTU：バイトで表されたルートの MTU サイズ（0 または任意の正の整数）
ディスタンス	内部距離：90 外部距離：170
EIGRP の隣接関係変更ログ	ディセーブル。隣接関係の変更はロギングされません。
IP 認証キーチェーン	認証なし

機能	デフォルト設定
IP 認証モード	認証なし
IP 帯域幅比率	50%
IP hello 間隔	低速非ブロードキャスト マルチアクセス（NBMA）ネットワークの場合：60 秒、それ以外のネットワークの場合：5 秒
IP ホールドタイム	低速 NBMA ネットワークの場合：180 秒、それ以外のネットワークの場合：15 秒
IP スプリットホライズン	イネーブル。
IP サマリー アドレス	サマリー集約アドレスは未定義
メトリック重み	tos：0、k1 および k3：1、k2、k4、および k5：0
ネットワーク	指定なし
ノンストップ フォワーディング（NSF）認識	を実行するスイッチ上で IPv4 に対してイネーブルになっています。レイヤ 3 スイッチでは、ハードウェアやソフトウェアの変更中に、隣接する NSF 対応ルータからのパケットを転送し続けることができます。
NSF 対応	ディセーブル。 （注） デバイスは EIGRP NSF 対応ルーティングを IPv4 に対してサポートします。
オフセットリスト	ディセーブル。
ルータ EIGRP	ディセーブル。
メトリック設定	ルート マップにはメトリック設定なし
トラフィック共有	メトリックの比率に応じて配分
バリアンス	1（等コスト ロード バランシング）

EIGRP NSF

デバイス スタックは、次の 2 つのレベルの EIGRP ノンストップ フォワーディングをサポートします。

- EIGRP NSF 認識

- EIGRP NSF 対応

EIGRP NSF 認識

は、EIGRP NSF 認識を IPv4 に対してサポートしています。隣接ルータが NSF 対応である場合、レイヤ3デバイスでは、ルータに障害が発生してプライマリ RP がバックアップ RP によって引き継がれる間、または処理を中断させずにソフトウェアアップグレードを行うためにプライマリ RP を手動でリロードしている間、隣接ルータからパケットを転送し続けます。

この機能をディセーブルにできません。この機能の詳細については、『*Cisco IOS IP Routing Protocols Configuration Guide, Release 12.4*』の「EIGRP Nonstop Forwarding (NSF) Awareness」を参照してください。

EIGRP NSF 対応

は、EIGRP Cisco NSF ルーティングをサポートし、スタックのアクティブスイッチ切り替え後のコンバージェンスの時間短縮と、トラフィック損失低減を実現します。

は、EIGRP NSF 対応ルーティングも IPv4 に対してサポートし、アクティブスイッチ切り替え後のコンバージェンス向上と、トラフィック損失低減を実現します。EIGRP NSF 対応のアクティブスイッチが再起動したとき、または新しいアクティブスイッチが起動して NSF が再起動したとき、このデバイスにはネイバーが存在せず、トポロジテーブルは空の状態です。デバイス、デバイススタックに対するトラフィックを中断することなく、インターフェイスの起動、ネイバーの再取得、およびトポロジテーブルとルーティングテーブルの再構築を行う必要があります。EIGRP ピアルータは新しいアクティブスイッチから学習したルートを維持し、NSF の再起動処理の間トラフィックの転送を継続します。

ネイバーによる隣接リセットを防ぐために、新しいアクティブスイッチは EIGRP パケットヘッダーの新しい Restart（RS）ビットを使用して再起動を示します。これを受信したネイバーは、ピア リスト内のスタックと同期を取り、スタックとの隣接関係を維持します。続いてネイバーは、RS ビットがセットされているアクティブスイッチにトポロジテーブルを送信して、自身が NSF 認識デバイスであることおよび新しいアクティブスイッチを補助していることを示します。

スタックのピアネイバーの少なくとも 1 つが NSF 認識デバイスであれば、アクティブスイッチはアップデート情報を受信してデータベースを再構築します。各 NSF 認識ネイバーは、最後のアップデート パケットに End of Table（EOT）マーカーを付けて送信して、テーブル情報の最後であることを示します。アクティブスイッチは、EOT マーカーを受信したときにコンバージェンスを認識し、続いてアップデートの送信を始めます。アクティブスイッチがネイバーからすべての EOT マーカーを受信した場合、または NSF コンバージタイマーが期限切れになった場合、EIGRP は RIB にコンバージェンスを通知し、すべての NSF 認識ピアにトポロジテーブルをフラッディングします。

基本的な EIGRP パラメータの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router eigrp** autonomous-system 例： `スイッチ(config)# router eigrp 10`	EIGRP ルーティング プロセスをイネーブルにし、ルータ コンフィギュレーション モードを開始します。AS 番号によって他の EIGRP ルータへのルートを特定し、ルーティング情報をタグ付けします。
ステップ 3	**nsf** 例： `スイッチ(config-router)# nsf`	（任意）EIGRP NSF をイネーブルにします。アクティブスイッチとそのすべてのピアでこのコマンドを入力します。
ステップ 4	**network** network-number 例： `スイッチ(config-router)# network 192.168.0.0`	ネットワークを EIGRP ルーティング プロセスに関連付けます。EIGRP は指定されたネットワーク内のインターフェイスにアップデートを送信します。
ステップ 5	**eigrp log-neighbor-changes** 例： `スイッチ(config-router)# eigrp log-neighbor-changes`	（任意）EIGRP 隣接関係変更のロギングをイネーブルにし、ルーティング システムの安定性をモニターします。
ステップ 6	**metric weights** *tos k1 k2 k3 k4 k5* 例： `スイッチ(config-router)# metric weights 0 2 0 2 0 0`	（任意）EIGRP メトリックを調整します。デフォルト値はほとんどのネットワークで適切に動作するよう入念に設定されていますが、調整することも可能です。 **注意** メトリックを設定する作業は複雑です。熟練したネットワーク設計者の指導がない場合は、行わないでください。
ステップ 7	**offset-list** [*access-list number* | *name*] {**in** | **out**} *offset* [*type number*] 例： `スイッチ(config-router)# offset-list 21 out 10`	（任意）オフセットリストをルーティングメトリックに適用し、EIGRP によって取得したルートへの着信および発信メトリックを増加します。アクセスリストまたはインターフェイスを使用し、オフセット リストを制限できます。

	コマンドまたはアクション	目的
ステップ 8	**auto-summary** 例： スイッチ(config-router)# **auto-summary**	（任意）ネットワーク レベル ルートへのサブネット ルートの自動サマライズをイネーブルにします。
ステップ 9	**interface***interface-id* 例： スイッチ(config-router)#**interface gigabitethernet 1/0/1**	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ3インターフェイスを指定します。
ステップ 10	**ip summary-address eigrp** *autonomous-system-number address mask* 例： スイッチ(config-if)# **ip summary-address eigrp 1 192.168.0.0 255.255.0.0**	（任意）サマリー集約を設定します。
ステップ 11	**end** 例： スイッチ(config-if)#**end**	特権 EXEC モードに戻ります。
ステップ 12	**show ip protocols** 例： スイッチ# **show ip protocols**	入力を確認します。 NSF認識の場合、出力に次のように表示されます。 *** IP Routing is NSF aware *** EIGRP NSF enabled
ステップ 13	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

EIGRP インターフェイスの設定

インターフェイスごとに、他の EIGRP パラメータを任意で設定できます。

手順

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**interface***interface-id* 例： スイッチ(config)# interface gigabitethernet 1/0/1	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ3インターフェイスを指定します。
ステップ 3	**ip bandwidth-percent eigrp** *percent* 例： スイッチ(config-if)# ip bandwidth-percent eigrp 60	（任意）インターフェイスで EIGRP が使用できる帯域幅の割合を設定します。デフォルト値は 50% です。
ステップ 4	**ip summary-address eigrp** *autonomous-system-number address mask* 例： スイッチ(config-if)# ip summary-address eigrp 109 192.161.0.0 255.255.0.0	（任意）指定されたインターフェイスのサマリー集約アドレスを設定します（auto-summary がイネーブルの場合は、通常設定する必要はありません）。
ステップ 5	**ip hello-interval eigrp** *autonomous-system-number seconds* 例： スイッチ(config-if)# ip hello-interval eigrp 109 10	（任意）EIGRP ルーティング プロセスの hello 時間間隔を変更します。指定できる範囲は 1 ～ 65535 秒です。低速 NBMA ネットワークの場合のデフォルト値は 60 秒、その他のすべてのネットワークでは 5 秒です。
ステップ 6	**ip hold-time eigrp** *autonomous-system-number seconds* 例： スイッチ(config-if)# ip hold-time eigrp 109 40	（任意）EIGRP ルーティング プロセスのホールド時間間隔を変更します。指定できる範囲は 1 ～ 65535 秒です。低速 NBMA ネットワークの場合のデフォルト値は 180 秒、その他のすべてのネットワークでは 15 秒です。 **注意** ホールド タイムを調整する前に、シスコのテクニカルサポートにお問い合わせください。
ステップ 7	**no ip split-horizon eigrp** *autonomous-system-number* 例： スイッチ(config-if)# no ip split-horizon eigrp 109	（任意）スプリット ホライズンをディセーブルにし、ルート情報が情報元インターフェイスからルータによってアドバタイズされるようにします。

	コマンドまたはアクション	目的
ステップ 8	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 9	**show ip eigrp interface** 例： スイッチ# show ip eigrp interface	EIGRP がアクティブであるインターフェイス、およびそれらのインターフェイスに関連する EIGRP の情報を表示します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

EIGRP ルート認証の設定

EIGRP ルート認証を行うと、EIGRP ルーティング プロトコルからのルーティング アップデートに関する MD5 認証が可能になり、承認されていない送信元から無許可または問題のあるルーティング メッセージを受け取ることがなくなります。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** interface-id 例： スイッチ(config)# interface gigabitethernet 1/0/1	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。
ステップ 3	**ip authentication mode eigrp** *autonomous-system* **md5** 例： スイッチ(config-if)# ip authentication mode eigrp 104 md5	IP EIGRP パケットの MD5 認証をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 4	**ip authentication key-chain eigrp** *autonomous-system key-chain* 例： `スイッチ(config-if)# ip authentication key-chain eigrp 105 chain1`	IP EIGRP パケットの認証をイネーブルにします。
ステップ 5	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**key chain** *name-of-chain* 例： `スイッチ(config)# key chain chain1`	キーチェーンを識別し、キーチェーン コンフィギュレーション モードを開始します。ステップ 4 で設定した名前を指定します。
ステップ 7	**key** *number* 例： `スイッチ(config-keychain)# key 1`	キーチェーン コンフィギュレーション モードで、キー番号を識別します。
ステップ 8	**key-string** *text* 例： `スイッチ(config-keychain-key)# key-string key1`	キーチェーン コンフィギュレーション モードで、キー ストリングを識別します。
ステップ 9	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} 例： `スイッチ(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 2011 duration 7200`	（任意）キーを受信できる期間を指定します。 *start-time* および *end-time* 構文には、*hh:mm:ss Month date year* または *hh:mm:ss date Month year* のいずれかを使用できます。デフォルトは、デフォルトの *start-time* 以降、無制限です。指定できる最初の日付は 1993 年 1 月 1 日です。デフォルトの *end-time* および **duration** は **infinite** です。
ステップ 10	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} 例： `スイッチ(config-keychain-key)# send-lifetime 14:00:00 Jan 25 2011 duration 3600`	（任意）キーを送信できる期間を指定します。 *start-time* および *end-time* 構文には、*hh:mm:ss Month date year* または *hh:mm:ss date Month year* のいずれかを使用できます。デフォルトは、デフォルトの *start-time* 以降、無制限です。指定できる最初の日付は 1993 年 1 月 1 日です。デフォルトの *end-time* および **duration** は **infinite** です。

	コマンドまたはアクション	目的
ステップ 11	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 12	**show key chain** 例： スイッチ# show key chain	認証キーの情報を表示します。
ステップ 13	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

EIGRP スタブ ルーティング

EIGRP スタブ ルーティング機能は、エンド ユーザーの近くにルーテッド トラフィックを移動することでリソースの利用率を低減させます。

（注） deviceはアクセス レイヤで EIGRP スタブ ルーティングを使用することにより、ほかのタイプのルーティング アドバタイズメントの必要性を排除しています。

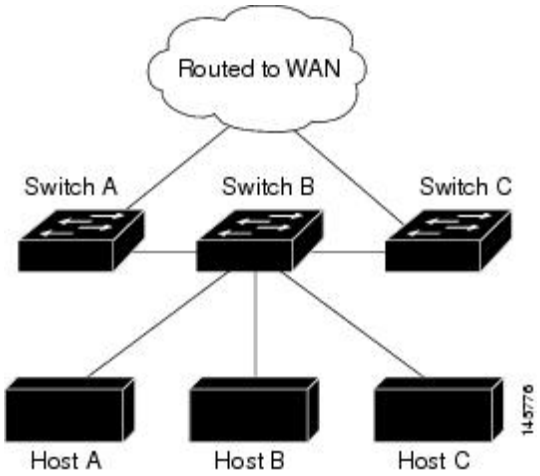
EIGRP スタブルーティングを使用するネットワークでは、ユーザーに対する IP トラフィックの唯一の許容ルートは、EIGRP スタブルーティングを設定しているdevice経由です。deviceは、ユーザーインターフェイスとして設定されているインターフェイスまたは他のデバイスに接続されているインターフェイスにルーテッドトラフィックを送信します。

EIGRP スタブルーティングを使用しているときは、EIGRP を使用してdeviceだけをスタブとして設定するように、分散ルータおよびリモートルータを設定する必要があります。指定したルートだけがdeviceから伝播されます。deviceは、サマリー、接続ルート、およびルーティングアップデートに対するすべてのクエリーに応答します。

スタブ ルータの状態を通知するパケットを受信した隣接ルータは、ルートについてはスタブルータに照会しません。また、スタブ ピアを持つルータは、そのピアについては照会しません。スタブ ルータは、ディストリビューション ルータを使用して適切なアップデートをすべてのピアに送信します。

次の図では、device B は EIGRP スタブルータとして設定されています。デバイス A および C は残りの WAN に接続されています。デバイス B は、接続ルート、スタティックルート、再配布ルート、およびサマリールートをデバイス A とデバイス C にアドバタイズします。 スイッチ B は、デバイス A から学習したルートをアドバタイズしません（逆の場合も同様です）。

図 *86 : EIGRP* スタブ ルータ設定



EIGRP のモニタリングおよびメンテナンス

ネイバー テーブルからネイバーを削除できます。さらに、各種 EIGRP ルーティング統計情報を表示することもできます。下の図に、ネイバーを削除し、統計情報を表示する特権 EXEC コマンドを示します。

表 *108 : IP EIGRP* の *clear* および *show* コマンド

clear ip eigrp neighbors [*if-address* | *interface*]	ネイバーテーブルからネイバーを削除します。
show ip eigrp interface [*interface*] [*as number*]	EIGRP に設定されているインターフェイスに関する情報を表示します。
show ip eigrp neighbors [*type-number*]	EIGRP によって検出されたネイバーを表示します。
show ip eigrp topology [*autonomous-system-number*] | [[*ip-address*] *mask*]]	指定されたプロセスの EIGRP トポロジテーブルを表示します。
show ip eigrp traffic [*autonomous-system-number*]	すべてまたは指定された EIGRP プロセスの送受信パケット数を表示します。

Multi-VRF CE に関する情報

バーチャル プライベート ネットワーク（VPN）は、ISP バックボーン ネットワーク上でお客様にセキュアな帯域幅共有を提供します。VPN は、共通ルーティング テーブルを共有するサイトの集合です。カスタマーサイトは、1 つまたは複数のインターフェイスでサービス プロバ

イダ ネットワークに接続され、サービス プロバイダは、VRF テーブルと呼ばれる VPN ルーティング テーブルと各インターフェイスを関連付けます。

スイッチが で稼働している場合、スイッチはカスタマーエッジ（CE）デバイスの Multiple VPN Routing/Forwarding（Multi-VRF）インスタンスをサポートします（Multi-VRF CE）。サービス プロバイダは、Multi-VRF CE により、重複する IP アドレスで複数の VPN をサポートできます。

（注） スイッチでは、VPN のサポートのためにマルチプロトコル ラベル スイッチング（MPLS）が使用されません。

Multi-VRF CE の概要

Multi-VRF CE は、サービス プロバイダが複数の VPN をサポートし、VPN 間で IP アドレスを重複して使用できるようにする機能です。Multi-VRF CE は入力インターフェイスを使用して、さまざまな VPN のルートを区別し、1 つまたは複数のレイヤ 3 インターフェイスと各 VRF を関連付けて仮想パケット転送テーブルを形成します。VRF 内のインターフェイスは、イーサネット ポートのように物理的なもの、または VLAN SVI のように論理的なものにもできますが、複数の VRF に属すことはできません。

（注） Multi-VRF CE インターフェイスは、レイヤ 3 インターフェイスである必要があります。

Multi-VRF CE には、次のデバイスが含まれます。

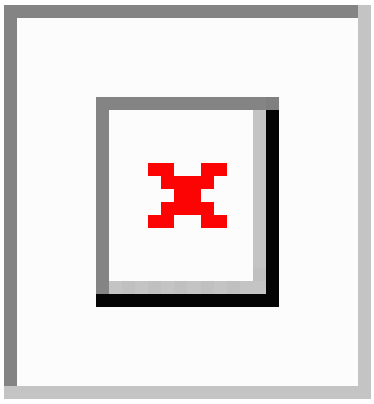
- お客様は、CE デバイスにより、1 つまたは複数のプロバイダ エッジ（PE）ルータへのデータ リンクを介してサービス プロバイダ ネットワークにアクセスできます。CE デバイスは、サイトのローカル ルートをルータにアドバタイズし、リモート VPN ルートをそこから学習します。スイッチを CE に設定することができます。
- PE ルータは、スタティック ルーティング、または BGP、RIPv2、OSPF、EIGRP などのルーティング プロトコルを使用して、CE デバイスとルーティング情報を交換します。PE は、直接接続している VPN に対する VPN ルートのみを保守する必要があります。そのため、すべてのサービス プロバイダ VPN ルートを PE が保守する必要はありません。各 PE ルータは、直接接続しているサイトごとに VRF を維持します。すべてのサイトが同じ VPN に存在する場合は、PE ルータの複数のインターフェイスを 1 つの VRF に関連付けることができます。各 VPN は、指定された VRF にマッピングされます。PE ルータは、ローカル VPN ルートを CE から学習したあとで、IBGP を使用して別の PE ルータと VPN ルーティング情報を交換します。
- CE デバイスに接続していないサービス プロバイダ ネットワークのルータは、プロバイダ ルータやコア ルータになります。

Multi-VRF CE では、複数のお客様が 1 つの CE を共有でき、CE と PE の間で 1 つの物理リンクだけが使用されます。共有 CE は、お客様ごとに別々の VRF テーブルを維持し、独自のルーティング テーブルに基づいて、お客様ごとにパケットをスイッチングまたはルーティングします。Multi-VRF CE は、制限付きの PE 機能を CE デバイスに拡張して、別々の VRF テーブルを維持し、VPN のプライバシーおよびセキュリティをブランチ オフィスに拡張します。

ネットワーク トポロジ

次の図に、スイッチを複数の仮想 CE として使用した構成例を示します。このシナリオは、中小企業など、VPN サービスの帯域幅要件の低いお客様に適しています。この場合、スイッチにはマルチ VRF CE のサポートが必要です。Multi-VRF CE はレイヤ 3 機能なので、VRF のそれぞれのインターフェイスはレイヤ 3 インターフェイスである必要があります。

図 *87*：複数の仮想 *CE* として機能するスイッチ



CE スイッチは、レイヤ3インターフェイスを VRF に追加するコマンドを受信すると、Multi-VRF CE 関連のデータ構造で VLAN ID と Policy Label（PL）の間に適切なマッピングを設定し、VLAN ID と PL を VLAN データベースに追加します。

Multi-VRF CE を設定すると、レイヤ 3 フォワーディング テーブルは、次の 2 つのセクションに概念的に分割されます。

- Multi-VRF CE ルーティング セクションには、さまざまな VPN からのルートが含まれます。
- グローバル ルーティング セクションには、インターネットなど、VPN 以外のネットワークへのルートが含まれます。

さまざまな VRF の VLAN ID はさまざまな PL にマッピングされ、処理中に VRF を区別するために使用されます。レイヤ 3 設定機能では、学習した新しい VPN ルートごとに、入力ポートの VLAN ID を使用して PL を取得し、Multi-VRF CE ルーティング セクションに PL および新しいルートを挿入します。ルーテッド ポートからパケットを受信した場合は、ポート内部 VLAN ID 番号が使用されます。SVI からパケットを受信した場合は、VLAN 番号が使用されます。

パケット転送処理

Multi-VRF CE 対応ネットワークのパケット転送処理は次のとおりです。

- スイッチは、VPN からパケットを受信すると、入力 PL 番号に基づいてルーティング テーブルを検索します。ルートが見つかると、スイッチはパケットを PE に転送します。
- 入力 PE は、CE からパケットを受信すると、VRF 検索を実行します。ルートが見つかると、ルータは対応する MPLS ラベルをパケットに追加し、MPLS ネットワークに送信します。
- 出力 PE は、ネットワークからパケットを受信すると、ラベルを除去してそのラベルを使用し、正しい VPN ルーティング テーブルを識別します。次に、通常のルート検索を実行します。ルートが見つかると、パケットを正しい隣接デバイスに転送します。
- CE は、出力 PE からパケットを受信すると、入力 PL を使用して正しい VPN ルーティング テーブルを検索します。ルートが見つかると、パケットを VPN 内で転送します。

ネットワーク コンポーネント

VRF を設定するには、VRF テーブルを作成し、VRF に関連するレイヤ 3 インターフェイスを指定します。次に、VPN、および CE と PE 間でルーティング プロトコルを設定します。プロバイダのバックボーンで VPN ルーティング情報を配信する場合は、BGP が優先ルーティング プロトコルです。Multi-VRF CE ネットワークには、次の 3 つの主要コンポーネントがあります。

- VPN ルート ターゲット コミュニティ：VPN コミュニティのその他すべてのメンバのリスト。VPN コミュニティ メンバーごとに VPN ルート ターゲットを設定する必要があります。
- VPN コミュニティ PE ルータのマルチプロトコル BGP ピアリング：VPN コミュニティのすべてのメンバーに VRF 到達可能性情報を伝播します。VPN コミュニティのすべての PE ルータで BGP ピアリングを設定する必要があります。
- VPN 転送：VPN サービス プロバイダ ネットワークを介し、全 VPN コミュニティ メンバー間で、全トラフィックを伝送します。

VRF 認識サービス

IP サービスはグローバル インターフェイスに設定可能で、グローバル ルーティング インスタンスで稼働します。IP サービスは複数のルーティング インスタンス上で稼働するように拡張されます。これが、VRF 認識です。システム内の任意の設定済み VRF であればいずれも、VRF 認識サービス用に指定できます。

VRF 認識サービスは、プラットフォームに依存しないモジュールに実装されます。VRF とは、Cisco IOS 内の複数のルーティング インスタンスを意味します。各プラットフォームには、サポートする VRF 数に関して独自の制限があります。

VRF 認識サービスには、次の特性があります。

- ユーザーは、ユーザー指定の VRF 内のホストに ping を実行できます。
- ARP エントリは、個別の VRF で学習されます。ユーザーは、特定の VRF の ARP エントリを表示できます。

Multi-VRF CE の設定方法

Multi-VRF CE のデフォルト設定

表 *109 : VRF* のデフォルト設定

機能	デフォルト設定
VRF	ディセーブル。VRF は定義されていません。
マップ	インポート マップ、エクスポート マップ、ルート マップは定義されていません。
VRF 最大ルート数	ファスト イーサネット スイッチ：8000 ギガビット イーサネット スイッチ：12000
転送テーブル	インターフェイスのデフォルトは、グローバル ルーティング テーブルです。

Multi-VRF CE の設定時の注意事項

（注）

Multi-VRF CE を使用するには、スイッチで をイネーブルにする必要があります。

- Multi-VRF CE を含むスイッチは複数のお客様によって共有され、各お客様には独自のルーティング テーブルがあります。
- お客様は別々の VRF テーブルを使用するので、同じ IP アドレスを再利用できます。別々の VPN では IP アドレスの重複が許可されます。
- Multi-VRF CE では、複数のお客様が、PE と CE の間で同じ物理リンクを共有できます。複数の VLAN を持つトランク ポートでは、パケットがお客様間で分離されます。それぞれのお客様には独自の VLAN があります。
- Multi-VRF CE ではサポートされない MPLS-VRF 機能があります。ラベル交換、LDP 隣接関係、ラベル付きパケットはサポートされません。
- PE ルータの場合、Multi-VRF CE の使用と複数の CE の使用に違いはありません。図 41-6 では、複数の仮想レイヤ 3 インターフェイスが Multi-VRF CE デバイスに接続されています。
- スイッチでは、物理ポートか VLAN SVI、またはその両方の組み合わせを使用して、VRF を設定できます。SVI は、アクセス ポートまたはトランク ポートで接続できます。
- お客様は、別のお客様と重複しないかぎり、複数の VLAN を使用できます。お客様の VLAN は、スイッチに保存されている適切なルーティング テーブルの識別に使用される特定のルーティング テーブル ID にマッピングされます。
- スイッチは、1 つのグローバルネットワークおよび最大 25 の VRF をサポートします。
- CE と PE の間では、ほとんどのルーティング プロトコル（BGP、OSPF、RIP、およびスタティック ルーティング）を使用できます。ただし、次の理由から External BGP（EBGP）を使用することを推奨します。
  - BGP では、複数の CE とのやり取りに複数のアルゴリズムを必要としません。
  - BGP は、さまざまな管理者によって稼働するシステム間でルーティング情報を渡すように設計されています。
  - BGP では、ルートの属性を CE に簡単に渡すことができます。
- Multi-VRF CE は、パケットのスイッチング レートに影響しません。
- VPN マルチキャストはサポートされません。
- プライベート VLAN で VRF をイネーブルにできます（逆も同様です）。
- インターフェイスでポリシーベース ルーティング（PBR）がイネーブルになっている場合は、VRF をイネーブルにできません（逆も同様です）。
- インターフェイスで Web Cache Communication Protocol（WCCP）がイネーブルになっている場合は、VRF をイネーブルにできません（逆も同様です）。

VRF の設定

次の操作を行ってください。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip routing** 例： `スイッチ(config)# ip routing`	IP ルーティングを有効にします。
ステップ 3	**ip vrf** *vrf-name* 例： `スイッチ(config)# ip vrf vpn1`	VRF 名を指定し、VRF コンフィギュレーション モードを開始します。
ステップ 4	**rd** *route-distinguisher* 例： `スイッチ(config-vrf)# rd 100:2`	ルート識別子を指定して VRF テーブルを作成します。AS 番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。
ステップ 5	**route-target** {**export** | **import** | **both**} *route-target-ext-community* 例： `スイッチ(config-vrf)# route-target both 100:2`	指定された VRF のインポート、エクスポート、またはインポートおよびエクスポート ルート ターゲット コミュニティのリストを作成します。AS システム番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。*route-target-ext-community* は、ステップ 4 で入力した *route-distinguisher* と同一にする必要があります。
ステップ 6	**import map** *route-map* 例： `スイッチ(config-vrf)# import map importmap1`	（任意）VRF にルート マップを対応付けます。
ステップ 7	**interface** *interface-id* 例： `スイッチ(config-vrf)# interface gigabitethernet 1/0/1`	VRF に関連付けるレイヤ 3 インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスにはルーテッド ポートまたは SVI を設定できます。

	コマンドまたはアクション	目的
ステップ 8	**ip vrf forwarding** *vrf-name* 例： `スイッチ(config-if)# ip vrf forwarding vpn1`	VRF をレイヤ 3 インターフェイスに対応付けます。 （注） **ip vrf forwarding**が管理インターフェイスで有効になっている場合、アクセスポイントは加入しません。
ステップ 9	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] 例： `スイッチ# show ip vrf interfaces vpn1`	設定を確認します。設定した VRF に関する情報を表示します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VRF 認識サービスの設定

次のサービスは、VRF 認識です。

- ARP
- ping
- 簡易ネットワーク管理プロトコル（SNMP）
- ユニキャスト RPF（uRPF）
- Syslog
- traceroute
- FTP および TFTP

ARP 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**show ip arp vrf** *vrf-name* 例： `スイッチ# show ip arp vrf vpn1`	指定された VRF 内の ARP テーブルを表示します。

ping 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**ping vrf** *vrf-name* **ip-host** 例： `スイッチ# ping vrf vpn1 ip-host`	指定された VRF 内の ARP テーブルを表示します。

SNMP 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**snmp-server trap authentication vrf** 例： `スイッチ(config)# snmp-server trap authentication vrf`	VRF で、パケットに対して SNMP トラップをイネーブルにします。
ステップ 3	**snmp-server engineID remote** *host* **vrf** *vpn-instance engine-id string* 例： `スイッチ(config)# snmp-server engineID remote 172.16.20.3 vrf vpn1 80000009030000B064EFE100`	スイッチ上で、リモート SNMP エンジンの名前を設定します。

	コマンドまたはアクション	目的
ステップ 4	**snmp-server host** *host* **vrf** *vpn-instance* **traps** *community* 例： `スイッチ(config)# snmp-server host 172.16.20.3 vrf vpn1 traps comaccess`	SNMP トラップ動作の受信側、および SNMP トラップの送信に使用される VRF テーブルを指定します。
ステップ 5	**snmp-server host** *host* **vrf** *vpn-instance* **informs** *community* 例： `スイッチ(config)# snmp-server host 172.16.20.3 vrf vpn1 informs comaccess`	SNMP 通知動作の受信先を指定し、SNMP 通知の送信に使用される VRF テーブルを指定します。
ステップ 6	**snmp-server user** *user group* **remote** *host* **vrf** *vpn-instance security model* 例： `スイッチ(config)# snmp-server user abcd remote 172.16.20.3 vrf vpn1 priv v2c 3des secure3des`	SNMP アクセス用に、VRF 上にあるリモート ホストの SNMP グループにユーザーを追加します。
ステップ 7	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

HSRP 用 VRF 認識サービスの設定

VRF の HSRP サポートにより、HSRP 仮想 IP アドレスが、確実に適切な IP ルーティング テーブルに追加されます。

コマンドの完全な構文と使用方法については、このリリースに対応するスイッチ コマンド リファレンスおよび『*Cisco IOS Switching Services Command Reference, Release 12.4*』を参照してください。

手順

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

	コマンドまたはアクション	目的
ステップ 2	**interface***interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。
ステップ 3	**no switchport** 例： `スイッチ(config-if)# no switchport`	レイヤ 2 コンフィギュレーション モードからインターフェイスを削除します（物理インターフェイスの場合）。
ステップ 4	**ip vrf forwarding** *vrf-name* 例： `スイッチ(config-if)# ip vrf forwarding vpn1`	インターフェイス上で VRF を設定します。
ステップ 5	**ip address** *ip-address* 例： `スイッチ(config-if)# ip address 10.1.5.1`	インターフェイスの IP アドレスを入力します。
ステップ 6	**standby 1 ip** *ip-address* 例： `スイッチ(config-if)#standby 1 ip 10.1.1.254`	HSRP をイネーブルにして、仮想 IP アドレスを設定します。
ステップ 7	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

uRPF 用 VRF 認識サービスの設定

uRPF は、VRF に割り当てられたインターフェイス上で設定でき、送信元検索が VRF テーブルで実行されます。

手順

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

	コマンドまたはアクション	目的
ステップ 2	**interface***interface-id* 例： スイッチ(config)# interface gigabitethernet 1/0/1	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。
ステップ 3	**no switchport** 例： スイッチ(config-if)# no switchport	レイヤ 2 コンフィギュレーション モードからインターフェイスを削除します（物理インターフェイスの場合）。
ステップ 4	**ip vrf forwarding** *vrf-name* 例： スイッチ(config-if)# ip vrf forwarding vpn2	インターフェイス上で VRF を設定します。
ステップ 5	**ip address** *ip-address* 例： スイッチ(config-if)# ip address 10.1.5.1	インターフェイスの IP アドレスを入力します。
ステップ 6	**ip verify unicast reverse-path** 例： スイッチ(config-if)# ip verify unicast reverse-path	インターフェイス上で uRPF を有効にします。
ステップ 7	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

VRF 認識 RADIUS の設定

VRF 認識 RADIUS を設定するには、まず RADIUS サーバー上で AAA をイネーブルにする必要があります。『Per VRF AAA Feature Guide』で説明されているとおり、スイッチで **ip vrf forwarding** *vrf-name* サーバーグループ コンフィギュレーション コマンドと **ip radius source-interface** グローバル コンフィギュレーション コマンドがサポートされます。

syslog 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**logging on** 例： スイッチ(config)# logging on	ストレージ ルータ イベント メッセージのロギングを、イネーブルまたは一時的にディセーブルにします。
ステップ 3	**logging host** *ip-address* **vrf** *vrf-name* 例： スイッチ(config)# logging host 10.10.1.0 vrf vpn1	ロギング メッセージが送信される Syslog サーバーのホスト アドレスを指定します。
ステップ 4	**logging buffered** *logging buffered size* **debugging** 例： スイッチ(config)# logging buffered critical 6000 debugging	メッセージを内部バッファにロギングします。
ステップ 5	**logging trap debugging** 例： スイッチ(config)# logging trap debugging	Syslog サーバーに送信されるロギング メッセージを制限します。
ステップ 6	**logging facility** *facility* 例： スイッチ(config)# logging facility user	ロギング ファシリティにシステム ロギング メッセージを送信します。
ステップ 7	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

traceroute 用 VRF 認識サービスの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**traceroute vrf** *vrf-name ipaddress* 例： `スイッチ(config)# traceroute vrf vpn2 10.10.1.1`	宛先アドレスを取得する VPN VRF の名前を指定します。

FTP および TFTP 用 VRF 認識サービスの設定

FTP および TFTP を VRF 認識とするには、いくつかの FTP/TFTP CLI を設定する必要があります。たとえば、インターフェイスに付加される VRF テーブルを使用する場合、E1/0 であれば、ip tftp source-interface E1/0 コマンドまたは ip ftp source-interface E1/0 コマンドを設定して、特定のルーティング テーブルを使用するように TFTP または FTP サーバーに通知する必要があります。この例では、VRF テーブルが宛先 IP アドレスを検索するのに使用されます。これらの変更には下位互換性があり、既存の動作には影響を及ぼしません。つまり、VRF がそのインターフェイスに設定されていない場合でも、送信元インターフェイス CLI を使用して、特定のインターフェイスにパケットを送信できます。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip ftp source-interface** *interface-type interface-number* 例： `スイッチ(config)# ip ftp source-interface gigabitethernet 1/0/2`	FTP 接続の発信元 IP アドレスを指定します。
ステップ 3	**end** 例： `スイッチ(config)#end`	特権 EXEC モードに戻ります。
ステップ 4	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 5	**ip tftp source-interface** *interface-type interface-number* 例： `スイッチ(config)# ip tftp source-interface gigabitethernet 1/0/2`	TFTP 接続用の送信元 IP アドレスを指定します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

マルチキャスト VRF の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip routing** 例： `スイッチ(config)# ip routing`	IP ルーティング モードをイネーブルにします
ステップ 3	**ip vrf** *vrf-name* 例： `スイッチ(config)# ip vrf vpn1`	VRF 名を指定し、VRF コンフィギュレーション モードを開始します。
ステップ 4	**rd** *route-distinguisher* 例： `スイッチ(config-vrf)# rd 100:2`	ルート識別子を指定して VRF テーブルを作成します。AS 番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。
ステップ 5	**route-target** {**export** | **import** | **both**} *route-target-ext-community* 例： `スイッチ(config-vrf)# route-target import 100:2`	指定された VRF のインポート、エクスポート、またはインポートおよびエクスポート ルート ターゲット コミュニティのリストを作成します。AS システム番号と任意の番号（xxx:y）または IP アドレスと任意の番号（A.B.C.D:y）を入力します。

	コマンドまたはアクション	目的
		route-target-ext-community は、ステップ 4 で入力した *route-distinguisher* と同一にする必要があります。
ステップ 6	**import map** *route-map* 例： `スイッチ(config-vrf)# import map importmap1`	（任意）VRF にルート マップを対応付けます。
ステップ 7	**ip multicast-routing vrf** *vrf-name* **distributed** 例： `スイッチ(config-vrf)# ip multicast-routing vrf vpn1 distributed`	（任意）VRF テーブルでグローバル マルチキャスト ルーティングをイネーブルにします。
ステップ 8	**interface** *interface-id* 例： `スイッチ(config-vrf)# interface gigabitethernet 1/0/2`	VRF に関連付けるレイヤ 3 インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスはルーテッド ポートまたは SVI に設定できます。
ステップ 9	**ip vrf forwarding** *vrf-name* 例： `スイッチ(config-if)# ip vrf forwarding vpn1`	VRF をレイヤ 3 インターフェイスに対応付けます。
ステップ 10	**ip address** *ip-address* **mask** 例： `スイッチ(config-if)# ip address 10.1.5.1 255.255.255.0`	レイヤ 3 インターフェイスの IP アドレスを設定します。
ステップ 11	**ip pim sparse-dense mode** 例： `スイッチ(config-if)# ip pim sparse-dense mode`	VRF に関連付けられているレイヤ 3 インターフェイス上で、PIM をイネーブルにします。
ステップ 12	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 13	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] 例： `スイッチ# show ip vrf detail vpn1`	設定を確認します。設定した VRF に関する情報を表示します。

	コマンドまたはアクション	目的
ステップ 14	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

VPN ルーティング セッションの設定

VPN内のルーティングは、サポートされている任意のルーティングプロトコル（RIP、OSPF、EIGRP、BGP）、またはスタティックルーティングで設定できます。ここで説明する設定はOSPF のものですが、その他のプロトコルでも手順は同じです。

（注） VRF インスタンス内で EIGRP ルーティングプロセスが実行されるように設定するには、**autonomous-system** *autonomous-system-number* アドレス ファミリ コンフィギュレーション モード コマンドを入力して、自律システム番号を設定する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# configure terminal	グローバル設定モードを開始します。
ステップ 2	**router ospf** *process-id* **vrf** *vrf-name* 例： スイッチ(config)# router ospf 1 vrf vpn1	OSPF ルーティングをイネーブルにして VPN 転送テーブルを指定し、ルータ コンフィギュレーション モードを開始します。
ステップ 3	**log-adjacency-changes** 例： スイッチ(config-router)# log-adjacency-changes	（任意）隣接ステートの変更を記録します。これは、デフォルトの状態です。
ステップ 4	**redistribute bgp** *autonomous-system-number* **subnets** 例： スイッチ(config-router)# redistribute bgp 10 subnets	BGP ネットワークから OSPF ネットワークに情報を再配布するようにスイッチを設定します。

	コマンドまたはアクション	目的
ステップ 5	**network** *network-number* **area** *area-id* 例： `スイッチ(config-router)# network 1 area 2`	OSPF が動作するネットワーク アドレスとマスク、およびそのネットワーク アドレスのエリア ID を定義します。
ステップ 6	**end** 例： `スイッチ(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show ip ospf** *process-id* 例： `スイッチ# show ip ospf 1`	OSPF ネットワークの設定を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

BGP PE/CE ルーティング セッションの設定

手順

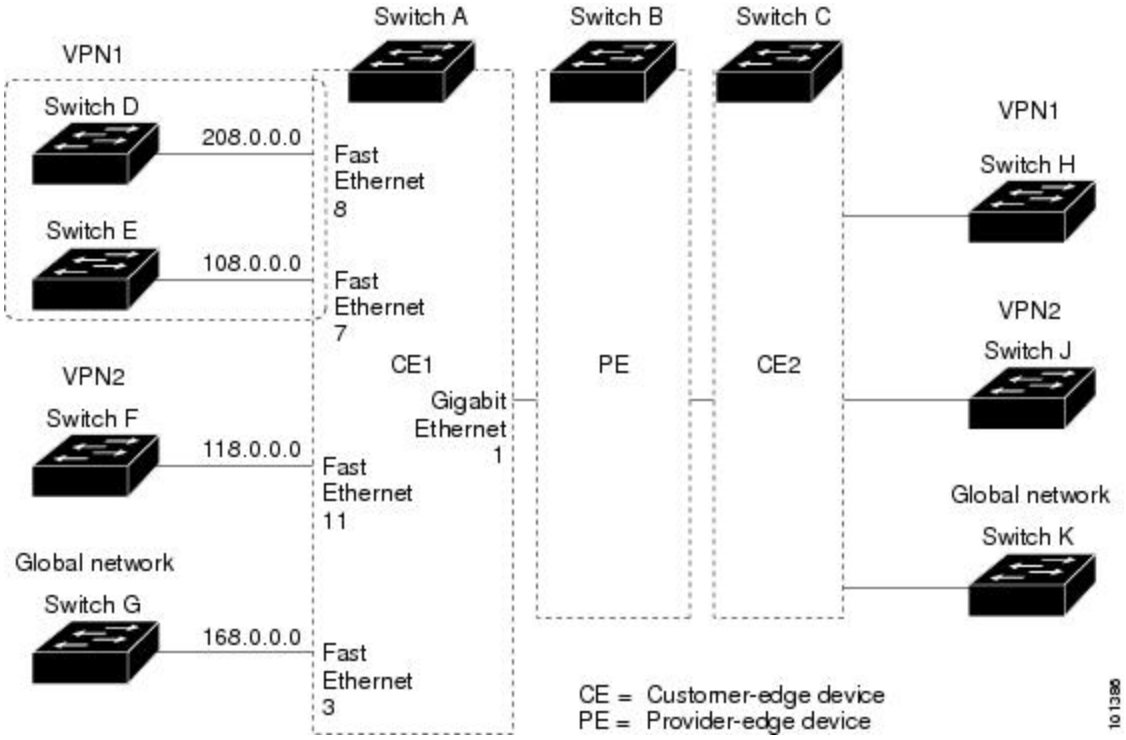
	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router bgp** *autonomous-system-number* 例： `スイッチ(config)# router bgp 2`	その他の BGP ルータに AS 番号を渡す BGP ルーティング プロセスを設定し、ルータ コンフィギュレーション モードを開始します。
ステップ 3	**network** *network-number* **mask** *network-mask* 例： `スイッチ(config-router)# network 5 mask 255.255.255.0`	BGP を使用してアナウンスするネットワークおよびマスクを指定します。

	コマンドまたはアクション	目的
ステップ 4	**redistribute ospf** *process-id* **match internal** 例： `スイッチ(config-router)# redistribute ospf 1 match internal`	OSPF 内部ルートを再配布するようにスイッチを設定します。
ステップ 5	**network** *network-number* **area** *area-id* 例： `スイッチ(config-router)# network 5 area 2`	OSPF が動作するネットワーク アドレスとマスク、およびそのネットワーク アドレスのエリア ID を定義します。
ステップ 6	**address-family ipv4 vrf** *vrf-name* 例： `スイッチ(config-router)# address-family ipv4 vrf vpn1`	PE/CE ルーティング セッションの BGP パラメータを定義し、VRF アドレス ファミリ モードを開始します。
ステップ 7	**neighbor** *address* **remote-as** *as-number* 例： `スイッチ(config-router)# neighbor 10.1.1.2 remote-as 2`	PE と CE ルータの間の BGP セッションを定義します。
ステップ 8	**neighbor** *address* **activate** 例： `スイッチ(config-router)# neighbor 10.2.1.1 activate`	IPv4 アドレス ファミリのアドバタイズメントをアクティブ化します。
ステップ 9	**end** 例： `スイッチ(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 10	**show ip bgp** [**ipv4**] [**neighbors**] 例： `スイッチ# show ip bgp ipv4 neighbors`	BGP 設定を確認します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

Multi-VRF CE の設定例

VPN1、VPN2、およびグローバル ネットワークで使用されるプロトコルは OSPF です。CE/PE 接続には BGP が使用されます。図のあとに続く出力は、スイッチを CE スイッチ A として設定する例、およびカスタマー スイッチ D と F の VRF 設定を示しています。CE スイッチ C とその他のカスタマー スイッチを設定するコマンドは含まれていませんが、内容は同様です。

図 *88 : Multi-VRF CE* の設定例



スイッチ A では、ルーティングをイネーブルにして VRF を設定します。

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# ip routing
スイッチ(config)# ip vrf v11
スイッチ(config-vrf)# rd 800:1
スイッチ(config-vrf)# route-target export 800:1
スイッチ(config-vrf)# route-target import 800:1
スイッチ(config-vrf)# exit
スイッチ(config)# ip vrf v12
スイッチ(config-vrf)# rd 800:2
スイッチ(config-vrf)# route-target export 800:2
スイッチ(config-vrf)# route-target import 800:2
スイッチ(config-vrf)# exit
```

スイッチ A のループバックおよび物理インターフェイスを設定します。ギガビット イーサネット ポート 1 は PE へのトランク接続です。ギガビット イーサネット ポート 8 と 11 は VPN に接続されます。

```
スイッチ(config)# interface loopback1
スイッチ(config-if)# ip vrf forwarding v11
スイッチ(config-if)# ip address 8.8.1.8 255.255.255.0
スイッチ(config-if)# exit

スイッチ(config)# interface loopback2
スイッチ(config-if)# ip vrf forwarding v12
スイッチ(config-if)# ip address 8.8.2.8 255.255.255.0
スイッチ(config-if)# exit

スイッチ(config)# interface gigabitethernet1/0/5
スイッチ(config-if)# switchport trunk encapsulation dot1q
スイッチ(config-if)# switchport mode trunk
スイッチ(config-if)# no ip address
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/8
スイッチ(config-if)# switchport access vlan 208
スイッチ(config-if)# no ip address
スイッチ(config-if)# exit
スイッチ(config)# interface gigabitethernet1/0/11
スイッチ(config-if)# switchport trunk encapsulation dot1q
スイッチ(config-if)# switchport mode trunk
スイッチ(config-if)# no ip address
スイッチ(config-if)# exit
```

スイッチ A で使用する VLAN を設定します。VLAN 10 は、CE と PE 間の VRF 11 によって使用されます。VLAN 20 は、CE と PE 間の VRF 12 によって使用されます。VLAN 118 と 208 は、それぞれスイッチ F とスイッチ D を含む VPN に使用されます。

```
スイッチ(config)# interface vlan10
スイッチ(config-if)# ip vrf forwarding v11
スイッチ(config-if)# ip address 38.0.0.8 255.255.255.0
スイッチ(config-if)# exit
スイッチ(config)# interface vlan20
スイッチ(config-if)# ip vrf forwarding v12
スイッチ(config-if)# ip address 83.0.0.8 255.255.255.0
スイッチ(config-if)# exit
スイッチ(config)# interface vlan118
スイッチ(config-if)# ip vrf forwarding v12
スイッチ(config-if)# ip address 118.0.0.8 255.255.255.0
スイッチ(config-if)# exit
スイッチ(config)# interface vlan208
スイッチ(config-if)# ip vrf forwarding v11
スイッチ(config-if)# ip address 208.0.0.8 255.255.255.0
スイッチ(config-if)# exit
```

VPN1 と VPN2 で OSPF ルーティングを設定します。

```
スイッチ(config)# router ospf 1 vrf vl1
スイッチ(config-router)# redistribute bgp 800 subnets
スイッチ(config-router)# network 208.0.0.0 0.0.0.255 area 0
スイッチ(config-router)# exit
```

```
スイッチ(config)# router ospf 2 vrf vl2
スイッチ(config-router)# redistribute bgp 800 subnets
スイッチ(config-router)# network 118.0.0.0 0.0.0.255 area 0
スイッチ(config-router)# exit
```

CE/PE ルーティングに BGP を設定します。

```
スイッチ(config)# router bgp 800
スイッチ(config-router)# address-family ipv4 vrf vl2
スイッチ(config-router-af)# redistribute ospf 2 match internal
スイッチ(config-router-af)# neighbor 83.0.0.3 remote-as 100
スイッチ(config-router-af)# neighbor 83.0.0.3 activate
スイッチ(config-router-af)# network 8.8.2.0 mask 255.255.255.0
スイッチ(config-router-af)# exit
スイッチ(config-router)# address-family ipv4 vrf vl1
スイッチ(config-router-af)# redistribute ospf 1 match internal
スイッチ(config-router-af)# neighbor 38.0.0.3 remote-as 100
スイッチ(config-router-af)# neighbor 38.0.0.3 activate
スイッチ(config-router-af)# network 8.8.1.0 mask 255.255.255.0
スイッチ(config-router-af)# end
```

スイッチ D は VPN 1 に属します。次のコマンドを使用して、スイッチ A への接続を設定します。

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# ip routing
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 208.0.0.20 255.255.255.0
スイッチ(config-if)# exit

スイッチ(config)# router ospf 101
スイッチ(config-router)# network 208.0.0.0 0.0.0.255 area 0
スイッチ(config-router)# end
```

スイッチ F は VPN 2 に属します。次のコマンドを使用して、スイッチ A への接続を設定します。

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# ip routing
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# switchport trunk encapsulation dot1q
スイッチ(config-if)# switchport mode trunk
スイッチ(config-if)# no ip address
スイッチ(config-if)# exit

スイッチ(config)# interface vlan118
スイッチ(config-if)# ip address 118.0.0.11 255.255.255.0
スイッチ(config-if)# exit

スイッチ(config)# router ospf 101
```

```
スイッチ(config-router)# network 118.0.0.0 0.0.0.255 area 0
スイッチ(config-router)# end
```

このコマンドをスイッチB（PEルータ）で使用すると、CEデバイス、スイッチAに対する接続だけが設定されます。

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip vrf v1
Router(config-vrf)# rd 100:1
Router(config-vrf)# route-target export 100:1
Router(config-vrf)# route-target import 100:1
Router(config-vrf)# exit

Router(config)# ip vrf v2
Router(config-vrf)# rd 100:2
Router(config-vrf)# route-target export 100:2
Router(config-vrf)# route-target import 100:2
Router(config-vrf)# exit
Router(config)# ip cef
Router(config)# interface Loopback1
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 3.3.1.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface Loopback2
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 3.3.2.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.10
Router(config-if)# encapsulation dot1q 10
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 38.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.20
Router(config-if)# encapsulation dot1q 20
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 83.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# router bgp 100
Router(config-router)# address-family ipv4 vrf v2
Router(config-router-af)# neighbor 83.0.0.8 remote-as 800
Router(config-router-af)# neighbor 83.0.0.8 activate
Router(config-router-af)# network 3.3.2.0 mask 255.255.255.0
Router(config-router-af)# exit
Router(config-router)# address-family ipv4 vrf vl
Router(config-router-af)# neighbor 38.0.0.8 remote-as 800
Router(config-router-af)# neighbor 38.0.0.8 activate
Router(config-router-af)# network 3.3.1.0 mask 255.255.255.0
Router(config-router-af)# end
```

Multi-VRF CE のモニタリング

表 *110 : Multi-VRF CE* 情報を表示するコマンド

show ip protocols vrf *vrf-name*	VRF に対応付けられたルーティング プロトコル情報を表示します。
show ip route vrf *vrf-name* [**connected**] [*protocol* [*as-number*]] [**list**] [**mobile**] [**odr**] [**profile**] [**static**] [**summary**] [**supernets-only**]	VRF に対応付けられた IP ルーティング テーブル情報を表示します。
show ip vrf [**brief** | **detail** | **interfaces**] [*vrf-name*]	定義された VRF インスタンスに関する情報を表示します。

ユニキャスト リバース パス転送の設定

ユニキャスト リバース パス転送（ユニキャスト RPF）機能は、検証可能な送信元 IP アドレスが不足している IP パケットを廃棄することで、間違ったまたは偽造（スプーフィングされた）送信元 IP アドレスがネットワークに流れて発生する問題を軽減するのに役立ちます。たとえば、Smurf や Tribal Flood Network（TFN）など、多くの一般的なタイプの DoS 攻撃は、偽造された、または次々に変わる送信元 IP アドレスを使用して、攻撃を突き止めたりフィルタすることを攻撃者が阻止できるようにします。パブリック アクセスを提供するインターネット サービス プロバイダ（ISP）の場合、uRPF が IP ルーティング テーブルと整合性の取れた有効な送信元アドレスを持つパケットだけを転送することによって、そのような攻撃をそらします。この処理により、ISP のネットワーク、その顧客、および残りのインターネットが保護されます。

（注）
- uRPF は、 でサポートされます。

プロトコル独立機能

この項では、IP ルーティング プロトコルに依存しない機能について説明します。これらの機能は、 フィーチャ セットが稼働するスイッチ上で使用できます。

分散型シスコ エクスプレス フォワーディング

シスコ エクスプレス フォワーディングに関する情報

シスコ エクスプレス フォワーディング（CEF）は、ネットワーク パフォーマンスを最適化するために使用されるレイヤ 3 IP スイッチング技術です。CEF には高度な IP 検索および転送アルゴリズムが実装されているため、レイヤ 3 スイッチングのパフォーマンスを最大化できま

す。高速スイッチング ルート キャッシュよりも CPU にかかる負担が少ないため、CEF はより多くの CPU 処理能力をパケット転送に割り当てることができます。スイッチ スタックでは、ハードウェアによって distributed CEF（dCEF）が使用されます。動的なネットワークでは、ルーティングの変更によって、高速スイッチング キャッシュ エントリが頻繁に無効になります。高速スイッチングキャッシュエントリが無効になると、トラフィックがルートキャッシュによって高速スイッチングされずに、ルーティング テーブルによってプロセス スイッチングされることがあります。CEFおよびdCEFは転送情報ベース（FIB）検索テーブルを使用して、宛先ベースの IP パケット スイッチングを実行します。

CEF および dCEF での 2 つの主要なコンポーネントは、分散 FIB と分散隣接テーブルです。

- FIB はルーティング テーブルや情報ベースと同様、IP ルーティング テーブルに転送情報のミラーイメージが保持されます。ネットワーク内でルーティングまたはトポロジが変更されると、IP ルーティング テーブルがアップデートされ、これらの変更が FIB に反映されます。FIBには、IPルーティングテーブル内の情報に基づいて、ネクストホップのアドレス情報が保持されます。FIBにはルーティングテーブル内の既知のルートがすべて格納されているため、CEF はルート キャッシュをメンテナンスする必要がなく、トラフィックのスイッチングがより効率化され、トラフィック パターンの影響も受けません。
- リンク層上でネットワーク内のノードが1ホップで相互に到達可能な場合、これらのノードは隣接関係にあると見なされます。CEF は隣接テーブルを使用し、レイヤ2アドレッシング情報を付加します。隣接テーブルには、すべての FIB エントリに対する、レイヤ2のネクストホップのアドレスが保持されます。

スイッチまたはスイッチスタックは、ギガビット速度の回線レート IP トラフィックを達成するため特定用途向け集積回路（ASIC）を使用しているので、CEF または dCEF 転送はソフトウェア転送パス（CPU により転送されるトラフィック）にだけ適用されます。

シスコ エクスプレス フォワーディングの設定方法

デフォルトで、CEF または dCEF はグローバルにイネーブルに設定されています。何らかの理由でこれが無効になった場合は、**ip cef**または **ip cef distributed** グローバル コンフィギュレーション コマンドを使用し、再度有効に設定できます。

デフォルト設定では、すべてのレイヤ3インターフェイスで CEF または dCEF がイネーブルです。**no ip route-cache cef** インターフェイス コンフィギュレーション コマンドを入力すると、ソフトウェアが転送するトラフィックに対してCEFが無効になります。このコマンドは、ハードウェア転送パスには影響しません。CEF を無効にして **debug ip packet detail** 特権 EXEC コマンドを使用すると、ソフトウェア転送トラフィックをデバッグするのに便利です。ソフトウェア転送パス用のインターフェイスでCEF を有効にするには、**ip route-cache cef** インターフェイス コンフィギュレーション コマンドを使用します。

⚠

注意 CLI には、インターフェイス上で CEF を無効にする **no ip route-cache cef** インターフェイス コンフィギュレーション コマンドが表示されますが、デバッグ以外の目的でインターフェイス上で CEF または dCEF を無効にしないようにしてください。

ディセーブルである CEF または dCEF をグローバルにイネーブルにしたり、ソフトウェア転送トラフィックのインターフェイス上でイネーブルにするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip cef** 例： `スイッチ(config)# ip cef`	非スタッキング スイッチで CEF の動作をイネーブルにします。 ステップ 4 に進みます。
ステップ 3	**ip cef distributed** 例： `スイッチ(config)# ip cef distributed`	アクティブ スイッチで CEF の動作をイネーブルにします。
ステップ 4	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するレイヤ 3 インターフェイスを指定します。
ステップ 5	**ip route-cache cef** 例： `スイッチ(config-if)# ip route-cache cef`	ソフトウェア転送トラフィック用のインターフェイスで CEF をイネーブルにします。
ステップ 6	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show ip cef** 例： `スイッチ# show ip cef`	すべてのインターフェイスの CEF ステータスを表示します。
ステップ 8	**show cef linecard** [**detail**] 例：	（任意）非スタッキング スイッチの CEF 関連インターフェイス情報を表示します。

	コマンドまたはアクション	目的
	`スイッチ# show cef linecard detail`	
ステップ 9	**show cef linecard** [*slot-number*] [**detail**] 例： `スイッチ# show cef linecard 5 detail`	（任意）スタック内のすべてのスイッチ、または指定されたスイッチに対して、スイッチの CEF 関連インターフェイス情報をスタック メンバ別に表示します。 （任意）*slot-number* には、スタック メンバーのスイッチ番号を入力します。
ステップ 10	**show cef interface** [*interface-id*] 例： `スイッチ# show cef interface gigabitethernet 1/0/1`	すべてのインターフェイスまたは指定されたインターフェイスの詳細な CEF 情報を表示します。
ステップ 11	**show adjacency** 例： `スイッチ# show adjacency`	CEF の隣接テーブル情報を表示します。
ステップ 12	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

等コスト ルーティング パスの個数

等コスト ルーティング パスに関する情報

同じネットワークへ向かう同じメトリックのルートが複数ルータに格納されている場合、これらのルートは等価コストを保有していると見なされます。ルーティング テーブルに複数の等コスト ルートが含まれる場合は、これらをパラレル パスと呼ぶこともあります。ネットワークへの等コスト パスがルータに複数格納されている場合、ルータはこれらを同時に使用できます。パラレル パスを使用すると、パスに障害が発生した場合に冗長性を確保できます。また、使用可能なパスにパケットの負荷を分散し、使用可能な帯域幅を有効利用することもできます。等コスト ルートは、スタック内の各スイッチでサポートされます。

等コスト ルートはルータによって自動的に取得、設定されますが、ルーティング テーブルの IP ルーティング プロトコルでサポートされるパラレル パスの最大数は制御可能です。スイッチ ソフトウェア では最大 32 の等コスト ルーティングが許可されていますが、スイッチ ハードウェアはルートあたり 17 パス以上は使用しません。

等コスト ルーティング パスの設定方法

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router** {**rip** | **ospf** | **eigrp**} 例： `スイッチ(config)# router eigrp`	ルータ コンフィギュレーション モードを開始します。
ステップ 3	**maximum-paths** *maximum* 例： `スイッチ(config-router)# maximum-paths 2`	プロトコル ルーティング テーブルのパラレル パスの最大数を設定します。指定できる範囲は 1 ～ 16 です。ほとんどの IP ルーティング プロトコルでデフォルトは 4 ですが、BGP の場合だけ 1 です。
ステップ 4	**end** 例： `スイッチ(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show ip protocols** 例： `スイッチ# show ip protocols`	*Maximum path* フィールドの設定を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

スタティック ユニキャスト ルート

スタティック ユニキャスト ルートに関する情報

スタティック ユニキャスト ルートは、特定のパスを通過して送信元と宛先間でパケットを送受信するユーザー定義のルートです。ルータが特定の宛先へのルートを構築できない場合、スタティック ルートは重要で、到達不能なすべてのパケットが送信される最終ゲートウェイを指定する場合に有効です。

ユーザーによって削除されるまで、スタティック ルートはスイッチに保持されます。ただし、アドミニストレーティブ ディスタンスの値を割り当て、スタティック ルートをダイナミック ルーティング情報で上書きできます。各ダイナミック ルーティング プロトコルには、デフォルトのアドミニストレーティブディスタンスが設定されています（表10を参照）。ダイナミック ルーティング プロトコルの情報でスタティック ルートを上書きする場合は、スタティック ルートのアドミニストレーティブディスタンスがダイナミック プロトコルのアドミニストレーティブ ディスタンスよりも大きな値になるように設定します。

表 *111*：ダイナミック ルーティング プロトコルのデフォルトのアドミニストレーティブ ディスタンス

ルートの送信元	デフォルト距離
接続されているインターフェイス	0
スタティック ルート	1
EIGRP サマリー ルート	5
内部 EIGRP	90
IGRP	100
OSPF	110
不明	225

インターフェイスを指し示すスタティックルートは、RIP、IGRP、およびその他のダイナミック ルーティング プロトコルを通してアドバタイズされます。**redistribute** スタティック ルータ コンフィギュレーション コマンドが、これらのルーティングプロトコルに対して指定されているかどうかは関係ありません。これらのスタティック ルートがアドバタイズされるのは、インターフェイスを指し示すスタティック ルートが接続された結果、静的な性質を失ったとルーティング テーブルで見なされるためです。ただし、network コマンドで定義されたネットワーク以外のインターフェイスに対してスタティックルートを定義する場合は、ダイナミック ルーティング プロトコルに **redistribute** スタティックコマンドを指定しない限り、ルートはアドバタイズされません。

インターフェイスがダウンすると、ダウンしたインターフェイスを経由するすべてのスタティック ルートが IP ルーティング テーブルから削除されます。転送ルータのアドレスとして指定されたアドレスへ向かう有効なネクストホップがスタティック ルート内に見つからない場合は、IP ルーティング テーブルからそのスタティック ルートも削除されます。

スタティック ユニキャスト ルートの設定

スタティック ユニキャスト ルートは、特定のパスを通過して送信元と宛先間でパケットを送受信するユーザー定義のルートです。ルータが特定の宛先へのルートを構築できない場合、スタティック ルートは重要で、到達不能なすべてのパケットが送信される最終ゲートウェイを指定する場合に有効です。

スタティック ルートを設定するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip route prefix mask** {*address* | *interface*} [*distance*] 例： `Device(config)# ip route prefix mask gigabitethernet 1/0/4`	スタティック ルートを確立します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show ip route** 例： `スイッチ# show ip route`	設定を確認するため、ルーティングテーブルの現在の状態を表示します。
ステップ 6	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

次のタスク

スタティックルートを削除するには、**no ip route** *prefix mask* {*address*| *interface*} グローバル コンフィギュレーション コマンドを使用します。ユーザーによって削除されるまで、スタティックルートはdeviceに保持されます。

デフォルトのルートおよびネットワーク

デフォルトのルートおよびネットワークに関する情報

ルータは、他のすべてのネットワークへのルートを学習できません。完全なルーティング機能を実現するには、一部のルータをスマートルータとして使用し、それ以外のルータのデフォルト ルートをスマート ルータ宛てに指定します（スマート ルータにはインターネットワーク全体のルーティング テーブルに関する情報が格納されます）。これらのデフォルト ルートは動的に学習できますが、ルータごとに設定することもできます。ほとんどのダイナミックな内部ルーティング プロトコルには、スマート ルータを使用してデフォルト情報を動的に生成し、他のルータに転送するメカニズムがあります。

指定されたデフォルト ネットワークに直接接続されたインターフェイスがルータに存在する場合は、そのデバイス上で動作するダイナミック ルーティング プロトコルによってデフォルト ルートが生成されます。RIP の場合は、疑似ネットワーク 0.0.0.0 がアドバタイズされます。

ネットワークのデフォルトを生成しているルータには、そのルータ自身のデフォルトルートも指定する必要があります。ルータが自身のデフォルトルートを生成する方法の 1 つは、適切なデバイスを経由してネットワーク 0.0.0.0 に至るスタティック ルートを指定することです。

ダイナミック ルーティング プロトコルによってデフォルト情報を送信するときは、特に設定する必要はありません。ルーティング テーブルは定期的にスキャンされ、デフォルト ルートとして最適なデフォルト ネットワークが選択されます。IGRP ネットワークでは、システムのデフォルト ネットワークの候補が複数存在する場合もあります。Cisco ルータでは、デフォルト ルートまたは最終ゲートウェイを設定するため、アドミニストレーティブ ディスタンスおよびメトリック情報を使用します。

ダイナミックなデフォルト情報がシステムに送信されない場合は、**ip default-network** グローバル コンフィギュレーション コマンドを使用し、デフォルトルートの候補を指定します。このネットワークが任意の送信元のルーティング テーブルに格納されている場合は、デフォルト ルートの候補としてフラグ付けされます。ルータにデフォルト ネットワークのインターフェイスが存在しなくても、そこへのパスが格納されている場合、そのネットワークは 1 つの候補と見なされ、最適なデフォルト パスへのゲートウェイが最終ゲートウェイになります。

デフォルトのルートおよびネットワークの設定方法

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例：** `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip default-network** *network number* **例：**	デフォルト ネットワークを指定します。

	コマンドまたはアクション	目的
	スイッチ(config)# ip default-network 1	
ステップ 3	**end** 例： スイッチ(config)# end	特権 EXEC モードに戻ります。
ステップ 4	**show ip route** 例： スイッチ# show ip route	最終ゲートウェイで選択されたデフォルト ルートを表示します。
ステップ 5	**copy running-config startup-config** 例： スイッチ# copy running-config startup-config	（任意）コンフィギュレーション ファイルに設定を保存します。

ルーティング情報を再配信するためのルート マップ

ルート マップの概要

スイッチでは複数のルーティング プロトコルを同時に実行し、ルーティング プロトコル間で情報を再配信できます。ルーティング プロトコル間での情報の再配信は、サポートされているすべての IP ベース ルーティング プロトコルに適用されます。

2 つのドメイン間で拡張パケット フィルタまたはルート マップを定義することにより、ルーティング ドメイン間でルートの再配信を条件付きで制御することもできます。**match** および **set** ルート マップ コンフィギュレーション コマンドは、ルートマップの条件部を定義します。**match** コマンドは、条件が一致する必要があることを指定しています。**set** コマンドは、ルーティングアップデートが match コマンドで定義した条件を満たす場合に行われる処理を指定します。再配布はプロトコルに依存しない機能ですが、**match** および **set** ルート マップ コンフィギュレーション コマンドの一部は特定のプロトコル固有のものです。

route-map コマンドのあとに、**match** コマンドおよび **set** コマンドをそれぞれ 1 つまたは複数指定します。**match** コマンドを指定しない場合は、すべて一致すると見なされます。**set** コマンドを指定しない場合、一致以外の処理はすべて実行されません。このため、少なくとも 1 つの **match** または **set** コマンドを指定する必要があります。

（注） **set** ルート マップ コンフィギュレーション コマンドを使用しないルートマップは、CPU に送信されるので、CPU の使用率が高くなります。

ルートマップステートメントは、**permit** または **deny** として識別することもできます。ステートメントが拒否としてマークされている場合、一致基準を満たすパケットは通常の転送チャネルを通じて送り返されます（宛先ベースルーティング）、ステートメントが許可としてマークされている場合は、一致基準を満たすパケットに set コマンドが適用されます。一致基準を満たさないパケットは、通常のルーティング チャネルを通じて転送されます。

ルート マップの設定方法

次に示すステップ 3 ～ 14 はそれぞれ任意ですが、少なくとも 1 つの **match** ルート マップ コンフィギュレーション コマンド、および 1 つの **set** ルート マップ コンフィギュレーション コマンドを入力する必要があります。

（注） キーワードは、ルート配信を制御する手順で定義されているものと同じです。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**route-map***map-tag* [**permit** | **deny**] [*sequence number*] 例： `スイッチ(config)# route-map rip-to-ospf permit 4`	再配信を制御するために使用するルート マップを定義し、ルート マップ コンフィギュレーション モードを開始します。 *map-tag*：ルート マップ用のわかりやすい名前を指定します。**redistribute**ルータ コンフィギュレーション コマンドはこの名前を使用して、このルートマップを参照します。複数のルート マップで同じマップ タグ名を共有できます。 （任意）**permit** が指定され、このルートマップの一致条件が満たされている場合は、set アクションの制御に従ってルートが再配信されます。**deny**が指定が指定されている場合、ルートは再配信されません。 *sequence number*（任意）：同じ名前によってすでに設定されているルート マップのリスト内で、新しいルート マップの位置を指定する番号です。
ステップ 3	**match as-path** *path-list-number* 例：	BGP AS パス アクセス リストと照合します。

	コマンドまたはアクション	目的
	`スイッチ(config-route-map)#match as-path 10`	
ステップ 4	**match community-list** *community-list-number* [**exact**] 例： `スイッチ(config-route-map)# match community-list 150`	BGP コミュニティ リストのマッチングを行います。
ステップ 5	**match ip address** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*] 例： `スイッチ(config-route-map)# match ip address 5 80`	名前または番号を指定し、標準アクセス リストと照合します。1 ～ 199 の整数を指定できます。
ステップ 6	**match metric** metric-value 例： `スイッチ(config-route-map)# match metric 2000`	指定されたルート メトリックと一致させます。*metric-value* には、0 ～ 4294967295 の値が指定された、EIGRP のメトリックを指定できます。
ステップ 7	**match ip next-hop** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*] 例： `スイッチ(config-route-map)# match ip next-hop 8 45`	指定されたアクセスリスト（番号1～199）のいずれかで送信される、ネクストホップのルータ アドレスと一致させます。
ステップ 8	**match tag** *tag value* [*...tag-value*] 例： `スイッチ(config-route-map)# match tag 3500`	1つまたは複数のルートタグ値からなるリスト内の指定されたタグ値と一致させます。0～4294967295 の整数を指定できます。
ステップ 9	**match interface***type number* [*...type-number*] 例： `スイッチ(config-route-map)# match interface gigabitethernet 1/0/1`	指定されたインターフェイスの1つから、指定されたネクスト ホップへのルートと一致させます。
ステップ 10	**match ip route-source** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*] 例： `スイッチ(config-route-map)# match ip route-source 10 30`	アドバタイズされた指定のアクセス リストによって指定したアドレスに一致します。

	コマンドまたはアクション	目的
ステップ 11	**match route-type** {**local** | **internal** | **external** [**type-1** | **type-2**]} 例： `スイッチ(config-route-map)# match route-type local`	指定された **route-type** と一致させます。 • **local**：ローカルに生成された BGP ルート。 • **internal**：OSPF エリア内およびエリア間ルート、または EIGRP 内部ルート。 • **external**：OSPF 外部ルート（タイプ 1 またはタイプ 2）または EIGRP 外部ルート。
ステップ 12	**set dampening** *halflife reuse suppress max-suppress-time* 例： `スイッチ(config-route-map)# set dampening 30 1500 10000 120`	BGP ルート ダンプニング係数を設定します。
ステップ 13	**set local-preference** *value* 例： `スイッチ(config-route-map)# set local-preference 100`	ローカル BGP パスに値を割り当てます。
ステップ 14	**set origin** {**igp** | **egp** *as* | **incomplete**} 例： `スイッチ(config-route-map)#set origin igp`	BGP 送信元コードを設定します。
ステップ 15	**set as-path** {**tag** | **prepend** *as-path-string*} 例： `スイッチ(config-route-map)# set as-path tag`	BGP の自律システム パスを変更します。
ステップ 16	**set level** {**level-1** | **level-2** | **level-1-2** | **stub-area** | **backbone**} 例： `スイッチ(config-route-map)# set level level-1-2`	ルーティング ドメインの指定エリアにアドバタイズされるルートのレベルを設定します。**stub-area** および **backbone** は、OSPF NSSA およびバックボーンエリアです。
ステップ 17	**set metric** *metric value* 例： `スイッチ(config-route-map)# set metric 100`	再配布されるルートを指定するためのメトリック値を設定します（EIGRP のみ）。*metric value* は -294967295 ～ 294967295 の整数です。
ステップ 18	**set metric***bandwidth delay reliability loading mtu* 例：	再配布されるルートを指定するためのメトリック値を設定します（EIGRP のみ）。

	コマンドまたはアクション	目的
	スイッチ(config-route-map)# set metric 10000 10 255 1 1500	• *bandwidth*：0 ～ 4294967295 の範囲のルートのメトリック値またはIGRP帯域幅（キロビット/秒単位）。 • *delay*：0 ～ 4294967295 の範囲のルート遅延（10 マイクロ秒単位）。 • *reliability*：0 ～ 255 の数値で表されるパケット伝送の成功可能性。255 は信頼性が 100% であること、0は信頼性がないことを意味します。 • *loading*：0 ～ 255 の数値で表されるルートの有効帯域幅（255 は 100% の負荷）。 • *mtu*：ルートの MTU の最小サイズ（バイト単位）。範囲は 0 ～ 4294967295 です。
ステップ 19	**set metric-type** {**type-1** | **type-2**} 例： スイッチ(config-route-map)# set metric-type type-2	再配信されるルートに OSPF 外部メトリック タイプを設定します。
ステップ 20	**set metric-type internal** 例： スイッチ(config-route-map)# set metric-type internal	ネクスト ホップの IGP メトリックと一致するように、EBGP ネイバーにアドバタイズされるプレフィックスのMulti-Exit識別子（MED）値を設定します。
ステップ 21	**set weight** *number* 例： スイッチ(config-route-map)# set weight 100	ルーティング テーブルの BGP 重みを設定します。指定できる値は 1 ～ 65535 です。
ステップ 22	**end** 例： スイッチ(config-route-map)# end	特権 EXEC モードに戻ります。
ステップ 23	**show route-map** 例： スイッチ# show route-map	設定を確認するため、設定されたすべてのルート マップ、または指定されたルート マップだけを表示します。
ステップ 24	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# copy running-config startup-config	

ルート配信の制御方法

次に示すステップ 3 ～ 14 はそれぞれ任意ですが、少なくとも 1 つの **match** ルート マップ コンフィギュレーション コマンド、および 1 つの **set** ルート マップ コンフィギュレーション コマンドを入力する必要があります。

（注） キーワードは、再配信用にルート マップを設定する手順で定義されているものと同じです。

ルーティング プロトコルのメトリックを、必ずしも別のルーティング プロトコルのメトリックに変換する必要はありません。たとえば、RIP メトリックはホップ カウントで、IGRP メトリックは5つの特性の組み合わせです。このような場合は、メトリックを独自に設定し、再配信されたルートに割り当てます。ルーティング情報を制御せずにさまざまなルーティング プロトコル間で交換するとルーティング ループが発生し、ネットワーク動作が著しく低下することがあります。

メトリック変換の代わりに使用されるデフォルトの再配信メトリックが定義されていない場合は、ルーティング プロトコル間で自動的にメトリック変換が発生することがあります。

- RIP はスタティック ルートを自動的に再配信できます。スタティック ルートにはメトリック 1（直接接続）が割り当てられます。
- デフォルト モードになっている場合、どのプロトコルも他のルーティング プロトコルを再配信できます。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例**： スイッチ# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router** { **rip** | **ospf** | **eigrp**} **例**： スイッチ(config)# router eigrp 10	ルータ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**redistribute** *protocol* [*process-id*] {**level-1** | **level-1-2** | **level-2**} [**metric** *metric-value*] [**metric-type** *type-value*] [**match internal** | **external** *type-value*] [**tag** *tag-value*] [**route-map** *map-tag*] [**weight** *weight*] [**subnets**] 例： `スイッチ(config-router)# redistribute eigrp 1`	ルーティング プロトコル間でルートを再配信します。route-map を指定しないと、すべてのルートが再配信されます。キーワード **route-map** に *map-tag* を指定しないと、ルートは配信されません。
ステップ 4	**default-metric** *number* 例： `スイッチ(config-router)# default-metric 1024`	現在のルーティング プロトコルが、再配信されたすべてのルートに対して同じメトリック値を使用するように設定します（RIP、OSPF）。
ステップ 5	**default-metric** bandwidth delay reliability loading mtu 例： `スイッチ(config-router)# default-metric 1000 100 250 100 1500`	EIGRP ルーティング プロトコルが、EIGRP 以外で再配信されたすべてのルートに対して同じメトリック値を使用するように設定します。
ステップ 6	**end** 例： `スイッチ(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show route-map** 例： `スイッチ# show route-map`	設定を確認するため、設定されたすべてのルートマップ、または指定されたルートマップだけを表示します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ポリシーベース ルーティング

ポリシーベース ルーティングの概要

PBR を使用すると、トラフィック フローに定義済みポリシーを設定できます。PBR を使用してルーティングをより細かく制御するには、ルーティング プロトコルから取得したルートの信頼度を小さくします。PBR は、次の基準に基づいて、パスを許可または拒否するルーティング ポリシーを設定したり、実装したりできます。

- 特定のエンド システムの ID

- アプリケーション
- プロトコル

PBR を使用すると、等価アクセスや送信元依存ルーティング、インタラクティブ対バッチ トラフィックに基づくルーティング、専用リンクに基づくルーティングを実現できます。たとえば、在庫記録を本社に送信する場合は高帯域で高コストのリンクを短時間使用し、電子メールなど日常的に使用するアプリケーションデータは低帯域で低コストのリンクで送信できます。

PBR が有効な場合は、アクセス コントロール リスト（ACL）を使用してトラフィックを分類し、各トラフィックがそれぞれ異なるパスを経由するようにします。PBR は着信パケットに適用されます。PBR が有効なインターフェイスで受信されたすべてのパケットは、ルート マップを通過します。ルートマップで定義された基準に基づいて、パケットは適切なネクストホップに転送（ルーティング）されます。

- 許可とマークされているルート マップ文は次のように処理されます。
  - match コマンドは長さまたは複数の ACL で照合できます。ルート マップ文には複数の match コマンドを含めることができます。論理関数またはアルゴリズム関数は、許可または拒否の決定がされるまで、すべての match コマンドで実行されます。

    次に例を示します。

    match length A B

    match ip address acl1 acl2

    match ip address acl3

  パケットは、match length A B または acl1 または acl2 または acl3 により許可される場合に許可されます。

  - 決定が許可の場合は、set コマンドで指定されたアクションがパケットで適用されます。
  - 下された決定が拒否の場合は、PBR アクション（set コマンドで指定された）が適用されません。代わりに、処理ロジックが、シーケンス内の次のルートマップ文（シーケンス番号が次に高い文）に移動します。次の文が存在しない場合は、PBR 処理が終了し、パケットがデフォルトの IP ルーティング テーブルを使用してルーティングされます。
- PBR では、拒否としてマークされているルートマップ ステートメントはサポートされません。

標準 IP ACL を使用すると、アプリケーション、プロトコル タイプ、またはエンド ステーションに基づいて一致基準を指定するように、送信元アドレスまたは拡張 IP ACL の一致基準を指定できます。一致が見つかるまで、ルートマップにこのプロセスが行われます。一致が見つからない場合、通常の宛先ベースルーティングが行われます。match ステートメント リストの末尾には、暗黙の拒否ステートメントがあります。

match 句が満たされた場合は、set 句を使用して、パス内のネクスト ホップ ルータを識別する IP アドレスを指定できます。

PBR の設定方法

- PBR を使用するには、スイッチまたはアクティブスタック上で フィーチャセットを有効にしておく必要があります。
- マルチキャスト トラフィックには、ポリシーによるルーティングが行われません。PBR が適用されるのはユニキャスト トラフィックだけです。
- ルーテッド ポートまたは SVI 上で、PBR を有効にできます。
- スイッチは一致長に基づき PBR をサポートします。
- レイヤ 3 モードの EtherChannel ポート チャネルにはポリシー ルート マップを適用できますが、EtherChannel のメンバーである物理インターフェイスには適用できません。適用しようとすると、コマンドが拒否されます。ポリシー ルート マップが適用されている物理インターフェイスは、EtherChannel のメンバーになることができません。
- スイッチまたはスイッチ スタックには最大 128 個の IP ポリシー ルート マップを定義できます。
- スイッチまたはスイッチ スタックには、PBR 用として最大 512 個のアクセス コントロール エントリ（ACE）を定義できます。
- ルート マップに一致基準を設定する場合は、次の注意事項に従ってください。
  - ローカル アドレス宛てのパケットを許可する ACL と照合させないでください。PBR がこれらのパケットを転送するため、ping または Telnet の失敗やルート プロトコルのフラッピングを発生させる可能性があります。
- VRF と PBR は、スイッチ インターフェイス上で相互に排他的です。PBR がインターフェイスで有効になっているときは、VRF を有効にはできません。その反対の場合も同じで、VRF がインターフェイスで有効になっているときは、PBR を有効にできません。
- PBR で使用されるハードウェア エントリ数は、ルート マップ自体、使用される ACL、ACL およびルート マップ エントリの順序によって異なります。
- TOS、DSCP、および IP Precedence に基づく PBR はサポートされません。
- set interface、set default next-hop、および set default interface はサポートされません。
- **ip next-hop recursive** および **ip next-hop verify availability** 機能は使用できません。next-hop は、直接接続される必要があります。
- set アクションのないポリシー マップはサポートされます。一致パケットは通常どおりにルーティングされます。
- match 句のないポリシー マップはサポートされます。set アクションはすべてのパケットに適用されます。

デフォルトでは、PBR はスイッチ上で無効です。PBR を有効にするには、一致基準および結果アクションを指定するルートマップを作成する必要があります。次に、特定のインターフェ

イスでそのルート マップ用の PBR を有効にします。指定したインターフェイスに着信したパケットのうち、match 句と一致したものはすべて PBR の対象になります。

スイッチで生成されたパケットまたはローカル パケットは、通常どおりにポリシー ルーティングされません。スイッチ上でローカル PBR をグローバルに有効にすると、そのスイッチから送信されたすべてのパケットがローカル PBR の影響を受けます。ローカル PBR は、デフォルトで無効に設定されています。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**route-map** *map-tag* [**permit**] [*sequence number*] 例： `スイッチ(config)# route-map pbr-map permit`	パケットの出力場所を制御するために使用するルート マップを定義し、ルート マップのコンフィギュレーション モードを開始します。 • *map-tag* － ：ルートマップ用のわかりやすい名前を指定します。 **ip policy route-map** インターフェイス コンフィギュレーション コマンドは、この名前を使用して、このルートマップを参照します。同じ map-tag がある複数の route-map 文は、1 つの route-map を定義します。 •（任意） **permit** － ： **permit** が指定され、このルートマップの一致条件が満たされている場合は、set アクションの制御に従ってルートがポリシールーティングされます。 •（任意） *sequence number* － ：シーケンス番号は、特定のルートマップ内の route-map ステートメントの位置を示します。
ステップ 3	**match ip address** {*access-list-number* | *access-list-name*} [*access-list-number* |*...access-list-name*] 例： `スイッチ(config-route-map)# match ip address 110 140`	1 つ以上の標準または拡張アクセス リストで許可されている送信元および宛先 IP アドレスを照合します。ACL は、複数の送信元および宛先 IP アドレスでも照合できます。 **match** コマンドを指定しない場合、ルートマップはすべてのパケットに適用されます。
ステップ 4	**match length** *min max* 例： `スイッチ(config-route-map)# match length 64 1500`	パケット長と照合します。

	コマンドまたはアクション	目的
ステップ 5	**set ip next-hop** *ip-address* [*...ip-address*] 例： `スイッチ(config-route-map)# set ip next-hop 10.1.6.2`	基準と一致するパケットの動作を指定します。パケットのルーティング先となるネクスト ホップを設定します（ネクスト ホップは隣接している必要があります）。
ステップ 6	**set ip next-hop verify-availability** [*next-hop-address sequence* **track** *object*] 例： `スイッチ(config-route-map)# set ip next-hop verify-availability 95.1.1.2.1 track 100`	ルートマップを設定し、トラッキング対象オブジェクトの到達可能性を確認します。 （注） このコマンドは、IPv6およびVRFではサポートされません。
ステップ 7	**exit** 例： `スイッチ(config-route-map)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 8	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するインタフェースを指定します。
ステップ 9	**ip policy route-map** *map-tag* 例： `スイッチ(config-if)# ip policy route-map pbr-map`	レイヤ 3 インターフェイス上で PBR を有効にし、使用するルート マップを識別します。1 つのインターフェイスに設定できるルートマップは、1つだけです。ただし、異なるシーケンス番号を持つ複数のルート マップ エントリを設定できます。これらのエントリは、最初の一致が見つかるまで、シーケンス番号順に評価されます。一致が見つからない場合、パケットは通常どおりにルーティングされます。
ステップ 10	**ip route-cache policy** 例： `スイッチ(config-if)# ip route-cache policy`	（任意）PBRの高速スイッチングを有効にします。PBR の高速スイッチングを有効にするには、PBR を有効にする必要があります。
ステップ 11	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 12	**ip local policy route-map** *map-tag* 例： `スイッチ(config)# ip local policy route-map local-pbr`	（任意）ローカル PBR を有効にして、スイッチから送信されるパケットに PBR を実行します。ローカル PBR は、スイッチによって生成されるパケットに適用されます。着信パケットには適用されません。

	コマンドまたはアクション	目的
ステップ 13	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 14	**show route-map** [*map-name*] 例： `スイッチ# show route-map`	（任意）設定を確認するため、設定されたすべてのルート マップ、または指定されたルート マップだけを表示します。
ステップ 15	**show ip policy** 例： `スイッチ# show ip policy`	（任意）インターフェイスに付加されたポリシー ルート マップを表示します。
ステップ 16	**show ip local policy** 例： `スイッチ# show ip local policy`	（任意）ローカル PBR が有効であるかどうか、および有効である場合は使用されているルート マップを表示します。

ルーティング情報のフィルタリング

ルーティング プロトコル情報をフィルタリングする場合は、以下の作業を実行します。

（注） OSPF プロセス間でルートが再配信される場合、OSPF メトリックは保持されません。

受動インターフェイスの設定

ローカルネットワーク上の他のルータが動的にルートを取得しないようにするには、**passive-interface** ルータ コンフィギュレーション コマンドを使用し、ルーティング アップデート メッセージがルータインターフェイスから送信されないようにします。OSPF プロトコルでこのコマンドを使用すると、パッシブに指定したインターフェイス アドレスが OSPF ドメインのスタブ ネットワークとして表示されます。OSPF ルーティング情報は、指定されたルータ インターフェイスから送受信されません。

多数のインターフェイスが存在するネットワークで、インターフェイスを手動でパッシブに設定する作業を回避するには、**passive-interface default** ルータ コンフィギュレーション コマンドを使用し、すべてのインターフェイスをデフォルトでパッシブになるように設定します。このあとで、隣接関係が必要なインターフェイスを手動で設定します。

パッシブとして有効にしたインターフェイスを確認するには、**show ip ospf interface** などのネットワークモニタリング用特権 EXEC コマンドを使用します。アクティブとして有効にしたインターフェイスを確認するには、**show ip interface** 特権 EXEC コマンドを使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router** { **rip** | **ospf** | **eigrp**} 例： `スイッチ(config)# router ospf`	ルータ コンフィギュレーション モードを開始します。
ステップ 3	**passive-interface** *interface-id* 例： `スイッチ(config-router)# passive-interface gigabitethernet 1/0/1`	指定されたレイヤ 3 インターフェイス経由のルーティング アップデートの送信を抑制します。
ステップ 4	**passive-interface default** 例： `スイッチ(config-router)# passive-interface default`	（任意）すべてのインターフェイスを、デフォルトでパッシブとなるように設定します。
ステップ 5	**no passive-interface** *interface type* 例： `スイッチ(config-router)# no passive-interface gigabitethernet1/0/3 gigabitethernet 1/0/5`	（任意）隣接関係を送信する必要があるインターフェイスだけをアクティブにします。
ステップ 6	**network** *network-address* 例： `スイッチ(config-router)# network 10.1.1.1`	（任意）ルーティング プロセス用のネットワーク リストを指定します。*network-address* は IP アドレスです。
ステップ 7	**end** 例： `スイッチ(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ルーティング アップデートのアドバタイズおよび処理の制御

アクセス制御リストと **distribute-list** ルータ コンフィギュレーション コマンドを組み合わせて使用すると、ルーティングアップデート中にルートのアドバタイズを抑制し、他のルータが 1 つまたは複数のルートを取得しないようにできます。この機能を OSPF で使用した場合は外部ルートにだけ適用されるため、インターフェイス名を指定できません。

distribute-list ルータ コンフィギュレーション コマンドを使用し、着信したアップデートのリストのうち特定のルートを処理しないようにすることもできます。（OSPF にこの機能は適用されません）。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例**： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router** { **rip** | **eigrp**} **例**： `スイッチ(config)# router eigrp 10`	ルータ コンフィギュレーション モードを開始します。
ステップ 3	**distribute-list** {*access-list-number* | *access-list-name*} **out** [*interface-name* | *routing process* | *autonomous-system-number*] **例**： `スイッチ(config-router)# distribute-list 120 out gigabitethernet 1/0/7`	アクセス リスト内のアクションに応じて、ルーティング アップデート内のルートのアドバタイズを許可または拒否します。
ステップ 4	**distribute-list** {*access-list-number* | *access-list-name*} **in** [*type-number*] **例**： `スイッチ(config-router)# distribute-list 125 in`	アップデートにリストされたルートの処理を抑制します。
ステップ 5	**end** **例**： `スイッチ(config-router)# end`	特権 EXEC モードに戻ります。
ステップ 6	**copy running-config startup-config** **例**： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ルーティング情報の送信元のフィルタリング

一部のルーティング情報が他の情報よりも正確な場合があるため、フィルタリングを使用して、さまざまな送信元から送られる情報にプライオリティを設定できます。「アドミニストレーティブディスタンス」は、ルータやルータのグループなど、ルーティング情報の送信元の信頼性を示す数値です。大規模ネットワークでは、他のルーティングプロトコルよりも信頼できるルーティング プロトコルが存在する場合があります。アドミニストレーティブ ディスタンスの値を指定すると、ルータはルーティング情報の送信元をインテリジェントに区別できるようになります。常にルーティング プロトコルのアドミニストレーティブ ディスタンスが最短（値が最小）であるルートが選択されます。

各ネットワークには独自の要件があるため、アドミニストレーティブディスタンスを割り当てる一般的な注意事項はありません。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**router** { **rip** | **ospf** | **eigrp**} 例： `スイッチ(config)# router eigrp 10`	ルータ コンフィギュレーション モードを開始します。
ステップ 3	**distance** *weight* {*ip-address* {*ip-address mask*}} [*ip access list*] 例： `スイッチ(config-router)# distance 50 10.1.5.1`	アドミニストレーティブ ディスタンスを定義します。 *weight*：アドミニストレーティブディスタンスは10 ～ 255 の整数です。単独で使用した場合、*weight* はデフォルトのアドミニストレーティブディスタンスを指定します。ルーティング情報の送信元に他の指定がない場合に使用されます。アドミニストレーティブディスタンスが255のルートはルーティングテーブルに格納されません。 （任意）*ip access list*：着信ルーティング アップデートに適用される IP 標準または IP 拡張アクセス リストです。
ステップ 4	**end** 例： `スイッチ(config-router)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 5	**show ip protocols** 例： `スイッチ# show ip protocols`	指定されたルーティング プロセス用のデフォルトのアドミニストレーティブ ディスタンスを表示します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

認証キーの管理

キー管理を使用すると、ルーティング プロトコルで使用される認証キーを制御できます。一部のプロトコルでは、キー管理を使用できません。認証キーは EIGRP および RIP バージョン 2 で使用できます。

前提条件

認証キーを管理する前に、認証をイネーブルにする必要があります。プロトコルに対して認証をイネーブルにする方法については、該当するプロトコルについての説明を参照してください。認証キーを管理するには、キー チェーンを定義してそのキー チェーンに属するキーを識別し、各キーの有効期間を指定します。各キーは、独自のキー識別子（**key** *number* キーチェーン コンフィギュレーション コマンドで指定されたもの）を保持し、ローカルに格納されています。キー ID、およびメッセージに関連付けられたインターフェイスの組み合わせにより、使用中の認証アルゴリズムおよび Message Digest 5（MD5）認証キーが一意に識別されます。

認証キーの設定方法

有効期間が指定された複数のキーを設定できます。存在する有効なキーの数にかかわらず、送信される認証パケットは1つだけです。最小の番号から順にキー番号が調べられ、最初に見つかった有効なキーが使用されます。キー変更中は、有効期間が重なっても問題ありません。これらの有効期間は、ルータに通知する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**key chain** *name-of-chain* 例：	キーチェーンを識別し、キーチェーン コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`スイッチ(config)# key chain key10`	
ステップ 3	**key** *number* 例： `スイッチ(config-keychain)# key 2000`	キー番号を識別します。有効値は 0 ～ 2147483647 です。
ステップ 4	**key-string** *text* 例： `スイッチ(config-keychain)# key-string Room 20, 10th floor`	キー ストリングを確認します。ストリングには 1 ～ 80 文字の大文字および小文字の英数字を指定できますが、最初の文字に数字を指定できません。
ステップ 5	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} 例： `スイッチ(config-keychain)# accept-lifetime 12:30:00 Jan 25 1009 infinite`	（任意）キーを受信できる期間を指定します。 *start-time* および *end-time* 構文には、*hh:mm:ss Month date year* または *hh:mm:ss date Month year* のいずれかを使用できます。デフォルトは、デフォルトの *start-time* 以降、無制限です。指定できる最初の日付は 1993 年 1 月 1 日です。デフォルトの *end-time* および **duration** は **infinite** です。
ステップ 6	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} 例： `スイッチ(config-keychain)# accept-lifetime 23:30:00 Jan 25 1019 infinite`	（任意）キーを送信できる期間を指定します。 *start-time* および *end-time* 構文には、*hh:mm:ss Month date year* または *hh:mm:ss date Month year* のいずれかを使用できます。デフォルトは、デフォルトの *start-time* 以降、無制限です。指定できる最初の日付は 1993 年 1 月 1 日です。デフォルトの *end-time* および **duration** は **infinite** です。
ステップ 7	**end** 例： `スイッチ(config-keychain)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show key chain** 例： `スイッチ# show key chain`	認証キーの情報を表示します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IP ネットワークのモニタリングおよびメンテナンス

特定のキャッシュ、テーブル、またはデータベースのすべての内容を削除できます。特定の統計情報を表示することもできます。

表 *112 : IP* ルートの削除またはルート ステータスの表示を行うコマンド

コマンド	目的
show ip route [*address* [*mask*] [**longer-prefixes**]]	ルーティング テーブルの現在の状態を表示します。
show ip route summary	サマリー形式でルーティング テーブルの現在のステータスを表示します。
show platform ip unicast	プラットフォームに依存する IP ユニキャストの情報を表示します。

第 48 章

ポリシーベースルーティング（PBR）の設定

ポリシーベース ルーティング

ポリシーベース ルーティングの概要

PBR を使用すると、トラフィック フローに定義済みポリシーを設定できます。PBR を使用してルーティングをより細かく制御するには、ルーティングプロトコルから取得したルートの信頼度を小さくします。PBR は、次の基準に基づいて、パスを許可または拒否するルーティングポリシーを設定したり、実装したりできます。

- 特定のエンド システムの ID
- アプリケーション
- プロトコル

PBR を使用すると、等価アクセスや送信元依存ルーティング、インタラクティブ対バッチ トラフィックに基づくルーティング、専用リンクに基づくルーティングを実現できます。たとえば、在庫記録を本社に送信する場合は高帯域で高コストのリンクを短時間使用し、電子メールなど日常的に使用するアプリケーションデータは低帯域で低コストのリンクで送信できます。

PBR が有効な場合は、アクセス コントロール リスト（ACL）を使用してトラフィックを分類し、各トラフィックがそれぞれ異なるパスを経由するようにします。PBR は着信パケットに適用されます。PBR が有効なインターフェイスで受信されたすべてのパケットは、ルート マップを通過します。ルートマップで定義された基準に基づいて、パケットは適切なネクストホップに転送（ルーティング）されます。

- 許可とマークされているルート マップ文は次のように処理されます。

- match コマンドは長さまたは複数の ACL で照合できます。ルート マップ文には複数の match コマンドを含めることができます。論理関数またはアルゴリズム関数は、許可または拒否の決定がされるまで、すべての match コマンドで実行されます。

  次に例を示します。

  match length A B

  match ip address acl1 acl2

  match ip address acl3

  パケットは、match length A B または acl1 または acl2 または acl3 により許可される場合に許可されます。

  - 決定が許可の場合は、set コマンドで指定されたアクションがパケットで適用されます。
  - 下された決定が拒否の場合は、PBR アクション（set コマンドで指定された）が適用されません。代わりに、処理ロジックが、シーケンス内の次のルートマップ文（シーケンス番号が次に高い文）に移動します。次の文が存在しない場合は、PBR 処理が終了し、パケットがデフォルトの IP ルーティング テーブルを使用してルーティングされます。

- PBR では、拒否としてマークされているルートマップ ステートメントはサポートされません。

標準 IP ACL を使用すると、アプリケーション、プロトコル タイプ、またはエンド ステーションに基づいて一致基準を指定するように、送信元アドレスまたは拡張 IP ACL の一致基準を指定できます。一致が見つかるまで、ルートマップにこのプロセスが行われます。一致が見つからない場合、通常の宛先ベースルーティングが行われます。match ステートメント リストの末尾には、暗黙の拒否ステートメントがあります。

match 句が満たされた場合は、set 句を使用して、パス内のネクスト ホップ ルータを識別する IP アドレスを指定できます。

PBR の設定方法

- PBR を使用するには、スイッチまたはアクティブスタック上で フィーチャセットを有効にしておく必要があります。
- マルチキャスト トラフィックには、ポリシーによるルーティングが行われません。PBR が適用されるのはユニキャスト トラフィックだけです。
- ルーテッド ポートまたは SVI 上で、PBR を有効にできます。
- スイッチは一致長に基づき PBR をサポートします。
- レイヤ 3 モードの EtherChannel ポート チャネルにはポリシー ルート マップを適用できますが、EtherChannel のメンバーである物理インターフェイスには適用できません。適用しようとすると、コマンドが拒否されます。ポリシー ルート マップが適用されている物理インターフェイスは、EtherChannel のメンバーになることができません。

- スイッチまたはスイッチ スタックには最大 128 個の IP ポリシー ルート マップを定義できます。
- スイッチまたはスイッチ スタックには、PBR 用として最大 512 個のアクセス コントロール エントリ（ACE）を定義できます。
- ルート マップに一致基準を設定する場合は、次の注意事項に従ってください。
  - ローカル アドレス宛てのパケットを許可する ACL と照合させないでください。PBR がこれらのパケットを転送するため、ping または Telnet の失敗やルート プロトコルのフラッピングを発生させる可能性があります。
- VRF と PBR は、スイッチ インターフェイス上で相互に排他的です。PBR がインターフェイスで有効になっているときは、VRF を有効にはできません。その反対の場合も同じで、VRF がインターフェイスで有効になっているときは、PBR を有効にできません。
- PBR で使用されるハードウェア エントリ数は、ルート マップ自体、使用される ACL、ACL およびルート マップ エントリの順序によって異なります。
- TOS、DSCP、および IP Precedence に基づく PBR はサポートされません。
- set interface、set default next-hop、および set default interface はサポートされません。
- **ip next-hop recursive** および **ip next-hop verify availability** 機能は使用できません。next-hop は、直接接続される必要があります。
- set アクションのないポリシー マップはサポートされます。一致パケットは通常どおりにルーティングされます。
- match 句のないポリシー マップはサポートされます。set アクションはすべてのパケットに適用されます。

デフォルトでは、PBR はスイッチ上で無効です。PBR を有効にするには、一致基準および結果アクションを指定するルートマップを作成する必要があります。次に、特定のインターフェイスでそのルート マップ用の PBR を有効にします。指定したインターフェイスに着信したパケットのうち、match 句と一致したものはすべて PBR の対象になります。

スイッチで生成されたパケットまたはローカル パケットは、通常どおりにポリシー ルーティングされません。スイッチ上でローカル PBR をグローバルに有効にすると、そのスイッチから送信されたすべてのパケットがローカル PBR の影響を受けます。ローカル PBR は、デフォルトで無効に設定されています。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** **例：** `スイッチ# configure terminal`	グローバル設定モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**route-map** *map-tag* [**permit**] [*sequence number*] 例： `スイッチ(config)# route-map pbr-map permit`	パケットの出力場所を制御するために使用するルート マップを定義し、ルート マップのコンフィギュレーション モードを開始します。 • *map-tag* － ：ルートマップ用のわかりやすい名前を指定します。 **ip policy route-map** インターフェイス コンフィギュレーション コマンドは、この名前を使用して、このルートマップを参照します。同じ map-tag がある複数の route-map 文は、1 つの route-map を定義します。 •（任意） **permit** － ： **permit** が指定され、このルートマップの一致条件が満たされている場合は、set アクションの制御に従ってルートがポリシールーティングされます。 •（任意） *sequence number* － ：シーケンス番号は、特定のルートマップ内の route-map ステートメントの位置を示します。
ステップ 3	**match ip address** {*access-list-number* | *access-list-name*} [*access-list-number* |*...access-list-name*] 例： `スイッチ(config-route-map)# match ip address 110 140`	1 つ以上の標準または拡張アクセス リストで許可されている送信元および宛先 IP アドレスを照合します。ACL は、複数の送信元および宛先 IP アドレスでも照合できます。 **match** コマンドを指定しない場合、ルートマップはすべてのパケットに適用されます。
ステップ 4	**match length** *min max* 例： `スイッチ(config-route-map)# match length 64 1500`	パケット長と照合します。
ステップ 5	**set ip next-hop** *ip-address* [*...ip-address*] 例： `スイッチ(config-route-map)# set ip next-hop 10.1.6.2`	基準と一致するパケットの動作を指定します。パケットのルーティング先となるネクスト ホップを設定します（ネクスト ホップは隣接している必要があります）。
ステップ 6	**set ip next-hop verify-availability** [*next-hop-address sequence* **track** *object*] 例： `スイッチ(config-route-map)# set ip next-hop verify-availability 95.1.1.2.1 track 100`	ルートマップを設定し、トラッキング対象オブジェクトの到達可能性を確認します。 （注） このコマンドは、IPv6 および VRF ではサポートされません。

	コマンドまたはアクション	目的
ステップ 7	**exit** 例： `スイッチ(config-route-map)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 8	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	インターフェイス コンフィギュレーション モードを開始し、設定するインタフェースを指定します。
ステップ 9	**ip policy route-map** *map-tag* 例： `スイッチ(config-if)# ip policy route-map pbr-map`	レイヤ 3 インターフェイス上で PBR を有効にし、使用するルート マップを識別します。1 つのインターフェイスに設定できるルートマップは、1つだけです。ただし、異なるシーケンス番号を持つ複数のルート マップ エントリを設定できます。これらのエントリは、最初の一致が見つかるまで、シーケンス番号順に評価されます。一致が見つからない場合、パケットは通常どおりにルーティングされます。
ステップ 10	**ip route-cache policy** 例： `スイッチ(config-if)# ip route-cache policy`	（任意）PBRの高速スイッチングを有効にします。PBR の高速スイッチングを有効にするには、PBR を有効にする必要があります。
ステップ 11	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 12	**ip local policy route-map** *map-tag* 例： `スイッチ(config)# ip local policy route-map local-pbr`	（任意）ローカル PBR を有効にして、スイッチから送信されるパケットに PBR を実行します。ローカル PBR は、スイッチによって生成されるパケットに適用されます。着信パケットには適用されません。
ステップ 13	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 14	**show route-map** [*map-name*] 例： `スイッチ# show route-map`	（任意）設定を確認するため、設定されたすべてのルート マップ、または指定されたルート マップだけを表示します。
ステップ 15	**show ip policy** 例： `スイッチ# show ip policy`	（任意）インターフェイスに付加されたポリシー ルート マップを表示します。

	コマンドまたはアクション	目的
ステップ 16	**show ip local policy** 例： `スイッチ# show ip local policy`	（任意）ローカル PBR が有効であるかどうか、および有効である場合は使用されているルート マップを表示します。

PBR を設定するための機能情報

表 *113 : PBR* の機能情報

機能名	リリース	機能情報
ポリシーベース ルーティング	Cisco IOS リリース 15.2(6)E2	ポリシーベースのルーティングを使用して、トラフィックフローに定義済みポリシーを設定します。

第 49 章

EIGRP スタブ ルーティングの設定

EIGRP スタブ ルーティング

EIGRP スタブルーティング機能は、ネットワークの安定性を高め、リソース利用率を抑え、スタブデバイス構成を簡素化します。

スタブルーティングは一般にハブアンドスポーク型のネットワークトポロジで使用されます。ハブアンドスポーク型ネットワークでは、1 つ以上のエンド（スタブ）ネットワークが 1 台のリモートデバイス（スポーク）に接続され、そのリモートデバイスは 1 つ以上のディストリビューションデバイス（ハブ）に接続されています。リモートデバイスは、1 つ以上のディストリビューション デバイスに隣接しています。IP トラフィックがリモートデバイスに到達するための唯一のルートは、ディストリビューション デバイスを経由するものです。

EIGRP スタブルーティングに関する情報

EIGRP スタブ ルーティング

EIGRP スタブルーティング機能は、ネットワークの安定性を高め、リソース利用率を抑え、スタブデバイス構成を簡素化します。

スタブルーティングは一般にハブアンドスポーク型のネットワークトポロジで使用されます。ハブアンドスポーク型ネットワークでは、1 つ以上のエンド（スタブ）ネットワークが 1 台のリモートデバイス（スポーク）に接続され、そのリモートデバイスは 1 つ以上のディストリビューションデバイス（ハブ）に接続されています。リモートデバイスは、1 つ以上のディストリビューション デバイスに隣接しています。IP トラフィックがリモートデバイスに到達するための唯一のルートは、ディストリビューションデバイスを経由するものです。このタイプの設定は、一般的に、ディストリビューション デバイスが WAN に直接接続されている WAN トポロジで使用されます。ディストリビューションデバイスは、多くの場合、多数のリモートデバイスに接続できます。ハブアンドスポーク型トポロジでは、リモートデバイスがすべての非ローカルトラフィックをディストリビューションデバイスに転送する必要があります。これにより、リモートデバイスが完全なルーティングテーブルを保有する必要はなくなります。一

般に、ディストリビューションデバイスはデフォルトルート以外の情報をリモートデバイスに送信する必要はありません。

EIGRP スタブルーティング機能を使用する場合、EIGRP を使用するように、ディストリビューションデバイスおよびリモートデバイスを設定し、さらにリモートデバイスだけをスタブとして設定する必要があります。指定されたルートのみが、リモート（スタブ）デバイスから伝播されます。スタブデバイスは、サマリー、接続されているルート、再配布されたスタティックルート、外部ルート、および内部ルートに対するクエリーすべてに、応答として「inaccessible」というメッセージを返します。スタブとして設定されているデバイスは、特殊なピア情報パケットをすべての隣接デバイスに送信して、そのステータスをスタブデバイスとして報告します。

スタブステータスの情報を伝えるパケットを受信したネイバーはすべて、スタブデバイスにルートのクエリーを送信しなくなり、スタブピアを持つデバイスはそのピアのクエリーを送信しなくなります。スタブデバイスは、ディストリビューションデバイスを使用して適切なアップデートをすべてのピアに送信します。

次の図は、単純なハブアンドスポーク型ネットワークを示しています。

ルートがリモートデバイスにアドバタイズされることを、スタブルーティング機能自体が回避することはありません。上の例では、リモートデバイスはディストリビューションデバイスを経由してのみ企業ネットワークおよびインターネットにアクセスできます。リモートデバイスが完全なルートテーブルを保有しても機能面での意味はありません。これは、企業ネットワークとインターネットへのパスは常にディストリビューション デバイスを経由するためです。ルートテーブルが大きくなると、リモートデバイスに必要なメモリ量が減るだけです。帯域幅とメモリは、ディストリビューションデバイスのルートを集約およびフィルタリングすることによって節約できます。リモートデバイスは、宛先に関係なく、ディストリビューションデバイスにすべての非ローカルトラフィックを送信する必要があるため、他のネットワークから学習されたルートを受け取る必要がありません。真のスタブネットワークが望ましい場合は、ディストリビューションデバイスがリモートデバイスにデフォルトルートだけを送信するように設定する必要があります。EIGRP スタブルーティング機能では、ディストリビューションデバイスでの集約を自動的に有効にしません。ほとんどの場合、ネットワーク管理者が、ディストリビューション デバイスにサマライズを設定する必要があります。

（注） ディストリビューション デバイスがリモートデバイスにデフォルトルートだけを送信するように設定する場合、リモートデバイスで **ip classless** コマンドを使用する必要があります。デフォルトでは、EIGRP スタブルーティング機能をサポートするシスコのすべてのイメージで **ip classless** コマンドが有効になっています。

EIGRP スタブルーティング機能がない場合、ディストリビューション デバイスからリモートデバイスに送信されたルートがフィルタリングまたは集約された後でも、問題が発生することがあります。企業ネットワーク内でルートが失われると、EIGRPはクエリーをディストリビューションデバイスに送信できます。ルートがサマライズされている場合でも、ディストリビューションデバイスが代わりにリモートデバイスにクエリーを送信します。ディストリビューションデバイスとリモートデバイスの間の通信（WANリンクを介した）に問題がある場合、EIGRP Stuck In Active（SIA）状態が発生し、ネットワークのどこかで不安定になる可能性がありま

す。EIGRP スタブルーティング機能を使用することにより、ネットワーク管理者はリモートデバイスへクエリーが送信されないようにできます。

デュアルホーム接続リモート トポロジ

リモートデバイスを単一のディストリビューション デバイスに接続する単純なハブアンドスポーク型ネットワーク以外に、リモートデバイスを複数のディストリビューションデバイスにデュアルホーム接続できます。この構成では冗長性が増し、一意性の問題が生じますが、スタブ機能がこれらの問題の対処に役立ちます。

デュアルホーム接続されたリモートデバイスは、複数のディストリビューション（ハブ）デバイスを持ちます。ただし、スタブ ルーティングの原理はハブ アンド スポーク型トポロジの場合と同じです。下の図は、リモートデバイスを 1 つ使用した一般的なデュアルホーム接続リモートトポロジを示していますが、ディストリビューション デバイス 1 とディストリビューションデバイス 2 の同じインターフェイスに 100 以上のデバイスを接続できます。リモートデバイスは、最適なルートを使用して宛先に到達します。ディストリビューションデバイス 1 に障害が発生した場合、リモートデバイスはディストリビューションデバイス 2 を使用して企業ネットワークに到達できます。

上の図は、1 つのリモートデバイスと 2 つのディストリビューション デバイスを持つ単純なデュアルホーム接続リモートトポロジを示しています。いずれのディストリビューションデバイスも企業ネットワークとスタブネットワーク 10.1.1.0/24 へのルートを維持します。

デュアルホーム接続ルーティングによって、EIGRP ネットワークが不安定になる場合があります。下の図では、ディストリビューション デバイス 1 はネットワーク 10.3.1.0/24 に直接接続しています。ディストリビューションデバイス 1 に要約またはフィルタリングが適用された場合、デバイスはネットワーク 10.3.1.0/24 を、直接接続されているすべての EIGRP ネイバー（ディストリビューション デバイス 2 およびリモートデバイス）にアドバタイズします。

上の図に、ディストリビューション デバイス 1 をネットワーク 10.3.1.0/24 とネットワーク 10.2.1.0/24 の両方に接続した単純なデュアルホーム接続リモートトポロジを示します。

ディストリビューション デバイス 1 とディストリビューション デバイス 2 間の 10.2.1.0/24 リンクに障害が発生した場合、ディストリビューション デバイス 2 からネットワーク 10.3.1.0/24 までの最低コストパスはリモートデバイスを経由します（下の図を参照）。それまで企業ネットワーク 10.2.1.0/24 を通過していたトラフィックが、今度は帯域幅の非常に低い接続に送信されるため、このルートは望ましくありません。低帯域幅 WAN 接続の利用率が高くなりすぎると、企業ネットワーク全体に影響するような多くの問題の原因になります。リモートデバイスを通過する低帯域幅ルートの利用によって、WAN EIGRP ディストリビューション デバイスがドロップする場合があります。ディストリビューションおよびリモートデバイスのシリアル回線もドロップし、ディストリビューションおよびコアデバイスで EIGRP SIA エラーが発生する可能性があります。

ディストリビューション デバイス 2 からのトラフィックがネットワーク 10.3.1.0/24 に到達するために、リモートデバイスを通過するのは望ましくありません。リンクが負荷を管理できるサイズに設定されている場合は、バックアップルートを使用できます。ただし、上の図に示しているタイプのほとんどのネットワークは、リモートデバイスをリンク速度が比較的遅いリモートオフィスに配置しています。ディストリビューションデバイスからのトラフィックがリ

モートデバイス経由でルーティングされないようにするために、ディストリビューションデバイスとリモートデバイスでルート集約を構成できます。

通常、ディストリビューションデバイスからのトラフィックが中継パスとしてリモートデバイスを使用するのは不適切です。ディストリビューションデバイスからリモートデバイスへの一般的な接続は、ネットワークコアにおける接続よりも帯域幅が相当低くなります。中継パスとして帯域幅接続に限りがあるリモートデバイスを使用した場合、一般にリモートデバイスに過度の輻輳が生じます。EIGRPスタブルーティング機能は、リモートデバイスがディストリビューションデバイスにコアルートをアドバタイズしないようにしてこの問題を防ぎます。上記の例では、リモートデバイスがディストリビューションデバイス1から学習したルートは、ディストリビューションデバイス2にアドバタイズされません。したがって、ディストリビューションデバイス2は、ネットワークコアを宛先とするトラフィックのトランジットとしてリモートデバイスを使用しません。

EIGRPスタブルーティング機能は、ネットワークの安定性をもたらします。ネットワークが不安定になったときに、EIGRPクエリが非中継デバイスへの制限された帯域幅リンクを介して送信されるのを防ぎます。代わりに、スタブデバイスの接続先のディストリビューションデバイスがスタブデバイスに代わってクエリに応答します。この機能により、輻輳している、または問題のある WAN リンクによってネットワークが不安定になる可能性が低減されます。また、EIGRP スタブ ルーティング機能を使用すると、ハブアンドスポーク ネットワークの設定とメンテナンスが簡略化されます。スタブルーティングをデュアルホーム接続のリモート設定でイネーブルにすると、リモートデバイスがハブデバイスへの中継パスとして表示されないようにリモートデバイスでフィルタリングを設定する必要がなくなります。

注意 EIGRP スタブ ルーティング機能は、スタブ デバイスだけで使用します。スタブ デバイスは、コア中継トラフィックが通過しないネットワーク コアまたはディストリビューション レイヤに接続されたデバイスとして定義されます。スタブデバイスがディストリビューション デバイス以外の EIGRP ネイバーを持つことはできません。この制限を無視すると、望ましくない動作が発生します。

（注） ATM、ギガビットイーサネット、フレームリレー、ISDN PRI、X.25 などのマルチアクセス インターフェイスは、そのインターフェイス上にあるハブを除く全デバイスがスタブデバイスとして設定される場合だけ、EIGRP スタブルーティング機能によってサポートされます。

EIGRP スタブ ルーティングの設定方法

EIGRP スタブルーティング自律システム設定の設定

手順の概要

1. **enable**

2. **configure terminal**
3. **router eigrp** *autonomous-system-number*
4. **network** *ip-address* [**wildcard-mask**]
5. **eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static**] [**summary**] [**redistributed**]
6. **end**
7. **show ip eigrp neighbors** [*interface-type* | *as-number* | **static** | **detail**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**router eigrp** *autonomous-system-number* 例： `Device(config)# router eigrp 1`	EIGRP プロセスを実行するリモートデバイスまたはディストリビューションデバイスを設定し、ルータ コンフィギュレーション モードを開始します。
ステップ 4	**network** *ip-address* [**wildcard-mask**] 例： `Device(config-router)# network 172.16.0.0`	EIGRP ディストリビューション デバイスのネットワークアドレスを指定します。
ステップ 5	**eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static**] [**summary**] [**redistributed**] 例： `Device(config-router)# eigrp stub connected static`	リモートデバイスを EIGRP スタブデバイスとして設定します。
ステップ 6	**end** 例： `Device(config-router)# end`	ルータ コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。
ステップ 7	**show ip eigrp neighbors** [*interface-type* | *as-number* | **static** | **detail**] 例： `Device# show ip eigrp neighbors detail`	（任意）リモートデバイスが、EIGRP のスタブデバイスとして設定されていることを確認します。 • 配布デバイスからこのコマンドを入力します。出力の最後の行には、リモートデバイスまたは

	コマンドまたはアクション	目的
		スポークデバイスのスタブステータスが表示されます。

EIGRP スタブルーティング名前付き設定の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **router eigrp** *virtual-instance-name*
4. 次のいずれか 1 つを入力します。
   - **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*
   - **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*
5. **network** *ip-address* [**wildcard-mask**]
6. **eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static**] [**summary**] [**redistributed**]
7. **exit-address-family**
8. **end**
9. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] [**neighbors**] [**static**] [**detail**] [*interface-type interface-number*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**router eigrp** *virtual-instance-name* 例： `Device(config)# router eigrp virtual-name1`	EIGRP ルーティング プロセスをイネーブルにし、ルータ コンフィギュレーション モードを開始します。
ステップ 4	次のいずれか 1 つを入力します。 • **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number* • **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*	アドレス ファミリ コンフィギュレーション モードを開始して、EIGRP IPv4 または IPv6 ルーティング インスタンスを設定します。

	コマンドまたはアクション	目的
	例： `Device(config-router)# address-family ipv4 autonomous-system 45000` `Device(config-router)# address-family ipv6 autonomous-system 45000`	
ステップ 5	**network** *ip-address* [**wildcard-mask**] 例： `Device(config-router-af)# network 172.16.0.0`	EIGRP ディストリビューション デバイスのネットワークアドレスを指定します。
ステップ 6	**eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static**] [**summary**] [**redistributed**] 例： `Device(config-router-af) eigrp stub leak-map map1`	デバイスを EIGRP を使用するスタブとして設定します。
ステップ 7	**exit-address-family** 例： `Device(config-router-af)# exit-address-family`	アドレス ファミリ コンフィギュレーション モードを終了します。
ステップ 8	**end** 例： `Device(config-router)# end`	ルータ コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。
ステップ 9	**show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] [**neighbors**] [**static**] [**detail**] [*interface-type interface-number*] 例： `Device# show eigrp address-family ipv4 neighbors detail`	（任意）EIGRP によって検出されたネイバーを表示します。

EIGRP スタブルーティングの設定例

例：EIGRP スタブルーティング：自律システム設定

eigrp stub コマンドでスタブとして設定されたデバイスは、デフォルトで接続および集約ルーティング情報をすべてのネイバーデバイスと共有します。この動作を変更するには、**eigrp stub** コマンドで次の 6 個のキーワードを使用します。

- **connected**
- **leak-map**
- **receive-only**
- **redistributed**

- **static**
- **summary**

ここでは、EIGRP 自律システム設定に対する、**eigrp stub** コマンドのすべての形式の設定例を示します。

例：eigrp stub コマンド

次の例では、**eigrp stub** コマンドを使用して、接続ルートとサマリールートをアドバタイズするスタブとしてデバイスを設定します。

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub
```

例：eigrp stub connected static コマンド

次の例では、**eigrp stub** コマンドを **connected** および **static** の各キーワードを指定して使用し、接続ルートとスタティックルートをアドバタイズするスタブとしてデバイスを設定しています（サマリールートの送信は許可されません）。

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub connected static
```

例：eigrp stub leak-map コマンド

次の例では、**leak-map** *name* キーワードと引数のペアを指定して **eigrp stub** コマンドを発行し、抑制されるルートを識別するリークマップを参照するようデバイスを設定しています。

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub leak-map map1
```

例：eigrp stub receive-only コマンド

次の例では、**eigrp stub** コマンドを **receive-only** キーワードを指定して発行し、受信専用のネイバーとしてデバイスを設定しています（接続ルート、サマリールート、およびスタティックルートは送信されません）。

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub receive-only
```

例：eigrp stub redistributed コマンド

次の例では、**eigrp stub** コマンドを **redistributed** キーワードを指定して発行し、他のプロトコルおよび自律システムをアドバタイズするようにデバイスを設定しています。

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# eigrp stub redistributed
```

例：EIGRP スタブルーティング：名前付き設定

eigrp stub コマンドでスタブとして設定されたデバイスは、デフォルトで接続および集約ルーティング情報をすべてのネイバーデバイスと共有します。この動作を変更するには、**eigrp stub** コマンドで次の 6 個のキーワードを使用します。

- **connected**
- **leak-map**
- **receive-only**
- **redistributed**
- **static**
- **summary**

ここでは、EIGRP 名前付き設定に対する、**eigrp stub** コマンドのすべての形式の設定例を示します。

例：eigrp stub コマンド

次の例では、**eigrp stub** コマンドを使用して、接続ルートとサマリールートをアドバタイズするスタブとしてデバイスを設定します。

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af) eigrp stub
```

例：eigrp stub connected static コマンド

次の名前付き設定の例では、**eigrp stub** コマンドを **connected** および **static** の各キーワードを指定して発行し、接続ルートとスタティックルートをアドバタイズするスタブとしてデバイスを設定しています（サマリールートの送信は許可されません）。

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# eigrp stub connected static
```

例：eigrp stub leak-map コマンド

次の名前付き設定の例では、**leak-map** *name* キーワードと引数のペアを指定して **eigrp stub** コマンドを発行し、通常は抑制されるルートを識別するリークマップを参照するようデバイスを設定しています。

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# eigrp stub leak-map map1
```

例：eigrp stub receive-only コマンド

次の名前付き設定の例では、**eigrp stub** コマンドを **receive-only** キーワードを指定して発行し、受信専用のネイバーとしてデバイスを設定しています（接続ルート、サマリールート、およびスタティックルートは送信されません）。

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# eigrp stub receive-only
```

例：eigrp stub redistributed コマンド

次の名前付き設定の例では、**eigrp stub** コマンドを **redistributed** キーワードを指定して発行し、他のプロトコルおよび自律システムをアドバタイズするようにデバイスを設定しています。

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af) eigrp stub redistributed
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
EIGRP コマンド	『Cisco IOS IP Routing: EIGRP Command Reference』
EIGRP に関する FAQ	EIGRP よく寄せられる質問（FAQ）

関連項目	マニュアル タイトル
EIGRP テクノロジーに関するホワイトペーパー	Enhanced Interior Gateway Routing Protocol

シスコのテクニカル サポート

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

EIGRP スタブルーティングの機能情報

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

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

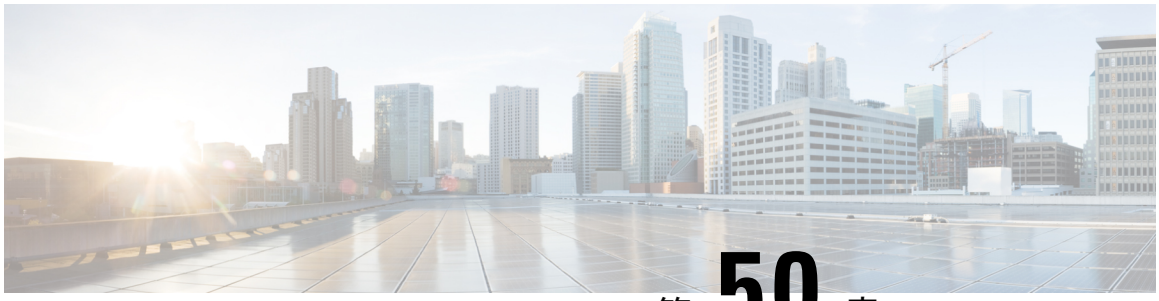
表 *114 : EIGRP* スタブルーティングの機能情報

機能名	リリース	機能情報
EIGRP スタブ ルーティング	Cisco IOS XE 15.2(6)E2	EIGRP スタブ ルーティング機能では、ネットワークの安定性の改善、リソース使用率の低減、およびスタブ ルータ設定の簡潔化が可能です。スタブ ルーティングは一般にハブ アンド スポーク型のネットワーク トポロジで使用されます。ハブ アンド スポーク ネットワークでは、1 つ以上のエンド（スタブ）ネットワークが 1 台のリモート ルータ（スポーク）に接続され、そのリモート ルータは 1 つ以上のディストリビューション ルータ（ハブ）に接続されています。リモート ルータは、1 つ以上のディストリビューション ルータにのみ隣接しています。

第 IX 部

セキュリティ

- 自動 ID （1717 ページ）
- ポート単位のトラフィック制御の設定 （1729 ページ）
- IPv6 ファースト ホップ セキュリティの設定 （1761 ページ）
- FIPS の設定 （1799 ページ）
- コントロール プレーン ポリシングの設定 （1801 ページ）



第 50 章

セキュリティ機能の概要

セキュリティ機能の概要

セキュリティ機能は次のとおりです。

- IPv6 ファースト ホップ セキュリティ：IPv6 ネットワークの持つ脆弱性から保護するためにファースト ホップ スイッチに適用されるセキュリティ機能のセット。これらには、バインディング統合ガード（バインディング テーブル）、ルータ アドバタイズメント ガード（RA ガード）、DHCP ガード、IPv6 ネイバー探索検査（ND ガード）などがあります。
- Web 認証：Web ブラウザを使用して認証する IEEE 802.1x 機能をサポートしないサプリカント（クライアント）を許可します。
- ローカル Web 認証バナー：Web 認証ログイン画面に表示されるカスタム バナーまたはイメージ ファイル。
- ACL および RADIUS Filter-Id 属性を使った IEEE 802.1x 認証。
- 管理インターフェイス（デバイス マネージャ、Network Assistant、CLI）へのパスワード保護付きアクセス（読み取り専用および読み書きアクセス）。不正な設定変更を防止します。
- セキュリティ レベル、通知、および対応するアクションを選択できる、マルチレベル セキュリティ。
- セキュリティを確保できるスタティック MAC アドレッシング。
- 保護ポート オプション。同一スイッチ上の指定ポートへのトラフィック転送を制限します。
- ポートにアクセスできるステーションの MAC アドレスを制限または特定するポート セキュリティ オプション。
- VLAN 認識ポート セキュリティ オプション。違反の発生時にポート全体をシャットダウンするのではなく、そのポート上の VLAN をシャットダウンします。

- ポート セキュリティ エージング。ポートのセキュア アドレスにエージング タイムを設定します。
- 指定した入力割合を超えたパケットをドロップして、スイッチへの着信プロトコル トラフィックの割合を制御する、プロトコル ストーム プロテクション。
- BPDU ガード。無効なコンフィギュレーションが発生した場合に、PortFast が設定されているポートをシャットダウンします。
- 標準および拡張 IP アクセス コントロール リスト（ACL）は、レイヤ 2 インターフェイス（ポート ACL）でのインバウンドなセキュリティ ポリシーを定義します。
- MAC 拡張アクセス コントロール リスト。レイヤ 2 インターフェイスの着信方向のセキュリティ ポリシーを定義します。
- 非 IP トラフィックをフィルタリングする、送信元および宛先 MAC ベースの ACL。
- 信頼できないホストと DHCP サーバの間の信頼できない DHCP メッセージをフィルタリングする DHCP スヌーピング。
- DHCP スヌーピングデータベース、および IP ソースバインディングに基づいてトラフィックをフィルタリングすることにより、非ルーテッドインターフェイスでのトラフィックを制限する IP ソースガード。
- 不正な ARP 要求や応答を同じ VLAN 上のその他のポートにリレーしないことにより、スイッチに対する悪意のある攻撃を回避するためのダイナミック ARP インスペクション。
- IEEE 802.1x ポートベース認証。不正なデバイス（クライアント）によるネットワーク アクセスを防止します。次の 802.1x 機能がサポートされます。
  - シングルホスト、マルチホスト、マルチ認証、およびマルチドメイン認証モードのサポート。
  - データ装置と IP Phone などの音声装置（シスコ製品またはシスコ以外の製品）の両方が、同じ IEEE 802.1x 対応スイッチ ポートにおいて、単独で認証できるようにするマルチドメイン認証（MDA）。
  - MDA のダイナミック音声 VLAN（仮想 LAN）。ダイナミック音声 VLAN が MDA 対応ポートで可能になります。
  - VLAN 割り当て。802.1x 認証ユーザを特定の VLAN に制限します。
  - マルチ認証モードで設定されたポートでの VLAN 割り当てのサポート。RADIUS サーバは、ポートで最初に認証されるホストに VLAN を割り当て、後続のホストは同じ VLAN を使用します。音声 VLAN 割り当ては、1 つの IP Phone に対してサポートされます。
  - ポート セキュリティ。802.1x ポートへのアクセスを制御します。
  - 音声 VLAN。ポートが許可ステートか無許可ステートかにかかわらず、Cisco IP Phone の音声 VLAN へのアクセスを許可します。
  - IP Phone 検出機能拡張。Cisco IP Phone を検出し識別します。

- ゲスト VLAN。802.1x に適合しないユーザに限定的なサービスを提供します。
- 制限付き VLAN。802.1x に準拠はしているが、標準の 802.1x で認証するためのクレデンシャルを持っていないユーザに制限付きのサービスを提供します。
- 802.1x アカウンティング。ネットワーク使用をトラッキングします。
- 802.1x と LAN の Wake-on-LAN（WoL）機能。休止状態の PC に、特定のイーサネット フレームを送信して起動させます。
- 802.1x 準備状態チェック。スイッチで IEEE 802.1x を設定する前に、接続されたエンド ホストの準備状態を判断します。
- セキュリティ違反が発生した VLAN だけでトラフィック違反アクションを適用するための音声認識 802.1x セキュリティ。
- MAC 認証バイパス（MAB）。クライアント MAC アドレスに基づいてクライアントを許可します。
- デバイスのネットワーク アクセスを許可する前の、エンドポイント システムやクライアントのウイルス対策の状態またはポスチャに関する Network Admission Control（NAC）レイヤ 2 802.1x 検証。

（注） NAC は LanLite イメージではサポートされません。

- 802.1X スイッチ サプリカントを持つ Network Edge Access Topology（NEAT）、CISP を使ったホスト認証、および自動イネーブル化。これらにより、別のスイッチへのサプリカントとして、配線クローゼットの外のスイッチが認証されます。

（注） NEAT は LanLite イメージではサポートされません。

- 認証される前にネットワークへのアクセスをホストに許可するための、オープン アクセスを使用した IEEE 802.1x。

（注） この機能は LanLite イメージではサポートされません。

- ダウンロード可能な ACL とリダイレクト URL を使用した IEEE 802.1x 認証。Cisco Secure ACS サーバーから認証されたスイッチへのユーザー単位の ACL ダウンロードを使用できるようになります。
- スタティック ACL が設定されていないポートでの認証デフォルト ACL のダイナミックな作成または接続のサポート。

（注） この機能は LanLite イメージではサポートされません。

- 新しいホストを認証するときに、ポートが思考する認証メソッドの順序を設定するための柔軟な認証シーケンス。
- マルチユーザー認証。複数のホストが、802.1x 対応ポートを認証できるようになります。

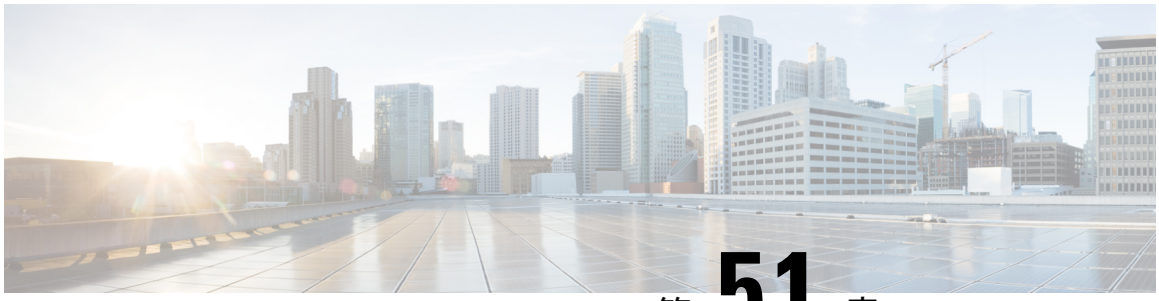
- TACACS+。IPv4 および IPv6 対応の TACACS サーバーを介してネットワーク セキュリティを管理する独自の機能。
- IPv4 および IPv6 対応の認証、許可、アカウンティング（AAA）サービスを使用して、リモートユーザーの ID の検証、アクセスの許可、アクションの追跡を実行するための RADIUS。
- IPv6 上での機能向けに、RADIUS、TACACS+、および SSH を拡張。
- HTTP 1.1 サーバ認証、暗号化、メッセージ整合性、HTTP クライアント認証用に Secure Socket Layer（SSL）バージョン 3.0 がサポートされ、安全な HTTP 通信が可能になります（ソフトウェアの暗号化バージョンが必要）。
- ACL および RADIUS Filter-Id 属性を使った IEEE 802.1x 認証。
- スタティック ホストでの IP ソース ガードのサポート。
- RADIUS 認証の変更（CoA）。特定のセッション認証された後で、その属性を変更します。AAA でユーザ、またはユーザ グループのポリシーに変更がある場合、管理者は Cisco Identity Services Engine または Cisco Secure ACS などの AAA サーバから、RADIUS CoA パケットを送信し、新しいポリシーに適用することができます。
- IEEE 802.1x User Distribution。さまざまな VLAN にわたってユーザをロード バランシングすることにより、（ユーザ グループに対して）複数の VLAN を使った配置で、ネットワークのスケーラビリティを向上させることができます。認証されたユーザは、RADIUS サーバにより割り当てられた、グループ内で最も空いている VLAN に割り当てられます。

（注） この機能は LanLite イメージではサポートされません。

- クリティカル VLAN のサポート: AAA サーバーが到達不能になった場合に、重要なリソースへのアクセスを許可するために、マルチホスト/マルチ認証対応ポートが重要な VLAN に配置されます。

（注） この機能は LanLite イメージではサポートされません。

- ポート ホスト モードを変更し、オーセンティケータのスイッチ ポートに標準ポート設定を適用するために Network Edge Access Topology（NEAT）をサポート。
- VLAN-ID ベースの MAC 認証。ユーザー認証のために VLAN と MAC のアドレス情報を結合して、許可されていない VLAN からのネットワーク アクセスを阻止します。
- MAC 移動。モビリティのイネーブル化を制約することなく、ホスト（IP Phone の背後で接続されたホストを含む）が同じスイッチ内のポート間を移動できるようになります。MAC 移動では、もう 1 つのポートに同じ MAC アドレスが再登場した場合、スイッチはこれをまったく新しい MAC アドレスと同様に扱います。
- 簡易ネットワーク管理プロトコル バージョン 3（SNMPv3）を使った 3DES および AES のサポート。このリリースでは、168 ビットの Triple Data Encryption Standard（3DES）と、SNMPv3 への 128 ビット、192 ビット、および 256 ビットの Advanced Encryption Standard（AES）暗号化アルゴリズムに対するサポートが追加されます。
- Cisco TrustSec SXP プロトコルのサポート。この機能は LanLite イメージではサポートされません。



第 51 章

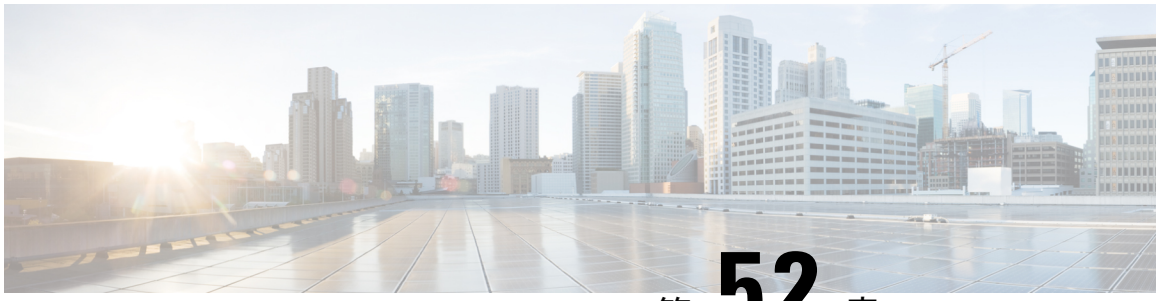
不正アクセスの防止

不正アクセスの防止

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

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

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



第 52 章

パスワードおよび権限レベルによるスイッチ アクセスの制御

パスワードおよび権限によるスイッチアクセスの制御の制約事項

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

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

パスワードおよび権限レベルに関する情報

デフォルトのパスワードおよび権限レベル設定

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

次の表に、デフォルトのパスワードおよび権限レベル設定を示します。

表 *115*：デフォルトのパスワードおよび権限レベル設定

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

追加のパスワード セキュリティ

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

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

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

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

マスクされたシークレットパスワード

（注） この機能は、Cisco Catalyst 3560-CX シリーズ スイッチでのみサポートされています。

enable secret コマンドを使用すると、パスワードは暗号化されますが、パスワードを入力するときに端末に表示されます。端末でパスワードをマスクするには、**masked-secret** グローバル コンフィギュレーション コマンドを使用します。このパスワードの暗号化タイプは、デフォルトではタイプ 9 です。

このコマンドを使用して、コモンクライテリアポリシーのマスクされたシークレットパスワードを設定できます。

パスワードの回復

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

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

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

パスワードの回復を再びイネーブルにするには、**service password-recovery** グローバル コンフィギュレーション コマンドを使用します。

端末回線の Telnet 設定

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

ユーザ名とパスワードのペア

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

権限レベル

シスコデバイスでは、権限レベルを使用して、スイッチ動作の異なるレベルに対してパスワード セキュリティを提供します。デフォルトでは、Cisco IOS ソフトウェアは、パスワード セキュリティの 2 つのモード（権限レベル）で動作します。ユーザ EXEC（レベル 1）および特権 EXEC（レベル 15）です。各モードに、最大 16 個の階層レベルからなるコマンドを設定で

きます。複数のパスワードを設定することにより、ユーザ グループ別に特定のコマンドへのアクセスを許可することができます。

回線の権限レベル

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

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

コマンド権限レベル

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

パスワードおよび権限レベルでスイッチ アクセスを制御する方法

スタティック 有効 パスワードの設定または変更

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

手順の概要

1. **enable**
2. **configure terminal**
3. **enable password** *password*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**enable password** *password* 例： スイッチ(config)# **enable password secret321**	特権 EXEC モードにアクセスするための新しいパスワードを定義するか、既存のパスワードを変更します。 デフォルトでは、パスワードは定義されません。 *password* には、1 ～ 25 文字の英数字のストリングを指定します。ストリングを数字で始めることはできません。大文字と小文字を区別し、スペースを使用できますが、先行スペースは無視されます。疑問符（?）は、パスワードを作成する場合に、疑問符の前に Ctrl+v を入力すれば使用できます。たとえば、パスワード abc?123 を作成するときは、次のようにします。 **1.** **abc** を入力します。 **2.** Ctrl+v を入力します。 **3.** **?123** を入力します。 システムからイネーブルパスワードを入力するように求められた場合、疑問符の前に Ctrl+v を入力する必要はなく、パスワードのプロンプトにそのまま abc?123 と入力できます。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例：	入力を確認します。

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

暗号化によるイネーブルおよびイネーブル シークレット パスワードの保護

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

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかを使用します。
   - `enable password [level` *level*`]` {*password encryption-type encrypted-password*}
   - `enable secret [level` *level*`]` {*password encryption-type encrypted-password*}
4. **service password-encryption**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	次のいずれかを使用します。 • enable password [level *level*] {*password encryption-type encrypted-password*} • enable secret [level *level*] {*password encryption-type encrypted-password*} **例**： スイッチ(config)# **enable password example102** または スイッチ(config)# **enable secret level 1 password secret123sample**	• 特権 EXEC モードにアクセスするための新しいパスワードを定義するか、既存のパスワードを変更します。 • シークレット パスワードを定義します。これは非可逆的な暗号化方式を使用して保存されます。 • （任意）*level* に指定できる範囲は 0 ～ 15 です。レベル 1 が通常のユーザ EXEC モード権限です。デフォルト レベルは 15 です（特権 EXEC モード権限）。 • *password* には、1 ～ 25 文字の英数字のストリングを指定します。ストリングを数字で始めることはできません。大文字と小文字を区別し、スペースを使用できますが、先行スペースは無視されます。デフォルトでは、パスワードは定義されません。 • （任意）*encryption-type* には、シスコ独自の暗号化アルゴリズムであるタイプ 5 しか使用できません。暗号化タイプを指定する場合は、暗号化されたパスワードを使用する必要があります。この暗号化パスワードは、別のスイッチの設定からコピーします。 （注） 暗号化タイプを指定してクリア テキスト パスワードを入力した場合は、再び特権 EXEC モードを開始することはできません。暗号化されたパスワードが失われた場合は、どのような方法でも回復することはできません。
ステップ 4	**service password-encryption** **例**： スイッチ(config)# **service password-encryption**	（任意）パスワードの定義時または設定の書き込み時に、パスワードを暗号化します。 暗号化を行うと、コンフィギュレーション ファイル内でパスワードが読み取り可能な形式になるのを防止できます。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

マスクされたシークレットパスワードの設定

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかを使用します。
   - **username** *name***masked-secret**
   - **username** *name***common-criteria-policy** *policy-name* **masked-secret**
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： Device> **enable**	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： Device# **configure terminal**	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	次のいずれかを使用します。 • **username** *name***masked-secret** • **username** *name***common-criteria-policy** *policy-name* **masked-secret** 例： `Device(config)# username cisco masked-secret` または `Device(config)# username common-criteria-policy test-policy masked-secret`	• マスクされたシークレットパスワードを定義します。これは非可逆的な暗号化方式を使用して保存されます。 • コモンクライテリアポリシーのマスクされたシークレットパスワードを定義します。 • マスクされたシークレットパスワードは 5 文字以上にする必要があります。マスクされたシークレットパスワードの最大長は 256 文字です。デフォルトでは、パスワードは定義されません。
ステップ 4	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

パスワード回復のディセーブル化

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

始める前に

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

手順の概要

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

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**system disable password recovery switch** {*all* | *<1-9>*} 例： スイッチ(config)# **system disable password recovery switch all**	パスワード回復をディセーブルにします。 • *all*：スタック内のスイッチで設定を行います。 • *<1-9>*：選択したスイッチ番号で設定を行います。 この設定は、フラッシュ メモリの中で、ブートローダおよび Cisco IOS イメージがアクセスできる領域に保存されますが、ファイル システムには含まれません。また、ユーザーがアクセスすることはできません。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

次のタスク

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

端末回線に対する Telnet パスワードの設定

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

始める前に

- エミュレーション ソフトウェアを備えた PC またはワークステーションをスイッチ コンソール ポートに接続するか、または PC をイーサネット管理ポートに接続します。

- コンソール ポートのデフォルトのデータ特性は、9600 ボー、8 データ ビット、1 ストップ ビット、パリティなしです。コマンドライン プロンプトが表示されるまで、Return キーを何回か押す必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **line vty 0 15**
4. **password** *password*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	（注） パスワードが特権 EXEC モードへのアクセスに必要な場合は、その入力が求められます。 特権 EXEC モードを開始します。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**line vty 0 15** 例： `スイッチ(config)# line vty 0 15`	Telnet セッション（回線）の数を設定し、ライン コンフィギュレーション モードを開始します。 コマンド対応スイッチでは、最大 16 のセッションが可能です。0 および 15 を指定すると、使用できる 16 の Telnet セッションすべてを設定することになります。
ステップ 4	**password** *password* 例： `スイッチ(config-line)# password abcxyz543`	1 つまたは複数の回線に対応する Telnet パスワードを設定します。 *password* には、1 ～ 25 文字の英数字のストリングを指定します。ストリングを数字で始めることはできません。大文字と小文字を区別し、スペースを使用できますが、先行スペースは無視されます。デフォルトでは、パスワードは定義されません。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config-line)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ユーザ名とパスワードのペアの設定

ユーザー名とパスワードのペアを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **username** *name* [**privilege** *level*] { **password** *encryption-type password*}
4. 次のいずれかを使用します。
   - **line console 0**
   - **line vty 0 15**
5. **login local**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**username** *name* [**privilege** *level*] { **password** *encryption-type password*} 例： `スイッチ(config)# username adamsample privilege 1 password secret456` `スイッチ(config)# username 111111111111 mac attribute`	各ユーザのユーザ名、権限レベル、パスワードを設定します。 • *name* には、ユーザー ID を 1 ワードで指定するか、またはMACアドレスを指定します。スペースと引用符は使用できません。 • ユーザ名と MAC フィルタの両方に対し、最大 12000 のクライアントを個別に設定できます。 • （任意）*level* には、アクセス権を得たユーザーに設定する権限レベルを指定します。指定できる範囲は 0 ～ 15 です。レベル 15 では特権 EXEC モードでのアクセスが可能です。レベル 1 では、ユーザ EXEC モードでのアクセスとなります。 • *encryption-type* には、暗号化されていないパスワードが後ろに続く場合は 0 を入力します。暗号化されたパスワードが後ろに続く場合は 7 を指定します。 • *password* には、スイッチへアクセスするためにユーザーが入力しなければならないパスワードを指定します。パスワードは 1 ～ 25 文字で、埋め込みスペースを使用でき、**username** コマンドの最後のオプションとして指定します。
ステップ 4	次のいずれかを使用します。 • **line console 0** • **line vty 0 15** 例： `スイッチ(config)# line console 0` または `スイッチ(config)# line vty 15`	ライン コンフィギュレーション モードを開始し、コンソール ポート（回線 0）または VTY 回線（回線 0 ～ 15）を設定します。

	コマンドまたはアクション	目的
ステップ 5	**login local** 例： `スイッチ(config-line)# login local`	ログイン時のローカルパスワードチェックをイネーブルにします。認証は、ステップ 3 で指定されたユーザ名に基づきます。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

コマンドの特権レベルの設定

コマンドの権限レベルを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **privilege** *mode* **level** *level command*
4. **enable password level** *level password*
5. **end**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**privilege** *mode* **level** *level command* 例： スイッチ(config)# **privilege exec level 14 configure**	コマンドの特権レベルを設定します。 • *mode* には、グローバル コンフィギュレーション モードの場合は **configure** を、EXEC モードの場合は **exec** を、インターフェイス コンフィギュレーション モードの場合は **interface** を、ライン コンフィギュレーション モードの場合は **line** をそれぞれ入力します。 • *level* の範囲は 0 ～ 15 です。レベル 1 が通常のユーザーEXEC モード権限です。レベル 15 は、**enable** パスワードによって許可されるアクセス レベルです。 • *command* には、アクセスを制限したいコマンドを指定します。
ステップ 4	**enable password level** *level password* 例： スイッチ(config)# **enable password level 14 SecretPswd14**	権限レベルをイネーブルにするためのパスワードを指定します。 • *level* の範囲は 0 ～ 15 です。レベル 1 が通常のユーザー EXEC モード権限です。 • *password* には、1 ～ 25 文字の英数字のストリングを指定します。ストリングを数字で始めることはできません。大文字と小文字を区別し、スペースを使用できますが、先行スペースは無視されます。デフォルトでは、パスワードは定義されません。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

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

回線のデフォルト特権レベルの変更

指定した回線のデフォルトの権限レベルを変更するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **line vty** *line*
4. **privilege level** *level*
5. **end**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**line vty** *line* 例： スイッチ(config)# **line vty 10**	アクセスを制限する仮想端末回線を選択します。
ステップ 4	**privilege level** *level* 例：	回線のデフォルト特権レベルを変更します。 *level* の範囲は 0 ～ 15 です。レベル 1 が通常のユーザー EXEC モード権限です。レベル 15 は、**enable**

	コマンドまたはアクション	目的
	スイッチ(config)# **privilege level 15**	パスワードによって許可されるアクセス レベルです。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

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

権限レベルへのログインおよび終了

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

手順の概要

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

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** *level* 例： スイッチ> **enable 15**	指定された特権レベルにログインします。 この例で、レベル 15 は特権 EXEC モードです。 *level* に指定できる範囲は 0 ～ 15 です。

	コマンドまたはアクション	目的
ステップ 2	**disable** *level* 例： スイッチ# **disable 1**	指定した特権レベルを終了します。 この例で、レベル 1 はユーザ EXEC モードです。 *level* に指定できる範囲は 0 ～ 15 です。

スイッチ アクセスのモニタリング

表 *116 : DHCP* 情報を表示するためのコマンド

show privilege	権限レベルの設定を表示します。

パスワードおよび権限レベルの設定例

例：スタティック イネーブル パスワードの設定または変更

次に、イネーブル パスワードを*l1u2c3k4y5* に変更する例を示します。パスワードは暗号化されておらず、レベル 15 のアクセスが与えられます（従来の特権 EXEC モード アクセス）。

```
スイッチ(config)# enable password l1u2c3k4y5
```

例：暗号化によるイネーブルおよびイネーブルシークレットパスワードの保護

次に、権限レベル 2 に対して暗号化パスワード *1FaD0$Xyti5Rkls3LoyxzS8* を設定する例を示します。

```
スイッチ(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

例：マスクされたシークレットパスワードの設定

次に、マスクされたシークレットパスワードを設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# username cisco masked-secret
Enter secret: ******
Confirm secret: ******
```

次に、コモンクライテリアポリシーのマスクされたシークレットパスワードを設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# username cisco common-criteria-policy test-policy masked-secret
Enter secret: ******
Confirm secret: ******
```

例：端末回線に対する Telnet パスワードの設定

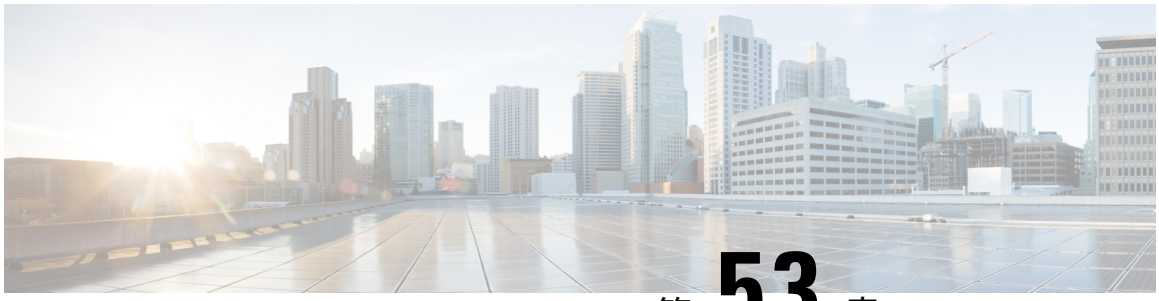
次に、Telnet パスワードを *let45me67in89* に設定する例を示します。

```
スイッチ(config)# line vty 10
スイッチ(config-line)# password let45me67in89
```

例：コマンドの権限レベルの設定

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

```
スイッチ(config)# privilege exec level 14 configure
スイッチ(config)# enable password level 14 SecretPswd14
```



第 53 章

TACACS+ の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

TACACS+ の前提条件

TACACS+によるスイッチアクセスのセットアップと設定の前提条件は、次のとおりです（示されている順序で実行する必要があります）。

1. スイッチに TACACS+ サーバー アドレスとスイッチを設定します。
2. 認証キーを設定します。
3. TACACS+ サーバでステップ 2 からキーを設定します。
4. 認証、許可、アカウンティング（AAA）をイネーブルにする。
5. ログイン認証方式リストを作成します。

6. 端末回線にリストを適用します。
7. 認証およびアカウンティング方式のリストを作成します。

TACACS+ によるスイッチ アクセスの制御の前提条件は、次のとおりです。

- スイッチ上で TACACS+ 機能を設定するには、設定済みの TACACS+ サーバーにアクセスする必要があります。また、通常 LINUX または Windows ワークステーション上で稼働する TACACS+ デーモンのデータベースで管理されている TACACS+ サービスにもアクセスする必要があります。
- スイッチ スタックと TACACS+ サーバーとの間に冗長接続を設定することを推奨します。これによって、接続済みのスタック メンバの 1 つがスイッチ スタックから削除された場合でも、TACACS+ サーバーにアクセスできます。
- スイッチで TACACS+ を使用するには、TACACS+ デーモン ソフトウェアが稼働するシステムが必要です。
- TACACS+ を使用するには、それをイネーブルにする必要があります。
- 許可は、使用するスイッチでイネーブルにする必要があります。
- ユーザは TACACS+ 許可に進む前に、まず TACACS+ 認証を正常に完了する必要があります。
- このセクションに記載されている AAA コマンドのいずれかを使用するには、まず **aaa new-model** コマンドを使用して AAA をイネーブルにする必要があります。
- 最低限、TACACS+ デーモンを維持するホスト（1 つまたは複数）を特定し、TACACS+ 認証の方式リストを定義する必要があります。また、任意で TACACS+ 許可およびアカウンティングの方式リストを定義できます。
- 方式リストは実行される認証のタイプと実行順序を定義します。このリストを特定のポートに適用してから、定義済み認証方式を実行する必要があります。唯一の例外はデフォルトの方式リスト（偶然に *default* と名前が付けられている）です。デフォルトの方式リストは、名前付き方式リストを明示的に定義されたインターフェイスを除いて、自動的にすべてのポートに適用されます。定義済みの方式リストは、デフォルトの方式リストに優先します。
- TACACS+ を使用して認証を行った場合は、TACACS+ を使用して特権 EXEC アクセスを許可します。
- 認証に TACACS+ を使用しなかった場合は、ローカル データベースを使用します。

TACACS+ の概要

TACACS+ およびスイッチ アクセス

ここでは、TACACS+ について説明します。TACACS+ は詳細なアカウンティング情報を提供し、認証と許可のプロセスを柔軟に管理します。TACACS+ は、認証、許可、アカウンティング（AAA）機能により拡張されており、TACACS+ をイネーブルにするには AAA コマンドを使用する必要があります。

（注） Cisco IOS リリース 15.2(7)E3 以降、レガシーコマンド **tacacs-server** は廃止されました。デバイスで実行されているソフトウェアが Cisco IOS リリース 15.2（7）E3 以降のリリースである場合は、**tacacs server** コマンドを使用します。

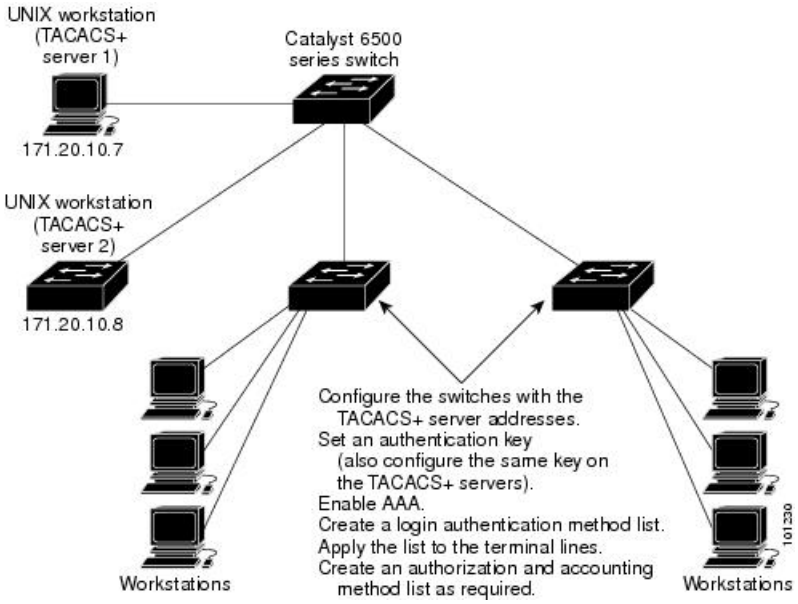
TACACS+ の概要

TACACS+ は、スイッチにアクセスしようとするユーザーの検証を集中的に行うセキュリティアプリケーションです。

TACACS+ では、独立したモジュラ型の認証、許可、アカウンティング機能が提供されます。TACACS+ では、単一のアクセス コントロール サーバ（TACACS+ デーモン）が各サービス（認証、許可、およびアカウンティング）を別個に提供します。各サービスを固有のデータベースに結合し、デーモンの機能に応じてそのサーバまたはネットワークで使用できる他のサービスを使用できます。

TACACS+ の目的は、1 つの管理サービスから複数のネットワーク アクセス ポイントを管理する方式を提供することです。スイッチは、他の Cisco ルータやアクセス サーバとともにネットワーク アクセス サーバにできます。

図 *89 :* 一般的な *TACACS+* ネットワーク構成



TACACS+ は、AAA セキュリティ サービスによって管理され、次のようなサービスを提供します。

- 認証：ログインおよびパスワードダイアログ、チャレンジおよび応答、メッセージサポートによって認証の完全制御を行います。

  認証機能は、ユーザとの対話を実行できます（たとえば、ユーザ名とパスワードが入力された後、自宅の住所、母親の旧姓、サービスタイプ、社会保険番号などのいくつかの質問をすることによりユーザを試します）。TACACS+認証サービスは、ユーザ画面にメッセージを送信することもできます。たとえば、会社のパスワード有効期間ポリシーに従い、パスワードの変更の必要があることをユーザに通知することもできます。

- 許可：autocommand、アクセス コントロール、セッション期間、プロトコル サポートの設定といった、ユーザー セッション時のユーザー機能についてきめ細かく制御します。また、TACACS+ 許可機能によって、ユーザーが実行できるコマンドを制限することもできます。

- アカウンティング：課金、監査、およびレポートに使用する情報を収集して TACACS+ デーモンに送信します。ネットワークの管理者は、アカウンティング機能を使用して、セキュリティ監査のためにユーザの活動状況をトラッキングしたり、ユーザ課金用の情報を提供したりできます。アカウンティング レコードには、ユーザ ID、開始時刻および終了時刻、実行されたコマンド（PPP など）、パケット数、およびバイト数が含まれます。

TACACS+プロトコルは、スイッチと TACACS+デーモン間の認証を行い、スイッチと TACACS+ デーモン間のプロトコル交換をすべて暗号化することによって機密保持を実現します。

TACACS+ の動作

ユーザーが、TACACS+ を使用しているスイッチに対して簡易 ASCII ログインを試行し、認証が必要になると、次のプロセスが発生します。

1. 接続が確立されると、スイッチは TACACS+ デーモンに接続してユーザー名プロンプトを取得し、これをユーザーに表示します。ユーザーがユーザー名を入力すると、スイッチは TACACS+ デーモンに接続してパスワード プロンプトを取得します。スイッチによってパスワードプロンプトが表示され、ユーザーがパスワードを入力すると、そのパスワードが TACACS+ デーモンに送信されます。

   TACACS+ によって、デーモンとユーザとの間の対話が可能になり、デーモンはユーザを認証できるだけの情報を取得できるようになります。デーモンは、ユーザ名とパスワードの組み合わせを入力するよう求めますが、ユーザの母親の旧姓など、その他の項目を含めることもできます。

2. スイッチは、最終的に TACACS+ デーモンから次のいずれかの応答を得ます。

   - ACCEPT：ユーザーが認証され、サービスを開始できます。許可を必要とするようにスイッチが設定されている場合は、この時点で許可処理が開始されます。
   - REJECT：ユーザーは認証されません。TACACS+ デーモンに応じて、ユーザーはアクセスを拒否されるか、ログイン シーケンスを再試行するように求められます。
   - ERROR：デーモンによる認証サービスのある時点で、またはデーモンとスイッチの間のネットワーク接続においてエラーが発生しました。ERROR 応答が表示された場合は、スイッチは、通常別の方法でユーザーを認証しようとします。
   - CONTINUE：ユーザーは、さらに認証情報の入力を求められます。

   認証後、スイッチで許可がイネーブルになっている場合、ユーザーは追加の許可フェーズに入ります。ユーザは TACACS+ 許可に進む前に、まず TACACS+ 認証を正常に完了する必要があります。

3. TACACS+ 許可が必要な場合は、再び TACACS+ デーモンに接続し、デーモンが ACCEPT または REJECT の許可応答を返します。ACCEPT 応答が返された場合は、その応答に、そのユーザおよびそのユーザがアクセスできるサービスの、EXEC または NETWORK セッション宛ての属性の形式でデータが含まれています。

   - Telnet、セキュア シェル（SSH）、rlogin、または特権 EXEC サービス
   - 接続パラメータ（ホストまたはクライアントの IP アドレス、アクセス リスト、およびユーザ タイムアウトを含む）

方式リスト

方式リストによって、ユーザの認証、許可、またはアカウント維持のための順序と方式を定義します。方式リストを使用して、使用するセキュリティ プロトコルを1つまたは複数指定できるので、最初の方式が失敗した場合のバックアップ システムが確保されます。ソフトウェア

は、リスト内の最初の方式を使用してユーザの認証、許可、アカウントの維持を行います。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の方式を選択します。このプロセスは、リスト内の方式による通信が成功するか、方式リストの方式をすべて試し終わるまで続きます。

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

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

次の例は、AAA で方式リストを設定する方法を示しています。

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

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

```
Device# configure terminal
Device(config)# line vty 0 4
```

次の例は、デフォルトの方式リストを設定する方法を示しています。

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

TACACS+ 設定オプション

認証用に 1 つのサーバーを使用することも、また、既存のサーバーホストをグループ化するために AAA サーバー グループを使用するように設定することもできます。サーバをグループ化して設定済みサーバホストのサブセットを選択し、特定のサービスにそのサーバを使用できます。サーバ グループは、グローバル サーバ ホスト リストとともに使用され、選択されたサーバ ホストの IP アドレスのリストが含まれています。

TACACS+ ログイン認証

方式リストは、ユーザ認証のためクエリ送信を行う手順と認証方式を記述したものです。認証に使用する 1 つまたは複数のセキュリティ プロトコルを指定できるので、最初の方式が失敗した場合のバックアップ システムが確保されます。ソフトウェアは、リスト内の最初の方式を使用してユーザを認証します。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の認証方式を選択します。このプロセスは、リスト内の認証方式による通信が成功するか、定義された方式をすべて試し終わるまで繰り返されます。この処理のある時点で認証が失敗した場合（つまり、セキュリティ サーバまたはローカルのユーザ名データベースがユーザ アクセスを拒否すると応答した場合）、認証プロセスは停止し、それ以上認証方式が試行されることはありません。

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

AAA 許可によってユーザが使用できるサービスが制限されます。AAA 許可がイネーブルに設定されていると、スイッチはユーザーのプロファイルから取得した情報を使用します。このプロファイルは、ローカルのユーザーデータベースまたはセキュリティ サーバー上にあり、ユーザーのセッションを設定します。ユーザは、ユーザプロファイル内の情報で認められている場合に限り、要求したサービスのアクセスが認可されます。

TACACS+ Accounting

AAA アカウンティング機能は、ユーザがアクセスしたサービスと、消費したネットワーク リソース量をトラッキングします。AAA アカウンティングをイネーブルにすると、スイッチはユーザーの活動状況をアカウンティング レコードの形式で TACACS+ セキュリティ サーバーに報告します。各アカウンティング レコードにはアカウンティングの Attribute-Value（AV）ペアが含まれ、レコードはセキュリティ サーバに格納されます。このデータを、ネットワーク管理、クライアント請求、または監査のために分析できます。

TACACS+ のデフォルト設定

TACACS+ および AAA は、デフォルトではディセーブルに設定されています。

セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して TACACS+ を設定することはできません。TACACS+ をイネーブルに設定した場合、CLI を通じてスイッチにアクセスするユーザを認証できます。

（注） TACACS+ の設定は CLI を使用して行いますが、TACACS+ サーバは権限レベル 15 に設定された HTTP 接続を許可します。

TACACS+ とスイッチ アクセスを設定する方法

ここでは、TACACS+ をサポートするようにスイッチを設定する方法について説明します。

TACACS+ サーバ ホストの指定および認証キーの設定

TACACS+ サーバー ホストを特定し、認証キーを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **tacacs server** *servername*
4. **aaa new-model**

5. **aaa group server tacacs+** *group-name*
6. **server** *ip-address*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**tacacs server** *servername* 例： `スイッチ(config)# tacacs server yourserver`	TACACS+ サーバを維持する IP ホストを特定します。このコマンドを複数回入力して、優先ホストのリストを作成します。ソフトウェアは、指定された順序でホストを検索します。
ステップ 4	**aaa new-model** 例： `スイッチ(config)# aaa new-model`	AAA をイネーブルにします。
ステップ 5	**aaa group server tacacs+** *group-name* 例： `スイッチ(config)# aaa group server tacacs+ your_server_group`	（任意）グループ名で AAA サーバグループを定義します。 このコマンドによって、スイッチをサーバー グループ サブコンフィギュレーション モードにします。
ステップ 6	**server** *ip-address* 例： `スイッチ(config)# server 10.1.2.3`	（任意）特定の TACACS+ サーバを定義済みサーバ グループに関連付けます。AAA サーバ グループの TACACS+ サーバごとに、このステップを繰り返します。 グループの各サーバーは、ステップ 3 で定義済みのものでなければなりません。

	コマンドまたはアクション	目的
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

TACACS+ ログイン認証の設定

TACACS+ ログイン認証を設定するには、次の手順を実行します。

始める前に

AAA 認証を設定するには、認証方式の名前付きリストを作成してから、各種ポートにそのリストを適用します。

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

ip http authentication コマンドの詳細については、『*Cisco IOS Security Command Reference, Release 12.4*』を参照してください。

手順の概要

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

7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa new-model** 例： `スイッチ(config)# aaa new-model`	AAA をイネーブルにします。
ステップ 4	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] 例： `スイッチ(config)# aaa authentication login default tacacs+ local`	ログイン認証方式リストを作成します。 • **login authentication** コマンドにリストが指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。 • *list-name* には、作成するリストの名前として使用する文字列を指定します。 • *method1...* には、認証アルゴリズムが試行する実際の方式を指定します。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。 次のいずれかの方式を選択します。 • *enable*：イネーブル パスワードを認証に使用します。この認証方式を使用するには、あらかじめ **enable** *password* グローバル コンフィギュレー

	コマンドまたはアクション	目的
		ション コマンドを使用してイネーブルパスワードを定義しておく必要があります。 • *group tacacs+*：TACACS+ 認証を使用します。この認証方式を使用するには、あらかじめ TACACS+ サーバーを設定しておく必要があります。 • *line*：回線パスワードを認証に使用します。この認証方式を使用するには、あらかじめ回線パスワードを定義しておく必要があります。**password** *password* ライン コンフィギュレーション コマンドを使用します。 • *local*：ローカル ユーザー名データベースを認証に使用します。データベースにユーザ名情報を入力しておく必要があります。**username** *password* グローバル コンフィギュレーション コマンドを使用します。 • *local-case*：大文字と小文字が区別されるローカル ユーザー名データベースを認証に使用します。**username** *name* **password** グローバル コンフィギュレーション コマンドを使用して、ユーザー名情報をデータベースに入力する必要があります。 • *none*：ログインに認証を使用しません。
ステップ 5	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] 例： `スイッチ(config)# line 2 4`	ライン コンフィギュレーション モードを開始し、認証リストを適用する回線を設定します。
ステップ 6	**login authentication** {**default** | *list-name*} 例： `スイッチ(config-line)# login authentication default`	1 つの回線または複数回線に認証リストを適用します。 • **default** を指定する場合は、**aaa authentication login** コマンドで作成したデフォルトのリストを使用します。 • *list-name* には、**aaa authentication login** コマンドで作成したリストを指定します。
ステップ 7	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config-line)# **end**	
ステップ 8	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

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

aaa authorization グローバル コンフィギュレーション コマンドと **tacacs+** キーワードを使用すると、ユーザのネットワークアクセスを特権EXECモードに制限するパラメータを設定できます。

（注） 許可が設定されていても、CLIを使用してログインし、認証されたユーザに対しては、許可は省略されます。

特権 EXEC アクセスおよびネットワーク サービスに関する TACACS+ 許可を指定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa authorization network tacacs+**
4. **aaa authorization exec tacacs+**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa authorization network tacacs+** 例： `スイッチ(config)# aaa authorization network tacacs+`	ネットワーク関連のすべてのサービス要求に対してユーザー TACACS+ 認可を行うことを設定します。
ステップ 4	**aaa authorization exec tacacs+** 例： `スイッチ(config)# aaa authorization exec tacacs+`	ユーザーの特権 EXEC アクセスに対してユーザー TACACS+ 許可を行うことを設定します。 **exec** キーワードを指定すると、ユーザープロファイル情報（**autocommand** 情報など）が返される場合があります。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

TACACS+ アカウンティングの起動

TACACS+ アカウンティングを開始するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa accounting network start-stop tacacs+**
4. **aaa accounting exec start-stop tacacs+**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa accounting network start-stop tacacs+** 例： `スイッチ(config)# aaa accounting network start-stop tacacs+`	ネットワーク関連のすべてのサービス要求について、TACACS+ アカウンティングをイネーブルにします。
ステップ 4	**aaa accounting exec start-stop tacacs+** 例： `スイッチ(config)# aaa accounting exec start-stop tacacs+`	TACACS+ アカウンティングをイネーブルにして、特権 EXEC プロセスの最初に記録開始アカウンティング通知、最後に記録停止通知を送信します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

次のタスク

AAA サーバが到達不能な場合にルータとのセッションを確立するには、**aaa accounting system guarantee-first** コマンドを使用します。これは、最初のレコードとしてシステム アカウンティングを保証します（これがデフォルトの条件です）。場合によっては、システムがリロードされるまでコンソールまたは端末接続でセッションを開始できない場合があります。システムのリロードにかかる時間は 3 分を超えることがあります。

ルータのリロード時に AAA サーバが到達不能な場合に、ルータとのコンソールセッションまたは Telnet セッションを確立するには、**no aaa accounting system guarantee-first** コマンドを使用します。

AAA サーバが到達不能な場合のルータとのセッションの確立

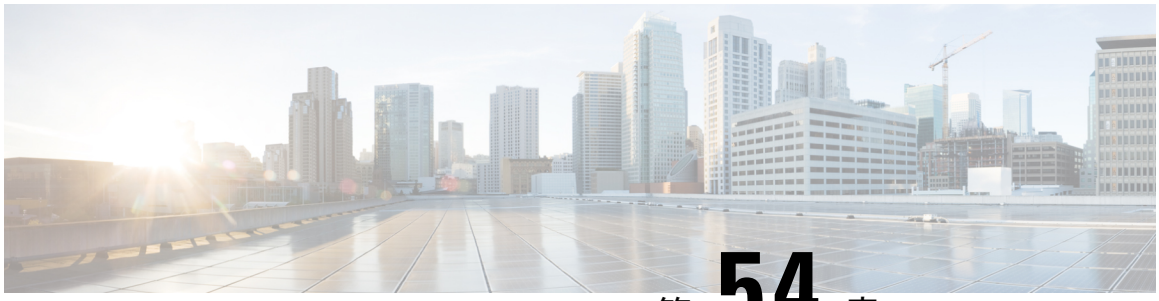
AAA サーバーが到達不能な場合にルータとのセッションを確立するには、**aaa accounting system guarantee-first** コマンドを使用します。これは、最初のレコードとしてシステム アカウンティングを保証します（これがデフォルトの条件です）。場合によっては、システムがリロードされるまでコンソールまたは端末接続でセッションを開始できない場合があります。システムのリロードにかかる時間は 3 分を超えることがあります。

ルータのリロード時に AAA サーバが到達不能な場合に、ルータとのコンソールセッションまたは Telnet セッションを確立するには、**no aaa accounting system guarantee-first** コマンドを使用します。

TACACS+ のモニタリング

表 *117 : TACACS+* 情報を表示するためのコマンド

コマンド	目的
show tacacs	TACACS+ サーバの統計情報を表示します。



第 54 章

RADIUS の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

RADIUS を設定するための前提条件

ここでは、RADIUS による スイッチ アクセスの制御の前提条件を示します。

全般：

- この章のいずれかのコンフィギュレーション コマンドを使用するには、RADIUSおよび認証、許可、ならびにアカウンティング（AAA）をイネーブルにする必要があります。
- RADIUS は、AAA を介して実装され、AAA コマンドを使用してのみイネーブルにできます。

- **aaa new-model** グローバル コンフィギュレーション コマンドを使用して、AAA をイネーブルにします。
- **aaa authentication** グローバル コンフィギュレーション コマンドを使用して、RADIUS 認証の方式リストを定義します。
- **line** および **interface** コマンドを使用して、使用する定義済みの方式リストをイネーブルにします。
- 最低限、RADIUS サーバ ソフトウェアが稼働するホスト（1 つまたは複数）を特定し、RADIUS 認証の方式リストを定義する必要があります。また、任意で RADIUS 許可およびアカウンティングの方式リストを定義できます。
- スイッチ 上で RADIUS 機能の設定を行う前に、RADIUS サーバにアクセスし、サーバを設定する必要があります。
- RADIUS ホストは、通常、シスコ（Cisco Secure Access Control Server バージョン 3.0）、Livingston、Merit、Microsoft、または他のソフトウェア プロバイダーの RADIUS サーバ ソフトウェアが稼働しているマルチユーザ システムです。詳細については、RADIUS サーバのマニュアルを参照してください。
- Change-of-Authorization（CoA）インターフェイスを使用するには、スイッチにセッションがすでに存在している必要があります。CoA を使用すると、セッションの識別と接続解除要求を実行できます。アップデートは、指定されたセッションにだけ作用します。

RADIUS 操作の場合：

- ユーザは RADIUS 許可に進む前に、まず RADIUS 認証を正常に完了する必要があります（イネーブルに設定されている場合）。

RADIUS の設定に関する制約事項

ここでは、RADIUS による スイッチ アクセスの制御の制約事項について説明します。

全般：

- セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して RADIUS を設定することはできません。

RADIUS は次のネットワーク セキュリティ状況には適していません。

- マルチプロトコル アクセス環境。RADIUS は、AppleTalk Remote Access（ARA）、NetBIOS Frame Control Protocol（NBFCP）、NetWare Asynchronous Services Interface（NASI）、または X.25 PAD 接続をサポートしません。
- スイッチ間またはルータ間状態。RADIUS は、双方向認証を行いません。RADIUS は、他社製のデバイスが認証を必要とする場合に、あるデバイスから他社製のデバイスへの認証に使用できます。

- 各種のサービスを使用するネットワーク。RADIUS は、一般に 1 人のユーザを 1 つのサービス モデルにバインドします。

RADIUS に関する情報

RADIUS およびスイッチ アクセス

この項では、RADIUS をイネーブルにし、設定する方法について説明します。RADIUS を使用すると、アカウンティングの詳細を取得したり、認証および許可プロセスの柔軟な管理制御を実現できます。

RADIUS の概要

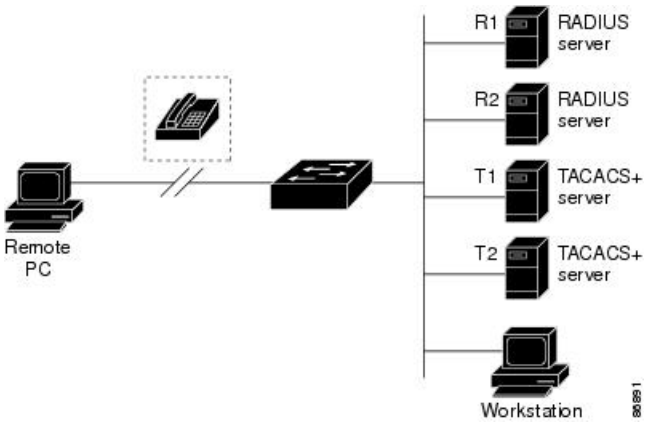
RADIUS は、不正なアクセスからネットワークのセキュリティを保護する分散クライアント/サーバ システムです。RADIUS クライアントは、サポート対象の Cisco ルータおよびスイッチ上で稼働します。クライアントは中央の RADIUS サーバに認証要求を送ります。中央の RADIUS サーバにはすべてのユーザ認証情報、ネットワーク サービス アクセス情報が登録されています。

RADIUS は、アクセスのセキュリティが必要な、次のネットワーク環境で使用します。

- それぞれが RADIUS をサポートする、マルチベンダー アクセス サーバによるネットワーク。たとえば、複数のベンダーのアクセス サーバが、1 つの RADIUS サーバベース セキュリティ データベースを使用します。複数ベンダーのアクセス サーバからなる IP ベースのネットワークでは、ダイヤルイン ユーザは RADIUS サーバを通じて認証されます。RADIUS サーバは、Kerberos セキュリティ システムで動作するようにカスタマイズされています。
- アプリケーションが RADIUS プロトコルをサポートするターンキー ネットワーク セキュリティ環境。たとえば、スマート カード アクセス コントロール システムを使用するアクセス環境。あるケースでは、RADIUS は Enigma のセキュリティ カードとともに使用してユーザを確認し、ネットワーク リソースのアクセスを許可します。
- すでに RADIUS を使用中のネットワーク。RADIUS クライアント装備のシスコ スイッチ ネットワークに追加できます。これが TACACS+ サーバへの移行の最初のステップとなることもあります。下の図 「RADIUS サービスから TACACS+ サービスへの移行」を参照してください。
- ユーザが 1 つのサービスにしかアクセスできないネットワーク。RADIUS を使用すると、ユーザのアクセスを 1 つのホスト、Telnet などの 1 つのユーティリティ、または IEEE 802.1x などのプロトコルを使用するネットワークに制御できます。このプロトコルの詳細については、「IEEE 802.1x ポートベース認証の設定」の章を参照してください。
- リソース アカウンティングが必要なネットワーク。RADIUS 認証または許可とは別個に RADIUS アカウンティングを使用できます。RADIUS アカウンティング機能によって、サービスの開始および終了時点でデータを送信し、このセッション中に使用されるリソー

ス（時間、パケット、バイトなど）の量を表示できます。インターネット サービス プロバイダーは、RADIUS アクセス コントロールおよびアカウンティング ソフトウェアのフリーウェアバージョンを使用して、特殊なセキュリティおよび課金に対するニーズを満たすこともできます。

図 *90 : RADIUS* サービスから *TACACS+* サービスへの移行



RADIUS の動作

RADIUS サーバーによってアクセスコントロールされる スイッチ に、ユーザーがログインおよび認証を試みると、次のイベントが発生します。

1. ユーザ名およびパスワードの入力を要求するプロンプトが表示されます。
2. ユーザ名および暗号化されたパスワードが、ネットワーク経由でRADIUSサーバに送信されます。
3. ユーザは、RADIUS サーバから次のいずれかの応答を受信します。
   - ACCEPT：ユーザーが認証されたことを表します。
   - REJECT：ユーザーの認証が失敗し、ユーザー名およびパスワードの再入力が要求されるか、またはアクセスが拒否されます。
   - CHALLENGE：ユーザーに追加データを要求します。
   - CHALLENGE PASSWORD：ユーザーは新しいパスワードを選択するように要求されます。

   ACCEPT または REJECT 応答には、特権 EXEC またはネットワーク許可に使用する追加データがバンドルされています。ACCEPT または REJECT パケットには次の追加データが含まれます。

   - Telnet、SSH、rlogin、または特権 EXEC サービス
   - 接続パラメータ（ホストまたはクライアントのIPアドレス、アクセスリスト、およびユーザ タイムアウトを含む）

RADIUS 許可の変更

RADIUS 許可の変更（CoA）は、認証、認可、およびアカウンティング（AAA）セッションの属性を認証された後に変更するためのメカニズムを提供します。AAA でユーザー、またはユーザー グループのポリシーが変更された場合、管理者は、AAA サーバーから Cisco Secure Access Control Server（ACS）などの RADIUS CoA パケットを送信し、認証を再初期化して新しいポリシーを適用することができます。このセクションでは、使用可能なプリミティブおよびそれらの CoA での使用方法を含む、RADIUS インターフェイスの概要について説明します。

- Change-of-Authorization 要求
- CoA 要求応答コード
- CoA 要求コマンド
- セッション再認証
- セッション強制終了のスタック構成ガイドライン

標準 RADIUS インターフェイスは通常、ネットワークに接続しているデバイスから要求が送信され、クエリーが送信されたサーバーが応答するプル モデルで使用されます。Catalyst は、RFC 5176 で規定された（通常はプッシュモデルで使用される）RADIUS CoA 拡張機能をサポートし、外部の AAA またはポリシーサーバーからのセッションを動的に再設定できるようにします。

は、次のセッション単位の CoA 要求をサポートしています。

- セッション再認証
- セッションの終了
- ポート シャットダウンでのセッション終了
- ポート バウンスでのセッション終了

この機能は、Cisco Secure Access Control Server (ACS) 5.1 に統合されています。

Catalyst で、RADIUS インターフェイスはデフォルトでイネーブルに設定されています。ただし、次の属性については、一部の基本的な設定が必要になります。

- セキュリティおよびパスワード：このガイドの「スイッチへの不正アクセスの防止」を参照してください。
- アカウンティング：このガイドの「スイッチベース認証の設定」の章の「RADIUS アカウンティングの起動」の項を参照してください。

Cisco IOS ソフトウェアは、RFC 5176 で定義されている RADIUS CoA の拡張をサポートします。この拡張は、一般に、外部 AAA またはポリシー サーバーからのセッションのダイナミックな再構成を可能にするプッシュ モデルで使用されます。セッションの特定、セッションの終了、ホストの再認証、ポートのシャットダウン、およびポートバウンスでは、セッションごとの CoA 要求がサポートされます。このモデルは、次のように、1 つの要求（CoA-Request）と 2 つの考えられる応答コードで構成されます。

- CoA acknowledgement（ACK）[CoA-ACK]
- CoA nonacknowledgement（NAK）[CoA-NAK]

要求はCoAクライアント（通常はAAAまたはポリシーサーバー）から開始されて、リスナーとして動作するデバイスに転送されます。

次の表は、Identity-Based Networking Services でサポートされている RADIUS CoA コマンドとベンダー固有属性（VSA）を示します。すべての CoA コマンドには、デバイスと CoA クライアント間のセッション ID が含まれている必要があります。

表 *118 : Identity-Based Networking Services* でサポートされている *RADIUS CoA* コマンド

CoA コマンド	**シスコの VSA**
Activate service	Cisco:Avpair="subscriber:command=activate-service" Cisco:Avpair="subscriber:service-name=<*service-name*>" Cisco:Avpair="subscriber:precedence=<*precedence-number*>" Cisco:Avpair="subscriber:activation-mode=replace-all"
Deactivate service	Cisco:Avpair="subscriber:command=deactivate-service" Cisco:Avpair="subscriber:service-name=<*service-name*>"
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"
Session query	Cisco:Avpair="subscriber:command=session-query"
Session reauthenticate	Cisco:Avpair="subscriber:command=reauthenticate" Cisco:Avpair="subscriber:reauthenticate-type=last" または Cisco:Avpair="subscriber:reauthenticate-type=rerun"
Session terminate	これは、VSA を必要としない、標準の接続解除要求です。
Interface template	Cisco:AVpair="interface-template-name=<*interfacetemplate*>"

Change-of-Authorization 要求

Change of Authorization（CoA）要求は、RFC 5176 に記載されているように、プッシュ モデルで使用することによって、セッション識別、ホスト再認証、およびセッション終了を行うことができます。このモデルは、1 つの要求（CoA-Request）と 2 つの可能な応答コードで構成されています。

- CoA acknowledgment（ACK）[CoA-ACK]
- CoA non-acknowledgment（NAK）[CoA-NAK]

要求は CoA クライアント（通常は RADIUS またはポリシー サーバー）から発信されて、リスナーとして動作するスイッチに送信されます。

RFC 5176 規定

Disconnect Request メッセージは Packet of Disconnect（POD）とも呼ばれますが、セッション終了に対してスイッチでサポートされています。

次の表に、この機能でサポートされている IETF 属性を示します。

表 *119 :* サポートされている *IETF* 属性

属性番号	属性名
24	状態
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

次の表に、Error-Cause 属性で取ることができる値を示します。

表 *120 : Error-Cause* の値

値	説明
201	削除された残留セッション コンテキスト
202	無効な EAP パケット（無視）
401	サポートされていない属性
402	見つからない属性
403	NAS 識別情報のミスマッチ
404	無効な要求
405	サポートされていないサービス
406	サポートされていない拡張機能
407	無効な属性値
501	管理上の禁止
502	ルート不可能な要求（プロキシ）

値	説明
503	セッション コンテキストが検出されない
504	セッション コンテキストが削除できない
505	その他のプロキシ処理エラー
506	リソースが使用不可能
507	要求が発信された
508	マルチ セッションの選択がサポートされてない

CoA 要求応答コード

CoA 要求応答コードを使用すると、スイッチにコマンドを伝達できます。

RFC 5176 で定義されている CoA 要求応答コードのパケットの形式は、コード、ID、長さ、オーセンティケータ、およびタイプ、長さ、値（TLV）形式の属性から構成されます。属性フィールドは、シスコのベンダー固有属性（VSA）を送信するために使用します。

セッションの識別

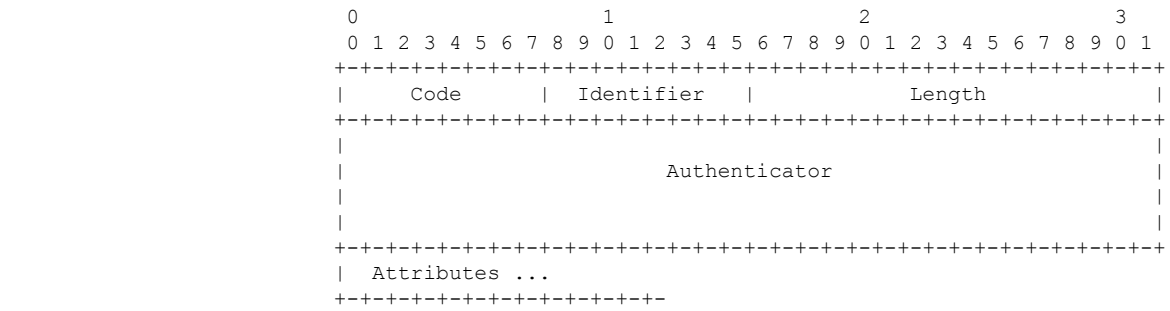
特定のセッションに向けられた切断と CoA 要求については、スイッチは 1 つ以上の次の属性に基づいて、セッションを検索します。

- Acct-Session-Id（IETF 属性 #44）
- Audit-Session-Id VSA（シスコの VSA）
- Calling-Station-Id（ホスト MAC アドレスを含む IETF 属性 #31）
- 次のいずれかの IPv6 属性。
  - Framed-IPv6-Prefix（IETF 属性 #97）および Framed-Interface-Id（IETF 属性 #96）。ともに RFC 3162 に従った完全な IPv6 アドレスを作成する
  - Framed-IPv6-Address
- プレーン IP アドレス（IETF 属性 #8）

CoA メッセージに含まれるすべてのセッション ID 属性がそのセッションと一致しない限り、スイッチは「Invalid Attribute Value」エラー コード属性を含む Disconnect-NAK または CoA-NAK を返します。

複数のセッション ID 属性がメッセージ含まれる場合は、すべての属性がセッションと一致しなければなりません。そうでない場合は、スイッチが Disconnect - negative acknowledgement（NAK）または CoA -NAK と、「Invalid Attribute Value」エラー コードを返します。

RFC 5176 で定義されている CoA 要求コードのパケットの形式は、コード、ID、長さ、オーセンティケータ、およびタイプ、長さ、値（TLV）形式の属性から構成されます。



属性フィールドは、シスコのベンダー固有属性（VSA）を送信するために使用します。

特定の適用ポリシーを対象とする CoA 要求の場合、上記のセッション ID 属性のいずれかがメッセージに含まれていると、デバイスはエラー コードが「Invalid Attribute Value」の CoA-NAK を返します。

CoA ACK 応答コード

許可ステートの変更に成功した場合は、肯定確認応答（ACK）が送信されます。CoA ACK 内で返される属性は CoA 要求によって異なり、個々の CoA コマンドで検討されます。

CoA NAK 応答コード

否定応答（NAK）は許可ステートの変更に失敗したことを示し、エラーの理由を示す属性を含めることができます。CoA が成功したかを確認するには、**show** コマンドを使用します。

CoA 要求コマンド

表 *121 :* でサポートされる *CoA* コマンド

コマンド [8]	シスコの VSA
Reauthenticate host	Cisco:Avpair="subscriber:command=reauthenticate"
Terminate session	これは、VSA を要求しない、標準の接続解除要求です。
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"

[8] すべての CoA コマンドには、と CoA クライアント間のセッション識別情報が含まれている必要があります。

セッション再認証

不明な ID またはポスチャを持つホストがネットワークに加入して、制限されたアクセス許可プロファイル（たとえば、ゲスト VLAN）に関連付けられると、AAA サーバーは通常、セッ

ション再認証要求を生成します。再認証要求は、クレデンシャルが不明である場合にホストが適切な認証グループに配置されることを許可します。

セッション認証を開始するために、AAA サーバーは *Cisco:Avpair="subscriber:command=reauthenticate"* の形式で Cisco VSA と 1 つ以上のセッション ID 属性を含む標準 CoA 要求メッセージを送信します。

現在のセッション ステートは、メッセージに対するスイッチの応答を決定します。セッションが現在、IEEE 802.1x によって認証されている場合、スイッチは EAPOL（LAN 経由の拡張認証プロトコル）RequestId メッセージをサーバーに送信することで応答します。

現在、セッションが MAC 認証バイパス（MAB）で認証されている場合は、スイッチはサーバーにアクセス要求を送信し、初期正常認証で使用されるものと同じ ID 属性を渡します。

スイッチがコマンドを受信した際にセッション認証が実行中である場合は、スイッチはプロセスを終了し、認証シーケンスを再開し、最初に試行されるように設定された方式で開始します。

セッションがまだ認証されてない、あるいはゲスト VLAN、クリティカル VLAN、または同様のポリシーで認証されている場合は、再認証メッセージがアクセス コントロール方式を再開し、最初に試行されるように設定された方式で開始します。セッションの現在の許可は、再認証によって異なる認証結果になるまで維持されます。

セッションの終了

セッションを終了させる 3 種類の CoA 要求があります。CoA 接続解除要求は、ホスト ポートをディセーブルにせずにセッションを終了します。このコマンドを使用すると、指定されたホストのオーセンティケータ ステート マシンが再初期化されますが、そのホストのネットワークへのアクセスは制限されません。

ホストのネットワークへのアクセスを制限するには、Cisco:Avpair="subscriber:command=disable-host-port" VSA の設定で CoA 要求を使用します。このコマンドは、ネットワーク上で障害を引き起こしたと認識されているホストがある場合に便利であり、そのホストに対してネットワーク アクセスをただちにブロックする必要があります。ポートへのネットワーク アクセスを復旧する場合は、非 RADIUS メカニズムを使用して再びイネーブルにします。

プリンタなどのサプリカントを持たないデバイスが新しい IP アドレスを取得する必要がある場合（たとえば、VLAN 変更後）は、ポート バウンスでホスト ポート上のセッションを終了します（ポートを一時的にディセーブルした後、再びイネーブルにする）。

CoA 接続解除要求

このコマンドは標準の接続解除要求です。セッションが見つからない場合、スイッチは Disconnect-NAK メッセージと「Session Context Not Found」エラー コード属性を返します。セッションがある場合は、スイッチはセッションを終了します。セッションが完全に削除された後、スイッチは接続解除 ACK を返します。

スイッチがクライアントに接続解除 ACK を返す前にスタンバイ スイッチにフェールオーバーする場合は、クライアントから要求が再送信される際に、新しいアクティブ スイッチ上でその

プロセスが繰り返されます。再送信後もセッションが見つからない場合は、Disconnect-ACK と「Session Context Not Found」エラー コード属性が送信されます。

CoA 要求：ホスト ポートのディセーブル化

RADIUS サーバーの CoA disable port コマンドを実行すると、セッションをホストしている認証ポートが管理的にシャットダウンされます。その結果、セッションは終了します。このコマンドは、ホストがネットワーク上で問題を起こしていることを把握し、ホストのネットワークアクセスを即座にブロックする必要がある場合に便利です。ポートのネットワーク アクセスを復元するには、非RADIUSメカニズムを使用して再びイネーブルにします。このコマンドは、次の新しいベンダー固有属性（VSA）が含まれている標準 CoA 要求メッセージで伝達されます。

Cisco:Avpair="subscriber:command=disable-host-port"

このコマンドはセッション指向であるため、「セッション ID」セクションに示されている 1 つ以上のセッション ID 属性とともに使用する必要があります。セッションが見つからない場合、スイッチはCoA-NAKメッセージと「Session Context Not Found」エラー コード属性を返します。このセッションがある場合は、スイッチはホストポートをディセーブルにし、CoA-ACK メッセージを返します。

スイッチが CoA-ACK をクライアントに返す前にスイッチに障害が発生した場合は、クライアントから要求が再送信される際に、新しいアクティブ スイッチ上でそのプロセスが繰り返されます。スイッチが CoA-ACK メッセージをクライアントに返した後で、かつその動作が完了していないときにスイッチに障害が発生した場合は、新しいアクティブ スイッチ上でその動作が再開されます。

（注） 再送信コマンドの後に接続解除要求が失敗すると、（接続解除 ACK が送信されてない場合に）チェンジオーバー前にセッションが正常終了し、または元のコマンドが実行されてスタンバイ スイッチがアクティブになるまでの間に発生した他の方法（たとえば、リンク障害）によりセッションが終了することがあります。

CoA 要求：バウンス ポート

RADIUS サーバーの CoA bounce port が RADIUS サーバーから送信されると、認証ポートでリンクのフラップが発生します。その結果、このポートに接続している1つまたは複数のホストから、DHCP の再ネゴシエーションが開始されます。この状況は、VLAN の変更があり、この認証ポートに関する変化を検出するメカニズムがないデバイス（プリンタなど）がエンドポイントの場合に発生する可能性があります。CoA bounce port は、次の新しい VSA を含む標準の CoA-Request メッセージで伝達されます。

Cisco:Avpair="subscriber:command=bounce-host-port"

このコマンドはセッション指向であるため、1 つ以上のセッション ID 属性とともに使用する必要があります。セッションが見つからない場合、スイッチはCoA-NAKメッセージと「Session Context Not Found」エラー コード属性を返します。このセッションがある場合は、スイッチはホストポートを10秒間ディセーブルし、再びイネーブルにし（ポートバウンス）、CoA-ACK を返します。

スイッチが CoA-ACK をクライアントに返す前にスイッチに障害が発生した場合は、クライアントから要求が再送信される際に、新しいアクティブ スイッチ上でそのプロセスが繰り返されます。スイッチが CoA-ACK メッセージをクライアントに返した後で、かつその動作が完了していないときにスイッチに障害が発生した場合は、新しいアクティブ スイッチ上でその動作が再開されます。

RADIUS のデフォルト設定

RADIUS および AAA は、デフォルトではディセーブルに設定されています。

セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して RADIUS を設定することはできません。RADIUS をイネーブルに設定した場合、CLI を通じてスイッチにアクセスするユーザを認証できます。

RADIUS サーバ ホスト

スイッチと RADIUS サーバの通信には、次の要素が関係します。

- ホスト名または IP アドレス
- 認証の宛先ポート
- アカウンティングの宛先ポート
- キー文字列
- タイムアウト時間
- 再送信回数

RADIUS セキュリティ サーバは、ホスト名または IP アドレス、ホスト名と特定の UDP ポート番号、または IP アドレスと特定の UDP ポート番号によって特定します。IP アドレスと UDP ポート番号の組み合わせによって、一意の ID が作成され、特定の AAA サービスを提供する RADIUS ホストとして個々のポートを定義できます。この一意の ID を使用することによって、同じ IP アドレスにあるサーバ上の複数の UDP ポートに、RADIUS 要求を送信できます。

同じ RADIUS サーバー上の異なる 2 つのホスト エントリに同じサービス（たとえばアカウンティング）を設定した場合、2 番めに設定したホストエントリは、最初に設定したホストエントリのフェールオーバー バックアップとして動作します。この例では、最初のホスト エントリがアカウンティング サービスを提供できなかった場合、スイッチは「%RADIUS-4-RADIUS_DEAD」メッセージを表示し、その後、同じデバイス上で 2 番めに設定されたホスト エントリでアカウンティング サービスを試みます（RADIUS ホスト エントリは、設定した順序に従って試行されます）。

RADIUS サーバとスイッチは、共有秘密テキスト文字列を使用して、パスワードの暗号化および応答の交換を行います。RADIUS で AAA セキュリティ コマンドを使用するように設定するには、RADIUS サーバデーモンが稼働するホストと、そのホストがスイッチと共有する秘密テキスト（キー）文字列を指定する必要があります。

タイムアウト、再送信回数、および暗号キーの値は、すべてのRADIUSサーバに対してグローバルに設定することもできますし、サーバ単位で設定することもできます。また、グローバルな設定とサーバ単位での設定を組み合わせることもできます。

RADIUS ログイン認証

AAA 認証を設定するには、認証方式の名前付きリストを作成してから、各種ポートにそのリストを適用します。方式リストは実行される認証のタイプと実行順序を定義します。このリストを特定のポートに適用してから、定義済み認証方式を実行する必要があります。唯一の例外は、デフォルトの方式リストです。デフォルトの方式リストは、名前付き方式リストを明示的に定義されたインターフェイスを除いて、自動的にすべてのポートに適用されます。

方式リストは、ユーザ認証のためクエリ送信を行う手順と認証方式を記述したものです。認証に使用する1つまたは複数のセキュリティプロトコルを指定できるので、最初の方式が失敗した場合のバックアップシステムが確保されます。ソフトウェアは、リスト内の最初の方式を使用してユーザを認証します。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の認証方式を選択します。このプロセスは、リスト内の認証方式による通信が成功するか、定義された方式をすべて試し終わるまで繰り返されます。この処理のある時点で認証が失敗した場合（つまり、セキュリティサーバまたはローカルのユーザ名データベースがユーザアクセスを拒否すると応答した場合）、認証プロセスは停止し、それ以上認証方式が試行されることはありません。

AAA サーバグループ

既存のサーバー ホストを認証用にグループ化するため、AAA サーバー グループを使用するようにスイッチを設定できます。設定済みのサーバホストのサブセットを選択して、それを特定のサービスに使用します。サーバ グループは、選択されたサーバ ホストの IP アドレスのリストを含むグローバルなサーバ ホスト リストとともに使用されます。

サーバ グループには、同じサーバの複数のホスト エントリを含めることもできますが、各エントリが一意の ID（IP アドレスと UDP ポート番号の組み合わせ）を持っていることが条件です。この場合、個々のポートをそれぞれ特定の AAA サービスを提供する RADIUS ホストとして定義できます。この一意の ID を使用することによって、同じ IP アドレスにあるサーバ上の異なる UDP ポートに、RADIUS 要求を送信できます。同じ RADIUS サーバ上の異なる 2 つのホストエントリに同じサービス（たとえばアカウンティング）を設定した場合、2 番めに設定したホスト エントリは、最初に設定したホスト エントリのフェールオーバー バックアップとして動作します。最初のホスト エントリがアカウンティング サービスの提供に失敗すると、ネットワーク アクセス サーバは同じデバイスに設定されている 2 番めのホスト エントリを使用してアカウンティング サービスを提供するように試行します。（試行される RADIUS ホスト エントリの順番は、設定されている順序に従います）。

AAA 許可

AAA 許可によってユーザが使用できるサービスが制限されます。AAA 許可をイネーブルにすると、スイッチは（ローカル ユーザー データベースまたはセキュリティ サーバー上に存在す

る）ユーザーのプロファイルから取得した情報を使用して、ユーザーのセッションを設定します。ユーザは、ユーザプロファイル内の情報で認められている場合に限り、要求したサービスのアクセスが認可されます。

RADIUS アカウンティング

AAA アカウンティング機能は、ユーザが使用したサービスと、消費したネットワーク リソース量を追跡します。AAA アカウンティングをイネーブルにすると、スイッチはユーザの活動状況をアカウンティング レコードの形式で RADIUS セキュリティ サーバに報告します。各アカウンティング レコードにはアカウンティングの Attribute-Value（AV）ペアが含まれ、レコードはセキュリティ サーバに格納されます。これらのデータは、ネットワーク管理、クライアントへの課金、または監査のために後で分析できます。

ベンダー固有の RADIUS 属性

Internet Engineering Task Force（IETF）ドラフト規格に、ベンダー固有の属性（属性 26）を使用して、スイッチと RADIUS サーバー間でベンダー固有の情報を通信するための方式が定められています。各ベンダーは、Vendor-Specific Attribute（VSA）を使用することによって、一般的な用途には適さない独自の拡張属性をサポートできます。シスコが実装する RADIUS では、この仕様で推奨されるフォーマットを使用して、ベンダー固有のオプションを 1 つサポートしています。シスコのベンダー ID は 9 であり、サポート対象のオプションはベンダー タイプ 1（名前は *cisco-avpair*）です。この値は、次のフォーマットのストリングです。

```
protocol : attribute sep value *
```

protocol は、特定の認証タイプに使用するシスコのプロトコル属性の値です。*attribute* および *value* は、シスコの TACACS+ 仕様で定義されている適切な属性値（AV）ペアです。*sep* は、必須の属性の場合は =、任意指定の属性の場合は * です。TACACS+ 認証で使用できるすべての機能は、RADIUS でも使用できます。

たとえば、次の AV ペアにより、IP 認証中（PPP の IPCP アドレス割り当て中）には、シスコの「multiple named IP address pools」機能がアクティブになります。

```
cisco-avpair= ”ip:addr-pool=first“
```

「*」を挿入すると、AV ペア「ip:addr-pool=first」は省略可能になります。任意の AV ペアを省略可能にすることができます。

```
cisco-avpair= ”ip:addr-pool*first“
```

次に、ネットワーク アクセス サーバからユーザがログインしたときに、すぐに EXEC コマンドを実行する方法の例を示します。

```
cisco-avpair= ”shell:priv-lvl=15“
```

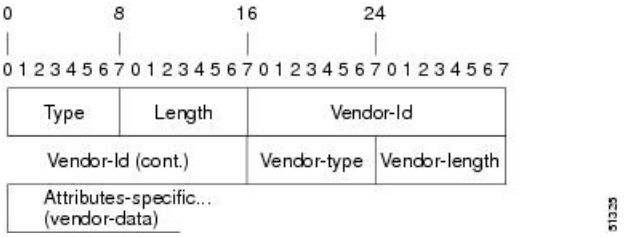
他のベンダーにも、それぞれ独自のベンダー ID、オプション、および対応する VSA があります。ベンダー ID および VSA の詳細については、RFC 2138『Remote Authentication Dial-In User Service (RADIUS)』を参照してください。

属性 26 には、次の 3 つの要素が含まれています。

- タイプ
- 長さ
- ストリング（またはデータ）
  - Vendor-Id
  - Vendor-Type
  - Vendor-Length
  - Vendor-Data

次の図は、属性 26 の「背後で」カプセル化される VSA のパケット形式を示します。

図 *91 :* 属性 *26* の背後でカプセル化される *VSA*



（注） VSA の形式はベンダーが指定します。Attribute-Specific フィールド（Vendor-Data とも呼ばれる）は、ベンダーによるその属性の定義によって異なります。

次の表に、「ベンダー固有 RADIUS IETF 属性テーブル」（次の 2 番目の表）で表示される重要なフィールドを示します。これは、サポート対象のベンダー固有 RADIUS 属性（IETF 属性 26）を表示します。

表 *122 :* ベンダー固有属性表のフィールドの説明

フィールド	説明
番号	次の表に示されるすべての属性は、IETF 属性 26 の拡張です。
ベンダー固有のコマンドコード	特定のベンダーの識別に使用する定義されたコード。コード 9 は Cisco VSA、311 は Microsoft VSA、529 は Ascend VSA を定義します。
サブタイプ番号	属性 ID 番号。この番号は、属性 26 の背後でカプセル化される「2 番めのレイヤ」の ID 番号であること以外、IETF 属性の ID 番号に似ています。
属性	属性の ASCII ストリング名。

フィールド	説明
説明	属性の説明。

表 *123* : ベンダー固有 *RADIUS IETF* 属性

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
MS-CHAP 属性				
26	311	1	MSCHAP-Response	PPP MS-CHAP ユーザがチャレンジに対する応答で提供するレスポンス値が含まれます。Access-Request パケットでしか使用されません。この属性は、PPP CHAP ID と同じです（RFC 2548）
26	311	11	MSCHAP-Challenge	ネットワーク アクセス サーバが MS-CHAP ユーザに送信するチャレンジが含まれます。これは、Access-Request パケットと Access-Challenge パケットの両方で使用できます。（RFC 2548）
VPDN 属性				
26	9	1	l2tp-cm-local-window-size	L2TP 制御メッセージの最大受信ウィンドウ サイズを指定します。この値は、トンネルの確立中にピアにアドバタイズされます。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	1	l2tp-drop-out-of-order	正しくない順序で受信したデータ パケットをドロップして、シーケンス番号を順守します。これは受信した場合の処理方法であって、データ パケット上でシーケンス番号が送信されるわけではありません。
26	9	1	l2tp-hello-interval	hello キープアライブ インターバルの秒数を指定します。ここで指定した秒数、トンネルでデータが送信されないと、hello パケットが送信されます。
26	9	1	l2tp-hidden-avp	イネーブルにすると、L2TP 制御メッセージで、大文字小文字を区別する AVP にスクランブルがかけられるか、または非表示になります。
26	9	1	l2tp-nosession-timeout	タイムアウトおよびシャットダウンまでに、セッションなしでトンネルがアクティブのままになる秒数を指定します。
26	9	1	tunnel-tos-reflect	LNS でトンネルに入るパケットに対して、IP ToS フィールドを各ペイロードパケットの IP ヘッダーからトンネル パケットの IP ヘッダーにコピーします。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	1	l2tp-tunnel-authen	この属性を設定すると、L2TP トンネル認証が実行されます。
26	9	1	l2tp-tunnel-password	L2TP トンネル認証および AVP 隠蔽に使用される共有秘密。
26	9	1	l2tp-udp-checksum	これは認可属性で、L2TP がデータ パケットに対して UDP チェックサムを実行する必要があるかどうかを定義します。有効な値は「yes」と「no」です。デフォルトは「no」です。
Store and Forward Fax 属性				
26	9	3	Fax-Account-Id-Origin	**mmoip aaa receive-id** コマンドまたは **mmoip aaa send-id** コマンドについて、システム管理者によって定義されたものとしてアカウント ID の発信元を示します。
26	9	4	Fax-Msg-Id=	Store and Forward Fax 機能によって割り当てられた一意のファクス メッセージ識別番号を示します。
26	9	5	Fax-Pages	このファクス セッション中に送信または受信したページ数を示します。このページ数には、カバー ページも含まれます。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	6	Fax-Coverpage-Flag	カバー ページがこのファクス セッションのオフランプ ゲートウェイで生成されたかどうかを示します。true はカバー ページが生成されたことを示します。false はカバー ページが生成されなかったことを意味します。
26	9	7	Fax-Modem-Time	モデムがファクス データを送信した時間（x）、およびファクス セッションの合計時間（y）を秒単位で示します。これには、fax-mail および PSTN 時間が x/y の形式で含まれます。たとえば、10/15 は送信時間が 10 秒で、合計ファクス セッションが 15 秒であったことを示します。
26	9	8	Fax-Connect-Speed	この fax-mail が最初に送信または受信された時点のモデム速度を示します。有効値は、1200、4800、9600、および 14400 です。
26	9	9	Fax-Recipient-Count	このファクス送信の受信者数を示します。E メール サーバがセッション モードをサポートするまで、この数字は 1 にする必要があります。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	10	Fax-Process-Abort-Flag	ファクスセッションが中断したこと、または正常に終了したことを示します。true はセッションが中断したことを示します。false はセッションが成功したことを示します。
26	9	11	Fax-Dsn-Address	DSN の送信先のアドレスを示します。
26	9	12	Fax-Dsn-Flag	DSN がイネーブルにされているかどうかを示します。true は DSN がイネーブルにされていることを示します。false は DSN がイネーブルにされていないことを示します。
26	9	13	Fax-Mdn-Address	MDN の送信先のアドレスを示します。
26	9	14	Fax-Mdn-Flag	メッセージ配信通知（MDN）がイネーブルにされているかどうかを示します。true は MDN がイネーブルにされていることを示します。false は MDN がイネーブルにされていないことを示します。
26	9	15	Fax-Auth-Status	このファクスセッションに対する認証が成功したかどうかを示します。このフィールドに対する有効値は、success、failed、bypassed、または unknown です。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	16	Email-Server-Address	オンランプ fax-mail メッセージを処理する E メール サーバの IP アドレスを示します。
26	9	17	Email-Server-Ack-Flag	オンランプ ゲートウェイが fax-mail メッセージを受け入れる E メール サーバから肯定確認応答を受信したことを示します。
26	9	18	Gateway-Id	ファクス セッションを処理したゲートウェイの名前を示します。名前は、hostname.domain-name という形式で表示されます。
26	9	19	Call-Type	ファクスのアクティビティのタイプを、fax receive または fax send のどちらかで記述します。
26	9	20	Port-Used	この fax-mail の送受信いずれかに使用される Cisco AS5300 のスロット/ポート番号を示します。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	21	Abort-Cause	ファクスセッションが中断した場合、中断操作の信号を送信したシステムコンポーネントを示します。中断する可能性のあるシステムコンポーネントには、FAP（Fax Application Process）、TIFF（TIFF リーダーまたは TIFF ライター）、fax-mail クライアント、fax-mail サーバー、ESMTP クライアント、ESMTP サーバーなどがあります。
H323 属性				
26	9	23	Remote-Gateway-ID（h323-remote-address）	リモート ゲートウェイの IP アドレスを示します。
26	9	24	Connection-ID（h323-conf-id）	会議 ID を識別します。
26	9	25	Setup-Time（h323-setup-time）	以前、グリニッジ標準時（GMT）およびズール タイムと呼ばれていた協定世界時（UTC）でのこの接続のセットアップ時間を示します。
26	9	26	Call-Origin（h323-call-origin）	ゲートウェイに対するコールの発行元を示します。有効値は、originating および terminating です（回答）。
26	9	27	Call-Type（h323-call-type）	コールのレグ タイプを示します。使用可能な値は **telephony** と **VoIP** です。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	28	Connect-Time（h323-connect-time）	このコール レッグの UTC での接続時間を示します。
26	9	29	Disconnect-Time（h323-disconnect-time）	このコール レッグが UTC で接続解除された時間を示します。
26	9	30	Disconnect-Cause（h323-disconnect-cause）	Q.931 仕様によって、接続がオフラインにされた理由を示します。
26	9	31	Voice-Quality（h323-voice-quality）	コールの音声品質に影響する Impairment Factor（ICPIF）を指定します。
26	9	33	Gateway-ID（h323-gw-id）	下位のゲートウェイの名前を示します。
大規模のダイヤルアウト属性				
26	9	1	callback-dialstring	コールバックに使用するダイヤリング文字列を定義します。
26	9	1	data-service	説明はありません。
26	9	1	dial-number	ダイヤルする番号を定義します。
26	9	1	force-56	チャネルの 64 K すべてが使用可能に見える場合でも、ネットワーク アクセス サーバが 56 K の部分のみを使用するかどうかを指定します。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	1	map-class	ユーザ プロファイルに、ダイヤルアウトするネットワーク アクセス サーバ上で同じ名前のマップ クラスで設定される情報の参照を許可します。
26	9	1	send-auth	CLID 認証に続く、username-password 認証で使用するプロトコル（PAP または CHAP）を定義します。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	1	send-name	

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
				PPP 名前認証。PAP に適用する場合、インターフェイスで **ppp pap sent-name password** コマンドは設定しないでください。PAP の場合、アウトバウンド認証の PAP ユーザ名および PAP パスワードとして、「preauth:send-name」および「preauth:send-secret」が使用されます。CHAP の場合、「preauth:send-name」は、アウトバウンド認証だけでなく、インバウンド認証にも使用されます。CHAP インバウンドの場合、NAS は発信元のボックスへのチャレンジ パケットに、「preauth:send-name」で定義された名前を使用します。 （注） send-name 属性は時間の経過とともに変わっています。最初は、現在 send-name および remote-name 属性の両方で提供されている機能を実行していました。remote-name

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
				属性が追加されたため、send-name 属性は現在の動作に制限されています。
26	9	1	send-secret	PPP パスワード認証。ベンダー固有属性（VSA）の場合、アウトバウンド認証の PAP ユーザ名および PAP パスワードとして、「preauth:send-name」および「preauth:send-secret」が使用されます。CHAP アウトバウンドの場合、「preauth:send-name」と「preauth:send-secret」の両方が応答パケットで使用されます。
26	9	1	remote-name	大規模のダイヤルアウトで使用するリモートホストの名前を提供します。ダイヤラは、大規模のダイヤルアウトのリモート名が認証された名前と一致することを確認し、偶発的なユーザ RADIUS 設定ミスから保護します（有効な電話番号にダイヤルしたが誤ったデバイスに接続されるなどのミスです）。
その他の属性				

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	2	Cisco-NAS-Port	NAS-Port アカウンティングに追加的なベンダー固有属性（VSA）を指定します。追加的な NAS-Port 情報を属性値ペア（AVPair）の形式で指定するには、**radius-server vsa send** グローバル コンフィギュレーション コマンドを使用します。 （注） この VSA は、通常アカウンティングで使用されますが認証（Access-Request）パケットで使用される場合もあります。
26	9	1	min-links	MLP に対するリンクの最小数を設定します。
26	9	1	proxyacl#<n>	ダウンロード可能なユーザ プロファイル（ダイナミック ACL）を、認証プロキシを使用して設定でき、これにより設定されたインターフェイスのトラフィックの通過を許可するよう、認証を設定できます。

番号	ベンダー固有の企業コード	サブタイプ番号	属性	説明
26	9	1	spi	登録中にホーム エージェントがモバイル ノードの認証で必要とする認証情報を伝送します。この情報は、**ip mobile secure host <addr>** コンフィギュレーション コマンドと同じ構文です。基本的に、この文字列に続く残りのコンフィギュレーション コマンドはそのまま含まれます。これにはセキュリティ パラメータ インデックス（SPI）、キー、認証アルゴリズム、認証モード、およびリプレイ保護タイムスタンプ範囲が含まれています。

ベンダー独自仕様の RADIUS サーバ通信

RADIUS に関する IETF ドラフト規格では、スイッチと RADIUS サーバ間でベンダー独自仕様の情報を通信する方式について定められていますが、RADIUS 属性セットを独自に機能拡張しているベンダーもあります。Cisco IOS ソフトウェアは、ベンダー独自仕様の RADIUS 属性のサブセットをサポートしています。

前述したように、RADIUS（ベンダーの独自仕様によるものか、IETF ドラフトに準拠するものかを問わず）を設定するには、RADIUS サーバ デーモンが稼働しているホストと、そのホストがスイッチと共有秘密テキスト文字列を指定する必要があります。RADIUS ホストおよび秘密テキスト文字列を指定するには、**radius server** グローバル コンフィギュレーション コマンドを使用します。

RADIUS の設定方法

RADIUS サーバ ホストの識別

スイッチと通信するすべての RADIUS サーバーにこのような設定をグローバルに適用するには、**radius-server timeout**、**radius-server retransmit**、および **radius-server key** という 3 つの固有なグローバル コンフィギュレーション コマンドを使用します。

認証時用に AAA サーバー グループを使用して既存のサーバー ホストをグループ化するようにスイッチを設定できます。詳細については、次の関連項目を参照してください。

RADIUS サーバ上でも、いくつかの値を設定する必要があります。これらの設定値としては、スイッチの IP アドレス、およびサーバーとスイッチの双方で共有するキー ストリングがあります。詳細については、RADIUS サーバのマニュアルを参照してください。

サーバ単位で RADIUS サーバとの通信を設定するには、次の手順を実行します。

始める前に

device にグローバルな機能とサーバー単位での機能（タイムアウト、再送信回数、およびキー コマンド）を設定した場合、サーバー単位で設定したタイムアウト、再送信回数、およびキーに関するコマンドは、グローバルに設定したタイムアウト、再送信回数、およびキーに関するコマンドを上書きします。すべての RADIUS サーバーに対してこれらの値を設定する方法については、次の関連項目を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] 例： スイッチ(config)# **radius-server host 172.29.36.49 auth-port 1612 key rad1**	リモート RADIUS サーバー ホストの IP アドレスまたはホスト名を指定します。 • （任意）**auth-port** *port-number* には、認証要求の UDP 宛先ポートを指定します。 • （任意）**acct-port** *port-number* には、アカウンティング要求のUDP宛先ポートを指定します。 • （任意）**timeout** *seconds* には、スイッチが RADIUS サーバーの応答を待ち、再送信するまでの時間間隔を指定します。指定できる範囲は 1 ～ 1000 です。この設定は、**radius-server timeout** グローバル コンフィギュレーション コマンドによる設定を上書きします。**radius-server host** コマンドでタイムアウトが設定されていない場合、**radius-server timeout** コマンドの設定が使用されます。 • （任意）**retransmit** *retries* には、サーバーが応答しない場合、または応答が遅い場合に、RADIUS 要求をサーバーに再送信する回数を指定します。指定できる範囲は 1 ～ 1000 です。**radius-server host** コマンドで再送信回数を指定しない場合、**radius-server retransmit** グローバル コンフィギュレーション コマンドの設定が使用されます。 • （任意）**key** *string* には、スイッチと RADIUS サーバーで動作するRADIUSデーモンの間で使用される認証と暗号キーを指定します。

	コマンドまたはアクション	目的
		（注） キーは、RADIUS サーバーで使用する暗号化キーに一致するテキスト ストリングでなければなりません。必ず **radius-server host** コマンドの最終項目としてキーを設定してください。先頭のスペースは無視されますが、キーの中間および末尾のスペースは使用されます。キーにスペースを使用する場合は、引用符がキーの一部分である場合を除き、引用符でキーを囲まないでください。 スイッチが単一の IP アドレスと関連付けられた複数のホストエントリを認識するように設定するには、このコマンドを必要な回数だけ入力します。その際、各 UDP ポート番号が異なっていることを確認してください。スイッチ ソフトウェアは、指定された順序でホストを検索します。各 RADIUS ホストで使用するタイムアウト、再送信回数、および暗号キーの値をそれぞれ設定してください。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

RADIUS ログイン認証の設定

RADIUS ログイン認証を設定するには、次の手順を実行します。

始める前に

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

手順の概要

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

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa new-model** 例： `スイッチ(config)# aaa new-model`	AAA をイネーブルにします。
ステップ 4	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] 例： `スイッチ(config)# aaa authentication login default local`	ログイン認証方式リストを作成します。 • **login authentication** コマンドにリストが指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。

	コマンドまたはアクション	目的
		• *list-name* には、作成するリストの名前として使用する文字列を指定します。 • *method1...* には、認証アルゴリズムが試行する実際の方式を指定します。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。 次のいずれかの方式を選択します。 • *enable*：イネーブル パスワードを認証に使用します。この認証方式を使用するには、あらかじめ **enable** *password* グローバル コンフィギュレーション コマンドを使用してイネーブルパスワードを定義しておく必要があります。 • *group radius*：RADIUS 認証を使用します。この認証方式を使用するには、あらかじめ RADIUS サーバーを設定しておく必要があります。 • *line*：回線パスワードを認証に使用します。この認証方式を使用するには、あらかじめ回線パスワードを定義しておく必要があります。**password** *password* ライン コンフィギュレーション コマンドを使用します。 • *local*：ローカル ユーザー名データベースを認証に使用します。データベースにユーザ名情報を入力しておく必要があります。**username** *name* **password** グローバル コンフィギュレーション コマンドを使用します。 • *local-case*：大文字と小文字が区別されるローカル ユーザー名データベースを認証に使用します。**username** *password* グローバル コンフィギュレーション コマンドを使用して、ユーザー名情報をデータベースに入力する必要があります。 • *none*：ログインに認証を使用しません。

	コマンドまたはアクション	目的
ステップ 5	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] 例： `スイッチ(config)# line 1 4`	ライン コンフィギュレーション モードを開始し、認証リストを適用する回線を設定します。
ステップ 6	**login authentication** {**default** | *list-name*} 例： `スイッチ(config)# login authentication default`	1 つの回線または複数回線に認証リストを適用します。 • **default** を指定する場合は、**aaa authentication login** コマンドで作成したデフォルトのリストを使用します。 • *list-name* には、**aaa authentication login** コマンドで作成したリストを指定します。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

AAA サーバ グループの定義

定義したグループサーバに特定のサーバを対応付けるには、**server** グループ サーバ コンフィギュレーション コマンドを使用します。サーバを IP アドレスで特定することもできますし、任意指定の **auth-port** および **acct-port** キーワードを使用して複数のホストインスタンスまたはエントリを特定することもできます。

AAA サーバ グループを定義するには、次の手順を実行します。

手順の概要

1. **enable**

2. **configure terminal**
3. **radius server** *name*
4. **address** {**ipv4** | **ipv6**} {*ip-address* | *hostname*} **auth-port** *port-number* **acct-port** *port-number*
5. **key** *string*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**radius server** *name* 例： スイッチ(config)# **radius server ISE**	RADIUS サーバの設定の名前を Protected Access Credential（PAC）のプロビジョニング用に指定し、RADIUS サーバ設定モードを開始します。 deviceは、IPv6 用の RADIUS もサポートしています。
ステップ 4	**address** {**ipv4**	**ipv6**} {*ip-address*	*hostname*} **auth-port** *port-number* **acct-port** *port-number* 例： スイッチ(config-radius-server)# **address ipv4 10.1.1.1 auth-port 1645 acct-port 1646**	RADIUS サーバのアカウンティングおよび認証パラメータの IPv4 アドレスを設定します。
ステップ 5	**key** *string* 例： スイッチ(config-radius-server)# **key cisco123**	デバイスと RADIUS サーバーとの間におけるすべての RADIUS 通信用の認証および暗号キーを指定します。
ステップ 6	**end** 例：	RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config-radius-server)# **end**	
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

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

（注） 許可が設定されていても、CLIを使用してログインし、認証されたユーザに対しては、許可は省略されます。

ユーザ特権アクセスおよびネットワーク サービスに関する RADIUS 許可を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa authorization network radius**
4. **aaa authorization exec radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa authorization network radius** 例： スイッチ(config)# **aaa authorization network radius**	ネットワーク関連のすべてのサービス要求に対して、ユーザーが RADIUS 許可を受けるようにdeviceを設定します。
ステップ 4	**aaa authorization exec radius** 例： スイッチ(config)# **aaa authorization exec radius**	ユーザに特権 EXEC のアクセス権限がある場合、ユーザが RADIUS 許可を受けるように device を設定します。 **exec** キーワードを指定すると、ユーザープロファイル情報（**autocommand** 情報など）が返される場合があります。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

aaa authorization グローバル コンフィギュレーション コマンドと **radius** キーワードを使用すると、ユーザのネットワーク アクセスを特権 EXEC モードに制限するパラメータを設定できます。

aaa authorization exec radius local コマンドは、次の許可パラメータを設定します。

- RADIUS を使用して認証を行った場合は、RADIUS を使用して特権 EXEC アクセスを許可します。
- 認証に RADIUS を使用しなかった場合は、ローカル データベースを使用します。

RADIUS アカウンティングの起動

RADIUS アカウンティングを開始するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa accounting network start-stop radius**
4. **aaa accounting exec start-stop radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa accounting network start-stop radius** 例： スイッチ(config)# **aaa accounting network start-stop radius**	ネットワーク関連のあらゆるサービス要求に関して、RADIUS アカウンティングをイネーブルにします。

	コマンドまたはアクション	目的
ステップ 4	**aaa accounting exec start-stop radius** 例： スイッチ(config)# **aaa accounting exec start-stop radius**	RADIUS アカウンティングをイネーブルにして、特権 EXEC プロセスの最初に記録開始アカウンティング通知、最後に記録停止通知を送信します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

すべての RADIUS サーバの設定

すべての RADIUS サーバーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **radius-server key** *string*
3. **radius-server retransmit** *retries*
4. **radius-server timeout** *seconds*
5. **radius-server deadtime** *minutes*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**radius-server key** *string* 例： スイッチ(config)# **radius-server key your_server_key** スイッチ(config)# **key your_server_key**	スイッチとすべてのRADIUSサーバ間で共有されるシークレット テキスト ストリングを指定します。 （注） キーは、RADIUSサーバで使用する暗号化キーに一致するテキスト ストリングでなければなりません。先頭のスペースは無視されますが、キーの中間および末尾のスペースは使用されます。キーにスペースを使用する場合は、引用符がキーの一部分である場合を除き、引用符でキーを囲まないでください。
ステップ 3	**radius-server retransmit** *retries* 例： スイッチ(config)# **radius-server retransmit 5**	スイッチがRADIUS要求をサーバに再送信する回数を指定します。デフォルトは 3 です。指定できる範囲は 1 ～ 1000 です。
ステップ 4	**radius-server timeout** *seconds* 例： スイッチ(config)# **radius-server timeout 3**	スイッチがRADIUS要求に対する応答を待って、要求を再送信するまでの時間（秒）を指定します。デフォルトは 5 秒です。指定できる範囲は 1 ～ 1000 です。
ステップ 5	**radius-server deadtime** *minutes* 例： スイッチ(config)# **radius-server deadtime 0**	RADIUS サーバが認証要求に応答していない場合、このコマンドはそのサーバに対する要求を停止する時刻を指定します。これにより、要求がタイムアウトするまで待たずとも、次に設定されているサーバを試行することができます。デフォルトは 0 です。指定できる範囲は 1 ～ 1440 分です。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。

	コマンドまたはアクション	目的
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ベンダー固有の RADIUS 属性を使用するデバイス設定

ベンダー固有の RADIUS 属性を使用するようにdeviceを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **radius-server vsa send** [**accounting** | **authentication**]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**radius-server vsa send** [**accounting**	**authentication**] 例： スイッチ(config)# **radius-server vsa send accounting**	deviceが VSA（RADIUS IETF 属性 26 で定義）を認識して使用できるようにします。 • （任意）認識されるベンダー固有属性の集合をアカウンティング属性だけに限定するには、**accounting** キーワードを使用します。 • （任意）認識されるベンダー固有属性の集合を認証属性だけに限定するには、**authentication** キーワードを使用します。

	コマンドまたはアクション	目的
		キーワードを指定せずにこのコマンドを入力すると、アカウンティングおよび認証のベンダー固有属性の両方が使用されます。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ベンダー独自の RADIUS サーバーとの通信に関するデバイスの設定

ベンダー独自仕様の RADIUS サーバー通信を使用するようにdeviceを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **radius-server host** {*hostname* | *ip-address*} **non-standard**
4. **radius-server key** *string*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**radius-server host** {*hostname* | *ip-address*} **non-standard** 例： スイッチ(config)# **radius-server host 172.20.30.15 non-standard**	リモート RADIUS サーバー ホストの IP アドレスまたはホスト名を指定し、RADIUS のベンダー独自仕様の実装を使用することを指定します。
ステップ 4	**radius-server key** *string* 例： スイッチ(config)# **radius-server key rad124**	deviceとベンダー独自仕様の RADIUS サーバーとの間で使用される共有秘密テキスト文字列を指定します。deviceと RADIUS サーバーはこのテキスト文字列を使用してパスワードを暗号化し、応答を交換します。 （注） キーは、RADIUS サーバで使用する暗号化キーに一致するテキスト ストリングでなければなりません。先頭のスペースは無視されますが、キーの中間および末尾のスペースは使用されます。キーにスペースを使用する場合は、引用符がキーの一部分である場合を除き、引用符でキーを囲まないでください。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

次の上での CoA の設定 デバイス

CoA を device で設定するには、次の手順を実行します。この手順は必須です。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa server radius dynamic-author**
5. **client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]
6. **server-key** [**0** | **7**] *string*
7. **port** *port-number*
8. **auth-type** {**any** | **all** | **session-key**}
9. **ignore session-key**
10. **ignore server-key**
11. **authentication command bounce-port ignore**
12. **authentication command disable-port ignore**
13. **end**
14. **show running-config**
15. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ(config)# **aaa new-model**	
ステップ 4	**aaa server radius dynamic-author** 例： スイッチ(config)# **aaa server radius dynamic-author**	device を認証、許可、アカウンティング（AAA）サーバーに設定し、外部ポリシーサーバーとの相互作用を実行します。
ステップ 5	**client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]	ダイナミック許可ローカル サーバー コンフィギュレーション モードを開始し、デバイスが CoA を受け取り、要求を取り外す RADIUS クライアントを指定します。
ステップ 6	**server-key** [**0** | **7**] *string* 例： スイッチ(config-sg-radius)# **server-key your_server_key**	RADIUS キーをデバイスと RADIUS クライアントとの間で共有されるように設定します。
ステップ 7	**port** *port-number* 例： スイッチ(config-sg-radius)# **port 25**	設定された RADIUS クライアントから RADIUS 要求をデバイスが受信するポートを指定します。
ステップ 8	**auth-type** {**any** | **all** | **session-key**} 例： スイッチ(config-sg-radius)# **auth-type any**	device が RADIUS クライアントに使用する許可のタイプを指定します。 クライアントは、許可用に設定されたすべての属性と一致していなければなりません。
ステップ 9	**ignore session-key**	（任意）セッションキーを無視するように device を設定します。 **ignore** コマンドの詳細については、Cisco.com 上の『*Cisco IOS Intelligent Services Gateway Command Reference*』を参照してください。
ステップ 10	**ignore server-key** 例： スイッチ(config-sg-radius)# **ignore server-key**	（任意）サーバキーを無視するように device を設定します。 **ignore** コマンドの詳細については、Cisco.com 上の『*Cisco IOS Intelligent Services Gateway Command Reference*』を参照してください。

	コマンドまたはアクション	目的
ステップ 11	**authentication command bounce-port ignore** 例： スイッチ(config-sg-radius)# **authentication command bounce-port ignore**	（任意）CoA 要求を無視して、セッションをホスティングするポートを一時的にディセーブルにするようにdeviceを設定します。ポートを一時的にディセーブルにする目的は、VLAN の変更が発生しても、その変更を検出するサプリカントがエンドポイント上にない場合に、ホストから DHCP 再ネゴシエーションを行わせることです。
ステップ 12	**authentication command disable-port ignore** 例： スイッチ(config-sg-radius)# **authentication command disable-port ignore**	（任意）セッションをホスティングしているポートを管理上のシャットダウン状態にするよう要求する非標準コマンドを無視するように device を設定します。ポートをシャットダウンすると、セッションが終了します。 ポートを再びイネーブルにするには、標準の CLI または SNMP コマンドを使用します。
ステップ 13	**end** 例： スイッチ(config-sg-radius)# **end**	特権 EXEC モードに戻ります。
ステップ 14	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 15	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

CoA 機能のモニタリング

表 *124 :* 特権 *EXEC* 表示コマンド

コマンド	目的
show aaa attributes protocol radius	RADIUS コマンドの AAA 属性を表示します。

表 *125 :* グローバル トラブルシューティング コマンド

コマンド	目的
debug radius	RADIUS のトラブルシューティングを行うための情報を表示します。
debug aaa coa	CoA 処理のトラブルシューティングを行うための情報を表示します。
debug aaa pod	POD パケットのトラブルシューティングを行うための情報を表示します。
debug aaa subsys	POD パケットのトラブルシューティングを行うための情報を表示します。
debug cmdhd[**detail** |**error** |**events**]	コマンド ヘッダーのトラブルシューティングを行うための情報を表示します。

出力フィールドの詳細については、このリリースに対応するコマンドリファレンスを参照してください。

RADIUS によるスイッチ アクセスの制御の設定例

例：RADIUS サーバー ホストの識別

次に、1 つの RADIUS サーバーを認証用に、もう 1 つの RADIUS サーバーをアカウンティング用に設定する例を示します。

```
スイッチ(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
スイッチ(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

次に、*host1* を RADIUS サーバーとして設定し、認証およびアカウンティングの両方にデフォルトのポートを使用するように設定する例を示します。

```
スイッチ(config)# radius-server host host1
```

例：2 台の異なる RADIUS グループ サーバーの使用

次の例では、2 つの異なる RADIUS グループ サーバー（*group1* および *group2*）を認識するようにスイッチを設定しています。group1 では、同じ RADIUS サーバー上の異なる 2 つのホストエントリを、同じサービス用に設定しています。2番目のホストエントリが、最初のエントリのフェールオーバー バックアップとして動作します。

```
スイッチ(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
スイッチ(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
スイッチ(config)# aaa new-model
スイッチ(config)# aaa group server radius group1
スイッチ(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
スイッチ(config-sg-radius)# exit
スイッチ(config)# aaa group server radius group2
スイッチ(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
スイッチ(config-sg-radius)# exit
```

例：ベンダー固有の RADIUS 属性を使用するスイッチ設定

たとえば、次の AV ペアを指定すると、IP 許可時（PPP の IPCP アドレスの割り当て時）に、シスコの複数の名前付き *IP* アドレス プール機能が有効になります。

```
cisco-avpair= "ip:addr-pool=first"
```

次に、スイッチから特権 EXEC コマンドへの即時アクセスが可能となるユーザー ログインを提供する例を示します。

```
cisco-avpair= "shell:priv-lvl=15"
```

次に、RADIUS サーバ データベース内の許可 VLAN を指定する例を示します。

```
cisco-avpair= "tunnel-type(#64)=VLAN(13)"
cisco-avpair= "tunnel-medium-type(#65)=802 media(6)"
cisco-avpair= "tunnel-private-group-id(#81)=vlanid"
```

次に、この接続中に ASCII 形式の入力 ACL をインターフェイスに適用する例を示します。

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"
cisco-avpair= "mac:inacl#3=deny any any decnet-iv"
```

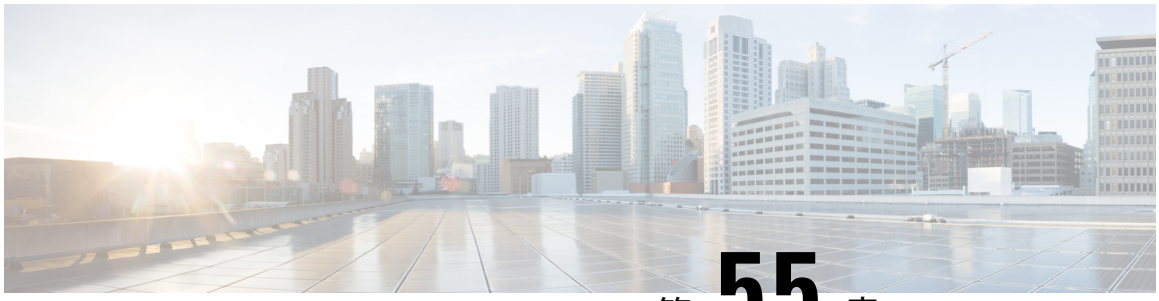
次に、この接続中に ASCII 形式の出力 ACL をインターフェイスに適用する例を示します。

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

例：ベンダー独自仕様の RADIUS サーバーとの通信に関するスイッチ設定

次に、ベンダー独自仕様の RADIUS ホストを指定し、スイッチとサーバーの間で *rad124* という秘密キーを使用する例を示します。

```
スイッチ(config)# radius-server host 172.20.30.15 nonstandard
スイッチ(config)# radius-server key rad124
```



第 55 章

Kerberos の設定

Kerberos によるスイッチ アクセスの制御の前提条件

次に、Kerberos を使用してスイッチ アクセスを制御するための前提条件を示します。

- リモート ユーザーがネットワーク サービスに対して認証を得るには、Kerberos レルム内のホストと KDC を設定し、ユーザーとネットワーク サービスの両方に通信を行い、相互に認証させる必要があります。これを実現するには、互いの識別が必要です。KDC 上の Kerberos データベースにホストのエントリを追加し、Kerberos レルム内のすべてのホストに KDC が生成した KEYTAB ファイルを追加します。また、KDC データベースにユーザー用のエントリも作成します。
- Kerberos サーバーには、ネットワーク セキュリティ サーバーとして設定されていて、Kerberos プロトコルを用いてユーザーを認証できるスイッチを使用できます。

ホストおよびユーザーのエントリを追加または作成する場合の注意事項は次のとおりです。

- Kerberos プリンシパル名はすべて小文字でなければなりません。
- Kerberos インスタンス名はすべて小文字でなければなりません。
- Kerberos レルム名はすべて大文字でなければなりません。

Kerberos に関する情報

ここでは、Kerberos の情報を提供します。

Kerberos とスイッチ アクセス

ここでは、Kerberosセキュリティ システムをイネーブルにして設定する方法について説明します。Kerberos セキュリティ システムは、信頼できるサードパーティを使用してネットワーク リソースに対する要求を認証します。

（注） Kerberos の設定例では、信頼できるサード パーティを、Kerberos をサポートし、ネットワーク セキュリティ サーバーとして設定され、Kerberos プロトコルを使用してユーザーを認証するスイッチとすることができます。

Kerberos の概要

Kerberos はマサチューセッツ工科大学（MIT）が開発した秘密キーによるネットワーク認証プロトコルです。データ暗号規格（DES）という暗号化アルゴリズムを暗号化と認証に使用し、ネットワーク リソースに対する要求を認証します。Kerberosは、信頼できるサードパーティという概念を使ってユーザーとサービスに対してセキュリティの検証を実行します。この信頼できるサードパーティをキー発行局（KDC）と呼びます。

Kerberosは、ユーザーが誰であるか、そのユーザーが使用しているネットワーク サービスは何であるかを検証します。これを実行するために、KDC（つまり信頼できるKerberosサーバー）がユーザーにチケットを発行します。これらのチケットには有効期限があり、ユーザー クレデンシャルのキャッシュに保存されます。Kerberosサーバーは、ユーザー名やパスワードの代わりにチケットを使ってユーザーとネットワーク サービスを認証します。

（注） Kerberos サーバーには、ネットワーク セキュリティ サーバーとして設定されていて、Kerberosプロトコルを用いてユーザーを認証できるのであれば、どのスイッチも使用できます。

Kerberosのクレデンシャル発行スキームでは、*single logon* という手順を使用します。この手順では、ユーザーを1回認証すると、ユーザー クレデンシャルが有効な間は（他のパスワードの暗号化を行わずに）セキュア認証が可能になります。

このソフトウェア リリースは Kerberos 5 に対応しています。Kerberos 5 では、すでに Kerberos 5 を使用している組織が、（UNIX サーバーや PC などの）他のネットワーク ホストが使用している KDC 上の Kerberos 認証データベースを使用できます。

Kerberos は次のネットワーク サービスをサポートしています。

- Telnet
- rlogin
- rsh

次の表に、一般的な Kerberos 関連用語とその定義を示します。

表 *126 : Kerberos* の用語

用語	定義
認証	ユーザーやサービスが他のサービスに対して自分自身の身元を証明する手順。たとえば、クライアントはスイッチに対して認証を得て、スイッチは他のスイッチに対して認証を得ます。
許可	ユーザーがネットワークやスイッチにおいてどのような権限を有しており、またどのような動作を実行できるかを、スイッチが識別する手段
クレデンシャル	認証チケット（TSG[9]、サービス クレデンシャルなど）を表す総称。Kerberos クレデンシャルで、ユーザーまたはサービスの ID を検証します。ネットワーク サービスがチケットを発行した Kerberos サーバーを信頼することにした場合、ユーザー名やパスワードを再入力する代わりにこれを使用できます。証明書の有効期限は、8 時間がデフォルトの設定です。
インスタンス	Kerberos プリンシパルの承認レベル ラベル。ほとんどの Kerberos プリンシパルは、*user@REALM* という形式です（たとえば、smith@EXAMPLE.COM）。Kerberos インスタンスのある Kerberos プリンシパルは、*user/instance@REALM* という形式です（たとえば、smith/admin@EXAMPLE.COM）。Kerberos インスタンスは、認証が成功した場合のユーザーの承認レベルを指定するために使用できます。各ネットワーク サービスのサーバーは、Kerberos インスタンスの許可マッピングを適用し実行できますが、必須ではありません。 （注） Kerberos プリンシパル名およびインスタンス名はすべて小文字でなければなりません。 （注） Kerberos レルム名はすべて大文字でなければなりません。
KDC[10]	ネットワーク ホストで稼働する Kerberos サーバーおよびデータベース プログラムで構成されるキー発行局
Kerberos 対応	Kerberos クレデンシャルのインフラストラクチャをサポートするために変更されたアプリケーションやサービスのことを指す用語
Kerberos レルム	Kerberos サーバーに登録されたユーザー、ホスト、およびネットワーク サービスで構成されるドメイン。Kerberos サーバーを信頼して、ユーザーまたはネットワーク サービスに対する別のユーザーまたはネットワーク サービスの ID を検証します。 （注） Kerberos レルム名はすべて大文字でなければなりません。

用語	定義
Kerberos サーバー	ネットワーク ホストで稼働しているデーモン。ユーザーおよびネットワーク サービスはそれぞれ Kerberos サーバーに ID を登録します。ネットワーク サービスは Kerberos サーバーにクエリーを送信して、他のネットワーク サービスの認証を得ます。
KEYTAB[11]	ネットワーク サービスが KDC と共有するパスワード。Kerberos 5 以降のバージョンでは、ネットワーク サービスは KEYTAB を使って暗号化されたサービス クレデンシャルを暗号解除して認証します。Kerberos 5 よりも前のバージョンでは、KEYTAB はSRVTAB[12]と呼ばれます。
プリンシパル	Kerberos ID とも呼ばれ、Kerberos サーバーに基づき、ユーザーが誰であるか、サービスが何であるかを表します。 （注） Kerberos プリンシパル名はすべて小文字でなければなりません。
サービス クレデンシャル	ネットワーク サービスのクレデンシャル。KDC からクレデンシャルが発行されると、ネットワーク サービスと KDC が共有するパスワードで暗号化されます。ユーザー TGT ともパスワードを共有します。
SRVTAB	ネットワーク サービスが KDC と共有するパスワード。SRVTAB は、Kerberos 5 以降のバージョンでは KEYTAB と呼ばれています。
TGT	身分証明書のことで、KDC が認証済みユーザーに発行するクレデンシャル。TGT を受け取ったユーザーは、KDC が示した Kerberos レルム内のネットワーク サービスに対して認証を得ることができます。

9 チケット認可チケット
10 キー発行局
11 キー テーブル
12 サーバー テーブル

Kerberos の動作

リモートユーザーが device を Kerberos サーバーとして使用してネットワークサービスで認証されるには、次の手順を実行する必要があります。Kerberos をカスタマイズする方法はいくつかありますが、ネットワーク サービスにアクセスしようとするリモート ユーザーは、3 つのセキュリティ レイヤを通過しないとネットワーク サービスにアクセスできません。

リモートユーザーが device を Kerberos サーバーとして使用してネットワークサービスで認証されるには、次の手順を実行する必要があります。

境界スイッチに対する認証の取得

ここでは、リモート ユーザーが通過しなければならない最初のセキュリティ レイヤについて説明します。ユーザーは、まず境界スイッチに対して認証を得なければなりません。リモート ユーザーが境界スイッチに対して認証を得る場合、次のプロセスが発生します。

1. ユーザーが境界スイッチに対して、Kerberos 未対応の Telnet 接続を開始します。
2. ユーザー名とパスワードの入力を求めるプロンプトをスイッチが表示します。
3. スイッチが、このユーザーの TGT を KDC に要求します。
4. KDC がユーザー ID を含む暗号化された TGT をスイッチに送信します。
5. スイッチは、ユーザーが入力したパスワードを使って TGT の暗号解除を試行します。
   - 暗号解除に成功した場合は、ユーザーはスイッチに対して認証を得ます。
   - 暗号解除に成功しない場合は、ユーザー名とパスワードを再入力（Caps Lock または Num Lock のオン/オフに注意）するか、別のユーザー名とパスワードを入力してステップ 2 の手順を繰り返します。

Kerberos 未対応の Telnet セッションを開始し、境界スイッチの認証を得ているリモート ユーザーはファイアウォールの内側にいますが、ネットワーク サービスにアクセスするには、KDC から直接認証を得る必要があります。ユーザーが KDC から認証を得なければならないのは、KDC が発行する TGT はスイッチに保存されており、ユーザーがこのスイッチにログオンしないかぎり、追加の認証に使用できないからです。

KDC からの TGT の取得

ここでは、リモート ユーザーが通過しなければならない 2 番めのセキュリティ レイヤについて説明します。ユーザーは、ネットワーク サービスにアクセスするために、このレイヤで KDC の認証を得て、KDC から TGT を取得しなければなりません。

KDC に対して認証を得る方法については、『*Cisco IOS Security Configuration Guide, Release 12.4*』の「Security Server Protocols」の章にある「Obtaining a TGT from a KDC」を参照してください。

ネットワーク サービスに対する認証の取得

ここでは、リモート ユーザーが通過しなければならない 3 番めのセキュリティ レイヤについて説明します。TGT を取得したユーザーは、このレイヤで Kerberos レルム内のネットワーク サービスに対して認証を得なければなりません。

ネットワーク サービスに対して認証を得る方法については、『*Cisco IOS Security Configuration Guide, Release 12.4*』の「Security Server Protocols」の章の「Authenticating to Network Services」を参照してください。

Kerberos を設定する方法

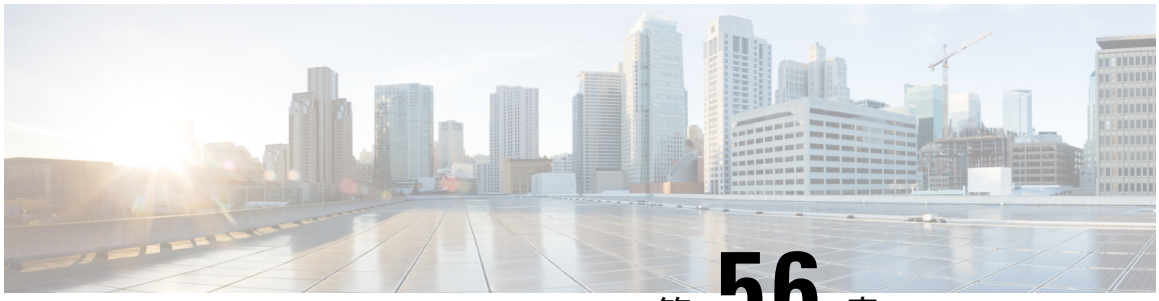
Kerberos 認証済みサーバー/クライアント システムを設定する手順は、次のとおりです。

- Kerberos コマンドを使用して KDC を設定します。
- Kerberos プロトコルを使用するようにスイッチを設定します。

Kerberos 設定の監視

Kerberos 設定を表示するには、次のコマンドを使用します。

- **show running-config**
- **show kerberos creds**：現在のユーザーの認定証キャッシュに含まれる認定証を一覧表示します。
- **clear kerberos creds**：転送済みの認定証を含め、現在のユーザーの認定証キャッシュに含まれるすべての認定証を破棄します。



第 56 章

ローカル認証および許可の設定

ローカル認証および許可の設定方法

スイッチのローカル認証および許可の設定

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

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

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

手順の概要

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

10. **copy running-config startup-config**

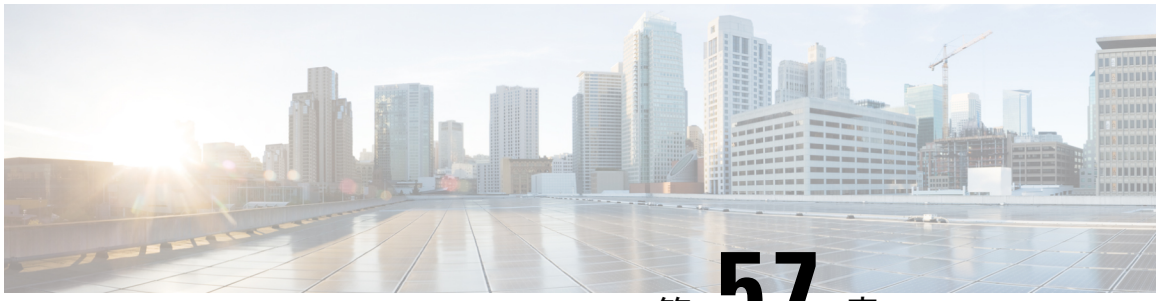
手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa new-model** 例： スイッチ(config)# **aaa new-model**	AAA をイネーブルにします。
ステップ 4	**aaa authentication login default local** 例： スイッチ(config)# **aaa authentication login default local**	ローカル ユーザ名データベースを使用するログイン認証を設定します。**default** キーワードにより、ローカルユーザデータベース認証がすべてのポートに適用されます。
ステップ 5	**aaa authorization exec default local** 例： スイッチ(config)# **aaa authorization exec default local**	ユーザのAAA許可を設定し、ローカルデータベースを確認して、そのユーザに EXEC シェルの実行を許可します。
ステップ 6	**aaa authorization network default local** 例： スイッチ(config)# **aaa authorization network default local**	ネットワーク関連のすべてのサービス要求に対してユーザ AAA 許可を設定します。
ステップ 7	**username** *name* [**privilege** *level*] { **password** *encryption-type password*} 例：	ローカル データベースを入力し、ユーザ名ベースの認証システムを設定します。 ユーザごとにコマンドを繰り返し入力します。

	コマンドまたはアクション	目的
	スイッチ(config)# **username your_user_name privilege 1 password 7 secret567**	• *name* には、ユーザー ID を 1 ワードで指定します。スペースと引用符は使用できません。 • （任意）*level* には、アクセス権を得たユーザーに設定する権限レベルを指定します。指定できる範囲は 0 ～ 15 です。レベル 15 では特権 EXEC モードでのアクセスが可能です。レベル 0 では、ユーザ EXEC モードでのアクセスとなります。 • *encryption-type* には、暗号化されていないパスワードが後ろに続く場合は 0 を、暗号化されたパスワードが後ろに続く場合は 7 を指定します。 • *password* には、ユーザーがスイッチにアクセスする場合に入力する必要のあるパスワードを指定します。パスワードは 1 ～ 25 文字で、埋め込みスペースを使用でき、**username** コマンドの最後のオプションとして指定します。
ステップ 8	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 9	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ローカル認証および許可のモニタリング

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



第 57 章

セキュア シェルの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

セキュア シェルを設定するための前提条件

セキュア シェル（SSH）用にスイッチを設定するための前提条件は、次のとおりです。

- SSH を動作させるには、スイッチに Rivest、Shamir、および Adleman（RSA）の公開キーと秘密キーのペアが必要です。これは SSH が必要なセキュア コピー プロトコル（SCP）も同様で、セキュアな転送を実現させるには、これらのキーのペアが必要です。
- SCP をイネーブルにする前に、スイッチの SSH、認証、許可、およびアカウンティングを適切に設定してください。
- SCP は SSH を使用してセキュアな転送を実行するため、ルータには RSA キーのペアが必要です。

- SCP はセキュリティについて SSH に依存します。
- SCP の設定には認証、許可、およびアカウンティング（AAA）の許可も必要なため、ルータはユーザが正しい権限レベルを保有しているか確認する必要があります。
- ユーザが SCP を使用するには適切な許可が必要です。
- 適切な許可を得ているユーザは、SCP を使用して Cisco IOS File System（IFS）のファイルをスイッチに（またはスイッチから）自由にコピーできます。コピーには **copy** コマンドを使用します。また、許可されている管理者もこの作業をワークステーションから実行できます。
- セキュア シェル（SSH）サーバは、IPsec（データ暗号規格（DES）または 3DES）の暗号化ソフトウェアイメージを必要とします。SSH クライアントは、IPsec（DES または 3DES）の暗号化ソフトウェア イメージが必要です。
- グローバル コンフィギュレーション モードで **hostname** および **ip domain-name** コマンドを使用して、デバイスのホスト名とホスト ドメインを設定します。

セキュア シェルの設定に関する制約事項

セキュア シェル用にデバイスを設定するための制約事項は、次のとおりです。

- スイッチは、Rivest, Shamir, and Adelman（RSA）認証をサポートします。
- SSH は、実行シェル アプリケーションだけをサポートします。
- SSH サーバおよび SSH クライアントは、データ暗号規格（DES）（56 ビット）および 3DES（168 ビット）データ暗号化ソフトウェアでのみサポートされます。DES ソフトウェア イメージの場合、使用できる暗号化アルゴリズムは DES だけです。3DES ソフトウェア イメージの場合、DES と 3DES の両方の暗号化アルゴリズムを使用できます。
- device は、128 ビット キー、192 ビット キー、または 256 ビット キーの Advanced Encryption Standard（AES）暗号化アルゴリズムをサポートします。ただし、キーを暗号化する対称暗号化 AES はサポートされません。
- SCP を使用する場合、**copy** コマンドにパスワードを入力することはできません。プロンプトが表示されたときに、入力する必要があります。
- ログイン バナーはセキュア シェル バージョン 1 ではサポートされません。セキュア シェル バージョン 2 ではサポートされています。
- リバース SSH の代替手段をコンソール アクセス用に設定する場合、-l キーワード、userid :{number} {ip-address} デリミタ、および引数が必須です。
- FreeRADIUS over RADSEC でクライアントを認証するには、1024 ビットよりも長い RSA キーを生成する必要があります。その場合は、**crypto key generate rsa general-keys exportable label** *label-name* コマンドを使用します。

セキュア シェルの設定について

セキュア シェル（SSH）は、デバイスに対する安全なリモート接続を可能にするプロトコルです。SSH は、デバイスの認証時に強力な暗号化を行うことで、リモート接続について Telnet 以上のセキュリティを実現します。このソフトウェア リリースは、SSH バージョン 1（SSHv1）および SSH バージョン 2（SSHv2）をサポートしています。

SSH およびデバイスアクセス

セキュア シェル（SSH）は、デバイスに対する安全なリモート接続を可能にするプロトコルです。SSH は、デバイスの認証時に強力な暗号化を行うことで、リモート接続について Telnet 以上のセキュリティを実現します。このソフトウェア リリースは、SSH バージョン 1（SSHv1）および SSH バージョン 2（SSHv2）をサポートしています。

IPv6 の SSH 機能は IPv4 における機能と同じです。IPv6 の場合、SSH は IPv6 アドレスをサポートし、IPv6 トランスポート上において、リモート IPv6 ノードとのセキュリティ保護および暗号化された接続を有効化します。

SSH サーバ、統合クライアント、およびサポートされているバージョン

セキュア シェル（SSH）統合クライアント機能は、SSH プロトコル上で動作し、デバイスの認証および暗号化を実現するアプリケーションです。SSH クライアントによって、シスコ デバイスは別のシスコ デバイスなど SSH サーバを実行するデバイスに対して、セキュアで暗号化された接続を実行できます。この接続は、接続が暗号化される点を除いて Telnet のアウトバウンド接続と同様の機能を提供します。SSH クライアントは、認証および暗号化により、保護されていないネットワーク上でもセキュアな通信ができます。

SSH サーバおよび SSH 統合クライアントは、スイッチ上で実行されるアプリケーションです。SSH サーバは、このリリースでサポートされている SSH クライアントおよび、他社製の SSH クライアントと使用します。SSH クライアントは、市販の一般的な SSH サーバと連動します。SSH クライアントは、Data Encryption Standard（DES）、3DES、およびパスワード認証の暗号をサポートします。

スイッチは、SSHv1 または SSHv2 サーバをサポートします。

スイッチは、SSHv1 クライアントをサポートします。

（注） SSH クライアント機能を使用できるのは、SSH サーバがイネーブルの場合だけです。

ユーザ認証は、デバイスに対する Telnet セッションの認証と同様に実行されます。SSH は、次のユーザ認証方式もサポートします。

- TACACS+

- RADIUS
- ローカル認証および許可

SSH 設定時の注意事項

スイッチをSSHサーバーまたはSSHクライアントとして設定する場合は、次の注意事項に従ってください。

- SSHv2 サーバーは、SSHv1 サーバーで生成される RSA キーのペアを使用できます（逆の場合も同様です）。
- SSHサーバーがアクティブスイッチ上で動作しており、アクティブスイッチに障害が発生した場合、新しいアクティブスイッチは、以前のアクティブスイッチによって生成された RSA キーペアを使用します。
- **crypto key generate rsa**グローバル コンフィギュレーション コマンドを入力した後、CLI エラー メッセージが表示される場合、RSA キーペアは生成されていません。ホスト名およびドメインを再設定してから、**crypto key generate rsa** コマンドを入力してください。
- RSA キーのペアを生成する場合に、メッセージ「No host name specified」が表示されることがあります。このメッセージが表示された場合は、**hostname** グローバル コンフィギュレーション コマンドを使用してホスト名を設定する必要があります。
- RSA キーのペアを生成する場合に、メッセージ「No domain specified」が表示されることがあります。このメッセージが表示された場合は、**ip domain-name** グローバル コンフィギュレーション コマンドを使用して IP ドメイン名を設定する必要があります。
- ローカル認証および許可の方法を設定する場合に、コンソール上で AAA がディセーブルにされていることを確認してください。

Secure Copy Protocol の概要

Secure Copy Protocol（SCP）機能は、スイッチの設定やイメージ ファイルのコピーにセキュアな認証方式を提供します。SCP にはセキュア シェル（SSH）が必要です（Berkeley の r-tool に代わるセキュリティの高いアプリケーションおよびプロトコルです）。

SSHを動作させるには、スイッチにRSAの公開キーと秘密キーのペアが必要です。これはSSHが必要なSCPも同様で、セキュアな転送を実現させるには、これらのキーのペアが必要です。

また、SSH には AAA 許可が必要のため、適切に設定するには、SCP にも AAA 認証が必要になります。

- SCPをイネーブルにする前に、スイッチのSSH、認証、許可、およびアカウンティングを適切に設定してください。
- SCP は SSH を使用してセキュアな転送を実行するため、ルータには RSA キーのペアが必要です。

（注） SCP を使用する場合、copy コマンドにパスワードを入力することはできません。プロンプトが表示されたときに、入力する必要があります。

Secure Copy Protocol

セキュア コピー プロトコル（SCP）機能は、deviceの設定やスイッチ イメージ ファイルのコピーにセキュアな認証方式を提供します。SCP は一連の Berkeley の r-tools に基づいて設計されているため、その動作内容は、SCP が SSH のセキュリティに対応している点を除けば、Remote Copy Protocol（RCP）と類似しています。また、SCP では認証、許可、およびアカウンティング（AAA）の設定が必要なため、deviceはユーザーが正しい権限レベルを保有しているかどうかを特定できます。セキュア コピー機能を設定するには、SCP の概念を理解する必要があります。

SSH の設定方法

SSH を実行するためのスイッチのセットアップ

SSH を実行するようにスイッチをセットアップするには、次の手順を実行します。

始める前に

ローカル アクセスまたはリモート アクセス用にユーザ認証を設定します。この手順は必須です。詳細については、次の関連項目を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **hostname** *hostname*
4. **ip domain-name** *domain_name*
5. **crypto key generate rsa**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**hostname** *hostname* 例： スイッチ(config)# **hostname your_hostname**	スイッチのホスト名およびIP ドメイン名を設定します。 （注） この手順を実行するのは、スイッチをSSH サーバーとして設定する場合だけです。
ステップ 4	**ip domain-name** *domain_name* 例： スイッチ(config)# **ip domain-name your_domain**	スイッチのホスト ドメインを設定します。
ステップ 5	**crypto key generate rsa** 例： スイッチ(config)# **crypto key generate rsa**	スイッチ上でローカルおよびリモート認証用にSSH サーバーを有効にし、RSA キーペアを生成します。スイッチの RSA キー ペアを生成すると、SSH が自動的に有効になります。 最小モジュラス サイズは、1024 ビットにすることを推奨します。 RSA キーのペアを生成する場合に、モジュラスの長さの入力を求められます。モジュラスが長くなるほど安全ですが、生成と使用に時間がかかります。 （注） この手順を実行するのは、スイッチをSSH サーバーとして設定する場合だけです。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。

	コマンドまたはアクション	目的
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SSH サーバの設定

SSH サーバーを設定するには、次の手順を実行します。

（注） スイッチを SSH サーバーとして設定する場合にのみ、この手順が必要です。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh version** [**1** | **2**]
4. **ip ssh version** [**2**]
5. **ip ssh** {**time-out** *seconds* | **authentication-retries** *number*}
6. 次のいずれかまたは両方を使用します。
   - `line vty` *line_number*[*ending_line_number*]
   - **transport input ssh**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**ip ssh version** [**1** | **2**] 例： `スイッチ(config)# ip ssh version 1`	（任意）SSH バージョン 1 または SSH バージョン 2 を実行するようにスイッチを設定します。 • **1**：SSH バージョン 1 を実行するようにスイッチを設定します。 • **2**：SSH バージョン 2 を実行するようにスイッチを設定します。 このコマンドを入力しない場合、またはキーワードを指定しない場合、SSH サーバは、SSH クライアントでサポートされている最新バージョンの SSH を選択します。たとえば、SSH クライアントが SSHv1 および SSHv2 をサポートする場合、SSH サーバは SSHv2 を選択します。
ステップ 4	**ip ssh version** [**2**] 例： `スイッチ(config)# ip ssh version 2`	（任意）SSH バージョン 2 を実行するようにスイッチ を設定します。
ステップ 5	**ip ssh** {**time-out** *seconds* | **authentication-retries** *number*} 例： `スイッチ(config)# ip ssh time-out 90` `OR` `スイッチ(config)# ip ssh authentication-retries 2`	SSH 制御パラメータを設定します。 • **time-out** *seconds*：タイムアウト値は秒単位で指定します（デフォルト値は 120 秒）。指定できる範囲は 0 ～ 120 秒です。このパラメータは、SSH ネゴシエーション フェーズに適用されます。接続が確立されると、スイッチは CLI ベースセッションのデフォルトのタイムアウト値を使用します。 デフォルトでは、ネットワーク上の複数の CLI ベース セッション（セッション 0 ～ 4）に対して、最大 5 つの暗号化同時 SSH 接続を使用できます。実行シェルが起動すると、CLI ベースセッションのタイムアウト値はデフォルトの 10 分に戻ります。 • **authentication-retries** *number*：クライアントをサーバーへ再認証できる回数を指定します。デフォルトは 3 です。指定できる範囲は 0 ～ 5 です。 両方のパラメータを設定する場合はこの手順を繰り返します。

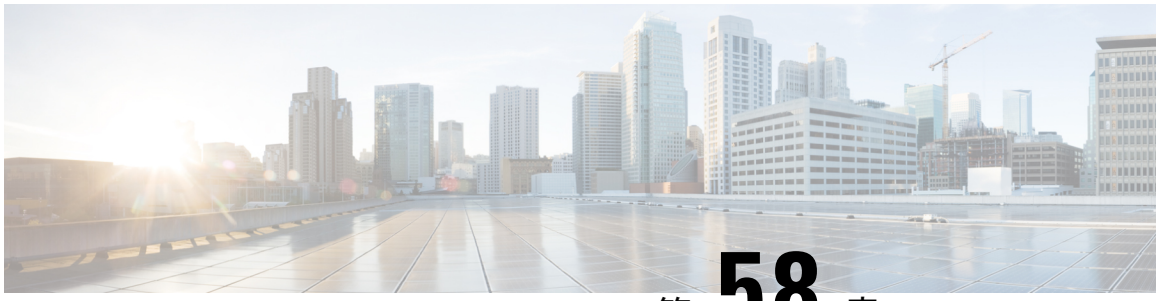
	コマンドまたはアクション	目的
ステップ 6	次のいずれかまたは両方を使用します。 • `line vty` *line_number*[*ending_line_number*] • **transport input ssh** 例： `スイッチ(config)# line vty 1 10` または `スイッチ(config-line)# transport input ssh`	（任意）仮想端末回線設定を設定します。 • ライン コンフィギュレーション モードを開始して、仮想端末回線設定を設定します。*line_number* および *ending_line_number* には、回線のペアを指定します。指定できる範囲は 0 ～ 15 です。 • スイッチが非 SSH Telnet 接続を阻止するように指定します。これにより、ルータはSSH接続に限定されます。
ステップ 7	**end** 例： `スイッチ(config-line)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

SSH の設定およびステータスのモニタリング

次の表に、SSH サーバの設定およびステータスを示します。

表 *127 : SSH* サーバの設定およびステータスを表示するコマンド

コマンド	目的
show ip ssh	SSHサーバのバージョンおよび設定情報を表示します。
show ssh	SSH サーバのステータスを表示します。



第 58 章

SSH File Transfer Protocol の設定

セキュアシェル（SSH）には、SSHv2 で導入された新たな標準ファイル転送プロトコルである SSH File Transfer Protocol（SFTP）のサポートが含まれています。この機能は、デバイス設定またはデバイスイメージファイルをコピーするための安全で認証された方式を提供します。

SSH File Transfer Protocol の前提条件

- SSH を有効にする必要があります。
- **ip ssh source-interface** *interface-type interface-number* コマンドを設定する必要があります。

SSH File Transfer Protocol の制約事項

- SFTP サーバはサポートされていません。
- SFTP 起動はサポートされていません。
- **sftp** コマンドでの **install add** オプションはサポートされていません。

SSH File Transfer Protocol に関する情報

SFTP クライアント機能は SSH コンポーネントの一部として提供され、対応するデバイスで常に有効になっています。したがって、適切な権限を持つ SFTP サーバのユーザは、デバイスとの間でファイルをコピーできます。

SFTP クライアントは VRF 対応です。接続の試行時に特定の送信元インターフェイスに関連付けられた仮想ルーティングおよび転送（VRF）を使用するようにセキュア FTP クライアントを設定できます。

SSH File Transfer Protocol の設定方法

ここでは、SFTP の設定を構成するさまざまな作業について説明します。

SFTP の設定

次の操作を行ってください。

始める前に

SFTP クライアント側機能用にシスコ デバイスを設定するには、最初に **ip ssh source-interface** *interface-type interface-number* コマンドを設定する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh source-interface** *interface-type interface-number*
4. **exit**
5. **show running-config**
6. **debug ip sftp**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**ip ssh source-interface** *interface-type interface-number* 例： `Device(config)# ip ssh source-interface GigabitEthernet 1/0/1`	SSH セッションの送信元 IP を定義します。
ステップ 4	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `Device# show running-config`	（任意）SFTP クライアント側機能を表示します。
ステップ 6	**debug ip sftp** 例： `Device# debug ip sftp`	（任意）SFTP デバッグを有効にします。

SFTP コピー操作の実行

ドメインネームシステム（DNS）が設定されている場合、SFTP コピーは対応するサーバの IP またはホスト名を取得します。SFTP コピー操作を実行するには、特権 EXEC モードで次のコマンドを使用します。

コマンド	目的
Device# **copy ios-file-system:file sftp://user:pwd@server-ip//filepath** または Device# **copy ios-file-system: sftp:**	ローカル Cisco IOS ファイルシステムからサーバにファイルをコピーします。 サーバのユーザ名、パスワード、IP アドレス、およびファイルパスを指定します。
Device# **copy sftp://user:pwd@server-ip //filepath ios-file-system:file** または Device# **copy sftp: ios-file-system:**	サーバからローカル Cisco IOS ファイルシステムにファイルをコピーします。 サーバのユーザ名、パスワード、IP アドレス、およびファイルパスを指定します。

例：SSH File Transfer Protocol の設定

次に、SFTP のクライアント側機能を設定する例を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip ssh source-interface gigabitethernet 1/0/1
Device(config)# exit
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
セキュアシェルバージョン 1 と 2 のサポート	セキュア シェルの設定

シスコのテクニカル サポート

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

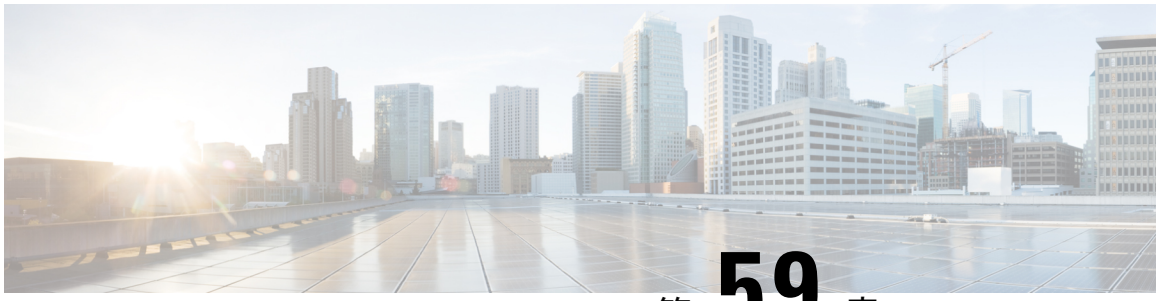
SSH File Transfer Protocol の機能情報

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

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

表 *128 : SFTP* の機能情報

機能名	リリース	機能情報
SSH File Transfer Protocol（SFTP）	Cisco IOS リリース 15.2(7)E	SSH には、SSHv2 で導入された新たな標準ファイル転送プロトコルである SFTP のサポートが含まれています。



第 59 章

SSH 認証の X.509v3 証明書

SSH 認証用の X.509v3 証明書機能は、公開キー アルゴリズム（PKI）を使用してサーバおよびユーザの認証を行い、認証局（CA）が署名し発行したデジタル証明書を介してキー ペアの所有者のアイデンティティをセキュア シェル（SSH）プロトコルによって検証することを可能します。

このモジュールでは、デジタル証明書用のサーバおよびユーザ証明書プロファイルを設定する方法について説明します。

SSH 認証の X.509v3 証明書の前提条件

SSH 認証用の X.509v3 証明書機能では、 **ip ssh server algorithm authentication** コマンドの代わりに **ip ssh server authenticate user** コマンドが置き換えられます。**default ip ssh server authenticate user** コマンドを設定し、コンフィギュレーションから **ip ssh server authenticate user** コマンドを削除します。IOS セキュアシェル（SSH）サーバは **ip ssh server algorithm authentication** コマンドを使用して起動します。

ip ssh server authenticate user コマンドを実行すると、次のメッセージが表示されます。

警告 SSH コマンドを受け入れました。ただし、このCLIはまもなく廃止されます。新しいCLI **ip ssh server algorithm authentication**に移動してください。「**default ip ssh server authenticate user**」を設定し、CLI を有効にします。

SSH 認証の X.509v3 証明書の制約事項

- SSH 認証用の X.509v3 証明書機能の実装は、Cisco IOS セキュア シェル（SSH）サーバ側にのみ適用できます。
- Cisco IOS SSH サーバは、サーバおよびユーザ認証について、x509v3-ssh-rsa アルゴリズム ベースの証明書のみをサポートします。

SSH 認証用の X.509v3 証明書に関する情報

SSH 認証用の X.509v3 証明書の概要

セキュア シェル（SSH）プロトコルは、ネットワーク デバイスへの安全なリモート アクセス接続を提供します。クライアントとサーバの間の通信は暗号化されます。

公開キー暗号化を使用して認証を行う SSH プロトコルが 2 つあります。トランスポート層プロトコルは、デジタル署名アルゴリズム（公開キーアルゴリズムと呼ばれます）を使用して、サーバをクライアントに対して認証します。一方、ユーザ認証プロトコルは、デジタル署名を使用して、クライアントをサーバに対して認証します（公開キー認証）。

認証の有効性は、公開署名キーとその署名者のアイデンティティとの関連の強さに依存します。X.509 バージョン 3（X.509v3）などのデジタル証明書は、アイデンティティ管理のために使用されます。X.509v3 は、信頼できるルート認証局とその中間認証局による署名の連鎖を使用して、公開署名キーを特定のデジタル アイデンティティにバインドします。この実装により、公開キー アルゴリズムを使用したサーバとユーザの認証が可能になるとともに、認証局（CA）が署名し発行したデジタル証明書を介してキー ペアの所有者のアイデンティティを SSH で検証することが可能になります。

X.509v3 を使用したサーバおよびユーザ認証

サーバ認証の場合、セキュア シェル（SSH）サーバが確認のためにそれ自体の証明書を SSH クライアントに送信します。このサーバ証明書は、サーバ証明書プロファイル（ssh-server-cert-profile-server コンフィギュレーション モード）で設定されたトラストポイントに関連付けられます。

ユーザ認証の場合、SSH クライアントが確認のためにユーザの証明書を IOS SSH サーバに送信します。SSH サーバは、サーバ証明書プロファイル（ssh-server-cert-profile-user コンフィギュレーション モード）で設定された公開キー インフラストラクチャ（PKI）トラストポイントを使用して、受信したユーザ証明書を確認します。

デフォルトでは、証明書ベースの認証が、IOS SSH サーバ端末でサーバおよびユーザに対して有効になっています。

OCSP 応答ステープリング

オンライン証明書ステータス プロトコル（OCSP）では、識別された証明書の（失効）状態をアプリケーションが判断することが可能です。このプロトコルは、証明書のステータスをチェックするアプリケーションとそのステータスを提供するサーバとの間でやり取りする必要があるデータを指定します。OCSP クライアントは OCSP レスポンダにステータス要求を発行し、応答を受信するまで証明書の受け入れを保留します。OCSP 応答には、少なくとも、要求の処理ステータスを示す responseStatus フィールドが含まれます。

公開キー アルゴリズムの場合、キーの形式は、1 つ以上の X.509v3 証明書のシーケンスと、その後に続く 0 個以上の OCSP 応答のシーケンスから成ります。

SSH 認証機能向けの X.509v3 証明書は、OCSP 応答ステープリングを使用します。OCSP 応答ステープリングを使用することにより、デバイスは、OCSP サーバにアクセスしてから結果を証明書とともにステープリングして、ピアから OCSP レスポンダにアクセスさせるのではなくピアに情報を送ることで、自身の証明書の失効情報を取得します。

SSH 認証用の X.509v3 証明書の設定方法

サーバ認証用のデジタル証明書の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh server algorithm hostkey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]}
4. **ip ssh server certificate profile**
5. **server**
6. **trustpoint sign** *PKI-trustpoint-name*
7. **ocsp-response include**
8. **end**
9. **line vty line_number** ［*ending_line_number*］
10. **transport input ssh**

手順の詳細

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

	コマンドまたはアクション	目的
	`Switch# configure terminal`	
ステップ 3	**ip ssh server algorithm hostkey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]} 例： `Switch(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa`	ホスト キー アルゴリズムの順序を定義します。セキュア シェル（SSH）クライアントとネゴシエートされるのは、設定済みアルゴリズムのみです。 （注） IOS SSH サーバには、1 つ以上の設定済みホスト キー アルゴリズムが必要です。 • **x509v3-ssh-rsa**：証明書ベースの認証 • **ssh-rsa**：公開キーベースの認証
ステップ 4	**ip ssh server certificate profile** 例： `Switch(config)# ip ssh server certificate profile`	サーバ証明書プロファイルおよびユーザ証明書プロファイルを設定し、SSH 証明書プロファイル コンフィギュレーション モードを開始します。
ステップ 5	**server** 例： `Switch(ssh-server-cert-profile)# server`	サーバ証明書プロファイルを設定し、SSH サーバ証明書プロファイルのユーザ コンフィギュレーション モードを開始します。 • サーバ プロファイルは、サーバ認証時にサーバ証明書を SSH クライアントに送信するために使用されます。
ステップ 6	**trustpoint sign** *PKI-trustpoint-name* 例： `Switch(ssh-server-cert-profile-server)# trustpoint sign trust1`	公開キーインフラストラクチャ（PKI）トラストポイントをサーバ証明書プロファイルにアタッチします。 • SSH サーバは、この PKI トラストポイントに関連付けられた証明書をサーバ認証に使用します。
ステップ 7	**ocsp-response include** 例： `Switch(ssh-server-cert-profile-server)# ocsp-response include`	（任意）Online Certificate Status Protocol（OCSP）の応答またはOCSP ステープリングをサーバ証明書と一緒に送信します。 （注） デフォルトでは、OCSP 応答はサーバ証明書と一緒には送信されません。
ステップ 8	**end** 例： `Switch(ssh-server-cert-profile-server)# end`	SSH サーバ証明書プロファイルのサーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 9	**line vty line_number** ［*ending_line_number*］ 例： `Switch(config)# line vty line_number [ending_line_number]`	ラインコンフィギュレーションモードを開始して、仮想端末回線設定を設定します。line_number および ending_line_number には、回線のペアを指定します。指定できる範囲は 0 ～ 15 です。
ステップ 10	**transport input ssh** 例： `Switch(config-line)#transport input ssh`	スイッチで非 SSH Telnet 接続を回避するように設定します。これにより、ルータは SSH 接続に限定されます。

ユーザ認証用のデジタル証明書の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ip ssh server algorithm authentication** {**publickey** | **keyboard** | **password**}
4. **ip ssh server algorithm publickey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]}
5. **ip ssh server certificate profile**
6. **user**
7. **trustpoint verify** *PKI-trustpoint-name*
8. **ocsp-response required**
9. **end**
10. **line vty line_number** ［*ending_line_number*］
11. **transport input ssh**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip ssh server algorithm authentication** {**publickey**	**keyboard**	**password**} 例：	ユーザ認証アルゴリズムの順序を定義します。セキュア シェル（SSH）クライアントとネゴシエートされるのは、設定済みアルゴリズムのみです。

	コマンドまたはアクション	目的
	`Switch(config)# ip ssh server algorithm authentication publickey`	（注） • IOS SSH サーバには、1 つ以上の設定済みユーザ認証アルゴリズムが必要です。 • ユーザ認証に証明書方式を使用するには、**publickey** キーワードを設定する必要があります。
ステップ 4	**ip ssh server algorithm publickey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]} 例： `Switch(config)# ip ssh server algorithm publickey x509v3-ssh-rsa`	公開キー アルゴリズムの順序を定義します。SSH クライアントによってユーザ認証に許可されるのは、設定済みのアルゴリズムのみです。 （注） IOS SSH クライアントには、1 つ以上の設定済み公開キー アルゴリズムが必要です。 • **x509v3-ssh-rsa**：証明書ベースの認証 • **ssh-rsa**：公開キーベースの認証
ステップ 5	**ip ssh server certificate profile** 例： `Switch(config)# ip ssh server certificate profile`	サーバ証明書プロファイルおよびユーザ証明書プロファイルを設定し、SSH 証明書プロファイル コンフィギュレーション モードを開始します。
ステップ 6	**user** 例： `Switch(ssh-server-cert-profile)# user`	ユーザ証明書プロファイルを設定し、SSH サーバ証明書プロファイルのユーザ コンフィギュレーション モードを開始します。
ステップ 7	**trustpoint verify** *PKI-trustpoint-name* 例： `Switch(ssh-server-cert-profile-user)# trustpoint verify trust2`	受信したユーザ証明書の確認に使用される公開キー インフラストラクチャ（PKI）トラストポイントを設定します。 （注） 同じコマンドを複数回実行することで、複数のトラストポイントを設定します。最大 10 のトラストポイントを設定できます。
ステップ 8	**ocsp-response required** 例： `Switch(ssh-server-cert-profile-user)# ocsp-response required`	（任意）受信したユーザ証明書による Online Certificate Status Protocol（OCSP）の応答の有無を要求します。 （注） デフォルトでは、ユーザ証明書は OCSP 応答なしで受け入れられます。

	コマンドまたはアクション	目的
ステップ 9	**end** 例： `Switch(ssh-server-cert-profile-user)# end`	SSH サーバ証明書プロファイルのユーザ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 10	**line vty line_number** ［*ending_line_number*］ 例： `Switch(config)# line vty line_number [ending_line_number]`	ラインコンフィギュレーションモードを開始して、仮想端末回線設定を設定します。line_number および ending_line_number には、回線のペアを指定します。指定できる範囲は 0 ～ 15 です。
ステップ 11	**transport input ssh** 例： `Switch(config-line)#transport input ssh`	スイッチで非 SSH Telnet 接続を回避するように設定します。これにより、ルータは SSH 接続に限定されます。

デジタル証明書を使用したサーバおよびユーザ認証の確認

手順の概要

1. **enable**
2. **show ip ssh**
3. **debug ip ssh detail**
4. **show log**
5. **debug ip packet**
6. **show log**

手順の詳細

ステップ 1 enable

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 show ip ssh

現在設定されている認証方式を表示します。証明書ベース認証の使用を確認するには、x509v3-ssh-rsa アルゴリズムが設定済みのホスト キー アルゴリズムであることを確認します。

例：

```
Device# show ip ssh

SSH Enabled - version 1.99
Authentication methods:publickey,keyboard-interactive,password
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa
Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
```

ステップ 3 debug ip ssh detail

SSH 詳細のデバッグメッセージをオンにします。

例：

```
Device# debug ip ssh detail

ssh detail messages debugging is on
```

ステップ 4 show log

デバッグメッセージログを表示します。

例：

```
Device# show log

Syslog logging: enabled (0 messages dropped, 9 messages rate-limited, 0 flushes, 0 overruns, xml
disabled, filtering disabled)

No Active Message Discriminator.



No Inactive Message Discriminator.


    Console logging: disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 233 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    File logging: disabled
    Persistent logging: disabled

No active filter modules.

    Trap logging: level informational, 174 message lines logged
        Logging Source-Interface:       VRF Name:

Log Buffer (4096 bytes):
5 IST: SSH2 CLIENT 0: SSH2_MSG_KEXINIT sent
*Sep  6 14:44:08.496 IST: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
*Sep  6 14:44:08.496 IST: SSH2 0: kexinit sent: kex algo =
diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1
*Sep  6 14:44:08.496 IST: SSH2 0: Server certificate trustpoint not found. Skipping hostkey algo =
 x509v3-ssh-rsa
*Sep  6 14:44:08.496 IST: SSH2 0: kexinit sent: hostkey algo = ssh-rsa
*Sep  6 14:44:08.496 IST: SSH2 0: kexinit sent: encryption algo = aes128-ctr,aes192-ctr,aes256-ctr
```

```
*Sep  6 14:44:08.496 IST: SSH2 0: kexinit sent: mac algo =
hmac-sha2-256,hmac-sha2-512,hmac-sha1,hmac-sha1-96
*Sep  6 14:44:08.496 IST: SSH2 0: SSH2_MSG_KEXINIT sent
*Sep  6 14:44:08.496 IST: SSH2 0: SSH2_MSG_KEXINIT received
*Sep  6 14:44:08.496 IST: SSH2 0: kex: client->server enc:aes128-ctr mac:hmac-sha2-256
*Sep  6 14:44:08.496 IST: SSH2 0: kex: server->client enc:aes128-ctr mac:hmac-sha2-256
*Sep  6 14:44:08.496 IST: SSH2 0: Using hostkey algo = ssh-rsa
*Sep  6 14:44:08.496 IST: SSH2 0: Using kex_algo = diffie-hellman-group-exchange-sha1
*Sep  6 14:44:08.497 IST: SSH2 CLIENT 0: SSH2_MSG_KEXINIT received
*Sep  6 14:44:08.497 IST: SSH2 CLIENT 0: kex: server->client enc:aes128-ctr mac:hmac-sha2-256
*Sep  6 14:44:08.497 IST: SSH2 CLIENT 0: kex: client->server enc:aes128-ctr mac:hmac-sha2-256
*Sep  6 14:44:08.497 IST: SSH2 CLIENT 0: Using hostkey algo = ssh-rsa
*Sep  6 14:44:08.497 IST: SSH2 CLIENT 0: Using kex_algo = diffie-hellman-group-exchange-sha1
*Sep  6 14:44:08.497 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_REQUEST sent
*Sep  6 14:44:08.497 IST: SSH2 CLIENT 0: Range sent- 2048  < 2048  < 4096
*Sep  6 14:44:08.497 IST: SSH2 0: SSH2_MSG_KEX_DH_GEX_REQUEST received
*Sep  6 14:44:08.497 IST: SSH2 0: Range sent by client is - 2048 < 2048 < 4096
*Sep  6 14:44:08.497 IST: SSH2 0:  Modulus size established : 2048 bits
*Sep  6 14:44:08.510 IST: SSH2 0: expecting SSH2_MSG_KEX_DH_GEX_INIT
*Sep  6 14:44:08.510 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_GROUP received
*Sep  6 14:44:08.510 IST: SSH2 CLIENT 0: Server has chosen 2048 -bit dh keys
*Sep  6 14:44:08.523 IST: SSH2 CLIENT 0: expecting SSH2_MSG_KEX_DH_GEX_REPLY
*Sep  6 14:44:08.524 IST: SSH2 0: SSH2_MSG_KEXDH_INIT received
*Sep  6 14:44:08.555 IST: SSH2: kex_derive_keys complete
*Sep  6 14:44:08.555 IST: SSH2 0: SSH2_MSG_NEWKEYS sent
*Sep  6 14:44:08.555 IST: SSH2 0: waiting for SSH2_MSG_NEWKEYS
*Sep  6 14:44:08.555 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_REPLY received
*Sep  6 14:44:08.555 IST: SSH2 CLIENT 0: Skipping ServerHostKey Validation
*Sep  6 14:44:08.571 IST: SSH2 CLIENT 0: signature length 271
*Sep  6 14:44:08.571 IST: SSH2: kex_derive_keys complete
*Sep  6 14:44:08.571 IST: SSH2 CLIENT 0: SSH2_MSG_NEWKEYS sent
*Sep  6 14:44:08.571 IST: SSH2 CLIENT 0: waiting for SSH2_MSG_NEWKEYS
*Sep  6 14:44:08.571 IST: SSH2 CLIENT 0: SSH2_MSG_NEWKEYS received
*Sep  6 14:44:08.571 IST: SSH2 0: SSH2_MSG_NEWKEYS received
*Sep  6 14:44:08.571 IST: SSH2 0: Authentications that can continue =
publickey,keyboard-interactive,password
*Sep  6 14:44:08.572 IST: SSH2 0: Using method = none
*Sep  6 14:44:08.572 IST: SSH2 0: Authentications that can continue =
publickey,keyboard-interactive,password
*Sep  6 14:44:08.572 IST: SSH2 0: Using method = keyboard-interactive
*Sep  6 14:44:11.983 IST: SSH2 0: authentication successful for cisco
*Sep  6 14:44:11.984 IST: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: cisco] [Source:
192.168.121.40] [localport: 22] at 14:44:11 IST Thu Sep 6 2018
*Sep  6 14:44:11.984 IST: SSH2 0: channel open request
*Sep  6 14:44:11.985 IST: SSH2 0: pty-req request
*Sep  6 14:44:11.985 IST: SSH2 0: setting TTY - requested: height 24, width 80; set: height 24,
width 80
*Sep  6 14:44:11.985 IST: SSH2 0: shell request
*Sep  6 14:44:11.985 IST: SSH2 0: shell message received
*Sep  6 14:44:11.985 IST: SSH2 0: starting shell for vty
*Sep  6 14:44:22.066 IST: %SYS-6-LOGOUT: User cisco has exited tty session 1(192.168.121.40)
*Sep  6 14:44:22.166 IST: SSH0: Session terminated normally
*Sep  6 14:44:22.167 IST: SSH CLIENT0: Session terminated normally
```

ステップ 5 debug ip packet

IP パケット詳細のデバッグをオンにします。

例：

```
Device# debug ip packet
```

ステップ 6 show log

デバッグメッセージログを表示します。

例：

```
Device# show log

yslog logging: enabled (0 messages dropped, 9 messages rate-limited, 0 flushes, 0 overruns, xml
disabled, filtering disabled)

No Active Message Discriminator.



No Inactive Message Discriminator.


    Console logging: disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 1363 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    File logging: disabled
    Persistent logging: disabled

No active filter modules.

    Trap logging: level informational, 176 message lines logged
        Logging Source-Interface:       VRF Name:

Log Buffer (4096 bytes):
bleid=0, s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, sending
*Sep  6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.177 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.177 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, sending
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, sending
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
```

```
 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, sending
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, sending
*Sep  6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
 len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
 (FortyGigabitEthernet1/0/1), routed via RIB
```

SSH 認証用の X.509v3 証明書の設定例

例：サーバ認証用のデジタル証明書の設定

```
Switch> enable
Switch# configure terminal
Switch(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa
Switch(config)# ip ssh server certificate profile
Switch(ssh-server-cert-profile)# server
Switch(ssh-server-cert-profile-server)# trustpoint sign trust1
Switch(ssh-server-cert-profile-server)# exit
```

例：ユーザ認証用のデジタル証明書の設定

```
Switch> enable
Switch# configure terminal
Switch(config)# ip ssh server algorithm authentication publickey
Switch(config)# ip ssh server algorithm publickey x509v3-ssh-rsa
Switch(config)# ip ssh server certificate profile
Switch(ssh-server-cert-profile)# user
Switch(ssh-server-cert-profile-user)# trustpoint verify trust2
Switch(ssh-server-cert-profile-user)# end
```

SSH 認証用の X.509v3 証明書に関するその他の参考資料

関連資料

関連項目	マニュアル タイトル
PKI 設定	PKI 展開での Cisco IOS 証明書サーバの設定および管理

シスコのテクニカル サポート

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

SSH 認証用の X.509v3 証明書の機能情報

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

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

表 *129 : SSH* 認証用の *X.509v3* 証明書の機能情報

機能名	リリース	機能情報
SSH 認証の X.509v3 証明書	Cisco IOS 15.2(4)E1	SSH 認証の X.509v3 証明書機能は、サーバー内で X.509v3 デジタル証明書を使用し、SSH サーバー側でユーザー認証を使用します。 次のコマンドが導入または変更されました。**ip ssh server algorithm hostkey**、**ip ssh server algorithm authentication**、**ip ssh server certificate profile** この機能は、次のプラットフォームに実装されていました。 • Catalyst 2960C、2960CX、2960P、2960X、および 2960XR シリーズ スイッチ • Catalyst 3560CX および 3560X シリーズ スイッチ • Catalyst 3750X シリーズ スイッチ • Catalyst 4500E Sup7-E、Sup7L-E、Sup8-E および 4500X シリーズ スイッチ • Catalyst 4900M、4900F-E シリーズ スイッチ

第 60 章

Secure Socket Layer HTTP の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

Secure Sockets Layer（SSL）HTTP に関する情報

セキュア HTTP サーバおよびクライアントの概要

セキュア HTTP 接続の場合、HTTP サーバが送受信するデータは暗号化されてインターネットに送信されます。SSL 暗号化を伴う HTTP は、Web ブラウザからスイッチを設定するような機能に、セキュアな接続を提供します。シスコが実装するセキュア HTTP サーバおよび HTTP クライアントでは、アプリケーション層の暗号化に SSL バージョン 3.0 を使用します。HTTP over SSL は、HTTPS と省略されます（セキュアな接続の場合、URL が http://の代わりに https://で始まります）。

（注） SSL は 1999 年に Transport Layer Security（TLS）に発展しましたが、このような特定のコンテキストでまだ使用されています。

セキュア HTTP サーバ（スイッチ）の主な役割は、指定のポート（デフォルトの HTTPS ポートは 443）で HTTPS 要求を待ち受けて、HTTP 1.1 Web サーバへその要求を渡すことです。HTTP 1.1 サーバはその要求を処理して、セキュア HTTP サーバへ応答（呼び出す）します。セキュア HTTP サーバは HTTP 1.1 サーバの代わりに、元の要求に応えます。

セキュア HTTP クライアント（Web ブラウザ）の主な役割は、Cisco IOS アプリケーション要求に応答して、そのアプリケーションが要求した HTTPS User Agent サービスを実行し、応答を（そのアプリケーションに）返すことです。

CA のトラストポイント

認証局（CA）は、要求を認可して参加するネットワーク デバイスに証明書を発行します。これらのサービスは、参加するデバイスに対する中央集中的なセキュリティ キーおよび証明書の管理を提供します。特定の CA サーバはトラストポイントと呼ばれます。

接続が実行されると、HTTPS サーバは、トラストポイントとなる特定の CA から得た X.509v3 の証明書を発行することで、セキュアな接続をクライアントに提供します。クライアント（通常、Web ブラウザ）は、その証明書の認証に必要な公開キーを保有しています。

セキュア HTTP 接続には、CA のトラストポイントを設定することを強く推奨します。HTTPS サーバを実行しているデバイスに CA のトラストポイントが設定されていないと、サーバは自身を認証して必要な RSA のキーのペアを生成します。自身で認証した（自己署名）証明書は適切なセキュリティではないので、接続するクライアントはその証明書が自己証明書であることを通知し、ユーザに接続の選択（確立または拒否）をさせる必要があります。この選択肢は内部ネットワーク トポロジ（テスト用など）に役立ちます。

CA のトラストポイントを設定していないと、セキュア HTTP 接続を有効にした場合、そのセキュア HTTP サーバ（またはクライアント）に対する一時的または永続的な自己署名証明書が自動的に生成されます。

- スイッチにホスト名とドメイン名が設定されてない場合、生成される自己署名証明書は一時的なものです。スイッチを再起動すると、この一時的な自己署名証明書は失われ、新たに自己署名証明書（一時的に）が割り当てられます。
- スイッチにホスト名とドメイン名が設定されている場合、生成される自己署名証明書は永続的なものです。この証明書は、スイッチを再起動しても、セキュア HTTP サーバーを無効にしても有効のままです。そのため、再度セキュア HTTP 接続を有効にしたときに使用できます。

（注） 認証局およびトラストポイントは、個々のデバイスで設定する必要があります。他のデバイスからコピーすると、それらはスイッチ上で無効になります。

新しい証明書を登録した場合、新しい設定の変更は、サーバが再起動するまで HTTPS サーバに適用されません。CLI を使用するか、または物理的な再起動によって、サーバを再起動できます。サーバーを再起動すると、スイッチは新しい証明書の使用を開始します。

自己署名証明書が生成された場合、その情報は **show running-config** 特権 EXEC コマンドで出力できます。自己署名証明書を表示するコマンドの出力（show running-config コマンド）を例として一部示します。

```
スイッチ# show running-config
Building configuration...

<output truncated>

crypto pki trustpoint TP-self-signed-3080755072
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-3080755072
 revocation-check none
 rsakeypair TP-self-signed-3080755072
!
!
crypto ca certificate chain TP-self-signed-3080755072
 certificate self-signed 01
  3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
  02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
  30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D

<output truncated>
```

自己署名証明書は、セキュア HTTP サーバを無効にして、**no crypto pki trustpoint TP-self-signed-30890755072** グローバル コンフィギュレーション コマンドを入力することで削除できます。その後、セキュア HTTP サーバを再度有効にすると、自己署名証明書が新たに生成されます。

（注） *TP self-signed* の後ろに表示されている値は、デバイスのシリアル番号によって異なります。

オプションのコマンド（**ip http secure-client-auth**）を使用すると、HTTPS サーバがクライアントからの X.509v3 証明書を要求します。クライアントの認証は、サーバ自身の認証よりも高いセキュリティを提供します。

CipherSuite

CipherSuite は暗号化アルゴリズムおよびダイジェストアルゴリズムを指定して、SSL 接続に使用します。HTTPS サーバに接続すると、クライアントの Web ブラウザは、サポート対象の CipherSuite のリストを提供します。その後クライアントとサーバは、両方でサポートされている暗号化アルゴリズムで最適なものをリストから選択してネゴシエートします。たとえば、Netscape Communicator 4.76 は、米国のセキュリティ（RSA 公開キー暗号 MD2、MD5、RC2-CBC、RC4、DES-CBC、および DES-EDE3-CBC）をサポートしています。

最適な暗号化には、128 ビット暗号化をサポートするクライアントブラウザ（Microsoft Internet Explorer バージョン 5.5 以降または Netscape Communicator バージョン 4.76 以降など）が必要です。SSL_RSA_WITH_DES_CBC_SHA CipherSuite は、128 ビット暗号化を提供しないため、他の CipherSuite よりもセキュリティが低くなります。

CipherSuite は、よりセキュリティが高く、複雑になればなるほど、わずかですが処理時間が必要になります。次に、スイッチでサポートされる CipherSuite およびルータの処理負荷（速さ）による CipherSuite のランク（速い順）を定義します。

1. SSL_RSA_WITH_DES_CBC_SHA：メッセージの暗号化に DES-CBC、およびメッセージダイジェストに SHA を使用した RSA のキー交換（RSA 公開キー暗号化）
2. SSL_RSA_WITH_NULL_SHA：メッセージの暗号化に NULL、およびメッセージ ダイジェストに SHA を使用したキー交換（SSL 3.0 専用）。
3. SSL_RSA_WITH_NULL_MD5：メッセージの暗号化に NULL、およびメッセージ ダイジェストに MD5 を使用したキー交換（SSL 3.0 専用）。
4. SSL_RSA_WITH_RC4_128_MD5：RC4 128 ビット暗号化、およびメッセージ ダイジェストに MD5 を使用した RSA のキー交換
5. SSL_RSA_WITH_RC4_128_SHA：RC4 128 ビット暗号化、およびメッセージ ダイジェストに SHA を使用した RSA のキー交換
6. SSL_RSA_WITH_3DES_EDE_CBC_SHA：メッセージの暗号化に 3DES と DES-EDE3-CBC、およびメッセージ ダイジェストに SHA を使用した RSA のキー交換（RSA 公開キー暗号化）
7. SSL_RSA_WITH_AES_128_CBC_SHA：AES 128 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換（SSL 3.0 専用）。
8. SSL_RSA_WITH_AES_256_CBC_SHA：AES 256 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換（SSL 3.0 専用）。
9. SSL_RSA_WITH_AES_128_CBC_SHA：AES 128 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換（SSL 3.0 専用）。
10. SSL_RSA_WITH_AES_256_CBC_SHA：AES 256 ビット暗号化、およびメッセージダイジェストに SHA を使用した RSA のキー交換（SSL 3.0 専用）。

（注） Chromeの最新バージョンは4つの元の暗号スイートをサポートしません。そのため、Web GUI とゲスト ポータル両方へのアクセスが拒否されます。

（暗号化およびダイジェスト アルゴリズムをそれぞれ指定して組み合わせた）RSA は、SSL 接続においてキーの生成および認証の両方に使用されます。これは、CA のトラストポイントが設定されているかどうかにかかわりません。

SSL のデフォルト設定

標準の HTTP サーバはイネーブルに設定されています。

SSL はイネーブルに設定されています。

CA のトラストポイントは設定されていません。

自己署名証明書は生成されていません。

SSL の設定時の注意事項

SSL をスイッチ クラスタで使用すると、SSL セッションがクラスタ コマンダで終了します。クラスタ メンバのスイッチは標準の HTTP で動作させる必要があります。

CA のトラストポイントを設定する前に、システム クロックが設定されていることを確認してください。クロックが設定されていないと、不正な日付により証明書が拒否されます。

スイッチスタック内のアクティブスイッチで、SSL セッションが終了します。

セキュア HTTP サーバおよびクライアントの設定方法

CA のトラストポイントの設定

セキュア HTTP 接続には、CA のトラストポイントを正式に設定することを推奨します。CA のトラストポイントは、自己署名証明書より高いセキュリティがあります。

CA のトラストポイントを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **hostname** *hostname*
3. **ip domain-name** *domain-name*
4. **crypto key generate rsa**
5. **crypto ca trustpoint** *name*
6. **enrollment url** *url*
7. **enrollment http-proxy** *host-name port-number*

8. **crl query** *url*
9. **primary** *name*
10. **exit**
11. **crypto ca authentication** *name*
12. **crypto ca enroll** *name*
13. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**hostname** *hostname* 例： `スイッチ(config)# hostname your_hostname`	スイッチのホスト名を指定します（以前ホスト名を設定していない場合のみ必須）。ホスト名はセキュリティ キーと証明書に必要です。
ステップ 3	**ip domain-name** *domain-name* 例： `スイッチ(config)# ip domain-name your_domain`	スイッチの IP ドメイン名を指定します（以前 IP ドメイン名を設定していない場合のみ必須）。IP ドメイン名はセキュリティキーと証明書に必要です。
ステップ 4	**crypto key generate rsa** 例： `スイッチ(config)# crypto key generate rsa`	（任意）RSA キー ペアを生成します。RSA キーのペアは、スイッチの証明書を入手する前に必要です。RSA キーのペアは自動的に生成されます。必要であれば、このコマンドを使用してキーを再生成できます。
ステップ 5	**crypto ca trustpoint** *name* 例： `スイッチ(config)# crypto ca trustpoint your_trustpoint`	CA のトラストポイントにローカルの設定名を指定して、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 6	**enrollment url** *url* 例： `スイッチ(ca-trustpoint)# enrollment url http://your_server:80`	スイッチによる証明書要求の送信先の URL を指定します。

	コマンドまたはアクション	目的
ステップ 7	**enrollment http-proxy** *host-name port-number* 例： `スイッチ(ca-trustpoint)# enrollment http-proxy your_host 49`	（任意）HTTP プロキシ サーバーを経由して CA から証明書を入手するようにスイッチを設定します。 • *host-name* には、CA を取得するために使用するプロキシ サーバを指定します。 • *port-number* には、CA にアクセスするために使用するポート番号を指定します。
ステップ 8	**crl query** *url* 例： `スイッチ(ca-trustpoint)# crl query ldap://your_host:49`	ピアの証明書が取り消されていないかを確認するために、証明書失効リスト（CRL）を要求するようにスイッチを設定します。
ステップ 9	**primary** *name* 例： `スイッチ(ca-trustpoint)# primary your_trustpoint`	（任意）トラストポイントが CA 要求に対してプライマリ（デフォルト）トラストポイントとして使用されるように指定します。 • *name* には、設定したトラストポイントを指定します。
ステップ 10	**exit** 例： `スイッチ(ca-trustpoint)# exit`	CA トラストポイント コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 11	**crypto ca authentication** *name* 例： `スイッチ(config)# crypto ca authentication your_trustpoint`	CA の公開キーを取得して CA を認証します。ステップ 5 で使用した名前と同じものを使用します。
ステップ 12	**crypto ca enroll** *name* 例： `スイッチ(config)# crypto ca enroll your_trustpoint`	指定した CA トラストポイントから証明書を取得します。このコマンドは、各 RSA キーのペアに対して 1 つの署名入りの証明書を要求します。
ステップ 13	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

セキュア HTTP サーバの設定

セキュア HTTP サーバを設定するには、特権 EXEC モードで次の手順を実行します。

始める前に

証明に証明書の認証を使用する場合、前の手順を使用してスイッチの CA トラストポイントを設定してから、HTTP サーバーを有効にする必要があります。CA のトラストポイントを設定していない場合、セキュア HTTP サーバを最初に有効にした時点で、自己署名証明書が生成されます。サーバを設定した後、標準およびセキュア HTTP サーバ両方に適用するオプション（パス、適用するアクセス リスト、最大接続数、またはタイムアウト ポリシー）を設定できます。

Web ブラウザを使用してセキュア HTTP 接続を確認するには、https://*URL* を入力します（URL は IP アドレス、またはサーバー スイッチのホスト名）。デフォルト ポート以外のポートを設定している場合、URL の後ろにポート番号も指定する必要があります。次に例を示します。

（注） AES256_SHA2 はサポートされません。

```
https://209.165.129:1026
```

または

```
https://host.domain.com:1026
```

アクセスリスト（IPv4 ACL のみ）を指定するための従来の **ip http access-class** *access-list-number* コマンドは廃止予定です。引き続きこのコマンドを使用して、HTTP サーバへのアクセスを許可するアクセス リストを指定できます。2 つの新しいコマンドは、IPv4 および IPv6 ACL を指定するためのサポートを有効にするために導入されました。これらは、IPv4 ACL を指定するための **ip http access-class ipv4** *access-list-name* | *access-list-number* と、IPv6 ACL を指定するための **ip http access-class ipv6** *access-list-name* です。警告メッセージの受信を防ぐために、新しい CLI の使用をお勧めします。

アクセス リストを指定する際は、次の考慮事項があります。

- 存在しないアクセス リストを指定すると、設定は実行されますが、次の警告メッセージを受信します。

  ```
  ACL being attached does not exist, please configure it
  ```

- HTTP サーバにアクセスリストを指定するために **ip http access-class** コマンドを使用すると、次の警告メッセージが表示されます。

  ```
  This CLI will be deprecated soon, Please use new CLI ip http
  access-class ipv4/ipv6 <access-list-name>| <access-list-number>
  ```

- **ip http access-class ipv4** *access-list-name* | *access-list-number* または **ip http access-class ipv6** *access-list-name* を使用した場合に、アクセスリストがすでに **ip http access-class** を使用して設定されていた場合は、次の警告メッセージが表示されます。

  ```
  Removing ip http access-class <access-list-number>
  ```

ip http access-class *access-list-number* と **ip http access-class ipv4** *access-list-name* | *access-list-number* は同じ機能を共有しています。コマンドを実行するごとに、その前のコマンドのコンフィギュレーションは上書きされます。2 つのコマンドの設定間の次の組み合わせによって、実行コンフィギュレーションへの影響が説明されます。

- **ip http access-class** *access-list-number* がすでに設定されている場合に、**ip http access-class ipv4** *access-list-number* コマンドを使用して設定を行おうとした場合、**ip http access-class** *access-list-number* の設定は削除され、**ip http access-class ipv4** *access-list-number* の設定が実行コンフィギュレーションに追加されます。
- **ip http access-class** *access-list-number* がすでに設定されている場合に、**ip http access-class ipv4** *access-list-name* コマンドを使用して設定を行おうとした場合、**ip http access-class** *access-list-number* の設定は削除され、**ip http access-class ipv4** *access-list-name* の設定が実行コンフィギュレーションに追加されます。
- **ip http access-class ipv4** *access-list-number* がすでに設定されている場合に、**ip http access-class** *access-list-name* を使用して設定を行おうとした場合、**ip http access-class ipv4** *access-list-number* の設定は削除され、**ip http access-class** *access-list-name* の設定が実行コンフィギュレーションに追加されます。
- **ip http access-class ipv4** *access-list-name* がすでに設定されている場合に、**ip http access-class** *access-list-number* を使用して設定を行おうとした場合、**ip http access-class ipv4** *access-list-name* の設定は削除され、**ip http access-class** *access-list-number* の設定が実行コンフィギュレーションに追加されます。

手順の概要

1. **show ip http server status**
2. **configure terminal**
3. **ip http secure-server**
4. **ip http secure-port** *port-number*
5. **ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}
6. **ip http secure-client-auth**
7. **ip http secure-trustpoint** *name*
8. **ip http path** *path-name*
9. **ip http access-class** *access-list-number*
10. **ip http access-class** { **ipv4** {*access-list-number* | *access-list-name*} | **ipv6** {*access-list-name*} }
11. **ip http max-connections** *value*
12. **ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*
13. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**show ip http server status** 例： スイッチ# **show ip http server status**	（任意）HTTP サーバのステータスを表示して、セキュア HTTP サーバの機能がソフトウェアでサポートされているかどうかを判断します。出力で、次のラインのどちらかを確認してください。 `HTTP secure server capability: Present` または `HTTP secure server capability: Not present`
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip http secure-server** 例： スイッチ(config)# **ip http secure-server**	HTTPS サーバがディセーブルの場合、イネーブルにします。HTTPS サーバは、デフォルトでイネーブルに設定されています。
ステップ 4	**ip http secure-port** *port-number* 例： スイッチ(config)# **ip http secure-port 443**	（任意）HTTPS サーバに使用するポート番号を指定します。デフォルトのポート番号は 443 です。443 または 1025 ～ 65535 の範囲で指定できます。
ステップ 5	**ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} 例： スイッチ(config)# **ip http secure-ciphersuite rc4-128-md5**	（任意）HTTPS 接続の暗号化に使用する CipherSuite（暗号化アルゴリズム）を指定します。特定の CipherSuite を指定する理由がなければ、サーバとクライアントが、両方がサポートする CipherSuite でネゴシエートするように設定します。これがデフォルトです。
ステップ 6	**ip http secure-client-auth** 例： スイッチ(config)# **ip http secure-client-auth**	（任意）HTTP サーバを設定して、接続処理の間、認証のために、クライアントからの X.509v3 証明書を要求します。デフォルトでは、クライアントがサーバからの証明書を要求する設定になっていますが、サーバはクライアントを認証しないようになっています。

	コマンドまたはアクション	目的
ステップ 7	**ip http secure-trustpoint** *name* 例： `スイッチ(config)# ip http secure-trustpoint your_trustpoint`	X.509v3 セキュリティ証明書の取得およびクライアントの証明書接続の認証に使用する CA のトラストポイントを指定します。 （注） このコマンドの使用は、前の手順に従って CA のトラストポイントをすでに設定しているという前提を踏まえて説明しています。
ステップ 8	**ip http path** *path-name* 例： `スイッチ(config)# ip http path /your_server:80`	（任意）HTML ファイルのベースとなる HTTP パスを設定します。パスは、ローカル システムにある HTTP サーバ ファイルの場所を指定します（通常、システムのフラッシュ メモリを指定します）。
ステップ 9	**ip http access-class** *access-list-number* 例： `スイッチ(config)# ip http access-class 2`	（任意）HTTP サーバへのアクセスの許可に使用するアクセス リストを指定します。
ステップ 10	**ip http access-class** { **ipv4** {*access-list-number* | *access-list-name*} | **ipv6** {*access-list-name*} } 例： `スイッチ(config)# ip http access-class ipv4 4`	（任意）HTTP サーバへのアクセスの許可に使用するアクセス リストを指定します。
ステップ 11	**ip http max-connections** *value* 例： `スイッチ(config)# ip http max-connections 4`	（任意）HTTP サーバへの同時最大接続数を指定します。値は 10 以上にすることを推奨します。これは、UI が想定どおりに機能するために必要な値です。
ステップ 12	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value* 例： `スイッチ(config)# ip http timeout-policy idle 120 life 240 requests 1`	（任意）指定の状況下における、HTTP サーバへの接続最大時間を指定します。 • **idle**：データの受信がないか、応答データが送信できない場合の最大時間。指定できる範囲は 1 ～ 600 秒です。デフォルト値は 180 秒（3 分）です。 • **life**：接続を確立している最大時間。指定できる範囲は 1 ～ 86400 秒（24 時間）です。デフォルト値は 180 秒です。

	コマンドまたはアクション	目的
		• **requests**：永続的な接続で処理される要求の最大数。最大値は86400です。デフォルトは1です。
ステップ 13	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

セキュア HTTP クライアントの設定

セキュア HTTP クライアントを設定するには、特権 EXEC モードで次の手順を実行します。

始める前に

標準の HTTP クライアントおよびセキュア HTTP クライアントは常にイネーブルです。証明書の認証にはセキュア HTTP クライアントの証明書が必要です。次の手順では、前の手順で CA のトラストポイントをスイッチに設定していることを前提にしています。CA のトラストポイントが設定されておらず、リモートの HTTPS サーバーがクライアントの認証を要求した場合、セキュア HTTP クライアントへの接続は失敗します。

手順の概要

1. **configure terminal**
2. **ip http client secure-trustpoint** *name*
3. **ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip http client secure-trustpoint** *name* 例： スイッチ(config)# **ip http client secure-trustpoint your_trustpoint**	（任意）リモートの HTTP サーバがクライアント認証を要求した場合に使用する、CA のトラストポイントを指定します。このコマンドの使用は、前の手順を使用して CA のトラストポイントをすでに設定しているという前提を踏まえて説明しています。クライアント認証が必要ない場合、またはプライマリ

	コマンドまたはアクション	目的
		のトラストポイントがすでに設定されている場合は、このコマンドは任意です。
ステップ 3	**ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} 例： `スイッチ(config)# ip http client secure-ciphersuite rc4-128-md5`	（任意）HTTPS 接続の暗号化に使用する CipherSuite（暗号化アルゴリズム）を指定します。特定の CipherSuite を指定する理由がなければ、サーバとクライアントが、両方がサポートする CipherSuite でネゴシエートするように設定します。これがデフォルトです。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

セキュア HTTP サーバおよびクライアントのステータスのモニタリング

SSL セキュア サーバおよびクライアントのステータスをモニタするには、次の表の特権 EXEC コマンドを使用します。

表 *130 : SSL* セキュア サーバおよびクライアントのステータスを表示するコマンド

コマンド	目的
show ip http client secure status	セキュア HTTP クライアントの設定を表示します。
show ip http server secure status	セキュア HTTP サーバの設定を表示します。
show running-config	セキュア HTTP 接続に対して生成された自己署名証明書を表示します。

第 61 章

認証局の相互運用性

この章では、IPSecプロトコルをサポートするために提供される、認証局（CA）の相互運用性を設定する方法について説明します。CA の相互運用性により、Cisco IOS デバイスと CA の通信が可能になり、Cisco IOS デバイスが CA からデジタル証明書を取得して使用できるようになります。IPSec は CA を使用せずにネットワークで実装できますが、CA を使用すると、IPSec の管理性と拡張性が提供されます。

認証局の前提条件

この相互運用性機能の設定を行う前に、ネットワークで認証局（CA）が使用可能になっている必要があります。CA が公開キーインフラストラクチャ（PKI）プロトコルと Simple Certificate Enrollment Protocol（SCEP）プロトコルをサポートしている必要があります。

認証局の制約事項

CA を設定する際には次の制約事項が適用されます。

- この機能を設定する必要があるのは、ネットワークに IPSec およびインターネットキー交換（IKE）を両方とも設定する場合だけです。
- Cisco IOS ソフトウェアでは、長さが 2048 ビットを超える CA サーバ公開キーはサポートされていません。

認証局について

CA でサポートされる規格

認証局（CA）の相互運用性がなければ、Cisco IOS デバイスは IPSec 実装時に CA を使用することができません。CA は、IPSec ネットワークに管理可能なスケーラブル ソリューションを提供します。

シスコでは、この機能で次の規格をサポートしています。

- IPsec：IPsec は、参加しているピア間のデータ機密性、データ整合性、およびデータ認証を提供するオープン スタンダードのフレームワークです。IPSec は、IP レイヤでこれらのセキュリティ サービスを提供し、インターネット キー交換を使用して、ローカルポリシーに基づいたプロトコルとアルゴリズムのネゴシエーションの処理を行い、IPSec で使用する暗号化および認証キーを生成します。IPSec を使用することにより、ホスト ペア間、セキュリティ ゲートウェイ ペア間、またはセキュリティ ゲートウェイとホスト間の 1 つ以上のデータ フローを保護できます。
- インターネット キー交換（IKE）：Oakley キー交換や Skeme キー交換をインターネット セキュリティ アソシエーション キー管理プロトコル（ISAKMP）フレームワーク内部に実装したハイブリッドプロトコルです。IKE は他のプロトコルで使用できますが、その初期実装時は IPSec プロトコルで使用します。IKE は、IPsec ピアの認証、IPsec キーのネゴシエーションを提供し、IPsec セキュリティ アソシエーションのネゴシエーションを実行します。
- Public-Key Cryptography Standard #7（PKCS #7）：証明書登録メッセージの暗号化および署名に使用される RSA Data Security, Inc. の標準。
- Public-Key Cryptography Standard #10（PKCS #10）：証明書要求のための RSA Data Security, Inc. の標準構文。
- RSA キー：RSA は公開キー暗号化システムで、Ron Rivest、Adi Shamir、Leonard Adleman の 3 名によって開発されました。RSA キーは、1 つの公開キーと 1 つの秘密キーのペアになっています。
- X.509v3 証明書：同等のデジタル ID カードを各デバイスに提供することで、IPSec で保護されたネットワークの拡張を可能にする証明書サポート。2 つの装置が通信する際、デジタル証明書を交換することで ID を証明します（これにより、各ピアが公開キーを手動で交換したり、各ピアで共有キーを手動で指定したりする必要がなくなります）。これらの証明書は CA から取得されます。X.509 は、ITU の X.500 標準の一部です。

CA の目的

認証局（CA）は、証明書要求を管理し、関係する IP セキュリティ ネットワーク デバイスへの証明書を発行します。これらのサービスは、参加デバイスのキー管理を一元化して行います。

CA は、IPSec ネットワーク デバイスの管理を簡素化します。CA は、ルータなど、複数の IPSec 対応デバイスを含むネットワークで使用できます。

公開キー暗号化によりイネーブルにされたデジタル署名は、デバイスおよび個人ユーザをデジタル認証します。RSA 暗号化システムなどの Public Key Cryptography では、各ユーザは、公開キーと秘密キーの両方を含むキーペアを使用します。これらのキーは、補足として機能し、一方で暗号化されたものは、もう一方で復号化できます。つまり、署名は、データがユーザの秘密キーで暗号化されるときに形成されます。受信者は、送信者の公開キーを使用してメッセージを復号化することで、署名を検証します。送信側の公開キーを使用してメッセージを復号できたという事実から、そのメッセージが秘密キーの所有者つまり送信者によって作成されたことがわかります。このプロセスでは、受信者が送信者の公開キーのコピーを持っていること、およびそのキーが送信者になりすました別人ではなく送信者本人のものであることを受信者が強く確信していることが重要です。

デジタル証明書はリンクを提供します。デジタル証明書には、名前、シリアル番号、企業、部署または IP アドレスなど、ユーザまたはデバイスを特定する情報を含んでいます。また、エンティティの公開キーのコピーも含まれています。証明書自体は、受信者が身元を証明しデジタル証明書を作成するうえで確実に信頼できるサード パーティである、認証局（CA）により署名されます。

CA の署名を検証するには、受信者が CA の公開キーを認識している必要があります。このプロセスは通常、アウトオブバンド、またはインストール時に実行される操作によって処理されます。たとえば、通常の Web ブラウザでは、デフォルトで、複数の CA の公開キーが設定されています。IPSec の基本コンポーネントであるインターネット キー交換（IKE）は、デジタル署名を使用して、セキュリティ アソシエーションを設定する前にピア デバイスをスケーラブルに認証できます。

デジタル署名がない場合は、IPSec を使用するデバイスの各ペア間で公開キーまたはシークレットを手動で交換して、通信を保護する必要があります。証明書がない場合、ネットワークに新しいデバイスが追加されるたびに、安全に通信を行う他のすべてのデバイスで設定を変更する必要があります。デジタル証明書がある場合、各デバイスは、認証局に登録されます。2 台のデバイスが通信する場合、証明書を交換し、データをデジタル署名して、お互いを認証します。ネットワークに新しいデバイスを追加する場合には、そのデバイスを CA に登録するだけでよく、他のデバイスの設定を変更する必要はありません。新しいデバイスが IPSec 接続を試行すると、証明書が自動的に交換され、デバイスを認証できます。

CA なしでの IPsec の実装

CA を使用せずに、2 つの Cisco デバイス間で IPSec サービス（暗号化など）を有効にする場合、最初に、各デバイスにもう一方のデバイスのキー（RSA 公開キーや共有キー）が存在することを確認する必要があります。つまり、次のいずれかの操作を手動で実行する必要があります。

- 各デバイスで、もう一方のデバイスの RSA 公開キーを入力します。
- 各デバイスで、両方のデバイスに使用される共有キーを指定します。

上の図では、各デバイスが他方のデバイスのキーを使用して、他方のデバイスのアイデンティティを認証します。この認証は、2 台のデバイス間で IPsec トラフィックが交換される場合には必ず実行されます。

複数の Cisco デバイスをメッシュ トポロジで配置し、すべてのデバイス間で IPSec トラフィックを交換させる場合には、最初に、すべてのデバイス間に共有キーまたは RSA 公開キーを設定する必要があります。

IPSec ネットワークに新しいデバイスを追加するごとに、新しいデバイスと既存の各デバイス間にキーを設定する必要があります。（図 34 の場合、このネットワークに 1 台の暗号化デバイスを追加するには、新たに 4 つのスイッチ間キー設定が必要になります）。

したがって、IPSec サービスを必要とするデバイスが増えるほど、キー管理は複雑になります。このアプローチでは、より大型で複雑な暗号化ネットワークには拡張できません。

CA での IPsec の実装

CA では、すべての暗号化デバイス間にキーを設定する必要はありません。代わりに、加入させる各デバイスを CA に個別に登録し、各デバイスの証明書を要求します。この設定が完了していれば、各加入デバイスは、他のすべての加入デバイスをダイナミックに認証できます。このプロセスについて、図で説明します。

ネットワークに新しい IPSec デバイスを追加する場合、新しいデバイスが CA に証明書を要求するように設定するだけでよく、既存の他のすべての IPSec デバイスとの間に複数のキー設定を行う必要はありません。

複数のルート CA での IPsec の実装

複数のルート CA がある場合、証明書をピアに発行した CA にデバイスを登録する必要はありません。その代わり、信頼できる複数の CA にデバイスを設定します。そのため、デバイスは、設定された CA（信頼できるルート）を使用して、デバイス ID で定義されている同じ CA 以外から発行された証明を、ピアが提供したかどうかを検証できます。

複数の CA を設定することにより、IKE を使用して IPSec トンネルを確立する場合に、異なるドメイン（異なる CA）に登録した 2 台以上のデバイス間で相互の ID を確認できます。

Simple Certificate Enrollment Protocol（SCEP）では、各デバイスは、CA（登録 CA）で設定されます。CA は、CA の秘密キーで署名されるデバイスに証明書を発行します。同じドメインのピアの証明書を確認するため、デバイスは、登録 CA のルート証明書でも設定されます。

異なるドメインからのピアの証明書を確認するには、そのピアのドメインの登録 CA のルート証明書をデバイスで安全に設定する必要があります。

インターネット キー交換（IKE）フェーズ I の署名の検証中、発信側は CA 証明書のリストを応答側に送信します。応答側は、リストのいずれかの CA により発行される証明書を送信する必要があります。証明書が検証されると、デバイスは、証明書に含まれる公開キーを公開キーリングに保存します。

複数のルート CA がある場合、VPN ユーザーは、1 つのドメインで信頼を確立して、それを他のドメインに簡単かつ安全に配布できます。そのため、異なるドメインで認証されるエンティティ間の必要なプライベート通信チャネルが発生します。

IPSec デバイスによる CA 証明書の使用方法

IPSec で保護されたトラフィックを 2 台の IPSec デバイス間で交換させるには、最初に相互に認証しあう必要があります。認証されていない場合、IPSec 保護が適用されません。この認証を行うには、IKE を使用します。

CA を使用しない場合、デバイスは、RSA 暗号化ナンスまたは事前共有キーを使用して、リモートデバイスに対して自身を認証します。いずれの方式でも、2 つのデバイス間でキーを事前に設定しておく必要があります。

CA を使用する場合、デバイスはリモート デバイスに証明書を送信し、何らかの公開キー暗号化を実行することによって、リモート デバイスに対して自身を認証します。各デバイスは、CA により発行されて検証された、固有の証明書を送信する必要があります。このプロセスが有効なのは、各デバイスの証明書にデバイスの公開キーがカプセル化され、各証明書が CA によって認証されることにより、すべての加入デバイスが CA を認証局として認識するからです。この機構は、RSA シグニチャを使用する IKE と呼ばれます。

デバイスは、証明書が期限切れになるまで、複数の IPSec ピアに対して、複数の IPSec セッション用に自身の証明書を継続的に送信できます。証明書が期限満了になったときは、デバイスの管理者は新しい証明書を CA から入手する必要があります。

また、CA は、IPSec に参加しなくなったデバイスの証明書を失効できます。失効された証明書は、他の IPSec デバイスから有効とは見なされません。失効された証明書は、証明書失効リスト（CRL）にリストされ、各ピアは相手側ピアの証明書を受け入れる前に、このリストを確認できます。

登録局

一部の CA に、実装の一部として登録局（RA）があります。RA は本質的に CA のプロキシの役割を果たすサーバであるため、CA がオフラインのときも CA 機能は継続しています。

このマニュアルに記載されている設定タスクの一部は、CA での RA のサポートの有無によって、多少の違いがあります。

認証局の設定方法

NVRAM メモリ使用率の管理

CA 証明書が使用されるとき、デバイスは証明書と証明書失効リスト（CRL）を使用します。通常、一部の証明書とすべての CRL は、デバイスの NVRAM にローカルに保存されており、各証明書および CRL は相応な量のメモリを使用します。

通常、デバイスには次の証明書が保存されます。

- デバイスの証明書
- CA の証明書
- CA サーバから取得したルート証明書（デバイスが初期化された後、すべてのルート証明書が RAM に保存されます）
- 2 つの登録局（RA）証明書（CA が RA をサポートしている場合のみ）

CRL は通常、次の条件に従ってデバイスで保存されます。

- CA が RA をサポートしていない場合、デバイスには 1 つの CRL のみ保存されます。
- CA が RA をサポートしている場合、複数の CRL をデバイスに保存できます。

これらの証明書と CRL をローカルに保存することが、何の問題にもならない場合もあります。しかし、メモリの問題が起こる可能性もあります。特に、CA が RA をサポートし、デバイスに多数の CRL は保存しなければならない場合に起こりやすくなります。NVRAM が小さすぎてルート証明書を保存できない場合は、ルート証明書のフィンガープリントのみ保存されます。

NVRAM スペースを節約するには、証明書と CRL をローカルに保存せず、必要に応じて CA から取得するよう指定します。この代替策では、NVRAM スペースを節約できますが、パフォーマンスに多少影響が出る可能性があります。証明書と CRL をデバイスにローカル保存せず必要なときに取得するよう指定するには、クエリ モードを有効にします。

クエリ モードの有効化は、この時点ではなく後で実施することもできます。証明書と CRL がすでにデバイスに保存されている場合でも可能です。このような場合、クエリ モードを有効にすると、設定を保存した後、保存済みの証明書と CRL がデバイスから削除されます（クエリ モードを有効にする前に TFTP サイトに設定をコピーしておくと、保存されていたあらゆる証明書と CRL を TFTP サイトで保管することができます）。

クエリモードを無効にする前に、**copy system:running-config nvram:startup-config** コマンドを実行して、現在の証明書と CRL をすべて NVRAM に保存します。そうしないと、リブート時にこれらが失われることがあります。

証明書と CRL をデバイスにローカル保存せず必要なときに取得するよう指定するには、グローバル コンフィギュレーション モードで次のコマンドを使用して、クエリ モードを有効にします。

（注） クエリ モードは、CA がダウン状態にある場合、可用性に影響を及ぼす可能性があります。

手順の概要

1. **crypto ca certificate query**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**crypto ca certificate query** 例： `Device(config)# crypto ca certificate query`	クエリ モードを有効にします。これにより、証明書と CRL のローカル保存が行われなくなります。

デバイス ホスト名および IP ドメイン名の設定

デバイスのホスト名および IP ドメイン名が未設定の場合には、これを設定する必要があります。これが必要になるのは、IPSec によって使用されるキーおよび証明書にデバイスが完全修飾ドメイン名（FQDN）を割り当てており、デバイスに割り当てられたホスト名および IP ドメイン名に FQDN が基づいているためです。たとえば、「device20.example.com」という名前の証明書は、「device20」というデバイスのホスト名と「example.com」というデバイスの IP ドメイン名に基づいています。

手順の概要

1. **enable**
2. **configure terminal**
3. **hostname** *name*
4. **ip domain-name** *name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**hostname** *name* 例： `Device(config)# hostname device1`	デバイスのホスト名を設定します。
ステップ 4	**ip domain-name** *name* 例： `Device(config)# ip domain-name domain.com`	デバイスの IP ドメイン名を設定します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `Device(config)# end`	グローバル コンフィギュレーションを終了して、特権 EXEC モードに戻ります。

RSA キー ペアの生成

Rivest、Shamir、Adelman（RSA）キー ペアは IKE キー管理メッセージの署名および暗号化に使用されます。また、デバイスの証明書を取得する前に必要になります。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto key generate rsa** [**usage-keys**]
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto key generate rsa** [**usage-keys**] 例： `Device(config)# crypto key generate rsa usage-keys`	RSA キー ペアを生成します。 • **usage-keys** キーワードを使用して、汎用キーではなく特定目的のキーを指定します。
ステップ 4	**end** 例： `Device(config)# end`	グローバル コンフィギュレーションを終了して、特権 EXEC モードに戻ります。

認証局の宣言

デバイスが使用する 1 つの認証局（CA）を宣言する必要があります。

手順の概要

1. **enable**

2. **configure terminal**
3. **crypto ca trustpoint** *name*
4. **enrollment url** *url*
5. **enrollment command**
6. **exit**
7. **crypto pki trustpoint** *name*
8. **crl query ldap**://*url*:[*port*]
9. **enrollment** {**mode ra** | **retry count** *number* | **retry period** *minutes* | **url** *url*}
10. **enrollment** {**mode ra** | **retry count** *number* | **retry period** *minutes* | **url** *url*}
11. **revocation-check** *method1* [*method2 method3*]
12. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto ca trustpoint** *name* 例： `Device(config)# crypto ca trustpoint ka`	デバイスが使用する認証局（CA）を宣言し、CA プロファイル登録コンフィギュレーション モードを開始します。
ステップ 4	**enrollment url** *url* 例： `Device(ca-profile-enroll)# enrollment url http://entrust:81`	登録要求の送信先とする CA サーバの URL を指定します。
ステップ 5	**enrollment command** 例： `Device(ca-profile-enroll)# enrollment command`	登録のため CA に送信される HTTP コマンドを指定します。
ステップ 6	**exit** 例： `Device(ca-profile-enroll)# exit`	CA プロファイル登録コンフィギュレーション モードを終了してグローバル コンフィギュレーション モードに戻ります。
ステップ 7	**crypto pki trustpoint** *name* 例： `Device(config)# crypto pki trustpoint ka`	デバイスで使用するトラストポイントを宣言し、CA トラストポイント コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 8	**crl query ldap**://*url*:[*port*] 例： `Device(ca-trustpoint)# crl query ldap://bar.cisco.com:3899`	証明書失効リスト（CRL）を照会し、ピアの証明書が失効していないことを確認します。
ステップ 9	**enrollment** {**mode ra** | **retry count** *number* | **retry period** *minutes* | **url** *url*} 例： `Device(ca-trustpoint)# enrollment retry period 2`	証明書要求を再試行するまでの登録待機時間を指定します。
ステップ 10	**enrollment** {**mode ra** | **retry count** *number* | **retry period** *minutes* | **url** *url*} 例： `Device(ca-trustpoint)# enrollment retry count 8`	以前の要求への応答が得られない場合にデバイスが証明書要求を再送信する回数を指定します。
ステップ 11	**revocation-check** *method1* [*method2 method3*] 例： `Device(ca-trustpoint)# revocation-check crl ocsp`	証明書の失効ステータスをチェックします。
ステップ 12	**end** 例： `Device(ca-trustpoint)# end`	CA トラストポイントコンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

ルート CA（信頼できるルート）の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto ca trustpoint** *name*
4. **revocation-check** *method1* [*method2 method3*]
5. **root tftp** *server-hostname filename*
6. **enrollment http-proxy** *hostname port-number*
7. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto ca trustpoint** *name* 例： `Device(config)# crypto ca trustpoint ka`	デバイスで使用するトラストポイントを宣言し、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 4	**revocation-check** *method1* [*method2 method3*] 例： `Device(ca-trustpoint)# revocation-check ocsp`	証明書の失効ステータスをチェックします。
ステップ 5	**root tftp** *server-hostname filename* 例： `Device(ca-trustpoint)# root tftp server1 file1`	TFTP 経由で認証局（CA）の証明書を取得します。
ステップ 6	**enrollment http-proxy** *hostname port-number* 例： `Device(ca-trustpoint)# enrollment http-proxy host2 8080`	HTTP を使用して、プロキシ サーバ経由で認証局（CA）にアクセスします。
ステップ 7	**end** 例： `Device(ca-trustpoint)# end`	CA トラストポイント コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

CA の認証

デバイスは認証局（CA）を認証する必要があります。CA を認証するには、CA の公開キーが含まれている CA の自己署名証明書を取得します。この CA の証明書は自己署名（CA が自身の証明書に署名したもの）であるため、CA の公開キーは、この手順実行時に、CA の管理者に連絡して CA 証明書のフィンガープリントを比較することにより、手動で認証する必要があります。

CA の公開キーを取得するには次のタスクを実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto pki authenticate***name*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto pki authenticate***name* 例： `Device(config)# crypto pki authenticate myca`	CA の証明書を取得することにより CA を認証します。
ステップ 4	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

署名証明書の要求

デバイスの RSA キーペアごとに、認証局（CA）から署名証明書を取得する必要があります。汎用 RSA キーを生成した場合、デバイスは 1 組の RSA キー ペアだけを持ち、1 個の証明書だけが必要です。特定目的の RSA キーを以前に生成している場合、デバイスは 2 組の RSA キーペアを持ち、2 個の証明書が必要です。

CA から署名証明書を要求するには、次の作業を実行します。

（注） **crypto pki enroll** コマンドを発行した後、証明書を受信する前にデバイスがリブートされた場合は、コマンドを再発行して CA の管理者に連絡する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto pki enroll** *number*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto pki enroll** *number* 例： `Device(config)# crypto pki enroll myca`	CA からデバイスの証明書を取得します。
ステップ 4	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

次のタスク

設定の保存

設定の変更を行った場合は、必ず作業結果を保存するようにしてください。

copy system:running-config nvram:startup-config コマンドを使用して、設定を保存します。このコマンドには、RSA キーをプライベート NVRAM に保存する命令が含まれています。**copy system:running-config rcp:** または copy **system:running-config tftp:** コマンドを使用すると、RSA キーは設定に保存されません 。

認証局のモニタリングと維持

証明書失効リストの要求

証明書失効リスト（CRL）の要求は、認証局（CA）が登録局（RA）をサポートしていないときのみ実施可能です。次のタスクは、CA が RA をサポートしていないときのみ適用されます。

デバイスがピアから証明書を受信すると、デバイスは CA から CRL をダウンロードします。次に、デバイスは CRL をチェックして、ピアから送信された証明書が無効になっていないことを確認します（証明書が CRL に表示されている場合、デバイスは証明書を受け付けず、ピアを認証しません）。

クエリ モードがオフの場合は、CRL の期限が切れるまで CRL を後続の証明書に再使用することができます。該当する CRL の期限が切れた後でデバイスがピアの証明書を受信すると、デバイスは新しい CRL をダウンロードします。

デバイスにある CRL は有効期限内だがそのコンテンツが古くなっていることが疑われる場合は、古い CRL と置き換える最新の CRL をすぐにダウンロードするよう要求することができます。

-

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto pki crl request** *name*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto pki crl request** *name* 例： `Device(config)# crypto pki crl request myca`	CA から新しい証明書失効リスト（CRL）をただちに取得するよう要求します。
ステップ 4	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

証明書失効リストの照会

証明書失効リスト（CRL）の照会は、信頼できるルートでデバイスを設定するときのみ実行可能です。デバイスが別のドメイン（異なる CA）のピアから証明書を受信した場合、デバイスの CA からダウンロードした CRL には、そのピアの証明書情報が含まれません。そのため、LDAP URL で設定したルートにより発行された CRL をチェックして、ピアの証明書が失効していないことを確認する必要があります。

デバイス再起動時にルート証明書の CRL を照会したい場合は、**crl query** コマンドを入力する必要があります。

LDAP URL で設定されたルートにより発行された CRL を照会するには、次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *name*
4. **crl query ldap** ://*url* : [*port*]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto pki trustpoint** *name* 例： `Device(ca-trustpoint)# crypto pki trustpoint mytp`	デバイスで使用するトラストポイントを宣言し、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 4	**crl query ldap** ://*url* : [*port*] 例： `Device(ca-trustpoint)# crl query ldap://url:[port]`	CRL を照会し、ピアの証明書が失効していないことを確認します。
ステップ 5	**end** 例： `Device(ca-trustpoint)# end`	CA トラストポイント コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

デバイスからの RSA キーの削除

特定の状況下では、デバイスから RSA キーを削除することが必要になる場合があります。たとえば、何らかの原因で RSA キーペアの信用性が失われ、使用しなくなった場合、そのキーペアを削除します。

]

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto key zeroize rsa** [*key-pair-label*]
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto key zeroize rsa** [*key-pair-label*] 例： `Device(config)# crypto key zeroize rsa`	すべての Rivest、Shamir、Adelman（RSA）キーをデバイスから削除します。
ステップ 4	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

次のタスク

デバイスから RSA キーを削除した後、次の 2 つの追加作業も完了する必要があります。

- CA の管理者に、CA でデバイスの証明書を無効にするよう依頼します。このとき、**crypto pki enroll** コマンドを使用して初めてデバイスの証明書を取得した際に作成したチャレンジパスワードを、提供する必要があります。
- デバイスの設定からデバイスの証明書を手動で削除します。

ピアの公開キーの削除

特定の状況下では、デバイスの設定からピア デバイスの RSA 公開キーを削除することが必要になる場合があります。たとえば、ピアの公開キーの整合性が信頼できなくなった場合、キーを削除する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto key pubkey-chain rsa**
4. **no named key** *key-name* [**encryption** | **signature**]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto key pubkey-chain rsa** 例： `Device(config)# crypto key pubkey-chain rsa`	他のデバイスのRSA公開キーを手動で指定できるようにするため、公開キーチェーン コンフィギュレーション モードを開始します。
ステップ 4	**no named key** *key-name* [**encryption** | **signature**] 例： `Device(config-pubkey-c)# no named-key otherpeer.example.com`	リモートピアのRSA公開キーを削除して、公開キーコンフィギュレーション モードを開始します。
ステップ 5	**end** 例： `Device(config-pubkey)# end`	公開キー コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

設定からの証明書の削除

必要に応じて、デバイスに保存された証明書を削除することができます。デバイスには、自身の証明書、CA の証明書、任意の RA 証明書が保存されています。

CA の証明書を削除するには、CA のアイデンティティ全体を削除する必要があります。これにより、CA に関連付けられたすべての証明書（ルータの証明書、CA 証明書、任意の RA 証明書）も削除されます。

手順の概要

1. **enable**
2. **show crypto pki certificates**
3. **configure terminal**
4. **crypto pki certificate chain** *name*
5. **no certificate** *certificate-serial-number*
6. **exit**
7. **no crypto pki import** *name* **certificate**
8. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show crypto pki certificates** 例： `Device# show crypto pki certificates`	デバイスの証明書、認証局（CA）証明書、および任意の登録局（RA）証明書に関する情報を表示します。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**crypto pki certificate chain** *name* 例： `Device(config)# crypto pki certificate chain myca`	証明書チェーン コンフィギュレーション モードを開始します。
ステップ 5	**no certificate** *certificate-serial-number* 例： `Device(config-cert-chain)# no certificate 0123456789ABCDEF0123456789ABCDEF`	証明書を削除します。
ステップ 6	**exit** 例： `Device(config-cert-chain)# exit`	証明書チェーン コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 7	**no crypto pki import** *name* **certificate** 例： `Device(config)# no crypto pki import MS certificate`	証明書を手動で削除します。
ステップ 8	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

キーと証明書の表示

キーと証明書を表示するには次の作業を実行します。

手順の概要

1. **enable**
2. **show crypto key mypubkey rsa** [*keyname*]

3. **show crypto key pubkey-chain rsa**
4. **show crypto key pubkey-chain rsa** [**name** *key-name* | **address** *key-address*]
5. **show crypto pki certificates**
6. **show crypto pki trustpoints**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show crypto key mypubkey rsa** [*keyname*] 例： `Device# show crypto key mypubkey rsa [keyname]`	デバイスで設定されている RSA 公開キーを表示します。
ステップ 3	**show crypto key pubkey-chain rsa** 例： `Device# show crypto key pubkey-chain rsa`	デバイスに保存されている、ピアの RSA 公開キーを表示します。
ステップ 4	**show crypto key pubkey-chain rsa** [**name** *key-name* | **address** *key-address*] 例： `Device# show crypto key pubkey-chain rsa address 209.165.202.129`	特定のキーのアドレスを表示します。
ステップ 5	**show crypto pki certificates** 例： `Device# show crypto pki certificates`	デバイスの証明書、認証局（CA）証明書、および任意の登録局（RA）証明書に関する情報を表示します。
ステップ 6	**show crypto pki trustpoints** 例： `Device# show crypto pki certificates`	デバイスで設定されているトラストポイントを表示します。

第 62 章

アクセス コントロール リストの概要

アクセスリストは、パケットをデバイスのインターフェイスで転送するかブロックするかを制御して、ネットワーク トラフィックをフィルタリングします。デバイスは各パケットを調べ、アクセスリスト内で指定されている基準に基づいて、そのパケットの転送またはドロップを決定します。

アクセスリストで指定できる条件には、トラフィックの送信元アドレス、トラフィックの宛先アドレス、または上位層のプロトコルなどが含まれます。

（注） これらのリストは認証を必要としないため、一部のユーザは基本的なアクセス リストを回避できる可能性があります。

• アクセス コントロール リストについて （1399 ページ）

アクセス コントロール リストについて

アクセス リストの定義

アクセスリストは、少なくとも 1 つの **permit** ステートメント、および任意の 1 つまたは複数の **deny** ステートメントで構成される順次リストです。IP アドレス リストの場合、ステートメントは IP アドレス、上位層の IP プロトコルなどの IP パケットのフィールドに適用できます。アクセスリストは名前または番号で識別および参照されます。アクセスリストはパケット フィルタとして動作し、アクセスリストに定義されている条件に基づいてパケットのフィルタ処理を行います。

アクセスリストを設定しても、アクセスリストがインターフェイスまたは仮想端末回線（VTY）に適用されるか、アクセスリストを受け入れるコマンドで参照されるまでは、有効になりません。複数のコマンドから同じアクセス リストを参照できます。

次に、branchoffices という名前の IP アクセス リストを作成するための設定例を示します。ACL は着信パケットのシリアルインターフェイス 0 に適用されます。このインターフェイスにアクセスできるのは、個々の各送信元アドレスとマスク ペアで指定されているネットワーク上の送

信元のみです。ネットワーク 172.20.7.0 上の送信元から発信されるパケットの宛先に、制限はありません。ネットワーク 172.29.2.0 上の送信元から発信されるパケットの宛先は、172.25.5.4 にする必要があります。

```
ip access-list extended branchoffices
 10 permit 172.20.7.0 0.0.0.3 any
 20 permit 172.29.2.0 0.0.0.255 host 172.25.5.4
!
interface serial 0
 ip access-group branchoffices in
```

アクセス コントロール リストの機能

アクセス リストを設定する理由は多数あります。たとえば、ルーティング アップデートのコンテンツの制限や、トラフィック フローの制御などです。アクセス リストを設定する最も重要な理由の 1 つは、このモジュールの要であるネットワークにセキュリティを提供することです。

アクセスリストを使用することで、ネットワークにアクセスするための基本的なセキュリティレベルが実現します。デバイスでアクセスリストを設定しないと、デバイスを通過するすべてのパケットに、ネットワーク全体へのアクセスが許可されます。

アクセスリストでは、あるホストにはネットワークの一部へのアクセスを許可する一方、別のホストにはそれと同じ領域へのアクセスを禁止することが可能です。次の図では、ホスト A にはヒューマン リソース ネットワークへのアクセスが許可されていますが、ホスト B にはヒューマン リソース ネットワークへのアクセスが禁止されています。

また、アクセス リストを使用して、デバイス インターフェイスで転送またはブロックするトラフィックの種類を定義することもできます。たとえば、電子メール トラフィックのルーティングを許可し、同時にすべての Telnet トラフィックをブロックすることができます。

IP アクセス リストの目的

アクセス リストは、パケット フィルタリングを実行して、ネットワークを介して移動するパケットとその場所を制御します。この処理は、ネットワーク トラフィックを制限し、ユーザやデバイスによるネットワークへのアクセスを制限するのに役立ちます。アクセス リストの用途は多様なので、多くのコマンドシンタックスでアクセス リストが参照されます。アクセス リストを使用して、次のようなことを実行できます。

- インターフェイスでの着信パケットのフィルタリング
- インターフェイスでの発信パケットのフィルタリング
- ルーティング アップデートの内容の制限
- アドレスまたはプロトコルに基づくデバッグ出力の制限
- 仮想端末回線アクセスの制御

- 輻輳回避、輻輳管理、プライオリティおよびカスタム キューイングなどの高度な機能に使用されるトラフィックの特定または分類
- ダイヤルオンデマンド ルーティング（DDR）呼び出しのトリガー

ACL を設定する理由

アクセス リストを設定する理由は多数あります。たとえば、アクセス リストを使用して、スイッチング アップデートのコンテンツを制限したり、トラフィック フローを制御したりできます。アクセス リストを設定する最も重要な理由の 1 つは、ネットワークに対するアクセスを制御することで、ネットワークに基本レベルのセキュリティを提供することです。デバイスでアクセスリストを設定しない場合、デバイスを通過するすべてのパケットは、ネットワークのすべての部分で許可される可能性があります。

アクセスリストで、ネットワークの一部に対してアクセスを許可するホストと、同じ領域に対してアクセスを禁止するホストを設定できます。たとえば、適切なアクセス リストをデバイスのインターフェイスに適用することで、ホスト A にはヒューマン リソース ネットワークへのアクセスが許可され、ホスト B にはヒューマン リソース ネットワークへのアクセスが禁止されます。

ネットワークの 2 つの部分の間に配置されたデバイスにアクセス リストを使用して、内部ネットワークの特定の部分で発着信するトラフィックを制御できます。

アクセスリストのセキュリティ上の利点を実現するために、少なくとも境界デバイスでアクセス リストを設定する必要があります。境界デバイスとは、ネットワークのエッジにあるデバイスです。このようなアクセスリストは、外部ネットワークから、または内部ネットワークのあまり制御されていない領域から、内部ネットワークの機密性が高い領域に対する基本的なバッファとして機能します。このような境界デバイスでは、デバイスのインターフェイスに設定されている各ネットワーク プロトコルに合わせてアクセス リストを設定する必要があります。着信トラフィック、発信トラフィック、またはその両方がインターフェイスでフィルタされるように、アクセス リストを設定できます。

アクセス リストは個々のプロトコル ベースで定義されます。つまり、各プロトコルのトラフィック フローを制御する場合、インターフェイスでイネーブルにするプロトコルごとにアクセス リストを定義する必要があります。

アクセス リストのソフトウェア処理

アクセス リストがインターフェイス、vty に適用されるとき、あるいはコマンドで参照されるときの処理方法を説明した一般的な手順を次に示します。この手順は、アクセス リスト エントリが 13 以下のアクセス リストに適用されます。

- ソフトウェアが IP パケットや各パケットのテスト部分を受け取ります。これらは、アクセスリストの条件に一度に 1 つずつ（**permit** または **deny** ステートメント）照らし合わせてフィルタリングされます。たとえば、ソフトウェアは、**permit** あるいは **deny** ステートメントの送信元アドレスおよび宛先アドレスに照らし合わせてパケットの送信元アドレスおよび宛先アドレスをテストします。

- パケットがアクセス リストのステートメントに一致しないと、そのパケットはリスト内の次のステートメントに対してテストされます。
- パケットとアクセス リスト ステートメントが一致すると、リスト内の残りのステートメントはスキップされ、パケットは一致したステートメントに指定されたとおりに許可または拒否されます。パケットが許可されるか拒否されるかは、パケットが一致する最初のエントリによって決まります。つまり、一致すると、それ以降のエントリは考慮されません。
- アクセス リストがパケットを拒否する場合、ソフトウェアはパケットを廃棄し、インターネット制御メッセージ プロトコル（ICMP）ホスト到達不能メッセージを返します。
- いずれの条件とも一致しなかった場合、パケットは廃棄されます。これは、各アクセスリストが暗黙の **deny** ステートメントで終了するためです。言い換えると、パケットが各ステートメントに対してテストされたときまでに許可されないと、このパケットは拒否されます。

13 を超えるエントリが含まれるアクセス リストは、trie ベースのルックアップ アルゴリズムを使用して処理されます。このプロセスは自動的に行われます。設定する必要はありません。

アクセス リストのルール

アクセス コントロール リスト（ACL）には、次のルールが適用されます。

- 1 つのインターフェイス、1 つのプロトコル、1 つの方向につき、許可されるアクセス リストは 1 つだけです。
- アクセスリストには少なくとも 1 つの **permit** ステートメントが含まれる必要があります。そうしないと、ネットワークに入るすべてのパケットが拒否されます。
- アクセスリスト条件または一致基準の構成順序は重要です。パケットを転送するかブロックするかを決定するときに、シスコソフトウェアは、それぞれの条件ステートメントに対してステートメントの作成順にパケットをテストします。一致が見つかると、条件ステートメントはそれ以上チェックされません。同じ **permit** ステートメントまたは **deny** ステートメントでも、順序が異なる場合、ある状況では通過し、別の状況では拒否されるパケットが生じる可能性があります。
- アクセス リストを名前によって参照したときに、そのアクセス リストが存在しない場合は、すべてのパケットが通過します。インターフェイスまたはコマンドに空のアクセス リストを適用すると、ネットワークに対するすべてのトラフィックが許可されます。
- 標準のアクセス リストと拡張のアクセス リストの名前は同じにできません。
- パケットがアウトバウンド インターフェイスに送信される前に、インバウンド アクセス リストがパケットを処理します。ネットワークへのパケット アクセスを拒否するフィルタ条件があるインバウンド アクセス リストは、ルート ルックアップ時のオーバーヘッドを削減します。構成されたフィルタ基準に基づいてネットワークへのアクセスを許可されたパケットはルーティング処理されます。インバウンド アクセス リストの場合、**permit** ス

テートメントを構成するとパケットは受信後に処理され、**deny** ステートメントを構成するとパケットは破棄されます。

- アウトバウンドアクセスリストの場合、パケットの処理後にデバイスから送信されます。着信パケットはアウトバウンドインターフェイスにルーティングされてから、アウトバウンドアクセス リストで処理されます。アウトバウンドアクセス リストの場合、**permit** ステートメントを構成するとパケットは出力バッファに送信され、**deny** ステートメントを構成するとパケットは破棄されます。
- アクセス リストで、デバイスに到達するトラフィック、またはデバイス経由で送信されるトラフィックは制御できますが、デバイスが送信元のトラフィックは制御できません。

IP アクセス リストを作成する際に役立つヒント

意図しない結果を回避し、より効率的なアクセス リストを作成するために役立つヒントを紹介します。

- アクセス リストを作成してから、インターフェイス（または別の対象）に適用します。その理由は、存在しないアクセス リストをインターフェイスに適用してから、アクセス リストを設定すると、最初のステートメントが有効になり、それに続く暗黙的な **deny** ステートメントによってアクセスに緊急の問題が発生するおそれがあるためです。
- アクセス リストを設定してから適用するもう 1 つの理由は、空のアクセス リストが適用されたインターフェイスはすべてのトラフィックを許可するためです。
- すべてのアクセス リストには、少なくとも 1 つの **permit** ステートメントが必要です。permit がないと、すべてのパケットは拒否され、トラフィックはまったく通過しません。
- 最初に（**permit** または **deny** ステートメントに対する）一致が見つかった後は条件のテストが終了するため、パケットが一致する可能性の高いステートメントをアクセス リストの先頭に配置すると処理にかかる時間とリソースが削減されます。最も頻繁に発生する条件を発生頻度の低い条件より前に配置します。
- ネットワークまたはサブネットのより具体的な参照が、より全般的な参照よりも前に出現するように、アクセス リストを構成します。
- まだ拒否されていないその他のパケットすべてを許可する場合、ステートメント **permit any any** を使用します。ステートメント **permit any any** を使用すると、実質的に、アクセス リストの末尾にある暗黙的な deny ステートメントでその他すべてのパケットが拒否されることを防ぎます。最初のアクセス リスト エントリは **permit any any** にしないでください。すべてのトラフィックが通過し、以降のテストに到達するパケットがなくなります。**permit any any** を指定すると、まだ拒否されていないすべてのトラフィックが通過します。
- すべてのアクセス リストは暗黙的な **deny** ステートメントで終了しますが、明示的な **deny** ステートメント（たとえば **deny ip any any**）の使用を推奨します。ほとんどのプラットフォームでは、**show access-list** コマンドを発行して拒否されるパケット数を表示し、アクセス リストが許可していないパケットに関する詳細情報を調査できます。明示的な **deny**

ステートメントで拒否されたパケットのみがカウントされます。これは、明示的な **deny** ステートメントによって、より詳細なデータが生成されるためです。

- アクセス リストの作成中、または作成後に、エントリを削除する場合があります。
  - 番号付きアクセスリストからはエントリを削除できません。削除しようとすると、アクセス リスト全体が削除されます。エントリを削除する必要がある場合、アクセス リスト全体を削除してから最初から作り直す必要があります。
  - 名前付きアクセス リストからはエントリを削除できます。**no permit** または **no deny** コマンドを使用すると、適切なエントリが削除されます。
- 個々のステートメントの用途をひと目で確認および理解しやすくするために、**remark** コマンドを使用して、ステートメントの前後に役立つ注記を書き込むことができます。
- 特定のホストまたはネットワークに対するアクセスを拒否し、そのネットワークまたはホストの誰かがアクセスしようとしたかどうかを検出する場合、対応する **deny** ステートメントを指定した **log** キーワードを含めます。それによって、その送信元からの拒否されたパケットがログに記録されます。
- このヒントは、アクセスリストの配置に適用されます。リソースを保存しようとすると、インバウンドアクセスリストでは常にフィルタ条件を適用した後に、ルーティングテーブルの検索を行います。アウトバウンドアクセスリストではフィルタ条件を適用する前に、ルーティング テーブルの検索を行います。

アクセスを制御するためにフィルタできる IP パケット フィールド

拡張アクセスリストを使用すると、IP パケットに含まれる次の任意のフィールドについてフィルタできます。送信元アドレスおよび宛先アドレスは、アクセスリストの基礎として最もよく指定される 2 つのフィールドです。

- 送信元アドレス - 特定のネットワーキングデバイスまたはホストから送信されるパケットを制御するために、送信元アドレスを指定します。
- 宛先アドレス - 特定のネットワーキングデバイスまたはホストに対して送信されるパケットを制御するために、宛先アドレスを指定します。
- プロトコル - キーワード **eigrp**、**gre**、**icmp**、**igmp**、**ip**、**ipinip**、**nos**、**ospf**、**tcp**、または **udp** で示される IP プロトコル、あるいは 0 ～ 255 の範囲の整数（インターネット プロトコルを示す）で示される IP プロトコルを指定します。トランスポート層プロトコル（**icmp**、**igmp**、**tcp**、または **udp**）を指定すると、コマンドは固有の構文になります。
  - ポートおよび非隣接ポート - ポート名またはポート番号で TCP または UDP ポートを指定します。ポート番号に非隣接ポート番号は指定できません。ポート番号は、Telnet トラフィックや HTTP トラフィックなどをフィルタする際に有効です。
  - TCP フラグ - TCP パケットに設定された任意のフラグまたはすべてのフラグにパケットが一致することを指定します。特定のフラグについてフィルタすることで、不正な同期パケットを回避できます。

- IP オプション - IP オプションを指定します。IP オプションに基づいてフィルタする理由の 1 つは、IP オプションを含む偽造パケットでルータが飽和状態にならないようにするためです。

送信元アドレスと宛先アドレス

IP パケットの送信元アドレスと宛先アドレスのフィールドは、アクセス リストの基礎となる典型的な2つのフィールドです。送信元アドレスを指定して、特定のネットワーキング デバイスまたはホストから送信されるパケットを制御します。宛先アドレスを指定して、特定のネットワーキング デバイスまたはホストに送信されるパケットを制御します。

アクセス リストのアドレスに対するワイルドカード マスク

アドレス フィルタリングでは、アクセス リスト エントリ内のアドレス ビットとアクセス リストに送信されるパケットを比較するときに、対応する IP アドレスを確認するか無視するかをソフトウェアに示すために、ワイルドカード マスクを使用します。注意してワイルドカード マスクを設定することで、許可または拒否テストのために 1 つまたは複数の IP アドレスを指定できます。

IP アドレス ビット用のワイルドカード マスクでは、数値 1 と数値 0 を使用して、対応する IP アドレス ビットをどのように扱うかを指定します。1 と 0 は、サブネット（ネットワーク）マスクで意味する内容が対照的なため、ワイルドカード マスクは逆マスクとも呼ばれます。

- ワイルドカード マスク ビット 0 は、対応するビット値を確認することを示します。ビット値は一致する必要があります。
- ワイルドカード マスク ビット 1 は、対応するビット値を無視することを示します。ビット値が一致する必要はありません。

アクセス リスト ステートメントの送信元アドレスまたは宛先アドレスでワイルドカード マスクを指定しない場合、0.0.0.0（すべての値が一致する必要があることを示します）という暗黙的なワイルドカード マスクが想定されます。

サブネット マスクでは、ネットワークとサブネットを示す隣接ビットをマスクにする必要がありますが、それとは異なり、ワイルドカード マスクではマスクに非隣接ビットを使用できます。

次の表に、アクセス リストの IP アドレスおよびマスクと、それに一致すると見なされる対応するアドレスの例を示します。

表 *131 : IP* アドレス、ワイルドカード マスク、および一致する結果の例

アドレス	ワイルドカード マスク	一致する結果
0.0.0.0	255.255.255.255	すべてのアドレスはアクセス リスト条件に一致します
172.18.0.0/16	0.0.255.255	ネットワーク 172.18.0.0

アドレス	ワイルドカード マスク	一致する結果
172.18.5.2/16	0.0.0.0	ホスト 172.18.5.2 のみが一致します
172.18.8.0	0.0.0.7	サブネット 172.18.8.0/29 のみが一致します
172.18.8.8	0.0.0.7	サブネット 172.18.8.8/29 のみが一致します
172.18.8.15	0.0.0.3	サブネット 172.18.8.15/30 のみが一致します
10.1.2.0	0.0.254.255（マスクの非隣接ビット）	10.1.2.0 ～ 10.1.254.0 に含まれる偶数のネットワークに一致します

アクセス リストのシーケンス番号

IP アクセス リスト エントリにシーケンス番号を適用する機能によって、アクセス リストの変更が簡易になります。IP アクセス リスト エントリ シーケンス番号機能の前には、アクセス リスト内のエントリの位置を指定する方法はありませんでした。以前は、既存のリストの途中にエントリを挿入する場合、目的の位置の後にあるすべてのエントリを削除してから、新しいエントリを追加し、削除したすべてのエントリを再入力する必要がありました。これは手間がかかり、エラーが起こりやすい方法です。

この新しい機能を使用すると、アクセス リスト エントリにシーケンス番号を追加し、順序を変更することができます。新しいエントリを追加する場合、アクセスリストの目的の位置に挿入されるようにシーケンス番号を指定します。必要に応じて、アクセスリストの現在のエントリを並べ替えて、新しいエントリを挿入できる場所を作成できます。

ACL でサポートされるタイプ

スイッチは、IP ACL とイーサネット（MAC）ACL をサポートします。

- IP ACL は、TCP、ユーザ データグラム プロトコル（UDP）、インターネット グループ管理プロトコル（IGMP）、およびインターネット制御メッセージプロトコル（ICMP）などの IPv4 トラフィックをフィルタリングします。
- イーサネット ACL は非 IP トラフィックをフィルタリングします。

このスイッチは、Quality of Service（QoS）分類 ACL もサポートしています。

サポートされる ACL

スイッチでは、トラフィックをフィルタリングするために、次に示す 3 種類の ACL がサポートされています。

- ポート ACL は、レイヤ 2 インターフェイスに入るトラフィックをアクセス コントロールします。IPv4 と MAC どちらのアクセス リスト タイプのどの方向に対してでも、レイヤ 2 インターフェイスにポート ACL を適応できます。

- ルータ ACL は、VLAN 間でルーティングされたトラフィックのアクセスを制御し、レイヤ 3 インターフェイスで特定の方向（着信または発信）に適用されます。

ACL 優先順位

VLAN マップ、ポート ACL、およびルータ ACL が同じスイッチに設定されている場合、入力トラフィックの場合のフィルタの優先順位は上からポート ACL、VLAN マップ、およびルータ ACL です。出力トラフィックの場合、フィルタの優先順位は、ルータ ACL、VLAN マップ、ポート ACL です。

次の例で、簡単な使用例を説明します。

- 入力ポート ACL と VLAN マップが両方とも適用されている場合に、ポート ACL が適用されたポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。その他のパケットは、VLAN マップによってフィルタリングされます。
- スイッチ仮想インターフェイス（SVI）に入力ルータ ACL および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。他のポートで受信した着信のルーティング IP パケットには、ルータ ACL のフィルタが適用されます。他のパケットはフィルタリングされません。
- SVI に出力ルータ ACL および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。発信するルーティング IP パケットには、ルータ ACL のフィルタが適用されます。他のパケットはフィルタリングされません。
- SVI に VLAN マップ、入力ルータ ACL、および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL だけによってフィルタリングされます。他のポートで受信した着信のルーティング IP パケットには、VLAN マップおよびルータ ACL のフィルタが適用されます。他のパケットには、VLAN マップのフィルタだけが適用されます。
- SVI に VLAN マップ、出力ルータ ACL、および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL だけによってフィルタリングされます。発信するルーティング IP パケットには、VLAN マップおよびルータ ACL のフィルタが適用されます。他のパケットには、VLAN マップのフィルタだけが適用されます。

ポート ACL

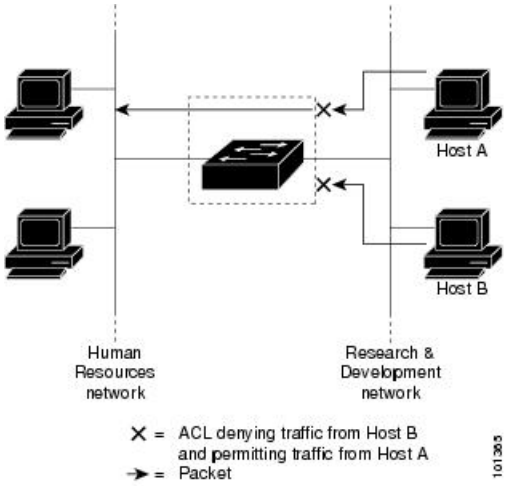
ポート ACL は、スイッチのレイヤ 2 インターフェイスに適用される ACL です。ポート ACL を使用できるのは、物理インターフェイスだけです。EtherChannel インターフェイスでは使用できません。ポート ACL は、インバウンド方向のインターフェイスに適用できます。次のアクセス リストがサポートされています。

- 送信元アドレスを使用する IP アクセス リスト

- 送信元および宛先のアドレスと任意でプロトコル タイプ情報を使用できる拡張 IP アクセス リスト
- 送信元および宛先の MAC アドレスと任意でプロトコル タイプ情報を使用できる MAC 拡張アクセス リスト

スイッチは、インターフェイス上の ACL を調べ、パケットが ACL 内のエントリとどのように一致するかに基づいてパケットの転送を許可または拒否します。このように、ACL がネットワークまたはネットワークの部分へのアクセスを制御します。

図 *93 : ACL* によるネットワーク内のトラフィックの制御



次に、すべてのワークステーションが同じ VLAN にある場合にポート ACL を使用してネットワークへのアクセスを制御する例を示します。レイヤ 2 の着信方向に適用された ACL は、ホスト A がヒューマン リソース ネットワークにアクセスすることを許可しますが、ホスト B が同一のネットワークにアクセスすることは拒否します。ポート ACL は、着信方向のレイヤ 2 インターフェイスだけに適用できます。

ポート ACL をトランク ポートに適用すると、ACL はそのトランク ポート上のすべての VLAN でトラフィックをフィルタリングします。ポート ACL を音声 VLAN ポートに適用すると、ACL はデータ VLAN と音声 VLAN の両方でトラフィックをフィルタリングします。

ポート ACL では、IP アクセス リストを使用して IP トラフィックをフィルタリングでき、MAC アドレスを使用して非 IP トラフィックをフィルタリングできます。同じレイヤ 2 インターフェイス上で IP トラフィックと非 IP トラフィックの両方をフィルタリングするには、そのインターフェイスに IP アクセス リストと MAC アクセス リストの両方を適用します。

（注） レイヤ 2 インターフェイスに適用できるのは、IP アクセス リスト 1 つと MAC アクセス リスト 1 つだけです。すでに IP アクセス リストまたは MAC アクセス リストが設定されているレイヤ 2 インターフェイスに、新しい IP アクセス リストまたは MAC アクセス リストを適用すると、前に設定した ACL が新しい ACL に置き換わります。

ルータ ACL

VLAN へのレイヤ 3 インターフェイスであるスイッチ仮想インターフェイス（SVI）、物理層 3 インターフェイス、およびレイヤ 3 EtherChannel インターフェイスに、ルータ ACL を適用できます。ルータ ACL はインターフェイスの特定の方向（着信または発信）に対して適用されます。1 つのインターフェイスの方向ごとに、ルータ ACL を 1 つ適用できます。

スイッチは、IPv4 トラフィックの次のアクセス リストをサポートしています。

- 標準 IP アクセス リストでは、照合操作に送信元アドレスを使用します。
- 拡張 IP アクセス リストは、送信元アドレス、宛先アドレス、およびオプションのプロトコル タイプ情報を使用して一致処理を行います。

ポート ACL の場合と同様、スイッチはインターフェイスに設定されている機能に関連付けられている ACL が照合されます。パケットがスイッチのインターフェイスに着信すると、そのインターフェイスに設定されているすべての着信機能に対応する ACL が照合されます。パケットがルーティングされてからネクストホップに転送されるまでの間に、出力インターフェイスに設定された発信機能に対応するすべての ACL が照合されます。

ACL は ACL 内のエントリとパケットの一致結果に応じて、パケット転送を許可するか、拒否するかを決めます。ACL を使用すると、ネットワーク全体またはネットワークの一部に対するアクセス コントロールが行えます。

アクセス コントロール エントリ

ACL には、アクセス コントロール エントリ（ACE）の順序付けられたリストが含まれています。各 ACE には、*permit* または *deny* と、パケットが ACE と一致するために満たす必要のある一連の条件を指定します。*permit* または *deny* の意味は、ACL が使用されるコンテキストによって変わります。

ACE およびフラグメント化されたトラフィックとフラグメント化されていないトラフィック

IP パケットは、ネットワークを通過するときにフラグメント化されることがあります。その場合、TCP または UDP ポート番号や ICMP タイプおよびコードなどのレイヤ 4 情報は、パケットの最初の部分があるフラグメントだけに含まれます。他のフラグメントには、この情報はありません。

アクセス コントロール エントリ（ACE）には、レイヤ 4 情報をチェックしないため、すべてのパケット フラグメントに適用されるものがあります。レイヤ 4 情報を調べる ACE は、フラグメント化された IP パケットのほとんどのフラグメントに標準的な方法では適用できません。フラグメントにレイヤ 4 情報が含まれておらず、ACE が一部のレイヤ 4 情報をチェックする場合、一致ルールは次のように変更されます。

- フラグメント内のレイヤ 3 情報（TCP や UDP などのプロトコル タイプを含む）をチェックする許可 ACE は、含まれていないレイヤ 4 情報の種類にかかわらず、フラグメントと一致すると見なされます。

（注） L4 Ops をともなう ACE の TCP では、フラグメント化パケットは RFC 1858 ごとにドロップします。

- レイヤ 4 情報をチェックする拒否 ACE は、フラグメントにレイヤ 4 情報が含まれていない限り、フラグメントと一致しません。

ACE およびフラグメント化されたトラフィックとフラグメント化されていないトラフィックの例

次のコマンドで構成され、フラグメント化された 3 つのパケットに適用されるアクセス リスト 102 を例に取って説明します。

```
スイッチ(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
スイッチ(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
スイッチ(config)# access-list 102 permit tcp any host 10.1.1.2
スイッチ(config)# access-list 102 deny tcp any any
```

（注） 最初の 2 つの ACE には宛先アドレスの後に *eq* キーワードがありますが、これは既知の TCP 宛先ポート番号がそれぞれシンプル メール転送プロトコル（SMTP）および Telnet と一致するかどうかをチェックすることを意味します。

- パケット A は、ホスト 10.2.2.2 のポート 65000 からホスト 10.1.1.1 の SMTP ポートに送信される TCP パケットです。このパケットがフラグメント化された場合、レイヤ 4 情報がすべて揃っているため、完全なパケットである場合と同じように最初のフラグメントが最初の ACE（permit）と一致します。残りのフラグメントも最初の ACE と一致します。これは、それらのフラグメントに SMTP ポート情報が含まれていなくても、最初の ACE が適用されたときにレイヤ 3 情報だけをチェックするからです。この例の情報は、パケットが TCP であることと、宛先が 10.1.1.1 であることです。
- パケット B は、ホスト 10.2.2.2 のポート 65001 からホスト 10.1.1.2 の Telnet ポートに送信されます。このパケットがフラグメント化された場合、レイヤ 3 情報とレイヤ 4 情報がすべて揃っているため、最初のフラグメントが 2 つめの ACE（deny）と一致します。残りのフラグメントは、レイヤ 4 情報が含まれていないため、2 つめの ACE と一致しません。残りのフラグメントは 3 つめの ACE（permit）と一致します。

  最初のフラグメントが拒否されたため、ホスト 10.1.1.2 は完全なパケットを再構成できず、その結果、パケット B は拒否されます。ただし、以降の許可されたフラグメントがネットワークの帯域幅を使用し、ホスト 10.1.1.2 がパケットを再構成しようとするときにホストのリソースが消費されます。
- フラグメント化されたパケット C は、ホスト 10.2.2.2 のポート 65001 からホスト 10.1.1.3 のポート ftp に送信されます。このパケットがフラグメント化された場合、最初のフラグ

メントが 4 つめの ACE（deny）と一致します。ACE はレイヤ 4 情報をチェックせず、すべてのフラグメントのレイヤ 3 情報に宛先がホスト 10.1.1.3 であることが示され、前の permit ACE は異なるホストをチェックしていたため、他のフラグメントもすべて 4 つめの ACE と一致します。

第 63 章

IPv4 ACL

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

IPv4アクセスコントロールリストの設定に関する制約事項

一般的なネットワーク セキュリティ

次は、ACL によるネットワーク セキュリティの設定の制約事項です。

- 番号付き ACL で使用できるすべてのコマンドが名前付き ACL でも使用できるわけではありません。インターフェイスのパケット フィルタおよびルート フィルタ用の ACL では、名前を使用できます。

- 標準 ACL と拡張 ACL に同じ名前は使用できません。
- **AppleTalk** は、コマンドラインのヘルプ ストリングに表示されますが、**deny** および **permit** MAC アクセスリスト コンフィギュレーション モード コマンドの一致条件としてサポートされていません。
- ACL ワイルドカードは、ダウンストリーム クライアント ポリシーではサポートされていません。

IPv4 ACL ネットワーク インターフェイス

次の制限事項が、ネットワーク インターフェイスへの IPv4 ACL に適用されます。

- インターフェイスへのアクセスを制御する場合、名前付き ACL または番号付き ACL を使用できます。
- レイヤ 3 インターフェイスに ACL が適用され、スイッチ上でルーティングがイネーブルになっていない場合は、SNMP、Telnet、Web トラフィックなど、CPU で処理されるパケットだけがフィルタリングされます。
- レイヤ 2 インターフェイスに ACL を適用する場合、ルーティングをイネーブルにする必要はありません。

レイヤ 2 インターフェイスの MAC ACL

MAC ACL を作成し、それをレイヤ 2 インターフェイスに適用すると、そのインターフェイスに着信する非 IP トラフィックをフィルタリングできます。MAC ACL を適用するときには、次の注意事項に留意してください。

- 同じレイヤ 2 インターフェイスには、IP アクセス リストと MAC アクセス リストを 1 つずつしか適用できません。IP アクセス リストは IP パケットだけをフィルタリングし、MAC アクセス リストは非 IP パケットをフィルタリングします。
- 1 つのレイヤ 2 インターフェイスに適用できる MAC アドレス リストは 1 つだけです。すでに MAC ACL が設定されているレイヤ 2 インターフェイスに MAC アクセス リストを適用すると、設定済みの ACL が新しい ACL に置き換えられます。

（注） **mac access-group** インターフェイス コンフィギュレーション コマンドは、物理レイヤ 2 インターフェイスに適用される場合のみ有効です。このコマンドは、EtherChannel ポート チャネルでは使用できません。

IP アクセス リスト エントリ シーケンス番号

- この機能は、ダイナミック アクセス リスト、再帰アクセス リスト、またはファイアウォール アクセス リストをサポートしていません。

ACL によるネットワーク セキュリティに関する情報

この章では、アクセス コントロール リスト（ACL）を使用して、スイッチのネットワーク セキュリティを設定する方法について説明します。コマンドや表では、ACL をアクセス リストと呼ぶこともあります。

Cisco TrustSec および ACL

IP ベース フィーチャ セットまたは IP サービス フィーチャ セットが稼働する Catalyst 3850 スイッチでは、Cisco TrustSec Security Group Tag（SCT）Exchange Protocol（SXP）もサポートされます。この機能は、IP アドレスに対してではなく、デバイスのグループに対して ACL ポリシーを定義するセキュリティ グループ アクセス コントロール リスト（SGACL）をサポートします。SXP 制御プロトコルは、ハードウェアをアップグレードせずに SCT によってパケットをタギングするためのプロトコルで、Cisco TrustSec ドメイン エッジにあるアクセス レイヤ デバイスと、Cisco TrustSec ドメイン内の配信レイヤ デバイスの間で実行されます。Catalyst 3850 スイッチは Cisco TrustSec ネットワーク内のアクセス レイヤ スイッチとして動作します。

SXP のセクションでは、Catalyst 3850 スイッチでサポートされる機能を定義します。

ACL の概要

パケット フィルタリングは、ネットワーク トラフィックを限定し、特定のユーザまたはデバイスによるネットワークの使用を制限するうえで役立ちます。ACL はルータまたはスイッチを通過するトラフィックをフィルタリングし、特定のインターフェイスを通過するパケットを許可または拒否します。ACL は、パケットに適用される許可条件および拒否条件の順序付けられた集まりです。パケットがインターフェイスに着信すると、スイッチはパケット内のフィールドを適用される ACL と比較し、アクセス リストに指定された基準に基づいて、パケットが転送に必要な権限を持っているかどうかを確認します。アクセス リスト内の条件を 1 つずつ調べ、パケットをテストします。最初に一致した条件によって、スイッチがパケットを受け入れるか拒否するかが決定されます。スイッチは最初に一致した時点でテストを中止するので、リストに条件を指定する順序が重要です。一致する条件がない場合、スイッチはパケットを拒否します。スイッチは、制限条件がない場合はパケットを転送し、制限条件がある場合はパケットをドロップします。スイッチは、転送されるすべてのパケットに ACL を使用します。

ネットワークに基本的なセキュリティを導入する場合は、ルータまたはレイヤ 3 スイッチにアクセス リストを設定します。ACL を設定しなければ、スイッチを通過するすべてのパケットがネットワークのあらゆる部分で許可される可能性があります。ACL を使用すると、ネットワークの場所ごとにアクセス可能なホストを制御したり、ルータ インターフェイスで転送またはブロックされるトラフィックの種類を決定したりできます。たとえば、電子メール トラフィックの転送を許可し、Telnet トラフィックの転送を拒否することもできます。

アクセス コントロール エントリ

ACL には、アクセス コントロール エントリ（ACE）の順序付けられたリストが含まれています。各 ACE には、*permit* または *deny* と、パケットが ACE と一致するために満たす必要のある一連の条件を指定します。*permit* または *deny* の意味は、ACL が使用されるコンテキストによって変わります。

ACL でサポートされるタイプ

スイッチは、IP ACL とイーサネット（MAC）ACL をサポートします。

- IP ACL は、TCP、ユーザ データグラム プロトコル（UDP）、インターネット グループ管理プロトコル（IGMP）、およびインターネット制御メッセージプロトコル（ICMP）などの IPv4 トラフィックをフィルタリングします。
- イーサネット ACL は非 IP トラフィックをフィルタリングします。

このスイッチは、Quality of Service（QoS）分類 ACL もサポートしています。

サポートされる ACL

スイッチでは、トラフィックをフィルタリングするために、次に示す 3 種類の ACL がサポートされています。

- ポート ACL は、レイヤ 2 インターフェイスに入るトラフィックをアクセス コントロールします。IPv4 と MAC どちらのアクセス リスト タイプのどの方向に対してでも、レイヤ 2 インターフェイスにポート ACL を適応できます。
- ルータ ACL は、VLAN 間でルーティングされたトラフィックのアクセスを制御し、レイヤ 3 インターフェイスで特定の方向（着信または発信）に適用されます。

ACL 優先順位

VLAN マップ、ポート ACL、およびルータ ACL が同じスイッチに設定されている場合、入力トラフィックの場合のフィルタの優先順位は上からポート ACL、VLAN マップ、およびルータ ACL です。出力トラフィックの場合、フィルタの優先順位は、ルータ ACL、VLAN マップ、ポート ACL です。

次の例で、簡単な使用例を説明します。

- 入力ポート ACL と VLAN マップが両方とも適用されている場合に、ポート ACL が適用されたポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。その他のパケットは、VLAN マップによってフィルタリングされます。
- スイッチ仮想インターフェイス（SVI）に入力ルータ ACL および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。他のポートで受信した着信のルーティング IP パケットには、ルータ ACL のフィルタが適用されます。他のパケットはフィルタリングされません。

- SVI に出力ルータ ACL および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL によってフィルタリングされます。発信するルーティング IP パケットには、ルータ ACL のフィルタが適用されます。他のパケットはフィルタリングされません。
- SVI に VLAN マップ、入力ルータ ACL、および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL だけによってフィルタリングされます。他のポートで受信した着信のルーティング IP パケットには、VLAN マップおよびルータ ACL のフィルタが適用されます。他のパケットには、VLAN マップのフィルタだけが適用されます。
- SVI に VLAN マップ、出力ルータ ACL、および入力ポート ACL が設定されている場合に、ポート ACL が適用されているポートにパケットが着信すると、このパケットはポート ACL だけによってフィルタリングされます。発信するルーティング IP パケットには、VLAN マップおよびルータ ACL のフィルタが適用されます。他のパケットには、VLAN マップのフィルタだけが適用されます。

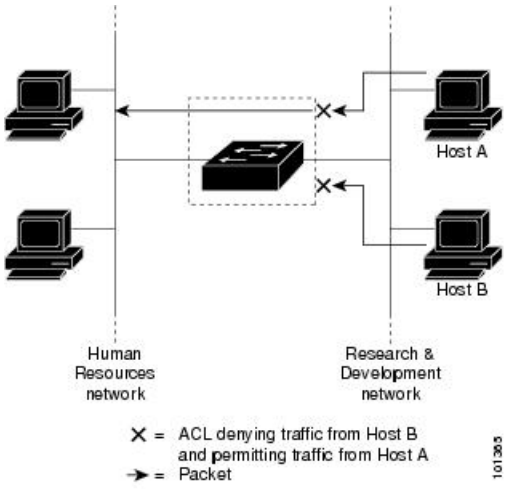
ポート ACL

ポート ACL は、スイッチのレイヤ 2 インターフェイスに適用される ACL です。ポート ACL を使用できるのは、物理インターフェイスだけです。EtherChannel インターフェイスでは使用できません。ポート ACL は、インバウンド方向のインターフェイスに適用できます。次のアクセス リストがサポートされています。

- 送信元アドレスを使用する IP アクセス リスト
- 送信元および宛先のアドレスと任意でプロトコル タイプ情報を使用できる拡張 IP アクセス リスト
- 送信元および宛先の MAC アドレスと任意でプロトコル タイプ情報を使用できる MAC 拡張アクセス リスト

スイッチは、インターフェイス上の ACL を調べ、パケットが ACL 内のエントリとどのように一致するかに基づいてパケットの転送を許可または拒否します。このように、ACL がネットワークまたはネットワークの部分へのアクセスを制御します。

図 *94 : ACL* によるネットワーク内のトラフィックの制御



次に、すべてのワークステーションが同じ VLAN にある場合にポート ACL を使用してネットワークへのアクセスを制御する例を示します。レイヤ 2 の着信方向に適用された ACL は、ホスト A がヒューマン リソース ネットワークにアクセスすることを許可しますが、ホスト B が同一のネットワークにアクセスすることは拒否します。ポート ACL は、着信方向のレイヤ 2 インターフェイスだけに適用できます。

ポート ACL をトランク ポートに適用すると、ACL はそのトランク ポート上のすべての VLAN でトラフィックをフィルタリングします。ポート ACL を音声 VLAN ポートに適用すると、ACL はデータ VLAN と音声 VLAN の両方でトラフィックをフィルタリングします。

ポート ACL では、IP アクセス リストを使用して IP トラフィックをフィルタリングでき、MAC アドレスを使用して非 IP トラフィックをフィルタリングできます。同じレイヤ 2 インターフェイス上で IP トラフィックと非 IP トラフィックの両方をフィルタリングするには、そのインターフェイスに IP アクセス リストと MAC アクセス リストの両方を適用します。

（注） レイヤ 2 インターフェイスに適用できるのは、IP アクセス リスト 1 つと MAC アクセス リスト 1 つだけです。すでに IP アクセス リストまたは MAC アクセス リストが設定されているレイヤ 2 インターフェイスに、新しい IP アクセス リストまたは MAC アクセス リストを適用すると、前に設定した ACL が新しい ACL に置き換わります。

ルータ ACL

VLAN へのレイヤ 3 インターフェイスであるスイッチ仮想インターフェイス（SVI）、物理層 3 インターフェイス、およびレイヤ 3 EtherChannel インターフェイスに、ルータ ACL を適用できます。ルータ ACL はインターフェイスの特定の方向（着信または発信）に対して適用されます。1 つのインターフェイスの方向ごとに、ルータ ACL を 1 つ適用できます。

スイッチは、IPv4 トラフィックの次のアクセス リストをサポートしています。

- 標準 IP アクセス リストでは、照合操作に送信元アドレスを使用します。
- 拡張 IP アクセス リストは、送信元アドレス、宛先アドレス、およびオプションのプロトコル タイプ情報を使用して一致処理を行います。

ポート ACL の場合と同様、スイッチはインターフェイスに設定されている機能に関連付けられている ACL が照合されます。パケットがスイッチのインターフェイスに着信すると、そのインターフェイスに設定されているすべての着信機能に対応する ACL が照合されます。パケットがルーティングされてからネクストホップに転送されるまでの間に、出力インターフェイスに設定された発信機能に対応するすべての ACL が照合されます。

ACL は ACL 内のエントリとパケットの一致結果に応じて、パケット転送を許可するか、拒否するかを決めます。ACL を使用すると、ネットワーク全体またはネットワークの一部に対するアクセス コントロールが行えます。

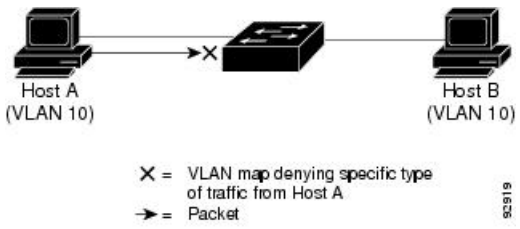
VLAN マップ

VLAN ACL または VLAN マップは、VLAN 内のネットワーク トラフィックを制御するために使用されます。スイッチまたはスイッチ スタックの VLAN 内でブリッジングされるすべてのパケットに VLAN マップを適用できます。VACL は、セキュリティ パケット フィルタリングおよび特定の物理インターフェイスへのトラフィックのリダイレクトだけを目的としたものです。VACL は方向（入力または出力）で定義されることはありません。

すべての非 IP プロトコルは、MAC VLAN マップを使用して、MAC アドレスおよび Ethertype によってアクセス コントロールされます（IP トラフィックは、MAC VACL マップではアクセス制御されません）。VLANマップはスイッチを通過するパケットにだけ適用できます。ハブ上またはこのスイッチに接続された別のスイッチ上のホスト間のトラフィックには、VLAN マップを適用させることができません。

VLANマップを使用すると、マップに指定されたアクションに基づいてパケットの転送が許可または拒否されます。

図 *95 : VLAN* マップによるトラフィックの制御



次の図に、VLAN マップを適用して、特定のトラフィック タイプを VLAN 10 のホスト A から転送できないように設定する例を示します。各 VLAN には、VLAN マップを 1 つだけ適用できます。

ACE およびフラグメント化されたトラフィックとフラグメント化されていないトラフィック

IPパケットは、ネットワークを通過するときにフラグメント化されることがあります。その場合、TCP または UDP ポート番号や ICMP タイプおよびコードなどのレイヤ 4 情報は、パケットの最初の部分があるフラグメントだけに含まれます。他のフラグメントには、この情報はありません。

アクセス コントロール エントリ（ACE）には、レイヤ 4 情報をチェックしないため、すべてのパケット フラグメントに適用されるものがあります。レイヤ 4 情報を調べる ACE は、フラグメント化されたIPパケットのほとんどのフラグメントに標準的な方法では適用できません。フラグメントにレイヤ4情報が含まれておらず、ACEが一部のレイヤ4情報をチェックする場合、一致ルールは次のように変更されます。

- フラグメント内のレイヤ 3 情報（TCP や UDP などのプロトコル タイプを含む）をチェックする許可 ACE は、含まれていないレイヤ 4 情報の種類にかかわらず、フラグメントと一致すると見なされます。

（注） L4 Ops をともなう ACE の TCP では、フラグメント化パケットは RFC 1858 ごとにドロップします。

- レイヤ 4 情報をチェックする拒否 ACE は、フラグメントにレイヤ 4 情報が含まれていない限り、フラグメントと一致しません。

ACE およびフラグメント化されたトラフィックとフラグメント化されていないトラフィックの例

次のコマンドで構成され、フラグメント化された3つのパケットに適用されるアクセス リスト 102 を例に取って説明します。

```
スイッチ(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
スイッチ(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
スイッチ(config)# access-list 102 permit tcp any host 10.1.1.2
スイッチ(config)# access-list 102 deny tcp any any
```

（注） 最初の 2 つの ACE には宛先アドレスの後に *eq* キーワードがありますが、これは既知の TCP 宛先ポート番号がそれぞれシンプル メール転送プロトコル（SMTP）および Telnet と一致するかどうかをチェックすることを意味します。

- パケット A は、ホスト 10.2.2.2 のポート 65000 からホスト 10.1.1.1 の SMTP ポートに送信される TCP パケットです。このパケットがフラグメント化された場合、レイヤ 4 情報がすべて揃っているため、完全なパケットである場合と同じように最初のフラグメントが最

初の ACE（permit）と一致します。残りのフラグメントも最初の ACE と一致します。これは、それらのフラグメントに SMTP ポート情報が含まれていなくても、最初の ACE が適用されたときにレイヤ 3 情報だけをチェックするからです。この例の情報は、パケットが TCP であることと、宛先が 10.1.1.1 であることです。

- パケット B は、ホスト 10.2.2.2 のポート 65001 からホスト 10.1.1.2 の Telnet ポートに送信されます。このパケットがフラグメント化された場合、レイヤ 3 情報とレイヤ 4 情報がすべて揃っているため、最初のフラグメントが 2 つめの ACE（deny）と一致します。残りのフラグメントは、レイヤ 4 情報が含まれていないため、2 つめの ACE と一致しません。残りのフラグメントは 3 つめの ACE（permit）と一致します。

  最初のフラグメントが拒否されたため、ホスト 10.1.1.2 は完全なパケットを再構成できず、その結果、パケット B は拒否されます。ただし、以降の許可されたフラグメントがネットワークの帯域幅を使用し、ホスト 10.1.1.2 がパケットを再構成しようとするときにホストのリソースが消費されます。

- フラグメント化されたパケット C は、ホスト 10.2.2.2 のポート 65001 からホスト 10.1.1.3 のポート ftp に送信されます。このパケットがフラグメント化された場合、最初のフラグメントが 4 つめの ACE（deny）と一致します。ACE はレイヤ 4 情報をチェックせず、すべてのフラグメントのレイヤ 3 情報に宛先がホスト 10.1.1.3 であることが示され、前の permit ACE は異なるホストをチェックしていたため、他のフラグメントもすべて 4 つめの ACE と一致します。

標準 IPv4 ACL および拡張 IPv4 ACL

ここでは、IP ACL について説明します。

ACL は、許可条件と拒否条件の順序付けられた集まりです。スイッチは、アクセス リスト内の条件を 1 つずつ調べ、パケットをテストします。最初に一致した条件によって、スイッチがパケットを受け入れるか拒否するかが決定されます。スイッチは一致する最初の条件が見つかった時点でパケットのテストを停止するため、条件の順序が重要な意味を持ちます。一致する条件がない場合、スイッチはパケットを拒否します。

このソフトウェアは、IPv4 について次の ACL（アクセス リスト）をサポートします。

- 標準 IP アクセス リストでは、照合操作に送信元アドレスを使用します。
- 拡張 IP アクセス リストでは、照合操作に送信元アドレスと宛先アドレスを使用し、任意でプロトコル タイプ情報を使用して制御のきめ細かさを高めることもできます。

IPv4 ACL スイッチでサポートされていない機能

このスイッチで IPv4 ACL を設定する手順は、他の Cisco スイッチやルータで IPv4 ACL を設定する手順と同じです。

以下の ACL 関連の機能はサポートされていません。

- 非 IP プロトコル ACL または

- IP アカウンティング
- 再帰 ACL およびダイナミック ACL はサポートされていません。
- ポート ACL および VLAN マップに関する ACL ロギング

アクセス リスト番号

ACL を識別するために使用する番号は、作成するアクセス リストのタイプを表します。

次の一覧に、アクセス リスト番号と対応するアクセス リスト タイプを挙げ、このスイッチでサポートされているかどうかを示します。このスイッチは、IPv4 標準アクセス リストおよび拡張アクセス リスト（1 ～ 199 および 1300 ～ 2699）をサポートします。

表 *132 :* アクセス リスト番号

アクセス リスト番号	タイプ	サポートあり
1 ～ 99	IP 標準アクセス リスト	あり
100 ～ 199	IP 拡張アクセス リスト	あり
200 ～ 299	プロトコル タイプコード アクセス リスト	なし
300 ～ 399	DECnet アクセス リスト	なし
400 ～ 499	XNS 標準アクセス リスト	なし
500 ～ 599	XNS 拡張アクセス リスト	なし
600 ～ 699	AppleTalk アクセス リスト	なし
700 ～ 799	48 ビット MAC アドレス アクセス リスト	なし
800 ～ 899	IPX 標準アクセス リスト	なし
900 ～ 999	IPX 拡張アクセス リスト	なし
1000 ～ 1099	IPX SAP アクセス リスト	なし
1100 ～ 1199	拡張 48 ビット MAC サマリー アドレス アクセス リスト	なし
1200 ～ 1299	IPX サマリー アドレス アクセス リスト	なし
1300 ～ 1999	IP 標準アクセス リスト（拡張範囲）	あり

アクセス リスト番号	タイプ	サポートあり
2000 ～ 2699	IP 拡張アクセス リスト（拡張範囲）	あり

番号付き標準 ACL および番号付き拡張 ACL に加え、サポートされる番号を使用して名前付き標準 ACL および名前付き拡張 ACL も作成できます。標準 IP ACL の名前は 1 ～ 99 で、拡張 IP ACL の名前は 100 ～ 199 です。番号付きリストの代わりに名前付き ACL を使用することには、エントリを個別に削除できるという利点があります。

番号付き標準 IPv4 ACL

ACL を作成するときには、ACL の末尾にデフォルトで暗黙的な deny ステートメントが追加され、ACL の終わりに到達するまで一致する条件が見つからなかったすべてのパケットに適用されることに注意してください。標準アクセス リストでは、関連付けられた IP ホスト アドレス ACL の指定からマスクを省略すると、0.0.0.0 がマスクと見なされます。

スイッチは、**host** 一致条件があるエントリと *don't care* マスク 0.0.0.0 を含む一致条件があるエントリがリストの先頭に移動し、0 以外の *don't care* マスクを含むエントリよりも前に位置するように、標準アクセス リストの順序を書き換えます。そのため、**show** コマンドの出力やコンフィギュレーション ファイルでは、ACE が必ずしも入力されたとおりの順序で配置されません。

作成した番号付き標準 IPv4 ACL を VLAN、端末回線、またはインターフェイスに適用できます。

番号付き拡張 IPv4 ACL

標準 ACL では照合に送信元アドレスだけを使用しますが、拡張 ACL では、照合操作に送信元アドレスと宛先アドレスを使用でき、任意でプロトコル タイプ情報を使用して制御のきめ細かさを高めることができます。番号付き拡張アクセス リストの ACE を作成するときには、作成した ACE がリストの末尾に追加されることに注意してください。番号付きリストでは、ACE の順序を変更したり、リスト内の特定の場所に対して ACE を追加または削除したりできません。

このスイッチは、ダイナミックまたは再帰アクセスリストをサポートしていません。また、タイプ オブ サービス（ToS）の minimize-monetary-cost ビットに基づくフィルタリングもサポートしていません。

一部のプロトコルには、特定のパラメータやキーワードも適用されます。

拡張 TCP、UDP、ICMP、IGMP、またはその他の IP ACL を定義できます。また、このスイッチはこれらの IP プロトコルをサポートします。

（注） ICMP エコー応答はフィルタリングできません。他の ICMP コードまたはタイプは、すべてフィルタリングできます。

これらの IP プロトコルがサポートされます。

- 認証ヘッダー プロトコル（**ahp**）
- カプセル化セキュリティペイロード（**esp**）
- Enhanced Interior Gateway Routing Protocol（**eigrp**）
- 総称ルーティング カプセル化（**gre**）
- インターネット制御メッセージ プロトコル（**icmp**）
- インターネット グループ管理プロトコル（**igmp**）
- すべての内部プロトコル（**ip**）
- IP-in-IP トンネリング（**ipinip**）
- KA9Q NOS 互換 IP over IP トンネリング（**nos**）
- Open Shortest Path First ルーティング（**ospf**）
- ペイロード圧縮プロトコル（**pcp**）
- プロトコル独立マルチキャスト（**pim**）
- 伝送制御プロトコル（**tcp**）
- ユーザ データグラム プロトコル（**udp**）

名前付き IPv4 ACL

IPv4 ACL を識別する手段として、番号ではなく英数字のストリング（名前）を使用できます。名前付き ACL を使用すると、ルータ上で番号付きアクセス リストの場合より多くの IPv4 アクセス リストを設定できます。アクセス リストの識別手段として名前を使用する場合のモードとコマンド構文は、番号を使用する場合とは多少異なります。ただし、IP アクセス リストを使用するすべてのコマンドを名前付きアクセス リストで使用できるわけではありません。

（注） 標準 ACL または拡張 ACL に指定する名前は、アクセス リスト番号のサポートされる範囲内の番号にすることもできます。標準 IP ACL の名前は 1 ～ 99 です。番号付きリストの代わりに名前付き ACL を使用することには、エントリを個別に削除できるという利点があります。

名前付き ACL を設定するときには、次の注意事項に留意してください。

- また、番号付き ACL も使用できます。
- 標準 ACL と拡張 ACL に同じ名前は使用できません。
- VLAN マップには、標準 ACL または拡張 ACL（名前付きまたは番号付き）を使用できます。

ACL ロギング

標準 IP アクセス リストによって許可または拒否されたパケットに関するログ メッセージが、スイッチのソフトウェアによって表示されます。つまり、ACL と一致するパケットがあった場合は、そのパケットに関するログ通知メッセージがコンソールに送信されます。コンソールに表示されるメッセージのレベルは、syslog メッセージを管理する **logging console** コマンドで管理されます。

（注） ACL ロギングは、RACL でのみサポートされます。

（注） ルーティングはハードウェアで、ロギングはソフトウェアで実行されます。したがって、**log** キーワードを含む許可（*permit*）または拒否（*deny*）ACE と一致するパケットが多数存在する場合、ソフトウェアはハードウェアの処理速度に追いつくことができないため、一部のパケットはロギングされない場合があります。

ACL を起動した最初のパケットについては、ログ メッセージがすぐに表示されますが、それ以降のパケットについては、5 分間の収集時間が経過してから表示またはロギングされます。ログ メッセージにはアクセス リスト番号、パケットの許可または拒否に関する状況、パケットの送信元 IP アドレス、および直前の 5 分間に許可または拒否された送信元からのパケット数が示されます。

（注） ロギング メッセージが多すぎて処理できない場合、または 1 秒以内に処理する必要があるロギング メッセージが複数ある場合、ロギング設備ではロギング メッセージ パケットの一部をドロップすることがあります。この動作によって、ロギング パケットが多すぎてルータがクラッシュすることを回避します。そのため、課金ツールや、アクセス リストと一致する数の正確な情報源としてロギング設備をを使用しないでください。

スマート ロギング

スイッチでスマート ロギングがイネーブルであり、スマート ロギングで設定された ACL がレイヤ 2 インターフェイス（ポート ACL）に割り当てられている場合、ACL に従って拒否または許可されたパケットの内容は NetFlow 収集装置にも送信されます。

ハードウェアおよびソフトウェアによる IP ACL の処理

ACL 処理はハードウェアで実行されます。ハードウェアで ACL の設定を保存する領域が不足すると、そのインターフェイス上のすべてのパケットがドロップします。

（注） スイッチまたはスタック メンバーのリソース不足が原因でハードウェアに ACL を設定できない場合、影響を受けるのは、スイッチに着信した該当 VLAN 内のトラフィックだけです。

ルータ ACL の場合は、次の場合にパケットが CPU に送信されることがあります。

- **log** キーワードの使用
- ICMP 到達不能メッセージを生成する。

トラフィック フローのロギングと転送の両方を行う場合、転送はハードウェアで処理されますが、ロギングはソフトウェアで処理される必要があります。ハードウェアとソフトウェアではパケット処理能力が異なるため、ロギング中であるすべてのフロー（許可フローと拒否フロー）の合計帯域幅が非常に大きい場合は、転送されたパケットの一部をロギングできません。

show ip access-lists 特権 EXEC コマンドを入力した場合、表示される一致カウントには、ハードウェアでアクセスが制御されるパケットは含まれません。スイッチド パケットおよびルーテッド パケットに関するハードウェアの ACL の基本的な統計情報を取得する場合は、 特権 EXEC コマンドを使用します。

ルータ ACL の機能は、次のとおりです。

- 標準 ACL および拡張 ACL（入力および出力）の許可アクションや拒否アクションをハードウェアで制御し、アクセス コントロールのセキュリティを強化します。
- *ip unreachables* が無効の場合、**log**を指定しないと、セキュリティ ACL の *deny* ステートメントと一致するフローがハードウェアによってドロップされます。許可ステートメントと一致するフローは、ハードウェアでスイッチングされます。
- ルータ ACL の ACE に **log** キーワードを追加すると、パケットのコピーが CPU に送信され、ロギングだけが行われます。ACEが許可ステートメントの場合も、パケットはハードウェアでスイッチングおよびルーティングされます。

VLAN マップの設定時の注意事項

VLANマップは、VLAN内でフィルタリングを制御する唯一の方法です。VLANマップには方向の指定がありません。VLANマップを使用して、特定の方向のトラフィックをフィルタリングするには、特定の送信元または宛先アドレスが指定されたACLを追加する必要があります。VLAN マップ内に該当パケット タイプ（IP または MAC）に対する match 句がある場合、デフォルトでは、マップ内のどのエントリにも一致しないパケットはドロップされます。該当パケット タイプに対する match コマンドがない場合、デフォルトでは、パケットが転送されます。

次は、VLAN マップ設定の注意事項です。

- インターフェイスでトラフィックを拒否するように設定された ACL がなく、VLAN マップが設定されていない場合、すべてのトラフィックが許可されます。

- 各 VLAN マップは一連のエントリで構成されます。VLAN マップのエントリの順序は重要です。スイッチに着信したパケットは、VLANマップの最初のエントリに対してテストされます。一致した場合は、VLANマップのその部分に指定されたアクションが実行されます。一致しなかった場合、パケットはマップ内の次のエントリに対してテストされます。
- 該当パケット タイプ（IP または MAC）に対する match 句が VLAN マップに 1 つまたは複数ある場合でも、パケットがそれらの match 句に一致しない場合、デフォルトではパケットがドロップされます。該当パケット タイプに対する match 句が VLAN マップ内にない場合、デフォルトではパケットが転送されます。
- VLAN マップのロギングはサポートされていません。
- レイヤ 2 インターフェイスに適用された IP アクセス リストまた MAC アクセス リストがスイッチにあって、ポートが属する VLAN に VLAN マップを適用する場合、ポート ACL が VLAN マップに優先します。
- ハードウェアに VLAN マップの設定を適用できない場合は、その VLAN 内のすべてのパケットがドロップします。

VLAN マップとルータ ACL

ブリッジングされたトラフィックおよびルーティングされたトラフィックの両方に対してアクセス コントロールを行うには、VLANマップを単独で使用するか、またはルータ ACL と VLAN マップを組み合わせて使用します。入力と出力両方のルーテッド VLAN インターフェイスでルータ ACL を定義したり、ブリッジングされたトラフィックのアクセスをコントロールする VLAN マップを定義したりできます。

パケット フローが ACL 内 VLAN マップの deny ステートメントと一致した場合、ルータ ACL の設定に関係なく、パケット フローは拒否されます。

（注） ルータ ACL を VLAN マップと組み合わせて使用し、ルータ ACL でのロギングを必要とするパケットが VLAN マップで拒否された場合、これらのパケットはロギングされません。

該当パケット タイプ（IP または MAC）に対する match 句が VLAN マップにある場合でも、パケットがそのタイプに一致しない場合、デフォルトではパケットがドロップされます。VLAN マップ内に match 句がなく、アクションが指定されていない場合、どの VLAN マップ エントリとも一致しないパケットは転送されます。

VLAN マップとルータ ACL の設定時の注意事項

ここに記載された注意事項は、ルータ ACL および VLAN マップを同じ VLAN 上で使用する必要がある設定に適用されます。ルータ ACL および VLAN マップを異なる VLAN に割り当てる設定には、これらの注意事項は適用されません。

ルータ ACL および VLAN マップを同じ VLAN に設定する必要がある場合は、ルータ ACL と VLAN マップの両方の設定に関し、ここで説明する注意事項に従ってください。

- VLAN インターフェイス上の各方向（入力および出力）に VLAN マップおよびルータの ACL を 1 つずつに限り設定できます。
- 可能な限り、すべてのエントリのアクションが同一で、末尾のデフォルトアクションだけが反対のタイプとなるように ACL を記述します。次のいずれかの形式を使用して、ACL を記述します。

  permit... permit... permit... deny ip any any

  または

  deny... deny... deny... permit ip any any

- ACL 内で複数のアクション（許可、拒否）を定義する場合は、それぞれのアクション タイプをまとめて、エントリ数を削減します。
- ACL 内にレイヤ 4 情報を指定しないでください。レイヤ 4 情報を追加すると、統合プロセスが複雑になります。ACL のフィルタリングが、full-flow（送信元 IP アドレス、宛先 IP アドレス、プロトコル、およびプロトコル ポート）でなく、IP アドレス（送信元および宛先）に基づいて行われる場合に、最適な統合結果が得られます。可能な限り、IP アドレスには *don't care* ビットを使用してください。

  IP ACE とレイヤ 4 情報を含む TCP/UDP/ICMP ACE が両方とも ACL 内に存在し、full-flow モードを指定する必要があるときは、レイヤ 4 ACE をリストの末尾に配置します。この結果、IP アドレスに基づくトラフィックのフィルタリングが優先されます。

VACL ロギング

VACL ロギングを設定する場合は、次の状況で拒否された IP パケットに対して Syslog メッセージが生成されます。

- 一致する最初のパケットを受信した場合
- 最後の 5 分間に一致するパケットを受信した場合
- 5 分経過する前にしきい値に達している場合

ログ メッセージはフロー単位で生成されます。フローは、同じ IP アドレスおよびレイヤ 4（UDP または TCP）ポート番号を持つパケットとして定義されます。フローで 5 分間パケットを受信しない場合、そのフローはキャッシュから削除されます。Syslog メッセージが生成されると、タイマーおよびパケット カウンタがリセットされます。

VACL ロギングの制限事項は次のとおりです。

- 拒否された IP パケットだけが記録されます。
- 発信ポート ACL でロギングが必要なパケットは、VACL で拒否された場合、ロギングされません。

ACL の時間範囲

time-range グローバル コンフィギュレーション コマンドを使用することによって、時刻および曜日に基づいて拡張 ACL を選択的に適用できます。まず、時間範囲の名前を定義し、その時間範囲内の時刻および日付または曜日を設定します。次に、ACL を適用してアクセス リストに制限を設定するときに時間範囲を入力します。時間範囲を使用すると、ACL の許可ステートメントまたは拒否ステートメントの有効期間（指定期間内や指定曜日など）を定義できます。**time-range** キーワードおよび引数については、名前付きおよび番号付き拡張 ACL タスクの表を参照してください。

時間範囲を使用するいくつかの利点を次に示します。

- アプリケーションなどのリソース（IP アドレスとマスクのペア、およびポート番号で識別）へのユーザ アクセスをより厳密に許可または拒否できます。
- ログ メッセージを制御できます。ACL エントリを使用して特定の時刻に関してのみトラフィックをロギングできるため、ピーク時間に生成される多数のログを分析しなくても、簡単にアクセスを拒否できます。

時間ベースのアクセス リストを使用すると、CPU に負荷が生じます。これは、アクセス リストの新規設定を他の機能や、ハードウェア メモリにロードされた結合済みの設定とマージする必要があるためです。そのため、複数のアクセス リストが短期間に連続して（互いに数分以内に）有効となるような設定とならないように注意する必要があります。

（注） 時間範囲は、スイッチのシステム クロックに基づきます。したがって、信頼できるクロック ソースが必要です。ネットワーク タイム プロトコル（NTP）を使用してスイッチ クロックを同期させることを推奨します。

IPv4 ACL のインターフェイスに関する注意事項

ip access-group インターフェイス コンフィギュレーション コマンドをレイヤ 3 インターフェイス（SVI、レイヤ 3 EtherChannel、またはルーテッド ポート）に適用するには、そのインターフェイスに IP アドレスが設定されている必要があります。レイヤ 3 アクセス グループは、CPU のレイヤ 3 プロセスによってルーティングまたは受信されるパケットをフィルタリングします。このグループは、VLAN 内でブリッジングされるパケットに影響を与えません。

インバウンド ACL の場合、パケットの受信後スイッチはパケットを ACL と照合します。ACL がパケットを許可する場合、スイッチはパケットの処理を継続します。ACL がパケットを拒否する場合、スイッチはパケットを廃棄します。

アウトバウンド ACL の場合、パケットを受信し制御対象インターフェイスにルーティングしたあと、スイッチはパケットを ACL と照合します。ACL がパケットを許可する場合、スイッチはパケットを送信します。ACL がパケットを拒否する場合、スイッチはパケットを廃棄します。

デフォルトでは、パケットが廃棄された場合は、その原因が入力インターフェイスの ACL または発信インターフェイスの ACL のいずれであっても、常に入力インターフェイスから ICMP 到達不能メッセージが送信されます。ICMP 到達不能メッセージは通常、入力インターフェイス 1 つにつき、0.5 秒ごとに 1 つだけ生成されます。ただし、この設定は **ip icmp rate-limit unreachable** グローバル コンフィギュレーション コマンドを使用して変更できます。

未定義の ACL をインターフェイスに適用すると、スイッチは ACL がインターフェイスに適用されていないと判断し、すべてのパケットを許可します。ネットワーク セキュリティのために未定義の ACL を使用する場合は、このような結果が生じることに注意してください。

ACL の設定方法

IPv4 ACL の設定

スイッチで IP ACL を使用するには、次の手順に従います。

手順の概要

1. アクセス リストの番号または名前とアクセス条件を指定して、ACL を作成します。
2. ACL をインターフェイスまたは端末回線に適用します。標準および拡張 IP ACL を VLAN マップに適用することもできます。

手順の詳細

ステップ 1 アクセス リストの番号または名前とアクセス条件を指定して、ACL を作成します。

ステップ 2 ACL をインターフェイスまたは端末回線に適用します。標準および拡張 IP ACL を VLAN マップに適用することもできます。

番号付き標準 ACL の作成

番号付き標準 ACL を作成するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard* [**log**]
3. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard* [**log**] 例： スイッチ(config)# **access-list 2 deny your_host**	送信元アドレスとワイルドカードを使用して標準 IPv4 アクセス リストを定義します。 *access-list-number* には、1 ～ 99 または 1300 ～ 1999 の 10 進数を指定します。 条件が一致した場合にアクセスを拒否する場合は **deny** を指定し、許可する場合は **permit** を指定します。 *source* には、パケットの送信元となるネットワークまたはホストのアドレスを次の形式で指定します。 • ドット付き 10 進表記による 32 ビット長の値。 • キーワード **any** は 0.0.0.0 255.255.255.255 という *source* および *source-wildcard* の省略形です。source-wildcard を入力する必要はありません。 • キーワード **host** は送信元および *source* 0.0.0.0 の *source-wildcard* の省略形です。 （任意）*source-wildcard* は、ワイルドカード ビットを送信元アドレスに適用します。 （任意）**log** を指定すると、エントリと一致するパケットに関するログ通知メッセージがコンソールに送信されます。 （任意）**smartlog** を指定すると、拒否または許可されたパケットのコピーが NetFlow 収集装置に送信されます。 （注） ロギングは、レイヤ 3 インターフェイスに割り当てられた ACL でだけサポートされます。
ステップ 3	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	

番号付き拡張 ACL の作成（CLI）

番号付き拡張 ACL を作成するには、次の手順に従います。

手順の概要

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]
3. **access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] **[established]** [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*]
4. **access-list** *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]
5. **access-list** *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*]
6. **access-list** *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] 例： スイッチ(config)# **access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log**	拡張 IPv4 アクセス リストおよびアクセス条件を定義します。 *access-list-number* には、100 ～ 199 または 2000 ～ 2699 の 10 進数を指定します。 条件が一致した場合にパケットを拒否する場合は **deny** を指定し、許可する場合は **permit** を指定します。

	コマンドまたはアクション	目的
		protocol には、インターネット プロトコルの名前または番号を入力します。**ahp**、**eigrp**、**esp**、**gre**、**icmp**、**igmp**、**igrp**、**ip**、**ipinip**、**nos**、**ospf**、**pcp**、**pim**、**tcp**、**udp**、または IP プロトコル番号を表す 0 ～ 255 の整数を使用できます。一致条件としてインターネット プロトコル（ICMP、TCP、UDP など）を指定するには、キーワード **ip** を使用します。 （注） この手順には、ほとんどの IP プロトコルのオプションが含まれています。TCP、UDP、ICMP、および IGMP の追加の特定パラメータについては、次のステップを参照してください。 *source* には、パラメータの送信元であるネットワークまたはホストの番号を指定します。 *source-wildcard* は、ワイルドカード ビットを送信元アドレスに適用します。 *destination* には、パラメータの宛先であるネットワークまたはホストの番号を指定します。 *destination-wildcard* は、ワイルドカード ビットを宛先アドレスに適用します。 source、source-wildcard、destination、および destination-wildcard の値は、次の形式で指定します。 • ドット付き 10 進表記による 32 ビット長の値。 • 0.0.0.0 255.255.255.255（任意のホスト）を表すキーワード **any**。 • 単一のホスト 0.0.0.0 を表すキーワード **host**。 その他のキーワードはオプションであり、次の意味を持ちます。 • **precedence**：パケットを 0 ～ 7 の番号または名前で指定する優先度と一致させる場合に入力します。指定できる値は、**routine**（0）、**priority**（1）、**immediate**（2）、**flash**（3）、**flash-override**（4）、**critical**（5）、**internet**（6）、**network**（7）です。 • **fragments**：2 つ目以降のフラグメントをチェックする場合に入力します。

	コマンドまたはアクション	目的
		• **tos**：パケットを 0 ～ 15 の番号または名前で指定するサービス タイプ レベルと一致させる場合に入力します。指定できる値は、**normal**（0）、**max-reliability**（2）、**max-throughput**（4）、**min-delay**（8）です。 • **log**：エントリと一致するパケットに関するログ通知メッセージを作成し、コンソールに送信します。**log-input** を指定すると、ログエントリに入力インターフェイスが追加されます。 • **time-range**：時間範囲の名前を指定します。 • **dscp**：パケットを 0 ～ 63 の番号で指定する DSCP 値と一致させる場合に入力します。または、指定できる値のリストを表示するには、疑問符（?）を使用します。 （注） コントローラは次の機能をサポートしている必要があります。 • DCSP のマーク • UP のマーク • DSCP と UP のマッピング 「DSCP から UP へのマッピング」の詳細については、次を参照してください。 https://tools.ietf.org/html/draft-ietf-tsvwg-ieee-802-11-01 （注） **dscp** 値を入力する場合は、**tos** または **precedence** を入力できません。**dscp** を入力せずに **tos** と **precedence** の両方の値を入力できます。
ステップ 3	**access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] **[established]** [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*] 例： `スイッチ(config)# access-list 101 permit tcp any`	拡張 TCP アクセス リストおよびアクセス条件を定義します。 次に示す例外を除き、拡張 IPv4 ACL に対して説明するパラメータと同じパラメータを使用します。 （任意）*operator* および *port* を入力すると、送信元ポート（*source source-wildcard* の後に入力した場合）または宛先ポート（*destination destination-wildcard* の

	コマンドまたはアクション	目的
	any eq 500	後に入力した場合）が比較されます。演算子の候補には、**eq**（次の値に等しい）、**gt**（次の値より大きい）、**lt**（次の値より小さい）、**neq**（次の値に等しくない）、および **range**（次の範囲）があります。演算子にはポート番号を指定する必要があります（range の場合は 2 つのポート番号をスペースで区切って指定する必要があります）。 *port* には、10 進数（0 ～ 65535）のポート番号または TCP ポート名を入力します。TCP をフィルタリングするときには、TCP ポートの番号または名前だけを使用します。 他のオプションのキーワードの意味は次のとおりです。 • **established**：確立された接続と照合する場合に入力します。このキーワードは、**ack** または **rst** フラグでの照合と同じ機能を果たします。 • *flag*：指定された TCP ヘッダー ビットを基準にして照合します。入力できるフラグは、**ack**（確認応答）、**fin**（終了）、**psh**（プッシュ）、**rst**（リセット）、**syn**（同期）、または **urg**（緊急）です。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] 例： `スイッチ(config)# access-list 101 permit udp any any eq 100`	（任意）拡張 UDP アクセス リストおよびアクセス条件を定義します。 UDP パラメータは TCP の説明にあるパラメータと同じです。ただし、[operator [port]] ポート番号またはポート名は、UDP ポートの番号または名前でなければなりません。また、UDP では、**flag** と **established** キーワードは無効です。
ステップ 5	**access-list** *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] 例： `スイッチ(config)# access-list 101 permit icmp any any 200`	拡張 ICMP アクセス リストおよびアクセス条件を定義します。 ICMP パラメータは拡張 IPv4 ACL の IP プロトコルの説明にあるパラメータとほとんど同じですが、ICMP メッセージ タイプおよびコード パラメータが追加されています。オプションのキーワードの意味は次のとおりです。

	コマンドまたはアクション	目的
		• *icmp-type*：ICMP メッセージ タイプでフィルタリングする場合に入力します。指定できる値の範囲は、0 ～ 255 です。 • *icmp-code*：ICMP パケットを ICMP メッセージ コードタイプでフィルタリングする場合に入力します。指定できる値の範囲は、0～255 です。 • *icmp-message*：ICMP パケットを ICMP メッセージ タイプ名または ICMP メッセージ タイプとコード名でフィルタリングする場合に入力します。
ステップ 6	**access-list** *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] 例： `スイッチ(config)# access-list 101 permit igmp any any 14`	（任意）拡張 IGMP アクセス リストおよびアクセス条件を定義します。 IGMP パラメータは拡張 IPv4 ACL の IP プロトコルの説明にあるパラメータとほとんど同じですが、次に示すオプションのパラメータが追加されています。 *igmp-type*IGMP メッセージ タイプと比較するには、0 ～ 15 の番号またはメッセージ名（**dvmrp**、**host-query**、**host-report**、**pim**、または **trace**）を入力します。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

名前付き標準 ACL の作成

名前を使用して標準 ACL を作成するには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list standard** *name*
4. 次のいずれかを使用します。
   - **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
   - **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
5. **end**

6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip access-list standard** *name* 例： `スイッチ(config)# ip access-list standard 20`	名前を使用して標準IPv4アクセス リストを定義し、アクセス リスト コンフィギュレーション モードを開始します。 名前には、1 ～ 99 の番号を使用できます。
ステップ 4	次のいずれかを使用します。 • **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**] • **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**] 例： `スイッチ(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255` または `スイッチ(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0`	アクセスリストコンフィギュレーションモードで、パケットを転送するのかドロップするのかを決定する 1 つ以上の拒否条件または許可条件を 指定します。 • **host** *source*：送信元および送信元ワイルドカードの値である *source* 0.0.0.0。 • **any**：送信元および送信元ワイルドカードの値である 0.0.0.0 255.255.255.255
ステップ 5	**end** 例： `スイッチ(config-std-nacl)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

名前付き拡張 ACL の作成

名前を使用して拡張 ACL を作成するには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *name*
4. {**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] **[established]** [**log**] [**time-range** *time-range-name*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**ip access-list extended** *name* 例： `スイッチ(config)# ip access-list extended 150`	名前を使用して拡張IPv4アクセスリストを定義し、アクセス リスト コンフィギュレーション モードを開始します。 名前には、100 ～ 199 の番号を使用できます。
ステップ 4	{**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] **[established]** [**log**] [**time-range** *time-range-name*] 例： `スイッチ(config-ext-nacl)# permit 0 any any`	アクセスリストコンフィギュレーションモードで、許可条件または拒否条件を指定します。**log** キーワードを使用すると、違反を含むアクセスリストのログメッセージを取得できます。 • **host** *source*：送信元および送信元ワイルドカードの値である *source* 0.0.0.0。 • **host** *destination*：接続先および接続先ワイルドカードの値である *destination* 0.0.0.0。 • **any**：source および source wildcard の値または destination および destination wildcard の値である 0.0.0.0 255.255.255.255
ステップ 5	**end** 例： `スイッチ(config-ext-nacl)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

拡張 ACL を作成するときには、ACL の末尾にデフォルトで暗黙的な deny ステートメントが追加され、ACL の終わりに到達するまで一致する条件が見つからなかったすべてのパケットに適用されることに注意してください。標準 ACL では、関連付けられた IP ホスト アドレス アクセス リストの指定からマスクを省略すると、0.0.0.0 がマスクと見なされます。

ACL の作成後に追加したエントリは、リストの末尾に追加されます。ACL エントリを特定の ACL に選択的に追加できません。ただし、**no permit** および **no deny** アクセスリスト コンフィギュレーション モード コマンドを使用すると、名前付き ACL からエントリを削除できます。

番号付き ACL ではなく名前付き ACL を使用する理由の 1 つとして、名前付き ACL では行を選択して削除できることがあります。

次のタスク

作成した名前付き ACL は、インターフェイスまたは VLAN に適用できます。

ACL の時間範囲の設定

ACL の時間範囲パラメータを設定するには、次の手順に従ってください。

手順の概要

1. **enable**
2. **configure terminal**
3. **time-range** *time-range-name*
4. 次のいずれかを使用します。
   - **absolute** [**start** *time date*] [**end** *time date*]
   - **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm*
   - **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ(config)# enable`	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**time-range** *time-range-name* 例： `スイッチ(config)# time-range workhours`	作成する時間範囲には意味のある名前（*workhours* など）を割り当て、時間範囲コンフィギュレーション モードを開始します。名前にスペースや疑問符を含めることはできません。また、文字から始める必要があります。

	コマンドまたはアクション	目的
ステップ 4	次のいずれかを使用します。 • **absolute** [**start** *time date*] [**end** *time date*] • **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm* • **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm* 例： `スイッチ(config-time-range)# absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006` または `スイッチ(config-time-range)# periodic weekdays 8:00 to 12:00`	適用対象の機能がいつ動作可能になるかを指定します。 • 時間範囲には、**absolute** ステートメントを 1 つだけ使用できます。複数の absolute ステートメントを設定した場合は、最後に設定したステートメントだけが実行されます。 • 複数の **periodic** ステートメントを入力できます。たとえば、平日と週末に異なる時間を設定できます。 設定例を参照してください。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

次のタスク

複数の項目をそれぞれ異なる時間に有効にする場合は、上記の手順を繰り返してください。

端末回線への IPv4 ACL の適用

番号付き ACL を使用して、1 つまたは複数の端末回線へのアクセスを制御できます。端末回線には名前付き ACL を適用できません。すべての仮想端末回線にユーザが接続する可能性があるため、すべてに同じ制限を設定する必要があります。

仮想端末回線と ACL に指定されたアドレス間の着信接続および発信接続を制限するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **line** [**console** | **vty**] *line-number*
4. **access-class** *access-list-number* {**in** | **out**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ(config)# enable`	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**line** [**console**	**vty**] *line-number* 例： `スイッチ(config)# line console 0`	設定する回線を指定し、インライン コンフィギュレーション モードを開始します。 • **console**：コンソール端末回線を指定します。コンソール ポートは DCE です。 • **vty**：リモート コンソール アクセス用の仮想端末を指定します。 *line-number* は、回線タイプを指定する場合に、設定する連続グループ内で最初の回線番号です。指定できる範囲は 0 ～ 16です。
ステップ 4	**access-class** *access-list-number* {**in**	**out**} 例： `スイッチ(config-line)# access-class 10 in`	（デバイスへの）特定の仮想端末回線とアクセス リストに指定されたアドレス間の着信接続および発信接続を制限します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `スイッチ(config-line)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

インターフェイスへの IPv4 ACL の適用（CLI）

ここでは、IPv4 ACL をネットワーク インターフェイスへ適用する方法について説明します。

インターフェイスへのアクセスを制御するには、特権 EXEC モードで次の手順に従います。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **ip access-group** {*access-list-number* | *name*} {**in** | **out**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例：	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`Device(config)# interface gigabitethernet1/0/1`	インターフェイスには、レイヤ 2 インターフェイス（ポート ACL）またはレイヤ 3 インターフェイス（ルータ ACL）を指定できます。
ステップ 3	**ip access-group** {*access-list-number* | *name*} {**in** | **out**} 例： `Device(config-if)# ip access-group 2 in`	指定されたインターフェイスへのアクセスを制御します。
ステップ 4	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `Device# show running-config`	アクセス リストの設定を表示します。
ステップ 6	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

名前付き MAC 拡張 ACL の作成

VLAN またはレイヤ 2 インターフェイスで非 IPv4 トラフィックをフィルタリングするには、MAC アドレスおよび名前付き MAC 拡張 ACL を使用します。その手順は、他の名前付き拡張 ACL を設定する場合と同様です。

名前付き MAC 拡張 ACL を作成するには、次の手順に従ってください。

手順の概要

1. **enable**
2. **configure terminal**
3. **mac access-list extended** *name*
4. {**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp** | 0-*65535*] [**cos** *cos*]
5. **end**

6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mac access-list extended** *name* 例： `スイッチ(config)# mac access-list extended mac1`	名前を使用して MAC 拡張アクセス リストを定義します。
ステップ 4	{**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp** | 0-*65535*] [**cos** *cos*] 例： `スイッチ(config-ext-macl)# deny any any decnet-iv` または `スイッチ(config-ext-macl)# permit any any`	拡張 MAC アクセス リスト コンフィギュレーション モードでは、すべての送信元 MAC アドレス、マスク付き送信元 MAC アドレス、または特定の**host**の送信元 MAC アドレスと、**any**の宛先 MAC アドレス、マスク付き宛先 MAC アドレス、または特定の宛先 MAC アドレスに、**permit** または **deny** を指定します。 （任意）次のオプションを入力することもできます。 • *type mask*：Ethernet II または SNAP でカプセル化されたパケットの任意の EtherType 番号。10 進数、16 進数、または 8 進数で表記できます。一致検査の前に、任意で指定できる *don't care* ビットのマスクが EtherType に適用されます。 • **lsap** *lsap mask*：IEEE 802.2 でカプセル化されたパケットの LSAP 番号。10 進数、16 進数、または 8 進数で表記できます。任意で *don't care* ビットのマスクを指定できます。 • **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios**

	コマンドまたはアクション	目的
		| **vines-echo** | **vines-ip** | **xns-idp**：非 IP プロトコル。 • **cos** *cos*：プライオリティを設定する 0 ～ 7 の IEEE 802.1Q CoS 番号。
ステップ 5	**end** 例： `スイッチ(config-ext-macl)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

レイヤ 2 インターフェイスへの MAC ACL の適用

レイヤ 2 インターフェイスへのアクセスを制御するために MAC アクセス リストを適用するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. **configure terminal**
3. **interface** *interface-id*
4. **mac access-group** {*name*} {**in** }
5. **end**
6. **show mac access-group** [**interface** *interface-id*]
7. **configure terminal**
8. **configure terminal**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/2`	特定のインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。指定するインターフェイスは物理レイヤ 2 インターフェイス（ポート ACL）でなければなりません。
ステップ 4	**mac access-group** {*name*} {**in** } 例： `スイッチ(config-if)# mac access-group mac1 in`	MAC アクセス リストを使用して、指定されたインターフェイスへのアクセスを制御します。 ポート ACL は発信および着信方向サポートされます。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show mac access-group** [**interface** *interface-id*] 例： `スイッチ# show mac access-group interface gigabitethernet1/0/2`	そのインターフェイスまたはすべてのレイヤ 2 インターフェイスに適用されている MAC アクセス リストを表示します。
ステップ 7	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 8	**configure terminal** 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	

スイッチは、パケットを受信すると、着信 ACL とパケットを照合します。ACL がパケットを許可する場合、スイッチはパケットの処理を継続します。ACLがパケットを拒否する場合、スイッチはパケットを廃棄します。未定義の ACL をインターフェイスに適用すると、スイッチは ACL がインターフェイスに適用されていないと判断し、すべてのパケットを許可します。ネットワーク セキュリティのために未定義の ACL を使用する場合は、このような結果が生じることに注意してください。

VLAN マップの設定

VLAN マップを作成して 1 つまたは複数の VLAN に適用するには、次の手順に従います。

始める前に

VLAN に適用する標準 IPv4 ACL または拡張 IP ACL、または名前付き MAC 拡張 ACL を作成します。

手順の概要

1. **vlan access-map** *name* [**number**]
2. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
3. IP パケットまたは非 IP パケットを（既知の 1 MAC アドレスのみを使って）指定し、1 つ以上の ACL（標準または拡張）とそのパケットを照合するには、次のコマンドのいずれかを入力します。
   - **action** { **forward**}

     ```
     スイッチ(config-access-map)# action forward
     ```

   - **action** { **drop**}

     ```
     スイッチ(config-access-map)# action drop
     ```

4. **vlan filter** *mapname* **vlan-list** *list*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**vlan access-map** *name* [**number**] 例： スイッチ(config)# **vlan access-map map_1 20**	VLAN マップを作成し、名前と、任意で番号を付けます。番号は、マップ内のエントリのシーケンス番号です。 同じ名前の VLAN マップを作成すると、10 ずつ増加する番号が順に割り当てられます。マップを変更または削除するときは、該当するマップエントリの番号を入力できます。 VLAN マップでは、特定の permit または deny キーワードを使用しません。VLAN マップを使用してパケットを拒否するには、パケットを照合する ACL を作成して、アクションをドロップに設定します。ACL 内の permit は、一致するという意味です。ACL 内の deny は、一致しないという意味です。 このコマンドを入力すると、アクセス マップ コンフィギュレーション モードに変わります。
ステップ 2	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*] 例： スイッチ(config-access-map)# **match ip address ip2**	1 つまたは複数の標準または拡張アクセス リストに対してパケットを照合します（IP または MAC アドレスを使用）。パケットの照合は、対応するプロトコル タイプのアクセス リストに対してだけ行われます。IP パケットは、標準または拡張 IP アクセス リストに対して照合されます。非 IP パケットは、名前付き MAC 拡張アクセス リストに対してだけ照合されます。 （注） パケット タイプ（IP または MAC）に対する match 句が VLAN マップに設定されている場合で、そのマップ アクションがドロップの場合は、そのタイプに一致するすべてのパケットがドロップされます。match 句が VLAN マップになく、設定されているアクションがドロップの場合は、すべての IP およびレイヤ 2 パケットがドロップされます。
ステップ 3	IP パケットまたは非 IP パケットを（既知の 1 MAC アドレスのみを使って）指定し、1 つ以上の ACL（標準または拡張）とそのパケットを照合するには、次のコマンドのいずれかを入力します。 • **action** { **forward**}	マップエントリに対するアクションを設定します。

	コマンドまたはアクション	目的
	スイッチ(config-access-map)# **action forward** • **action** { **drop**} スイッチ(config-access-map)# **action drop**	
ステップ 4	**vlan filter** *mapname* **vlan-list** *list* 例： スイッチ(config)# **vlan filter map 1 vlan-list 20-22**	VLAN マップを 1 つまたは複数の VLAN に適用します。 list には単一の VLAN ID（22）、連続した範囲（10 ～ 22）、または VLAN ID のストリング（12、22、30）を指定できます。カンマやハイフンの前後にスペースを挿入することもできます。

VLAN マップの作成

各 VLAN マップは順番に並べられた一連のエントリで構成されます。VLAN マップ エントリを作成、追加、または削除するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **vlan access-map** *name* [**number**]
3. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
4. **action** {**drop** | **forward**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vlan access-map** *name* [**number**] 例： スイッチ(config)# **vlan access-map map_1 20**	VLAN マップを作成し、名前と、任意で番号を付けます。番号は、マップ内のエントリのシーケンス番号です。

	コマンドまたはアクション	目的
		同じ名前の VLAN マップを作成すると、10 ずつ増加する番号が順に割り当てられます。マップを変更または削除するときは、該当するマップエントリの番号を入力できます。 VLAN マップでは、特定の permit または deny キーワードを使用しません。VLAN マップを使用してパケットを拒否するには、パケットを照合する ACL を作成して、アクションをドロップに設定します。ACL 内の permit は、一致するという意味です。ACL 内の deny は、一致しないという意味です。 このコマンドを入力すると、アクセス マップ コンフィギュレーション モードに変わります。
ステップ 3	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*] 例： `スイッチ(config-access-map)# match ip address ip2`	1 つまたは複数の標準または拡張アクセス リストに対してパケットを照合します（IP または MAC アドレスを使用）。パケットの照合は、対応するプロトコル タイプのアクセス リストに対してだけ行われます。IP パケットは、標準または拡張 IP アクセス リストに対して照合されます。非 IP パケットは、名前付き MAC 拡張アクセス リストに対してだけ照合されます。
ステップ 4	**action** {**drop** | **forward**} 例： `スイッチ(config-access-map)# action forward`	（任意）マップエントリに対するアクションを設定します。デフォルトは転送（forward）です。
ステップ 5	**end** 例： `スイッチ(config-access-map)# end`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	アクセス リストの設定を表示します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VLAN への VLAN マップの適用

VLAN マップを 1 つまたは複数の VLAN に適用するには、次の手順に従います。

手順の概要

1.
2. **configure terminal**
3. **vlan filter** *mapname* **vlan-list** *list*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1		
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vlan filter** *mapname* **vlan-list** *list* 例： `スイッチ(config)# vlan filter map 1 vlan-list 20-22`	VLAN マップを 1 つまたは複数の VLAN に適用します。 list には単一の VLAN ID（22）、連続した範囲（10 ～ 22）、または VLAN ID のストリング（12、22、30）を指定できます。カンマやハイフンの前後にスペースを挿入することもできます。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	アクセス リストの設定を表示します。
ステップ 6	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

VACL ロギングの設定

特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **vlan access-map name** [*number*]
3. **action drop log**
4. **exit**
5. **vlan access-log** { **maxflow** *max_number* | **threshold** *pkt_count*}
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vlan access-map name** [*number*] 例： スイッチ(config)# **vlan access-map gandymede 10**	VLAN マップを作成します。VLAN マップに名前と番号（任意）を付けます。番号は、マップ内のエントリのシーケンス番号です。 シーケンス番号の範囲は 0 ～ 65535 です。 同じ名前の VLAN マップを作成すると、10 ずつ増加する番号が順に割り当てられます。マップを変更または削除するときは、該当するマップエントリの番号を入力できます。 マップ名と番号（任意）を指定すると、アクセス マップ コンフィギュレーション モードが開始されます。
ステップ 3	**action drop log** 例： スイッチ(config-access-map)# **action drop log**	IP パケットを破棄およびロギングするよう VLAN アクセス マップを設定します。

	コマンドまたはアクション	目的
ステップ 4	**exit** 例： スイッチ(config-access-map)# **exit**	VLANアクセスマップコンフィギュレーションモードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 5	**vlan access-log** { **maxflow** *max_number* | **threshold** *pkt_count*} 例： スイッチ(config)# **vlan access-log threshold 4000**	VACL ロギング パラメータを設定します。 • **maxflow** *max_number*：ログテーブルのサイズを設定します。**maxflow** の値を 0 に設定すると、ログテーブルの内容を削除できます。ログテーブルがいっぱいの場合は、ロギングされたパケットがソフトウェアによって新しいフローから破棄されます。 値の範囲は、0 ～ 2048 です。デフォルトは 500 です。 • **threshold** *pkt_count*：ロギングしきい値を設定します。5 分経過する前にフローのしきい値に達すると、ログ メッセージが生成されます。 しきい値の範囲は 0 ～ 2147483647 です。デフォルトのしきい値は 0 であり、Syslog メッセージが 5 分ごとに生成されます。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

IPv4 ACL のモニタリング

スイッチに設定されている ACL、およびインターフェイスと VLAN に適用済みの ACL を表示することで、IPv4 ACL をモニターできます。

ip access-group インターフェイス コンフィギュレーション コマンドを使用して、レイヤ 2 またはレイヤ 3 インターフェイスに ACL を適用した場合は、そのインターフェイスのアクセス

グループを表示できます。また、レイヤ 2 インターフェイスに適用された MAC ACL も表示できます。この情報を表示するには、次の表に記載された特権 EXEC コマンドを使用します。

表 *133 :* アクセス リストおよびアクセス グループを表示するコマンド

コマンド	目的
show access-lists [*number* | *name*]	最新の IP および MAC アドレス アクセス リストの全体やその一部、または特定のアクセス リスト（番号付きまたは名前付き）の内容を表示します。
show ip access-lists [*number* | *name*]	最新の IP アクセス リスト全体、または特定の IP アクセス リスト（番号付きまたは名前付き）を表示します。
show ip interface *interface-id*	インターフェイスの詳細設定およびステータスを表示します。IP がイネーブルになっているインターフェイスに、**ip access-group** インターフェイス コンフィギュレーション コマンドを使用して ACL を適用した場合は、アクセス グループも表示に含まれます。
show running-config [**interface** *interface-id*]	スイッチまたは指定されたインターフェイスのコンフィギュレーション ファイルの内容（設定されたすべての MAC および IP アクセス リストや、どのアクセス グループがインターフェイスに適用されたかなど）を表示します。
show mac access-group [**interface** *interface-id*]	すべてのレイヤ 2 インターフェイスまたは指定されたレイヤ 2 インターフェイスに適用されている MAC アクセス リスト を表示します。

ACL の設定例

例：ACL での時間範囲を使用

次の例に、*workhours*（営業時間）の時間範囲および会社の休日（2006 年 1 月 1 日）を設定し、設定を確認する例を示します。

```
スイッチ# show time-range
time-range entry: new_year_day_2003 (inactive)
   absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
   periodic weekdays 8:00 to 12:00
```

```
   periodic weekdays 13:00 to 17:00
```

時間範囲を適用するには、時間範囲を実装できる拡張 ACL 内に時間範囲名を入力します。次に、拡張アクセス リスト 188 を作成して確認する例を示します。このアクセス リストでは、定義された休業時間中はすべての送信元からすべての宛先への TCP トラフィックを拒否し、営業時間中はすべての TCP トラフィックを許可します。

```
スイッチ(config)# access-list 188 deny tcp any any time-range new_year_day_2006
スイッチ(config)# access-list 188 permit tcp any any time-range workhours
スイッチ(config)# end
スイッチ# show access-lists
Extended IP access list 188
   10 deny tcp any any time-range new_year_day_2006 (inactive)
   20 permit tcp any any time-range workhours (inactive)
```

次に、名前付き ACL を使用して同じトラフィックを許可および拒否する例を示します。

```
スイッチ(config)# ip access-list extended deny_access
スイッチ(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
スイッチ(config-ext-nacl)# exit
スイッチ(config)# ip access-list extended may_access
スイッチ(config-ext-nacl)# permit tcp any any time-range workhours
スイッチ(config-ext-nacl)# end
スイッチ# show ip access-lists
Extended IP access list lpip_default
    10 permit ip any any
Extended IP access list deny_access
    10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
    10 permit tcp any any time-range workhours (inactive)
```

例：ACL へのコメントの挿入

remark キーワードを使用すると、任意の IP 標準または拡張 ACL にエントリに関するコメント（注釈）を追加できます。コメントを使用すると、ACL の理解とスキャンが容易になります。1 つのコメント行の最大長は 100 文字です。

コメントは、permit ステートメントまたは deny ステートメントの前後どちらにでも配置できます。コメントがどの permit ステートメントまたは deny ステートメントの説明であるのかが明確になるように、コメントの位置に関して一貫性を保つ必要があります。たとえば、あるコメントは対応する permit または deny ステートメントの前にあり、他のコメントは対応するステートメントの後ろにあると、混乱を招く可能性があります。

番号付き IP 標準または拡張 ACL にコメントを挿入するには、**access-list** *access-list number* **remark** *remark* グローバル コンフィギュレーション コマンドを使用します。コメントを削除するには、このコマンドの **no** 形式を使用します。

次の例では、Jones のワークステーションにはアクセスを許可し、Smith のワークステーションにはアクセスを許可しません。

```
スイッチ(config)# access-list 1 remark Permit only Jones workstation through
スイッチ(config)# access-list 1 permit 171.69.2.88
スイッチ(config)# access-list 1 remark Do not allow Smith through
スイッチ(config)# access-list 1 deny 171.69.3.13
```

名前付き IP ACL のエントリには、**remark** アクセスリスト コンフィギュレーション コマンドを使用します。コメントを削除するには、このコマンドの **no** 形式を使用します。

次の例では、Jones のサブネットには発信 Telnet の使用が許可されません。

```
スイッチ(config)# ip access-list extended telnetting
スイッチ(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
スイッチ(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

例：ACL のトラブルシューティング

次の ACL マネージャ メッセージが表示されて [chars] がアクセス リスト名である場合は次のようにします。

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

スイッチには、ACL のハードウェア表現を作成するのに使用可能なリソースが不足しています。このリソースには、ハードウェア メモリおよびラベル スペースが含まれますが、CPU メモリは含まれません。この問題の原因は、使用可能な論理演算ユニットまたは専用のハードウェア リソースの不足です。論理演算ユニットは、TCP フラグの一致、または TCP、UDP、SCTP ポート番号での **eq**（**ne**、**gt**、**lt**、または **range**）のテストで必要です。

次のいずれかの回避策を使用します。

- ACL の設定を変更して使用するリソースを減らします。
- 他の ACL 名または番号よりも英数字順で先に表示される名前または番号に ACL の名前を変更します。

専用のハードウェアリソースを識別するには、**show platform layer4 acl** 特権 EXEC コマンドを入力します。スイッチに使用可能なリソースがない場合は、出力に index 0 ～ index 15 が使用できないことが示されます。

十分なリソースがない ACL の設定の詳細については、Bug Toolkit の CSCsq63926 を参照してください。

たとえば、次の ACL をインターフェイスに適用します。

```
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
permit tcp source source-wildcard destination destination-wildcard
```

なおかつ次のメッセージが表示される場合は次のようにします。

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

フラグ関連の演算子は使用できません。この問題を回避するには、

- **ip access-list resequence** グローバル コンフィギュレーション コマンドを使用することによって、4 つ目の ACE を 1 つ目の ACE の前に移動させます。

```
permit tcp source source-wildcard destination destination-wildcard
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
```

  または

- 他の ACL 名または番号よりも英数字順で先に表示される名前または番号に ACL の名前を変更します（たとえば、ACL *79* を ACL *1* に変更します）。

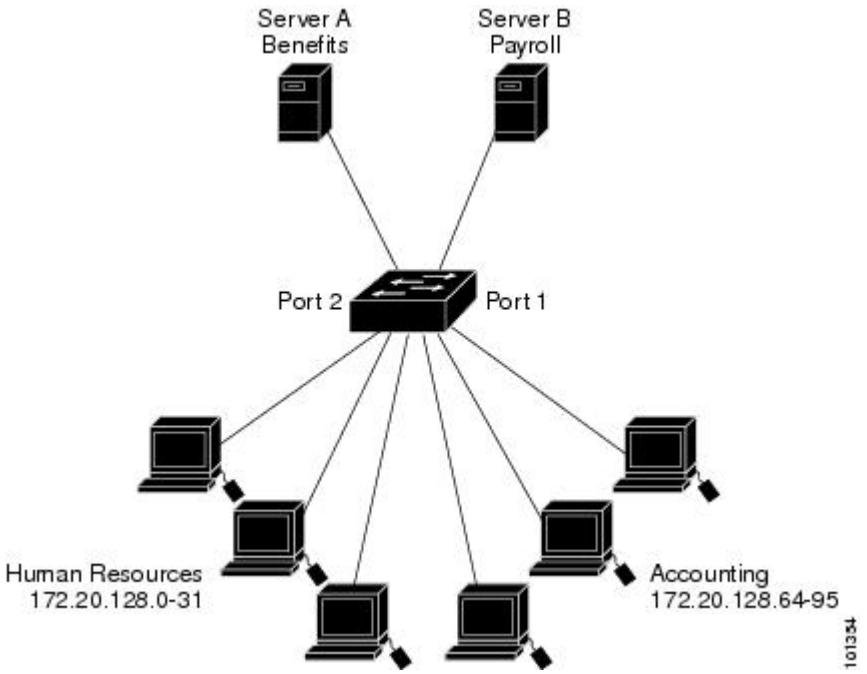
これで、ACL 内の 1 つ目の ACE をインターフェイスに適用できます。スイッチによって、ACE が、Opselect インデックス内の利用可能なマッピング ビットに割り当てられ、次に、ハードウェア メモリ内の同じビットを使用するフラグ関連の演算子が割り当てられます。

IPv4 ACL の設定例

ここでは、IPv4 ACL を設定および適用する例を示します。ACL のコンパイルに関する詳細については、『*Cisco IOS Security Configuration Guide, Release 12.4*』および『*Cisco IOS IP Configuration Guide, Release 12.4*』の「IP Adderssing and Services」の章にある「Configuring IP Services」の項を参照してください。

小規模ネットワークが構築されたオフィス用の ACL

図 *96* : ルータ *ACL* によるトラフィックの制御



次に、小規模ネットワークが構築されたオフィス環境を示します。ルーテッドポート2に接続されたサーバー A には、すべての従業員がアクセスできる収益などの情報が格納されています。ルーテッド ポート 1 に接続されたサーバー B には、機密扱いの給与支払いデータが格納されています。サーバー A にはすべてのユーザーがアクセスできますが、サーバー B にアクセスできるユーザーは制限されています。

ルータ ACL を使用して上記のように設定するには、次のいずれかの方法を使用します。

- 標準 ACL を作成し、ポート 1 からサーバーに着信するトラフィックをフィルタリングします。
- 拡張 ACL を作成し、サーバーからポート 1 に着信するトラフィックをフィルタリングします。

例：小規模ネットワークが構築されたオフィスの ACL

次に、標準 ACL を使用してポートからサーバー B に着信するトラフィックをフィルタリングし、経理部の送信元アドレス 172.20.128.64 ～ 172.20.128.95 から送信されるトラフィックだけを許可する例を示します。この ACL は、指定された送信元アドレスを持つルーテッド ポート 1 から送信されるトラフィックに適用されます。

```
スイッチ(config)# access-list 6 permit 172.20.128.64 0.0.0.31
スイッチ(config)# end
スイッチ# how access-lists
Standard IP access list 6
```

```
    10 permit 172.20.128.64, wildcard bits 0.0.0.31
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# ip access-group 6 out
```

次に、拡張 ACL を使用してサーバー B からポートに着信するトラフィックをフィルタリングし、任意の送信元アドレス（この場合はサーバー B）から経理部の宛先アドレス 172.20.128.64 ～ 172.20.128.95 に送信されるトラフィックだけを許可する例を示します。この ACL は、ルーテッド ポート 1 に着信するトラフィックに適用され、指定の宛先アドレスに送信されるトラフィックだけを許可します。拡張 ACL を使用する場合は、送信元および宛先情報の前に、プロトコル（IP）を入力する必要があります。

```
スイッチ(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
スイッチ(config)# end
スイッチ# show access-lists
Extended IP access list 106
    10 permit ip any 172.20.128.64 0.0.0.31
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# ip access-group 106 in
```

例：番号付き ACL

次の例のネットワーク 10.0.0.0 は、2 番目のオクテットがサブネットを指定するクラス A ネットワークです。つまり、サブネットマスクは 255.255.0.0 です。ネットワーク 10.0.0.0 アドレスの 3 番目および 4 番目のオクテットで特定のホストを指定します。アクセス リスト 2 を使用して、サブネット 48 のアドレスを 1 つ許可し、同じサブネットの他のアドレスはすべて拒否します。このアクセス リストの最終行は、ネットワーク 10.0.0.0 の他のすべてのサブネット上のアドレスが許可されることを示します。この ACL は、ポートに着信するパケットに適用されます。

```
スイッチ(config)# access-list 2 permit 10.48.0.3
スイッチ(config)# access-list 2 deny 10.48.0.0 0.0.255.255
スイッチ(config)# access-list 2 permit 10.0.0.0 0.255.255.255
スイッチ(config)# interface gigabitethernet2/0/1
スイッチ(config-if)# ip access-group 2 in
```

例：拡張 ACL

次の例の先頭行は、1023 よりも大きい宛先ポートへの着信 TCP 接続を許可します。2 番目の行は、ホスト 128.88.1.2 のシンプルメール転送プロトコル（SMTP）ポートへの着信 TCP 接続を許可します。3 番めの行は、エラー フィードバック用の着信 ICMP メッセージを許可します。

```
スイッチ(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
スイッチ(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
スイッチ(config)# access-list 102 permit icmp any any
スイッチ(config)# interface gigabitethernet2/0/1
```

```
スイッチ(config-if)# ip access-group 102 in
```

次の例では、インターネットに接続されたネットワークがあり、そのネットワーク上の任意のホストがインターネット上の任意のホストと TCP 接続を確立できるようにする場合を想定しています。ただし、IP ホストからは、専用メール ホストのメール（SMTP）ポートを除き、ネットワーク上のホストと TCP 接続を確立できないようにします。

SMTP は、接続の一端では TCP ポート 25、もう一端ではランダムなポート番号を使用します。接続している間は、同じポート番号が使用されます。インターネットから着信するメール パケットの宛先ポートは 25 です。発信パケットのポート番号は予約されています。安全なネットワーク システムでは常にポート 25 でのメール接続が使用されているため、着信サービスと発信サービスを個別に制御できます。ACL は発信インターフェイスの入力 ACL および着信インターフェイスの出力 ACL として設定される必要があります。

```
スイッチ(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
スイッチ(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# ip access-group 102 in
```

次の例では、ネットワークはアドレスが 128.88.0.0 のクラス B ネットワークで、メール ホストのアドレスは 128.88.1.2 です。**established** キーワードは、確立された接続を表示する TCP 専用のキーワードです。TCP データグラムに ACK または RST ビットが設定され、パケットが既存の接続に属していることが判明すると、一致と見なされます。スタック メンバー 1 のギガビット イーサネット インターフェイス 1 は、ルータをインターネットに接続するインターフェイスです。

```
スイッチ(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
スイッチ(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# ip access-group 102 in
```

例：名前付き ACL

名前付き標準 ACL および名前付き拡張 ACL の作成

次に、*Internet_filter* という名前の標準 ACL および *marketing_group* という名前の拡張 ACL を作成する例を示します。*Internet_filter* ACL は、送信元アドレス 1.2.3.4 から送信されるすべてのトラフィックを許可します。

```
スイッチ(config)# ip access-list standard Internet_filter
スイッチ(config-ext-nacl)# permit 1.2.3.4
スイッチ(config-ext-nacl)# exit
```

marketing_group ACL は、宛先アドレスとワイルドカードの値 171.69.0.0 0.0.255.255 への任意の TCP Telnet トラフィックを許可し、その他の TCP トラフィックを拒否します。ICMP トラフィックを許可し、任意の送信元から、宛先ポートが 1024 より小さい 171.69.0.0 ～ 179.69.255.255 の

宛先アドレスへ送信されるUDPトラフィックを拒否します。それ以外のすべてのIPトラフィックを拒否して、結果を示すログが表示されます。

```
スイッチ(config)# ip access-list extended marketing_group
スイッチ(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
スイッチ(config-ext-nacl)# deny tcp any any
スイッチ(config-ext-nacl)# permit icmp any any
スイッチ(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
スイッチ(config-ext-nacl)# deny ip any any log
スイッチ(config-ext-nacl)# exit
```

Internet_filter ACL は発信トラフィックに適用され、*marketing_group* ACL はレイヤ 3 ポートの着信トラフィックに適用されます。

```
スイッチ(config)# interface gigabitethernet3/0/1
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 2.0.5.1 255.255.255.0
スイッチ(config-if)# ip access-group Internet_filter out
スイッチ(config-if)# ip access-group marketing_group in
```

名前付き ACL からの個別 ACE の削除

次に、名前付きアクセス リスト *border-list* から ACE を個別に削除する例を示します。

```
スイッチ(config)# ip access-list extended border-list
スイッチ(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

例：IP ACL に適用される時間範囲

次に、月曜日から金曜日の午前 8 時 ～午後 6 時（18 時）の間、IP の HTTP トラフィックを拒否する例を示します。UDP トラフィックは、土曜日および日曜日の正午～午後 8 時（20 時）の間だけ許可されます。

```
スイッチ(config)# time-range no-http
スイッチ(config)# periodic weekdays 8:00 to 18:00
!
スイッチ(config)# time-range udp-yes
スイッチ(config)# periodic weekend 12:00 to 20:00
!
スイッチ(config)# ip access-list extended strict
スイッチ(config-ext-nacl)# deny tcp any any eq www time-range no-http
スイッチ(config-ext-nacl)# permit udp any any time-range udp-yes
!
スイッチ(config-ext-nacl)# exit
スイッチ(config)# interface gigabitethernet2/0/1
スイッチ(config-if)# ip access-group strict in
```

例：コメント付き IP ACL エントリの設定

次に示す番号付き ACL の例では、Jones が所有するワークステーションにはアクセスを許可し、Smith が所有するワークステーションにはアクセスを許可しません。

```
スイッチ(config)# access-list 1 remark Permit only Jones workstation through
スイッチ(config)# access-list 1 permit 171.69.2.88
スイッチ(config)# access-list 1 remark Do not allow Smith workstation through
スイッチ(config)# access-list 1 deny 171.69.3.13
```

次に示す番号付き ACL の例では、Winter および Smith のワークステーションに Web 閲覧を許可しません。

```
スイッチ(config)# access-list 100 remark Do not allow Winter to browse the web
スイッチ(config)# access-list 100 deny host 171.69.3.85 any eq www
スイッチ(config)# access-list 100 remark Do not allow Smith to browse the web
スイッチ(config)# access-list 100 deny host 171.69.3.13 any eq www
```

次に示す名前付き ACL の例では、Jones のサブネットにアクセスを許可しません。

```
スイッチ(config)# ip access-list standard prevention
スイッチ(config-std-nacl)# remark Do not allow Jones subnet through
スイッチ(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

次に示す名前付き ACL の例では、Jones のサブネットに発信 Telnet の使用を許可しません。

```
スイッチ(config)# ip access-list extended telnetting
スイッチ(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
スイッチ(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

例：ACL ロギング

ACL では 2 種類のロギングがサポートされています。**log** キーワードを指定すると、エントリと一致するパケットに関するログ通知メッセージがコンソールに送信されます。**log-input** キーワードを指定すると、ログ エントリに入力インターフェイスが追加されます。

次の例では、名前付き標準アクセス リスト *stan1* は 10.1.1.0 0.0.0.255 からのトラフィックを拒否し、その他のすべての送信元からのトラフィックを許可します。**log** キーワードも指定されています。

```
スイッチ(config)# ip access-list standard stan1
スイッチ(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
スイッチ(config-std-nacl)# permit any log
スイッチ(config-std-nacl)# exit
スイッチ(config)# interface gigabitethernet1/0/1
スイッチ(config-if)# ip access-group stan1 in
スイッチ(config-if)# end
```

```
スイッチ# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 37 messages logged
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 37 messages logged
    File logging: disabled
    Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

次に、名前付き拡張アクセス リスト *ext1* によって、任意の送信元から 10.1.1.0 0.0.0.255 への ICMP パケットを許可し、すべての UDP パケットを拒否する例を示します。

```
スイッチ(config)# ip access-list extended ext1
スイッチ(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
スイッチ(config-ext-nacl)# deny udp any any log
スイッチ(config-std-nacl)# exit
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# ip access-group ext1 in
```

次に、拡張 ACL のログの例を示します。

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1
packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8
packets
```

IP ACL のすべてのロギング エントリは %SEC-6-IPACCESSLOG で開始します。エントリの形式は、一致した ACL やアクセス エントリの種類に応じて若干異なります。

次に、**log-input** キーワードを指定した場合の出力メッセージの例を示します。

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1 0001.42ef.a400)
 ->
10.1.1.61 (0/0), 1 packet
```

log キーワードを指定した場合、同様のパケットに関するログ メッセージには入力インターフェイス情報が含まれません。

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0),
 1 packet
```

ACL および VLAN マップの設定例

例：パケットを拒否する ACL および VLAN マップの作成

ここでは、パケットを拒否する ACL および VLAN マップを作成する例を示します。最初のマップでは、*ip1* ACL（TCP パケット）に一致するすべてのパケットがドロップされます。最初に、すべての TCP パケットを許可し、それ以外のパケットをすべて拒否する *ip1* ACL を作成します。VLAN マップには IP パケットに対する match 句が存在するため、デフォルトのアクションでは、どの match 句とも一致しない IP パケットがすべてドロップされます。

```
スイッチ(config)# ip access-list extended ip1
スイッチ(config-ext-nacl)# permit tcp any any
スイッチ(config-ext-nacl)# exit
スイッチ(config)# vlan access-map map_1 10
スイッチ(config-access-map)# match ip address ip1
スイッチ(config-access-map)# action drop
```

例：パケットを許可する ACL および VLAN マップの作成

次に、パケットを許可する VLAN マップを作成する例を示します。ACL *ip2* は UDP パケットを許可し、*ip2* ACL と一致するすべてのパケットが転送されます。このマップでは、これ以前のどの ACL とも一致しないすべての IP パケット（TCP でも UDP でもないパケット）がドロップされます。

```
スイッチ(config)# ip access-list extended ip2
スイッチ(config-ext-nacl)# permit udp any any
スイッチ(config-ext-nacl)# exit
スイッチ(config)# vlan access-map map_1 20
スイッチ(config-access-map)# match ip address ip2
スイッチ(config-access-map)# action forward
```

例：IP パケットのドロップおよび MAC パケットの転送のデフォルト アクション

次の例の VLAN マップでは、デフォルトで IP パケットがドロップされ、MAC パケットが転送されます。標準の ACL 101 および名前付き拡張アクセス リスト **igmp-match** および **tcp-match** をこのマップと組み合わせて使用すると、次のようになります。

- すべての UDP パケットが転送されます。
- すべての IGMP パケットがドロップされます。
- すべての TCP パケットが転送されます。
- その他のすべての IP パケットがドロップされます。
- すべての非 IP パケットが転送されます。

```
スイッチ(config)# access-list 101 permit udp any any
スイッチ(config)# ip access-list extended igmp-match
```

```
スイッチ(config-ext-nacl)# permit igmp any any
```

```
スイッチ(config-ext-nacl)# permit tcp any any
スイッチ(config-ext-nacl)# exit
スイッチ(config)# vlan access-map drop-ip-default 10
スイッチ(config-access-map)# match ip address 101
スイッチ(config-access-map)# action forward
スイッチ(config-access-map)# exit
スイッチ(config)# vlan access-map drop-ip-default 20
スイッチ(config-access-map)# match ip address igmp-match
スイッチ(config-access-map)# action drop
スイッチ(config-access-map)# exit
スイッチ(config)# vlan access-map drop-ip-default 30
スイッチ(config-access-map)# match ip address tcp-match
スイッチ(config-access-map)# action forward
```

例：MAC パケットのドロップおよび IP パケットの転送のデフォルト アクション

次の例の VLAN マップでは、デフォルトで MAC パケットがドロップされ、IP パケットが転送されます。MAC 拡張アクセス リスト **good-hosts** および **good-protocols** をこのマップと組み合わせて使用すると、次のようになります。

- ホスト 0000.0c00.0111 および 0000.0c00.0211 からの MAC パケットが転送されます。
- decnet-iv または vines-ip プロトコルを使用する MAC パケットが転送されます。
- その他のすべての非 IP パケットがドロップされます。
- すべての IP パケットが転送されます。

例：すべてのパケットをドロップするデフォルト アクション

次の例の VLAN マップでは、デフォルトですべてのパケット（IP および非 IP）がドロップされます。例 2 および例 3 のアクセス リスト **tcp-match** および **good-hosts** をこのマップと組み合わせて使用すると、次のようになります。

- すべての TCP パケットが転送されます。
- ホスト 0000.0c00.0111 および 0000.0c00.0211 からの MAC パケットが転送されます。
- その他のすべての IP パケットがドロップされます。
- その他のすべての MAC パケットがドロップされます。

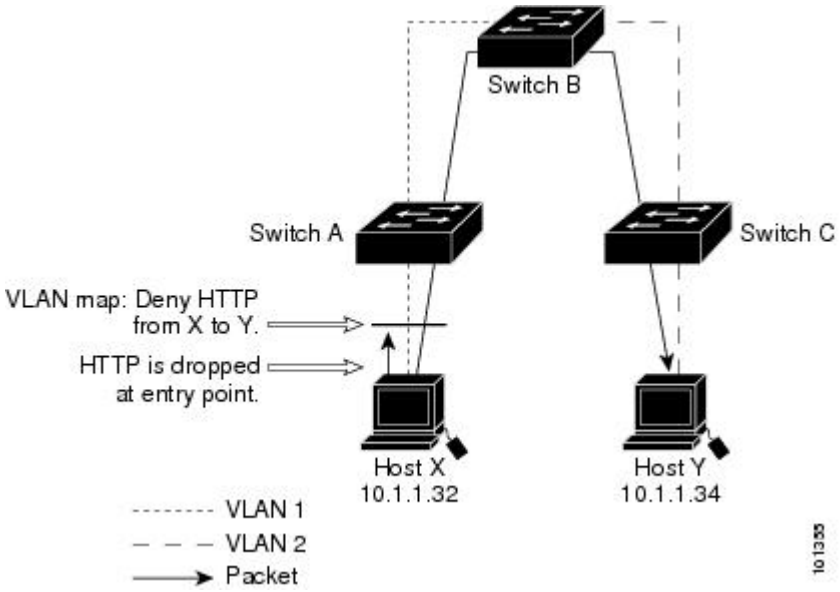
```
スイッチ(config)# vlan access-map drop-all-default 10
スイッチ(config-access-map)# match ip address tcp-match
スイッチ(config-access-map)# action forward
スイッチ(config-access-map)# exit
スイッチ(config)# vlan access-map drop-all-default 20
スイッチ(config-access-map)# match mac address good-hosts
スイッチ(config-access-map)# action forward
```

ネットワークでの VLAN マップの使用方法の設定例

例：ワイヤリング クローゼットの設定

図 *97 :* ワイヤリング クローゼットの設定

ワイヤリング クローゼット構成では、ルーティングがスイッチ上で有効にされていない場合があります。ただし、この設定でも VLAN マップおよび QoS 分類 ACL はサポートされています。ホスト X およびホスト Y は異なる VLAN 内にあり、ワイヤリング クローゼット スイッチ A およびスイッチ C に接続されていると想定します。ホスト X からホスト Y へのトラフィックは、ルーティングが有効に設定されたレイヤ 3 スイッチであるスイッチ B によって最終的にルーティングされます。ホスト X からホスト Y へのトラフィックは、トラフィックのエントリ ポイントであるスイッチ A でアクセス コントロールできます。



HTTP トラフィックをホスト X からホスト Y へスイッチングしない場合は、ホスト X（IP アドレス 10.1.1.32）からホスト Y（IP アドレス 10.1.1.34）に向かうすべての HTTP トラフィックがスイッチ A でドロップされ、スイッチ B にブリッジングされないように、スイッチ A の VLAN マップを設定できます。

最初に、HTTP ポート上ですべての TCP トラフィックを許可（一致）する IP アクセス リスト *http* を定義します。

```
スイッチ(config)# ip access-list extended http
スイッチ(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
スイッチ(config-ext-nacl)# exit
```

次に、*http* アクセス リストと一致するトラフィックがドロップされ、その他のすべての IP トラフィックが転送されるように、VLAN アクセス マップ *map2* を作成します。

```
スイッチ(config)# vlan access-map map2 10
```

```
スイッチ(config-access-map)# match ip address http
スイッチ(config-access-map)# action drop
スイッチ(config-access-map)# exit
スイッチ(config)# ip access-list extended match_all
スイッチ(config-ext-nacl)# permit ip any any
スイッチ(config-ext-nacl)# exit
スイッチ(config)# vlan access-map map2 20
スイッチ(config-access-map)# match ip address match_all
スイッチ(config-access-map)# action forward
```

次に、VLAN アクセス マップ *map2* を VLAN 1 に適用します。

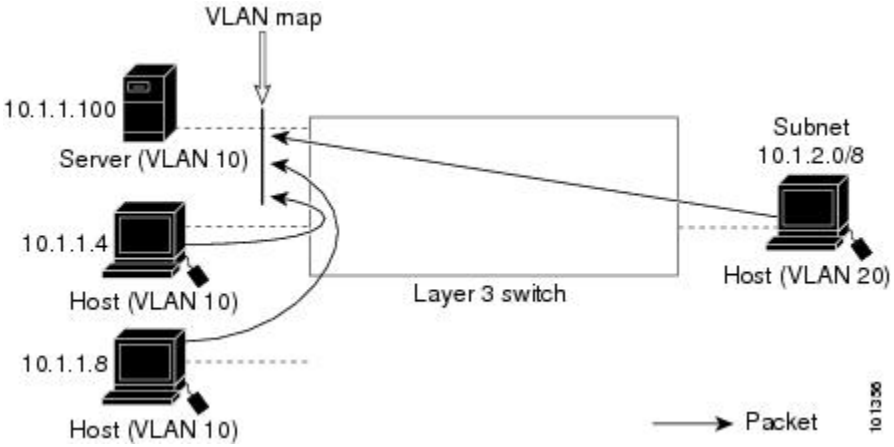
```
スイッチ(config)# vlan filter map2 vlan 1
```

例：別の VLAN にあるサーバーへのアクセスの制限

図 *98* : 別の *VLAN* 上のサーバーへのアクセスの制限

別の VLAN にあるサーバーへのアクセスを制限できます。たとえば、VLAN 10 内のサーバー 10.1.1.100 では、次のホストへのアクセスを拒否する必要があります。

- VLAN 20 内のサブネット 10.1.2.0/8 にあるホストのアクセスを禁止します。
- VLAN 10 内のホスト 10.1.1.4 および 10.1.1.8 のアクセスを禁止します。



例：別の VLAN にあるサーバーへのアクセスの拒否

次に、サブネット 10.1.2.0.8 内のホスト、ホスト 10.1.1.4、およびホスト 10.1.1.8 のアクセスを拒否し、その他の IP トラフィックを許可する VLAN マップ SERVER1-ACL を作成して、別の VLAN 内のサーバーへのアクセスを拒否する例を示します。最後のステップでは、マップ SERVER1 を VLAN 10 に適用します。

正しいパケットと一致する IP ACL を定義します。

```
スイッチ(config)# ip access-list extended SERVER1_ACL
```

```
スイッチ(config-ext-nacl))# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
スイッチ(config-ext-nacl))# permit ip host 10.1.1.4 host 10.1.1.100
スイッチ(config-ext-nacl))# permit ip host 10.1.1.8 host 10.1.1.100
スイッチ(config-ext-nacl))# exit
```

SERVER1_ACL と一致する IP パケットをドロップして、この ACL と一致しない IP パケットを転送する ACL を使用して、VLAN マップを定義します。

```
スイッチ(config)# vlan access-map SERVER1_MAP
スイッチ(config-access-map)# match ip address SERVER1_ACL
スイッチ(config-access-map)# action drop
スイッチ(config)# vlan access-map SERVER1_MAP 20
スイッチ(config-access-map)# action forward
スイッチ(config-access-map)# exit
```

VLAN 10 に VLAN マップを適用します。

```
スイッチ(config)# vlan filter SERVER1_MAP vlan-list 10
```

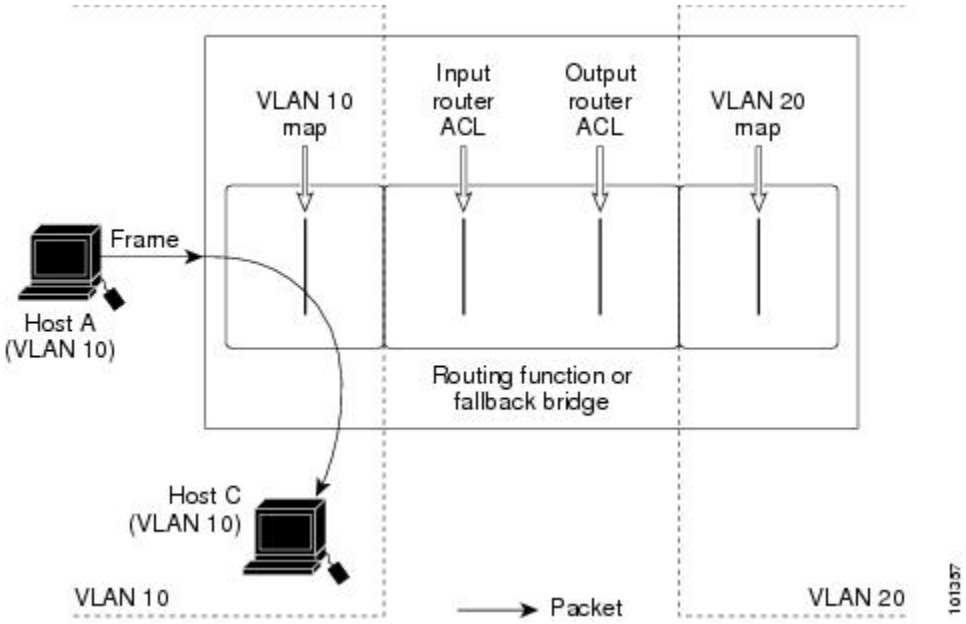
VLAN に適用されるルータ ACL と VLAN マップの設定例

ここでは、ルータ ACL および VLAN マップを VLAN に適用し、スイッチド パケット、ブリッジド パケット、ルーテッド パケット、およびマルチキャスト パケットを処理する例を示します。次の図ではそれぞれの宛先に転送されるパケットを示します。パケットのパスが VLAN マップや ACL を示す線と交差するポイントで、パケットを転送せずにドロップする可能性もあります。

例：ACL およびスイッチド パケット

図 *99 :* スイッチド パケットへの *ACL* の適用

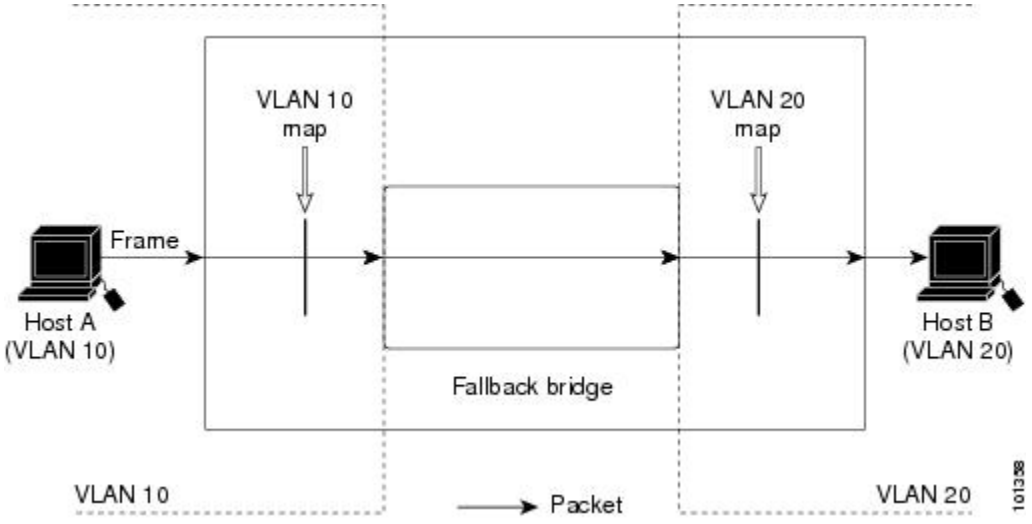
次の例に、VLAN 内でスイッチングされるパケットに ACL を適用する方法を示します。フォールバック ブリッジングによってルーティングまたは転送されず、VLAN 内でスイッチングされるパケットには、入力 VLAN の VLAN マップだけが適用されます。



例：ACL およびブリッジド パケット

図 *100* : ブリッジド パケットへの *ACL* の適用

次の例に、フォールバック ブリッジド パケットに ACL を適用する方法を示します。ブリッジド パケットの場合は、入力 VLAN にレイヤ 2 ACL だけが適用されます。また、非 IP および非 ARP パケットだけがフォールバック ブリッジド パケットとなります。

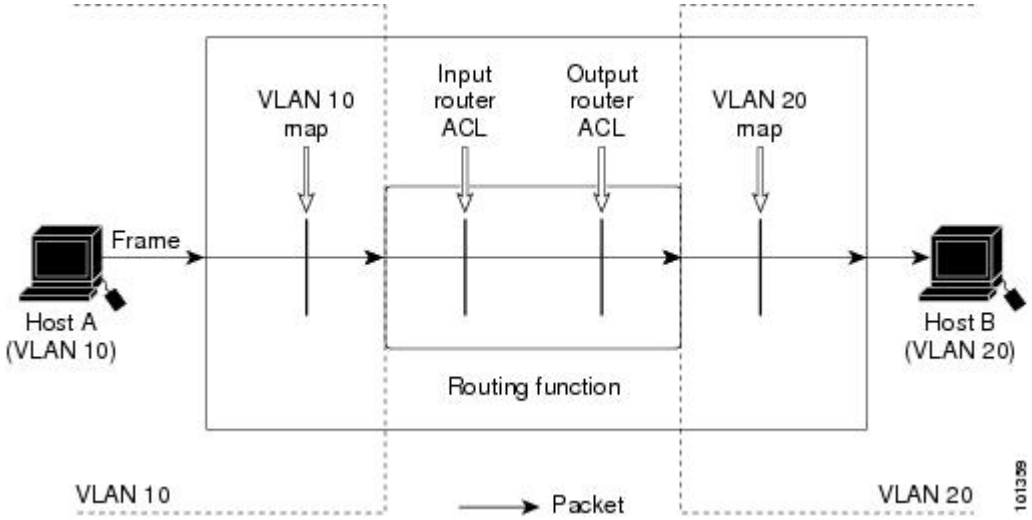


例：ACL およびルーテッド パケット

図 *101* : ルーテッド パケットへの *ACL* の適用

次の例に、ルーテッド パケットに ACL を適用する方法を示します。ACL は次の順番で適用されます。

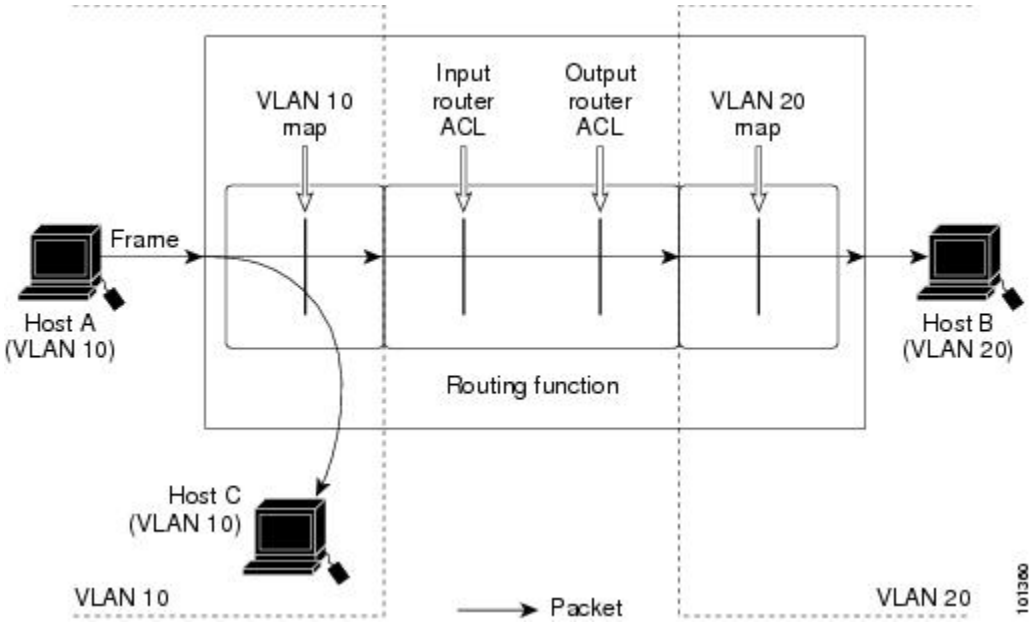
1. 入力 VLAN の VLAN マップ
2. 入力ルータ ACL
3. 出力ルータ ACL
4. 出力 VLAN の VLAN マップ



例：ACL およびマルチキャスト パケット

図 *102:* マルチキャスト パケットへの *ACL* の適用

次の例に、IPマルチキャスト用に複製されたパケットにACLを適用する方法を示します。ルーティングされるマルチキャストパケットには、2つの異なるフィルタが適用されます。1つは、宛先が入力VLAN内の他のポートである場合に使用され、もう1つは、宛先がパケットのルーティング先である別の VLAN 内にある場合に使用されます。パケットは複数の出力 VLAN にルーティングされる場合がありますが、この場合は宛先 VLAN ごとに異なるルータ出力 ACL およびVLANマップが適用されます。最終的に、パケットは一部の出力VLAN内で許可され、それ以外の VLAN で拒否されます。パケットのコピーが、許可された宛先に転送されます。ただし、入力 VLAN マップによってパケットがドロップされる場合、パケットのコピーは宛先に送信されません。



IPv4 アクセス コントロール リストに関する機能情報

リリース	機能情報
Cisco IOS Release 15.2(3)E	IPv4 アクセス コントロール リストは、パケット フィルタリングを実行して、ネットワークを介して移動するパケットとその場所を制御します。このような制御によって、ネットワーク トラフィックを制限し、ユーザーおよびデバイスのネットワークに対するアクセスを制限し、トラフィックがネットワークから外部に送信されるのを防ぐことで、セキュリティを実現します。この機能が導入されました。
Cisco IOS 15.2(2)E	アクセス コントロール エントリの非隣接ポートに対する名前付き ACL のサポート機能を使用すると、1 つのアクセス コントロール エントリで非隣接ポートを指定できるため、複数のエントリが同じ送信元アドレス、宛先アドレス、およびプロトコルを持ち、ポートのみが異なる場合に、アクセス コントロール リストで必要なエントリ数を大幅に減らすことができます。

リリース	機能情報
Cisco IOS 15.2(2)E	IP アクセス リスト エントリ シーケンス番号機能により、permit または deny ステートメントにシーケンス番号を適用したり、名前付き IP アクセス リストでそのようなステートメントを順序変更、追加、削除することができます。この機能により、IP アクセス リストを簡単に変更できるようになります。この機能が実装される前は、アクセス リストの最後にエントリを追加することしかできませんでした。そのため、末尾以外の任意の場所にステートメントを追加する必要があるときは、アクセスリスト全体を再設定する必要がありました。 次のコマンドが導入または変更されました。**deny (IP)**、**ip access-list resequence deny (IP)**、**permit (IP)**

第 64 章

IPv6 ACL

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

IPv6 ACL の概要

IP Version 6（IPv6）アクセス コントロール リスト（ACL）を作成し、それをインターフェイスに適用することによって、IPv6 トラフィックをフィルタリングできます。これは、IP Version 4（IPv4）の名前付き ACL を作成し、適用する方法と同じです。また、スイッチで IP ベースおよび LAN ベース フィーチャ セットが稼働している場合、入力ルータ ACL を作成し、それを適用してレイヤ 3 管理トラフィックをフィルタリングすることもできます。

スイッチは、次の 3 種類の IPv6 ACL をサポートします。

- IPv6 ルータ ACL は、ルーテッド ポート、スイッチ仮想インターフェイス（SVI）、またはレイヤ 3 EtherChannel に設定できるレイヤ 3 インターフェイスのアウトバウンド トラ

フィックまたはインバウンド トラフィックでサポートされます。IPv6 ルータ ACL は、ルーティングされる IPv6 パケットに対してだけ適用されます。

- IPv6 ポート ACL は、アウトバウンドおよびインバウンドのレイヤ 2 インターフェイスでサポートされます。IPv6 ポート ACL は、インターフェイスに着信するすべての IPv6 パケットに対して適用されます。
- VLAN ACL または VLAN マップは、VLAN 内のすべてのパケットのアクセスを制御します。VLAN マップを使用すると、同じ VLAN 内のデバイス間で転送されるトラフィックをフィルタリングできます。ACL VLANマップは、L2 VLANに適用されます。VLANマップは、IPv6 のレイヤ 3 アドレスに基づいてアクセス コントロールするように設定されています。イーサネットACEを使用するとMACアドレスにより、サポートされていないプロトコルがアクセスコントロールされます。VLANマップをVLANに適用すると、VLANに入るすべてのパケットが VLAN マップと照合されます。

スイッチは、IPv6 トラフィックの VLAN ACL（VLAN マップ）をサポートします。

1 つのインターフェイスに、IPv4 ACL および IPv6 ACL の両方を適用できます。IPv4 ACL の場合と同様に、IPv6 ポート ACL はルータ ACL よりも優先されます。

他の機能およびスイッチとの相互作用

- IPv6 ルータ ACL がパケットを拒否するよう設定されている場合、パケットはルーティングされません。パケットのコピーがインターネット制御メッセージ プロトコル（ICMP）キューに送信され、フレームに ICMP 到達不能メッセージが生成されます。
- ブリッジド フレームがポート ACL によってドロップされる場合、このフレームはブリッジングされません。
- IPv4 ACL および IPv6 ACL の両方を 1 つのスイッチまたはスイッチスタックに作成したり、同一インターフェイスに適用できます。各 ACL には一意の名前が必要です。設定済みの名前を使用しようとすると、エラー メッセージが表示されます。

  IPv4 ACL と IPv6 ACL の作成、および同一のレイヤ 2 インターフェイスまたはレイヤ 3 インターフェイスへの IPv4 ACL または IPv6 ACL の適用には、異なるコマンドを使用します。ACL を付加するのに誤ったコマンドを使用すると（例えば、IPv6 ACL の付加に IPv4 コマンドを使用するなど）、エラー メッセージが表示されます。
- MAC ACL を使用して、IPv6 フレームをフィルタリングできません。MAC ACL は非 IP フレームだけをフィルタリングできます。
- ハードウェア メモリに空きがない場合、パケットはインターフェイスでドロップされ、アンロードのエラー メッセージが記録されます。

IPv6 ACL の制限

IPv4 では、番号制の標準 IP ACL および拡張 IP ACL、名前付き IP ACL、および MAC ACL を設定できます。IPv6 がサポートするのは名前付き ACL だけです。

スイッチは Cisco IOS がサポートする IPv6 ACL の大部分をサポートしますが、一部例外もあります。

- スイッチは、**routing header**、および **undetermined-transport** というキーワードの照合をサポートしません。
- スイッチは、再帰 ACL（**reflect** キーワード）をサポートしません。
-
-
- このリリースは、IPv6 のポート ACL、ルータ ACL および VLAN ACL（VLAN マップ）をサポートしています。
-
- IPv6 の出力ルータ ACL および入力ポート ACL は、スイッチ スタックでだけサポートされています。スイッチは、コントロール プレーン（着信）IPv6 ACL だけをサポートします。
- スイッチは、IPv6 フレームに MAC ベース ACL を適用しません。
- ACL を設定する場合、ACL に入力されるキーワードには、それがプラットフォームでサポートされるかどうかにかかわらず、制限事項はありません。ハードウェア転送が必要なインターフェイス（物理ポートまたは SVI）に ACL を適用する場合、スイッチはインターフェイスで ACL がサポートされるかどうか判別します。サポートされない場合、ACL の付加は拒否されます。
- インターフェイスに適用される ACL に、サポートされないキーワードを持つアクセス コントロールエントリ（ACE）を追加しようとする場合、スイッチは現在インターフェイスに適用されている ACL に ACE が追加されるのを許可しません。

スイッチの IPv6 ACL には、次の特性があります。

- 分割フレーム（IPv4 では **fragments** キーワード）がサポートされます。
- IPv6 ACL では、IPv4 と同じ統計情報がサポートされます。
- スイッチのハードウェア スペースがなくなった場合、ACL に関連付けられたパケットはインターフェイスでドロップされます。
- ホップバイホップ オプションがあるルーテッドパケットまたはブリッジドパケットには、ソフトウェアで適用される IPv6 ACL が設定されます。
- ロギングは、ルータ ACL ではサポートされますが、ポート ACL ではサポートされません。

- スイッチは、プレフィックス長の最大範囲の IPv6 アドレス一致をサポートしません。

IPv6 ACL のデフォルト設定

デフォルトの IPv6 ACL 設定は次のとおりです。

```
Switch# show access-lists preauth_ipv6_acl
IPv6 access list preauth_ipv6_acl (per-user)
permit udp any any eq domain sequence 10
permit tcp any any eq domain sequence 20
permit icmp any any nd-ns sequence 30
permit icmp any any nd-na sequence 40
permit icmp any any router-solicitation sequence 50
permit icmp any any router-advertisement sequence 60
permit icmp any any redirect sequence 70
permit udp any eq 547 any eq 546 sequence 80
permit udp any eq 546 any eq 547 sequence 90
deny ipv6 any any sequence 100
```

IPv6 ACL の設定

IPv6 トラフィックをフィルタリングするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. {**ipv6 access-list** *list-name*
4. {**deny** | **permit**} protocol {*source-ipv6-prefix/ |prefix-length* |**any**| **host** *source-ipv6-address*} [operator [*port-number*]] { *destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]][**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]
5. {**deny** | **permit**} **tcp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [**operator** [**port-number**]] {*destination-ipv6- prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [**ack**] [**dscp** *value*] [**established**] [**fin**] [**log**] [**log-input**] [**neq** {**port** | protocol}] [**psh**] [**range** {**port** | protocol}] [**rst**] [**routing**] [**sequence** *value*] [**syn**] [**time-range** *name*] [**urg**]
6. {**deny** | **permit**} **udp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [operator [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [**dscp** *value*] [**log**] [**log-input**] [**neq** {*port* | *protocol*}] [**range** {*port* | *protocol*}] [**routing**] [**sequence** *value*] [**time-range** *name*]]
7. {**deny** | **permit**} **icmp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [operator [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [*icmp-type* [*icmp-code*] | icmp-message] [**dscp** *value*] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]
8. **end**
9. **show ipv6 access-list**
10. **show running-config**

11. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	{**ipv6 access-list** *list-name* 例： スイッチ(config)# **ipv6 access-list example_acl_list**	IPv6 ACL 名を定義し、IPv6 アクセス リスト コンフィギュレーション モードを開始します。
ステップ 4	{**deny** | **permit**} protocol {*source-ipv6-prefix/ |prefix-length* |**any**| **host** *source-ipv6-address*} [operator [*port-number*]] { *destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]][**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]	条件が一致した場合にパケットを拒否する場合は deny、許可する場合は permit を指定します。次に、条件について説明します。 • protocol には、IP の名前または番号を入力します。 **ahp**、**esp**、**icmp**、**ipv6**、**pcp**、**stcp**、**tcp**、**udp** または IPv6 プロトコル番号を表す 0 ～ 255 の整数を使用できます。 • *source-ipv6-prefix/prefix-length* または *destination-ipv6-prefix/ prefix-length* は、拒否条件または許可条件を設定する送信元または宛先 IPv6 ネットワークあるいはネットワーク クラスで、コロン区切りの 16 ビット値を使用した 16 進形式で指定します（RFC 2373 を参照）。 • IPv6 プレフィックス ::/0 の短縮形として、**any** を入力します。 • **host** *source-ipv6-address* または *destination-ipv6-address* には、拒否条件または許可条件を設定する送信元または宛先 IPv6 ホストアドレスを入力します。アドレスはコロン区切りの 16 ビット値を使用した 16 進形式で指定します。

	コマンドまたはアクション	目的
		• （任意）operator には、指定のプロトコルの送信元ポートまたは宛先ポートを比較するオペランドを指定します。オペランドには、**lt**（より小さい）、**gt**（より大きい）、**eq**（等しい）、**neq**（等しくない）、および**range**（包含範囲）があります。 *source-ipv6-prefix/prefix-length* 引数のあとの operator は、送信元ポートに一致する必要があります。*destination-ipv6- prefix/prefix-length* 引数のあとの operator は、宛先ポートに一致する必要があります。 • （任意）**port-number** は、0 ～ 65535 の 10 進数または TCP あるいは UDP ポートの名前です。TCP ポート名を使用できるのは、TCP のフィルタリング時だけです。UDP ポート名を使用できるのは、UDP のフィルタリング時だけです。 • （任意）**dscp** value を入力して、各 IPv6 パケットヘッダーの Traffic Class フィールド内のトラフィッククラス値と DiffServ コードポイント値を照合します。指定できる範囲は 0 ～ 63 です。 • （任意）**fragments** を入力して、先頭ではないフラグメントを確認します。このキーワードが表示されるのは、プロトコルが ipv6 の場合だけです。 • （任意）**log** を指定すると、エントリと一致するパケットに関するログメッセージがコンソールに送信されます。**log-input** を指定すると、ログエントリに入力インターフェイスが追加されます。ロギングはルータ ACL でだけサポートされます。 • （任意）**routing** を入力して、IPv6 パケットのルーティングを指定します。 • （任意）**sequence** *value* を入力して、アクセスリスト ステートメントのシーケンス番号を指定します。指定できる範囲は 1 ～ 4,294,967,295 です。

	コマンドまたはアクション	目的
		• （任意）**time-range** name を入力して、拒否または許可ステートメントに適用される時間の範囲を指定します。
ステップ 5	{**deny** | **permit**} **tcp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [**operator** [**port-number**]] {*destination-ipv6- prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [**ack**] [**dscp** *value*] [**established**] [**fin**] [**log**] [**log-input**] [**neq** {**port** | protocol}] [**psh**] [**range** {**port** | protocol}] [**rst**] [**routing**] [**sequence** *value*] [**syn**] [**time-range** *name*] [**urg**]	（任意）TCP アクセス リストおよびアクセス条件を定義します。 TCP の場合は **tcp** を入力します。パラメータはステップ 3a で説明されているパラメータと同じですが、次に示すオプションのパラメータが追加されています。 • **ack**：確認応答（ACK）ビットセット。 • **established**：確立された接続。TCP データグラムに ACK または RST ビットが設定されている場合、照合が行われます。 • **fin**：終了ビットセット。送信者からのデータはそれ以上ありません。 • **neq** {*port* | **protocol**}：所定のポート番号上にないパケットだけを照合します。 • **psh**：プッシュ機能ビットセット • **range** {*port* | **protocol**}：ポート番号の範囲内のパケットだけを照合します。 • **rst**：リセットビットセット • **syn**：同期ビットセット • **urg**：緊急ポインタビットセット
ステップ 6	{**deny** | **permit**} **udp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [operator [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [**dscp** *value*] [**log**] [**log-input**] [**neq** {*port* | *protocol*}] [**range** {*port* | *protocol*}] [**routing**] [**sequence** *value*] [**time-range** *name*]]	（任意）UDP アクセス リストおよびアクセス条件を定義します。 ユーザ データグラム プロトコルの場合は、**udp** を入力します。UDP パラメータは TCP に関して説明されているパラメータと同じです。ただし、[operator [*port*]] のポート番号またはポート名は、UDP ポートの番号または名前でなければなりません。UDP の場合、established パラメータは無効です。
ステップ 7	{**deny** | **permit**} **icmp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [operator [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host**	（任意）ICMP アクセス リストおよびアクセス条件を定義します。

	コマンドまたはアクション	目的
	destination-ipv6-address} [operator [*port-number*]] [*icmp-type* [*icmp-code*] | icmp-message] [**dscp** *value*] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]	インターネット制御メッセージプロトコルの場合は、**icmp** を入力します。ICMP パラメータはステップ 1 の IP プロトコルの説明にあるパラメータとほとんど同じですが、ICMP メッセージタイプおよびコード パラメータが追加されています。オプションのキーワードの意味は次のとおりです。 • *icmp-type*：ICMP メッセージタイプでフィルタリングする場合に入力します。指定できる値の範囲は、0 ～ 255 です。 • *icmp-code*：ICMP パケットを ICMP メッセージコードタイプでフィルタリングする場合に入力します。指定できる値の範囲は、0 ～ 255 です。 • *icmp-message*：ICMP パケットを ICMP メッセージタイプ名または ICMP メッセージタイプとコード名でフィルタリングする場合に入力します。ICMP メッセージのタイプ名およびコード名のリストについては、?キーを使用するか、またはこのリリースのコマンド リファレンスを参照してください。
ステップ 8	**end**	特権 EXEC モードに戻ります。
ステップ 9	**show ipv6 access-list**	アクセス リストの設定を確認します。
ステップ 10	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 11	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

インターフェイスへの IPv6 ACL の付加

レイヤ 3 インターフェイスで発信または着信トラフィックに ACL を、あるいはレイヤ 2 インターフェイスで着信トラフィックに を適用できます。レイヤ 3 インターフェイスで着信トラフィックにだけ ACL を適用できます。

インターフェイスへのアクセスを制御するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no switchport**
5. **ipv6 address** *ipv6-address*
6. **ipv6 traffic-filter** *access-list-name* {**in** | **out**}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id*	アクセスリストを適用するレイヤ 2 インターフェイス（ポート ACL 用）またはレイヤ 3 インターフェイス（ルータ ACL 用）を特定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**no switchport**	ルータ ACL を適用する場合は、これによってインターフェイスがレイヤ 2 モード（デフォルト）からレイヤ 3 モードに変化します。
ステップ 5	**ipv6 address** *ipv6-address*	レイヤ 3 インターフェイス（ルータ ACL 用）で IPv6 アドレスを設定します。

	コマンドまたはアクション	目的
ステップ 6	**ipv6 traffic-filter** *access-list-name* {**in** | **out**}	インターフェイスの着信トラフィックまたは発信トラフィックにアクセスリストを適用します。 （注） out キーワードはレイヤ 2 インターフェイス（ポート ACL）ではサポートされません。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 ACL のモニタリング

次の表に示された 1 つまたは複数の特権 EXEC コマンドを使用して、設定済みのすべてのアクセス リスト、すべての IPv6 アクセス リスト、または特定のアクセス リストに関する情報を表示できます。

表 *134 : show ACL* コマンド

コマンド	目的
show access-lists	スイッチに設定されたすべてのアクセス リストを表示します。
show ipv6 access-list [*access-list-name*]	設定済みのすべての IPv6 アクセス リストまたは名前で指定されたアクセス リストを表示します。
show vlan access-map [*map-name*]	VLAN アクセス マップ設定を表示します。
show vlan filter [**access-map** *access-map* | **vlan** *vlan-id*]	VACL と VLAN 間のマッピングを表示します。

次に、show access-lists 特権 EXEC コマンドの出力例を示します。出力には、スイッチまたはスイッチ スタックに設定済みのすべてのアクセス リストが表示されます。

```
Switch # show access-lists
Extended IP access list hello
    10 permit ip any any
IPv6 access list ipv6
    permit ipv6 any any sequence 10
```

次に、**show ipv6 access-lists** 特権 EXEC コマンドの出力例を示します。出力には、スイッチまたはスイッチ スタックに設定済みの IPv6 アクセス リストだけが表示されます。

```
Switch# show ipv6 access-list
IPv6 access list inbound
    permit tcp any any eq bgp (8 matches) sequence 10
    permit tcp any any eq telnet (15 matches) sequence 20
    permit udp any any sequence 30
IPv6 access list outbound
    deny udp any any sequence 10
    deny tcp any any eq telnet sequence 20
```

次に、show vlan access-map 特権 EXEC コマンドの出力例を示します。出力には、VLAN アクセス マップ情報が表示されます。

```
Switch# show vlan access-map
Vlan access-map "m1"  10
  Match clauses:
    ipv6 address: ip2
  Action: drop
```

第 65 章

DHCP の設定

DHCP の制限

次のシナリオはサポートされていません。

非 DHCP スヌーピング VLAN、および非 DHCP スヌーピング VLAN の SVI がデバイスに設定されています。非 DHCP スヌーピング VLAN の SVI は no shutdown のステータスで設定されます。このシナリオでは、非 DHCP スヌーピング VLAN の DHCP パケットは信頼できるポートに転送されません。

非 DHCP スヌーピング VLAN の SVI が設定されていないか、shutdown ステータスで設定されている場合、DHCP パケットは信頼できるポートに転送され、DHCP クライアントは DHCP サーバーから IP アドレスを取得できます。

DHCP に関する情報

DHCP サーバ

DHCP サーバーは、スイッチまたはルータ上の指定されたアドレス プールから DHCP クライアントに IP アドレスを割り当て、それらのアドレスを管理します。DHCP サーバーがそのデータベースから要求された設定パラメータを取得して DHCP クライアントに渡すことができない場合は、ネットワーク管理者が定義した 1 つまたは複数のセカンダリ DHCP サーバーに要求を転送します。スイッチは、DHCP サーバーとして機能できます。

DHCP リレー エージェント

DHCP リレー エージェントは、クライアントとサーバの間で DHCP パケットを転送するレイヤ3デバイスです。リレーエージェントは、同じ物理サブネット上にないクライアントとサーバの間で要求および応答を転送します。リレー エージェントによる転送は、IP データグラムをネットワーク間で透過的に交換するレイヤ 2 での通常の転送とは異なります。リレー エージェントは、DHCP メッセージを受け取ると、新しい DHCP メッセージを生成して、出力インターフェイス上で送信します。

DHCP スヌーピング

DHCP スヌーピングは、信頼できない DHCP メッセージのフィルタリングと DHCP スヌーピング バインディング データベース（DHCP スヌーピング バインディング テーブルとも呼ばれる）の作成および管理によってネットワーク セキュリティを確保する DHCP セキュリティ機能です。

DHCP スヌーピングは、信頼できないホストと DHCP サーバの間でファイアウォールに似た役割を果たします。DHCP スヌーピングを使用することにより、エンドユーザーに接続された信頼できないインターフェイスと DHCP サーバーまたは別のスイッチに接続された信頼できるインターフェイスを区別できます。

（注） DHCP スヌーピングを正しく機能させるためには、すべての DHCP サーバーを信頼できるインターフェイス経由でスイッチに接続する必要があります。

信頼できない DHCP メッセージとは、信頼できないインターフェイス経由で送信されたメッセージのことです。デフォルトでは、スイッチはすべてのインターフェイスを信頼できないものと見なします。そのため、スイッチはいくつかのインターフェイスを信頼して DHCP スヌーピングを使用するように設定する必要があります。サービス プロバイダ環境で DHCP スヌーピングを使用する場合は、カスタマーのスイッチなど、サービス プロバイダ ネットワーク内には存在しないデバイスから送信されたメッセージが信頼できないメッセージとなります。不明なデバイスから送信されたメッセージは、トラフィック攻撃の原因になりうるため、信頼できません。

DHCP スヌーピング バインディング データベースには、MAC アドレス、IP アドレス、リース期間、バインディングの種類、VLAN 番号、およびスイッチの信頼できないローカルインターフェイスのインターフェイス情報が含まれています。このデータベースには、信頼できるインターフェイスに接続されたホストの情報はありません。

（注） DHCP スヌーピングを設定し、インターフェイスで **ip verify source prot-security** コマンドを使用して未認可の IP アドレスをブロックする場合は、**switchport port-security** コマンドも設定する必要があります。

サービス プロバイダー ネットワークでは、信頼できるインターフェイスとして設定できるものの例として、同じネットワーク内のデバイスのポートに接続されたインターフェイスがあります。信頼できないインターフェイスには、ネットワーク内の信頼できないインターフェイスまたはネットワークに属さないデバイスのインターフェイスに接続されたインターフェイスがあります。

スイッチが信頼できないインターフェイスでパケットを受信し、そのインターフェイスが属している VLAN で DHCP スヌーピングがイネーブルに設定されている場合、スイッチは送信元 MAC アドレスと DHCP クライアントのハードウェア アドレスを比較します。アドレスが一致した場合（デフォルト）、スイッチはパケットを転送します。アドレスが一致しない場合、スイッチはパケットをドロップします。

スイッチは、次のいずれかの状況が発生した場合に DHCP パケットをドロップします。

- DHCPOFFER パケット、DHCPACK パケット、DHCPNAK パケット、DHCPLEASEQUERY パケットなど、DHCP サーバからのパケットがネットワークまたはファイアウォールの外側から着信した。
- パケットが信頼できないインターフェイスに着信し、送信元 MAC アドレスと DHCP クライアントのハードウェア アドレスが一致しない。
- スイッチが DHCPRELEASE または DHCPDECLINE ブロードキャスト メッセージを受信し、その MAC アドレスは DHCP スヌーピング バインディング データベースに含まれているが、バインディングデータベース内のインターフェイス情報がメッセージを受信したインターフェイスと一致しない。
- DHCP リレー エージェントが 0.0.0.0 以外のリレー エージェント IP アドレスを含む DHCP パケットを転送し、Option 82 情報が含まれないパケットを信頼できないポートに転送する。

DHCP スヌーピングをサポートする集約スイッチであり、DHCP オプション 82 情報を挿入するエッジ スイッチに接続されているスイッチは、オプション 82 情報を含むパケットが信頼できないインターフェイスに着信した場合、それらのパケットをドロップします。DHCP スヌーピングがイネーブルに設定されている場合に、パケットが信頼できるポートに着信しても、集約スイッチは接続されたデバイスの DHCP スヌーピング バインディングを認識せず、完全な DHCP スヌーピング バインディング データベースを作成できません。

集約スイッチを信頼できないインターフェイス経由でエッジスイッチに接続できる場合、**ip dhcp snooping information option allow-untrusted** グローバル コンフィギュレーション コマンドを入力すると、集約スイッチはエッジスイッチによって挿入されたオプション 82 情報を含むパケットを受け入れます。集約スイッチは、信頼できないスイッチインターフェイスを介して接続されたホストのバインディングを認識します。集約スイッチで、ダイナミック ARP インスペクションや IP ソース ガードなど、DHCP セキュリティ機能をイネーブルに設定することもできますが、その場合でもスイッチは Option 82 情報を含むパケットをホストが接続されている信頼できない入力インターフェイスで受信します。集約スイッチ上のエッジスイッチとの接続ポートは、信頼できるインターフェイスとして設定する必要があります。

通常、ワイヤレス クライアントにパケットをブロードキャストするのは望ましくありません。したがって、DHCP スヌーピングは、宛先ブロードキャスト MAC アドレス（ffff.ffff.ffff）を

サーバからワイヤレス クライアントに送信される DHCP パケットのユニキャスト MAC アドレスに置き換えます。ユニキャスト MAC アドレスは DHCP ペイロード内の CHADDR フィールドから取得されます。この処理は、DHCP OFFER、DHCP ACK および DHCP NACK メッセージなどのクライアント パケットにサーバ用に適用されます。**ip dhcp snooping wireless bootp-broadcast enable**を使用して、この動作を元に戻すことができます。ワイヤレス BOOTP ブロードキャストがイネーブルの場合、サーバからのブロードキャスト DHCP パケットは、宛先 MAC アドレスを変更せずにワイヤレス クライアントに転送されます。

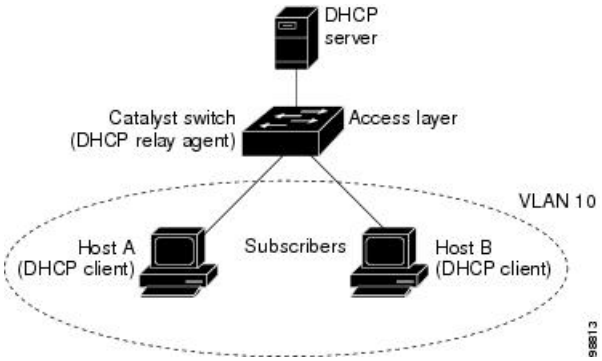
オプション 82 データ挿入

住宅地域にあるメトロポリタン イーサネット アクセス環境では、DHCP は多数の加入者に対し、IP アドレスの割り当てを一元的に管理できます。スイッチで DHCP スヌーピングの Option 82 機能をイネーブルにすると、加入者装置は MAC アドレスだけでなく、その装置をネットワークに接続するスイッチ ポートによっても識別されます。サブスクライバ LAN 上の複数のホストをアクセス スイッチの同じポートに接続できます。これらのホストは一意に識別されます。

（注） DHCP オプション 82 機能は、DHCP スヌーピングがグローバルに有効であり、オプション 82 を使用する加入者装置が割り当てられた VLAN で有効である場合に限りサポートされます。

次の図に、一元的な DHCP サーバーがアクセス レイヤのスイッチに接続された加入者に IP アドレスを割り当てるメトロポリタン イーサネット ネットワークを示します。DHCP クライアントとそれらに関連付けられた DHCP サーバは同じ IP ネットワークまたはサブネット内に存在しないため、DHCP リレー エージェント（Catalyst スイッチ）にヘルパー アドレスを設定することにより、ブロードキャスト転送をイネーブルにし、クライアントとサーバ間で DHCP メッセージを転送します。

図 *103 :* メトロポリタン イーサネット ネットワークにおける *DHCP* リレー エージェント



スイッチで DHCP スヌーピング情報 オプション 82 を有効にすると、次のイベントがこの順序で発生します。

- ホスト（DHCP クライアント）は DHCP 要求を生成し、これをネットワーク上にブロードキャストします。
- スイッチは、この DHCP 要求を受信すると、パケットに Option 82 情報を追加します。デフォルトでは、リモート ID サブオプションがスイッチの MAC アドレスで、回線 ID サブオプションはパケットを受信するポート ID（**vlan-mod-port**）です。リモート ID および回線 ID は設定できます。
- リレー エージェントの IP アドレスが設定されている場合、スイッチはこの IP アドレスを DHCP パケットに追加します。
- スイッチは、オプション 82 フィールドを含む DHCP 要求を DHCP サーバーに転送します。
- DHCP サーバはこのパケットを受信します。Option 82 に対応しているサーバであれば、リモート ID と回線 ID のいずれか一方または両方を使用して、IP アドレスを割り当てたり、1 つのリモート ID または回線 ID に割り当てることができる IP アドレスの数を制限するようなポリシーを実装したりできます。次に DHCP サーバは、DHCP 応答内にオプション 82 フィールドをエコーします。
- スイッチによって要求がサーバーにリレーされた場合、DHCP サーバーは応答をスイッチにユニキャストします。スイッチは、リモート ID フィールドと、場合によっては回線 ID フィールドを調べ、Option 82 データが挿入済みであることを確認します。スイッチは Option 82 フィールドを削除してから、DHCP 要求を送信した DHCP クライアントに接続するスイッチ ポートにパケットを転送します。

デフォルトのサブオプション設定では、前述のイベントのシーケンスが発生すると、次のフィールドの値は変化しません（図「サブオプションのパケット形式」を参照）。

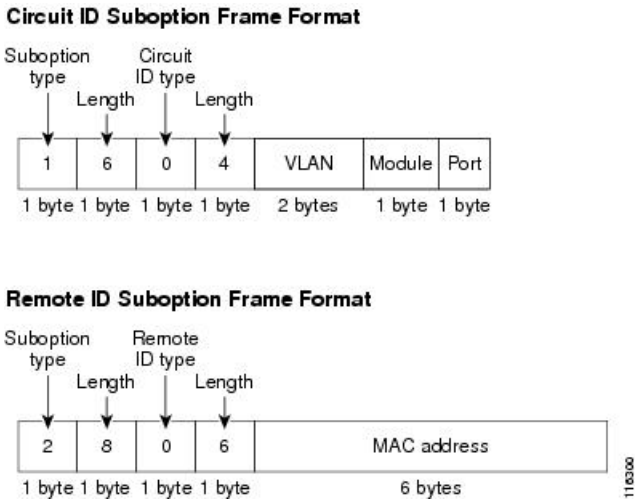
- 回線 ID サブオプション フィールド
  - サブオプション タイプ
  - サブオプション タイプの長さ
  - 回線 ID タイプ
  - 回線 ID タイプの長さ
- リモート ID サブオプション フィールド
  - サブオプション タイプ
  - サブオプション タイプの長さ
  - リモート ID タイプ
  - リモート ID タイプの長さ

回線 ID サブオプションのポート フィールドでは、ポート番号が 3 から始まります。たとえば、24 個の 10/100/1000 ポートおよび 4 つの Small Form-Factor Pluggable（SFP）モジュール ス

ロットを搭載するスイッチでは、ポート 3 がギガビット イーサネット 1/0/1 ポート、ポート 4 がギガビット イーサネット 1/0/2 ポートとなり、以降同様に続きます。ポート 27 は SFP モジュール スロットのギガビット イーサネット 1/0/25 となり、以降同様に続きます。

図「サブオプションのパケット形式」に、デフォルトのサブオプション設定が使用されている場合のリモート ID サブオプションおよび回線 ID サブオプションのパケット形式を示します。回線 ID サブオプションでは、モジュール番号は、スタックにあるスイッチ番号に対応します。スイッチがこれらのパケット形式を使用するのは、DHCP スヌーピングをグローバルに有効にし、ip dhcp snooping information option グローバル コンフィギュレーション コマンドを入力した場合です。

図 *104* : サブオプションのパケット形式

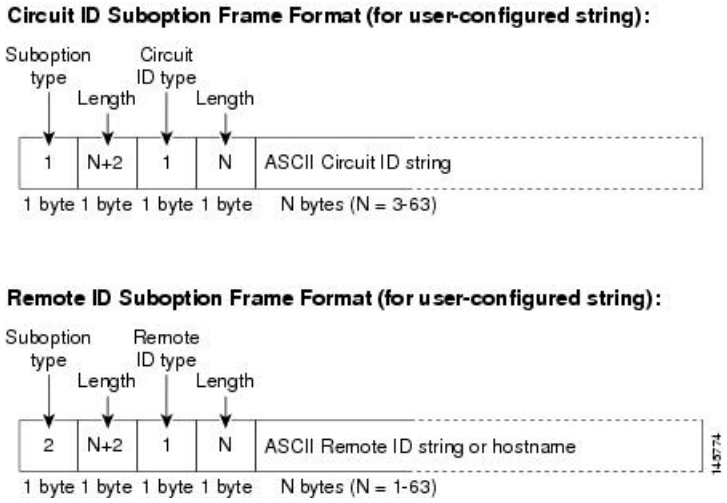


図「ユーザー設定のサブオプションのパケット形式」は、ユーザー設定のリモート ID サブオプション、および回線 ID サブオプションのパケット形式を示しています。スイッチでは、DHCP スヌーピングをグローバルにイネーブルにし、**ip dhcp snooping information option format remote-id** グローバル コンフィギュレーション コマンド、および **ip dhcp snooping vlan information option format-type circuit-id string** インターフェイス コンフィギュレーション コマンドを入力した場合に、これらのパケット形式が使用されます。

パケットでは、リモート ID および回線 ID サブオプションを次のように設定した場合、これらのフィールドの値がデフォルト値から変更されます。

- 回線 ID サブオプション フィールド
  - 回線 ID タイプが 1 である。
  - 設定した文字列の長さに応じて、長さの値が変化する。
- リモート ID サブオプション フィールド
  - リモート ID タイプが 1 である。
  - 設定した文字列の長さに応じて、長さの値が変化する。

図 *105 :* ユーザ設定のサブオプションのパケット形式



Cisco IOS DHCP サーバ データベース

DHCP ベースの自動設定プロセスの間、指定 DHCP サーバは Cisco IOS DHCP サーバ データベースを使用します。これには IP アドレス、アドレス バインディング、およびブート ファイルなどの設定パラメータが含まれます。

アドレス バインディングは、Cisco IOS DHCP サーバ データベース内のホストの IP アドレスおよびMAC アドレス間のマッピングです。クライアント IP アドレスを手動で割り当てること、または、DHCP サーバが DHCP アドレス プールから IP アドレスを割り当てることが可能です。手動および自動アドレス バインディングの詳細については、『*Cisco IOS IP Configuration Guide, Release 12.4*』の「Configuring DHCP」の章を参照してください。

Cisco IOS DHCP サーバー データベースをイネーブルにして設定する手順については、『*Cisco IOS IP Configuration Guide, Release 12.4*』の「Configuring DHCP」の章にある「DHCP Configuration Task List」の項を参照してください。

DHCP スヌーピング バインディング データベース

DHCP スヌーピングをイネーブルにすると、スイッチは信頼できないインターフェイスに関する情報を DHCP スヌーピング バインディング データベースに保存します。データベースには、64,000 のバインディングを含めることができます。

各データベース エントリ（バインディング）は、IP アドレス、それに関連付けられた MAC アドレス、リース期間（16 進形式）、バインディングが適用されるインターフェイス、およびインターフェイスが属する VLAN で構成されます。データベース エージェントは、設定された場所のファイルにバインディングを保存します。各エントリの末尾にあるチェックサムは、ファイルの先頭のバイトを含め、エントリに関連付けられたすべてのバイトを対象として計算されます。各エントリは、まず 72 バイトのデータがあり、その後に 1 つのスペースとチェックサム値が続きます。

スイッチのリロード後もバインディングを保持するには、DHCP スヌーピング データベース エージェントを使用する必要があります。エージェントがディセーブルで、ダイナミック ARP インスペクションまたは IP ソース ガードがイネーブルにされ、DHCP スヌーピング バインディング データベースがダイナミック バインディングされている場合、スイッチは接続を切断されます。このエージェントがディセーブルで、DHCP スヌーピングだけがイネーブルである場合、スイッチの接続は切断されませんが、DHCP スヌーピングは DHCP スプーフィング攻撃を防止できないことがあります。

リロードすると、スイッチはバインディング ファイルを読み込み、DHCP スヌーピング バインディング データベースを作成します。スイッチは、データベースに変更が加えられたときにはバインディング ファイルを更新します。

スイッチは、新しいバインディングを認識するか、バインディングを失うと、ただちにデータベース内のエントリを更新します。スイッチはバインディング ファイル内のエントリも更新します。バインディング ファイルの更新頻度は設定可能な遅延時間によって決まり、更新はバッチ処理されます。ファイルが指定された時間内（書き込み遅延およびキャンセルタイムアウトの値によって設定される）に更新されない場合、更新は停止します。

バインディングが含まれるファイルの形式は次のとおりです。

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-..-n>
END
```

このファイルの各エントリにはチェックサム値を示すタグが付けられます。スイッチは、ファイルを読み取るときに、このチェックサムを使用してエントリを検証します。最初の行の initial-checksum エントリは、最新のファイル更新に関連するエントリを以前のファイル更新に関連するエントリと区別します。

次に、バインディング ファイルの例を示します。

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Gi1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Gi1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Gi1/0/4 584a38f0
END
```

スイッチが起動し、計算されたチェックサム値が保存されているチェックサム値と一致した場合、スイッチはバインディング ファイルのエントリを読み取り、バインディングを DHCP スヌーピング バインディング データベースに追加します。次のいずれかの状況が発生した場合、スイッチはエントリを無視します。

- スイッチがエントリを読み取り、計算されたチェックサム値が保存されているチェックサム値と一致しない。この場合、そのエントリとそれ以降のエントリは無視されます。
- エントリに含まれているリース期間が終了している（スイッチはリース期間の終了時にバインディング エントリを削除しないことがある）。
- エントリに含まれるインターフェイスが現在はシステムに存在しない。
- インターフェイスがルーテッド インターフェイスまたは DHCP スヌーピングにおける信頼できるインターフェイスである。

DHCP 機能の設定方法

DHCP スヌーピングのデフォルト設定

表 *135 : DHCP* のデフォルト設定

機能	デフォルト設定
DHCP サーバ	Cisco IOS ソフトウェアではイネーブル、設定が必要[13]
DHCP リレー エージェント	イネーブル[14]
DHCP パケット転送アドレス	未設定
リレー エージェント情報の確認	イネーブル（無効なメッセージは廃棄）
DHCP リレー エージェント転送ポリシー	既存のリレー エージェント情報を置換。
DHCP スヌーピングをグローバルにイネーブル	ディセーブル
DHCP スヌーピング情報オプション	イネーブル
パケットを信頼できない入力インターフェイスで受け取る DHCP スヌーピング オプション[15]	ディセーブル
DHCP スヌーピング レート制限	未設定
DHCP スヌーピング信頼状態	信頼できない
DHCP スヌーピング VLAN	ディセーブル
DHCP スヌーピングの MAC アドレス検証	イネーブル

機能	デフォルト設定
Cisco IOS DHCP サーバ バインディング データベース	Cisco IOS ソフトウェアではイネーブル、設定が必要。 （注） スイッチは、DHCP サーバとして設定されているデバイスからだけ、ネットワーク アドレスおよび設定パラメータを取得します。
DHCP スヌーピング バインディング データベース エージェント	Cisco IOS ソフトウェアではイネーブル、設定が必要。この機能は宛先が設定されている場合に限り有効。

[13] スイッチは、DHCP サーバとして設定されている場合に限り DHCP 要求に応答します。
[14] スイッチは、DHCP サーバの IP アドレスが DHCP クライアントの SVI に設定されている場合に限り DHCP パケットをリレーします。
[15] この機能は、スイッチがエッジスイッチによってオプション 82 情報が挿入されたパケットを受信する集約スイッチである場合に使用します。

DHCP スヌーピング設定時の注意事項

- スイッチポートが DHCP サーバに接続されている場合は、**ip dhcp snooping trust interface** コンフィギュレーション コマンドを入力して、ポートを信頼できるポートとして設定してください。
- スイッチポートが DHCP クライアントに接続されている場合は、**no ip dhcp snooping trust** インターフェイス コンフィギュレーション コマンドを入力して、ポートを信頼できないポートとして設定してください。
- **show ip dhcp snooping statistics** ユーザー EXEC コマンドを入力して DHCP スヌーピング統計情報を表示したり、**clear ip dhcp snooping statistics** 特権 EXEC コマンドを入力してスヌーピング統計情報をクリアしたりできるようになりました。

DHCP サーバの設定

スイッチは、DHCP サーバーとして機能できます。

スイッチを DHCP サーバーとして設定するときの手順については、『*Cisco IOS IP Configuration Guide, Release 12.4*』の「IP Addressing and Services」の項の「Configuring DHCP」を参照してください。

DHCP リレー エージェントの設定

スイッチ上で DHCP リレー エージェントをイネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **service dhcp**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**service dhcp** 例： `スイッチ(config)# service dhcp`	スイッチ上でDHCPサーバおよびDHCPリレーエージェントをイネーブルにします。デフォルトでは、この機能はイネーブルです。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

- リレー エージェント情報のチェック（検証）
- リレー エージェント転送ポリシーの設定

パケット転送アドレスの指定

DHCP サーバーおよび DHCP クライアントが異なるネットワークまたはサブネットにある場合、スイッチを **ip helper-address** *address* インターフェイス コンフィギュレーション コマンドで設定する必要があります。一般的なルールは、クライアントに最も近いレイヤ 3 インターフェイス上にコマンドを設定することです。**ip helper-address** コマンドで使用されているアドレスは、特定の DHCP サーバ IP アドレスか、または他の DHCP サーバが宛先ネットワークセグメントにある場合はネットワークアドレスにすることができます。ネットワーク アドレスを使用することで、どの DHCP サーバも要求に応答できるようになります。

パケット転送アドレスを指定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface vlan** *vlan-id*
4. **ip address** *ip-address subnet-mask*
5. **ip helper-address** *address*
6. **end**
7. 次のいずれかを使用します。
   - **interface range** *port-range*
   - **interface** *interface-id*
8. **switchport mode access**
9. **switchport access vlan** *vlan-id*
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface vlan** *vlan-id* 例： `スイッチ(config)# interface vlan 1`	VLAN ID を入力してスイッチ仮想インターフェイスを作成し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip address** *ip-address subnet-mask* 例： `スイッチ(config-if)# ip address 192.108.1.27 255.255.255.0`	インターフェイスに IP アドレスおよび IP サブネットを設定します。
ステップ 5	**ip helper-address** *address* 例： `スイッチ(config-if)# ip helper-address 172.16.1.2`	DHCP パケット転送アドレスを指定します。 ヘルパー アドレスは特定の DHCP サーバ アドレスにするか、他の DHCP サーバが宛先ネットワーク セグメントにある場合は、ネットワーク アドレスにすることができます。ネットワーク アドレスを使用することで、他のサーバも DHCP 要求に応答できるようになります。 複数のサーバがある場合、各サーバに 1 つのヘルパー アドレスを設定できます。
ステップ 6	**end** 例： `スイッチ(config-if)# end`	グローバル コンフィギュレーション モードに戻ります。
ステップ 7	次のいずれかを使用します。 • **interface range** *port-range* • **interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/2`	DHCP クライアントに接続されている複数の物理ポートを設定し、インターフェイス範囲コンフィギュレーション モードを開始します。 または DHCP クライアントに接続されている単一の物理ポートを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8	**switchport mode access** 例：	ポートの VLAN メンバーシップ モードを定義します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# switchport mode access	
ステップ 9	**switchport access vlan** *vlan-id* 例： スイッチ(config-if)# switchport access vlan 1	ステップ 2 で設定したのと同じ VLAN をポートに割り当てます。
ステップ 10	**end** 例： スイッチ(config-if)# end	特権 EXEC モードに戻ります。
ステップ 11	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 12	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

DHCP スヌーピングおよびオプション 82 を設定するための前提条件

DHCP スヌーピングおよびオプション 82 の前提条件は次のとおりです。

- DHCP スヌーピングは、スイッチ上でグローバルにイネーブルにする必要があります。
- スイッチ上で DHCP スヌーピングをグローバルにイネーブルにする前に、DHCP サーバーや DHCP リレー エージェントとして機能するデバイスが設定され、イネーブルになっていることを確認してください。
- スイッチを DHCP 要求に応答するようにする場合は、DHCP サーバーとして設定する必要があります。
- スイッチで DHCP スヌーピング情報オプションを設定する前に、DHCP サーバーとして機能するデバイスを設定してください。DHCP サーバーが割り当てたり除外したりできる IP アドレスを指定するか、またはそれらのデバイスの DHCP オプションを設定する必要があります。
- DHCP スヌーピングを正しく機能させるためには、すべての DHCP サーバーを信頼できるインターフェイス経由でスイッチに接続する必要があります。サービス プロバイダ ネットワークでは、同じネットワーク内のデバイスのポートに接続されたインターフェイスが信頼できるインターフェイスとなります。

- DHCP スヌーピングで Cisco IOS DHCP サーバー バインディング データベースを使用するには、Cisco IOS DHCP サーバー バインディング データベースを使用するようにスイッチを設定する必要があります。
- 信頼できない入力でパケットを受け入れる DHCP スヌーピング オプションを使用するには、スイッチがエッジ スイッチからオプション 82 情報を含むパケットを受信する集約スイッチである必要があります。
- 次の前提条件が DHCP スヌーピング バインディング データベースの設定に適用されます。
  - DHCP スヌーピング用にスイッチを使用するには、DHCP スヌーピング バインディング データベースで宛先を設定する必要があります。
  - NVRAM とフラッシュ メモリは、いずれも記憶容量が限られているため、バインディング ファイルを TFTP サーバーに保存することを推奨します。
  - ネットワーク ベースの URL（TFTP や FTP など）については、スイッチがバインディングをその URL のバインディング ファイルに初めて書き込む前に、設定された URL に空のファイルを作成する必要があります。空のファイルをサーバ上に作成する必要があるかどうかについては、TFTP サーバのマニュアルを参照してください。TFTP サーバによっては、そのように設定できないことがあります。
  - データベースに正しいリース期間が記録されるように、ネットワーク タイム プロトコル（NTP）をイネーブルにし、設定することを推奨します。
  - NTP が設定されている場合、スイッチのシステム クロックが NTP と同期化されたときにだけ、スイッチがバインディングの変更内容をバインディング ファイルに書き込みます。
- スイッチで DHCP リレー エージェントを設定する前に、DHCP サーバーとして機能するデバイスを設定してください。DHCP サーバーが割り当てたり除外したりできる IP アドレスを指定するか、デバイスの DHCP オプションを設定するか、または DHCP データベース エージェントをセットアップする必要があります。
- スイッチが DHCP パケットをリレーするようにする場合は、DHCP サーバーの IP アドレスは DHCP クライアントのスイッチ仮想インターフェイス（SVI）に設定する必要があります。
- スイッチポートが DHCP サーバに接続されている場合は、**ip dhcp snooping trust interface** コンフィギュレーション コマンドを入力して、ポートを信頼できるポートとして設定してください。
- スイッチポートが DHCP クライアントに接続されている場合は、**no ip dhcp snooping trust** インターフェイス コンフィギュレーション コマンドを入力して、ポートを信頼できないポートとして設定してください。

DHCP スヌーピングおよび Option 82 のイネーブル化

スイッチ上で DHCP スヌーピングをイネーブルにするには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip dhcp snooping**
4. **ip dhcp snooping vlan** *vlan-range*
5. **ip dhcp snooping information option**
6. **ip dhcp snooping information option format remote-id** [**string** *ASCII-string* | **hostname**]
7. **ip dhcp snooping information option allow-untrusted**
8. **interface** *interface-id*
9. **ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] **string** *ASCII-string*
10. **ip dhcp snooping trust**
11. **ip dhcp snooping limit rate** *rate*
12. **exit**
13. **ip dhcp snooping verify mac-address**
14. **end**
15. **show running-config**
16. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip dhcp snooping** 例： `スイッチ(config)# ip dhcp snooping`	DHCP スヌーピングをグローバルにイネーブル化します。
ステップ 4	**ip dhcp snooping vlan** *vlan-range* 例： `スイッチ(config)# ip dhcp snooping vlan 10`	VLAN または VLAN 範囲で DHCP スヌーピングをイネーブルにします。指定できる範囲は 1 ～ 4094 です。VLAN ID 番号によって特定される単一の VLAN ID、それぞれをカンマで区切った一連の VLAN ID、ハイフンを間に挿入した VLAN ID の範囲、または先頭および末尾の VLAN ID で区切られ

	コマンドまたはアクション	目的
		た VLAN ID の範囲を入力することができます。これらはスペースで区切ります。 • VLAN ID 番号によって特定される単一の VLAN ID、それぞれをカンマで区切った一連の VLAN ID、ハイフンを間に挿入した VLAN ID の範囲、または先頭および末尾の VLAN ID で区切られた VLAN ID の範囲を入力することができます。これらはスペースで区切ります。
ステップ 5	**ip dhcp snooping information option** 例： `スイッチ(config)# ip dhcp snooping information option`	スイッチが、転送された DHCP 要求メッセージにある DHCP リレー情報（オプション 82 フィールド）を DHCP サーバーに挿入したり削除したりできるようにイネーブルにします。これがデフォルト設定です。
ステップ 6	**ip dhcp snooping information option format remote-id** [**string** *ASCII-string* | **hostname**] 例： `スイッチ(config)# ip dhcp snooping information option format remote-id string acsiistring2`	（任意）リモート ID サブオプションを設定します。 リモート ID は次のように設定できます。 • 63 文字までの ASCII 文字列（スペースなし） • スイッチに設定されたホスト名 （注） ホスト名が 64 文字以上の場合、リモート ID 設定で 63 文字に切り捨てられます。 デフォルトのリモート ID はスイッチ MAC アドレスです。
ステップ 7	**ip dhcp snooping information option allow-untrusted** 例： `スイッチ(config)# ip dhcp snooping information option allow-untrusted`	（任意）スイッチが、エッジ スイッチに接続された集約スイッチである場合、エッジ スイッチからのオプション 82 情報付き着信 DHCP スヌーピング パケットを受け入れるようにこのコマンドによってスイッチをイネーブルにします。 デフォルト設定では無効になっています。 （注） このコマンドは、信頼できるデバイスに接続された集約スイッチだけで入力してください。
ステップ 8	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/1`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 9	**ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] **string** *ASCII-string* 例： `スイッチ(config-if)# ip dhcp snooping vlan 1 information option format-type curcuit-id override string ovrride2`	（任意）指定したインターフェイスの回線 ID サブオプションを設定します。 1 ～ 4094 の範囲の VLAN ID を使用して、VLAN およびポート ID を指定します。デフォルトの回線 ID はポート ID で、フォーマットは **vlan-mod-port** です。 回線 ID は 3 ～ 63 の ASCII 文字列（スペースなし）を設定できます。 （任意）**override** キーワードは、加入者情報を定義するための TLV 形式に回線 ID サブオプションを挿入したくない場合に使用します。
ステップ 10	**ip dhcp snooping trust** 例： `スイッチ(config-if)# ip dhcp snooping trust`	（任意）インターフェイスの信頼性を trusted または untrusted に設定します。信頼できないクライアントからのメッセージを受信するようにインターフェイスを設定するには、**no** キーワードを使用します。デフォルト設定は untrusted です。
ステップ 11	**ip dhcp snooping limit rate** *rate* 例： `スイッチ(config-if)# ip dhcp snooping limit rate 100`	（任意）インターフェイスが受信できる 1 秒あたりの DHCP パケット数を設定します。指定できる範囲は 1 ～ 2048 です。デフォルトでは、レート制限は設定されません。 （注） 信頼できないインターフェイスのレート制限を 1 秒あたり 100 パケット以下に設定することを推奨します。信頼できるインターフェイスのレート制限を設定する場合、DHCP スヌーピングを使った複数の VLAN に割り当てられたトランク ポートでは、レート制限の値を大きくすることが必要になることがあります。
ステップ 12	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 13	**ip dhcp snooping verify mac-address** 例： `スイッチ(config)# ip dhcp snooping verify mac-address`	（任意）信頼できないポートに着信した DHCP パケットの送信元 MAC アドレスがパケットのクライアント ハードウェア アドレスと一致することを確認するようにスイッチを設定します。デフォルトでは、送信元 MAC アドレスがパケットのクライアント ハードウェア アドレスと一致することを確認します。

	コマンドまたはアクション	目的
ステップ 14	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 15	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 16	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

Cisco IOS DHCP サーバ データベースのイネーブル化

Cisco IOS DHCP サーバ データベースをイネーブルにして設定する手順については、『Cisco IOS IP Configuration Guide, Release 12.4』の「Configuring DHCP」の章にある「DHCP Configuration Task List」の項を参照してください。

DHCP スヌーピング情報のモニタリング

表 *136 : DHCP* 情報を表示するためのコマンド

show ip dhcp snooping	スイッチの DHCP スヌーピングの設定を表示します。
show ip dhcp snooping binding	DHCP スヌーピング バインディング データベース内の動的に設定されたバインディングだけを表示します。このようなバインディングは、バインディング テーブルとも呼ばれます。
show ip dhcp snooping database	DHCP スヌーピング バインディング データベースのステータスおよび統計情報を表示します。
show ip dhcp snooping statistics	DHCP スヌーピングの統計情報を要約または詳細形式で表示します。

show ip source binding	動的および静的に設定されたバインディングを表示します。

（注） DHCP スヌーピングがイネーブルでインターフェイスがダウン ステートに変更された場合、静的に設定されたバインディングは削除されません。

DHCP サーバー ポートベースのアドレス割り当ての設定

DHCP サーバー ポートベースのアドレス割り当ての設定に関する情報

DHCP サーバー ポートベースのアドレス割り当ては、接続されたデバイス クライアントの ID またはクライアント ハードウェア アドレスに関係なく、DHCP がイーサネット スイッチ ポートで同じ IP アドレスを維持できるようにする機能です。

ネットワークに導入されたイーサネット スイッチは、直接接続されたデバイスに接続を提供します。工場の作業場など、一部の環境では、あるデバイスで不具合が発生した場合は、それと同時に、そのネットワークで代わりのデバイスが動作を開始しなければなりません。現在の DHCP 実装では、この代わりのデバイスに、DHCP が同じ IP アドレスを提供する保証はありません。コントロールやモニタリングなどを行うソフトウェアは、各デバイスに関連付けられた IP アドレスが一定であることを期待しています。デバイスを交換した場合、DHCP クライアントが変更された場合でも、アドレスの割り当ては一定のままでなければなりません。

DHCP サーバ ポートベースのアドレス割り当て機能が設定されている場合、この機能により、ある接続ポートで受信された DHCP メッセージでクライアント ID やクライアント ハードウェア アドレスが変更されたとしても、同じ接続ポートには常に同じ IP アドレスが提供されることが保証されます。DHCP プロトコルは、DHCP パケットのクライアント ID オプションにより、DHCP クライアントを識別します。クライアント ID オプションを含まないクライアントは、クライアント ハードウェア アドレスにより識別されます。この機能を設定すると、インターフェイスのポート名が、クライアント ID またはハードウェア アドレスよりも優先され、実際の接続ポイントであるスイッチ ポートがクライアント ID になります。

すべてのケースで、同じポートにイーサネット ケーブルを接続することにより、接続されたデバイスに、DHCP 経由で同じ IP アドレスが割り当てられます。

DHCP サーバ ポートベースのアドレス割り当て機能がサポートされているのは、Cisco IOS DHCP サーバだけです。サードパーティ製のサーバではサポートされていません。

ポートベースのアドレス テーブルのデフォルト設定

デフォルトでは、DHCP サーバ ポートベースのアドレス割り当てはディセーブルにされています。

ポートベースのアドレス割り当て設定時の注意事項

- デフォルトでは、DHCP サーバ ポートベースのアドレス割り当てはディセーブルにされています。
- DHCP プールから事前に設定された予約への割り当てを制限する（予約されていないアドレスはクライアントに提供されず、その他のクライアントはプールによるサービスを受けない）ために、**reserved-only** DHCP プール コンフィギュレーション コマンドを入力することができます。

DHCP スヌーピング バインディング データベース エージェントのイネーブル化

スイッチ上で DHCP スヌーピング バインディング データベース エージェントをイネーブルにし、設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip dhcp snooping database** {**flash**[*number*]**:**/*filename* | **ftp:**//*user***:***password***@***host*/*filename* | **http:**//[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** | **rcp:**//*user***@***host*/*filename*}| **tftp:**//*host*/*filename*
4. **ip dhcp snooping database timeout** *seconds*
5. **ip dhcp snooping database write-delay** *seconds*
6. **end**
7. **ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*
8. **show ip dhcp snooping database** [**detail**]
9. **show running-config**
10. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**ip dhcp snooping database** {**flash**[*number*]**:**/*filename* | **ftp://***user***:***password***@***host*/*filename* | **http://**[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** | **rcp://***user***@***host*/*filename*}| **tftp://***host*/*filename* 例： スイッチ(config)# **ip dhcp snooping database tftp://10.90.90.90/snooping-rp2**	次のいずれかの形式を使用して、データベースエージェントまたはバインディング ファイルの URL を指定します。 • **flash**[*number*]**:**/*filename* （任意）アクティブスイッチのスタックメンバー番号を指定するには、*number* パラメータを使用します。*number* の指定できる範囲は 1 ～ 9 です。 • **ftp://***user***:***password***@***host*/*filename* • **http://**[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** • **rcp://***user***@***host*/*filename* • **tftp://***host*/*filename*
ステップ 4	**ip dhcp snooping database timeout** *seconds* 例： スイッチ(config)# **ip dhcp snooping database timeout 300**	データベース転送プロセスが完了するのを待ち、それまでに完了しない場合はプロセスを停止する時間（秒数）を指定します。 デフォルトは 300 秒です。指定できる範囲は 0 ～ 86400 です。無期限の期間を定義するには、0 を使用します。これは転送を無期限に試行することを意味します。
ステップ 5	**ip dhcp snooping database write-delay** *seconds* 例： スイッチ(config)# **ip dhcp snooping database write-delay 15**	バインディング データベースが変更されてから転送を開始するまでの遅延時間を指定します。指定できる範囲は 15 ～ 86400 秒です。デフォルトは 300 秒（5 分）です。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id* *ip-address* **interface** *interface-id* **expiry** *seconds* 例：	（任意）DHCP スヌーピング バインディング データベースにバインディングエントリを追加します。*vlan-id* に指定できる範囲は 1 ～ 4904 です。*seconds* の範囲は 1 ～ 4294967295 です。

	コマンドまたはアクション	目的
	スイッチ# **ip dhcp snooping binding 0001.1234.1234 vlan 1 172.20.50.5 interface gi1/1 expiry 1000**	このコマンドは、追加するエントリごとに入力します。 このコマンドは、スイッチをテストまたはデバッグするときに使用します。
ステップ 8	**show ip dhcp snooping database** [**detail**] 例： スイッチ# show ip dhcp snooping database detail	DHCP スヌーピング バインディング データベース エージェントのステータスおよび統計情報を表示します。
ステップ 9	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

DHCP サーバ ポートベースのアドレス割り当てのイネーブル化

ポートベースのアドレス割り当てをグローバルにイネーブル化し、インターフェイス上で加入者 ID を自動的に生成するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip dhcp use subscriber-id client-id**
4. **ip dhcp subscriber-id interface-name**
5. **interface** *interface-id*
6. **ip dhcp server use subscriber-id client-id**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip dhcp use subscriber-id client-id** 例： スイッチ(config)# **ip dhcp use subscriber-id client-id**	すべての着信 DHCP メッセージで、加入者 ID がクライアント ID としてグローバルに使用されるように DHCP サーバを設定します。
ステップ 4	**ip dhcp subscriber-id interface-name** 例： スイッチ(config)# **ip dhcp subscriber-id interface-name**	インターフェイスの短い名前に基づいて、加入者IDを自動的に生成します。 特定のインターフェイスで設定された加入者IDは、このコマンドで優先されます。
ステップ 5	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	設定するインターフェイスを指定し、インターフェイス コンフィギュレーションモードを開始します。
ステップ 6	**ip dhcp server use subscriber-id client-id** 例： スイッチ(config-if)# **ip dhcp server use subscriber-id client-id**	インターフェイス上ですべての着信DHCPメッセージで、加入者 ID がクライアント ID として使用されるように DHCP サーバーを設定します。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例：	入力を確認します。

	コマンドまたはアクション	目的
	スイッチ# **show running-config**	
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

スイッチ上での DHCP ポートベースのアドレス割り当てをイネーブルにした後で、**ip dhcp pool** グローバル コンフィギュレーション コマンドを使用して、IP アドレスの事前割り当てと、クライアントへの関連付けを行います。

DHCP サーバ ポートベースのアドレス割り当てのモニタリング

表 *137 : DHCP* ポートベースのアドレス割り当て情報を表示するためのコマンド

コマンド	目的
show interface *interface id*	特定のインターフェイスのステータスおよび設定を表示します。
show ip dhcp pool	DHCP アドレス プールを表示します。
show ip dhcp binding	Cisco IOS DHCP サーバのアドレス バインディングを表示します。

第 66 章

IP ソース ガードの設定

IP ソース ガード（IPSG）は、ルーティングされないレイヤ 2 インターフェイス上の IP トラフィックを制限するセキュリティ機能で、DHCP スヌーピング バインディング データベースと手動で設定された IP ソース バインディングに基づいてトラフィックをフィルタリングすることで実現されます。

この章は、次の内容で構成されています。

IP ソース ガードの概要

IP ソース ガード

ネイバーの IP アドレスを使用する場合に、トラフィック攻撃を防ぐために IP ソース ガードを使用でき、そして信頼できないインターフェイスで DHCP スヌーピングがイネーブルの場合に、IP アドレスを使用しようとすると、IP ソース ガードをイネーブルにできます。

インターフェイス上で IPSG をイネーブルにすると、スイッチは、DHCP スヌーピングにより許可された DHCP パケットを除き、このインターフェイスで受信したすべての IP トラフィックをブロックします。

スイッチは IP アドレスをポートにバインドするためにハードウェアの発信元 IP 検索テーブルを使用します。IP および MAC のフィルタリングでは、送信元 IP 検索および送信元 MAC 検索の組み合わせが使用されます。バインディングテーブル内の送信元 IP アドレスを使用する IP トラフィックは許可され、他のすべてのトラフィックは拒否されます。

IP ソース バインディング テーブルには、DHCP スヌーピングで学習されたバインディング、または手動で設定されたバインディング（スタティック IP 送信元バインディング）があります。このテーブルのエントリには IP アドレスと、関連 MAC アドレス、および関連 VLAN 番号があります。スイッチは、IP ソース ガードがイネーブルにされている場合だけ、IP ソース バインディング テーブルを使用します。

IPSG は、アクセス ポートおよびトランク ポートを含むレイヤ 2 ポートだけでサポートされます。送信元 IP アドレスと送信元 IP および MAC アドレス フィルタリングで IPSG を設定できます。

スタティック ホスト用 IP ソース ガード

（注） アップリンク ポート、またはトランク ポートで、スタティック ホスト用 IP ソース ガード（IPSG）を使用しないでください。

スタティック ホスト用 IPSG は、IPSG の機能を DHCP ではない、スタティックな環境に拡張するものです。これまでの IPSG は、DHCP スヌーピングにより作成されたエントリを使用して、スイッチに接続されたホストを検証していました。ホストから受信したトラフィックのうち、有効な DHCP を持たないものはすべてドロップされます。このセキュリティ機能によって、ルーティングされないレイヤ 2 インターフェイス上の IP トラフィックが制限されます。この機能は、DHCP スヌーピング バインディング データベース、および手動で設定された IP ソースバインディングに基づいてトラフィックをフィルタリングします。前バージョンの IPSG では、IPSG を動作させるために DHCP 環境が必要でした。

スタティック ホスト用 IPSG では、DHCP なしで IPSG を動作させることができます。スタティック ホスト用 IPSG は、ポート ACL をインストールするために IP デバイス トラッキング テーブルエントリに依存しています。このスイッチは、指定されたポートで有効なホストのリストを維持するために、ARP リクエスト、またはその他の IP パケットに基づいてスタティック エントリを作成します。また、指定されたポートにトラフィックを送信できるホストの数を指定することもできます。これはレイヤ 3 でのポート セキュリティと同じです。

スタティック ホスト用 IPSG はダイナミック ホストもサポートしています。ダイナミック ホストが、IP DHCP スヌーピング テーブルに存在する DHCP が割り当てられた IP アドレスを受信すると、IP デバイス トラッキング テーブルは同じエントリを学習します。スタック化環境では、アクティブスイッチのフェールオーバーが発生すると、メンバポートに接続されたスタティックホストの IP ソースガードエントリは、そのまま残ります。**show ip device tracking all** 特権 EXEC コマンドを入力すると、IP デバイス トラッキング テーブルには、これらのエントリが ACTIVE であると表示されます。

（注） 複数のネットワーク インターフェイスを持つ IP ホストの一部は、ネットワーク インターフェイスに無効なパケットを注入することができます。この無効なパケットには、ソース アドレスとして、別のホスト ネットワーク インターフェイスの IP アドレス、または MAC アドレスが含まれます。この無効パケットは、スタティック ホスト用 IPSG がホストに接続され、無効な IP アドレス バインディングまたは MAC アドレス バインディングが学習されて、有効なバインディングが拒否される原因となります。ホストによる無効なパケットの注入を回避する方法については、対応するオペレーティング システムとネットワーク インターフェイスのベンダーにお問い合わせください。

最初、スタティック ホスト用 IPSG は ACL ベースのスヌーピング メカニズムを通じて、動的に IP バインディング、または MAC バインディングを学習します。IP バインディング、または MAC バインディングは、ARP パケット、および IP パケットにより、スタティック ホストから学習されます。これらはデバイス トラッキング データベースに保存されます。指定されたポートで動的に学習、または静的に設定された IP アドレスの数が最大値に達した場合、新しい IP アドレスを持つパケットはすべて、ハードウェアによりドロップされます。何らかの理由で移動された、またはなくなったホストを解決するために、スタティック ホスト用 IPSG は IP デバイス トラッキングを活用して、動的に学習した IP アドレス バインディングをエージング アウトします。この機能は、DHCP スヌーピングとともに使用できます。複数バインディングは、DHCP ホストとスタティック ホストの両方に接続されたポートに確立されます。たとえば、バインディングは、デバイス トラッキング データベースと DHCP スヌーピング バインディング データベースの両方に保存されます。

IP ソース ガードの設定時の注意事項

- スタティック IP バインディングは、非ルーテッド ポートだけで設定できます。ルーテッドインターフェイスで **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* グローバル コンフィギュレーション コマンドを入力すると、次のエラーメッセージが表示されます。

```
Static IP source binding can only be configured on switch port.
```

- 送信元 IP フィルタリング機能を持つ IP ソース ガードがインターフェイスでイネーブルにされている場合、このインターフェイスのアクセス VLAN で、DHCP スヌーピングをイネーブルにしておく必要があります。
- 複数の VLAN を持つトランク インターフェイス上で IP ソース ガードをイネーブルにし、これらすべての VLAN で DHCP スヌーピングをイネーブルにした場合、すべての VLAN に、送信元 IP アドレス フィルタが適用されます。

（注） IP ソース ガードがイネーブルにされているときに、トランク インターフェイスの VLAN 上で DHCP スヌーピングをイネーブル、またはディセーブルにした場合、スイッチは適切にトラフィックをフィルタリングできない可能性があります。

- この機能は、802.1x ポートベースの認証がイネーブルにされている場合にイネーブルにできます。
- IP ソース ガード スマート ロギングを設定すると、指定されたアドレスや DHCP によって学習されたアドレス以外の送信元アドレスを持つパケットは拒否され、そのパケットの内容が NetFlow 収集装置に送信されます。この機能を設定する場合は、スマート ロギングがグローバルにイネーブルになっていることを確認してください。

- スイッチスタックでは、IP ソースガードがスタック メンバ インターフェイスに設定されていて、**no switch** *stack-member-number* **provision** グローバル コンフィギュレーション コマンドの入力によりそのスイッチの設定を削除した場合、インターフェイス スタティック バインディングはバインディングテーブルから削除されますが、実行コンフィギュレーションからは削除されません。**switch** *stack-member-number* **provision** コマンドを入力することによって、スイッチを再度プロビジョニングした場合、バインディングは復元されます。

  実行コンフィギュレーションからバインディングを削除するには、**no switch provision** コマンドを入力する前に IP ソースガードを無効化する必要があります。インターフェイスがバインディングテーブルから削除される間にスイッチがリロードされると、設定も削除されます。

IP ソース ガードの設定方法

IP ソース ガードのイネーブル化

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip verify source**［**mac-check** ］
5. **exit**
6. **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip verify source**［**mac-check** ］ 例： `スイッチ(config-if)# ip verify source`	送信元 IP アドレス フィルタリングによる IP ソース ガードを有効にします。 （任意）**mac-check**：送信元 IP アドレスによる IP ソース ガードおよびMAC アドレス フィルタリングをイネーブルにします。
ステップ 5	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* 例： `スイッチ(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet1/0/1`	スタティック IP ソース バインディングを追加します。 スタティック バインディングごとにこのコマンドを入力します。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

レイヤ 2 アクセス ポートでのスタティック ホスト用 IP ソース ガードの設定

スタティックホスト用 IPSG を動作させるには、**ip device tracking maximum** *limit-number* インターフェイス コンフィギュレーション コマンドをグローバルに設定する必要があります。このコマンドをポートに対して実行したが、IP デバイス トラッキングをグローバルに有効にしていない、または IP device tracking maximum をそのインターフェイスに対して設定していない場合は、スタティック ホストの IPSG によって、そのインターフェイスからの IP トラフィックはすべて拒否されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip device tracking**
4. **interface** *interface-id*
5. **switchport mode access**
6. **switchport access vlan** *vlan-id*
7. **ip verify source**[**tracking**] [**mac-check**]
8. **ip device tracking maximum** *number*
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip device tracking** 例： `スイッチ(config)# ip device tracking`	IP ホスト テーブルをオンにし、IP デバイス トラッキングをグローバルに有効にします。
ステップ 4	**interface** *interface-id* 例：	インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ(config)# **interface gigabitethernet 1/0/1**	
ステップ 5	**switchport mode access** 例： スイッチ(config-if)# **switchport mode access**	アクセスとしてポートを設定します。
ステップ 6	**switchport access vlan** *vlan-id* 例： スイッチ(config-if)# **switchport access vlan 10**	このポートに VLAN を設定します。
ステップ 7	**ip verify source**[**tracking**] [**mac-check**] 例： スイッチ(config-if)# **ip verify source tracking mac-check**	送信元 IP アドレス フィルタリングによる IP ソース ガードを有効にします。 （任意）**tracking**：スタティック ホスト用 IP ソース ガードを有効にします。 （任意）**mac-check**：MAC アドレス フィルタリングを有効にします。 **ip verify source tracking mac-check** コマンドは、MAC アドレス フィルタリングのあるスタティック ホストに対して IP ソース ガードを有効にします。
ステップ 8	**ip device tracking maximum** *number* 例： スイッチ(config-if)# **ip device tracking maximum 8**	そのポートで、IP デバイス トラッキング テーブルにより許可されるスタティック IP 数の上限を設定します。指定できる範囲は 1 ～ 10 です。最大値は 10 です。 （注） **ip device tracking maximum** *limit-number* インターフェイス コンフィギュレーション コマンドを設定する必要があります。
ステップ 9	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

IP ソース ガードのモニタリング

表 *138* : 特権 *EXEC* 表示コマンド

コマンド	目的
show ip verify source [**interface** *interface-id*]	スイッチ上または特定のインターフェイス上のIP ソース ガードの設定を表示します。
show ip device tracking { **all** | **interface** *interface-id* | **ip** *ip-address* | **mac** *mac-address*}	IP デバイス トラッキング テーブル内のエントリに関する情報を表示します。

表 *139* : インターフェイス コンフィギュレーション コマンド

コマンド	目的
ip verify source tracking	データ ソースを確認します。

出力フィールドの詳細については、このリリースに対応するコマンドリファレンスを参照してください。

第 67 章

ダイナミック ARP インスペクションの設定

ダイナミック ARP インスペクションの制約事項

ここでは、スイッチにダイナミック ARP インスペクションを設定するときの制約事項および注意事項を示します。

- ダイナミック ARP インスペクションは入力セキュリティ機能です。出力チェックはまったく行いません。
- ダイナミック ARP インスペクションは、ダイナミック ARP インスペクションをサポートしていないスイッチ、またはこの機能がイネーブルにされていないスイッチに接続されるホストに対しては有効ではありません。中間者攻撃は単一のレイヤ 2 ブロードキャスト ドメインに制限されているため、チェックされないドメインと、ダイナミック ARP インスペクションによりチェックされるドメインは区別します。このアクションは、ダイナミック ARP インスペクションのためにイネーブルにされているドメインでホストの ARP キャッシュを保護します。
- 着信 ARP 要求、および ARP 応答で IP/MAC アドレス バインディングを検証するために、ダイナミック ARP インスペクション DHCP スヌーピング バインディング データベースの

エントリに依存します。IP アドレスがダイナミックに割り当てられた ARP パケットを許可する際は、DHCP スヌーピングをイネーブルにしてください。

DHCP スヌーピングをディセーブルにしている場合、または DHCP 以外の環境では、ARP ACL を使用してパケットの許可または拒否を行います。

- ダイナミック ARP インスペクションは、アクセス ポート、トランク ポート、および EtherChannel ポートでサポートされます。

（注） RSPAN VLAN では、ダイナミック ARP インスペクションをイネーブルにしないでください。RSPAN VLAN でダイナミック ARP インスペクションをイネーブルにすると、ダイナミック ARP インスペクション パケットが RSPAN 宛先ポートに届かない可能性があります。

- 物理ポートを EtherChannel ポート チャネルに結合するには、この物理ポートとチャネル ポートの信頼状態が一致する必要があります。そうでない物理ポートは、ポート チャネル内で中断状態のままとなります。ポート チャネルは、チャネルと結合された最初の物理ポートの信頼状態を継承します。したがって、最初の物理ポートの信頼状態は、チャネルの信頼状態と一致する必要はありません。

  逆に、ポート チャネルで信頼状態を変更すると、スイッチは、チャネルを構成するすべての物理ポートで新しい信頼状態を設定します。

- レート制限は、スイッチ スタックの各スイッチで別々に算出されます。クロススタック EtherChannel の場合、これは実際のレート制限が設定値よりも高い可能性があることを意味します。たとえば、レート制限が 30 pps に設定された EtherChannel で、スイッチ 1 に 1 つのポート、およびスイッチ 2 に 1 つのポートがある場合、EtherChannel が errdisable にならずに、各ポートは 29 pps でパケットを受信できます。

- ポート チャネルの動作レートは、チャネル内のすべての物理ポートによる累積値です。たとえば、ポート チャネルの ARP レート制限を 400 pps に設定すると、このチャネルに結合されたすべてのインターフェイスは、合計で 400 pps を受信します。EtherChannel ポートで受信される ARP パケットのレートは、すべてのチャネル メンバーからの受信パケット レートの合計となります。EtherChannel ポートのレート制限は、各チャネル ポート メンバーが受信する ARP パケットのレートを確認してから設定してください。

  物理ポートで受信されるパケットのレートは、物理ポートの設定ではなく、ポート チャネルの設定に照合して検査されます。ポート チャネルのレート制限設定は、物理ポートの設定には依存しません。

  EtherChannel が、設定したレートより多くの ARP パケットを受信すると、このチャネル（すべての物理ポートを含む）は errdisable ステートとなります。

- 着信トランク ポートでは、ARP パケットを必ずレート制限してください。トランク ポートの集約を反映し、複数のダイナミック ARP インスペクションがイネーブルにされた VLAN にわたってパケットを処理するために、トランク ポートのレートをより高く設定します。また、**ip arp inspection limit none** インターフェイス コンフィギュレーション コ

マンドを使用して、レートを無制限に設定することもできます。1 つの VLAN に高いレート制限値を設定すると、ソフトウェアによってこのポートが errdisable ステートにされた場合に、他の VLAN への DoS 攻撃を招く可能性があります。

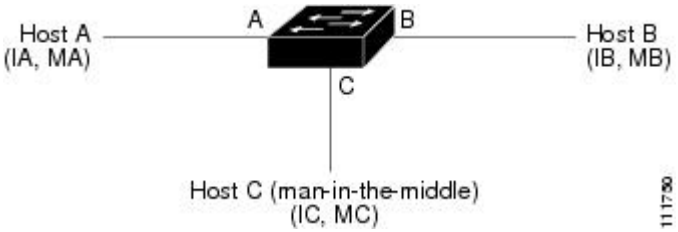
- スイッチで、ダイナミック ARP インスペクションをイネーブルにすると、ARP トラフィックをポリシングするように設定されたポリサーの有効性は失われます。この結果、すべての ARP トラフィックは CPU に送信されます。

ダイナミック ARP インスペクションの概要

ARP では、IP アドレスを MAC アドレスにマッピングすることで、レイヤ 2 ブロードキャストドメイン内の IP 通信を実現します。たとえば、ホスト B はホスト A に情報を送信する必要がありますが、ARP キャッシュにホスト A の MAC アドレスを持っていないとします。ホスト B は、ホスト A の IP アドレスと関連付けられた MAC アドレスを取得するために、このブロードキャストドメインにあるホストすべてに対してブロードキャストメッセージを生成します。このブロードキャスト ドメイン内のホストはすべて ARP 要求を受信し、ホスト A は MAC アドレスで応答します。しかし、ARP は、ARP 要求が受信されなった場合でも、ホストからの余分な応答を許可するため、ARP スプーフィング攻撃や ARP キャッシュのポイズニングが発生することがあります。攻撃が開始されると、攻撃を受けたデバイスからのすべてのトラフィックは、攻撃者のコンピュータを経由してルータ、スイッチ、またはホストに送信されるようになります。

悪意のあるユーザーは、サブネットに接続されているシステムの ARP キャッシュをポイズニングし、このサブネット上の他のホストを目的とするトラフィックを代行受信することにより、レイヤ 2 ネットワークに接続されているホスト、スイッチ、およびルータを攻撃することができます。図 26-1 に、ARP キャッシュ ポイズニングの例を示します。

図 *106 : ARP* キャッシュ ポイズニング



ホスト A、B、および C は、インターフェイス A、B、および C 上にあるスイッチに接続されています。これらはすべて同一のサブネット上にあります。カッコ内に示されているのは、これらの IP アドレス、および MAC アドレスです。たとえば、ホスト A が使用する IP アドレスは IA、MAC アドレスは MA です。ホスト A が IP レイヤにあるホスト B と通信する必要がある場合、ホスト A は IP アドレス IB と関連付けられている MAC アドレスに ARP 要求をブロードキャストします。スイッチとホスト B は、この ARP 要求を受信すると、IP アドレスが IA で、MAC アドレスが MA のホストに対する ARP バインディングを ARP キャッシュに読み込みます。たとえば、IP アドレス IA は、MAC アドレス MA にバインドされています。ホスト B が応答すると、スイッチ、およびホスト A は、IP アドレスが IB で、MAC アドレスが MB のホストに対するバインディングを ARP に読み込みます。

ホスト C は、IP アドレスが IA（または IB）で、MAC アドレスが MC のホストに対するバインディングを持つ偽造 ARP 応答をブロードキャストすることにより、スイッチ、ホスト A、およびホスト B の ARP キャッシュをポイズニングすることができます。ARP キャッシュがポイズニングされたホストは、IA または IB 宛てのトラフィックに、宛先 MAC アドレスとして MAC アドレス MC を使用します。つまり、ホスト C がこのトラフィックを代行受信することになります。ホスト C は IA および IB に関連付けられた本物の MAC アドレスを知っているため、正しい MAC アドレスを宛先として使用することで、代行受信したトラフィックをこれらのホストに転送できます。ホスト C は自身をホスト A からホスト B へのトラフィック ストリームに挿入します。従来の中間者攻撃です。

ダイナミック ARP インスペクションは、ネットワーク内の ARP パケットの正当性を確認するセキュリティ機能です。不正な IP/MAC アドレス バインディングを持つ ARP パケットを代行受信し、ログに記録して、廃棄します。この機能により、ネットワークをある種の中間者攻撃から保護することができます。

ダイナミック ARP インスペクションにより、有効な ARP 要求と応答だけが確実にリレーされるようになります。スイッチが実行する機能は次のとおりです。

- 信頼できないポートを経由したすべての ARP 要求および ARP 応答を代行受信します。
- 代行受信した各パケットが、IP アドレスと MAC アドレスの有効なバインディングを持つことを確認してから、ローカル ARP キャッシュを更新するか、または適切な宛先にパケットを転送します。
- 無効な ARP パケットはドロップします。

ダイナミック ARP インスペクションは、信頼できるデータベースである DHCP スヌーピング バインディング データベースに格納されている有効な IP/MAC アドレス バインディングに基づいて、ARP パケットの正当性を判断します。このデータベースは、VLAN およびスイッチ上で DHCP スヌーピングが有効になっている場合に、DHCP スヌーピングにより構築されます。信頼できるインターフェイスで ARP パケットが受信されると、スイッチは何もチェックせずに、このパケットを転送します。信頼できないインターフェイスでは、スイッチはこのパケットが有効である場合だけ、このパケットを転送します。

ip arp inspection vlan *vlan-range* グローバル コンフィギュレーション コマンドを使用して、VLAN ごとにダイナミック ARP インスペクションを有効にすることができます。

非 DHCP 環境では、ダイナミック ARP インスペクションは、静的に設定された IP アドレスを持つホストに対するユーザー設定の ARP アクセス コントロール リスト（ACL）と照らし合わせて、ARP パケットの正当性を確認することができます。ARP ACL を定義するには、**arp access-list** *acl-name* グローバル コンフィギュレーション コマンドを使用します。

パケットの IP アドレスが無効である場合、または ARP パケットの本文にある MAC アドレスが、イーサネット ヘッダーで指定されたアドレスと一致しない場合、ARP パケットをドロップするようにダイナミック ARP インスペクションを設定することができます。このためには、**ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]} グローバル コンフィギュレーション コマンドを使用します。

インターフェイスの信頼状態とネットワーク セキュリティ

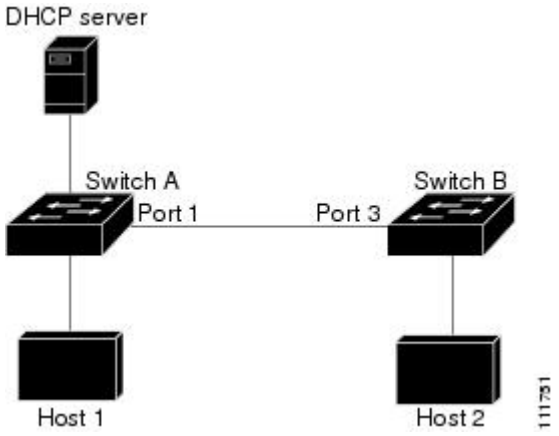
ダイナミック ARP インスペクションは、スイッチの各インターフェイスに信頼状態を関連付けます。信頼できるインターフェイスに到着するパケットは、ダイナミック ARP インスペクションの確認検査をすべてバイパスし、信頼できないインターフェイスに到着するパケットには、ダイナミック ARP インスペクションの検証プロセスを受けます。

一般的なネットワーク構成では、ホスト ポートに接続されているスイッチ ポートすべてを信頼できないものに設定し、スイッチに接続されているスイッチ ポートすべてを信頼できるものに設定します。この構成では、指定されたスイッチからネットワークに入ってくるARPパケットはすべて、セキュリティ チェックをバイパスします。VLAN 内、またはネットワーク内のその他の場所では、他の検査を実行する必要はありません。信頼状態を設定するには、ip arp inspection trust インターフェイス コンフィギュレーション コマンドを使用します。

注意 信頼状態の設定は、慎重に行ってください。信頼すべきインターフェイスを信頼できないインターフェイスとして設定すると、接続が失われる場合があります。

次の図では、スイッチ A とスイッチ B の両方が、ホスト 1 とホスト 2 を含む VLAN でダイナミック ARP インスペクションを実行しているとします。ホスト 1 とホスト 2 が、スイッチ A に接続している DHCP サーバーから IP アドレスを取得している場合、スイッチ A だけが、ホスト 1 の IP/MAC アドレスをバインディングします。したがって、スイッチ A とスイッチ B 間のインターフェイスが信頼できない場合は、ホスト 1 からの ARP パケットはスイッチ B ではドロップされ、ホスト 1 およびホスト 2 の間の接続は切断されます。

図 *107*: ダイナミック *ARP* インスペクションのために有効にされた *VLAN* 上の *ARP* パケット検証



実際には信頼できないインターフェイスを信頼できるインターフェイスとして設定すると、ネットワーク内にセキュリティ ホールが生じます。スイッチ A でダイナミック ARP インスペクションが実行されていない場合、ホスト 1 はスイッチ B の ARP キャッシュを簡単にポイズニングできます（および、これらのスイッチの間のリンクが信頼できるものとして設定されている場合はホスト 2）。この状況は、スイッチ B がダイナミック ARP インスペクションを実行している場合でも発生します。

ダイナミック ARP インスペクションは、ダイナミック ARP インスペクションを実行しているスイッチに接続された（信頼できないインターフェイス上の）ホストが、そのネットワークにあるその他のホストのARPキャッシュをポイズニングしていないことを保証します。しかし、ダイナミック ARP インスペクションにより、ネットワークの他の部分にあるホストが、ダイナミックARPインスペクションを実行しているスイッチに接続されているホストのキャッシュをポイズニングできないようにすることはできません。

VLAN のスイッチの一部がダイナミック ARP インスペクションを実行し、残りのスイッチは実行していない場合、このようなスイッチに接続しているインターフェイスは信頼できないものとして設定します。ただし、非ダイナミック ARP インスペクションスイッチからパケットのバインディングを検証するには、ARP ACL を使用して、ダイナミック ARP インスペクションを実行するスイッチを設定します。このようなバインディングが判断できない場合は、レイヤ 3 で、ダイナミック ARP インスペクション スイッチを実行していないスイッチから、ダイナミック ARP インスペクションを実行しているスイッチを分離します。

（注） DHCP サーバーとネットワークの設定によっては、VLAN 上のすべてのスイッチで指定された ARP パケットを検証できない可能性があります。

ARP パケットのレート制限

スイッチの CPU は、ダイナミック ARP インスペクション確認検査を実行します。したがって、DoS 攻撃を阻止するために、着信 ARP パケット数はレート制限されます。デフォルトでは、信頼できないインターフェイスのレートは 15 パケット/秒（pps）です。信頼できるインターフェイスはレート制限されません。この設定を変更するには、**ip arp inspection limit** インターフェイス コンフィギュレーション コマンドを使用します。

着信 ARP パケットのレートが設定された制限を超えると、スイッチはポートを errdisable ステートにします。ユーザーが介入するまで、ポートはこの状態を維持します。**errdisable recovery** グローバル コンフィギュレーション コマンドを使用すると、errdisable ステートの回復をイネーブルにできます。これによって、ポートは指定のタイムアウト時間が経過すると、この状態から自動的に回復するようになります。

（注） EtherChannel のレート制限は、スタックにある各スイッチに個別に適用されます。たとえば、EtherChannel で 20 pps の制限が設定されている場合、EtherChannel にあるポートの各スイッチでは、最大 20 pps まで実行できます。スイッチが制限を超過した場合、EtherChannel 全体が errdisable ステートになります。

ARP ACL および DHCP スヌーピング エントリの相対的なプライオリティ

ダイナミック ARP インスペクションでは、有効な IP/MAC アドレス バインディングのリストとして、DHCP スヌーピング バインディング データベースが使用されます。

DHCP スヌーピング バインディング データベース内のエントリより、ARP ACL の方が優先されます。スイッチが ACL を使用するのは、ACL が **ip arp inspection filter vlan** グローバル コンフィギュレーション コマンドを使用して作成されている場合だけです。スイッチは、まず、ARP パケットをユーザー設定の ARP ACL と比較します。DHCP スヌーピングによりデータが入力されたデータベースに有効なバインディングが存在していても、ARP ACL が ARP パケットを拒否する場合、スイッチもこのパケットを拒否します。

廃棄パケットのロギング

スイッチがパケットをドロップすると、ログ バッファにエントリが記録され、その割合に応じて、システム メッセージが生成されます。メッセージの生成後、スイッチにより、ログ バッファからこのエントリが消去されます。各ログエントリには、受信側の VLAN、ポート番号、送信元 IP アドレスおよび宛先 IP アドレス、送信元 MAC アドレスおよび宛先 MAC アドレスといったフロー情報が記録されます。

ip arp inspection log-buffer グローバル コンフィギュレーション コマンドを使用して、バッファ内のエントリ数や、システム メッセージ生成までの指定のインターバルに必要とされるエントリ数を設定します。記録されるパケットの種類を指定するには、**ip arp inspection vlan logging** グローバル コンフィギュレーション コマンドを使用します。

ダイナミック ARP インスペクションのデフォルト設定

機能	デフォルト設定
ダイナミック ARP インスペクション	すべての VLAN でディセーブル。
インターフェイスの信頼状態	すべてのインターフェイスは untrusted。
着信 ARP パケットのレート制限	1 秒間に 15 台の新規ホストに接続するホストが配置されたスイッチドネットワークの場合、信頼できないインターフェイスのレートは 15 pps に設定されます。 信頼できるすべてのインターフェイスでは、レート制限は行われません。 バースト インターバルは 1 秒です。
非 DHCP 環境に対する ARP ACL	ARP ACL は定義されません。
有効性検査	検査は実行されません。

機能	デフォルト設定
ログ バッファ	ダイナミック ARP インスペクションがイネーブル化されると、拒否またはドロップされた ARP パケットはすべてが記録されます。 ログ内のエントリ数は 32 です。 システム メッセージ数は、毎秒 5 つに制限されます。 ロギング レート インターバルは 1 秒です。
VLAN 単位のロギング	拒否または廃棄されたすべての ARP パケットが記録されます。

ARP ACL および DHCP スヌーピング エントリの相対的なプライオリティ

ダイナミック ARP インスペクションでは、有効な IP/MAC アドレス バインディングのリストとして、DHCP スヌーピング バインディング データベースが使用されます。

DHCP スヌーピング バインディング データベース内のエントリより、ARP ACL の方が優先されます。スイッチが ACL を使用するのは、ACL が ip arp inspection filter vlan グローバル コンフィギュレーション コマンドを使用して作成されている場合だけです。スイッチは、まず、ARP パケットをユーザー設定の ARP ACL と比較します。DHCP スヌーピングによりデータが入力されたデータベースに有効なバインディングが存在していても、ARP ACL が ARP パケットを拒否する場合、スイッチもこのパケットを拒否します。

非 DHCP 環境での ARP ACL の設定

この手順は、図 2 に示すスイッチ B がダイナミック ARP インスペクション、または DHCP スヌーピングをサポートしていないときにダイナミック ARP インスペクションを設定する方法を示しています。

スイッチ A のポート 1 を信頼できるものとして設定した場合、スイッチ A とホスト 1 は両方とも、スイッチ B またはホスト 2 により攻撃される可能性があるため、セキュリティ ホールが作り出されます。これを阻止するには、スイッチ A のポート 1 を信頼できないものとして設定する必要があります。ホスト 2 からの ARP パケットを許可するには、ARP ACL をセットアップして、これを VLAN 1 に適用する必要があります。ホスト 2 の IP アドレスがスタティックではない（スイッチ A で ACL 設定を適用することは不可能である）場合、レイヤ 3 でスイッチ A をスイッチ B から分離し、これらの間では、ルータを使用してパケットをルートする必要があります。

スイッチ A で ARP ACL を設定するには、次の手順を実行します。この手順は、非 DHCP 環境では必須です。

手順の概要

1. **enable**
2. **configure terminal**
3. **arp access-list** *acl-name*
4. **permit ip host** *sender-ip* **mac host** *sender-mac*
5. **exit**
6. **ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]
7. **interface** *interface-id*
8. **no ip arp inspection trust**
9. **end**
10. 次の show コマンドを使用します。
    - **show arp access-list** acl-name
    - **show ip arp inspection vlan** *vlan-range*
    - **show ip arp inspection interfaces**
11. **show running-config**
12. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**arp access-list** *acl-name*	ARP ACL を定義し、ARP アクセス リスト コンフィギュレーション モードを開始します。デフォルトでは、ARP アクセス リストは定義されません。 （注） ARP アクセスリストの末尾に暗黙的な **deny ip any mac any** コマンドが指定されています。
ステップ 4	**permit ip host** *sender-ip* **mac host** *sender-mac*	指定されたホスト（ホスト 2）からの ARP パケットを許可します。

	コマンドまたはアクション	目的
		• *sender-ip* には、ホスト 2 の IP アドレスを入力します。 • *sender-mac* には、ホスト 2 の MAC アドレスを入力します。
ステップ 5	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	VLAN に ARP ACL を適用します。デフォルトでは、定義済みの ARP ACL は、どのような VLAN にも適用されません。 • *arp-acl-name* には、ステップ 2 で作成した ACL の名前を指定します。 • *vlan-range* では、スイッチとホストが存在する VLAN を指定します。VLAN ID 番号により識別される単一の VLAN、ハイフンで区切られた VLAN 範囲、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。 • （任意）**static** を指定すると、ARP ACL 内の暗黙的な拒否が明示的な拒否と見なされ、それ以前に指定された ACL 句に一致しないパケットは廃棄されます。DHCP バインディングは使用されません。 このキーワードを指定しない場合は、ACL 内にはパケットを拒否する明示的な拒否が存在しないことになります。この場合は、ACL 句に一致しないパケットを許可するか拒否するかは、DHCP バインディングによって決定されます。 IP アドレスと MAC アドレスとのバインディングしか持たない ARP パケットは、ACL に照合されます。パケットは、アクセス リストで許可された場合だけに許可されます。
ステップ 7	**interface** *interface-id*	スイッチ B に接続されたスイッチ A のインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8	**no ip arp inspection trust**	スイッチ B に接続されたスイッチ A のインターフェイスを untrusted として設定します。

	コマンドまたはアクション	目的
		デフォルトでは、すべてのインターフェイスは信頼できません。 信頼できないインターフェイスでは、スイッチはすべての ARP 要求と応答を代行受信します。ルータは、代行受信した各パケットが、IP アドレスと MAC アドレスとの有効なバインディングを持つことを確認してから、ローカル キャッシュを更新するか、適切な宛先にパケットを転送します。スイッチは、無効なパケットをドロップし、**ip arp inspection vlan logging** グローバル コンフィギュレーション コマンドで指定されたロギング設定に従ってログバッファに記録します。
ステップ 9	**end**	特権 EXEC モードに戻ります。
ステップ 10	次の show コマンドを使用します。 • **show arp access-list** acl-name • **show ip arp inspection vlan** *vlan-range* • **show ip arp inspection interfaces**	入力を確認します。
ステップ 11	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 12	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

DHCP 環境でのダイナミック ARP インスペクションの設定

始める前に

この手順では、2つのスイッチがダイナミック ARP インスペクションをサポートしているときに、この機能を設定する方法を示します。ホスト 1 はスイッチ A に、ホスト 2 はスイッチ B にそれぞれ接続されています。スイッチは両方とも、ホストが配置されている VLAN 1 でダイナミック ARP インスペクションを実行しています。DHCP サーバーはスイッチ A に接続されています。両方のホストは、同一の DHCP サーバーから IP アドレスを取得します。したがっ

て、スイッチ A はホスト 1 およびホスト 2 に対するバインディングを、スイッチ B はホスト 2 に対するバインディングを持ちます。

（注） 着信 ARP 要求、および ARP 応答で IP/MAC アドレス バインディングを検証するために、ダイナミック ARP インスペクション DHCP スヌーピング バインディング データベースのエントリに依存します。IP アドレスがダイナミックに割り当てられた ARP パケットを許可する際は、DHCP スヌーピングをイネーブルにしてください。

ダイナミック ARP インスペクションを設定するには、次の手順を実行します。この処理は、両方のスイッチで行う必要があります。この手順は必須です。

手順の概要

1. **enable**
2. **show cdp neighbors**
3. **configure terminal**
4. **ip arp inspection vlan** *vlan-range*
5. **Interface***interface-id*
6. **ip arp inspection trust**
7. **end**
8. **show ip arp inspection interfaces**
9. **show ip arp inspection vlan** *vlan-range*
10. **show ip dhcp snooping binding**
11. **show ip arp inspection statistics vlan** *vlan-range*
12. **configure terminal**
13. **configure terminal**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show cdp neighbors** 例： スイッチ(config-if)#**show cdp neighbors**	スイッチ間の接続を確認します。
ステップ 3	**configure terminal** 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 4	**ip arp inspection vlan** *vlan-range* 例： スイッチ(config)# **ip arp inspection vlan 1**	VLAN 単位で、ダイナミック ARP インスペクションをイネーブルにします。デフォルトでは、すべての VLAN 上でダイナミック ARP インスペクションはディセーブルになっています。vlan-range には、VLAN ID 番号で識別された単一の VLAN、ハイフンで区切られた範囲の VLAN、またはカンマで区切られた一連の VLAN を指定できます。指定できる範囲は 1 ～ 4094 です。両方のスイッチに同じ VLAN ID を指定します。
ステップ 5	**Interface***interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	他のスイッチに接続されるインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 6	**ip arp inspection trust** 例： スイッチ(config-if)#**ip arp inspection trust**	スイッチ間の接続を trusted に設定します。デフォルトでは、すべてのインターフェイスは信頼できません。 スイッチは、信頼できるインターフェイスにあるもう 1 つのスイッチから受信した ARP パケットは確認しません。この場合、パケットはそのまま転送されます。 信頼できないインターフェイスでは、スイッチはすべての ARP 要求と応答を代行受信します。ルータは、代行受信した各パケットが、IP アドレスと MAC アドレスとの有効なバインディングを持つことを確認してから、ローカル キャッシュを更新するか、適切な宛先にパケットを転送します。スイッチは、無効なパケットをドロップし、ip arp inspection vlan logging グローバル コンフィギュレーション コマンドで指定されたロギング設定に従ってログ バッファに記録します。
ステップ 7	**end** 例： スイッチ(config-if)#**end**	特権 EXEC モードに戻ります。
ステップ 8	**show ip arp inspection interfaces** 例：	インターフェイスでダイナミック ARP インスペクションの設定を検証します。

	コマンドまたはアクション	目的
ステップ 9	**show ip arp inspection vlan** *vlan-range* 例： `スイッチ(config-if)#show ip arp inspection vlan 1`	VLAN でダイナミック ARP インスペクションの設定を検証します。
ステップ 10	**show ip dhcp snooping binding** 例： `スイッチ(config-if)#show ip dhcp snooping binding`	DHCP バインディングを確認します。
ステップ 11	**show ip arp inspection statistics vlan** *vlan-range* 例： `スイッチ(config-if)#show ip arp inspection statistics vlan 1`	VLAN でダイナミック ARP インスペクションの統計情報を確認します。
ステップ 12	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 13	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。

着信 ARP パケットのレート制限

スイッチの CPU は、ダイナミック ARP インスペクション確認検査を実行します。したがって、DoS 攻撃を阻止するために、着信 ARP パケット数はレート制限されます。

着信 ARP パケットのレートが設定された制限を超えると、スイッチはポートを errdisable ステートにします。errordisable 回復をイネーブルにして、指定されたタイムアウト時間の後にポートがこのステートから自動的に抜け出すようにするまで、ポートはこのステートのままです。

（注） インターフェイス上のレート制限を設定しない限り、インターフェイスの信頼状態を変更することは、レート制限をその信頼状態のデフォルト値に変更することになります。レート制限を設定すると、信頼状態が変更された場合でもインターフェイスはレート制限を保ちます。**no ip arp inspection limit** インターフェイス コンフィギュレーション コマンドを入力すると、インターフェイスはデフォルトのレート制限に戻ります。

着信 ARP パケットのレートを制限するには、次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip arp inspection limit {rate pps [burst interval seconds] | none}**
5. **exit**
6. 次のコマンドを使用します。
   - **errdisable detect cause arp-inspection**
   - **errdisable recovery cause arp-inspection**
   - **errdisable recovery interval** *interval*
7. **exit**
8. 次の show コマンドを使用します。
   - **show ip arp inspection interfaces**
   - **show errdisable recovery**
9. **show running-config**
10. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id*	レート制限されるインターフェイスを指定して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip arp inspection limit {rate pps [burst interval seconds] | none}**	インターフェイス上の着信 ARP 要求および ARP 応答のレートを制限します。デフォルト レートは、信頼できないインターフェイスでは 15 pps、信頼できるインターフェイスでは無制限です。バースト インターバルは 1 秒です。

	コマンドまたはアクション	目的
		キーワードの意味は次のとおりです。 • **rate***pps* には、1 秒あたりに処理される着信パケット数の上限を指定します。有効な範囲は 0 ～ 2048 pps です。 • （任意）**burst interval***seconds* は、レートの高い ARP パケットの有無についてインターフェイスがモニタリングされる間隔（秒）を指定します。指定できる範囲は 1 ～ 15 です。 • **rate none** には、処理可能な着信 ARP パケットのレートに上限を指定しません。
ステップ 5	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	次のコマンドを使用します。 • **errdisable detect cause arp-inspection** • **errdisable recovery cause arp-inspection** • **errdisable recovery interval** *interval*	（任意）ダイナミック ARP インスペクションの errdisable ステートからのエラー回復をイネーブルにし、ダイナミック ARP インスペクションの回復メカニズムで使用する変数を設定します。 デフォルトでは、回復はディセーブルで、回復のインターバルは 300 秒です。 **interval** *interval* には、error-disabled ステートから回復する時間を秒単位で指定します。指定できる範囲は 30 ～ 86400 です。
ステップ 7	**exit**	特権 EXEC モードに戻ります。
ステップ 8	次の show コマンドを使用します。 • **show ip arp inspection interfaces** • **show errdisable recovery**	設定を確認します。
ステップ 9	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 10	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ダイナミック ARP インスペクション検証チェックの実行

ダイナミック ARP インスペクションは、不正な IP/MAC アドレス バインディングを持つ ARP パケットを代行受信し、ログに記録して、廃棄します。宛先 MAC アドレス、送信側および宛先の IP アドレス、および送信元 MAC アドレスで追加検証を実行するように、スイッチを設定できます。

着信 ARP パケットで特定のチェックを実行するには、次の手順を実行します。この手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip arp inspection validate {[src-mac] [dst-mac] [ip]}**
4. **exit**
5. **show ip arp inspection vlan** *vlan-range*
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip arp inspection validate {[src-mac] [dst-mac] [ip]}**	着信 ARP パケットで特定の検査を実行します。デフォルトでは、検証は実行されません。 キーワードの意味は次のとおりです。 • **src-mac** では、イーサネットヘッダーの送信元 MAC アドレスと ARP 本文の送信元 MAC アドレスが比較されます。この検査は、ARP要求およびARP応答の両方に対して実行されます。イネーブルにすると、異なる MAC アドレスを持

	コマンドまたはアクション	目的
		つパケットは無効パケットとして分類され、廃棄されます。 • **dst-mac** では、イーサネットヘッダーの宛先 MAC アドレスと ARP 本文の宛先 MAC アドレスが比較されます。この検査は、ARP応答に対して実行されます。イネーブルにすると、異なる MAC アドレスを持つパケットは無効パケットとして分類され、廃棄されます。 • **ip** では、ARP 本文から、無効な IP アドレスや予期しないIPアドレスがないかを確認します。アドレスには 0.0.0.0、255.255.255.255、およびすべての IP マルチキャスト アドレスが含まれます。送信元 IP アドレスはすべての ARP 要求および ARP 応答内で検査され、宛先 IP アドレスは ARP 応答内だけで検査されます。 少なくとも 1 つのキーワードを指定する必要があります。コマンドを実行するたびに、その前のコマンドの設定は上書きされます。つまり、コマンドがsrc および dst mac の検証をイネーブルにし、別のコマンドが IP 検証だけをイネーブルにすると、2 番めのコマンドによって src および dst mac の検証がディセーブルになります。
ステップ 4	**exit**	特権 EXEC モードに戻ります。
ステップ 5	**show ip arp inspection vlan** *vlan-range*	設定を確認します。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

DAI のモニタリング

DAI をモニターするには、次のコマンドを使用します。

コマンド	説明
clear ip arp inspection statistics	ダイナミック ARP インスペクション統計情報をクリアします。
show ip arp inspection statistics [**vlan** *vlan-range*]	指定の VLAN において、転送されたパケット、廃棄されたパケット、MAC 検証に失敗したパケット、IP 検証に失敗したパケット、ACL によって許可および拒否されたパケット、DHCP によって許可および拒否されたパケットの統計情報を表示します。VLAN が指定されていない場合、または範囲が指定されている場合は、ダイナミック ARP インスペクションがイネーブルにされた（アクティブ）VLAN だけの情報を表示します。
clear ip arp inspection log	ダイナミック ARP インスペクション ログ バッファをクリアします。
show ip arp inspection log	ダイナミック ARP インスペクション ログ バッファの設定と内容を表示します。

show ip arp inspection statistics コマンドでは、スイッチは信頼されたダイナミック ARP インスペクション ポート上の各 ARP 要求および応答パケットの転送済みパケット数を増加させます。スイッチは、送信元 MAC、宛先 MAC、または IP 検証チェックによって拒否された各パケットの ACL または DHCP 許可済みパケット数を増加させ、適切な失敗数を増加させます。

DAI の設定の確認

DAI の設定を表示して確認するには、次のコマンドを使用します。

コマンド	説明
show arp access-list [*acl-name*]	ARP ACL についての詳細情報を表示します。
show ip arp inspection interfaces [interface-id]	指定されたインターフェイスまたはすべてのインターフェイスの ARP パケットの信頼状態およびレート制限を表示します。

コマンド	説明
show ip arp inspection vlan *vlan-range*	指定された VLAN のダイナミック ARP インスペクションの設定および動作ステートを表示します。VLAN が指定されていない場合、または範囲が指定されている場合は、ダイナミック ARP インスペクションがイネーブルにされた（アクティブ）VLAN だけの情報を表示します。

第 68 章

IEEE 802.1x ポートベースの認証の設定

この章では、IEEE 802.1x ポート ベース認証を設定する方法について説明します。IEEE 802.1x 認証は、不正なデバイス（クライアント）によるネットワーク アクセスを防止します。別途記載のないかぎり、スイッチという用語はスタンドアロンスイッチまたはスイッチスタックを意味します。

802.1x ポートベース認証について

802.1x 規格では、一般の人がアクセス可能なポートから不正なクライアントが LAN に接続しないように規制する（適切に認証されている場合を除く）、クライアント/サーバ型のアクセス コントロールおよび認証プロトコルを定めています。認証サーバーがスイッチ ポートに接続する各クライアントを認証したうえで、スイッチまたは LAN が提供するサービスを利用できるようにします。

（注） TACACS は、802.1x 認証ではサポートされていません。

802.1x アクセス コントロールでは、クライアントを認証するまでの間、そのクライアントが接続しているポート経由では Extensible Authentication Protocol over LAN（EAPOL）、Cisco Discovery Protocol（CDP）、およびスパニングツリー プロトコル（STP）トラフィックしか許可されません。認証に成功すると、通常のトラフィックはポートを通過できるようになります。

クライアント セッション	サポートされる最大セッション数
dot1x または MAB クライアント セッションの最大数	2000
Web ベース認証セッションの最大数	2000

クライアント セッション	サポートされる最大セッション数
クリティカル認証 VLAN を有効にしてサーバを再初期化した dot1x セッションの最大数	2000
さまざまなセッション機能が適用される MAB セッションの最大数	2000
サービス テンプレートまたはセッション機能が適用される dot1x セッションの最大数	2000

ポートベース認証プロセス

IEEE 802.1X ポートベース認証を設定するには、認証、認可、およびアカウンティング（AAA）を有効にし、認証方式リストを指定する必要があります。方式リストは、ユーザ認証のためにクエリー送信を行う手順と認証方式を記述したものです。

AAA プロセスは認証から始まります。802.1x ポートベース認証がイネーブルであり、クライアントが 802.1x 準拠のクライアント ソフトウェアをサポートしている場合、次のイベントが発生します。

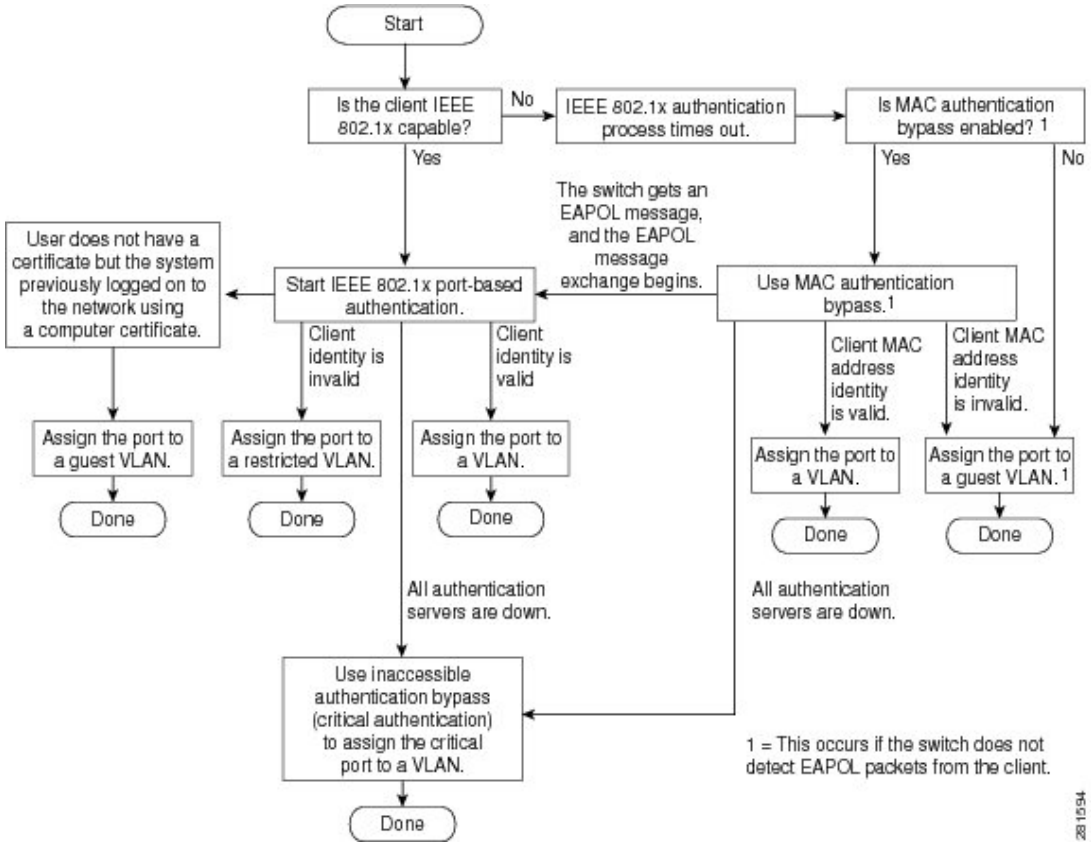
- クライアント ID が有効で 802.1x 認証に成功した場合、スイッチはクライアントにネットワークへのアクセスを許可します。
- EAPOL メッセージ交換の待機中に 802.1x 認証がタイムアウトし、MAC 認証バイパスがイネーブルの場合、スイッチはクライアント MAC アドレスを認証用に使用します。このクライアント MAC アドレスが有効で認証に成功した場合、スイッチはクライアントにネットワークへのアクセスを許可します。クライアント MAC アドレスが無効で認証に失敗した場合、ゲスト VLAN が設定されていれば、スイッチはクライアントに限定的なサービスを提供するゲスト VLAN を割り当てます。
- スイッチが 802.1x 対応クライアントから無効な ID を取得し、制限付き VLAN が指定されている場合、スイッチはクライアントに限定的なサービスを提供する制限付き VLAN を割り当てることができます。
- RADIUS 認証サーバーが使用できず（ダウンしていて）アクセスできない認証バイパスがイネーブルの場合、スイッチは、RADIUS 設定 VLAN またはユーザー指定アクセス VLAN で、ポートをクリティカル認証ステートにして、クライアントにネットワークのアクセスを許可します。

（注） アクセスできない認証バイパスは、クリティカル認証、または AAA 失敗ポリシーとも呼ばれます。

ポートで Multi Domain Authentication（MDA）が有効になっている場合、音声許可に該当する例外をいくつか伴ったフローを使用できます。

図 *108 :* 認証フローチャート

次の図は認証プロセスを示します。



次の状況のいずれかが発生すると、スイッチはクライアントを再認証します。

- 定期的な再認証がイネーブルで、再認証タイマーの期限が切れている場合。

  スイッチ固有の値を使用するか、RADIUS サーバーからの値に基づいて再認証タイマーを設定できます。

  RADIUS サーバーを使用した 802.1x 認証の後で、スイッチは Session-Timeout RADIUS 属性（Attribute[27]）、および Termination-Action RADIUS 属性（Attribute[29]）に基づいてタイマーを使用します。

  Session-Timeout RADIUS 属性（Attribute[27]）には再認証が行われるまでの時間を指定します。指定できる範囲は 1 ～ 65535 秒です。

  Termination-Action RADIUS 属性（Attribute[29]）には、再認証中に行われるアクションを指定します。アクションは *Initialize* および *ReAuthenticate* に設定できます。アクションに *Initialize*（属性値は *DEFAULT*）を設定した場合、802.1x セッションは終了し、認証中、接続は失われます。アクションに *ReAuthenticate*（属性値は RADIUS-Request）を設定した場合、セッションは再認証による影響を受けません。

- クライアントを手動で再認証するには、**dot1x re-authenticate interface** *interface-id* 特権 EXEC コマンドを入力します。

ポートベース認証の開始およびメッセージ交換

802.1x認証中に、スイッチまたはクライアントは認証を開始できます。**authentication port-control auto** インターフェイス コンフィギュレーション コマンドを使用してポート上で認証をイネーブルにすると、スイッチは、リンクステートがダウンからアップに移行したときに認証を開始し、ポートがアップしていて認証されていない場合は定期的に認証を開始します。スイッチはクライアントにEAP-Request/Identityフレームを送信し、そのIDを要求します。クライアントはフレームを受信すると、EAP-Response/Identityフレームで応答します。

ただし、クライアントが起動時にスイッチからのEAP-Request/Identityフレームを受信しなかった場合、クライアントはEAPOL-Startフレームを送信して認証を開始できます。このフレームはスイッチに対し、クライアントの識別情報を要求するように指示します。

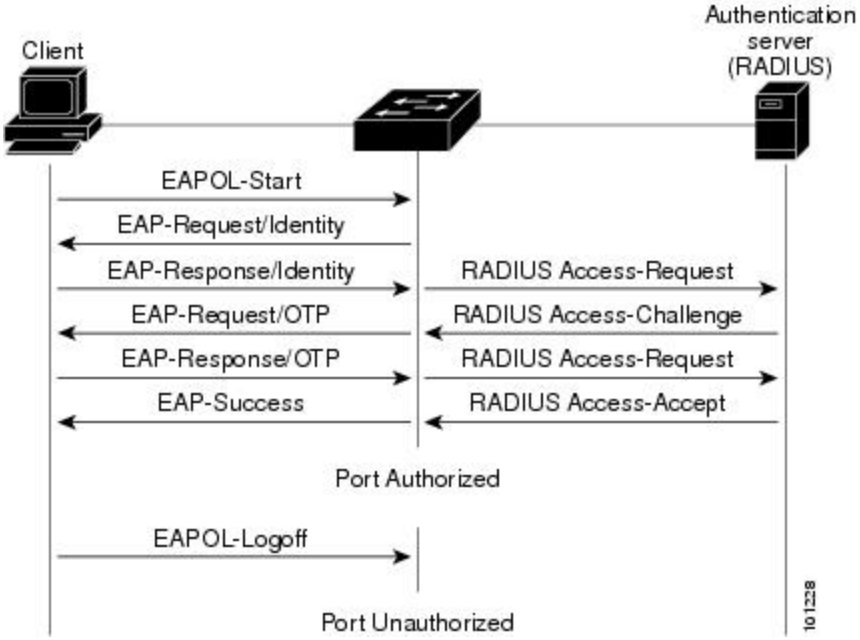
（注） ネットワーク アクセス デバイスで802.1x認証がイネーブルに設定されていない、またはサポートされていない場合には、クライアントからのEAPOLフレームはすべて廃棄されます。クライアントが認証の開始を3回試みてもEAP-Request/Identityフレームを受信しなかった場合、クライアントはポートが許可ステートであるものとしてフレームを送信します。ポートが許可ステートであるということは、クライアントの認証が成功したことを実質的に意味します。

クライアントが自らの識別情報を提示すると、スイッチは仲介デバイスとしての役割を開始し、認証が成功または失敗するまで、クライアントと認証サーバーの間でEAPフレームを送受信します。認証が成功すると、スイッチポートは許可ステートになります。認証に失敗した場合、認証が再試行されるか、ポートが限定的なサービスを提供するVLANに割り当てられるか、あるいはネットワーク アクセスが許可されないかのいずれかになります。

実際に行われるEAPフレーム交換は、使用する認証方式によって異なります。

図 *109* : メッセージ交換

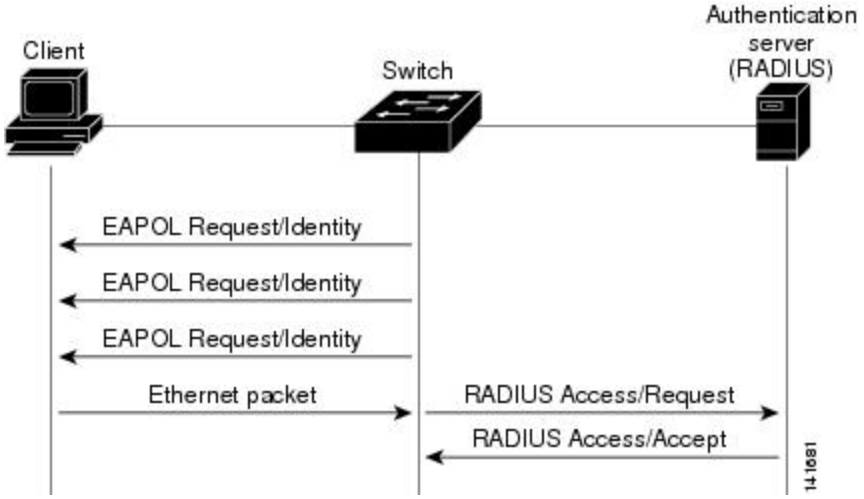
次の図に、クライアントが RADIUS サーバとの間で OTP（ワンタイム パスワード）認証方式を使用する際に行われるメッセージ交換を示します。



EAPOL メッセージ交換の待機中に 802.1x 認証がタイムアウトし、MAC 認証バイパスがイネーブルの場合、スイッチはクライアントからイーサネットパケットを検出するとそのクライアントを認証できます。スイッチは、クライアントの MAC アドレスを ID として使用し、RADIUS サーバーに送信される RADIUS Access/Request フレームにこの情報を保存します。サーバーがスイッチに RADIUS Access/Accept フレームを送信（認証が成功）すると、ポートが許可されます。認証に失敗してゲスト VLAN が指定されている場合、スイッチはポートをゲスト VLAN に割り当てます。イーサネット パケットの待機中にスイッチが EAPOL パケットを検出すると、スイッチは MAC 認証バイパス プロセスを停止して、802.1x 認証を開始します。

図 *110 : MAC* 認証バイパス中のメッセージ交換

次の図に、MAC 認証バイパス中のメッセージ交換を示します。



ポートベース認証の認証マネージャ

ポートベース認証方法

表 *140 : 802.1x* 機能

認証方法	モード			
	シングル ホスト	マルチ ホスト	**MDA**	複数認証
802.1x	VLAN 割り当て ユーザ単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	VLAN 割り当て	VLAN 割り当て ユーザー単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	VLAN 割り当て ユーザー単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL

認証方法	モード			
	シングル ホスト	マルチ ホスト	MDA	複数認証
MAC 認証バイパス	VLAN 割り当て ユーザ単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	VLAN 割り当て	VLAN 割り当て ユーザー単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	VLAN 割り当て ユーザー単位 ACL Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL
スタンドアロン Web 認証	プロキシ ACL、Filter-ID 属性、ダウンロード可能な ACL			
NAC レイヤ 2 IP 検証	Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL	Filter-ID 属性 ダウンロード可能 ACL リダイレクト URL
フォールバック方式としての Web 認証	プロキシ ACL Filter-ID 属性 ダウンロード可能 ACL	プロキシ ACL Filter-ID 属性 ダウンロード可能 ACL	プロキシ ACL Filter-ID 属性 ダウンロード可能 ACL	プロキシ ACL Filter-ID 属性 ダウンロード可能 ACL

[16] Cisco IOS リリース 12.2(50)SE 以降でサポートされています。
[17] 802.1x 認証をサポートしないクライアント用。

ユーザー単位 ACL および Filter-Id

（注） **any** は、ACL の発信元としてだけ設定できます。

（注） マルチホスト モードで設定された ACL では、ステートメントの発信元部分は *any* でなければなりません。（たとえば、 **permit icmp** *any* **host 10.10.1.1**）。

（注） Filter-Id としてロールベース ACL を使用することは推奨されません。

定義された ACL の発信元ポートには **any** を指定する必要があります。指定しない場合、ACL は適用できず、認証は失敗します。シングルホストは唯一例外的に後方互換性をサポートします。

MDA 対応ポートおよびマルチ認証ポートでは、複数のホストを認証できます。ホストに適用される ACL ポリシーは、別のホストのトラフィックには影響を与えません。マルチ ホスト ポートで認証されるホストが1つだけで、他のホストが認証なしでネットワーク アクセスを取得する場合、発信元アドレスに any を指定することで、最初のホストの ACL ポリシーを他の接続ホストに適用できます。

ポートベース認証マネージャ CLI コマンド

認証マネージャ インターフェイス コンフィギュレーション コマンドは、802.1x、MAC 認証バイパスおよび Web 認証など、すべての認証方法を制御します。認証マネージャ コマンドは、接続ホストに適用される認証方法のプライオリティと順序を決定します。

認証マネージャ コマンドは、ホスト モード、違反モードおよび認証タイマーなど、一般的な認証機能を制御します。一般的な認証コマンドには、**authentication host-mode**、**authentication violation**、および **authentication timer** インターフェイス コンフィギュレーション コマンドが含まれます。

802.1x 専用コマンドは、先頭に **dot1x** キーワードが付きます。たとえば、**authentication port-control auto** インターフェイス コンフィギュレーション コマンドは、インターフェイスでの認証をイネーブルにします。

スイッチでの dot1x を無効にするには、**no dot1x system-auth-control** コマンドを使用して、設定をグローバルに削除し、設定されているすべてのインターフェイスからも削除します。

（注） 802.1x 認証がグローバルにディセーブル化されても、Web 認証など他の認証方法はそのポートでイネーブルのままです。

authentication manager コマンドは、以前の 802.1x コマンドと同じ機能を提供します。

認証マネージャが生成する冗長なシステム メッセージをフィルタリングすると、通常は、フィルタリングされた内容が認証の成功に結びつきます。802.1x 認証および MAB 認証の冗長なメッセージをフィルタリングすることもできます。認証方式ごとに異なるコマンドが用意されています。

- **no authentication logging verbose** グローバル コンフィギュレーション コマンドは、認証マネージャからの詳細メッセージをフィルタリングします。
- **no dot1x logging verbose** グローバル コンフィギュレーション コマンドは、802.1x 認証の詳細メッセージをフィルタリングします。
- **no mab logging verbose** グローバル コンフィギュレーション コマンドは、MAC 認証バイパス（MAB）の詳細メッセージをフィルタリングします。

表 *141 :* 認証マネージャ コマンドおよび以前の *802.1x* コマンド

Cisco IOS Release 12.2(50)SE 以降での認証マネージャ コマンド	Cisco IOS Release 12.2(46)SE 以前での同等の 802.1x コマンド	説明
authentication control-direction {**both** | **in**}	**dot1x control-direction** {**both** | **in**}	Wake-on-LAN（WoL）機能を使用して 802.1x 認証をイネーブルにし、ポート制御を単一方向または双方向に設定します。
authentication event	**dot1x auth-fail vlan** **dot1x critical (interface configuration)** **dot1x guest-vlan6**	ポート上で制限付き VLAN をイネーブルにします。 アクセス不能認証バイパス機能をイネーブルにします。 アクティブ VLAN を 802.1x ゲスト VLAN として指定します。
authentication fallback *fallback-profile*	**dot1x fallback** *fallback-profile*	802.1x 認証をサポートしていないクライアント用に、Web 認証をフォールバック方式として使用するようにポートを設定します。
authentication host-mode[**multi-auth** |**multi-domain** |**multi-host** |**single-host**]	**dot1x host-mode** {**single-host** | **multi-host** | **multi-domain**}	802.1x 許可ポートで単一のホスト（クライアント）または複数のホストの接続を許可します。
authentication order	**mab**	使用される認証方法の順序を柔軟に定義できるようにします。
authentication periodic	**dot1x reauthentication**	クライアントの定期的再認証をイネーブルにします。
authentication port-control {**auto** | **force-authorized** | **force-un authorized**}	**dot1x port-control** {**auto** | **force-authorized** | **force-unauthorized**}	ポートの許可ステートの手動制御をイネーブルにします。
authentication timer	**dot1x timeout**	802.1x タイマーを設定します。

Cisco IOS Release 12.2(50)SE 以降での認証マネージャ コマンド	Cisco IOS Release 12.2(46)SE 以前での同等の 802.1x コマンド	説明
authentication violation {**protect** | **restrict** | **shutdown**}	**dot1x violation-mode** {**shutdown** | **restrict** | **protect**}	新しいデバイスがポートに接続された場合、または最大数のデバイスがポートに接続された後に新しいデバイスがそのポートに接続された場合に発生する違反モードを設定します。

許可ステートおよび無許可ステートのポート

802.1x認証中に、スイッチのポートステートによって、スイッチはネットワークへのクライアントアクセスを許可します。ポートは最初、無許可ステートです。このステートでは、音声VLAN（仮想LAN）ポートとして設定されていないポートは802.1x認証、CDP、およびSTPパケットを除くすべての入力および出力トラフィックを禁止します。クライアントの認証が成功すると、ポートは許可ステートに変更し、クライアントのトラフィック送受信を通常どおりに許可します。ポートが音声VLANポートとして設定されている場合、VoIPトラフィックおよび802.1xプロトコルパケットが許可された後クライアントが正常に認証されます。

（注） CDPバイパスはサポートされていないため、ポートがerror-disabledステートになる場合があります。

802.1xをサポートしていないクライアントが、無許可ステートの802.1xポートに接続すると、スイッチはそのクライアントの識別情報を要求します。この状況では、クライアントは要求に応答せず、ポートは引き続き無許可ステートとなり、クライアントはネットワークアクセスを許可されません。

反対に、802.1x対応のクライアントが、802.1x標準が稼働していないポートに接続すると、クライアントはEAPOL-Startフレームを送信して認証プロセスを開始します。応答がなければ、クライアントは同じ要求を所定の回数だけ送信します。応答がないので、クライアントはポートが許可ステートであるものとしてフレーム送信を開始します。

authentication port-controlインターフェイスコンフィギュレーションコマンドおよび次のキーワードを使用して、ポートの許可ステートを制御できます。

- **force-authorized**：802.1x認証をディセーブルにし、認証情報の交換を必要とせずに、ポートを許可ステートに変更します。ポートはクライアントとの802.1xベース認証を行わずに、通常のトラフィックを送受信します。これがデフォルト設定です。
- **force-unauthorized**：ポートが無許可ステートのままになり、クライアントからの認証の試みをすべて無視します。スイッチはポートを介してクライアントに認証サービスを提供できません。

- **auto**：802.1x 認証をイネーブルにします。ポートは最初、無許可ステートであり、ポート経由で送受信できるのは EAPOL フレームだけです。ポートのリンク ステートがダウンからアップに変更した際、または EAPOL-Start フレームを受信した際に、認証プロセスが開始されます。スイッチはクライアントの識別情報を要求し、クライアントと認証サーバとの間で認証メッセージのリレーを開始します。スイッチはクライアントの MAC アドレスを使用して、ネットワーク アクセスを試みる各クライアントを一意に識別します。

クライアントが認証に成功すると（認証サーバから Accept フレームを受信すると）、ポートが許可ステートに変わり、認証されたクライアントからの全フレームがポート経由での送受信を許可されます。認証が失敗すると、ポートは無許可ステートのままですが、認証を再試行することはできます。認証サーバに到達できない場合、スイッチは要求を再送信します。所定の回数だけ試行してもサーバから応答が得られない場合には、認証が失敗し、ネットワーク アクセスは許可されません。

クライアントはログオフするとき、EAPOL-Logoff メッセージを送信します。このメッセージによって、スイッチ ポートが無許可ステートになります。

ポートのリンク ステートがアップからダウンに変更した場合、または EAPOL-Logoff フレームを受信した場合に、ポートは無許可ステートに戻ります。

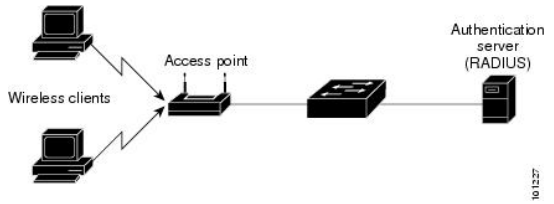
802.1X のホスト モード

802.1x ポートは、シングル ホスト モードまたはマルチ ホスト モードで設定できます。シングル ホスト モードでは、802.1x 対応のスイッチ ポートに接続できるのはクライアント 1 つだけです。スイッチは、ポートのリンク ステートがアップに変化したときに、EAPOL フレームを送信することでクライアントを検出します。クライアントがログオフしたとき、または別のクライアントに代わったときには、スイッチはポートのリンク ステートをダウンに変更し、ポートは無許可ステートに戻ります。

マルチホスト モードでは、複数のホストを単一の 802.1x 対応ポートに接続できます。このモードでは、接続されたクライアントのうち 1 つが許可されれば、クライアントすべてのネットワーク アクセスが許可されます。ポートが無許可ステートになると（再認証が失敗するか、または EAPOL-Logoff メッセージを受信した場合）、スイッチは接続しているクライアントのネットワーク アクセスをすべて禁止します。

このトポロジでは、ワイヤレス アクセス ポイントが接続しているクライアントの認証を処理し、スイッチに対してクライアントとしての役割を果たします。

図 *111* : マルチ ホスト モードの例



（注） すべてのホスト モードで、ポートベース認証が設定されている場合、ライン プロトコルは許可の前にアップのままです。

スイッチはマルチドメイン認証（MDA）をサポートしています。これにより、データ装置とIP Phoneなどの音声装置（シスコ製品またはシスコ以外の製品）の両方を同じスイッチ ポートに接続できます。

802.1x マルチ認証モード

マルチ認証（multiauth）モードでは、データ VLAN および音声 VLAN で複数のクライアントを認証できます。各ホストは個別に認証されます。マルチ認証ポートで認証できるデータデバイスまたは音声デバイスの数には制限はありません。

ハブまたはアクセス ポイントが 802.1x 対応ポートに接続されている場合、接続されている各クライアントを認証する必要があります。802.1x 以外のデバイスでは、MAC 認証バイパスまたは Web 認証をホスト単位認証フォールバック メソッドとして使用し、単一のポートで異なる方法で異なるホストを認証できます。

（注） ポートがマルチ認証モードの場合、認証失敗 VLAN 機能はアクティブになりません。

次の条件で、RADIUS サーバから提供された VLAN をマルチ認証モードで割り当てることができます。

- ホストがポートで最初に許可されたホストであり、RADIUS サーバが VLAN 情報を提供している。
- 後続のホストが、動作 VLAN に一致する VLAN を使用して許可される。
- ホストは VLAN が割り当てられていないポートで許可され、後続のホストでは VLAN 割り当てが設定されていないか、VLAN 情報が動作 VLAN と一致している。
- ポートで最初に許可されたホストにはグループ VLAN が割り当てられ、後続のホストでは VLAN 割り当てが設定されていないか、グループ VLAN がポート上のグループ VLAN と一致している。後続のホストが、最初のホストと同じ VLAN グループの VLAN を使用する必要がある。VLAN リストが使用されている場合、すべてのホストは VLAN リストで指定された条件に従う。
- VLAN がポート上のホストに割り当てられると、後続のホストは一致する VLAN 情報を持つ必要があり、この情報がなければポートへのアクセスを拒否される。
- ゲスト VLAN または認証失敗 VLAN をマルチ認証モードに設定できない。
- クリティカル認証 VLAN の動作が、マルチ認証モード用に変更されない。ホストが認証を試みたときにサーバに到達できない場合、許可されたすべてのホストは、設定された VLAN で再初期化される。

ユーザーごとのマルチ認証 VLAN 割り当て

ユーザーごとのマルチ認証VLAN割り当て機能を使用すると、単一の設定済みアクセスVLANを持つポート上のクライアントに割り当てられたVLANに基づいて複数の運用アクセスVLANを作成することができます。データ ドメインに関連付けられたすべての VLAN に対するトラフィックが dot1q とタグ付けされていないアクセス ポートとして設定されているポートおよびこれらの VLAN は、ネイティブ VLAN として処理されます。

マルチ認証ポート 1 つあたりのホストの数は 8 ですが、さらに多くのホストが存在する場合があります。

次のシナリオは、ユーザーごとのマルチ認証 VLAN 割り当てに関連しています。

シナリオ 1

ハブがアクセス ポートに接続されている場合、およびポートがアクセス VLAN（V0）で設定されている場合。

ホスト（H1）は、ハブを介してVLAN（V1）に割り当てられます。ポートの運用VLANはV1に変更されます。この動作は、単一ホスト ポートまたはマルチ ドメイン認証ポートで同様です。

2 番目のホスト（H2）が接続され、VLAN（V2）に割り当てられる場合、ポートには 2 つの運用VLANがあります（V1およびV2）。H1 と H2 がタグなし入力トラフィックを送信すると、H1 トラフィックは VLAN（V1）に、H2 トラフィックは VLAN（V2）にマッピングされ、VLAN（V1）および VLAN（V2）のポートからの出トラフィックはすべてタグなしになります。

両方のホスト H1 と H2 がログアウトするか、またはセッションがなんらかの理由で削除されると、VLAN（V1）と VLAN（V2）がポートから削除され、設定された VLAN（V0）がポートに復元されます。

シナリオ 2

ハブがアクセス ポートに接続されている場合、およびポートがアクセス VLAN（V0）で設定されている場合。ホスト（H1）は、ハブを介して VLAN（V1）に割り当てられます。ポートの運用 VLAN は V1 に変更されます。

2 番目のホスト（H2）が接続され明示的な VLAN ポリシーなしで承認されると、H2 はポート上で復元される設定済み VLAN（V0）を使用することを予期されます。2 つの運用 VLAN、VLAN（V0）および VLAN（V1）からの出トラフィックはすべてタグなしになります。

ホスト（H2）がログアウトするか、またはセッションがなんらかの理由で削除されると、設定されたVLAN（V0）がポートから削除され、VLAN（V1）がそのポートでの唯一の運用VLANになります。

シナリオ 3

ハブがオープン モードでアクセス ポートに接続されている場合、およびポートがアクセス VLAN（V0）で設定されている場合。

ホスト（H1）は、ハブを介してVLAN（V1）に割り当てられます。ポートの運用VLANはV1に変更されます。2 番目のホスト（H2）が接続され無許可のままだと、オープン モードにより、運用 VLAN（V1）に引き続きアクセスできます。

ホスト H1 がログアウトするか、またはセッションがなんらかの理由で削除されると、VLAN（V1）はポートから削除され、ホスト（H2）は VLAN（V0）に割り当てられます。

（注） オープン モードと VLAN 割り当ての組み合わせは、ホスト（H2）に悪影響を与えます。そのホストは VLAN（V1）に対応するサブネット内に IP アドレスを含んでいるからです。

ユーザーごとのマルチ認証 VLAN 割り当ての制限

ユーザーごとのマルチ認証VLAN割り当て機能では、複数のVLANからの出トラフィックは、ホストが自分宛てではないトラフィックを受信するポート上ではタグなしになります。これは、ブロードキャストおよびマルチキャスト トラフィックで問題になる可能性があります。

- **IPv4 ARP**：ホストは他のサブネットからの ARP パケットを受信します。これは、IP アドレス範囲が重複する異なる仮想ルーティングおよび転送（VRF）テーブルの 2 個のサブネットがポート上でアクティブな場合に問題となります。ホスト ARP キャッシュのエントリが無効になる可能性があります。
- **IPv6 制御パケット**：IPv6 の導入環境では、ルータ アドバタイズメント（RA）は、その受信を想定されていないホストによって処理されます。ある VLAN からのホストが別の VLAN からの RA を受信すると、ホストはそれ自身に間違った IPv6 アドレスを割り当てます。このようなホストは、ネットワークにアクセスできません。

  回避策は、IPv6 ファースト ホップ セキュリティをイネーブルにして、ブロードキャスト ICMPv6 パケットがユニキャストに変換され、マルチ認証がイネーブルのポートから送信されるようにすることです。パケットは VLAN に属するマルチ認証ポートの各クライアント用に複製され、宛先 MAC が個々のクライアントに設定されます。1 つの VLAN を持つポートで、ICMPv6 パケットは正常にブロードキャストされます。
- **IP マルチキャスト**：送信先のマルチキャスト グループへのマルチキャスト トラフィックは、異なる VLAN 上のホストがそのマルチキャスト グループに参加している場合それらの VLAN 用に複製されます。異なる VLAN の 2 つのホストが（同じマルチ認証ポート上の）マルチキャスト グループに参加している場合、各マルチキャスト パケットのコピー 2 部がそのポートから送信されます。

MAC 移動

あるスイッチ ポートでMACアドレスが認証されると、そのアドレスは同じスイッチの別の認証マネージャ対応ポートでは許可されません。スイッチが同じ MAC アドレスを別の認証マネージャ対応ポートで検出すると、そのアドレスは許可されなくなります。

場合によっては、MAC アドレスを同じスイッチ上のポート間で移動する必要があります。たとえば、認証ホストとスイッチ ポート間に別のデバイス（ハブまたは IP Phone など）がある場合、ホストをデバイスから接続して、同じスイッチの別のポートに直接接続する必要があります。

デバイスが新しいポートで再認証されるように、MAC 移動をグローバルにイネーブルにできます。ホストが別のポートに移動すると、最初のポートのセッションが削除され、ホストは新しいポートで再認証されます。MAC 移動はすべてのホスト モードでサポートされます（認証ホストは、ポートでイネーブルにされているホストモードに関係なく、スイッチの任意のポートに移動できます）。MAC アドレスがあるポートから別のポートに移動すると、スイッチは元のポートで認証済みセッションを終了し、新しいポートで新しい認証シーケンスを開始します。MAC 移動の機能は、音声およびデータ ホストの両方に適用されます。

（注） オープン認証モードでは、MAC アドレスは、新しいポートでの許可を必要とせずに、元のポートから新しいポートへただちに移動します。

MAC 置換

MAC 置換機能は、ホストが、別のホストがすでに認証済みであるポートに接続しようとすると発生する違反に対処するように設定できます。

（注） 違反はマルチ認証モードでは発生しないため、マルチ認証モードのポートにこの機能は適用されません。マルチホスト モードで認証が必要なのは最初のホストだけなので、この機能はこのモードのポートには適用されません。

replace キーワードを指定して **authentication violation** インターフェイス コンフィギュレーション コマンドを設定すると、マルチドメインモードのポートでの認証プロセスは、次のようになります。

- 既存の認証済みMACアドレスを使用するポートで新しいMACアドレスが受信されます。
- 認証マネージャは、ポート上の現在のデータ ホストの MAC アドレスを、新しい MAC アドレスで置き換えます。
- 認証マネージャは、新しい MAC アドレスに対する認証プロセスを開始します。
- 認証マネージャによって新しいホストが音声ホストであると判断された場合、元の音声ホストは削除されます。

ポートがオープン認証モードになっている場合、MACアドレスはただちにMACアドレステーブルに追加されます。

802.1x アカウンティング

802.1x標準では、ユーザの認証およびユーザのネットワーク アクセスに対する許可方法を定義しています。ただし、ネットワークの使用法についてはトラッキングしません。802.1xアカウンティングは、デフォルトでディセーブルです。802.1xアカウンティングをイネーブルにすると、次の処理を 802.1x 対応のポート上でモニタできます。

- 正常にユーザを認証します。
- ユーザがログ オフします。
- リンクダウンが発生します。
- 再認証が正常に行われます。
- 再認証が失敗します。

スイッチは802.1xアカウンティング情報を記録しません。その代わり、スイッチはこの情報をRADIUSサーバに送信します。RADIUSサーバは、アカウンティング メッセージを記録するように設定する必要があります。

802.1x アカウンティング属性値ペア

RADIUS サーバに送信された情報は、属性値（AV）ペアの形式で表示されます。これらの AV ペアのデータは、各種アプリケーションによって使用されます（たとえば課金アプリケーションの場合、RADIUS パケットの Acct-Input-Octets または Acct-Output-Octets 属性の情報が必要です）。

AV ペアは、802.1x アカウンティングが設定されているスイッチによって自動的に送信されます。次の種類の RADIUS アカウンティング パケットがスイッチによって送信されます。

- START：新規ユーザー セッションが始まると送信されます。
- INTERIM：既存のセッションが更新されると送信されます。
- STOP：セッションが終了すると送信されます。

スイッチによって送信された AV ペアは、**debug radius accounting** 特権 EXEC コマンドを入力して表示できます。このコマンドの詳細については、『*Cisco IOS Debug Command Reference, Release 12.4*』を参照してください。

次の表に、AV ペアおよびスイッチによって送信される AV ペアの条件を示します。

表 *142 :* アカウンティング *AV* ペア

属性番号	AV ペア名	START	INTERIM	STOP
属性 [1]	User-Name	送信	送信	送信
属性 [4]	NAS-IP-Address	送信	送信	送信

属性番号	AV ペア名	START	INTERIM	STOP
属性 [5]	NAS-Port	送信	送信	送信
属性 [8]	Framed-IP-Address	非送信	条件に応じて送信[18]	条件に応じて送信
属性 [25]	Class	送信	送信	送信
属性 [30]	Called-Station-ID	送信	送信	送信
属性 [31]	Calling-Station-ID	送信	送信	送信
属性 [40]	Acct-Status-Type	送信	送信	送信
属性 [41]	Acct-Delay-Time	送信	送信	送信
属性 [42]	Acct-Input-Octets	非送信	送信	送信
属性 [43]	Acct-Output-Octets	非送信	送信	送信
属性 [47]	Acct-Input-Packets	非送信	送信	送信
属性 [48]	Acct-Output-Packets	非送信	送信	送信
属性 [44]	Acct-Session-ID	送信	送信	送信
属性 [45]	Acct-Authentic	送信	送信	送信
属性 [46]	Acct-Session-Time	非送信	送信	送信
属性 [49]	Acct-Terminate-Cause	非送信	非送信	送信
属性 [61]	NAS-Port-Type	送信	送信	送信

[18] 有効な静的 IP アドレスが設定されているか、ホストに対する Dynamic Host Control Protocol（DHCP）バインディングが DHCP スヌーピング バインディング テーブルに存在している場合に、Framed-IP-Address の AV ペアが送信されます。

802.1x 準備状態チェック

802.1x 準備状態チェックは、すべてのスイッチ ポートの 802.1x アクティビティをモニタリングし、802.1x をサポートするポートに接続されているデバイスの情報を表示します。この機能を使用して、スイッチ ポートに接続されているデバイスが 802.1x に対応できるかどうかを判別できます。802.1x 機能をサポートしていないデバイスでは、MAC 認証バイパスまたは Web 認証などの代替認証を使用します。

この機能が有用なのは、クライアントのサプリカントで NOTIFY EAP 通知パケットでのクエリーがサポートされている場合だけです。クライアントは、802.1x タイムアウト値内に応答しなければなりません。

スイッチと RADIUS サーバー間の通信

RADIUS セキュリティ サーバは、ホスト名または IP アドレス、ホスト名と特定の UDP ポート番号、または IP アドレスと特定の UDP ポート番号によって識別します。IP アドレスと UDP ポート番号の組み合わせによって、一意の ID が作成され、同一 IP アドレスのサーバ上にある複数の UDP ポートに RADIUS 要求を送信できるようになります。同じ RADIUS サーバ上の異なる 2 つのホスト エントリに同じサービス（たとえば認証）を設定した場合、2 番めに設定されたホスト エントリは、最初に設定されたホスト エントリのフェールオーバー バックアップとして動作します。RADIUS ホスト エントリは、設定した順序に従って試行されます。

VLAN 割り当てを使用した 802.1x 認証

スイッチは、VLAN 割り当てを使用した 802.1x 認証をサポートしています。ポートの 802.1x 認証が成功すると、RADIUS サーバーは VLAN 割り当てを送信し、スイッチ ポートを設定します。RADIUS サーバーデータベースは、ユーザー名と VLAN のマッピングを維持し、スイッチ ポートに接続するクライアントのユーザー名に基づいて VLAN を割り当てます。この機能を使用して、特定のユーザーのネットワーク アクセスを制限できます。

音声デバイス認証は、Cisco IOS Release 12.2(37)SE のマルチドメイン ホスト モードでサポートされています。Cisco IOS Release 12.2(40)SE 以降、音声デバイスが許可されており、RADIUS サーバーが許可された VLAN を返した場合、割り当てられた音声 VLAN 上でパケットを送受信するようにポート上の音声 VLAN が設定されます。音声 VLAN 割り当ては、マルチドメイン認証（MDA）対応のポートでのデータ VLAN 割り当てと同じように機能します。

スイッチと RADIUS サーバー上で設定された場合、VLAN 割り当てを使用した 802.1x 認証には次の特性があります。

- RADIUS サーバから VLAN が提供されない場合、または 802.1x 認証がディセーブルの場合、認証が成功するとポートはアクセス VLAN に設定されます。アクセス VLAN とは、アクセス ポートに割り当てられた VLAN です。このポート上で送受信されるパケットはすべて、この VLAN に所属します。
- 802.1x 認証がイネーブルで、RADIUS サーバからの VLAN 情報が有効でない場合、認証に失敗して、設定済みの VLAN が引き続き使用されます。これにより、設定エラーによって不適切な VLAN に予期せぬポートが現れることを防ぎます。

  設定エラーには、ルーテッド ポートの VLAN、間違った VLAN ID、存在しないまたは内部（ルーテッド ポート）の VLAN ID、RSPAN VLAN、シャットダウンしている VLAN、あるいは一時停止している VLAN ID の指定などがあります。マルチドメイン ホスト ポートの場合、設定エラーには、設定済みまたは割り当て済み VLAN ID と一致するデータ VLAN の割り当て試行（またはその逆）のために発生するものもあります。
- 802.1x 認証がイネーブルで、RADIUS サーバからのすべての情報が有効の場合、許可されたデバイスは認証後、指定した VLAN に配置されます。
- 802.1x ポートでマルチ ホスト モードがイネーブルの場合、すべてのホストは最初に認証されたホストと同じ VLAN（RADIUS サーバにより指定）に配置されます。

- ポート セキュリティをイネーブル化しても、RADIUS サーバが割り当てられた VLAN の動作には影響しません。
- 802.1x 認証がポートでディセーブルの場合、設定済みのアクセス VLAN と設定済みの音声 VLAN に戻ります。
- 802.1x ポートが認証され、RADIUS サーバによって割り当てられた VLAN に配置されると、そのポートのアクセス VLAN 設定への変更は有効になりません。マルチドメイン ホストの場合、ポートが完全にこれらの例外で許可されている場合、同じことが音声デバイスに適用されます。
  - あるデバイスでVLAN設定を変更したことにより、他のデバイスに設定済みまたは割り当て済みのVLANと一致した場合、ポート上の全デバイスの認証が中断して、データおよび音声デバイスに設定済みのVLANが一致しなくなるような有効な設定が復元されるまで、マルチドメイン ホスト モードがディセーブルになります。
  - 音声デバイスが許可されて、ダウンロードされた音声VLANを使用している場合、音声 VLAN設定を削除したり設定値を *dot1p* または *untagged* に修正したりすると、音声デバイスが未許可になり、マルチドメインホストモードがディセーブルになります。

ポートが、強制許可（force-authorized）ステート、強制無許可（force-unauthorized）ステート、無許可ステート、またはシャットダウンステートの場合、ポートは設定済みのアクセスVLANに配置されます。

VLAN 割り当てを設定するには、次の作業を実行する必要があります。

- **network** キーワードを使用して AAA 認証をイネーブルにし、RADIUS サーバからのインターフェイス設定を可能にします。
- 802.1x認証をイネーブルにします。（アクセスポートで802.1x認証を設定すると、VLAN割り当て機能は自動的にイネーブルになります）。
- RADIUSサーバにベンダー固有のトンネル属性を割り当てます。RADIUSサーバーは次の属性をスイッチに返す必要があります。
  - [64] Tunnel-Type = VLAN
  - [65] Tunnel-Medium-Type = 802
  - [81] Tunnel-Private-Group-ID = VLAN 名または VLAN ID
  - [83] Tunnel-Preference

  属性 [64] は、値 *VLAN*（タイプ 13）でなければなりません。属性 [65] は、値*802*（タイプ 6）でなければなりません。属性 [81] は、IEEE 802.1x 認証ユーザに割り当てられた *VLAN* 名または *VLAN ID* を指定します。

ユーザー単位 ACL を使用した 802.1x 認証

ユーザー単位アクセス コントロール リスト（ACL）をイネーブルにして、異なるレベルのネットワーク アクセスおよびサービスを 802.1x 認証ユーザーに提供できます。RADIUS サーバーは、802.1x ポートに接続されるユーザーを認証する場合、ユーザー ID に基づいて ACL 属性を受け取り、これらをスイッチに送信します。スイッチは、ユーザー セッションの期間中、その属性を 802.1x ポートに適用します。セッションが終了すると、認証が失敗した場合、またはリンクダウン状態の発生時に、ユーザー単位 ACL 設定が削除されます。スイッチは、RADIUS 指定の ACL を実行コンフィギュレーションには保存しません。ポートが無許可の場合、スイッチはそのポートから ACL を削除します。

ユーザーは同一のスイッチ上で、ルータ ACL および入力ポート ACL を使用できます。ただし、ポートの ACL はルータ ACL より優先されます。入力ポート ACL を VLAN に属するインターフェイスに適用する場合、ポート ACL は VLAN インターフェイスに適用する入力ルータ ACL よりも優先されます。ポート ACL が適用されたポート上で受信した着信パケットは、ポート ACL によってフィルタリングされます。その他のポートに着信したルーテッド パケットは、ルータ ACL によってフィルタリングされます。発信するルーテッド パケットには、ルータ ACL のフィルタが適用されます。コンフィギュレーションの矛盾を回避するには、RADIUS サーバーに保存するユーザー プロファイルを慎重に計画しなければなりません。

RADIUS は、ベンダー固有属性などのユーザー単位属性をサポートします。ベンダー固有属性（VSA）は、オクテット ストリング形式で、認証プロセス中にスイッチに渡されます。ユーザー単位 ACL に使用される VSA は、入力方向では inacl#<*n*> で、出力方向では outacl#<*n*> です。MAC ACL は、入力方向に限りサポートされます。VSA は入力方向に限りサポートされます。レイヤ 2 ポートの出力方向ではポート ACL をサポートしません。

拡張 ACL 構文形式だけを使用して、RADIUS サーバに保存するユーザ単位コンフィギュレーションを定義します。RADIUS サーバから定義が渡される場合、拡張命名規則を使用して作成されます。ただし、Filter-Id 属性を使用する場合、標準 ACL を示すことができます。

Filter-Id 属性を使用して、すでにスイッチに設定されているインバウンドまたはアウトバウンド ACL を指定できます。属性には、ACL 番号と、その後ろに入力フィルタリング、出力フィルタリングを示す *.in* または *.out* が含まれています。RADIUS サーバが *.in* または *.out* 構文を許可しない場合、アクセス リストはデフォルトで発信 ACL に適用されます。RADIUS サーバーから送信された Filter-Id がデバイスで設定されていない場合、ユーザーは未承認としてマークされます。スイッチでの Cisco IOS のアクセスリストに関するサポートが制限されているため、Filter-ID 属性は 1 ～ 199（IP 標準 ACL）および 1300 ～ 2699（IP 拡張 ACL）の範囲の IP ACL に対してだけサポートされます。

ユーザ単位 ACL の最大サイズは、4000 ASCII 文字ですが、RADIUS サーバのユーザ単位 ACL の最大サイズにより制限されます。

ユーザ単位の ACL を設定するには、次の前提条件を満たす必要があります。

- AAA 認証をイネーブルにします。
- **network** キーワードを使用して AAA 認証をイネーブルにし、RADIUS サーバからのインターフェイス設定を可能にします。
- 802.1x 認証をイネーブルにします。

- RADIUS サーバにユーザ プロファイルと VSA を設定します。
- 802.1x ポートをシングル ホスト モードに設定します。

ダウンロード可能 ACL およびリダイレクト URL を使用した 802.1x 認証

（注） IPv6 はリダイレクト URL をサポートしていません。

ACL およびリダイレクト URL は、ホストの 802.1x 認証または MAC 認証バイパス中に、RADIUS サーバーからスイッチにダウンロードできます。また、Web 認証中に ACL をダウンロードすることもできます。

（注） ダウンロード可能な ACL は *dACL* とも呼ばれます。

複数のホストが認証され、それらのホストがシングル ホスト モード、MDA モード、またはマルチ認証モードである場合、スイッチは ACL の送信元アドレスをホスト IP アドレスに変更します。

ACL およびリダイレクト URL は、802.1x 対応のポートに接続されるすべてのデバイスに適用できます。

ACL が 802.1x 認証中にダウンロードされない場合、スイッチは、ポートのスタティック デフォルト ACL をホストに適用します。マルチ認証モードまたは MDA モードで設定された音声 VLAN ポートでは、スイッチは ACL を認証ポリシーの一部として電話にだけ適用します。

（注） スタック構成がある dACL の制限は、ポートベースの dACL あたり 64 ACE です。スタック構成なしの制限は、利用可能な TCAM エントリの数になり、これはアクティブな他の ACL 機能によって異なります。

ポート上にスタティック ACL がない場合、ダイナミックな認証デフォルト ACL が作成され、dACL がダウンロードされて適用される前にポリシーが実施されます。

（注） 認証デフォルト ACL は、実行コンフィギュレーションでは表示されません。

認証デフォルト ACL は、ポートで許可ポリシーを持つホストが 1 つ以上検出されると作成されます。認証デフォルト ACL は、最後の認証セッションが終了すると削除されます。IPv4 用の認証デフォルト ACL は、グローバル コンフィギュレーション モードで **ip access-list extended auth-default-acl** コマンドを使用して設定できます。IPv6 の場合、グローバル コンフィギュレーション モードで **ipv6 access-list extended auth-default-acl** コマンドを使用します。

（注） 認証デフォルト ACL は、シングルホストモードの Cisco Discovery Protocol バイパスをサポートしていません。Cisco Discovery Protocol バイパスをサポートするには、インターフェイス上のスタティック ACL を設定する必要があります。

802.1x および MAB 認証方式では、オープンおよびクローズの 2 つの認証方式がサポートされます。クローズ認証モードのポートにスタティック ACL がない場合、次のようになります。

- 認証デフォルト ACL が作成されます。
- 認証デフォルト ACL は、ポリシーが実施されるまで DHCP トラフィックのみを許可します。
- 最初のホスト認証では、許可ポリシーは IP アドレスを挿入せずに適用されます。
- 別のホストが検出されると、最初のホストのポリシーがリフレッシュされ、最初のセッションと後続セッションのポリシーが IP アドレスを挿入して実施されます。

オープン認証モードのポートにスタティック ACL がない場合、次のようになります。

- 認証デフォルト ACL-OPEN が作成され、すべてのトラフィックが許可されます。
- セキュリティ違反を防ぐために、IP アドレスを挿入してポリシーが実施されます。
- Web 認証は、認証デフォルト ACL-OPEN に従います。

許可ポリシーのないホストへのアクセスを制御するために、ディレクティブを設定することができます。サポートされているディレクティブの値は、*open* と *default* です。*open* ディレクティブを設定すると、すべてのトラフィックが許可されます。*default* ディレクティブは、ポートから提供されるアクセスにトラフィックを従わせます。ディレクティブは、AAA サーバー上のユーザープロファイル、またはスイッチ上のいずれかで設定できます。AAA サーバー上でディレクティブを設定するには、**authz-directive =<open/default>** グローバルコマンドを使用します。スイッチ上でディレクティブを設定するには、**epm access-control open** グローバル コンフィギュレーション コマンドを使用します。

（注） ディレクティブのデフォルト値は *default* です。

設定された ACL なしでポート上の Web 認証にホストがフォールバックする場合は、次のようになります。

- ポートがオープン認証モードの場合、認証デフォルト ACL-OPEN が作成されます。
- ポートがクローズ認証モードの場合、認証デフォルト ACL が作成されます。

フォールバック ACL のアクセス コントロール エントリ（ACE）は、ユーザー単位のエントリに変換されます。設定されたフォールバック プロファイルにフォールバック ACL が含まれていない場合、ホストはポートに関連付けられた認証デフォルト ACL に従います。

（注） Web認証でカスタム ロゴを使用し、それを外部サーバーに格納する場合、認証の前にポートの ACL で外部サーバーへのアクセスを許可する必要があります。外部サーバーに適切なアクセスを提供するには、スタティック ポート ACL を設定するか、認証デフォルト ACL を変更する必要があります。

Cisco Secure ACS およびリダイレクト URL の属性と値のペア

スイッチはこれらの *cisco-av-pair* VSA を使用します。

- url-redirect は HTTP URL または HTTPS URL です。
- url-redirect-acl はスイッチ ACL 名または番号です。

スイッチは、CiscoSecure-defined-ACL 属性値ペアを使用して、エンドポイントからの HTTP または HTTPS リクエストを代行受信します。スイッチは、クライアント Web ブラウザを指定されたリダイレクト アドレスに転送します。Cisco Secure ACS 上の url-redirect AV ペアには、Web ブラウザがリダイレクトされる URL が格納されます。url-redirect-acl 属性値ペアには、リダイレクトする HTTP または HTTPS トラフィックを指定する ACL の名前または番号が含まれます。

（注）
- ACL の permit ACE と一致するトラフィックがリダイレクトされます。
- スイッチの URL リダイレクト ACL およびデフォルト ポート ACL を定義します。

リダイレクト URL が認証サーバーのクライアントに設定される場合、接続されるクライアントのスイッチ ポートのデフォルト ポート ACL も設定する必要があります。

このセクションでは、ACS サーバーのスイッチオーバーまたはフェールオーバーの動作について説明します。

最初の認可要求が ACS プライマリ サーバーに送られます。tacacs-server timeout コマンドによって設定されたタイムアウト期間の経過後、要求は認可のためセカンダリ サーバーにスイッチオーバーされます。最初の認可要求の後に続くすべての要求は、セカンダリ ACS サーバーに送られます。スイッチオーバー後にセカンダリ サーバーが利用できない場合は、サーバーへの到達が試みられ、タイムアウト期間が過ぎると、認可要求はプライマリ ACS サーバーに送信されます。どちらのサーバーもダウン状態の場合、認可要求は、設定されたタイムアウト期間経過後に、リスト内の次の ACS サーバーに送信されます。成功しない場合は、その次のサーバーに送信されます。いずれのサーバーにも到達できない場合、ユーザーは認可失敗のメッセージを受け取ります。

Cisco Secure ACS およびダウンロード可能な ACL の属性と値のペア

Cisco Secure ACS で、RADIUS cisco-av-pair ベンダー固有属性（VSA）を使用して、CiscoSecure-Defined-ACL 属性と値（AV）ペアを設定できます。このペアは、

#ACL#-IP-name-number 属性（IPv4 用）および #ACL#-.in.ipv6 属性（IPv6 用）を使って、Cisco Secure ACS でダウンロード可能な ACL の名前を指定します。

- *name* は ACL の名前です。
- *number* はバージョン番号（たとえば 3f783768）です。

ダウンロード可能な ACL が認証サーバーのクライアントに設定される場合、接続されるクライアント スイッチ ポートのデフォルト ポート ACL も設定する必要があります。

デフォルト ACL がスイッチで設定されている場合、Cisco Secure ACS がホスト アクセス ポリシーをスイッチに送信すると、スイッチは、スイッチ ポートに接続されるホストからのトラフィックにこのポリシーを適用します。ポリシーが適用されない場合、デフォルト ACL が適用されます。Cisco Secure ACS がダウンロード可能な ACL をスイッチに送信する場合、この ACL は、スイッチ ポートに設定されているデフォルト ACL より優先されます。ただし、スイッチが Cisco Secure ACS からホスト アクセス ポリシーを受信し、デフォルト ACL が設定されていない場合、許可失敗が宣言されます。

VLAN ID ベース MAC 認証

ダウンロード可能な VLAN ではなくスタティック VLAN ID に基づいてホストを認証する場合、VLAN ID ベース MAC 認証を使用できます。スタティック VLAN ポリシーがスイッチで設定されている場合、認証用の各ホストの MAC アドレスとともに、VLAN 情報が IAS（Microsoft）RADIUS サーバーに送信されます。接続ポートに設定されている VLAN ID は MAC 認証に使用されます。VLAN ID ベース MAC 認証を IAS サーバーで使用することで、ネットワークで一定数の VLAN を使用できます。

機能は、STP によってモニターおよび処理される VLAN の数も制限します。ネットワークは固定 VLAN として管理できます。

ゲスト VLAN を使用した 802.1x 認証

スイッチ上の各 802.1x ポートにゲスト VLAN を設定し、クライアントに対して限定的なサービスを提供できます（802.1x クライアントのダウンロードなど）。これらのクライアントは 802.1x 認証用にシステムをアップグレードできる場合がありますが、一部のホスト（Windows 98 システムなど）は IEEE 802.1x 対応ではありません。

スイッチが EAP Request/Identity フレームに対する応答を受信していない場合、または EAPOL パケットがクライアントによって送信されない場合に、802.1x ポート上でゲスト VLAN をイネーブルにすると、スイッチはクライアントにゲスト VLAN を割り当てます。

スイッチは EAPOL パケット履歴を保持します。EAPOL パケットがリンクの存続時間中にインターフェイスで検出された場合、スイッチはそのインターフェイスに接続されているデバイスが IEEE 802.1x 対応のものであると判断します。インターフェイスはゲスト VLAN ステートにはなりません。インターフェイスのリンク ステータスがダウンした場合、EAPOL 履歴はクリアされます。EAPOL パケットがインターフェイスで検出されない場合、そのインターフェイスはゲスト VLAN のステートになります。

スイッチが 802.1x 対応の音声デバイスを許可しようとしたが、AAA サーバーが使用できない場合、許可は失敗します。ただし、EAPOLパケットの検出はEAPOL履歴に保存されます。この音声デバイスは、AAA サーバーが使用可能になると許可されます。ただし、他のデバイスによるゲスト VLAN へのアクセスは許可されなくなります。この状況を防ぐには、次のいずれかのコマンド シーケンスを使用します。

- **authentication event no-response action authorize vlan** *vlan-id* インターフェイス コンフィギュレーション コマンドを入力し、ゲスト VLAN へのアクセスを許可します。
- **shutdown** インターフェイス コンフィギュレーション コマンドを入力し、さらに **no shutdown** インターフェイス コンフィギュレーション コマンドを入力してポートを再起動します。

リンクの存続時間中にデバイスが EAPOL パケットを送信した場合、スイッチはゲスト VLAN への認証アクセスに失敗したクライアントを許可しません。

（注） インターフェイスがゲスト VLAN に変わってから EAPOL パケットが検出された場合、無許可ステートに戻って 802.1x 認証を再起動します。

スイッチ ポートがゲスト VLAN に変わると、802.1x 非対応クライアントはすべてアクセスを許可されます。ゲスト VLAN が設定されているポートに 802.1x 対応クライアントが加入すると、ポートは、ユーザ設定によるアクセス VLAN で無許可ステートになり、認証が再起動されます。

ゲスト VLAN は、単一のホスト、複数のホスト、複数認証、またはマルチドメイン モードにおける 802.1x ポートでサポートされています。

RSPAN VLAN、プライベート VLAN、音声 VLAN を除いて、アクティブ VLAN を 802.1x ゲスト VLAN として設定できます。ゲスト VLAN 機能は、内部 VLAN（ルーテッド ポート）またはトランク ポートではサポートされていません。アクセス ポート上でだけサポートされます。

スイッチは *MAC* 認証バイパス をサポートします。MAC 認証バイパスが 802.1x ポートでイネーブルの場合、スイッチは、IEEE 802.1x 認証のタイムアウト時に EAPOL メッセージ交換を待機している間、クライアント MAC アドレスに基づいてクライアントを許可できます。スイッチは、802.1x ポート上のクライアントを検出したあとで、クライアントからのイーサネット パケットを待機します。スイッチは、MAC アドレスに基づいたユーザー名およびパスワードを持つ RADIUS-access/request フレームを認証サーバーに送信します。認証に成功すると、スイッチはクライアントにネットワークへのアクセスを許可します。認証に失敗すると、スイッチはポートにゲスト VLAN を割り当てます（指定されていない場合）。

制限付き VLAN を使用した 802.1x 認証

ゲスト VLAN にアクセスできないクライアント向けに、限定されたサービスを提供するために、スイッチスタックまたはスイッチの各 IEEE 802.1x ポートに対して制限付き VLAN（認証失敗 *VLAN* と呼ばれることもあります）を設定できます。これらのクライアントは、認証プロセスに失敗したため他の VLAN にアクセスできない 802.1x 対応クライアントです。制限付き VLAN を使用すると、認証サーバの有効なクレデンシャルを持っていないユーザ（通常、企業

にアクセスするユーザ）に、サービスを制限したアクセスを提供できます。管理者は制限付き VLAN のサービスを制御できます。

（注） 両方のタイプのユーザに同じサービスを提供する場合、ゲスト VLAN と制限付き VLAN の両方を同じに設定できます。

この機能がないと、クライアントは認証失敗を永遠に繰り返すことになるため、スイッチ ポートがスパニングツリーのブロッキング ステートから変わることができなくなります。制限付き VLAN の機能を使用することで、クライアントの認証試行回数を指定し（デフォルト値は 3 回）、一定回数後にスイッチ ポートを制限付き VLAN の状態に移行させることができます。

認証サーバはクライアントの認証試行回数をカウントします。このカウントが設定した認証試行回数を超えると、ポートが制限付き VLAN の状態に変わります。失敗した試行回数は、RADIUS サーバが *EAP failure* で応答したときや、EAP パケットなしの空の応答を返したときからカウントされます。ポートが制限付き VLAN に変わったら、このカウント数はリセットされます。

認証に失敗したユーザの VLAN は、もう一度認証を実行するまで制限された状態が続きます。VLAN 内のポートは設定された間隔に従って再認証を試みます（デフォルトは 60 秒）。再認証に失敗している間は、ポートの VLAN は制限された状態が続きます。再認証に成功した場合、ポートは設定された VLAN もしくは RADIUS サーバによって送信された VLAN に移行します。再認証はディセーブルにすることもできますが、ディセーブルにすると、*link down* または *EAP logoff* イベントを受信しない限り、ポートの認証プロセスを再起動できません。クライアントがハブを介して接続している場合、再認証機能はイネーブルにしておくことを推奨します。クライアントの接続をハブから切り離すと、ポートに *link down* や *EAP logoff* イベントが送信されない場合があります。

ポートが制限付き VLAN に移行すると、EAP 成功の疑似メッセージがクライアントに送信されます。このメッセージによって、繰り返し実行している再認証を停止させることができます。クライアントによっては（Windows XP が稼働しているデバイスなど）、EAP なしで DHCP を実装できません。

制限付き VLAN は、すべてのホスト モードでの 802.1x ポート上、およびレイヤ 2 ポート上でサポートされます。

RSPAN VLAN、プライマリ プライベート VLAN、音声 VLAN を除いて、アクティブ VLAN を 802.1x 制限付き VLAN として設定できます。制限付き VLAN 機能は、内部 VLAN（ルーテッド ポート）またはトランク ポートではサポートされていません。アクセス ポート上でだけサポートされます。

ダイナミック ARP インスペクション、DHCP スヌーピング、IP 送信元ガードなどの他のセキュリティ ポート機能は、制限付き VLAN に対して個別に設定できます。

アクセス不能認証バイパスを使用した 802.1x 認証

スイッチが設定された RADIUS サーバーに到達できず、新しいホストを認証できない場合、アクセス不能認証バイパス機能を使用します。この機能は、クリティカル認証または *AAA* 失敗

ポリシーとも呼ばれます。これらのホストをクリティカルポートに接続するようにスイッチを設定できます。

新しいホストがクリティカルポートに接続しようとすると、そのホストはユーザ指定のアクセス VLAN、クリティカル *VLAN* に移動されます。管理者はこれらのホストに制限付き認証を付与します。

スイッチは、クリティカルポートに接続されているホストを認証しようとする場合、設定されている RADIUS サーバーのステータスをチェックします。利用可能なサーバーが 1 つあれば、スイッチはホストを認証できます。ただし、すべての RADIUS サーバーが利用不可能な場合は、スイッチはホストへのネットワーク アクセスを許可して、ポートを認証ステートの特別なケースであるクリティカル認証ステートにします。

（注） クリティカル認証をインターフェイスで設定する場合は、クリティカル承認（クリティカル *vlan*）に使用する vlan をスイッチでアクティブにする必要があります。クリティカル *vlan* が非アクティブまたはダウンしていると、クリティカル認証セッションは非アクティブな vlan の有効化を試行し続け、繰り返し失敗します。これは大量のメモリ保持の原因となる可能性があります。

複数認証ポートのアクセス不能認証バイパスのサポート

ポートが任意のホスト モードで設定されていて、AAA サーバーを使用できない場合、ポートはマルチホスト モードに設定され、クリティカル VLAN に移動されます。マルチ認証（multiauth）ポートで、このアクセス不能バイパスをサポートするには、**authentication event server dead action reinitialize vlan** *vlan-id* コマンドを使用します。新しいホストがクリティカルポートに接続しようとすると、そのポートは再初期化され、接続されているすべてのホストがユーザ指定のアクセス VLAN に移動されます。

このコマンドは、すべてのホスト モードでサポートされます。

アクセス不能認証バイパスの認証結果

アクセス不能認証バイパス機能の動作は、ポートの許可ステートにより異なります。

- クリティカルポートに接続されているホストが認証しようとする際にポートが無許可ですべてのサーバーが利用できない場合、スイッチは RADIUS 設定済み VLAN またはユーザー指定のアクセス VLAN にあるポートをクリティカル認証ステートにします。
- ポートが許可済みで、再認証が行われた場合、スイッチは現在の VLAN（事前に RADIUS サーバーにより割り当てられた）でクリティカルポートをクリティカル認証ステートにします。
- 認証交換中に RADIUS サーバーが利用不可能となった場合、現在の交換はタイム アウトとなり、スイッチは次の認証試行の間にクリティカルポートをクリティカル認証ステートとします。

RADIUSサーバが再び使用可能になったときにホストを再初期化し、クリティカルVLANから移動するように、クリティカルポートを設定できます。このように設定した場合、クリティカル認証ステートのすべてのクリティカル ポートは自動的に再認証されます。

アクセス不能認証バイパス機能の相互作用

アクセス不能認証バイパスは、次の機能と相互に作用します。

- ゲスト VLAN：アクセス不能認証バイパスは、ゲスト VLAN と互換性があります。ゲストVLANが8021.xポートでイネーブルの場合、この機能は次のように相互に作用します。
  - スイッチが EAP Request/Identity フレームへの応答を受信しないとき、または EAPOL パケットがクライアントによって送信されないときに、少なくとも 1 つの RADIUS サーバーが使用できれば、スイッチはクライアントにゲストVLANを割り当てます。
  - すべての RADIUS サーバーが使用できず、クライアントがクリティカル ポートに接続されている場合、スイッチはクライアントを認証して、クリティカル ポートを RADIUS 認証済み VLAN またはユーザー指定のアクセス VLAN でクリティカル認証ステートにします。
  - すべての RADIUS サーバーが使用できず、クライアントがクリティカル ポートに接続されていない場合、ゲストVLANが設定されていても、スイッチはクライアントにゲスト VLAN を割り当てられません。
  - すべての RADIUS サーバーが使用できず、クライアントがクリティカル ポートに接続されていて、すでにゲストVLANが割り当てられている場合、スイッチはそのポートをゲスト VLAN に保持します。
- 制限付き VLAN：ポートがすでに制限付き VLAN で許可されていて RADIUS サーバーが使用できない場合、スイッチはクリティカル ポートを制限付き VLAN でクリティカル認証ステートにします。
- 802.1xアカウンティング：RADIUSサーバーが使用できない場合、アカウンティングは影響を受けません。
- プライベート VLAN：プライベート VLAN ホスト ポートにアクセス不能認証バイパスを設定できます。アクセス VLAN は、セカンダリ VLAN でなければなりません。
- 音声VLAN：アクセス不能認証バイパスは音声 VLAN と互換性がありますが、RADIUS 設定済み VLAN またはユーザー指定のアクセス VLAN は、音声 VLAN と異なっていなければなりません。
- Remote Switched Port Analyzer（RSPAN）：アクセス不能認証バイパスの RADIUS 設定またはユーザー指定のアクセス VLAN として RSPAN VLAN を指定しないでください。

802.1x クリティカル音声 VLAN

ポートに接続されている IP フォンが Cisco Identity Services Engine（ISE）によって認証される際、そのIPフォンは音声ドメインに参加します。ISEが到達不能である場合、スイッチはデバ

イスが音声デバイスなのかどうかを判断できません。サーバーが使用できない場合、電話機は音声ネットワークにアクセスできないため、動作できません。

データ トラフィックの場合、アクセス不能認証バイパス（クリティカル認証）を設定し、サーバーが使用できない場合にトラフィックがネイティブ VLAN を通過できるようにすることができます。RADIUS 認証サーバーが使用できず（ダウンしていて）、アクセスできない認証バイパスがイネーブルの場合、スイッチは、クライアントにネットワークのアクセスを許可し、RADIUS 設定 VLAN またはユーザー指定アクセス VLAN でポートをクリティカル認証ステートにします。設定された RADIUS サーバーにスイッチが到達できず、新しいホストを認証できない場合、スイッチはこれらのホストをクリティカルポートに接続します。クリティカルポートに接続を試行している新しいホストは、ユーザー指定のアクセス VLAN（クリティカル VLAN）に移動され、制限付き認証を許可されます。

（注） クリティカル音声 VLAN のダイナミック割り当ては、ネストされたサービステンプレートではサポートされません。そのため、デバイスはループ内で VLAN を連続的に切り替えます。

authentication event server dead action authorize voice インターフェイス コンフィギュレーション コマンドを使用して、クリティカル音声 VLAN 機能を設定できます。ISE が応答しない場合、ポートはクリティカル認証モードになります。ホストからのトラフィックが音声 VLAN でタグ付けされると、接続デバイス（電話機）は、ポートに対して設定された音声 VLAN に配置されます。IP フォンは Cisco Discovery Protocol（シスコデバイス）や LLDP または DHCP を介して音声 VLAN ID を学習します。

switchport voice vlan *vlan-id* インターフェイス コンフィギュレーション コマンドを入力して、ポートの音声 VLAN を設定できます。

この機能は、マルチドメイン モードおよびマルチ認証ホスト モードでサポートされます。スイッチがシングルホスト モードまたはマルチホスト モードの場合にコマンドを入力できますが、デバイスがマルチドメインまたはマルチ認証ホストモードに変わらない限りコマンドは有効になりません。

802.1x ユーザ ディストリビューション

802.1x ユーザ ディストリビューションを設定すると、複数の異なる VLAN で同じグループ名のユーザのロード バランシングを行うことができます。

VLAN は、RADIUS サーバーにより提供されるか、VLAN グループ名でスイッチ CLI を介して設定します。

- RADIUS サーバを設定して、ユーザの複数の VLAN 名を送信します。複数の VLAN 名は、ユーザへの応答の一部として送信できます。802.1x ユーザ ディストリビューションは、特定の VLAN のすべてのユーザを追跡し、許可されたユーザをユーザ数が最も少ない VLAN に移動することでロード バランシングを行います。
- RADIUS サーバを設定してユーザの VLAN グループ名を送信します。VLAN グループ名は、ユーザへの応答の一部として送信できます。スイッチ CLI を使用して設定した VLAN

グループ名で、選択された VLAN グループ名を検索できます。VLAN グループ名が検出されると、この VLAN グループ名で対応する VLAN を検索して、ユーザ数が最も少ない VLAN が検出されます。ロード バランシングは、対応する許可済みユーザをその VLAN に移動することで行われます。

（注） RADIUS サーバーは、VLAN-ID、VLAN 名または VLAN グループを任意に組み合わせて VLAN 情報を送信できます。

802.1x ユーザ ディストリビューションの設定時の注意事項

- 少なくとも 1 つの VLAN が VLAN グループにマッピングされることを確認してください。
- 複数の VLAN を VLAN グループにマッピングできます。
- VLAN を追加または削除することで、VLAN グループを変更できます。
- 既存の VLAN を VLAN グループ名からクリアする場合、VLAN の認証済みポートはクリアされませんが、既存の VLAN グループからマッピングが削除されます。
- 最後の VLAN を VLAN グループ名からクリアすると、VLAN グループがクリアされます。
- アクティブ VLAN がグループにマッピングされても VLAN グループをクリアできます。VLAN グループをクリアすると、グループ内で任意の VLAN の認証ステートであるポートまたはユーザはクリアされませんが、VLAN の VLAN グループへのマッピングはクリアされます。

音声 VLAN ポートを使用した IEEE 802.1x 認証

音声 VLAN ポートは特殊なアクセス ポートで、次の 2 つの VLAN ID が対応付けられています。

- IP Phone との間で音声トラフィックを伝送する VVID。VVID は、ポートに接続された IP Phone を設定するために使用されます。
- IP Phone を通じて、スイッチと接続しているワークステーションとの間でデータ トラフィックを伝送する PVID。PVID は、ポートのネイティブ VLAN です。

ポートの許可ステートにかかわらず、IP Phone は音声トラフィックに対して VVID を使用します。これにより、IP Phone は IEEE 802.1x 認証とは独立して動作できます。

シングル ホスト モードでは、IP Phone だけが音声 VLAN で許可されます。マルチ ホスト モードでは、サプリカントが PVID で認証された後、追加のクライアントがトラフィックを音声 VLAN 上で送信できます。マルチ ホスト モードがイネーブルの場合、サプリカント認証は PVID と VVID の両方に影響します。

リンクがあるとき、音声 VLAN ポートはアクティブになり、IP Phone からの最初の CDP メッセージを受け取るとデバイスの MAC アドレスが表示されます。Cisco IP Phone は、他のデバイ

スから受け取った CDP メッセージをリレーしません。その結果、複数の IP Phone が直列に接続されている場合、スイッチは直接接続されている 1 台の IP Phone のみを認識します。音声 VLAN ポートで IEEE 802.1x 認証がイネーブルの場合、スイッチは 2 ホップ以上離れた認識されない IP Phone からのパケットをドロップします。

IEEE 802.1x 認証をスイッチ ポート上でイネーブルにすると、音声 VLAN でもあるアクセス ポート VLAN を設定できます。

IP 電話がシングル ホスト モードで 802.1x 対応のスイッチ ポートに接続されている場合、スイッチは認証を行わずに電話ネットワーク アクセスを承認します。ポートで Multidomain Authentication（MDA）を使用して、データ デバイスと IP フォンなどの音声デバイスの両方を認証することを推奨します。

（注） 音声 VLAN が設定され、Cisco IP Phone が接続されているアクセス ポートで IEEE 802.1x 認証をイネーブルにした場合、Cisco IP Phone のスイッチへの接続が最大 30 秒間失われます。

ポート セキュリティを使用した IEEE 802.1x 認証

通常、IEEE 802.1x がイネーブルの場合に、ポート セキュリティをイネーブルにすることは推奨されません。IEEE 802.1x ではポート単位（IP テレフォニーに MDA が設定されている場合は VLAN 単位）で単一の MAC アドレスが適用されるため、ポート セキュリティは冗長であり、場合によっては期待される IEEE 802.1x の動作と干渉することがあります。

WoL 機能を使用した IEEE 802.1x 認証

IEEE 802.1x 認証の Wake-on-LAN（WoL）機能を使用すると、スイッチにマジック パケットと呼ばれる特定のイーサネット フレームを受信させて、休止状態の PC を起動させることができます。この機能は、管理者が休止状態のシステムへ接続しなければならない場合に役立ちます。

WoL を使用するホストが IEEE 802.1x ポートを通じて接続され、ホストの電源がオフになると、IEEE 802.1x ポートは無許可になります。無許可になったポートは EAPOL パケットしか送受信できないため、WoL マジック パケットはホストに届きません。さらに PC が休止状態になると、PC が認証されなくなるため、スイッチ ポートは閉じたままになります。

スイッチが WoL 機能を有効にした IEEE 802.1x 認証を使用している場合、スイッチはマジック パケットを含むトラフィックを無許可の IEEE 802.1x ポートに転送します。ポートが無許可の間、スイッチは EAPOL パケット以外の入力トラフィックをブロックし続けます。ホストはパケットを受信できますが、パケットをネットワーク内にある他のデバイスに送信できません。

（注） PortFast がポートでイネーブルになっていないと、そのポートは強制的に双方向ステートになります。

authentication control-direction in インターフェイス コンフィギュレーション コマンドを使用してポートを単一方向として設定すると、ポートはスパニングツリーフォワーディング ステートに変更されます。ポートは、ホストにパケットを送信できますが、受信はできません。

authentication control-direction both インターフェイス コンフィギュレーション コマンドを使用してポートを双方向として設定すると、ポートは、両方向でアクセスコントロールされます。ポートは、ホストとの間でパケットを送受信しません。

MAC 認証バイパスを使用した IEEE 802.1x 認証

MAC 認証バイパス機能を使用し、クライアント MAC アドレスに基づいてクライアントを許可するようにスイッチを設定できます。たとえば、プリンタなどのデバイスに接続された IEEE 802.1x ポートでこの機能をイネーブルにできます。

クライアントからの EAPOL 応答の待機中に IEEE 802.1x 認証がタイムアウトした場合、スイッチは MAC 認証バイパスを使用してクライアントを許可しようとします。

MAC 認証バイパス機能が IEEE 802.1x ポートでイネーブルの場合、スイッチはクライアント ID として MAC アドレスを使用します。認証サーバーには、ネットワーク アクセスを許可されたクライアント MAC アドレスのデータベースがあります。スイッチは、IEEE 802.1x ポート上のクライアントを検出した後で、クライアントからのイーサネット パケットを待機します。スイッチは、MAC アドレスに基づいたユーザー名およびパスワードを持つ RADIUS-access/request フレームを認証サーバーに送信します。認証に成功すると、スイッチはクライアントにネットワークへのアクセスを許可します。許可が失敗した場合、ゲスト VLAN が設定されていれば、スイッチはポートをゲスト VLAN に割り当てます。このプロセスは、ほとんどのクライアントデバイスで動作します。ただし、代替の MAC アドレス形式を使用しているクライアントでは動作しません。標準の形式とは異なる MAC アドレスを持つクライアントに対して MAB 認証をどのように実行するかや、RADIUS の設定のどこでユーザ名とパスワードが異なることが要求されるかを設定できます。

リンクのライフタイム中に EAPOL パケットがインターフェイス上で検出された場合、スイッチは、そのインターフェイスに接続されているデバイスが 802.1x 対応サプリカントであることを確認し、（MAC 認証バイパス機能ではなく）802.1x 認証を使用してインターフェイスを認証します。インターフェイスのリンク ステータスがダウンした場合、EAPOL 履歴はクリアされます。

スイッチがすでに MAC 認証バイパスを使用してポートを許可し、IEEE 802.1x サプリカントを検出している場合、スイッチはポートに接続されているクライアントを許可します。再認証が発生するときに、Termination-Action RADIUS 属性値が DEFAULT であるために前のセッションが終了した場合、スイッチはポートに設定されている認証または再認証手法を使用します。

MAC 認証バイパスで認証されたクライアントは再認証できます。再認証プロセスは、IEEE 802.1x を使用して認証されたクライアントに対するプロセスと同じです。再認証中は、ポートは前に割り当てられた VLAN のままです。再認証に成功すると、スイッチはポートを同じ VLAN に保持します。再認証に失敗した場合、ゲスト VLAN が設定されていれば、スイッチはポートをゲスト VLAN に割り当てます。

再認証が Session-Timeout RADIUS 属性（Attribute[27]）、および Termination-Action RADIUS 属性（Attribute[29]）に基づいて行われるときに、Termination-Action RADIUS 属性（Attribute[29]）

のアクションが *Initialize*（属性値は *DEFAULT*）である場合、MAC 認証バイパス セッションは終了し、再認証の間の接続は失われます。MAC 認証バイパス機能が IEEE 802.1x 認証がタイムアウトした場合、スイッチは MAC 認証バイパス機能を使用して再認証を開始します。これらの AV ペアの詳細については、RFC 3580『IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines』を参照してください。

MAC 認証バイパスは、次の機能と相互に作用します。

- IEEE 802.1x 認証：802.1x 認証がポートでイネーブルの場合にのみ MAC 認証バイパスをイネーブルにできます。
- ゲスト VLAN：クライアントの MAC アドレス ID が無効な場合、ゲスト VLAN が設定されていれば、スイッチは VLAN にクライアントを割り当てます。
- 制限付き VLAN：IEEE 802.1x ポートに接続されているクライアントが MAC 認証バイパスで認証されている場合には、この機能はサポートされません。
- ポート セキュリティ
- 音声 VLAN
- プライベート VLAN：クライアントをプライベート VLAN に割り当てられます。
- Network Edge Access Topology（NEAT）：MAB と NEAT は相互に排他的です。インターフェイス上で NEAT が有効の場合は、MAB を有効にすることはできません。また、インターフェイス上で MAB が有効の場合は、NEAT を有効にすることはできません。

Cisco IOS Release 12.2(55)SE 以降では、冗長 MAB システム メッセージのフィルタリングをサポートします。

Network Admission Control レイヤ 2 IEEE 802.1x 検証

スイッチは、デバイスのネットワーク アクセスを許可する前にエンドポイント システムやクライアントのウイルス対策の状態またはポスチャを調べる Network Admission Control（NAC）レイヤ 2 IEEE 802.1x 検証をサポートしています。NAC レイヤ 2 IEEE 802.1x 検証を使用すると、以下の作業を実行できます。

- Session-Timeout RADIUS 属性（属性 [27]）と Termination-Action RADIUS 属性（属性 [29]）を認証サーバーからダウンロードします。
- Session-Timeout RADIUS 属性（属性 [27]）の値として再認証試行間の秒数を指定し、RADIUS サーバーからクライアントのアクセス ポリシーを取得します。
- スイッチが Termination-Action RADIUS 属性（属性[29]）を使用してクライアントを再認証する際のアクションを設定します。値が *DEFAULT* であるか、値が設定されていない場合、セッションは終了します。値が RADIUS 要求の場合、再認証プロセスが開始します。
- VLAN の番号や名前、または VLAN グループ名のリストを Tunnel Group Private ID（属性 [81]）の値として設定し、VLAN の番号や名前、または VLAN グループ名のプリファレンスを Tunnel Preference（属性 [83]）の値として設定します。Tunnel Preference を設定しない場合、最初の Tunnel Group Private ID（属性 [81]）属性がリストから選択されます。

- NAC ポスチャトークンを表示します。これは、**show authentication** 特権 EXEC コマンドを使用して、クライアントのポスチャを示します。
- ゲスト VLAN としてセカンダリ プライベート VLAN を設定します。

NAC レイヤ 2 IEEE 802.1x 検証の設定は、RADIUS サーバーにポスチャ トークンを設定する必要があることを除いて、IEEE 802.1x ポートベース認証と似ています。

柔軟な認証の順序設定

柔軟な認証の順序設定を使用して、ポートが新しいホストを認証するときに使用する方法の順序を設定できます。The IEEE 802.1X の柔軟な認証機能では、以下の 3 つの認証方法をサポートしています。

- dot1X：IEEE 802.1X 認証はレイヤ 2 の認証方式です。
- mab：MAC 認証バイパスはレイヤ 2 の認証方式です。
- webauth：Web 認証はレイヤ 3 の認証方式です。

この機能を使用すると、各ポートでどの認証方式を使用するかを制御できます。また、そのポートの方式についてフェールオーバー順も制御できます。たとえば、MAC 認証バイパスおよび 802.1x は、プライマリまたはセカンダリ認証方法として使用し、Web 認証は、これらの認証のいずれか、または両方が失敗した場合のフォールバック方法として使用できます。

The IEEE 802.1X の柔軟な認証機能では、以下のホスト モードをサポートしています。

- multi-auth：マルチ認証では、音声 VLAN に 1 つの認証、データ VLAN に複数の認証を使用できます。
- multi-domain：マルチドメイン認証では、音声 VLAN に 1 つ、データ VLAN に 1 つ、計 2 つの認証を使用できます。

Open1x 認証

Open1x 認証によって、デバイスが認証される前に、そのデバイスがポートにアクセスできるようになります。オープン認証が設定されている場合、新しいホストはポートに定義されているアクセス コントロール リスト（ACL）に基づいてトラフィックを渡します。ホストが認証されると、RADIUS サーバに設定されているポリシーがそのホストに適用されます。

オープン認証を次の状況で設定できます。

- シングル ホスト モードでのオープン認証：1 人のユーザーだけが認証の前後にネットワークにアクセスできます。
- MDA モードでのオープン認証：音声ドメインの 1 人のユーザーだけ、およびデータ ドメインの 1 人のユーザーだけが許可されます。
- マルチ ホスト モードでのオープン認証：任意のホストがネットワークにアクセスできます。

- 複数認証モードでのオープン認証：MDA の場合と似ていますが、複数のホストを認証できます。

（注） オープン認証が設定されている場合は、他の認証制御よりも優先されます。これは、**authentication open** インターフェイス コンフィギュレーション コマンドを使用した場合、**authentication port-control** インターフェイス コンフィギュレーション コマンドに関係なく、ポートがホストにアクセス権を付与することを意味します。

マルチドメイン認証

スイッチはマルチドメイン認証（MDA）をサポートしています。これにより、データ装置と IP Phone などの音声装置（シスコ製品またはシスコ以外の製品）の両方を同じスイッチ ポート上で認証できます。ポートはデータ ドメインと音声ドメインに分割されます。

（注） すべてのホスト モードで、ポートベース認証が設定されている場合、ライン プロトコルは許可の前にアップのままです。

MDA では、デバイス認証の順序が指定されません。ただし、最適な結果を得るには、MDA 対応のポート上のデータ デバイスよりも前に音声デバイスを認証することを推奨します。

MDA を設定するときには、次の注意事項に従ってください。

- MDA のスイッチ ポートを設定する必要があります。
- ホスト モードがマルチドメインに設定されている場合、IP Phone の音声 VLAN を設定する必要があります。
- MDA 対応ポートでの音声 VLAN 割り当ては、Cisco IOS Release 12.2(40)SE 以降でサポートされています。
- 音声デバイスを認可するには、値を *device-traffic-class=voice* に設定した Cisco 属性値（AV）ペア属性を送信するように AAA サーバーを設定する必要があります。この値を使用しない場合、音声デバイスはデータ デバイスとして扱われます。
- ゲスト VLAN および制限付き VLAN 機能は、MDA 対応のポートのデータ デバイスだけに適用されます。許可に失敗した音声デバイスは、データ デバイスとして扱われます。
- 複数のデバイスでポートの音声またはデータ ドメインの許可を行おうとすると、errordisable になります。
- デバイスが許可されるまで、ポートはそのトラフィックをドロップします。他社製 IP Phone または音声デバイスはデータおよび音声 VLAN の両方に許可されます。データ VLAN では、音声デバイスを DHCP サーバーに接続して IP アドレスおよび音声 VLAN 情報を取得

することができます。音声デバイスが 音声 VLAN で送信を開始すると、データ VLAN へのアクセスはブロックされます。

- データ VLAN とバインドしている音声デバイス MAC アドレスは、ポート セキュリティ MAC アドレス制限にカウントされません。
- MDA では、IEEE 802.1x 認証をサポートしていないデバイスへのスイッチ ポートの接続を許可するフォールバック メカニズムとして、MAC 認証バイパスを使用できます。
- データまたは音声デバイスがポートで検出されると、認証に成功するまでその MAC アドレスがブロックされます。許可に失敗した場合、MAC アドレスが 5 分間ブロックされたままになります。
- ポートが未認証中に 6 つ以上のデバイスがデータ VLAN で検出された場合や、複数の音声デバイスが音声 VLAN で検出された場合、ポートは errdisable になります。
- ポートのホスト モードをシングルホスト モードまたはマルチホスト モードからマルチドメイン モードに変更すると、ポートでは許可されたデータ デバイスは許可されたままになります。ただし、ポートの音声 VLAN で許可されている Cisco IP Phone は自動的に削除されるので、そのポートでは再認証を行う必要があります。
- ゲスト VLAN や制限付き VLAN などのアクティブ フォールバック メカニズムは、ポートをシングル モードまたはマルチホスト モードからマルチドメイン モードに変更したあとでも設定されたままになります。
- ポートのホスト モードをマルチドメイン モードからシングル モードまたはマルチホスト モードに変更すると、許可されているすべてのデバイスがポートから削除されます。
- まずデータ ドメインを許可してゲスト VLAN に参加させる場合、IEEE 802.1x 非対応の音声デバイスは、音声 VLAN のパケットをタグ付けして、認証を開始する必要があります。
- MDA 対応ポートでは、ユーザー単位 ACL を推奨しません。ユーザー単位 ACL ポリシーを備えた、許可されたデバイスは、ポートの音声 VLAN とデータ VLAN の両方のトラフィックに影響を与えることがあります。このようなデバイスを使用する場合は、ポートでユーザー単位 ACL を適用するデバイスは 1 台だけにしてください。

ユーザのログイン制限

ログイン制限機能では、ネットワーク管理者が、ユーザーによるネットワークへのログイン試行を制限することができます。ユーザーによるネットワークへのログインの試行が、設定可能な時間制限内かつ設定可能な回数以内に成功しなかった場合、ユーザーをブロックできます。この機能は、ローカル ユーザに対してだけ有効であり、リモート ユーザは利用できません。この機能を有効にするには、グローバル コンフィギュレーション モードで **aaa authentication rejected** コマンドを設定する必要があります。

Network Edge Access Topology（NEAT）を使用した 802.1x サプリカントおよびオーセンティケータ

Network Edge Access Topology（NEAT）機能は、ワイヤリング クローゼット（会議室など）外の領域まで識別を拡張します。これにより、任意のタイプのデバイスをポートで認証できます。

- 802.1x スイッチ サプリカント：802.1x サプリカント機能を使用することで、別のスイッチのサプリカントとして機能するようにスイッチを設定できます。この設定は、たとえば、スイッチがワイヤリング クローゼット外にあり、トランク ポートを介してアップストリーム スイッチに接続される場合に役に立ちます。802.1x スイッチ サプリカント機能を使用して設定されたスイッチは、セキュアな接続のためにアップストリーム スイッチで認証します。サプリカント スイッチが認証に成功すると、オーセンティケータ スイッチでポート モードがアクセスからトランクに変更されます。サプリカント スイッチでは、CISP を有効にするときに手動でトランクを設定する必要があります。
- アクセス VLAN は、オーセンティケータ スイッチで設定されている場合、認証が成功した後にトランク ポートのネイティブ VLAN になります。

デフォルトでは、BPDU ガードが有効にされたオーセンティケータ スイッチにサプリカントのスイッチを接続する場合、オーセンティケータのポートはサプリカント スイッチが認証する前にスパニングツリー プロトコル（STP）のブリッジ プロトコル データ ユニット（BPDU）を受信した場合、errdisable 状態になる可能性があります。Cisco IOS Release 15.0(1) SE 以降では、認証中にサプリカントのポートから送信されるトラフィックを制御できます。**dot1x supplicant controlled transient** グローバル コンフィギュレーション コマンドを入力すると、認証が完了する前にオーセンティケータ ポートがシャットダウンすることがないように、認証中に一時的にサプリカントのポートをブロックします。認証に失敗すると、サプリカントのポートが開きます。**no dot1x supplicant controlled transient** グローバル コンフィギュレーション コマンドを入力すると、認証期間中にサプリカント ポートが開きます。これはデフォルトの動作です。

BPDU ガードが **spanning-tree bpduguard enable** インターフェイス コンフィギュレーション コマンドによりオーセンティケータのスイッチ ポートで有効になっている場合、サプリカント スイッチで **dot1x supplicant controlled transient** コマンドを使用することを強く推奨します。

（注） **spanning-tree portfast bpduguard default** グローバル コンフィギュレーション コマンドを使用して、グローバルにオーセンティケータ スイッチで BPDU ガードを有効にした場合、**dot1x supplicant controlled transient** コマンドを入力すると、BPDU の違反が避けられなくなります。

1 つ以上のサプリカント スイッチに接続するオーセンティケータ スイッチ インターフェイスで MDA または multiauth モードをイネーブルにできます。マルチホスト モードはオーセンティケータ スイッチ インターフェイスではサポートされていません。

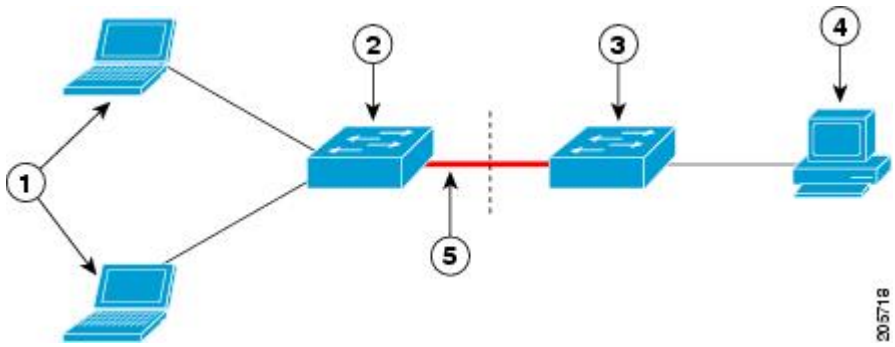
インターフェイスで有効になっているシングルホスト モードでオーセンティケータ スイッチをリブートすると、インターフェイスが認証前に err-disabled 状態に移行する場合があります。

err-disabled 状態から回復するには、オーセンティケータ ポートをフラップしてインターフェイスを再度アクティブにし、認証を開始します。

すべてのホストモードで機能するように **dot1x supplicant force-multicast** グローバル コンフィギュレーション コマンドを Network Edge Access Topology（NEAT）のサプリカントスイッチで使用します。

- ホスト許可：許可済み（サプリカントでスイッチに接続する）ホストからのトラフィックだけがネットワークで許可されます。これらのスイッチは、Client Information Signalling Protocol（CISP）を使用して、サプリカント スイッチに接続する MAC アドレスをオーセンティケータ スイッチに送信します。
- 自動有効化：オーセンティケータ スイッチでのトランク コンフィギュレーションを自動的に有効化します。これにより、サプリカント スイッチから着信する複数の VLAN のユーザー トラフィックが許可されます。ISE で cisco-av-pair を *device-traffic-class=switch* として設定します（この設定は *group* または *user* 設定で行うことができます）。

図 *112 : CISP* を使用したオーセンティケータまたはサプリカント スイッチ



1	ワークステーション（クライアント）	2	サプリカント スイッチ（ワイヤリング クローゼット外）
3	オーセンティケータ スイッチ	4	Cisco ISE
5	トランク ポート		

（注） **switchport nonegotiate** コマンドは、NEAT を使用したサプリカントおよびオーセンティケータ スイッチではサポートされません。このコマンドは、トポロジのサプリカント側で設定しないでください。オーセンティケータ サーバ側で設定した場合は、内部マクロによってポートからこのコマンドが自動的に削除されます。

音声認識 802.1x セキュリティ

（注） 音声認識 IEEE 802.1x 認証を使用するには、スイッチが LAN Base イメージを実行している必要があります。

音声認識 802.1x セキュリティ機能を使用して、セキュリティ違反が発生した場合にデータまたは音声 VLAN に関係なく VLAN だけをディセーブルにするようにスイッチを設定します。以前のリリースでは、セキュリティ違反の原因であるデータ クライアントを認証しようとすると、ポート全体がシャットダウンし、接続が完全に切断されます。

この機能は、PC が IP Phone に接続されている IP Phone 環境で使用できます。データ VLAN でセキュリティ違反が検出されると、データ VLAN だけがシャットダウンされます。音声 VLAN のトラフィックは中断することなくスイッチで送受信されます。

コモン セッション ID

認証マネージャは、使用された認証方式が何であれ、クライアントの単一のセッション ID（共通セッション ID）を使用します。この ID は、表示コマンドや MIB などのすべてのレポートに使用されます。セッション ID は、セッション単位のすべての Syslog メッセージに表示されます。

セッション ID には、次の情報が含まれます。

- ネットワーク アクセス デバイス（NAD）の IP アドレス
- 一意の 32 ビット整数（機械的に増加します）
- セッション開始タイム スタンプ（32 ビット整数）

次に、show authentication コマンドの出力に表示されたセッション ID の例を示します。この例では、セッション ID は 160000050000000B288508E5 です。

```
スイッチ# show authentication sessions
Interface  MAC Address     Method   Domain   Status         Session ID
Fa4/0/4    0000.0000.0203  mab      DATA     Authz Success  160000050000000B288508E5
```

次に、Syslog 出力にセッション ID が表示される例を示します。この例でも、セッション ID は 160000050000000B288508E5 です。

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on Interface
Fa4/0/4 AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client
(0000.0000.0203) on Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

セッションIDは、NAD、AAAサーバー、その他のレポート分析アプリケーションでクライアントを識別するために使用されます。ID は自動的に表示されます。設定は必要ありません。

802.1x ポートベース認証の設定方法

802.1x 認証のデフォルト設定

表 *143 : 802.1x* 認証のデフォルト設定

機能	デフォルト設定
スイッチの 802.1x イネーブル ステート	ディセーブル
ポート単位の 802.1x イネーブル ステート	ディセーブル（force-authorized） ポートはクライアントとの 802.1x ベース認証を行わずに、通常のトラフィックを送受信します。
AAA	ディセーブル
RADIUS サーバ • IP アドレス • UDP 認証ポート • デフォルトのアカウンティング ポート • キー	• 指定なし • 1645 • 1646 • 指定なし
ホスト モード	シングル ホスト モード
制御方向	双方向制御
定期的な再認証	ディセーブル。
再認証の間隔（秒）	3600 秒
再認証回数	2回（ポートが無許可ステートに変わる前に、スイッチが認証プロセスを再開する回数）
待機時間	60 秒（スイッチがクライアントとの認証情報の交換に失敗した後、待機状態を続ける秒数）
再送信時間	30 秒（スイッチが EAP-Request/Identity フレームに対するクライアントからの応答を待ち、要求を再送信するまでの秒数）

機能	デフォルト設定
最大再送信回数	2 回（スイッチが認証プロセスを再開する前に、EAP-Request/Identity フレームを送信する回数）
クライアント タイムアウト時間	30 秒（認証サーバーからの要求をクライアントにリレーするとき、スイッチが返答を待ち、クライアントに要求を再送信するまでの時間）
認証サーバー タイムアウト時間	30秒（クライアントからの応答を認証サーバーにリレーするとき、スイッチが応答を待ち、応答をサーバーに再送信するまでの時間） dot1x timeout server-timeout インターフェイス コンフィギュレーション コマンドを使用して、このタイムアウト時間を変更できます。
無活動タイムアウト	ディセーブル
ゲスト VLAN	指定なし
アクセス不能認証バイパス	ディセーブル
制限付き VLAN	指定なし
オーセンティケータ（スイッチ）モード	指定なし
MAC 認証バイパス	ディセーブル
音声認識セキュリティ	ディセーブル

802.1x 認証設定時の注意事項

802.1X 認証

802.1x 認証を設定する場合の注意事項は、次のとおりです。

- 802.1x 認証をイネーブルにすると、他のレイヤ 2 またはレイヤ 3 機能がイネーブルになる前に、ポートが認証されます。
- 802.1x 対応ポートが割り当てられている VLAN が変更された場合、この変更は透過的でスイッチには影響しません。たとえば、ポートが RADIUS サーバに割り当ててられた VLAN に割り当てられ、再認証後に別の VLAN に割り当てられた場合に、この変更が発生します。

  802.1x ポートが割り当てられている VLAN がシャットダウン、ディセーブル、または削除される場合、ポートは無許可になります。たとえば、ポートが割り当てられたアクセス VLAN がシャットダウンまたは削除された後、ポートは無許可になります。

- 802.1x プロトコルは、レイヤ 2 スタティックアクセス ポート、音声 VLAN ポート、およびレイヤ 3 ルーテッド ポートでサポートされますが、次のポート タイプではサポートされません。
  - ダイナミック ポート：ダイナミック モードのポートは、ネイバーとトランク ポートへの変更をネゴシエートする場合があります。ダイナミック ポートで 802.1x 認証をイネーブルにしようとすると、エラー メッセージが表示され、802.1x 認証はイネーブルになりません。802.1x 対応ポートのモードをダイナミックに変更しようとしても、エラー メッセージが表示され、ポート モードは変更されません。
  - EtherChannel ポート：アクティブまたはアクティブでない EtherChannel メンバを 802.1x ポートとして設定しないでください。EtherChannel ポートで 802.1x 認証をイネーブルにしようとすると、エラー メッセージが表示され、802.1x 認証はイネーブルになりません。
  - スイッチド ポート アナライザ（SPAN）およびリモート SPAN（RSPAN）宛先ポート：SPAN または RSPAN 宛先ポートであるポートの 802.1x 認証をイネーブルにすることができます。ただし、ポートを SPAN または RSPAN 宛先ポートとして削除するまでは、802.1x 認証はディセーブルになります。SPAN または RSPAN 送信元ポートでは 802.1x 認証をイネーブルにすることができます。
- スイッチ上で、**dot1x system-auth-control** グローバル コンフィギュレーション コマンドを入力して 802.1x 認証をグローバルにイネーブルにする前に、802.1x 認証と EtherChannel が設定されているインターフェイスから、EtherChannel の設定を削除してください。
- Cisco IOS Release 12.2(55)SE 以降のリリースでは、802.1x 認証に関連するシステム メッセージのフィルタリングがサポートされています。

VLAN 割り当て、ゲスト VLAN、制限付き VLAN、アクセス不能認証バイパス

VLAN 割り当て、ゲスト VLAN、制限付き VLAN、およびアクセス不能認証バイパス設定時の注意事項は、次のとおりです。

- 802.1x 認証をポート上でイネーブルにすると、音声 VLAN の機能を持つポート VLAN は設定できません。
- RSPAN VLAN または音声 VLAN を除くあらゆる VLAN を、802.1x ゲスト VLAN として設定できます。ゲスト VLAN 機能は、内部 VLAN（ルーテッド ポート）またはトランク ポートではサポートされていません。アクセス ポート上でだけサポートされます。
- DHCP クライアントが接続されている 802.1x ポートのゲスト VLAN を設定した後、DHCP サーバからホスト IP アドレスを取得する必要があります。クライアント上の DHCP プロセスが時間切れとなり DHCP サーバーからホスト IP アドレスを取得しようとする前に、スイッチ上の 802.1x 認証プロセスを再起動する設定を変更できます。802.1x 認証プロセスの設定を減らします（**authentication timer inactivity** および **authentication timer reauthentication** インターフェイス コンフィギュレーション コマンド）。設定の減少量は、接続された 802.1x クライアントのタイプによって異なります。
- アクセス不能認証バイパス機能を設定する際には、次の注意事項に従ってください。

- この機能はシングル ホスト モードおよびマルチホスト モードの 802.1x ポートでサポートされます。
- Windows XP を稼働しているクライアントに接続されたポートがクリティカル認証ステートの場合、Windows XP はインターフェイスが認証されないと報告する場合があります。
- Windows XP クライアントで DHCP が設定され、DHCP サーバーからの IP アドレスがある場合、クリティカル ポートで EAP 認証成功メッセージを受信しても DHCP 設定プロセスを再初期化しません。
- アクセス不能認証バイパス機能および制限 VLAN を 802.1x ポート上に設定できます。スイッチが制限付き VLAN 内でクリティカル ポートを再認証しようとし、すべての RADIUS サーバーが利用不可能な場合、スイッチはポート ステートをクリティカル認証ステートに変更し、制限付き VLAN に残ります。
- CTS リンクがクリティカル認証モードである場合にアクティブスイッチがリロードすると、SGT をデバイスに設定したポリシーは新しいアクティブスイッチでは使用できません。これは、内部バインドが 3750-X スイッチ スタックのスタンバイ スイッチと同期しないためです。

- RSPAN VLAN または音声 VLAN を除くあらゆる VLAN を、802.1x 制限付き VLAN として設定できます。制限付き VLAN 機能は、内部 VLAN（ルーテッド ポート）またはトランク ポートではサポートされていません。アクセス ポート上でだけサポートされます。
- ワイヤレス ゲスト クライアントが固定クライアント VLAN の代わりに外部クライアント VLAN から IP を取得する際には、クライアントに新しい DHCP 要求の発行を求めるために、WLAN 設定で **ip dhcp required** コマンドを使用する必要があります。これは、クライアントがアンカーで正しくない IP を取得することを防止します。
- Cisco WLC（外部の）のリロード後に、有線ゲスト クライアントが IP アドレスの取得に失敗した場合は、クライアントによって使用されているポートで shut/no shut を実行して再接続します。

MAC 認証バイパス

MAC 認証バイパス設定時の注意事項は次のとおりです。

- 特に明記していないかぎり、MAC 認証バイパスの注意事項は 802.1x 認証のものと同じです。
- ポートが MAC アドレスで許可された後に、ポートから MAC 認証バイパスをディセーブルにしても、ポート ステートに影響はありません。
- ポートが未許可ステートであり、クライアント MAC アドレスが認証サーバ データベースにない場合、ポートは未許可ステートのままです。ただし、クライアント MAC アドレスがデータベースに追加されると、スイッチは MAC 認証バイパス機能を使用してポートを再認証できます。

- ポートが認証ステートにない場合、再認証が行われるまでポートはこのステートを維持します。
- MAC 認証バイパスにより接続されているが、非アクティブなホストのタイムアウト時間を設定できます。指定できる範囲は 1 ～ 65535 秒です。

ポートあたりのデバイスの最大数

802.1x 対応のポートに接続できるデバイスの最大数です。

- シングル ホスト モードの場合、アクセス VLAN で接続できるデバイスは 1 台だけです。ポートが音声 VLAN でも設定されている場合、音声 VLAN を介して送受信できる Cisco IP Phone の数には制限はありません。
- マルチドメイン認証（MDA）モードの場合、アクセス VLAN で 1 台のデバイス、音声 VLAN で 1 台の IP Phone が許可されます。
- マルチホスト モードでは、1 つの 802.1x サプリカントだけがポートで許可されますが、非 802.1x ホストは数に制限なく、アクセス VLAN で許可されます。音声 VLAN で許可されるデバイスの数には制限はありません。

802.1x 準備状態チェックの設定

802.1x 準備状態チェックは、すべてのスイッチ ポートの 802.1x アクティビティをモニタリングし、802.1x をサポートするポートに接続されているデバイスの情報を表示します。この機能を使用して、スイッチ ポートに接続されているデバイスが 802.1x に対応できるかどうかを判別できます。

802.1x 準備状態チェックは、802.1x で設定できるすべてのポートで使用できます。準備状態チェックは、**dot1x force-unauthorized** として設定されるポートでは使用できません。

802.1x 準備状態チェックをスイッチでイネーブルにする場合には、次の手順に従ってください。

始める前に

準備状態チェックをスイッチでイネーブルにする場合、次の注意事項に従ってください。

- 準備状態チェックは通常、802.1x がスイッチでイネーブルにされる前に使用されます。
- インターフェイスを指定せずに **dot1x test eapol-capable** 特権 EXEC コマンドを使用すると、スイッチ スタックのすべてのポートがテストされます。
- **dot1x test eapol-capable** コマンドを 802.1x 対応のポートで設定し、リンクがアップになると、ポートは、802.1x に対応するかどうか、接続クライアントでクエリーを実行します。クライアントが通知パケットに応答すると、802.1x 対応です。クライアントがタイムアウト時間内に応答すると Syslog メッセージが生成されます。クライアントがクエリーに応答しない場合、クライアントは 802.1x 対応ではありません。Syslog メッセージは生成されません。

- **dot1x test eapol-capable** コマンドを 802.1x 対応のポートで設定し、リンクがアップになると、ポートは、802.1x に対応するかどうか、接続クライアントでクエリーを実行します。クライアントが通知パケットに応答すると、802.1x 対応です。クライアントがタイムアウト時間内に応答すると Syslog メッセージが生成されます。クライアントがクエリーに応答しない場合、クライアントは 802.1x 対応ではありません。Syslog メッセージは生成されません。
- 準備状態チェックは、複数のホスト（たとえば、IP Phone に接続される PC）を扱うポートに送信できます。Syslog メッセージは、タイマー時間内に準備状態チェックに応答する各クライアントに生成されます。

手順の概要

1. **enable**
2. **dot1x test eapol-capable** [**interface** *interface-id*]
3. **configure terminal**
4. **dot1x test timeout** *timeout*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**dot1x test eapol-capable** [**interface** *interface-id*] 例： `スイッチ# dot1x test eapol-capable interface gigabitethernet1/0/13` `DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL capable`	スイッチ上で 802.1x 準備状態チェックをイネーブルにします。 （任意）*interface-id* では、IEEE 802.1x の準備状態をチェックするポートを指定します。 （注） オプションの **interface** キーワードを省略した場合、スイッチのすべてのインターフェイスがテストされます。
ステップ 3	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**dot1x test timeout** *timeout* 例： `スイッチ(config)# dot1x test timeout 54`	（任意）EAPOL 応答の待機に使用するタイムアウトを設定します。範囲は 1 ～ 65535 秒です。デフォルトは 10 秒です。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

音声認識 802.1x セキュリティの設定

（注） 音声認識 IEEE 802.1x 認証を使用するには、スイッチが LAN Base イメージを実行している必要があります。

音声認識 802.1x セキュリティ機能をスイッチで使用して、セキュリティ違反が発生した場合にデータまたは音声 VLAN に関係なく VLAN だけをディセーブルにします。この機能は、PC が IP Phone に接続されている IP Phone 環境で使用できます。データ VLAN でセキュリティ違反が検出されると、データ VLAN だけがシャットダウンされます。音声 VLAN のトラフィックは中断することなくスイッチで送受信されます。

スイッチで音声認識 802.1x 音声セキュリティを設定する場合、次の注意事項に従ってください。

- 音声認識 802.1x セキュリティをイネーブルにするには、**errdisable detect cause security-violation shutdown vlan** グローバル コンフィギュレーション コマンドを入力します。音声認識 802.1x セキュリティをディセーブルにするには、このコマンドの **no** バージョンを入力します。このコマンドは、スイッチの 802.1x 設定ポートのすべてに適用されます。

（注） **shutdown vlan** キーワードを指定しない場合、error-disabled ステートになった際にポート全体がシャットダウンされます。

- **errdisable recovery cause security-violation** グローバル コンフィギュレーション コマンドを使用して、error-disabled リカバリを設定すると、ポートは自動的に再びイネーブルにされます。error-disabled リカバリがポートで設定されていない場合、**shutdown** および **no shutdown** インターフェイス コンフィギュレーション コマンドを使用してポートを再びイネーブルにします。
- 個々の VLAN を再びイネーブルにするには、**clear errdisable interface** *interface-id* **vlan** [*vlan-list*] 特権 EXEC コマンドを使用します。範囲を指定しない場合、ポートのすべての VLAN がイネーブルにされます。

音声認識 802.1x セキュリティをイネーブルにするには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **errdisable detect cause security-violation shutdown vlan**
3. **errdisable recovery cause security-violation**
4. **clear errdisable interface***interface-id* **vlan** *[vlan-list]*
5. 次を入力します。
   - **shutdown**
   - **no shutdown**
6. **end**
7. **show errdisable detect**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**errdisable detect cause security-violation shutdown vlan**	セキュリティ違反エラーが発生したすべての VLAN をシャットダウンします。 （注） **shutdown vlan** キーワードを指定しない場合、すべてのポートが errdisable ステートになり、シャットダウンされます。
ステップ 3	**errdisable recovery cause security-violation**	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**clear errdisable interface***interface-id* **vlan** *[vlan-list]*	（任意）errordisable になっている個々の VLAN を再びイネーブルにします。 • interface-id の場合、個々の VLAN を再びイネーブルにするポートを指定します。 • （任意）vlan-list の場合、再びイネーブルにする VLAN のリストを指定します。vlan-list を指定しない場合は、すべての VLAN が再びイネーブルになります。
ステップ 5	次を入力します。 • **shutdown** • **no shutdown**	（任意）errordisable の VLAN を再びイネーブルにして、すべての errordisable 指示をクリアします。
ステップ 6	**end**	特権 EXEC モードに戻ります。
ステップ 7	**show errdisable detect**	入力内容を確認します。

例

次に、セキュリティ違反エラーが発生した任意の VLAN をシャットダウンするようにスイッチを設定する例を示します。

```
Switch(config)# errdisable detect cause security-violation shutdown vlan
```

次に、ポート ギガビット イーサネット 40/2 で errdisable ステートであったすべての VLAN を再度イネーブルにする方法を示します。

```
Switch# clear errdisable interface gigabitethernet40/2
vlan
```

show errdisable detect 特権 EXEC コマンドを入力すると、設定を確認できます。

802.1x 違反モードの設定

次に示す状況で、シャットダウン、Syslog エラーを生成、または新しいデバイスからのパケットを廃棄するように 802.1x ポートを設定できます。

- デバイスが 802.1x 対応のポートに接続した
- ポートで認証されるデバイスの最大数に達した

スイッチ上にセキュリティ違反アクションを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**

2. **aaa new-model**
3. **aaa authentication dot1x**{ **default** } *method1*
4. **interface** *interface-id*
5. **switchport mode access**
6. **authentication violation** {**shutdown** | **restrict** | **protect** | **replace**}
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**aaa new-model** 例： `スイッチ(config)# aaa new-model`	AAA をイネーブルにします。
ステップ 3	**aaa authentication dot1x**{ **default** } *method1* 例： `スイッチ(config)# aaa authentication dot1x default group radius`	802.1x 認証方式リストを作成します。 **authentication** コマンドにリストが指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。 *method1* には、**group radius** キーワードを入力して、認証用のすべての RADIUS サーバリストを使用できるようにします。
ステップ 4	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/4`	IEEE 802.1x 認証をイネーブルにするクライアントに接続しているポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**switchport mode access** 例： `スイッチ(config-if)# switchport mode access`	ポートをアクセス モードに設定します。

	コマンドまたはアクション	目的
ステップ 6	**authentication violation** {**shutdown** | **restrict** | **protect** | **replace**} 例： `スイッチ(config-if)# authentication violation restrict`	違反モードを設定します。キーワードの意味は次のとおりです。 • **shutdown**：エラーによってポートがディセーブルになります。 • **restrict**：Syslog エラーを生成します。 • **protect**：トラフィックをポートに送信するすべての新しいデバイスからパケットをドロップします。 • **replace**：現在のセッションを削除し、新しいホストで認証します。
ステップ 7	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

802.1X 認証の設定

ユーザー単位 ACL または VLAN 割り当てを可能にするには、AAA 許可をイネーブルにしてネットワーク関連のすべてのサービス要求に対してスイッチを設定する必要があります。

次に、802.1x の AAA プロセスを示します。

始める前に

802.1x ポートベース認証を設定するには、認証、許可、アカウンティング（AAA）をイネーブルにして認証方式リストを指定する必要があります。方式リストは、ユーザ認証のためにクエリー送信を行う手順と認証方式を記述したものです。

手順の概要

1. ユーザーがスイッチのポートに接続します。
2. 認証が実行されます。
3. RADIUS サーバ設定に基づいて、VLAN 割り当てが適宜イネーブルになります。
4. スイッチが開始メッセージをアカウンティング サーバーに送信します。
5. 必要に応じて、再認証が実行されます。
6. スイッチが仮のアカウンティングアップデートを、再認証結果に基づいたアカウンティング サーバーに送信します。
7. ユーザーがポートから切断します。
8. スイッチが停止メッセージをアカウンティング サーバーに送信します。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	ユーザーがスイッチのポートに接続します。	
ステップ 2	認証が実行されます。	
ステップ 3	RADIUS サーバ設定に基づいて、VLAN 割り当てが適宜イネーブルになります。	
ステップ 4	スイッチが開始メッセージをアカウンティング サーバーに送信します。	
ステップ 5	必要に応じて、再認証が実行されます。	
ステップ 6	スイッチが仮のアカウンティング アップデートを、再認証結果に基づいたアカウンティング サーバーに送信します。	
ステップ 7	ユーザーがポートから切断します。	
ステップ 8	スイッチが停止メッセージをアカウンティング サーバーに送信します。	

802.1x ポートベース認証の設定

802.1x ポートベース認証を設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x**{ **default** } *method1*
4. **dot1x system-auth-control**
5. **aaa authorization network** {**default**} **group radius**
6. **radius-server host** *ip-address*
7. **radius-server key** *string*
8. **interface** *interface-id*
9. **switchport mode access**
10. **authentication port-control auto**
11. **dot1x pae authenticator**
12. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**aaa new-model** 例： スイッチ(config)# **aaa new-model**	AAA をイネーブルにします。
ステップ 3	**aaa authentication dot1x**{ **default** } *method1* 例： スイッチ(config)# **aaa authentication dot1x default group radius**	802.1x 認証方式リストを作成します。 **authentication** コマンドにリストが指定されていない場合に使用するデフォルトのリストを作成するには、**default** キーワードの後ろにデフォルト状況で使用される方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。 *method1* には、**group radius** キーワードを入力して、認証用のすべての RADIUS サーバリストを使用できるようにします。 （注） コマンドラインのヘルプストリングには他のキーワードも表示されますが、サポートされるのは **group radius** キーワードのみです。
ステップ 4	**dot1x system-auth-control** 例： スイッチ(config)# **dot1x system-auth-control**	スイッチで 802.1x 認証をグローバルに有効にします。
ステップ 5	**aaa authorization network** {**default**} **group radius** 例： スイッチ(config)# **aaa authorization network default group radius**	（任意）ユーザー単位 ACL や VLAN 割り当てなど、ネットワーク関連のすべてのサービス要求に対するユーザー RADIUS 許可をスイッチに設定します。
ステップ 6	**radius-server host** *ip-address* 例： スイッチ(config)# **radius-server host 124.2.2.12**	（任意）RADIUS サーバーの IP アドレスを指定します。

	コマンドまたはアクション	目的
ステップ 7	**radius-server key** *string* 例： `スイッチ(config)# radius-server key abc1234`	（任意）RADIUS サーバー上で動作する RADIUS デーモンとスイッチの間で使用する認証および暗号キーを指定します。
ステップ 8	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/2`	IEEE 802.1x 認証をイネーブルにするクライアントに接続しているポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 9	**switchport mode access** 例： `スイッチ(config-if)# switchport mode access`	（任意）ステップ 6 および 7 で RADIUS サーバを設定した場合のみ、ポートをアクセス モードに設定します。
ステップ 10	**authentication port-control auto** 例： `スイッチ(config-if)# authentication port-control auto`	ポートでの 802.1x 認証を有効にします。
ステップ 11	**dot1x pae authenticator** 例： `スイッチ(config-if)# dot1x pae authenticator`	インターフェイスのポート アクセス エンティティを、オーセンティケータとしてのみ動作し、サプリカント用のメッセージは無視するように設定します。
ステップ 12	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

スイッチと RADIUS サーバー間の通信の設定

すべての RADIUS サーバについて、タイムアウト、再送信回数、および暗号キー値をグローバルに設定するには、**radius-server host** グローバル コンフィギュレーション コマンドを使用します。これらのオプションをサーバ単位で設定するには、**radius-server timeout**、**radius-server retransmit**、および **radius-server key** グローバル コンフィギュレーション コマンドを使用します。

RADIUSサーバ上でも、いくつかの値を設定する必要があります。これらの設定値としては、スイッチの IP アドレス、およびサーバーとスイッチの双方で共有するキー ストリングがあります。詳細については、RADIUS サーバーのマニュアルを参照してください。

スイッチでRADIUSサーバーのパラメータを設定するには、次の手順を実行します。この手順は必須です。

始める前に

認証、許可、およびアカウンティング（AAA）をイネーブルにし、認証方式リストを指定する必要があります。方式リストは、ユーザ認証のためにクエリ送信を行う手順と認証方式を記述したものです。

手順の概要

1. **enable**
2. **configure terminal**
3. **radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string* 例： `スイッチ(config)# radius-server host 125.5.5.43 auth-port 1645 key rad123`	RADIUS サーバー パラメータを設定します。 *hostname* | *ip-address* には、リモート RADIUS サーバーのサーバー名またはIPアドレスを指定します。 **auth-port** *port-number* には、認証要求の UDP 宛先ポートを指定します。デフォルトは 1645 です。指定できる範囲は 0 ～ 65536 です。 **key** *string* には、スイッチと、RADIUSサーバー上で動作するRADIUSデーモンとの間で使用する、認証および暗号キーを指定します。キーは、RADIUS サーバーで使用する暗号化キーに一致するテキスト ストリングでなければなりません。

	コマンドまたはアクション	目的
		（注） キーの先行スペースは無視されますが、途中および末尾のスペースは有効なので、キーは必ず **radius-server host** コマンドシンタックスの最後の項目として設定してください。キーにスペースを使用する場合は、引用符がキーの一部分である場合を除き、引用符でキーを囲まないでください。キーは RADIUS デーモンで使用する暗号に一致している必要があります。 複数の RADIUS サーバーを使用する場合には、このコマンドを繰り返し入力します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

ホスト モードの設定

authentication port-control インターフェイス コンフィギュレーション コマンドが **auto** に設定されている IEEE 802.1x 許可ポート上で、複数のホスト（クライアント）を許可するには、特権 EXEC モードで次の手順を実行します。マルチドメイン認証（MDA）を設定してイネーブルにするには、**multi-domain** キーワードを使用します。これにより、ホストデバイス、および IP Phone（シスコ製または他社製）など音声デバイスの両方が同じスイッチポートで許可されます。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication host-mode**[**multi-auth** |**multi-domain** |**multi-host** |**single-host**]
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet2/0/1**	複数ホストが間接的に接続されているポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**authentication host-mode**[**multi-auth** |**multi-domain** |**multi-host** |**single-host**] 例： スイッチ(config-if)# **authentication host-mode multi-host**	単一の 802.1x 許可ポートで複数のホスト（クライアント）を許可することができます。 キーワードの意味は次のとおりです。 • **multi-auth**：音声 VLAN とデータ VLAN の両方で複数の認証クライアントを許可します。 （注） **multi-auth** キーワードは、**authentication host-mode** コマンドでのみ使用でき ます。 • **multi-host**：シングル ホストの認証後に 802.1x 許可ポートで複数のホスト（クライアント）の接続を許可します。 • **multi-domain**：ホスト デバイスと IP Phone（シスコ製または他社製）など音声デバイスの両方が、IEEE 802.1x 許可ポートで認証されるようにします。 （注） ホストモードが **multi-domain** に設定されている場合、IP Phone の音声 VLAN を設定する必要があります。 指定のインターフェイスに対し **authentication port-control** インターフェイス コンフィギュレーション コマンドが **auto** に設定されていることを確認してください。
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

定期的な再認証の設定

802.1x クライアントの定期的な再認証をイネーブルにし、再認証の間隔を指定できます。再認証を行う間隔を指定しない場合、3600 秒おきに再認証が試みられます。

クライアントの定期的な再認証をイネーブルにし、再認証を行う間隔（秒）を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication periodic**
4. **authentication timer** {{[**inactivity** | **reauthenticate** | **restart** | **unauthorized**]} {*value*}}
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/1`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**authentication periodic** 例： `スイッチ(config-if)# authentication periodic`	クライアントの定期的な再認証（デフォルトではディセーブル）をイネーブルにします。 （注） デフォルト値は 3600 秒です。再認証タイマーの値を変更するか、スイッチに RADIUS-provided セッションタイムアウトを使用させるには、**authentication timer reauthenticate** コマンドを入力します。
ステップ 4	**authentication timer** {{[**inactivity** | **reauthenticate** | **restart** | **unauthorized**]} {*value*}} 例： `スイッチ(config-if)# authentication timer reauthenticate 180`	再認証の試行の間隔（秒）を設定します。 **authentication timer** キーワードの意味は次のとおりです。 • **inactivity**：クライアントからのアクティビティがなくなり無許可になるまでの間隔（秒） • **reauthenticate**：自動再認証試行が開始されるまでの時間（秒） • **restart** *value*：無許可ポートの認証の試行が行われるまでの間隔（秒）

	コマンドまたはアクション	目的
		• **unauthorized** *value*：不正セッションが削除されるまでの間隔（秒） このコマンドがスイッチの動作に影響するのは、定期的な再認証をイネーブルに設定した場合だけです。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

待機時間の変更

スイッチはクライアントを認証できなかった場合に、所定の時間だけアイドル状態を続け、その後再び認証を試みます。**authentication timer restart** インターフェイス コンフィギュレーション コマンドは、アイドル状態の期間を制御します。認証が失敗する理由としては、クライアントが無効なパスワードを提示した場合などが考えられます。デフォルトよりも小さい値を入力することによって、ユーザへの応答時間を短縮できます。

待機時間を変更するには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer restart** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例：	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ(config)# **interface gigabitethernet2/0/1**	
ステップ 3	**authentication timer restart** *seconds* 例： スイッチ(config-if)# **authentication timer restart 30**	クライアントとの認証のやり取りに失敗した場合に、スイッチが待機状態のままでいる秒数を設定します。 指定できる範囲は 1 ～ 65535 秒です。デフォルトは 60 秒です。
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show authentication sessions interface** *interface-id* 例： スイッチ# **show authentication sessions interface gigabitethernet2/0/1**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

スイッチからクライアントへの再送信時間の変更

クライアントはスイッチからの EAP-Request/Identity フレームに対し、EAP-Response/Identity フレームで応答します。スイッチがこの応答を受信できなかった場合、所定の時間（再送信時間）だけ待機し、その後フレームを再送信します。

（注） このコマンドのデフォルト値は、リンクの信頼性が低下した場合や、特定のクライアントおよび認証サーバーの動作に問題がある場合など、異常な状況に対する調整を行う必要があるときに限って変更してください。

スイッチがクライアントからの通知を待機する時間を変更するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer reauthenticate** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/1`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**authentication timer reauthenticate** *seconds* 例： `スイッチ(config-if)# authentication timer reauthenticate 60`	スイッチが EAP-Request/Identity フレームに対するクライアントからの応答を待ち、要求を再送信するまでの秒数を設定します。 指定できる範囲は 1 ～ 65535 秒です。デフォルトは 5 秒です。
ステップ 4	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show authentication sessions interface** *interface-id* 例： `スイッチ# show authentication sessions interface gigabitethernet2/0/1`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

スイッチからクライアントへのフレーム再送信回数の設定

スイッチからクライアントへの再送信時間を変更できるだけでなく、（クライアントから応答が得られなかった場合に）スイッチが認証プロセスを再起動する前に、クライアントにEAP-Request/Identity フレームを送信する回数を変更できます。

（注） このコマンドのデフォルト値は、リンクの信頼性が低下した場合や、特定のクライアントおよび認証サーバーの動作に問題がある場合など、異常な状況に対する調整を行う必要があるときに限って変更してください。

スイッチからクライアントへのフレーム再送信回数を設定するには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x max-reauth-req** *count*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet2/0/1**	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**dot1x max-reauth-req** *count* 例：	スイッチが認証処理を再開するまでに、クライアントへ EAP 要求/アイデンティティ フレームを送信する回数を変更できます。指定できる範囲は 1 ～ 10 です。デフォルトは 2 です。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **dot1x max-reauth-req 5**	
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

再認証回数の設定

ポートが無許可ステートに変わる前に、スイッチが認証プロセスを再開する回数を変更することもできます。

（注） このコマンドのデフォルト値は、リンクの信頼性が低下した場合や、特定のクライアントおよび認証サーバの動作に問題がある場合など、異常な状況に対する調整を行う必要がある際に限って変更してください。

再認証回数を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **dot1x max-req** *count*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例：	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ# **interface gigabitethernet2/0/1**	
ステップ 3	**switchport mode access** 例： スイッチ(config-if)# **switchport mode access**	RADIUS サーバを事前に設定した場合に限り、ポートをアクセス モードに設定します。
ステップ 4	**dot1x max-req** *count* 例： スイッチ(config-if)# **dot1x max-req 4**	ポートが無許可ステートに変わる前に、スイッチが認証プロセスを再開する回数を設定します。指定できる範囲は 0 ～ 10 です。デフォルトは 2 です。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

MAC 移動のイネーブル化

MAC 移動を使用すると、認証されたホストをスイッチのポート間で移動できます。

スイッチで MAC 移動をグローバルにイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **authentication mac-move permit**
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**authentication mac-move permit** 例： `スイッチ(config)# authentication mac-move permit`	スイッチでMAC移動をイネーブルにします。デフォルトは deny です。 セッション認識型ネットワーク モードでは、デフォルト CLI は **access-session mac-move deny** です。セッション認識型ネットワークで MAC 移動をイネーブルにするには、**no access-session mac-move** グローバル コンフィギュレーション コマンドを使用します。 **mac-move** のデフォルト値は、レガシーモード（IBNS 1.0）の場合は **deny** で、C3PL モード（IBNS 2.0）の場合は **permit** です。
ステップ 3	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

MAC 移動の無効化

スイッチのセキュアポートから非セキュアポートへの MAC 移動をディセーブルにするには、特権 EXEC モードを開始して、次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **authentication mac-move deny-uncontrolled**
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**authentication mac-move deny-uncontrolled** 例： スイッチ(config)# **authentication mac-move deny-uncontrolled**	スイッチで MAC 移動をディセーブルにします。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MAC 置換のイネーブル化

MAC 置換を使用すると、ホストはポート上の認証ホストを置換できます。

インターフェイス上で MAC 置換をイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication violation** {**protect** | **replace** | **restrict** | **shutdown**}
4. **end**

5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/2`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**authentication violation** {**protect** | **replace** | **restrict** | **shutdown**} 例： `スイッチ(config-if)# authentication violation replace`	インターフェイス上で MAC 置換をイネーブルにするには、**replace** キーワードを使用します。ポートが現在のセッションを削除し、新しいホストを使用して認証を開始します。 他のキーワードは、次のような機能があります。 • **protect**：ポートは、システム メッセージを生成せずに、予期しない MAC を使用するパケットをドロップします。 • **restrict**：違反パケットが CPU によってドロップされ、システム メッセージが生成されます。 • **shutdown**：ポートは、予期しない MAC アドレスを受信すると error disabled になります。
ステップ 4	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。

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

802.1x アカウンティングの設定

802.1xアカウンティングを使用して、AAAシステムアカウンティングをイネーブルにすると、ロギングのためにシステム リロード イベントをアカウンティング RADIUS サーバに送信できます。サーバは、アクティブな 802.1x セッションすべてが終了したものと判断します。

（注） Cisco IOS XE Denali 16.3.x および Cisco IOS XE Everest 16.6.x では、定期的な AAA アカウンティングのアップデートはサポートされていません。スイッチは、定期的中間アカウンティングレコードをアカウンティングサーバーに送信しません。定期的なAAAアカウンティングのアップデートは、Cisco IOS XE Fuji 16.9.x 以降のリリースで利用できます。

RADIUS は信頼性の低い UDP トランスポート プロトコルを使用するため、ネットワーク状態が良好でないと、アカウンティング メッセージが失われることがあります。設定した回数のアカウンティング要求の再送信後、スイッチがRADIUSサーバーからアカウンティング応答メッセージを受信しない場合、次のメッセージが表示されます。

```
Accounting message %s for session %s failed to receive Accounting Response.
```

このストップ メッセージが正常に送信されない場合、次のメッセージが表示されます。

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```

（注） ロギングの開始、停止、仮のアップデート メッセージ、タイム スタンプなどのアカウンティング タスクを実行するように、RADIUS サーバを設定する必要があります。これらの機能をオンにするには、RADIUS サーバーの [Network Configuration] タブの [Update/Watchdog packets from this AAA client] のロギングをイネーブルにします。次に、RADIUS サーバーの [System Configuration] タブの [CVS RADIUS Accounting] をイネーブルにします。

AAA がスイッチでイネーブルになった後、802.1x アカウンティングを設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **aaa accounting dot1x default start-stop group radius**
4. **aaa accounting system default start-stop group radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/3`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**aaa accounting dot1x default start-stop group radius** 例： `スイッチ(config-if)# aaa accounting dot1x default start-stop group radius`	すべてのRADIUSサーバのリストを使用して802.1xアカウンティングをイネーブルにします。
ステップ 4	**aaa accounting system default start-stop group radius** 例： `スイッチ(config-if)# aaa accounting system default start-stop group radius`	（任意）システムアカウンティングをイネーブルにし（すべてのRADIUSサーバーのリストを使用）、スイッチがリロードするときにシステム アカウンティング リロード イベント メッセージを生成します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例：	入力を確認します。

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

ゲスト VLAN の設定

サーバが EAP Request/Identity フレームに対する応答を受信しない場合、ゲスト VLAN を設定すると、802.1x 対応でないクライアントはゲスト VLAN に配置されます。802.1x 対応であっても、認証に失敗したクライアントは、ネットワークへのアクセスが許可されません。スイッチは、シングル ホスト モードまたはマルチ ホスト モードでゲスト VLAN をサポートします。

ゲスト VLAN を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. 次のいずれかを使用します。
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 2/0/2**	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	次のいずれかを使用します。 • **switchport mode access** • **switchport mode private-vlan host** 例： `スイッチ(config-if)# switchport mode private-vlan host`	• ポートをアクセス モードに設定します。 • レイヤ 2 ポートをプライベート VLAN ホスト ポートとして設定します。
ステップ 4	**authentication event no-response action authorize vlan** *vlan-id* 例： `スイッチ(config-if)# authentication event no-response action authorize vlan 2`	アクティブ VLAN を 802.1x ゲスト VLAN として指定します。指定できる範囲は 1 ～ 4094 です。 内部 VLAN（ルーテッド ポート）、RSPAN VLAN、音声 VLAN を除くあらゆるアクティブ VLAN を 802.1x ゲスト VLAN として設定できます。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

制限付き VLAN の設定

スイッチスタックまたはスイッチ上に制限付き VLAN を設定している場合、認証サーバーが有効なユーザー名またはパスワードを受信できないと、IEEE 802.1x に準拠しているクライアントは制限付き VLAN に移されます。スイッチは、シングル ホスト モードでのみ制限付き VLAN をサポートします。

制限付き VLAN を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. 次のいずれかを使用します。
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/2`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかを使用します。 • **switchport mode access** • **switchport mode private-vlan host** 例： `スイッチ(config-if)# switchport mode access`	• ポートをアクセス モードに設定します。 • レイヤ 2 ポートをプライベート VLAN ホスト ポートとして設定します。
ステップ 4	**authentication port-control auto** 例： `スイッチ(config-if)# authentication port-control auto`	ポートでの 802.1x 認証をイネーブルにします。
ステップ 5	**authentication event fail action authorize vlan** *vlan-id* 例： `スイッチ(config-if)# authentication event fail action authorize vlan 2`	アクティブ VLAN を 802.1x 制限付き VLAN として指定します。指定できる範囲は 1 ～ 4094 です。 内部 VLAN（ルーテッド ポート）、RSPAN VLAN または音声 VLAN を除き、任意のアクティブ VLAN を 802.1x 制限 VLAN として設定できます。
ステップ 6	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

制限付き VLAN の認証試行回数の設定

ユーザーに制限付き VLAN を割り当てる前に、**authentication event retry** *retry count* インターフェイス コンフィギュレーション コマンドを使用して、認証試行回数を最大に設定できます。指定できる試行回数は 1 ～ 3 です。デフォルトは 3 回に設定されています。

認証試行回数を最大に設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. 次のいずれかを使用します。
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **authentication event retry** *retry count*
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/3`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかを使用します。 • **switchport mode access** • **switchport mode private-vlan host** 例： または `スイッチ(config-if)# switchport mode access`	• ポートをアクセス モードに設定します。 • レイヤ 2 ポートをプライベート VLAN ホスト ポートとして設定します。

	コマンドまたはアクション	目的
ステップ 4	**authentication port-control auto** 例： `スイッチ(config-if)# authentication port-control auto`	ポートでの 802.1x 認証をイネーブルにします。
ステップ 5	**authentication event fail action authorize vlan** *vlan-id* 例： `スイッチ(config-if)# authentication event fail action authorize vlan 8`	アクティブ VLAN を 802.1x 制限付き VLAN として指定します。指定できる範囲は 1 ～ 4094 です。 内部 VLAN（ルーテッド ポート）、RSPAN VLAN または音声 VLAN を除き、任意のアクティブ VLAN を 802.1x 制限 VLAN として設定できます。
ステップ 6	**authentication event retry** *retry count* 例： `スイッチ(config-if)# authentication event retry 2`	ポートが制限付き VLAN に移行するための認証試行回数を指定します。指定できる範囲は 1 ～ 3 秒です。デフォルトは 3 回に設定されています。
ステップ 7	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

クリティカル音声 VLAN を使用した 802.1x アクセス不能認証バイパスの設定

ポートにクリティカル音声 VLAN を設定し、アクセス不能認証バイパス機能をイネーブルにするには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **aaa new-model**
3. **radius-server dead-criteria**{**time** *seconds* } [**tries** *number*]
4. **radius-serverdeadtime**分
5. **radius-server host ip-address** *address*[**acct-port** *udp-port*][**auth-port** *udp-port*] [**testusername** *name*[**idle-time** *time*] [**ignore-acct-port**][**ignore auth-port**]] [**key** *string*]
6. **dot1x critical** {**eapol** | **recovery delay** *milliseconds*}
7. **interface** *interface-id*
8. **authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*]
9. **switchport voice vlan** *vlan-id*

10. **authentication event server dead action authorize voice**
11. **show authentication interface** *interface-id*
12. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**aaa new-model** 例： スイッチ(config)# **aaa new-model**	AAA をイネーブルにします。
ステップ 3	**radius-server dead-criteria**{**time** *seconds* } [**tries** *number*] 例： スイッチ(config)# **radius-server dead-criteria time 20 tries 10**	RADIUS サーバーが使用不可またはダウン（切断）と見なされる条件を設定します。 • **time**：1 ～ 120 秒。スイッチは、デフォルトの *seconds* 値を 10 ～ 60 の間で動的に決定します。 • **number**：1 ～ 100 の試行回数。スイッチは、デフォルトの **tries***number* を 10 ～ 100 の間で動的に決定します。
ステップ 4	**radius-serverdeadtime**分 例： スイッチ(config)# **radius-server deadtime 60**	（任意）RADIUS サーバに要求が送信されない分数を設定します。指定できる範囲は 0 ～ 1440 分（24 時間）です。デフォルト値は 0 分です。
ステップ 5	**radius-server host ip-address** *address*[**acct-port** *udp-port*][**auth-port** *udp-port*] [**testusername** *name*[**idle-time** *time*] [**ignore-acct-port**][**ignore auth-port**]] [**key** *string*] 例： スイッチ(config)# **radius-server host 10.0.0.10 acct-port 1550 auth-port 1560 test username user1 idle-time 30 key abc1234**	（任意）次のキーワードを使用して RADIUS サーバー パラメータを設定します。 • **acct-port***udp-port*：RADIUS アカウンティング サーバーの UDP ポートを指定します。UDP ポート番号の範囲は 0 ～ 65536 です。デフォルトは 1646 です。 • **auth-port***udp-port*：RADIUS 認証サーバーの UDP ポートを指定します。UDP ポート番号の範囲は 0 ～ 65536 です。デフォルトは 1645 です。

	コマンドまたはアクション	目的
		（注） RADIUS アカウンティング サーバーの UDP ポートと RADIUS 認証サーバーの UDP ポートを非デフォルト値に設定します。 • **test username***name*：RADIUS サーバー ステータスの自動テストをイネーブルにして、使用するユーザー名を指定します。 • **idle-time** time：スイッチがテスト パケットをサーバーに送信した後の間隔を分数で設定します。範囲は 1 ～ 35791 分です。デフォルトは 60 分（1 時間）です。 • **ignore-acct-port**：RADIUS サーバー アカウンティング ポートのテストをディセーブルにします。 • **ignore-auth-port**：RADIUS サーバー認証ポートのテストをディセーブルにします。 • **key***string* には、スイッチと RADIUS サーバー上で動作する RADIUS デーモンとの間で使用する認証および暗号キーを指定します。キーは、RADIUS サーバーで使用する暗号化キーに一致するテキスト ストリングでなければなりません。 （注） キーの先行スペースは無視されますが、途中および末尾のスペースは有効なので、キーは必ず **radius-server host** コマンド構文の最後の項目として設定してください。キーにスペースを使用する場合は、引用符がキーの一部分である場合を除き、引用符でキーを囲まないでください。キーは RADIUS デーモンで使用する暗号に一致している必要があります。 **radius-server key** {**0**string | **7***string* | *string*} グローバル コンフィギュレーション コマンドを使用しても認証および暗号キーを設定できます。
ステップ 6	**dot1x critical** {**eapol** | **recovery delay** *milliseconds*} 例：	（任意）アクセス不能認証バイパスのパラメータを設定します。

	コマンドまたはアクション	目的
	スイッチ(config)# **dot1x critical eapol** (config)# **dot1x critical recovery delay 2000**	• **eapol**：スイッチがクリティカル ポートを正常に認証すると、スイッチが EAPOL 成功メッセージを送信するように指定します。 • **recovery delay***milliseconds*：使用できない RADIUS サーバーが使用できるようになったときに、スイッチがクリティカル ポートを再初期化するために待機する回復遅延期間を設定します。指定できる範囲は 1 ～ 10000 ミリ秒です。デフォルトは 1000 ミリ秒です（ポートは毎秒再初期化できます）。
ステップ 7	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8	**authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*] 例： スイッチ(config-if)# **authentication event server dead action** **reinitialicze vlan 20**	これらのキーワードを使用して、RADIUS サーバが到達不能な場合にポートでホストを移動します。 • **authorize**：認証しようとする新しいホストをユーザー指定のクリティカル VLAN に移動します。 • **reinitialize**：ポートのすべての許可済みホストをユーザー指定のクリティカル VLAN に移動します。
ステップ 9	**switchport voice vlan** *vlan-id* 例： スイッチ(config-if)# **switchport voice vlan**	ポートの音声 VLAN を指定します。音声 VLAN はステップ 6 で設定されたクリティカルデータ VLAN と同じにはできません。
ステップ 10	**authentication event server dead action authorize voice** 例： スイッチ(config-if)# **authentication event server dead action** **authorize voice**	RADIUS サーバが到達不能な場合、ポートのデータトラフィックを音声 VLAN に移動するために、クリティカル音声 VLAN を設定します。
ステップ 11	**show authentication interface** *interface-id* 例：	（任意）設定を確認します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **do show authentication interface gigabit 1/0/1**	
ステップ 12	**copy running-config startup-config** 例： スイッチ(config-if)# **do copy running-config startup-config**	（任意）設定を確認します。

例

RADIUS サーバーのデフォルト設定に戻すには、**no radius-server dead-criteria**、**no radius-server deadtime**、および **no radius-server host** グローバル コンフィギュレーション コマンドを使用します。アクセス不能な認証バイパスをディセーブルにするには、**no authentication event server dead action** インターフェイス コンフィギュレーション コマンドを使用します。クリティカル音声 VLAN をディセーブルにするには、**authentication event server dead action authorize voice** インターフェイス コンフィギュレーション コマンドを使用します。

アクセス不能認証バイパスの設定例

次に、アクセス不能認証バイパス機能を設定する例を示します。

```
スイッチ(config)# radius-server dead-criteria time 30 tries 20
スイッチ(config)# radius-server deadtime 60
スイッチ(config)# radius-server host 10.0.0.10 acct-port 1550 auth-port 1560 test username
 user1 idle-time 30 key abc1234
スイッチ(config)# dot1x critical eapol
スイッチ(config)# dot1x critical recovery delay 2000
スイッチ(config)# interface gigabitethernet 1/0/1
スイッチ(config-if)# dot1x critical
スイッチ(config-if)# dot1x critical recovery action reinitialize
スイッチ(config-if)# dot1x critical vlan 20
スイッチ(config-if)# end
```

WoL を使用した 802.1x 認証の設定

WoL を使用した 802.1x 認証をイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication control-direction** {**both** | **in**}
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/3`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**authentication control-direction** {**both**	**in**} 例： `スイッチ(config-if)# authentication control-direction both`	ポートで WoL を使用して 802.1x 認証をイネーブルにし、次のキーワードを使用してポートを双方向または単方向に設定します。 • **both**：ポートを双方向に設定します。ポートは、ホストにパケットを送受信できません。デフォルトでは、ポートは双方向です。 • **in**：ポートを単方向に設定します。ポートは、ホストにパケットを送信できますが、受信はできません。
ステップ 4	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show authentication sessions interface** *interface-id* 例： `スイッチ# show authentication sessions interface gigabitethernet2/0/3`	入力を確認します。

	コマンドまたはアクション	目的
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

MAC 認証バイパスの設定

MAC 認証バイパスをイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication port-control auto**
4. **mab** [**eap**]
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/1`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**authentication port-control auto** 例： `スイッチ(config-if)# authentication port-control auto`	ポートでの 802.1x 認証をイネーブルにします。
ステップ 4	**mab** [**eap**] 例：	MAC 認証バイパスをイネーブルにします。 （任意）**eap** キーワードを使用して、認可用に EAP を使用できるようにスイッチを設定します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **mab**	
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

MAC 認証バイパスのユーザ名とパスワードの形式作成

オプションの **mab request format** コマンドを使用して認証サーバによって受け入れられる形式で MAB のユーザ名とパスワードをフ形式作成します。ユーザ名とパスワードは通常、クライアントの MAC アドレスです。認証サーバ設定の中には、ユーザ名と異なるパスワードを必要とするものがあります。

MAC 認証バイパス ユーザ名およびパスワードを形式作成するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **mab request format attribute 1 groupsize**{**1** | **2** | **4** |**12**}[**separator** {**-** | **:** | **.**}{**lowercase** | **uppercase**}]
3. **mab request format attribute2** {**0** | **7**} *text*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mab request format attribute 1 groupsize**{**1**	**2**	**4** |**12**}[**separator** {**-**	**:**	**.**}{**lowercase**	**uppercase**}] 例： スイッチ(config)# **mab request format attribute 1 groupsize 12**	MAB で生成された Access-Request パケットの User-Name 属性内の MAC アドレスの形式を指定します。 1：MAC アドレスの 12 桁の十六進数のユーザー名形式を設定します。 group size：区切り文字の挿入の前に連結する 16 進ニブルの数。有効なグループ サイズは、1、2、4、12 のいずれかである必要があります。

	コマンドまたはアクション	目的
		separator：グループ サイズに従って 16 進ニブルを区切る文字。有効な区切り文字は、ハイフン、コロン、ピリオドのいずれかである必要があります。12 のグループ サイズでは、区切り文字は使用されません。 {lowercase | uppercase}：数字以外の 16 進ニブルを小文字または大文字のどちらにするかを指定します。
ステップ 3	**mab request format attribute2** {**0** | **7**} *text* 例： `スイッチ(config)# mab request format attribute 2 7 A02f44E18B12`	**2**：MAB で生成された Access-Request パケット内の User-Password 属性のカスタム（デフォルト以外の）値を指定します。 **0**：追跡するクリア テキスト パスワードを指定します。 **7**：追跡する暗号化パスワードを指定します。 *text*：User-Password 属性で使用するパスワードを指定します。 （注） 設定情報を電子メールで送信する場合、タイプ 7 のパスワード情報を削除してください。**show tech-support** コマンドは、デフォルトで出力からこの情報を削除します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

802.1x ユーザー ディストリビューションの設定

VLAN グループを設定して、VLAN をそのグループにマッピングするには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **vlan group** *vlan-group-name* **vlan-list** *vlan-list*
3. **end**
4. **no vlan group** *vlan-group-name* **vlan-list** *vlan-list*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vlan group** *vlan-group-name* **vlan-list** *vlan-list* 例： スイッチ(config)# **vlan group eng-dept vlan-list 10**	VLAN グループを設定し、単一の VLAN または VLAN の範囲をそのグループにマッピングします。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 4	**no vlan group** *vlan-group-name* **vlan-list** *vlan-list* 例： スイッチ(config)# **no vlan group eng-dept vlan-list 10**	VLAN グループ コンフィギュレーションまたは VLAN グループ コンフィギュレーションの要素をクリアします。

VLAN グループの設定例

次に、VLAN グループを設定し、そのグループに VLAN をマッピングし、VLAN グループ コンフィギュレーションおよび指定 VLAN とのマッピングを確認する例を示します。

```
スイッチ(config)# vlan group eng-dept vlan-list 10

スイッチ(config)# show vlan group group-name eng-dept
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10

スイッチ(config)# show dot1x vlan-group all
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10
hr-dept                         20
```

次に、VLAN を既存の VLAN グループに追加し、VLAN が追加されたことを確認する例を示します。

```
スイッチ(config)# vlan group eng-dept vlan-list 30
スイッチ(config)# show vlan group eng-dept
Group Name                      Vlans Mapped
--------------                  --------------
eng-dept                        10,30
```

次に、VLAN を VLAN グループから削除する例を示します。

```
スイッチ# no vlan group eng-dept vlan-list 10
```

次に、すべての VLAN が VLAN グループからクリアされたときに、その VLAN グループもクリアされる例を示します。

```
スイッチ(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.

スイッチ(config)# show vlan group group-name eng-dept
```

次の例では、すべての VLAN グループをクリアする方法を示します。

```
スイッチ(config)# no vlan group end-dept vlan-list all
スイッチ(config)# show vlan-group all
```

これらのコマンドの詳細については、『*Cisco IOS Security Command Reference*』を参照してください。

NAC レイヤ 2 802.1x 検証の設定

NAC レイヤ 2 802.1x 検証を設定できます。これは、RADIUS サーバーを使用した 802.1x 認証とも呼ばれます。

NAC レイヤ 2 802.1x 検証を設定するには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **authentication periodic**
6. **authentication timer reauthenticate**
7. **end**
8. **show authentication sessions interface** *interface-id*
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/3`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**switchport mode access** 例： `スイッチ(config-if)# switchport mode access`	RADIUS サーバを設定した場合に限り、ポートをアクセス モードに設定します。
ステップ 4	**authentication event no-response action authorize vlan** *vlan-id* 例： `スイッチ(config-if)# authentication event no-response action authorize vlan 8`	アクティブ VLAN を 802.1x ゲスト VLAN として指定します。指定できる範囲は 1 ～ 4094 です。 内部 VLAN（ルーテッド ポート）、RSPAN VLAN、音声 VLAN を除くあらゆるアクティブ VLAN を 802.1x ゲスト VLAN として設定できます。
ステップ 5	**authentication periodic** 例： `スイッチ(config-if)# authentication periodic`	クライアントの定期的な再認証（デフォルトではディセーブル）をイネーブルにします。
ステップ 6	**authentication timer reauthenticate** 例： `スイッチ(config-if)# authentication timer reauthenticate`	クライアントに対する再認証試行を設定します（1 時間に設定）。 このコマンドがスイッチの動作に影響するのは、定期的な再認証をイネーブルに設定した場合だけです。
ステップ 7	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	**show authentication sessions interface** *interface-id* 例： スイッチ# **show authentication sessions interface gigabitethernet2/0/3**	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

ユーザのログイン制限の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default local**
5. **aaa authentication rejected** *n* **in** *m* **ban** *x*
6. **end**
7. **show aaa local user blocked**
8. **clear aaa local user blocked username** *username*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa new-model** 例： `Device(config)# aaa new-model`	認証、許可、およびアカウンティング（AAA）アクセス コントロール モデルをイネーブルにします。
ステップ 4	**aaa authentication login default local** 例：	デフォルトの認証方法を使用して、認証、許可、およびアカウンティング（AAA）認証を設定します。

	コマンドまたはアクション	目的
	`Device(config)# aaa authentication login default local`	
ステップ 5	**aaa authentication rejected** *n* **in** *m* **ban** *x* 例： `Device(config)# aaa authentication rejected 3 in 20 ban 300`	ユーザによるログインが指定の時間および試行回数以内に成功しなかった場合にユーザをブロックする時間を設定します。 • *n*：ユーザーがログインを試行できる回数を指定します。 • *m*：ユーザーがログインを試行できる時間を秒数で指定します。 • *x*：ログインに成功しなかったユーザーのアクセスを禁止する期間を指定します。
ステップ 6	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 7	**show aaa local user blocked** 例： `Device# show aaa local user blocked`	ブロックされたローカルユーザのリストを表示します。
ステップ 8	**clear aaa local user blocked username** *username* 例： `Device# clear aaa local user blocked username user1`	ブロックされたローカルユーザに関する情報を消去します。

例

次に、**show aaa local user blocked** コマンドの出力例を示します。

```
Device# show aaa local user blocked

       Local-user                  State

       user1                       Watched (till 11:34:42 IST Feb 5 2015)
```

NEAT を使用したオーセンティケータ スイッチの設定

この機能を設定するには、ワイヤリング クローゼット外の 1 つのスイッチがサプリカントとして設定され、オーセンティケータ スイッチに接続されている必要があります。

（注）
- CISP またはNEAT セッションがアクティブなときにラインカードを取り外してシャーシに挿入する場合は、オーセンティケータ スイッチ インターフェイスの設定を明示的にフラッピングすることによって、アクセスモードに復元する必要があります。
- *cisco-av-pairs* は、ISE で *device-traffic-class=switch* として設定されている必要があります。これにより、サプリカントが正常に認証された後でトランクとしてインターフェイスが設定されます。

スイッチをオーセンティケータに設定するには、特権EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **cisp enable**
3. **interface** *interface-id*
4. **switchport mode access**
5. **authentication port-control auto**
6. **dot1x pae authenticator**
7. **spanning-tree portfast**
8. **end**
9. **show running-config interface** *interface-id*
10. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**cisp enable** 例： `スイッチ(config)# cisp enable`	CISP をイネーブルにします。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 2/0/1`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**switchport mode access** 例： `スイッチ(config-if)# switchport mode access`	ポートモードを **access** に設定します。
ステップ 5	**authentication port-control auto** 例： `スイッチ(config-if)# authentication port-control auto`	ポート認証モードを auto に設定します。
ステップ 6	**dot1x pae authenticator** 例： `スイッチ(config-if)# dot1x pae authenticator`	インターフェイスをポート アクセス エンティティ（PAE）オーセンティケータとして設定します。
ステップ 7	**spanning-tree portfast** 例： `スイッチ(config-if)# spanning-tree portfast trunk`	単一ワーク ステーションまたはサーバに接続されたアクセス ポート上で Port Fast をイネーブルにします。
ステップ 8	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 9	**show running-config interface** *interface-id* 例： `スイッチ# show running-config interface gigabitethernet 2/0/1`	設定を確認します。
ステップ 10	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	（注） 変更をコンフィギュレーション ファイルに保存すると、オーセンティケータ インターフェイスがリロード後も引き続きトランク モードになることを意味します。オーセンティケータインターフェイスをアクセスポートとして維持する場合は、コンフィギュレーション ファイルに変更を保存しないでください。

NEAT を使用したサプリカント スイッチの設定

スイッチをサプリカントに設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **cisp enable**
3. **dot1x credential**s *profile*
4. **username** *suppswitch*
5. **password** *password*
6. **dot1x supplicant force-multicast**
7. **interface** *interface-id*
8. **switchport trunk encapsulation dot1q**
9. **switchport mode trunk**
10. **dot1x pae supplicant**
11. **dot1x credentials** *profile-name*
12. **end**
13. **show running-config interface** *interface-id*
14. **copy running-config startup-config**
15. Auto Smartport マクロを使用した NEAT の設定

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**cisp enable** 例：	CISP をイネーブルにします。

	コマンドまたはアクション	目的
	スイッチ(config)# **cisp enable**	
ステップ 3	**dot1x credential**s *profile* 例： スイッチ(config)# **dot1x credentials test**	802.1x クレデンシャル プロファイルを作成します。これは、サプリカントとして設定されるポートに接続する必要があります。
ステップ 4	**username** *suppswitch* 例： スイッチ(config)# **username suppswitch**	ユーザ名を作成します。
ステップ 5	**password** *password* 例： スイッチ(config)# **password myswitch**	新しいユーザ名のパスワードを作成します。
ステップ 6	**dot1x supplicant force-multicast** 例： スイッチ(config)# **dot1x supplicant force-multicast**	ユニキャストまたはマルチキャスト パケットのいずれかを受信した場合にスイッチに強制的にマルチキャスト EAPOL だけを送信させます。 これにより、NEAT がすべてのホスト モードでのサプリカント スイッチで機能できるようにもなります。
ステップ 7	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8	**switchport trunk encapsulation dot1q** 例： スイッチ(config-if)# **switchport trunk encapsulation dot1q**	ポートをトランク モードに設定します。
ステップ 9	**switchport mode trunk** 例： スイッチ(config-if)# **switchport mode trunk**	インターフェイスを VLAN トランク ポートとして設定します。

	コマンドまたはアクション	目的
ステップ 10	**dot1x pae supplicant** 例： `スイッチ(config-if)# dot1x pae supplicant`	インターフェイスをポート アクセス エンティティ（PAE）サプリカントとして設定します。
ステップ 11	**dot1x credentials** *profile-name* 例： `スイッチ(config-if)# dot1x credentials test`	802.1x クレデンシャルプロファイルをインターフェイスに対応付けます。
ステップ 12	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 13	**show running-config interface** *interface-id* 例： `スイッチ# show running-config interface gigabitethernet1/0/1`	設定を確認します。
ステップ 14	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。
ステップ 15	Auto Smartport マクロを使用した NEAT の設定	スイッチ VSA ではなく Auto Smartport ユーザー定義マクロを使用して、オーセンティケータ スイッチを設定することもできます。詳細については、このリリースに対応する『*Auto Smartports Configuration Guide*』を参照してください。

ダウンロード可能 ACL およびリダイレクト URL を使用した 802.1x 認証の設定

（注） スイッチにダウンロードする前に、ダウンロード可能な ACL を ACS で設定する必要があります。

ポートでの認証後、**show ip access-list** 特権 EXEC コマンドを使用して、ポートにダウンロードした ACL を表示できます。

ダウンロード可能な ACL の設定

これらのポリシーは、クライアントが認証され、クライアント IP アドレスが IP デバイス トラッキングテーブルに追加された後で有効になります。その後スイッチがダウンロード可能な ACL をポートに適用します。

特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **ip device tracking**
3. **aaa new-model**
4. **aaa authorization network default local group radius**
5. **radius-server vsa send authentication**
6. **interface** *interface-id*
7. **ip access-group** *acl-id* **in**
8. **show running-config interface** *interface-id*
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip device tracking** 例： `スイッチ(config)# ip device tracking`	IP デバイス トラッキング テーブルを設定します。
ステップ 3	**aaa new-model** 例： `スイッチ(config)# aaa new-model`	AAA をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 4	**aaa authorization network default local group radius** 例： スイッチ(config)# **aaa authorization network default local group radius**	許可の方法をローカルに設定します。認可方式を削除するには、**no aaa authorization network default local group radius** コマンドを使用します。
ステップ 5	**radius-server vsa send authentication** 例： スイッチ(config)# **radius-server vsa send authentication**	RADIUS VSA 送信認証を設定します。
ステップ 6	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet2/0/4**	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 7	**ip access-group** *acl-id* **in** 例： スイッチ(config-if)# **ip access-group default_acl in**	ポートの入力方向のデフォルト ACL を設定します。 （注） *acl-id* はアクセス リストの名前または番号です。
ステップ 8	**show running-config interface** *interface-id* 例： スイッチ(config-if)# **show running-config interface gigabitethernet2/0/4**	設定を確認します。
ステップ 9	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ダウンロード ポリシーの設定

特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**

2. **access-list** *access-list-number* **{ deny | permit } { hostname | any | host } log**
3. **interface** *interface-id*
4. **ip access-group** *acl-id* **in**
5. **exit**
6. **aaa new-model**
7. **aaa authorization network default group radius**
8. **ip device tracking**
9. **ip device tracking probe**[**count** |**interval** |**use-svi**]
10. **radius-server vsa send authentication**
11. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**access-list** *access-list-number* **{ deny | permit } { hostname | any | host } log** 例： `スイッチ(config)# access-list 1 deny any log`	デフォルト ポート ACL を定義します。 access-list-number には、1 ～ 99 または 1300 ～ 1999 の 10 進数を指定します。 条件が一致した場合にアクセスを拒否する場合は **deny** を指定し、許可する場合は **permit** を指定します。 source は、次のようなパケットを送信するネットワークまたはホストの送信元アドレスです。 • **hostname**：ドット付き 10 進表記による 32 ビット長の値。 • **any**：source および source-wildcard の値 0.0.0.0 255.255.255.255 の省略形を意味するキーワード any。source-wildcard 値を入力する必要はありません。 • **host**：source および source-wildcard の値 source 0.0.0.0 の省略形を意味するキーワード host。 （任意）source-wildcard ビットを送信元アドレスに適用します。 （任意）ログを入力して、エントリと一致するパケットに関する情報ロギング メッセージをコンソールに送信します。

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet2/0/2`	インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ip access-group** *acl-id* **in** 例： `スイッチ(config-if)# ip access-group default_acl in`	ポートの入力方向のデフォルトACLを設定します。 （注） acl-id はアクセス リストの名前または番号です。
ステップ 5	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	**aaa new-model** 例： `スイッチ(config)# aaa new-model`	AAA をイネーブルにします。
ステップ 7	**aaa authorization network default group radius** 例： `スイッチ(config)# aaa authorization network default group radius`	許可の方法をローカルに設定します。認可方式を削除するには、**no aaa authorization network default group radius** コマンドを使用します。
ステップ 8	**ip device tracking** 例： `スイッチ(config)# ip device tracking`	IP デバイス トラッキング テーブルをイネーブルにします。 IP デバイス トラッキング テーブルをディセーブルにするには、**no ip device tracking** グローバル コンフィギュレーション コマンドを使用します。
ステップ 9	**ip device tracking probe**[**count** |**interval** |**use-svi**] 例： `スイッチ(config)# ip device tracking probe count`	（任意）IP デバイス トラッキング テーブルを設定します。 • **count** *count*：スイッチが ARP プローブを送信する回数を設定します。指定できる範囲は1～5 です。デフォルトは 3 です。 • **interval** *interval*：スイッチが応答を待ち、ARP プローブを再送信するまでの秒数を設定しま

	コマンドまたはアクション	目的
		す。範囲は 30 ～ 300 秒です。デフォルトは 30 秒です。 • **use-svi**：スイッチ仮想インターフェイス（SVI）IP アドレスを ARP プローブのソースとして使用します。
ステップ 10	**radius-server vsa send authentication** 例： `スイッチ(config)# radius-server vsa send authentication`	ベンダー固有属性を認識し使用するために、ネットワーク アクセス サーバーを設定します。 （注） ダウンロード可能な ACL が機能する必要があります。
ステップ 11	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

VLAN ID ベース MAC 認証の設定

特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **mab request format attribute 32 vlan access-vlan**
3. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mab request format attribute 32 vlan access-vlan** 例： `スイッチ(config)# mab request format attribute 32 vlan access-vlan`	VLAN ID ベース MAC 認証をイネーブルにします。

	コマンドまたはアクション	目的
ステップ 3	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

柔軟な認証順序の設定

下の手順で使用される例は、MAB が IEEE 802.1x 認証（dot1x）の前に試行されるように柔軟な認証の順序設定の順序を変更します。MAB は最初の認証方式として設定されているため、MAB は他のすべての認証方式よりも優先されます。

特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication order** [**dot1x** | **mab**] | {**webauth**}
5. **authentication priority** [**dot1x** | **mab**] | {**webauth**}
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**switchport mode access** 例： スイッチ(config-if)# **switchport mode access**	RADIUS サーバを事前に設定した場合に限り、ポートをアクセス モードに設定します。

	コマンドまたはアクション	目的
ステップ 4	**authentication order** [**dot1x** | **mab**] | {**webauth**} 例： `スイッチ(config-if)# authentication order mab dot1x`	（任意）ポート上で使用される認証方式の順序を設定します。
ステップ 5	**authentication priority** [**dot1x** | **mab**] | {**webauth**} 例： `スイッチ(config-if)# authentication priority mab dot1x`	（任意）認証方式をポート プライオリティ リストに追加します。
ステップ 6	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

Open1x の設定

ポートの許可ステートの手動制御をイネーブルにするには、特権EXECモードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication control-direction** {**both** | **in**}
5. **authentication fallback** *name*
6. **authentication host-mode**[**multi-auth** |**multi-domain** |**multi-host** |**single-host**]
7. **authentication open**
8. **authentication order** [**dot1x** | **mab**] | {**webauth**}
9. **authentication periodic**
10. **authentication port-control** {**auto** | **force-authorized** | **force-un authorized**}
11. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**switchport mode access** 例： スイッチ(config-if)# **switchport mode access**	RADIUS サーバを設定した場合に限り、ポートをアクセス モードに設定します。
ステップ 4	**authentication control-direction** {**both** | **in**} 例： スイッチ(config-if)# **authentication control-direction both**	（任意）ポート制御を単一方向モードまたは双方向モードに設定します。
ステップ 5	**authentication fallback** *name* 例： スイッチ(config-if)# **authentication fallback profile1**	（任意）802.1x 認証をサポートしないクライアント用のフォールバック方法として Web 認証を使用するようポートを設定します。
ステップ 6	**authentication host-mode**[**multi-auth** |**multi-domain** |**multi-host** |**single-host**] 例： スイッチ(config-if)# **authentication host-mode multi-auth**	（任意）ポート上で認証マネージャ モードを設定します。
ステップ 7	**authentication open** 例： スイッチ(config-if)# **authentication open**	（任意）ポート上でオープン アクセスをイネーブルまたはディセーブルにします。
ステップ 8	**authentication order** [**dot1x** | **mab**] | {**webauth**} 例：	（任意）ポート上で使用される認証方式の順序を設定します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **authentication order dot1x webauth**	
ステップ 9	**authentication periodic** 例： スイッチ(config-if)# **authentication periodic**	（任意）ポート上で再認証をイネーブルまたはディセーブルにします。
ステップ 10	**authentication port-control** {**auto** | **force-authorized** | **force-un authorized**} 例： スイッチ(config-if)# **authentication port-control auto**	（任意）ポートの許可ステートの手動制御をイネーブルにします。
ステップ 11	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

ポート上での 802.1x 認証のディセーブル化

802.1x 認証をポートでディセーブルにするには、**no dot1x pae** インターフェイス コンフィギュレーション コマンドを使用します。

ポートで 802.1x 認証をディセーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **no dot1x pae authenticator**
5. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 2/0/1**	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**switchport mode access** 例： スイッチ(config-if)# **switchport mode access**	（任意）RADIUS サーバを設定した場合に限り、ポートをアクセス モードに設定します。
ステップ 4	**no dot1x pae authenticator** 例： スイッチ(config-if)# **no dot1x pae authenticator**	ポートでの 802.1x 認証をディセーブルにします。
ステップ 5	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

802.1x 認証設定のデフォルト値へのリセット

802.1x認証設定をデフォルト値に戻すには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x default**
4. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 2	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/2**	インターフェイス コンフィギュレーション モードを開始し、設定するポートを指定します。
ステップ 3	**dot1x default** 例： スイッチ(config-if)# **dot1x default**	設定可能な 802.1x のパラメータをデフォルト値へ戻します。
ステップ 4	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。

802.1x の統計情報およびステータスのモニターリング

表 *144 :* 特権 *EXEC* 表示コマンド

コマンド	目的
show dot1x all statistics	すべてのポートの 802.1x 統計情報を表示します。
show dot1x interface *interface-id* **statistics**	指定されたポートの 802.1x 統計情報を表示します。
show dot1x all[**count** |**details** |**statistics** |**summary**]	スイッチの 802.1x 管理ステータスおよび動作ステータスを表示します。
show dot1x interface *interface-id*	指定されたポートの 802.1x 管理ステータスおよび動作ステータスを表示します。

表 *145 :* グローバル コンフィギュレーション コマンド

コマンド	目的
no dot1x logging verbose	冗長な 802.1x 認証メッセージをフィルタに掛けます（Cisco IOS Release 12.2(55) SE 以降）

出力フィールドの詳細については、このリリースに対応するコマンドリファレンスを参照してください。

第 69 章

MACsec の暗号化設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

MACsec 暗号化について

この章では、Catalyst スイッチで Media Access Control Security（MACsec）暗号化を設定する方法について説明します。MACsec は 2 台の MACsec 対応デバイス間のパケットの認証と暗号化の IEEE 802.1AE 規格です。また、スイッチは、Cisco TrustSec ネットワーク デバイス アドミッション コントロール（NDAC）および Security Association Protocol（SAP）キー交換を使用して MACsec リンク層スイッチ間セキュリティをサポートします。リンク層セキュリティはスイッチ間のパケット認証とスイッチ間の MACsec 暗号化の両方を含みます（暗号化は任意です）。

（注） MACsec は NPE または LAN ベース イメージを実行しているスイッチではサポートされません。

Cisco TrustSec MACsec リンク層スイッチ間セキュリティは、スイッチ上のすべてのダウンリンク ポートで実行できます。

表 *146 :* スイッチ ポートの *MACsec* サポート

インターフェイス（**Interface**）	接続	MACsec のサポート
他のスイッチに接続されたスイッチポート	スイッチからスイッチへ	Cisco TrustSec NDAC MACsec

Cisco TrustSec と Cisco SAP はスイッチ間のリンクにのみ使用され、PC や IP フォンなどのエンドホストに接続されたスイッチ ポートではサポートされません。Cisco NDAC および SAP は、コンパクトなスイッチがワイヤリング クローゼットの外側にセキュリティを拡張するために使用する、ネットワーク エッジ アクセス トポロジ（NEAT）と相互排他的です。

Media Access Control Security と MACsec Key Agreement

802.1AE で定義された MACsec では、暗号化キー入力のためにアウトオブバンド方式を使用することによって、有線ネットワーク上で MAC レイヤの暗号化を実現します。MACsec Key Agreement（MKA）プロトコルでは、必要なセッション キーを提供し、必要な暗号化キーを管理します。MKA と MACsec は、802.1x 拡張認証プロトコル（EAP-TLS）または事前共有キー（PSK）フレームワークを使用した認証に成功した後に実装されます。

MACsec を使用するスイッチでは、MKA ピアに関連付けられたポリシーに応じて、MACsec フレームまたは非 MACsec フレームを許可します。MACsec フレームは暗号化され、整合性チェック値（ICV）で保護されます。スイッチは MKA ピアからフレームを受信すると、MKA によって提供されたセッション キーを使用してこれらのフレームを暗号化し、正しい ICV を計算します。スイッチはこの ICV をフレーム内の ICV と比較します。一致しない場合は、フレームが破棄されます。また、スイッチは現在のセッション キーを使用して、ICV を暗号化し、セキュアなポート（セキュアな MAC サービスを MKA ピアに提供するために使用されるアクセス ポイント）を介して送信されたフレームに追加します。

MKA プロトコルは、基礎となる MACsec プロトコルで使用される暗号キーを管理します。MKA の基本的な要件は 802.1x-REV で定義されています。MKA プロトコルでは 802.1x を拡張し、相互認証の確認によってピアを検出し、MACsec 秘密キーを共有してピアで交換されるデータを保護できます。

EAP フレームワークでは、新しく定義された EAP-over-LAN（EAPOL）パケットとして MKA を実装します。EAP 認証では、データ交換で両方のパートナーで共有されるマスター セッション キー（MSK）を生成します。EAP セッション ID を入力すると、セキュアな接続アソシエーション キー名（CKN）が生成されます。スイッチは、アップリンクおよびダウンリンクの両

方のオーセンティケータとして機能します。また、ダウンリンクのキー サーバーとして機能します。これによってランダムなセキュア アソシエーション キー（SAK）が生成され、クライアント パートナーに送信されます。クライアントはキー サーバーではなく、単一の MKA エンティティであるキー サーバーとだけ対話できます。キーの派生と生成の後で、スイッチは定期的にトランスポートをパートナーに送信します。デフォルトの間隔は 2 秒間です。

EAPOL プロトコル データ ユニット（PDU）のパケット本体は、MACsec Key Agreement PDU（MKPDU）と呼ばれます。MKA セッションと参加者は、MKA ライフタイム（6 秒間）が経過しても参加者から MKPDU を受信していない場合に削除されます。たとえば、MKA ピアが接続を解除した場合、スイッチ上の参加者は MKA ピアから最後の MKPDU を受信した後、6 秒間が経過するまで MKA の動作を継続します。

（注） MKPDU の整合性チェック値（ICV）インジケータはオプションです。トラフィックが暗号化されている場合、ICV はオプションではありません。

EAPoL 通知は、キー関連情報のタイプの使用を示します。通知は、サプリカントとオーセンティケータの機能を通知するために使用できます。各側の機能に基づいて、キー関連情報の最大公分母を使用できます。

Cisco IOS XE Fuji 16.8.1a よりも前のリリースでは、MKA と SAP で should-secure がサポートされていました。should-secure を有効にすると、ピアが MACsec に設定されている場合はデータトラフィックが暗号化され、それ以外の場合はクリアテキストで送信されます。Cisco IOS XE Fuji 16.8.1a 以降、入力と出力の両方で must-secure のサポートが有効になります。MKA および SAP では、Must-secure がサポートされています。must-secure を有効にすると、EAPoL トラフィックのみが暗号化されません。他のトラフィックは暗号化されます。暗号化されないパケットはドロップされます。

（注） デフォルトでは、Must-secure モードが有効になっています。

MKA ポリシー

インターフェイスで MKA を有効にするには、定義された MKA ポリシーをインターフェイスに適用する必要があります。MKA ポリシーを削除すると、そのインターフェイス上で MKA がディセーブルになります。次のオプションを設定可能です。

- 16 ASCII 文字未満のポリシー名。
- 物理インターフェイスごとの 0 バイト、30 バイト、または 50 バイトの機密保持（暗号化）オフセット。
- 再送保護。許可される順序外のフレームの数によって定義される MACsec ウィンドウ サイズを設定できます。この値は MACsec でセキュリティ アソシエーションをインストールする際に使用されます。値 0 は、フレームが正しい順序で許可されることを意味します。

仮想ポート

仮想ポートは、1 つの物理ポート上の複数のセキュアな接続アソシエーションに使用します。各接続アソシエーション（ペア）は仮想ポートを表します。1つの物理ポートにつき、仮想ポートは最大 2 つです。2 つの仮想ポートのうち、1 つだけをデータ VLAN の一部とすることができます。もう1つは、音声VLANに対してパケットを外部的にタグ付けする必要があります。同じポートで同じ VLAN 内のセキュアなセッションとセキュアでないセッションを同時にホストすることはできません。この制限のため、802.1x マルチ認証モードはサポートされません。

この制限の例外は、マルチホスト モードで最初の MACsec サプリカントが正常に認証され、スイッチに接続されたハブに接続される場合です。ハブに接続された非MACsecホストでは、マルチホストモードであるため、認証なしでトラフィックを送信できます。最初にクライアントが成功した後、他のクライアントでは認証が必要ないため、マルチホストモードの使用は推奨しません。

仮想ポートは、接続アソシエーションの任意のIDを表し、MKAプロトコル外では意味を持ちません。仮想ポートは個々の論理ポート ID に対応します。仮想ポートの有効なポート ID は 0x0002 ～ 0xFFFF です。各仮想ポートは、16 ビットのポート ID に連結された物理インターフェイスの MAC アドレスに基づいて、一意のセキュア チャネル ID（SCI）を受け取ります。

MACsec およびスタッキング

MACsec を実行している（Catalyst 3560cx）スイッチ スタックマスターは、MACsec をサポートしているメンバースイッチ上のポートを示すコンフィギュレーション ファイルを維持します。スタック マスターは、次に示す機能を実行します。

- セキュアなチャネルとセキュアなアソシエーションの作成および削除を処理します。
- スタック メンバーにセキュアなアソシエーション サービス要求を送信します。
- ローカル ポートまたはリモート ポートからのパケット番号とリプレイ ウィンドウ情報を処理し、キー管理プロトコルを通知します。
- オプションがグローバルに設定されたMACsec初期化要求を、スタックに追加される新しいスイッチに送信します。
- ポート単位の設定をメンバー スイッチに送信します。

メンバー スイッチは、次の機能を実行します。

- スタック マスターからの MACsec 初期化要求を処理します。
- スタック マスターから送信された MACsec サービス要求を処理します。
- スタック マスターにローカル ポートに関する情報を送信します。

スタックマスターの切り替えの場合、すべてのセキュアなセッションがダウンし、再確立されます。認証マネージャはセキュアなセッションを認識し、これらのセッションのティアダウンを開始します。

（注） スイッチ間接続に 1G SFP モジュールを使用している場合、MACsec のオーバーヘッドを確実にするため、システム MTU を 1550 バイトに変更します。

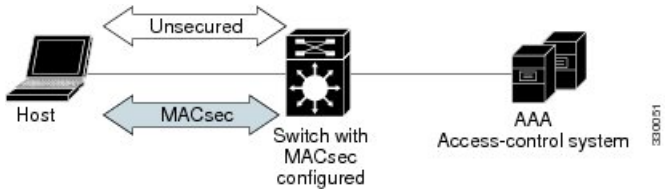
MACsec、MKA、および 802.1x ホスト モード

MACsec と MKA プロトコルは、802.1x シングルホスト モード、、またはマルチドメイン認証（MDA）モードで使用できます。マルチ認証モードはサポートされません。

シングルホスト モード

次の図に、MKA を使用して、MACsec で 1 つの EAP 認証済みセッションをセキュアにする方法を示します。

図 *113*：セキュアなデータ セッションでのシングルホスト モードの *MACsec*



MKA 統計情報

一部の MKA カウンタはグローバルに集約され、その他のカウンタはグローバルとセッション単位の両方で更新されます。また、MKA セッションのステータスに関する情報も取得できます。

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

```
SWitch# show mka statistics
MKA Global Statistics
=====================
MKA Session Totals
Secured..................... 32
Reauthentication Attempts.. 31

Deleted (Secured)........... 1
Keepalive Timeouts......... 0

CA Statistics
Pairwise CAKs Derived...... 32
Pairwise CAK Rekeys........ 31
Group CAKs Generated....... 0
Group CAKs Received........ 0

SA Statistics
SAKs Generated............. 32
SAKs Rekeyed............... 31
SAKs Received.............. 0
SAK Responses Received..... 32


MKPDU Statistics
```

```
MKPDUs Validated & Rx..... 580
"Distributed SAK"..... 0
"Distributed CAK"..... 0
MKPDUs Transmitted..... 597
"Distributed SAK"..... 32
"Distributed CAK"..... 0

MKA Error Counter Totals
========================

Bring-up Failures.................. 0
Reauthentication Failures.......... 0

SAK Failures

SAK Generation.................. 0
Hash Key Generation............. 0
SAK Encryption/Wrap............. 0
SAK Decryption/Unwrap........... 0


CA Failures

Group CAK Generation............ 0
Group CAK Encryption/Wrap....... 0
Group CAK Decryption/Unwrap..... 0
Pairwise CAK Derivation......... 0
CKN Derivation.................. 0
ICK Derivation.................. 0
KEK Derivation.................. 0
Invalid Peer MACsec Capability.. 2

MACsec Failures

Rx SC Creation.................. 0
Tx SC Creation.................. 0
Rx SA Installation.............. 0
Tx SA Installation.............. 0

MKPDU Failures

MKPDU Tx........................ 0
MKPDU Rx Validation............. 0
MKPDU Rx Bad Peer MN............ 0
MKPDU Rx Non-recent Peerlist MN.. 0
```

EAP-TLS を使用した MACsec MKA に関する情報

MACsec MKA はスイッチ間リンクでサポートされます。Extensible Authentication Protocol（EAP-TLS）による IEE 802.1X ポートベース認証を使用して、デバイスのアップリンク ポート間で MACsec MKA を設定できます。EAP-TLS は相互認証を許可し、MSK（マスター セッション キー）を取得します。そのキーから、MKA 操作用の接続アソシエーション キー（CAK）が取得されます。デバイスの証明書は、AAA サーバーへの認証用に、EAP-TLS を使用して伝送されます。

EAP-TLS を使用した MACsec MKA の前提条件

- 認証局（CA）サーバーがネットワークに設定されていることを確認します。
- CA 証明書を生成します。
- Cisco Identity Services Engine（ISE）リリース 2.0 が設定されていることを確認します。

- 両方の参加デバイス（CA サーバーと Cisco Identity Services Engine（ISE））が Network Time Protocol（NTP）を使用して同期されていることを確認します。時間がすべてのデバイスで同期されていないと、証明書は検証されません。
- 802.1x 認証と AAA がデバイスに設定されていることを確認します。

EAP-TLS を使用した MACsec MKA の制限事項

- MKA は、ポート チャネルではサポートされていません。
- Cisco Catalyst 3560-CX スイッチは、EtherChannel での MACSec MKA 設定をサポートしていません。
- MKA は、高可用性とローカル認証ではサポートされていません。
- MKA と EAPTLS は、無差別 PVLAN プライマリポートではサポートされません。
- EAP-TLS を使用して MACsec MKA を設定している間、MACsec セキュアチャネル暗号化カウンタは最初のキー再生成の前に増加しません。

Cisco TrustSec の概要

次の表に、TrustSec がイネーブルになった Cisco スイッチで実装される TrustSec 機能を示します。継続的な TrustSec の General Availability リリースによって、サポートされるスイッチの数および各スイッチでサポートされる TrustSec 機能の数は増加しています。

Cisco TrustSec の機能	説明
802.1AE タギング（MACSec）	IEEE 802.1AE に基づくワイヤレート ホップ単位レイヤ 2 暗号化のプロトコル。 MACSec 対応デバイス間において、パケットは送信デバイスからの出力で暗号化され、受信デバイスへの入力で復号化されます。デバイス内では平文です。 この機能は、TrustSec ハードウェア対応デバイス間だけで利用できます。
エンドポイントアドミッション コントロール（EAC）	EAC は、TrustSec ドメインに接続しているエンドポイント ユーザーまたはデバイスの認証プロセスです。通常、EAC はアクセス レベル スイッチで実行されます。EAC プロセスの認証および許可に成功すると、ユーザーまたはデバイスに対してセキュリティ グループ タグが割り当てられます。現在、EAC は 802.1X、MAC 認証バイパス（MAB）、および Web 認証プロキシ（WebAuth）とすることができます。

Cisco TrustSec の機能	説明
ネットワーク デバイス アドミッション コントロール（NDAC）	NDAC は、TrustSec ドメイン内の各ネットワーク デバイスがピア デバイスのクレデンシャルおよび信頼性を確認できる認証プロセスです。NDAC は、IEEE 802.1X ポート ベースの認証に基づく認証フレームワークを利用し、EAP 方式として EAP-FAST を使用します。NDAC プロセスの認証および許可に成功すると、IEEE 802.1AE 暗号化のセキュリティ アソシエーション プロトコル ネゴシエーションとなります。
セキュリティ アソシエーション プロトコル（SAP）	NDAC 認証のあと、セキュリティ アソシエーション プロトコル（SAP）は、その後の TrustSec ピア間の MACSec リンク暗号化のキーおよび暗号スイートについて、自動的にネゴシエーションを行います。SAP は IEEE 802.11i で定義されます。
セキュリティ グループ タグ（SGT）	SGT は、TrustSec ドメイン内の送信元のセキュリティ分類を示す 16 ビットの単一ラベルです。イーサネット フレームまたは IP パケットに追加されます。
SGT 交換プロトコル（SXP）	Security Group Tag Exchange Protocol（SXP）。SXP を使用すると、TrustSec にハードウェアで対応していないデバイスが Cisco Identity Services Engine（ISE）または Cisco Secure アクセス コントロール システム（ACS）から認証されたユーザーとデバイスの SGT 属性を受信できます。デバイスは、次にセキュリティ グループ アクセス コントロール リスト（SGACL）強制のために、送信元トラフィックをタグ付けする TrustSec にハードウェアで対応しているデバイスに、sourceIP-to-SGT バインディングを転送できます。

リンクの両端で 802.1AE MACsec をサポートしている場合、SAP ネゴシエーションが実行されます。サプリカントとオーセンティケータの間で EAPOL-Key が交換され、暗号スイートのネゴシエーション、セキュリティ パラメータの交換、およびキーの管理が実行されます。これらの作業が正常に完了すると、セキュリティ アソシエーション（SA）が確立します。

ソフトウェア バージョンとライセンスおよびリンク ハードウェア サポートに応じて、SAP ネゴシエーションは次の動作モードの 1 つを使用できます。

- Galois Counter Mode（GCM）：認証と暗号化
- GCM authentication（GMAC）：GCM 認証、暗号化なし

- No Encapsulation：カプセル化なし（クリア テキスト）
- null：カプセル化、認証または暗号化なし

MKA および MACsec の設定

MACsec MKA のデフォルト設定

MACsec はディセーブルです。MKA ポリシーは設定されていません。

MKA ポリシーの設定

手順の概要

1. **configure terminal**
2. **mka policy** *policy name*
3. **confidentiality-offset** オフセット値
4. **replay-protection window-size** *frames*
5. **end**
6. **show mka policy**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**mka policy** *policy name*	MKA ポリシーを指定し、MKA ポリシー コンフィギュレーション モードを開始します。ポリシー名の長さは最大で 16 文字です。
ステップ 3	**confidentiality-offset** オフセット値	各物理インターフェイスに機密性（暗号化）オフセットを設定します。 （注） オフセット値は、0、30、または 50 を指定できます。クライアントで Anyconnect を使用している場合は、オフセット 0 を使用することをお勧めします。
ステップ 4	**replay-protection window-size** *frames*	再送保護をイネーブルにして、ウィンドウ サイズをフレームの数で設定します。範囲は 0 ～ 4294967295 です。デフォルトのウィンドウ サイズは 0 です。

	コマンドまたはアクション	目的
		ウィンドウサイズに 0 を入力することと、 **no replay-protection command** を入力することとは異なります。ウィンドウサイズを 0 に設定すると、厳密なフレーム順序でリプレイ保護が使用されます。**no replay-protection** を入力すると、MACsec 再送保護が無効になります。
ステップ 5	**end**	特権 EXEC モードに戻ります。
ステップ 6	**show mka policy**	入力内容を確認します。

例

次に、MKA ポリシー *relay-policy* を設定する例を示します。

```
Switch(config)# mka policy replay-policy
Switch(config-mka-policy)# confidentiality-offset 0
Switch(config-mka-policy)# replay-protection window-size 300
Switch(config-mka-policy)# end
```

インターフェイスでの MACsec の設定

音声用に 1 つの MACsec セッションとデータ用に 1 つの MACsec セッションが存在するインターフェイスで MACsec を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface***interface-id*
4. **switchport access vlan***vlan-id*
5. **switchport mode access**
6. **macsec**
7. **authentication event linksec fail action authorize vlan** *vlan-id*
8. **authentication host-mode multi-domain**
9. **authentication linksec policy must-secure**
10. **authentication port-control auto**
11. **authentication periodic**
12. **authentication timer reauthenticate**
13. **authentication violation protect**
14. **mka policy** *policy name*
15. **dot1x pae authenticator**
16. **spanning-tree portfast**
17. **end**

18. **show authentication session interface** *interface-id*
19. **show authentication session interface** *interface-id* details
20. **show macsec interface** *interface-id*
21. **show mka sessions**
22. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface***interface-id*	MACsecインターフェイスを指定し、インターフェイスコンフィギュレーションモードを開始します。インターフェイスは物理インターフェイスでなければなりません。
ステップ 4	**switchport access vlan***vlan-id*	このポートのアクセス VLAN を設定します。
ステップ 5	**switchport mode access**	インターフェイスをアクセス ポートとして設定します。
ステップ 6	**macsec**	インターフェイスで 802.1ae MACsec をイネーブルにします。
ステップ 7	**authentication event linksec fail action authorize vlan** *vlan-id*	（任意）認証の試行に失敗した後で、ポート上の制限付き VLAN を許可することによって、ユーザー証明書が認識されない認証リンク セキュリティの問題をスイッチが処理することを指定します。
ステップ 8	**authentication host-mode multi-domain**	ホストと音声デバイスの両方が、802.1x で許可されたポート上で認証されるように、ポート上の認証マネージャ モードを設定します。設定されていない場合、デフォルトのホストモードはシングルです。
ステップ 9	**authentication linksec policy must-secure**	LinkSecセキュリティ ポリシーを設定して、ピアを利用できる場合に、MACsec でセッションをセキュアにします。設定されていない場合、デフォルト値は *should secure* です。

	コマンドまたはアクション	目的
ステップ 10	**authentication port-control auto**	ポート上で 802.1x 認証をイネーブルにします。スイッチとクライアント間の認証交換に基づいてポートが許可ステートまたは無許可ステートに変わります。
ステップ 11	**authentication periodic**	このポートの再認証を有効または無効にします。
ステップ 12	**authentication timer reauthenticate**	1 ～ 65535 の値を入力します。サーバから再認証タイムアウト値を取得します。
ステップ 13	**authentication violation protect**	新しいデバイスがポートに接続された場合、または最大数のデバイスがポートに接続されたあとに新しいデバイスがそのポートに接続された場合に、予期しない着信 MAC アドレスを破棄するようポートを設定します。設定されていない場合、デフォルトではポートをシャット ダウンします。
ステップ 14	**mka policy** *policy name*	既存の MKA プロトコル ポリシーをインターフェイスに適用し、インターフェイス上で MKA をイネーブルにします。（**mka policy** グローバル コンフィギュレーション コマンドを入力して）MKA ポリシーが設定されていない場合、**mka default-policy** インターフェイス コンフィギュレーション コマンドを入力して、MKA のデフォルトのポリシーをインターフェイスに適用する必要があります。
ステップ 15	**dot1x pae authenticator**	ポートを 802.1x ポート アクセス エンティティ（PAE）オーセンティケータとして設定します。
ステップ 16	**spanning-tree portfast**	関連するすべての VLAN 内の特定のインターフェイスで、スパニングツリー Port Fast をイネーブルにします。Port Fast 機能がイネーブルの場合、インターフェイスはブロッキング ステートからフォワーディング ステートに直接移行します。その際に、中間のスパニングツリー ステートは変わりません。
ステップ 17	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 18	**show authentication session interface** *interface-id*	許可されたセッションのセキュリティ ステータスを確認します。
ステップ 19	**show authentication session interface** *interface-id* details	承認されたセッションのセキュリティ ステータスの詳細を確認します。

	コマンドまたはアクション	目的
ステップ 20	**show macsec interface** *interface-id*	インターフェイスの MacSec ステータスを確認します。
ステップ 21	**show mka sessions**	確立された mka セッションを確認します。
ステップ 22	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

PSK を使用した MACsec MKA の設定

手順の概要

1. **configure terminal**
2. **key chain** *key-chain-name* **macsec**
3. **key** *hex-string*
4. **key-string** *{ [0|6|7] pwd-string | pwd-string}*
5. **lifetime local** [*start timestamp {hh::mm::ss | day | month | year}*] [**duration** *seconds* | *end timestamp {hh::mm::ss | day | month | year}*]
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**key chain** *key-chain-name* **macsec**	キー チェーンを設定して、キー チェーン コンフィギュレーション モードを開始します。
ステップ 3	**key** *hex-string*	キーチェーン内の各キーの固有識別子を設定し、キーチェーンのキー コンフィギュレーション モードを開始します。 （注） 128 ビット暗号の場合は、32 文字の 16 進数キー文字列を使用します。
ステップ 4	**key-string** *{ [0|6|7] pwd-string	pwd-string}*	キー文字列のパスワードを設定します。16進数の文字のみを入力する必要があります。

	コマンドまたはアクション	目的
ステップ 5	**lifetime local** [*start timestamp {hh::mm::ss	day	month	year}*] [**duration** *seconds*	*end timestamp {hh::mm::ss	day	month	year}*]	事前共有キーの有効期間を設定します。
ステップ 6	**end**	特権 EXEC モードに戻ります。

例

次に例を示します。

```
Switch(config)# Key chain keychain1 macsec
Switch(config-key-chain)# key 1000
Switch(config-keychain-key)# cryptographic-algorithm gcm-aes-128
Switch(config-keychain-key)# key-string 12345678901234567890123456789012
Switch(config-keychain-key)# lifetime local 12:12:00 July 28 2016 12:19:00
July 28 2016
Switch(config-keychain-key)# end
```

PSK を使用した、インターフェイスでの MACsec MKA の設定

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **macsec network-link**
4. **mka policy** *policy-name*
5. **mka pre-shared-key key-chain** *key-chain name*
6. **macsec replay-protection window-size** *frame number*
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id*	インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**macsec network-link**	インターフェイス上で MACsec をイネーブルにします。
ステップ 4	**mka policy** *policy-name*	MKA ポリシーを設定します。
ステップ 5	**mka pre-shared-key key-chain** *key-chain name*	MKA 事前共有キーのキーチェーン名を設定します。

	コマンドまたはアクション	目的
		（注） MKA 事前共有キーは、物理インターフェイスまたはサブインターフェイスで設定できますが、両方で設定することはできません。
ステップ 6	**macsec replay-protection window-size** *frame number*	リプレイ保護の MACsec ウィンドウ サイズを設定します。
ステップ 7	**end**	特権 EXEC モードに戻ります。

例

次に例を示します。

```
Switch(config)# interface GigabitEthernet 0/0/0
Switch(config-if)# mka policy mka_policy
Switch(config-if)# mka pre-shared-key key-chain key-chain-name
Switch(config-if)# macsec replay-protection window-size 10
Switch(config-if)# end
```

次のタスク

セッションの実行中に MKA PSK が設定されたインターフェイスで MKA ポリシーを変更することは推奨されません。ただし、変更が必要な場合は、次のようにポリシーを再設定する必要があります。

1. **no macsec network-link** コマンドを使用して、各参加ノードの macsec network-link 設定を削除し、既存のセッションを無効にします。
2. **mka policy policy-name** コマンドを使用して、各参加ノードのインターフェイスで MKA ポリシーを設定します。
3. **macsec network-link** コマンドを使用して、各参加ノードで新しいセッションを有効にします。

EAP-TLS を使用した MACsec MKA の設定

ポイントツーポイント リンクで MKA による MACsec を設定するには、次のタスクを実行します。

- 証明書登録の設定
  - キー ペアの生成
  - SCEP 登録の設定

- 証明書の手動設定

- 認証ポリシーの設定
- EAP-TLS プロファイルおよび IEEE 802.1x クレデンシャルの設定
- インターフェイスでの EAP-TLS を使用した MKA MACsec の設定

リモート認証

キー ペアの生成

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto key generate rsa label** *label-name* **general-keys modulus** *size*	署名および暗号化用に RSA キーペアを作成します。 label キーワードを使用すると、各キーペアにラベルを割り当てることもできます。このラベルは、キーペアを使用するトラストポイントによって参照されます。ラベルを割り当てなかった場合、キーペアには <Default-RSA-Key> というラベルが自動的に付けられます。 追加のキーワードを使用しない場合、このコマンドは汎用 RSA キー ペアを 1 つ生成します。係数が指定されていない場合は、デフォルトのキー係数である 1024 が使用されます。その他の係数サイズを指定するには、modulus キーワードを使用します。
ステップ 3	**end**	特権 EXEC モードに戻ります。
ステップ 4	**show authentication session interface** *interface-id*	許可されたセッションのセキュリティ ステータスを確認します。
ステップ 5	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SCEP による登録の設定

Simple Certificate Enrollment Protocol（SCEP）は、HTTP を使用して認証局（CA）または登録局（RA）と通信する、シスコが開発した登録プロトコルです。SCEP は、要求および証明書の送受信用に最も一般的に使用される方式です。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto pki trustpoint** *server name*	トラストポイントおよび設定された名前を宣言して、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 3	**enrollment url** *url name pem*	デバイスが証明書要求を送信する CA の URL を指定します。 URL 内の IPv6 アドレスは括弧で囲む必要があります。たとえば、http:// [2001:DB8:1:1::1]:80 です。 pem キーワードは、証明書要求に Privacy Enhanced Mail（PEM）の境界を追加します。
ステップ 4	**rsakeypair** *label*	証明書に関連付けるキー ペアを指定します。 （注） **rsakeypair** 名は、信頼ポイント名と一致している必要があります。
ステップ 5	**serial-number none**	**none** キーワードは、証明書要求にシリアル番号が含まれないことを指定します。
ステップ 6	**ip-address none**	**none** キーワードは、証明書要求に IP アドレスが含まれないことを指定します。
ステップ 7	**revocation-check crl**	ピアの証明書が取り消されていないことを確認する方法として CRL を指定します。
ステップ 8	**auto-enroll** *percent* **regenerate**	自動登録をイネーブルにします。これにより、クライアントはCAから自動的にロールオーバー証明書を要求できます。 自動登録がイネーブルでない場合、証明書の失効時にクライアントを手動で PKI に再登録する必要があります。 デフォルトでは、デバイスのドメイン ネーム システム（DNS）名だけが証明書に含められます。 現行の証明書の有効期間が指定のパーセンテージに達したときに、新しい証明書が要求されるように指定するには、percent 引数を使用します。

	コマンドまたはアクション	目的
		名前付きのキーがすでに存在する場合でも、証明書の新しいキーを生成するには、regenerate キーワードを使用します。 ロールオーバー中のキー ペアがエクスポート可能な場合、新しいキーペアもエクスポート可能です。次のコメントがトラストポイントコンフィギュレーションに表示され、キー ペアがエクスポート可能かどうかが示されます。「! RSA key pair associated with trustpoint is exportable.」 新しいキー ペアは、セキュリティ上の問題に対処するために生成することを推奨します。
ステップ 9	**crypto pki authenticate** *name*	CA 証明書を取得して、認証します。
ステップ 10	**exit**	グローバル コンフィギュレーション モードを終了します。
ステップ 11	**show crypto pki certificate** *trustpoint name*	信頼ポイントの証明書に関する情報を表示します。

登録の手動設定

CA が SCEP をサポートしない場合、またはルータと CA 間のネットワーク接続が不可能な場合。手動での証明書登録を設定するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**crypto pki trustpoint** *server name*	トラストポイントおよび設定された名前を宣言して、CA トラストポイントコンフィギュレーション モードを開始します。
ステップ 3	**enrollment url** *url name pem*	デバイスが証明書要求を送信する CA の URL を指定します。 URL 内の IPv6 アドレスは括弧で囲む必要があります。たとえば、http:// [2001:DB8:1:1::1]:80 です。 pem キーワードは、証明書要求に Privacy Enhanced Mail（PEM）の境界を追加します。
ステップ 4	**rsakeypair** *label*	証明書に関連付けるキー ペアを指定します。

	コマンドまたはアクション	目的
ステップ 5	**serial-number none**	**none** キーワードは、証明書要求にシリアル番号が含まれないことを指定します。
ステップ 6	**ip-address none**	**none** キーワードは、証明書要求に IP アドレスが含まれないことを指定します。
ステップ 7	**revocation-check crl**	ピアの証明書が取り消されていないことを確認する方法として CRL を指定します。
ステップ 8	**exit**	グローバル コンフィギュレーション モードから抜けます。
ステップ 9	**crypto pki authenticate** *name*	CA 証明書を取得して、認証します。
ステップ 10	**crypto pki enroll** *name*	証明書要求を生成し、証明書サーバーにコピーおよびペーストするために要求を表示します。 プロンプトが表示されたら、登録情報を入力します。たとえば、証明書要求にデバイスの FQDN および IP アドレスを含めるかどうかを指定します。 コンソール端末に対して証明書要求を表示するかについても選択できます。 必要に応じて、Base 64 符号化証明書を PEM ヘッダーを付けて、または付けずに表示します。
ステップ 11	**crypto pki import** *name* **certificate**	許可された証明書を取得するコンソール端末で、TFTP によって証明書をインポートします。 デバイスは、拡張子が「.req」から「.crt」に変更されたことを除いて、要求の送信に使用した同じファイル名を使用して、許可された証明書を TFTP によって取得しようと試みます。用途キー証明書の場合、拡張子「-sign.crt」および「-encr.crt」が使用されます。 デバイスは、受信したファイルを解析して証明書を検証し、証明書をスイッチの内部証明書データベースに挿入します。 （注） 一部の CA は、証明書要求の用途キー情報を無視し、汎用目的の証明書を発行します。ご使用の CA が証明書要求の用途キー情報を無視する場合は、汎用目的の証明書だけをインポートしてください。ルータは、生成される 2 つのキー ペアのいずれも使用しません。

	コマンドまたはアクション	目的
ステップ 12	**exit**	グローバル コンフィギュレーション モードを終了します。
ステップ 13	**show crypto pki certificate** *trustpoint name*	信頼ポイントの証明書に関する情報を表示します。
ステップ 14	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

802.1x 認証の有効化と AAA の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa new-model**	AAA をイネーブルにします。
ステップ 4	**dot1x system-auth-control**	デバイス上で 802.1X を有効にします。
ステップ 5	**radius server** *name*	RADIUS サーバの設定の名前を Protected Access Credential（PAC）のプロビジョニング用に指定し、RADIUS サーバ設定モードを開始します。
ステップ 6	**address** *ip-address* **auth-port** *port-number* **acct-port** *port-number*	RADIUSサーバーのアカウンティングおよび認証パラメータの IPv4 アドレスを設定します。
ステップ 7	**automate-tester username** *username*	RADIUS サーバーの自動テスト機能を有効にします。 このようにすると、デバイスは RADIUS サーバーにテスト認証メッセージを定期的に送信し、サーバーからの RADIUS 応答を待機します。成功メッセージは必須ではありません。認証失敗であっても、サーバーが稼働していることを示しているため問題ありません。
ステップ 8	**key** *string*	デバイスと RADIUS サーバーとの間におけるすべての RADIUS 通信用の認証および暗号キーを指定します。

	コマンドまたはアクション	目的
ステップ 9	**radius-server deadtime** *minutes*	いくつかのサーバーが使用不能になったときの RADIUS サーバーの応答時間を短くし、使用不能になったサーバーがすぐにスキップされるようにします。
ステップ 10	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 11	**aaa group server radius** *group-name*	異なる RADIUS サーバー ホストを別々のリストと方式にグループ化し、サーバー グループ コンフィギュレーション モードを開始します。
ステップ 12	**server** *name*	RADIUS サーバー名を割り当てます。
ステップ 13	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 14	**aaa authentication dot1x default group** *group-name*	IEEE 802.1x 用にデフォルトの認証サーバー グループを設定します。
ステップ 15	**aaa authorization network default group** *group-name*	ネットワーク認証のデフォルト グループを設定します。

EAP-TLS プロファイルと 802.1x クレデンシャルの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**eap profile** *profile-name*	EAP プロファイルを設定し、EAP プロファイル コンフィギュレーション モードを開始します。
ステップ 4	**method tls**	デバイスで EAP-TLS 方式を有効にします。
ステップ 5	**pki-trustpoint** *name*	デフォルトの PKI トラストポイントを設定します。
ステップ 6	**exit**	グローバル コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
ステップ 7	**dot1x credentials** *profile-name*	802.1x クレデンシャルプロファイルを設定し、dot1x クレデンシャル コンフィギュレーション モードを開始します。
ステップ 8	**username** *username*	認証ユーザー ID を設定します。
ステップ 9	**pki-trustpoint** *name*	デフォルトの PKI トラストポイントを設定します。
ステップ 10	**end**	特権 EXEC モードに戻ります。

インターフェイスでの 802.1x MACsec MKA 設定の適用

EAP-TLS を使用して MACsec MKA をインターフェイスに適用するには、次のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id*	MACsec インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。インターフェイスは物理インターフェイスでなければなりません。
ステップ 3	**macsec network-link**	インターフェイス上で MACsec をイネーブルにします。
ステップ 4	**authentication periodic**	このポートの再認証をイネーブルにします。
ステップ 5	**authentication timer reauthenticate interval**	再認証間隔を設定します。
ステップ 6	**access-session host-mode multi-domain**	ホストにインターフェイスへのアクセスを許可します。
ステップ 7	**access-session closed**	インターフェイスへの事前認証アクセスを防止します。
ステップ 8	**access-session port-control auto**	ポートの認可状態を設定します。
ステップ 9	**dot1x pae both**	ポートを 802.1X ポート アクセス エンティティ（PAE）のサプリカントおよびオーセンティケータとして設定します。
ステップ 10	**dot1x credentials profile**	802.1x クレデンシャルプロファイルをインターフェイスに割り当てます。

	コマンドまたはアクション	目的
ステップ 11	**dot1x supplicant eap profile** *name*	EAP-TLS プロファイルをインターフェイスに割り当てます。
ステップ 12	**service-policy type control subscriber** *control-policy name*	インターフェイスに加入者制御ポリシーを適用します。
ステップ 13	**exit**	特権 EXEC モードに戻ります。
ステップ 14	**show macsec interface**	インターフェイスの MACsec の詳細を表示します。
ステップ 15	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ローカル認証

ローカル認証を使用した EAP クレデンシャルの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa new-model**	AAA をイネーブルにします。
ステップ 4	**aaa local authentication default authorization default**	デフォルトのローカル認証およびデフォルトのローカル認証方法を設定します。
ステップ 5	**aaa authentication dot1x default local**	IEEE 802.1x 用にデフォルトのローカル ユーザー名認証リストを設定します。
ステップ 6	**aaa authorization network default local**	ローカル ユーザーの認可方式リストを設定します。
ステップ 7	**aaa authorization credential-download default local**	ローカル クレデンシャルの使用に関する認可方式リストを設定します。
ステップ 8	**exit**	特権 EXEC モードに戻ります。

ローカル EAP-TLS 認証と認証プロファイルの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**aaa new-model**	AAA をイネーブルにします。
ステップ 4	**dot1x credentials** *profile-name*	dot1x クレデンシャル プロファイルを設定し、dot1x クレデンシャル コンフィギュレーション モードを開始します。
ステップ 5	**username** *name* **password** *password*	認証のユーザー ID およびパスワードを設定します。
ステップ 6	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 7	**aaa attribute list** *list-name*	（任意）AAA 属性リスト定義を設定し、属性リスト コンフィギュレーション モードを開始します。
ステップ 8	**aaa attribute type linksec-policy must-secure**	（任意）AAA 属性タイプを指定します。
ステップ 9	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 10	**username** *name* **aaa attribute list** *name*	（任意）ユーザー ID に AAA 属性リストを指定します。
ステップ 11	**end**	特権 EXEC モードに戻ります。

SCEP による登録の設定

Simple Certificate Enrollment Protocol（SCEP）は、HTTP を使用して認証局（CA）または登録局（RA）と通信する、シスコが開発した登録プロトコルです。SCEP は、要求および証明書の送受信用に最も一般的に使用される方式です。

手順

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

	コマンドまたはアクション	目的
		• パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto pki trustpoint** *server name*	トラストポイントおよび設定された名前を宣言して、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 4	**enrollment url** *url name pem*	デバイスが証明書要求を送信する CA の URL を指定します。 URL 内の IPv6 アドレスは括弧で囲む必要があります。たとえば、http:// [2001:DB8:1:1::1]:80 です。 pem キーワードは、証明書要求に Privacy Enhanced Mail（PEM）の境界を追加します。
ステップ 5	**rsakeypair** *label*	証明書に関連付けるキー ペアを指定します。 （注） **rsakeypair** 名は、信頼ポイント名と一致している必要があります。
ステップ 6	**serial-number none**	**none** キーワードは、証明書要求にシリアル番号が含まれないことを指定します。
ステップ 7	**ip-address none**	**none** キーワードは、証明書要求に IP アドレスが含まれないことを指定します。
ステップ 8	**revocation-check crl**	ピアの証明書が取り消されていないことを確認する方法として CRL を指定します。
ステップ 9	**auto-enroll** *percent* **regenerate**	自動登録をイネーブルにします。これにより、クライアントはCAから自動的にロールオーバー証明書を要求できます。 自動登録がイネーブルでない場合、証明書の失効時にクライアントを手動で PKI に再登録する必要があります。 デフォルトでは、デバイスのドメイン ネーム システム（DNS）名だけが証明書に含められます。 現行の証明書の有効期間が指定のパーセンテージに達したときに、新しい証明書が要求されるように指定するには、percent 引数を使用します。

	コマンドまたはアクション	目的
		名前付きのキーがすでに存在する場合でも、証明書の新しいキーを生成するには、regenerate キーワードを使用します。 ロールオーバー中のキー ペアがエクスポート可能な場合、新しいキーペアもエクスポート可能です。次のコメントがトラストポイントコンフィギュレーションに表示され、キー ペアがエクスポート可能かどうかが示されます。「! RSA key pair associated with trustpoint is exportable.」 新しいキー ペアは、セキュリティ上の問題に対処するために生成することを推奨します。
ステップ 10	**crypto pki authenticate** *name*	CA 証明書を取得して、認証します。
ステップ 11	**exit**	グローバル コンフィギュレーション モードを終了します。
ステップ 12	**show crypto pki certificate** *trustpoint name*	信頼ポイントの証明書に関する情報を表示します。

登録の手動設定

CA が SCEP をサポートしない場合、またはルータと CA 間のネットワーク接続が不可能な場合。手動での証明書登録を設定するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**crypto pki trustpoint** *server name*	トラストポイントおよび設定された名前を宣言して、CA トラストポイント コンフィギュレーション モードを開始します。
ステップ 4	**enrollment url** *url name pem*	デバイスが証明書要求を送信する CA の URL を指定します。 URL 内の IPv6 アドレスは括弧で囲む必要があります。たとえば、http:// [2001:DB8:1:1::1]:80 です。

	コマンドまたはアクション	目的
		pem キーワードは、証明書要求に Privacy Enhanced Mail（PEM）の境界を追加します。
ステップ 5	**rsakeypair** *label*	証明書に関連付けるキー ペアを指定します。
ステップ 6	**serial-number none**	**none** キーワードは、証明書要求にシリアル番号が含まれないことを指定します。
ステップ 7	**ip-address none**	**none** キーワードは、証明書要求に IP アドレスが含まれないことを指定します。
ステップ 8	**revocation-check crl**	ピアの証明書が取り消されていないことを確認する方法として CRL を指定します。
ステップ 9	**exit**	グローバル コンフィギュレーション モードから抜けます。
ステップ 10	**crypto pki authenticate** *name*	CA 証明書を取得して、認証します。
ステップ 11	**crypto pki enroll** *name*	証明書要求を生成し、証明書サーバーにコピーおよびペーストするために要求を表示します。 プロンプトが表示されたら、登録情報を入力します。たとえば、証明書要求にデバイスの FQDN および IP アドレスを含めるかどうかを指定します。 コンソール端末に対して証明書要求を表示するかについても選択できます。 必要に応じて、Base 64 符号化証明書を PEM ヘッダーを付けて、または付けずに表示します。
ステップ 12	**crypto pki import** *name* **certificate**	許可された証明書を取得するコンソール端末で、TFTP によって証明書をインポートします。 デバイスは、拡張子が「.req」から「.crt」に変更されたことを除いて、要求の送信に使用した同じファイル名を使用して、許可された証明書を TFTP によって取得しようと試みます。用途キー証明書の場合、拡張子「-sign.crt」および「-encr.crt」が使用されます。 デバイスは、受信したファイルを解析して証明書を検証し、証明書をスイッチの内部証明書データベースに挿入します。

	コマンドまたはアクション	目的
		（注） 一部の CA は、証明書要求の用途キー情報を無視し、汎用目的の証明書を発行します。ご使用の CA が証明書要求の用途キー情報を無視する場合は、汎用目的の証明書だけをインポートしてください。ルータは、生成される 2 つのキー ペアのいずれも使用しません。
ステップ 13	**exit**	グローバル コンフィギュレーション モードから抜けます。
ステップ 14	**show crypto pki certificate** *trustpoint name*	信頼ポイントの証明書に関する情報を表示します。
ステップ 15	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

EAP-TLS プロファイルと 802.1x クレデンシャルの設定

手順

	コマンドまたはアクション	目的
ステップ 1	**enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**eap profile** *profile-name*	EAP プロファイルを設定し、EAP プロファイル コンフィギュレーション モードを開始します。
ステップ 4	**method tls**	デバイスで EAP-TLS 方式を有効にします。
ステップ 5	**pki-trustpoint** *name*	デフォルトの PKI トラストポイントを設定します。
ステップ 6	**exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 7	**dot1x credentials** *profile-name*	802.1x クレデンシャル プロファイルを設定し、dot1x クレデンシャル コンフィギュレーション モードを開始します。
ステップ 8	**username** *username*	認証ユーザー ID を設定します。
ステップ 9	**pki-trustpoint** *name*	デフォルトの PKI トラストポイントを設定します。

	コマンドまたはアクション	目的
ステップ 10	**end**	特権 EXEC モードに戻ります。

インターフェイスでの 802.1x MKA MACsec 設定の適用

EAP-TLS を使用して MKA MACsec をインターフェイスに適用するには、次のタスクを実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id*	MACsecインターフェイスを指定し、インターフェイスコンフィギュレーションモードを開始します。インターフェイスは物理インターフェイスでなければなりません。
ステップ 4	**macsec**	インターフェイス上で MACsec をイネーブルにします。
ステップ 5	**authentication periodic**	このポートの再認証をイネーブルにします。
ステップ 6	**authentication timer reauthenticate interval**	再認証間隔を設定します。
ステップ 7	**access-session host-mode multi-domain**	ホストにインターフェイスへのアクセスを許可します。
ステップ 8	**access-session closed**	インターフェイスへの事前認証アクセスを防止します。
ステップ 9	**access-session port-control auto**	ポートの認可状態を設定します。
ステップ 10	**dot1x pae both**	ポートを 802.1X ポート アクセス エンティティ（PAE）のサプリカントおよびオーセンティケータとして設定します。
ステップ 11	**dot1x credentials profile**	802.1x クレデンシャル プロファイルをインターフェイスに割り当てます。
ステップ 12	**dot1x authenticator eap profile** *name*	EAP-TLS オーセンティケータ プロファイルをインターフェイスに割り当てます。

	コマンドまたはアクション	目的
ステップ 13	**dot1x supplicant eap profile** *name*	EAP-TLSサプリカントプロファイルをインターフェイスに割り当てます。
ステップ 14	**service-policy type control subscriber** *control-policy name*	インターフェイスに加入者制御ポリシーを適用します。
ステップ 15	**exit**	特権 EXEC モードに戻ります。
ステップ 16	**show macsec interface**	インターフェイスのMACsecの詳細を表示します。
ステップ 17	**copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

EAP-TLS を使用した MACsec MKA の確認

EAP-TLS を使用して MACsec MKA の設定を確認するには、次のように **show** コマンドを使用します。以下に、**show** コマンドの出力例を示します。

show mka sessions コマンドは、アクティブな MACsec Key Agreement（MKA）プロトコルのセッションの概要を表示します。

```
Device# show mka sessions

  Total MKA Sessions....... 1
      Secured Sessions... 1
      Pending Sessions... 0

===============================================================================================
Interface      Local-TxSCI         Policy-Name      Inherited         Key-Server

Port-ID        Peer-RxSCI          MACsec-Peers     Status            CKN

===============================================================================================
Te0/1/3        74a2.e625.4413/0013 *DEFAULT POLICY* NO                YES

19             74a2.e625.4c22/0012 1                Secured
10000000000000000000000000000000000000000000000000000000000000000000
```

show macsec status interface *interface-id* は、指定されたインターフェイスの MACsec ステータス情報を表示します。

```
Device# show macsec status interface te0/1/2

  Capabilities:
  Ciphers Supported:        GCM-AES-128 GCM-AES-256
  Cipher:                   GCM-AES-128
  Confidentiality Offset:   0
  Replay Window:            64
  Delay Protect Enable:     FALSE
  Access Control:           must-secure
```

```
Transmit SC:
  SCI:                      74A2E6254C220012
  Transmitting:             TRUE
Transmit SA:
  Next PN:                  412
  Delay Protect AN/nextPN:  99/0

Receive SC:
  SCI:                      74A2E62544130013
  Receiving:                TRUE
Receive SA:
  Next PN:                  64
  AN:                       0
  Delay Protect AN/LPN:     0/0
```

show access-session interface *interface-id* **details** は、指定されたインターフェイスのアクセス セッションに関する詳細情報を表示します。

```
Device# show access-session interface te1/0/1 details

   Interface:  TenGigabitEthernet1/0/1
             IIF-ID:  0x17298FCD
         MAC Address:  f8a5.c592.13e4
        IPv6 Address:  Unknown
        IPv4 Address:  Unknown
           User-Name:  DOT1XCRED
              Status:  Authorized
              Domain:  DATA
      Oper host mode:  multi-host
    Oper control dir:  both
     Session timeout:  N/A
   Common Session ID:  000000000000000BB72E8AFA
     Acct Session ID:  Unknown
              Handle:  0xc3000001
      Current Policy:  MUSTS_1


Local Policies:
 Security Policy:  Must Secure
      Security Status:  Link Secured

Server Policies:


Method status list:
       Method           State
     dot1xSup           Authc Success
        dot1x           Authc Success
```

Cisco TrustSec MACsec の設定

スイッチの Cisco TrustSec クレデンシャルの設定

Cisco TrustSec 機能をイネーブルにするには、他の TrustSec 設定で使用するスイッチで Cisco TrustSec クレデンシャルを作成する必要があります。Cisco TrustSec クレデンシャルを設定するには、特権 EXEC モードで次の手順を行います。

手順の概要

1. **cts credentials id** *device-id* **password** *cts-password*
2. **show cts credentials**
3. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**cts credentials id** *device-id* **password** *cts-password* 例： `Switch# cts credentials id trustsec password mypassword`	EAP-FAST を使用して他の Cisco TrustSec デバイスで認証するときにこのスイッチが使用する Cisco TrustSec クレデンシャルを指定します。 • **id** *device-id*：スイッチの Cisco TrustSec デバイス ID を指定します。device-id 引数は、最大 32 文字で大文字と小文字を区別します。 • **password** *cts-password*：デバイスの Cisco TrustSec パスワードを指定します。
ステップ 2	**show cts credentials** 例： `Switch# show cts credentials`	（任意）スイッチで設定された Cisco TrustSec クレデンシャルを表示します。
ステップ 3	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

例

Cisco TrustSec クレデンシャルを削除するには、**clear cts credentials** 特権 EXEC コマンドを入力します。

次に、Cisco TrustSec クレデンシャルを作成する例を示します。

```
Switch# cts credentials id trustsec password mypassword
CTS device ID and password have been inserted in the local keystore. Please make
sure that the same ID and password are configured in the server database.

Switch# show cts credentials
CTS password is defined in keystore, device-id = trustsec
```

次のタスク

Cisco TrustSec MACsec 認証を設定する前に、Cisco TrustSec シードおよび非シード デバイスを設定する必要があります。802.1x モードでは、アクセス コントロール システム（ACS）に最も近い少なくとも 1 台のシード デバイスを設定する必要があります。『Cisco TrustSec Configuration Guide』の次のセクションを参照してください。
http://www.cisco.com/en/US/docs/switches/lan/trustsec/configuration/guide/ident-conn_config.html

802.1X モードでの Cisco TrustSec スイッチ間のリンク セキュリティの設定

始める前に

別の Cisco TrustSec デバイスに接続されているインターフェイス上で Cisco TrustSec リンク層スイッチ間セキュリティをイネーブルにします。インターフェイス上で 802.1X モードの Cisco TrustSec を設定する場合は、次の注意事項に従ってください。

- 802.1x モードを使用するには、各デバイスでグローバルに 802.1x をイネーブルにする必要があります。802.1x の詳細については、「IEEE 802.1x ポートベースの認証の設定」の章を参照してください。
- SAP 動作モードとして GCM を選択すると、シスコの MACsec 暗号化ソフトウェア ライセンスが必要です。MACsec は、Catalyst 3560cx の汎用 IP Base ライセンスと IP サービス ライセンスでサポートされます。これは NPE ライセンスまたは LAN ベース サービス イメージではサポートされません。

  必要なライセンスなしで GCM を選択した場合、インターフェイスはリンク ダウン状態になります。

特権 EXEC モードから 802.1x で Cisco TrustSec のスイッチ間のリンク層セキュリティを設定する手順は、次のとおりです。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **cts dot1x**
4. **sap mode-list***mode1* [*mode2* [*mode3* [*mode4*]]]
5. **no propagate sgt**
6. **exit**
7. **end**

8. **show cts interface** [*interface-id* | **brief** | **summary**]
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Switch(config)# interface tengigabitethernet 1/1/2`	（注） インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**cts dot1x** 例： `Switch(config-if)# cts dot1x`	インターフェイスを、NDAC認証を実行するように設定します。
ステップ 4	**sap mode-list***mode1* [*mode2* [*mode3* [*mode4*]]] 例： `Switch(config-if-cts-dot1x)# sap mode-list gcm-encrypt null no-encap`	（任意）インターフェイスにSAP動作モードを設定します。インターフェイスは相互に受け入れ可能なモード用のピアとネゴシエートします。優先する順序で許容されるモードを入力します。 *mode* の選択肢は次のとおりです。 • **gcm-encrypt**：認証および暗号化 （注） ソフトウェア ライセンスが MACsec 暗号化をサポートする場合、MACsec の認証と暗号化にこのモードを選択します。 • **gmac**：認証、暗号化なし • **no-encap**：カプセル化なし • **null**：カプセル化、認証または暗号化なし （注） インターフェイスでデータ リンク暗号化を使用できない場合は、デフォルトおよび唯一使用可能なSAP動作モードは no-encap です。SGT はサポートされません。 （注） CLI ヘルプには表示されますが、timer reauthentication および propagate sgt キーワードはサポートされません。

	コマンドまたはアクション	目的
ステップ 5	**no propagate sgt** 例： `Switch(config-if-cts-dot1x)# no propagate sgt`	スイッチ（Catalyst 3560cx）は SGT のタギングをサポートしていません。このコマンドは、CTS リンクでの SGT タグの伝達を無効にします。トラフィックが CTS リンクを適切に流れるには、ピア スイッチでも「no propagate sgt」が設定されていることが必須です。
ステップ 6	**exit** 例： `Switch(config-if-cts-dot1x)# exit`	Cisco TrustSec 802.1x インターフェイス コンフィギュレーション モードを終了します。
ステップ 7	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show cts interface** [*interface-id* | **brief** I**summary**]	（任意）TrustSec 関連のインターフェイス特性を表示して、設定を確認します。
ステップ 9	**copy running-config startup-config** 例： `Switch# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

例

次に、優先 SAP モードとして GCM を使用してインターフェイス上で 802.1X モードで Cisco TrustSec 認証をイネーブルにする例を示します。

```
Switch# configure terminal
Switch(config)# interface tengigabitethernet 1/1/2
Switch(config-if)# cts dot1x
Switch(config-if-cts-dot1x)# sap mode-list gcm-encrypt null no-encap
Switch(config-if-cts-dot1x)# no propagate sgt
Switch(config-if-cts-dot1x)# exit
Switch(config-if)# end
```

手動モードでの Cisco TrustSec スイッチ間リンク セキュリティの設定

始める前に

インターフェイスの Cisco TrustSec を手動で設定する場合は、次のような使用上の注意事項、および制約事項を考慮してください。

- SAP パラメータが定義されていない場合、Cisco TrustSec カプセル化または暗号化は行われません。

- SAP 動作モードとして GCM を選択すると、シスコの MACsec 暗号化ソフトウェア ライセンスが必要です。必要なライセンスなしで GCM を選択した場合、インターフェイスはリンク ダウン状態になります。
- これらの保護レベルは、SAP の Pairwise Master Key（sap pmk）を設定する場合にサポートされます。
  - SAP が設定されていない：保護は行われません。
  - **sap mode-list gcm-encrypt gmac no-encap**：保護が望ましいが必須ではない。
  - **sap mode-list gcm-encrypt gmac**：機密性が推奨され、整合性が必須。保護はサプリカントの設定に応じてサプリカントによって選択されます。
  - **sap mode-list gmac**：整合性のみ。
  - **sap mode-list gcm-encrypt**：機密性が必須。
  - **sap mode-list gmac gcm-encrypt**：整合性が必須であり推奨される。機密性は任意。

別の Cisco TrustSec デバイスへのインターフェイスで Cisco TrustSec を手動で設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **cts manual**
4. **sap pmk** *key* [**mode-list** *mode1* [*mode2* [*mode3* [*mode4*]]]]
5. **no propagate sgt**
6. **exit**
7. **end**
8. **show cts interface** [*interface-id* | **brief** | **summary**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Switch(config)# interface tengigabitethernet 1/1/2`	（注） インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**cts manual** 例：	Cisco TrustSec 手動コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`Switch(config-if)# cts manual`	
ステップ 4	**sap pmk** *key* [**mode-list** *mode1* [*mode2* [*mode3* [*mode4*]]]] 例： `Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap`	（任意）SAP の Pairwise Master Key（PMK）と動作モードを設定します。Cisco TrustSec の手動モードでは、SAP はデフォルトでディセーブルになっています。 • *key*：文字数が偶数個で最大 32 文字の 16 進値。 SAP 動作モードのオプションは次のとおりです。 • **gcm-encrypt**：認証および暗号化 （注） ソフトウェア ライセンスが MACsec 暗号化をサポートする場合、MACsec の認証と暗号化にこのモードを選択します。 • **gmac**：認証、暗号化なし • **no-encap**：カプセル化なし • **null**：カプセル化、認証または暗号化なし （注） インターフェイスでデータ リンク暗号化を使用できない場合は、デフォルトおよび唯一使用可能な SAP 動作モードは **no-encap** です。SGT はサポートされません。
ステップ 5	**no propagate sgt** 例： `Switch(config-if-cts-manual)# no propagate sgt`	ピアが SGT を処理できない場合、このコマンドの **no** 形式を使用します。**no propagate sgt** コマンドを使用すると、インターフェイスからピアに SGT が送信されなくなります。
ステップ 6	**exit** 例： `Switch(config-if-cts-manual)# exit`	Cisco TrustSec 802.1x インターフェイス コンフィギュレーション モードを終了します。
ステップ 7	**end** 例： `Switch(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show cts interface** [*interface-id* | **brief** | **summary**]	（任意）TrustSec 関連のインターフェイス特性を表示して、設定を確認します。

例

次に、インターフェイスにCisco TrustSec認証を手動モードで設定する例を示します。

```
Switch# configure terminal
Switch(config)# interface tengigabitethernet 1/1/2
Switch(config-if)# cts manual
Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap
Switch(config-if-cts-manual)# no propagate sgt
Switch(config-if-cts-manual)# exit
Switch(config-if)# end
```

MACsec 暗号化の設定例

例：インターフェイスでの MACsec の設定

インターフェイスでの MACsec の設定

```
スイッチ(config)# interface GigabitEthernet1/0/25
スイッチ(config-if)# switchport access vlan 10
スイッチ(config-if)# switchport mode access
スイッチ(config-if)# macsec
スイッチ(config-if)# authentication event linksec fail action authorize vlan
 2
スイッチ(config-if)# authentication host-mode multi-domain
スイッチ(config-if)# authentication linksec policy must-secure
スイッチ(config-if)# authentication port-control auto
スイッチ(config-if)# authentication periodic
スイッチ(config-if)# authentication timer reauthenticate
スイッチ(config-if)# authentication violation protect
スイッチ(config-if)# mka policy replay-policy
スイッチ(config-if)# dot1x pae authenticator
スイッチ(config-if)# spanning-tree portfast
スイッチ(config-if)# end
スイッチ# show authentication session interface gigabitethernet1/0/5

Interface MAC Address Method Domain Status Fg Session ID
---------------------------------------------------------------------------
Gi1/0/5 88f0.7788.9205 dot1x VOICE Auth 1E0000010000001300030B0F
Gi1/0/5 000c.2923.6ff1 dot1x DATA  Auth 1E0000010000001400030D80


Key to Session Events Blocked Status Flags:

A - Applying Policy (multi-line status for details)
D - Awaiting Deletion
F - Final Removal in progress
I - Awaiting IIF ID allocation
N - Waiting for AAA to come up
P - Pushed Session
R - Removing User Profile (multi-line status for details)
U - Applying User Profile (multi-line status for details)
```

```
X - Unknown Blocker

Runnable methods list:
Handle Priority Name
7 5 dot1x
21 10 mab

19 15 webauth
```

スイッチ# **show authentication session interface gigabitethernet1/0/5 details**

```
Interface: GigabitEthernet1/0/5
MAC Address: 88f0.7788.9205
IPv6 Address: Unknown
IPv4 Address: Unknown
User-Name: CP-9971-SEP88F077889205
Status: Authorized
Domain: VOICE
Oper host mode: multi-domain
Oper control dir: both
Session timeout: N/A
Common Session ID: 1E000001000000001300030B0F
Acct Session ID: Unknown
Handle: 0xC0000006
Current Policy: POLICY_Gi1/0/5

Local Policies:
Service Template: DEFAULT_LINKSEC_POLICY_SHOULD_SECURE (priority 150)
Security Policy: Should Secure
Security Status: Link Unsecure

Server Policies:

Method status list:

Method State

dot1x Authc Success

----------------------------------------
Interface: GigabitEthernet1/0/5
MAC Address: 000c.2923.6ff1
IPv6 Address: Unknown
IPv4 Address: 172.30.30.50
User-Name: dataMustSecure
Status: Authorized
Domain: DATA
Oper host mode: multi-domain
Oper control dir: both
Session timeout: N/A
Common Session ID: 1E000001000000001400030D80
Acct Session ID: Unknown
Handle: 0x22000007
Current Policy: POLICY_Gi1/0/5

Local Policies:
Service Template: DEFAULT_LINKSEC_POLICY_SHOULD_SECURE (priority 150)
Security Policy: Should Secure

Security Status: Link Secured

Server Policies:

Method status list:
```

```
Method State

dot1x Authc Success

スイッチ#
スイッチ# show  macsec interface gigabitethernet1/0/5
MACsec is enabled
Replay protect : enabled
Replay window : 0
Include SCI : yes
Use ES Enable : no
Use SCB Enable : no
Admin Pt2Pt MAC : forceTrue(1)
Pt2Pt MAC Operational : no
Cipher : GCM-AES-128
Confidentiality Offset : 0

Capabilities
Identifier :
Name :
ICV length : 16
Data length change supported: yes
Max. Rx SA : 8
Max. Tx SA : 8
Max. Rx SC : 4
Max. Tx SC : 4
Validate Frames : strict
PN threshold notification support : Yes
Ciphers supported : GCM-AES-128

Transmit Secure Channels
SCI : 547C69B687850002
SC state : inUse(1)
Elapsed time : 16:36:44
Start time : 7w0d
Current AN: 0
Previous AN: -
Next PN: 0
SA State: inUse(1)
Confidentiality : no
SAK Unchanged : no
SA Create time : 00:09:21
SA Start time : 7w0d
SC Statistics
Auth-only Pkts : 0
Auth-only Bytes : 0
Encrypt Pkts : 52960
Encrypt Bytes : 0
SA Statistics
Auth-only Pkts : 0
Encrypt Pkts : 52960

Port Statistics

Receive Secure Channels
SCI : 000C29236FF10000
SC state : inUse(1)
Elapsed time : 16:36:44
Start time : 7w0d
Current AN: 0
Previous AN: -
Next PN: 0
RX SA Count: 0
SA State: inUse(1)
```

```
SAK Unchanged : no
SA Create time : 00:09:19
SA Start time : 7w0d
SC Statistics
Notvalid pkts 0
Invalid pkts 0
Valid pkts 9914
Valid bytes 0
Late pkts 0
Uncheck pkts 0
Delay pkts 0
UnusedSA pkts 0
NousingSA pkts 0
Decrypt bytes 0
SA Statistics
Notvalid pkts 0
Invalid pkts 0
Valid pkts 9914
UnusedSA pkts 0
NousingSA pkts 0

Port Statistics

Switch#
```

EAP-TLS を使用した MACsec MKA の設定例

例:：証明書の登録

```
Configure Crypto PKI Trustpoint:
  crypto pki trustpoint POLESTAR-IOS-CA
   enrollment terminal
   subject-name CN=ASR1000x1@polestar.com, C=IN, ST=KA, OU=ENG,O=Polestar
   revocation-check none
   rsakeypair mkaioscarsa
   storage nvram:
  !
Manual Installation of Root CA certificate:
 crypto pki authenticate POLESTAR-IOS-CA
```

例：802.1x 認証の有効化と AAA の設定

```
aaa new-model
dot1x system-auth-control
radius server ISE
 address ipv4 <ISE ipv4 address> auth-port 1645 acct-port 1646
 automate-tester username dummy
 key dummy123
 radius-server deadtime 2
!
aaa group server radius ISEGRP
 server name ISE
!
```

```
aaa authentication dot1x default group ISEGRP
aaa authorization network default group ISEGRP
```

例：EAP-TLS プロファイルと 802.1x クレデンシャルの設定

```
eap profile EAPTLS-PROF-IOSCA
 method tls
 pki-trustpoint POLESTAR-IOS-CA
!

dot1x credentials EAPTLSCRED-IOSCA
 username asr1000@polestar.company.com
 pki-trustpoint POLESTAR-IOS-CA
!
```

例：インターフェイスでの 802.1 X、PKI、および MACsec の設定の適用

```
interface TenGigabitEthernet0/1
 macsec network-link
 authentication periodic
 authentication timer reauthenticate <reauthentication interval>
 access-session host-mode multi-host
 access-session closed
 access-session port-control auto
 dot1x pae both
 dot1x credentials EAPTLSCRED-IOSCA
 dot1x supplicant eap profile EAPTLS-PROF-IOSCA
 service-policy type control subscriber DOT1X_POLICY_RADIUS
```

Cisco TrustSec スイッチ間リンク セキュリティの設定例

次に、Cisco TrustSec スイッチ間のセキュリティのためにシードおよび非シード デバイスに必要な設定を示します。リンク セキュリティ用に AAA および RADIUS を設定する必要があります。この例では、ACS-1 から ACS-3 は任意のサーバー名、cts-radius は Cisco TrustSec サーバーです。

シード デバイスの設定

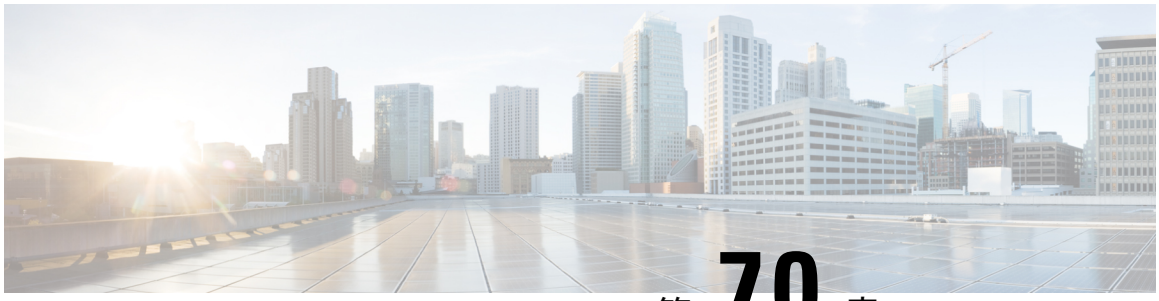
```
Switch(config)# aaa new-model
Switch(config)# radius server ACS-1
Switch(config-radius-server)# address ipv4 10.5.120.12 auth-port 1812 acct-port
 1813
Switch(config-radius-server)# pac key cisco123
Switch(config-radius-server)# exit
Switch(config)# radius server ACS-2
Switch(config-radius-server)# address ipv4 10.5.120.14 auth-port 1812 acct-port
 1813
Switch(config-radius-server)# pac key cisco123
Switch(config-radius-server)# exit
Switch(config)# radius server ACS-3
```

```
Switch(config-radius-server)# address ipv4 10.5.120.15 auth-port 1812 acct-port
 1813
Switch(config-radius-server)# pac key cisco123
Switch(config-radius-server)# exit
Switch(config)# aaa group server radius cts-radius
Switch(config-sg-radius)# server name ACS-1
Switch(config-sg-radius)# server name ACS-2
Switch(config-sg-radius)# server name ACS-3
Switch(config-sg-radius)# exit
Switch(config)# aaa authentication login default none
Switch(config)# aaa authentication dot1x default group cts-radius
Switch(config)# aaa authorization network cts-radius group cts-radius
Switch(config)# aaa session-id common
Switch(config)# cts authorization list cts-radius
Switch(config)# dot1x system-auth-control

Switch(config)# interface gi1/1/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# cts dot1x
Switch(config-if-cts-dot1x)# sap mode-list gcm-encrypt gmac
Switch(config-if-cts-dot1x)# no propagate sgt
Switch(config-if-cts-dot1x)# exit
Switch(config-if)# exit

Switch(config)# interface gi1/1/4
Switch(config-if)# switchport mode trunk
Switch(config-if)# cts manual
Switch(config-if-cts-manual)# sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
 gmac
Switch(config-if-cts-manual)# no propagate sgt
Switch(config-if-cts-manual)# exit
Switch(config-if)# exit

Switch(config)# radius-server vsa send authentication
Switch(config)# exit
Switch# cts credentials id cts-36 password trustsec123
```

非シード デバイス

```
Switch(config)# aaa new-model
Switch(config)# aaa session-id common
Switch(config)# dot1x system-auth-control

Switch(config)# interface gi1/1/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# shutdown
Switch(config-if)# cts dot1x
Switch(config-if-cts-dot1x)#sap mode-list gcm-encrypt gmac
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit

Switch(config)# interface gi1/1/4
Switch(config-if)# switchport mode trunk
Switch(config-if)# shutdown
Switch(config-if)# cts manual
```

```
Switch(config-if-cts-manual)# sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
 gmac
Switch(config-if-cts-manual)# no propagate sgt
Switch(config-if-cts-manual)# exit
Switch(config-if)# exit

Switch(config)# radius-server vsa send authentication
Switch(config)# end
Switch# cts credentials id cts-72 password trustsec123
```



第 70 章

Web ベース認証

この章では、デバイスで Web ベース認証を設定する方法について説明します。この章の内容は、次のとおりです。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

Web ベース認証の概要

IEEE 802.1x サプリカントが実行されていないホストシステムでエンドユーザーを認証するには、Web 認証プロキシとして知られている Web ベース認証機能を使用します。

HTTP セッションを開始すると、Web ベース認証は、ホストからの受信 HTTP パケットを横取りし、ユーザーに HTML ログイン ページを送信します。ユーザーはクレデンシャルを入力します。このクレデンシャルは、Webベース認証機能により、認証のために認証、許可、アカウンティング（AAA）サーバーに送信されます。

認証が成功すると、Web ベース認証はログイン成功 HTML ページをホストに送信し、AAA サーバーから返されたアクセス ポリシーを適用します。

認証に失敗した場合、Web ベース認証は、ログインの失敗を示す HTML ページをユーザーに転送し、ログインを再試行するように、ユーザーにプロンプトを表示します。最大試行回数を超過した場合、Web ベース認証は、ログインの期限切れを示す HTML ページをホストに転送し、このユーザーは待機期間中、ウォッチ リストに載せられます。

（注） 中央 Web 認証リダイレクト用の HTTPS トラフィック インターセプションはサポートされていません。

（注） グローバル パラメータ マップ（method-type、custom、redirect）は、すべてのクライアントおよび SSID で同じ Web 認証方式（consent、web consent、webauth など）を使用するときにのみ使用する必要があります。これにより、すべてのクライアントが同じ Web 認証方式になります。

要件により、1 つの SSID に consent、別の SSID に webauth を使用する場合、名前付きパラメータ マップを 2 つ使用する必要があります。1 番目のパラメータ マップには consent を設定し、2 番目のパラメータ マップには webauth を設定する必要があります。

（注） Webauth クライアントの認証試行時に受信する traceback には、パフォーマンスや行動への影響はありません。これは、ACL アプリケーションの EPM に FFM が返信したコンテキストがすでにキュー解除済み（タイマーの有効期限切れの可能性あり）で、セッションが「未承認」になった場合にまれに発生します。

Web ページがホストされている場所に基づいて、ローカル Web 認証は次のように分類できます。

- 内部：ローカル Web 認証時に、コントローラの内部デフォルト HTML ページ（ログイン、成功、失敗、および期限切れ）が使用されます。
- カスタマイズ：ローカル Web 認証時に、カスタマイズされた Web ページ（ログイン、成功、失敗、および期限切れ）がコントローラにダウンロードされ、使用されます。
- 外部：組み込みまたはカスタム Web ページを使用する代わりに、外部 Web サーバー上でカスタマイズされた Web ページがホストされます。

さまざまな Web 認証ページに基づき、Web 認証のタイプは次のように分類できます。

- *Webauth*：これが基本的な Web 認証です。この場合、コントローラはユーザー名とパスワードの入力が必要なポリシーページを提示します。ネットワークにアクセスするには、ユーザーは正しいクレデンシャルを入力する必要があります。
- *Consent* または *web-passthrough*：この場合、コントローラは [Accept] ボタンまたは [Deny] ボタンが表示されたポリシー ページを提示します。ネットワークにアクセスするには、ユーザーは [Accept] ボタンをクリックする必要があります。

- *Webconsent*：これは webauth と consent の Web 認証タイプの組み合わせです。この場合、コントローラは [Accept] ボタンまたは [Deny] ボタンがあり、ユーザー名とパスワードの入力が必要なポリシーページを提示します。ネットワークにアクセスするには、ユーザーは正しいクレデンシャルを入力して [Accept] ボタンをクリックする必要があります。

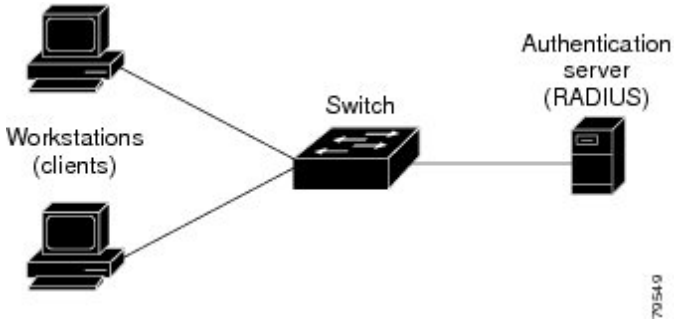
デバイスのロール

Web ベース認証では、ネットワーク上のデバイスに次のような固有の役割があります。

- クライアント：LAN およびサービスへのアクセスを要求し、スイッチからの要求に応答するデバイス（ワークステーション）。このワークステーションでは、Java Script が有効な HTML ブラウザが実行されている必要があります。
- 認証サーバー：クライアントを認証します。認証サーバーはクライアントの識別情報を確認し、そのクライアントがLANおよびスイッチのサービスへのアクセスを許可されたか、あるいはクライアントが拒否されたのかをスイッチに通知します。
- スイッチ：クライアントの認証ステータスに基づいて、ネットワークへの物理アクセスを制御します。スイッチはクライアントと認証サーバーとの仲介デバイス（プロキシ）として動作し、クライアントに識別情報を要求し、その情報を認証サーバーで確認し、クライアントに応答をリレーします。

図 *114 : Web* ベース認証デバイスの役割

次の図は、ネットワーク上でのこれらのデバイスの役割を示します。



ホストの検出

スイッチは、検出されたホストに関する情報を格納するために、IP デバイス追跡 テーブルを維持します。

（注） デフォルトでは、スイッチの IP デバイス追跡機能は無効にされています。Web ベース認証を使用するには、IP デバイス追跡機能を有効にする必要があります。

レイヤ 2 インターフェイスでは、Web ベース認証は、これらのメカニズムを使用して、IP ホストを検出します。

- ARP ベースのトリガー：ARP リダイレクト ACL により、Web ベース認証は、スタティック IP アドレス、またはダイナミック IP アドレスを持つホストを検出できます。
- ダイナミック ARP検査
- DHCP スヌーピング：スイッチがホストの DHCP バインディング エントリを作成するときに Web ベース認証が通知されます。

セッションの作成

Web ベース認証により、新しいホストが検出されると、次のようにセッションが作成されます。

- 例外リストをレビューします。

  ホスト IP が例外リストに含まれている場合、この例外リスト エントリからポリシーが適用され、セッションが確立されます。

- 認証バイパスをレビューします。

  ホストIPが例外リストに含まれていない場合、Webベース認証は応答しないホスト（NRH）要求をサーバーに送信します。

  サーバーの応答が access accepted であった場合、認証はこのホストにバイパスされます。セッションが確立されます。

- HTTP インターセプト ACL を設定します。

  NRH 要求に対するサーバーの応答が access rejected であった場合、HTTP インターセプト ACL がアクティブ化され、セッションはホストからの HTTP トラフィックを待機します。

認証プロセス

Web ベース認証を有効にすると、次のイベントが発生します。

- ユーザーが HTTP セッションを開始します。
- HTTP トラフィックが横取りされ、認証が開始されます。スイッチは、ユーザーにログインページを送信します。ユーザーはユーザー名とパスワードを入力します。スイッチはこのエントリを認証サーバーに送信します。
- 認証に成功した場合、スイッチは認証サーバーからこのユーザーのアクセスポリシーをダウンロードし、アクティブ化します。ログインの成功ページがユーザーに送信されます
- 認証に失敗した場合は、スイッチはログインの失敗ページを送信します。ユーザーはログインを再試行します。失敗の回数が試行回数の最大値に達した場合、スイッチはログイン期限切れページを送信します。このホストはウォッチ リストに入れられます。ウォッチ リストのタイム アウト後、ユーザーは認証プロセスを再試行することができます。

- 認証サーバーがスイッチに応答せず、AAA 失敗ポリシーが設定されている場合、スイッチはホストに失敗アクセス ポリシーを適用します。ログインの成功ページがユーザーに送信されます
- ホストがレイヤ 2 インターフェイス上の ARP プローブに応答しなかった場合、またはホストがレイヤ 3 インターフェイスでアイドル タイムアウト内にトラフィックを送信しなかった場合、スイッチはクライアントを再認証します。
- ホストがレイヤ 2 インターフェイス上の ARP プローブに応答しない場合、スイッチはクライアントを再認証します。
- この機能は、ダウンロードされたタイムアウト、またはローカルに設定されたセッション タイムアウトを適用します。
- Termination-Action が RADIUS である場合、この機能は、サーバーに NRH 要求を送信します。Termination-Action は、サーバーからの応答に含まれます。
- Termination-Action がデフォルトである場合、セッションは廃棄され、適用されたポリシーは削除されます。

ローカル Web 認証バナー

Web 認証を使用して、デフォルトのカスタマイズ済み Web ブラウザ バナーを作成して、スイッチにログインしたときに表示するようにできます。

このバナーは、ログイン ページと認証結果ポップアップ ページの両方に表示されます。デフォルトのバナー メッセージは次のとおりです。

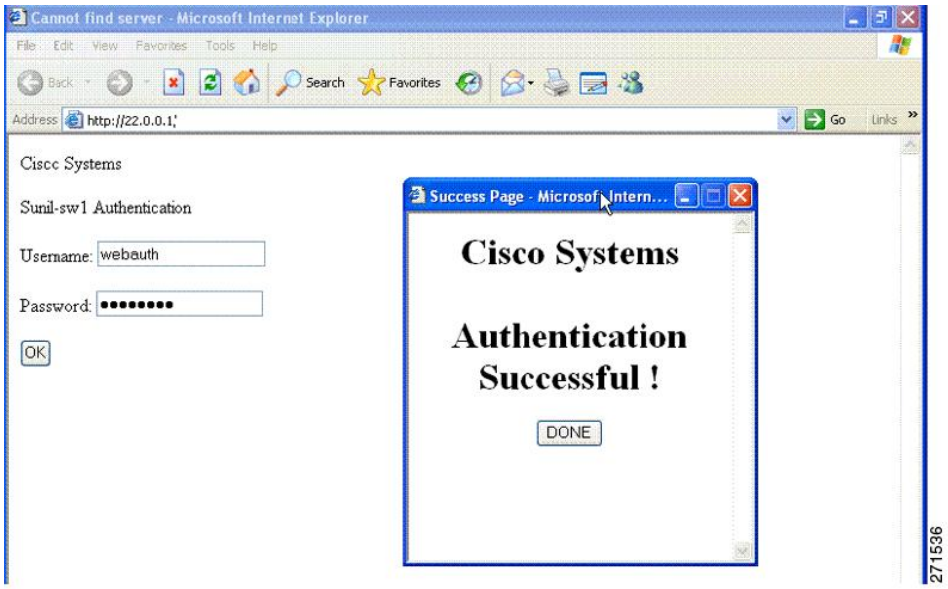
- 認証成功
- 認証失敗
- 認証期限切れ

ローカル ネットワーク認証バナーは、レガシーおよび新スタイル（セッション アウェア）の CLI で次のように設定できます。

- レガシー モード：**ip admission auth-proxy-banner http** グローバル コンフィギュレーション コマンドを使用します。
- 新スタイル モード：**parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。

ログイン ページには、デフォルトのバナー、*Cisco Systems*、および *Switch host-name Authentication* が表示されます。*Cisco Systems* は認証結果ポップアップ ページに表示されます。

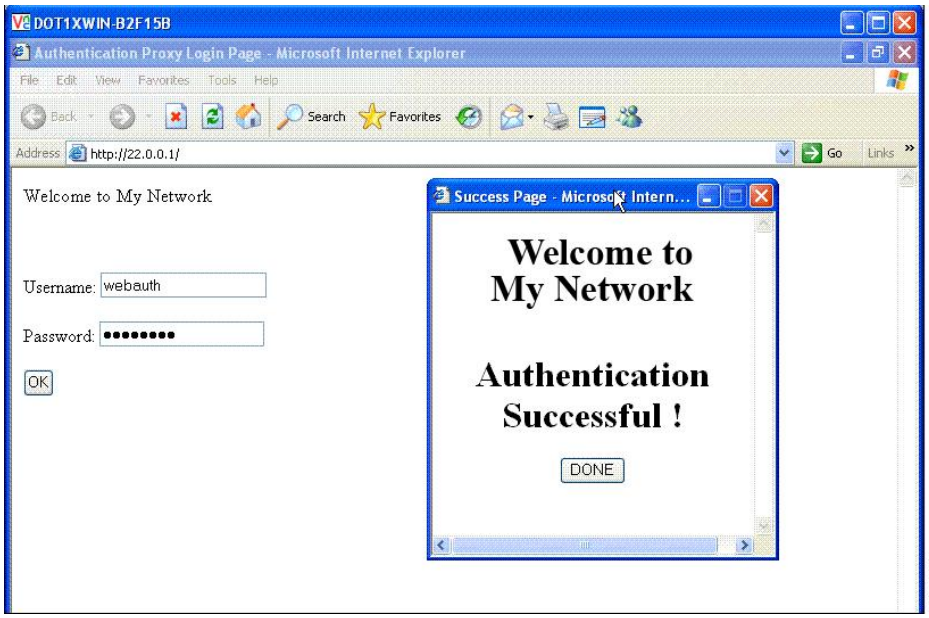
図 *115 :* 認証成功バナー



バナーは次のようにカスタマイズ可能です。

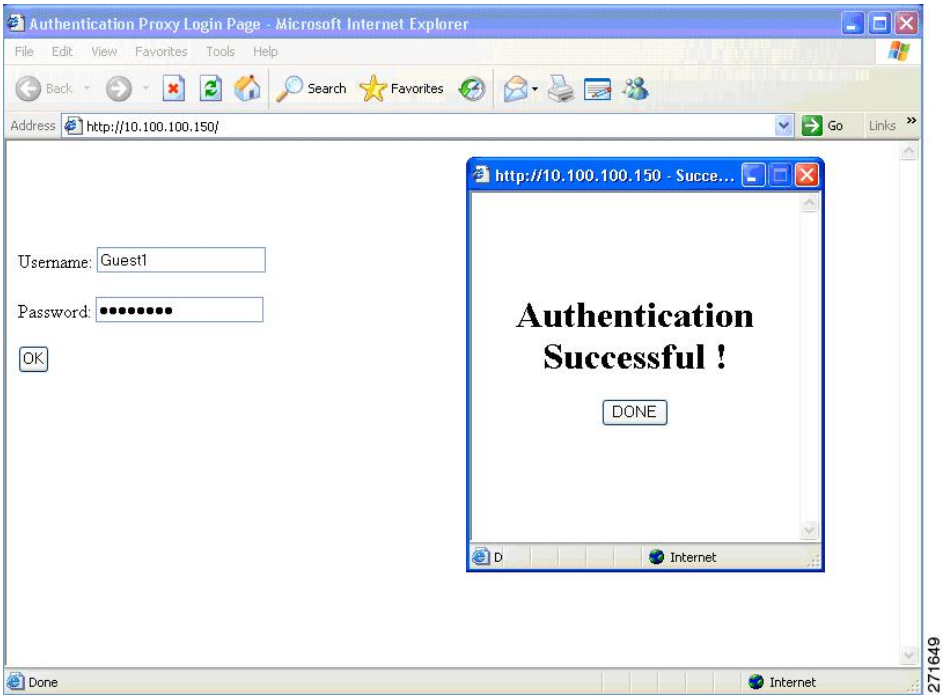
- スイッチ名、ルータ名、または会社名などのメッセージをバナーに追加する。
  - レガシーモード： **ip admission auth-proxy-banner http** *banner-text* グローバル コンフィギュレーション コマンドを使用します。
  - 新スタイル モード：**parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。
- ロゴまたはテキスト ファイルをバナーに追加する。
  - レガシーモード：**ip admission auth-proxy-banner http** *file-path* グローバル コンフィギュレーション コマンドを使用します。
  - 新スタイル モード：**parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。

図 *116* : カスタマイズされた *Web* バナー



バナーが有効にされていない場合、Web認証ログイン画面にはユーザー名とパスワードのダイアログボックスだけが表示され、スイッチにログインしたときにはバナーは表示されません。

図 *117* : バナーが表示されていないログイン画面



Web 認証カスタマイズ可能な Web ページ

Webベース認証プロセスでは、スイッチ内部のHTTPサーバーは、認証中のクライアントに配信される4種類のHTMLページをホストします。サーバーはこれらのページを使用して、ユーザーに次の 4 種類の認証プロセス ステートを通知します。

- ログイン：資格情報が要求されています。
- 成功：ログインに成功しました。
- 失敗：ログインに失敗しました。
- 期限切れ：ログインの失敗回数が多すぎて、ログイン セッションが期限切れになりました。

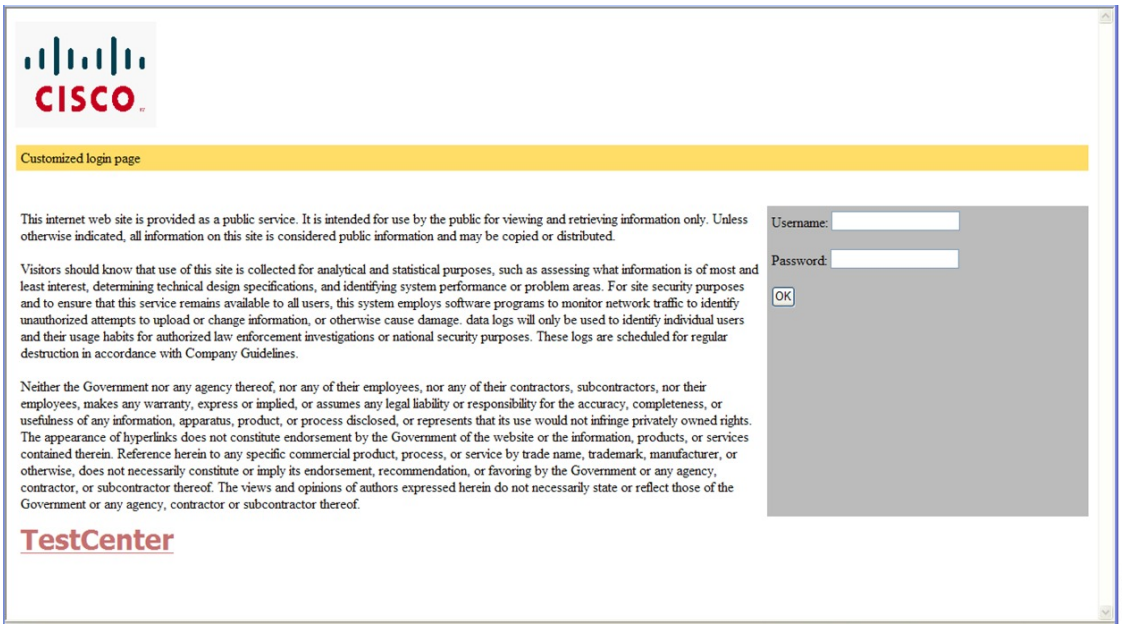
ガイドライン

- デフォルトの内部 HTML ページの代わりに、独自の HTML ページを使用することができます。
- ロゴを使用することもできますし、ログイン、成功、失敗、および期限切れ Web ページでテキストを指定することもできます。
- バナー ページで、ログイン ページのテキストを指定できます。
- これらのページは、HTML で記述されています。
- 成功ページには、特定の URL にアクセスするための HTML リダイレクト コマンドを記入する必要があります。
- この URL 文字列は有効な URL（例：http://www.cisco.com）でなければなりません。不完全なURLは、Webブラウザで、「ページが見つかりません」またはこれに類似するエラーの原因となる可能性があります。
- HTTP 認証で使用される Web ページを設定する場合、これらのページには適切な HTML コマンド（例：ページのタイムアウトを設定、暗号化されたパスワードの設定、同じページが 2 回送信されていないことの確認など）を記入する必要があります.
- 設定されたログイン フォームが有効な場合、特定の URL にユーザーをリダイレクトする CLI コマンドは使用できません。管理者は、Webページにリダイレクトが設定されていることを保証する必要があります。
- 認証後、特定の URL にユーザーをリダイレクトする CLI コマンドを入力してから、Web ページを設定するコマンドを入力した場合、特定の URL にユーザーをリダイレクトする CLI コマンドは効力を持ちません。
- 設定された Web ページは、スイッチのブート フラッシュ、またはフラッシュにコピーできます。
- ログインページを任意のフラッシュ上に、成功ページと失敗ページを別のフラッシュ（たとえば、アクティブスイッチ、またはメンバスイッチのフラッシュ）に配置できます。

- 4 ページすべてを設定する必要があります。
- Web ページを使ってバナー ページを設定した場合、このバナー ページには効果はありません。
- システムディレクトリ（たとえば、flash、disk0、disk）に保存されていて、ログイン ページに表示する必要のあるロゴ ファイル（イメージ、フラッシュ、オーディオ、ビデオなど）すべてには、必ず、*web_auth_<filename>* の形式で名前をつけてください。
- 設定された認証プロキシ機能は、HTTP と SSL の両方をサポートしています。

デフォルトの内部 HTML ページの代わりに、自分の HTML ページを使用することができます。認証後のユーザーのリダイレクト先で、内部成功ページの代わりとなる URL を指定することもできます。

図 *118* : カスタマイズ可能な認証ページ



認証プロキシ Web ページの注意事項

カスタマイズされた認証プロキシ Web ページを設定する際には、次の注意事項に従ってください。

- カスタム Web ページ機能を有効にするには、カスタム HTML ファイルを 4 個すべて指定します。指定したファイルの数が 4 個未満の場合、内部デフォルト HTML ページが使用されます。
- これら 4 個のカスタム HTML ファイルは、スイッチのフラッシュ メモリ内に存在しなければなりません。各 HTML ファイルの最大サイズは 8 KB です。
- カスタム ページ上のイメージはすべて、アクセス可能は HTTP サーバー上に存在しなければなりません。インターセプト ACL は、管理ルール内で設定します。

- カスタム ページからの外部リンクはすべて、管理ルール内でのインターセプト ACL の設定を必要とします。
- 有効な DNS サーバーにアクセスするには、外部リンクまたはイメージに必要な名前解決で、管理ルール内にインターセプト ACL を設定する必要があります。
- カスタム Web ページ機能が有効に設定されている場合、設定された auth-proxy-banner は使用されません。
- カスタム Web ページ機能が有効に設定されている場合、ログインの成功に対するリダイレクション URL は使用できません。
- カスタム ファイルの指定を解除するには、このコマンドの **no** 形式を使用します。

カスタム ログイン ページはパブリック Web フォームであるため、このページについては、次の注意事項に従ってください。

- ログイン フォームは、ユーザーによるユーザー名とパスワードの入力を受け付け、これらを **uname** および **pwd** として示す必要があります。
- カスタム ログイン ページは、ページ タイムアウト、暗号化されたパスワード、冗長送信の防止など、Web フォームに対するベスト プラクティスに従う必要があります。

成功ログインに対するリダイレクト URL の注意事項

成功ログインに対するリダイレクション URL を設定する場合、次の注意事項に従ってください。

- カスタム認証プロキシ Web ページ機能がイネーブルに設定されている場合、リダイレクション URL 機能はディセーブルにされ、CLI では使用できません。リダイレクションは、カスタム ログイン成功ページで実行できます。
- リダイレクション URL 機能が有効に設定されている場合、設定された auth-proxy-banner は使用されません。
- リダイレクション URL の指定を解除するには、このコマンドの **no** 形式を使用します。
- Web ベースの認証クライアントが正常に認証された後にリダイレクション URL が必要な場合、URL 文字列は有効な URL（たとえば http://）で開始し、その後に URL 情報が続く必要があります。http:// を含まない URL が指定されると、正常に認証が行われても、そのリダイレクション URL によって Web ブラウザでページが見つからないまたは同様のエラーが生じる場合があります。

その他の機能と Web ベース認証の相互作用

ポート セキュリティ

Web ベース認証とポート セキュリティは、同じポートに設定できます。Web ベース認証はポートを認証し、ポート セキュリティは、クライアントの MAC アドレスを含むすべての MAC ア

ドレスに対するネットワーク アクセスを管理します。この場合、このポートを介してネットワークへアクセスできるクライアントの数とグループを制限できます。

LAN ポート IP

LAN ポート IP（LPIP）とレイヤ 2 Web ベース認証は、同じポートに設定できます。ホストは、まず Web ベース認証、次に LPIP ポスチャ検証を使用して認証されます。LPIP ホスト ポリシーは、Web ベース認証のホスト ポリシーに優先されます。

Web ベース認証のアイドル時間が満了すると、NAC ポリシーは削除されます。ホストが認証され、ポスチャが再度検証されます。

ゲートウェイ IP

VLAN のいずれかのスイッチ ポートで Web ベース認証が設定されている場合、レイヤ 3 VLAN インターフェイス上にゲートウェイ IP（GWIP）を設定することはできません。

Web ベース認証はゲートウェイ IP と同じレイヤ 3 インターフェイスに設定できます。ソフトウェアで、両方の機能のホスト ポリシーが適用されます。GWIP ホスト ポリシーは、Web ベース認証のホスト ポリシーに優先されます。

ACL

インターフェイスで VLAN ACL、または Cisco IOS ACL を設定した場合、ACL は、Web ベース認証のホスト ポリシーが適用された後だけ、ホスト トラフィックに適用されます。

レイヤ 2 Web ベース認証では、ポートに接続されたホストからの受信トラフィックについて、ポート ACL（PACL）をデフォルトのアクセス ポリシーとして設定することが必須ではないものの、より安全です。認証後、Web ベース認証のホスト ポリシーは、PACL に優先されます。ポートに設定された ACL がなくても、ポリシー ACL はセッションに適用されます。

MAC ACL と Web ベース認証を同じインターフェイスに設定することはできません。

アクセス VLAN が VACL キャプチャ用に設定されているポートには Web ベース認証は設定できません。

コンテキストベース アクセス コントロール

コンテキストベース アクセス コントロール（CBAC）が、ポート VLAN のレイヤ 3 VLAN インターフェイスで設定されている場合、レイヤ 2 ポートで Web ベース認証は設定できません。

EtherChannel

Web ベース認証は、レイヤ 2 EtherChannel インターフェイス上に設定できます。Web ベース認証設定は、すべてのメンバ チャネルに適用されます。

Web ベース認証の設定方法

デフォルトの Web ベース認証の設定

次の表に、デフォルトの Web ベース認証の設定を示しています。

表 *147* : デフォルトの *Web* ベース認証の設定

機能	デフォルト設定
AAA	無効
RADIUS サーバ • IP アドレス • UDP 認証ポート • キー	• 指定なし • 1645 • 指定なし
無活動タイムアウトのデフォルト値	3600 秒
無活動タイムアウト	有効

Web ベース認証の設定に関する注意事項と制約事項

- Web ベース認証は受信時だけの機能です。
- Web ベース認証は、アクセス ポートだけで設定できます。Web ベース認証は、トランク ポート、EtherChannel メンバ ポート、またはダイナミック トランク ポートではサポートされていません。
- スイッチが特定のホストまたは Web サーバーにクライアントをリダイレクトしてログイン メッセージを表示する場合、外部 Web 認証はサポートされません。
- スタティックな ARP キャッシュが割り当てられているレイヤ 2 インターフェイス上のホストは認証できません。これらのホストは ARP メッセージを送信しないため、Web ベース認証機能では検出されません。
- デフォルトでは、スイッチの IP デバイス追跡機能は無効にされています。Web ベース認証を使用するには、IP デバイス追跡機能を有効にする必要があります。
- Web ベース認証を使用するには、SISF ベースのデバイス追跡を有効にする必要があります。デフォルトでは、SISF ベースのデバイス追跡はスイッチで無効になっています。

- スイッチ HTTP サーバーを実行するには、IP アドレスを少なくとも 1 つ設定する必要があります。また、各ホスト IP アドレスに到達するようにルートを設定する必要もあります。HTTP サーバーは、ホストに HTTP ログイン ページを送信します。
- 2 ホップ以上離れたところにあるホストでは、STP トポロジの変更により、ホスト トラフィックの到着するポートが変わってしまった場合、トラフィックが停止する可能性があります。これは、レイヤ 2（STP）トポロジの変更後に、ARP および DHCP の更新が送信されていない場合に発生します。
- Web ベース認証は、ダウンロード可能なホスト ポリシーとして、VLAN 割り当てをサポートしていません。
- Web ベース認証はセッション認識型ポリシー モードで IPv6 をサポートします。IPv6 Web 認証には、スイッチで設定された少なくても 1 つの IPv6 アドレスおよびスイッチ ポートに設定された IPv6 スヌーピングが必要です。
- Web ベース認証および Network Edge Access Topology（NEAT）は、相互に排他的です。インターフェイス上で NEAT が有効の場合、Web ベース認証を使用できず、インターフェイス上で Web ベース認証が実行されている場合は、NEAT を使用できません。
- スイッチから RADIUS サーバーへの通信の設定に使用される次の RADIUS セキュリティ サーバー設定を確認します。
  - ホスト名
  - ホスト IP アドレス
  - ホスト名と特定の UDP ポート番号
  - IP アドレスと特定の UDP ポート番号

  IP アドレスと UDP ポート番号の組み合わせによって、一意の ID が作成され、サーバーの同一 IP アドレス上にある複数の UDP ポートに RADIUS 要求を送信できるようになります。同じ RADIUS サーバー上の異なる 2 つのホスト エントリに同じサービス（たとえば認証）を設定した場合、2 番めに設定されたホスト エントリは、最初に設定されたホスト エントリのフェールオーバー バックアップとして動作します。RADIUS ホスト エントリは、設定した順序に従って選択されます。
- RADIUS サーバー パラメータを設定する場合は、次の点に注意してください。
  - 別のコマンドラインに、**key** *string* を指定します。
  - **key** *string* には、スイッチと、RADIUS サーバー上で動作する RADIUS デーモンとの間で使用する、認証および暗号キーを指定します。キーは、RADIUS サーバーで使用する暗号化キーに一致するテキスト ストリングでなければなりません。
  - **key** *string* を指定する場合、キーの中間、および末尾にスペースを使用します。キーにスペースを使用する場合は、引用符がキーの一部分である場合を除き、引用符でキーを囲まないでください。キーは RADIUS デーモンで使用する暗号に一致している必要があります。

- すべてのRADIUSサーバーについて、タイムアウト、再送信回数、および暗号キー値をグローバルに設定するには、**radius-server host** グローバル コンフィギュレーション コマンドを使用します。これらのオプションをサーバー単位で設定するには、**radius-server timeout**、radius-server transmit、および radius-server key グローバル コンフィギュレーション コマンドを使用します。

（注） RADIUS サーバーでは、スイッチの IP アドレス、サーバーとスイッチで共有される key string、およびダウンロード可能なACL（DACL）などの設定を行う必要があります。詳細については、RADIUSサーバーのマニュアルを参照してください。

認証ルールとインターフェイスの設定

認証ルールおよびインターフェイスを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission name** *name* **proxy http**
4. **interface** *type slot/port*
5. **ip access-group** *name*
6. **ip admission name**
7. **exit**
8. **ip device tracking**
9. **end**
10. **show ip admission status**
11. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**ip admission name** *name* **proxy http** 例： スイッチ(config)# **ip admission name webauth1 proxy http**	Web ベース許可の認証ルールを設定します。
ステップ 4	**interface** *type slot/port* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	インターフェイス コンフィギュレーション モードを開始し、Web ベース認証を有効にする受信レイヤ 2 またはレイヤ 3 インターフェイスを指定します。 *type* には、fastethernet、gigabit ethernet、または tengigabitethernet を指定できます。
ステップ 5	**ip access-group** *name* 例： スイッチ(config-if)# **ip access-group webauthag**	デフォルト ACL を適用します。
ステップ 6	**ip admission name** 例： スイッチ(config)# **ip admission name**	インターフェイスの Web ベース認可の認証ルールを設定します。
ステップ 7	**exit** 例： スイッチ(config-if)# **exit**	コンフィギュレーション モードに戻ります。
ステップ 8	**ip device tracking** 例： スイッチ(config)# **ip device tracking**	IP デバイス トラッキング テーブルをイネーブルにします。
ステップ 9	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 10	**show ip admission status** 例： スイッチ# **show ip admission**	設定を表示します。
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

AAA 認証の設定

手順の概要

1. **aaa new-model**
2. **aaa authentication login default group** {**tacacs+** | **radius**}
3. **aaa authorization auth-proxy default group** {**tacacs+** | **radius**}

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**aaa new-model** 例： スイッチ(config)# **aaa new-model**	AAA 機能を有効にします。
ステップ 2	**aaa authentication login default group** {**tacacs+**	**radius**} 例： スイッチ(config)# **aaa authentication login default group tacacs+**	ログイン時の認証方法のリストを定義します。 **named_authentication_list** は、31 文字未満の名前を示します。 **AAA_group_name** はサーバー グループ名を示します。サーバーグループ **server_name** をその先頭で定義する必要があります。
ステップ 3	**aaa authorization auth-proxy default group** {**tacacs+**	**radius**} 例： スイッチ(config)# **aaa authorization auth-proxy default group tacacs+**	Web ベース許可の許可方式リストを作成します。

スイッチ/RADIUS サーバー間通信の設定

RADIUS サーバーのパラメータを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip radius source-interface vlan** *vlan interface number*
4. **radius-server host** {*hostname* | *ip-address*} **test username** *username*
5. **radius-server key** *string*
6. **radius-server dead-criteria tries** *num-tries*
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip radius source-interface vlan** *vlan interface number* 例： `スイッチ(config)# ip radius source-interface vlan 80`	RADIUS パケットが、指定されたインターフェイスの IP アドレスを含むように指定します。
ステップ 4	**radius-server host** {*hostname* | *ip-address*} **test username** *username* 例： `スイッチ(config)# radius-server host 172.120.39.46 test username user1`	リモート RADIUS サーバーのホスト名または IP アドレスを指定します。 **test username** *username* は、RADIUS サーバー接続の自動テストをイネーブルにするオプションです。指定された *username* は有効なユーザー名である必要はありません。 **key** オプションは、スイッチと RADIUS サーバ間で使用される認証と暗号キーを指定します。

	コマンドまたはアクション	目的
		複数のRADIUSサーバを使用するには、それぞれのサーバでこのコマンドを再入力してください。
ステップ 5	**radius-server key** *string* 例： スイッチ(config)# **radius-server key rad123**	スイッチと、RADIUS サーバーで動作する RADIUS デーモン間で使用される認証および暗号キーを設定します。
ステップ 6	**radius-server dead-criteria tries** *num-tries* 例： スイッチ(config)# **radius-server dead-criteria tries 30**	RADIUS サーバーに送信されたメッセージへの応答がない場合に、このサーバーが非アクティブであると見なすまでの送信回数を指定します。指定できる *num-tries* の範囲は 1 ～ 100 です。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

HTTP サーバーの設定

Web ベース認証を使用するには、スイッチで HTTP サーバをイネーブルにする必要があります。このサーバーは HTTP または HTTPS のいずれかについて有効にできます。

（注） Apple の疑似ブラウザは、**ip http secure-server** コマンドを設定するだけでは開きません。**ip http server** コマンドも設定する必要があります。

HTTP または HTTPS のいずれかについてサーバーを有効にするには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **ip http secure-server**
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip http server** 例： スイッチ(config)# **ip http server**	HTTP サーバーを有効にします。Web ベース認証機能は、HTTP サーバーを使用してホストと通信し、ユーザー認証を行います。
ステップ 4	**ip http secure-server** 例： スイッチ(config)# **ip http secure-server**	HTTPS を有効にします。 カスタム認証プロキシ Web ページを設定するか、成功ログインのリダイレクション URL を指定します。 （注） **ip http secure-server** コマンドを入力したときに、セキュア認証が確実に行われるようにするには、ユーザーが HTTP 要求を送信した場合でも、ログイン ページは必ず HTTPS（セキュア HTTP）形式になるようにします。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

認証プロキシ Web ページのカスタマイズ

Web ベースの認証中、スイッチのデフォルト HTML ページではなく、代わりの HTML ページがユーザーに表示されるように、Web 認証を設定できます。

カスタム認証プロキシ Web ページの使用を指定するには、次の手順を実行してください。

始める前に

スイッチのフラッシュ メモリにカスタム HTML ファイルを保存します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission proxy http login page file** *device:login-filename*
4. **ip admission proxy http success page file** *device:success-filename*
5. **ip admission proxy http failure page file** *device:fail-filename*
6. **ip admission proxy http login expired page file** *device:expired-filename*
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip admission proxy http login page file** *device:login-filename* 例： `スイッチ(config)# ip admission proxy http login page file disk1:login.htm`	スイッチのメモリ ファイル システム内で、デフォルトのログイン ページの代わりに使用するカスタム HTML ファイルの場所を指定します。*device:* はフラッシュ メモリです。
ステップ 4	**ip admission proxy http success page file** *device:success-filename* 例： `スイッチ(config)# ip admission proxy http success page file disk1:success.htm`	デフォルトのログイン成功ページの代わりに使用するカスタム HTML ファイルの場所を指定します。
ステップ 5	**ip admission proxy http failure page file** *device:fail-filename* 例：	デフォルトのログイン失敗ページの代わりに使用するカスタム HTML ファイルの場所を指定します。

	コマンドまたはアクション	目的
	スイッチ(config)# **ip admission proxy http fail page file disk1:fail.htm**	
ステップ 6	**ip admission proxy http login expired page file** *device:expired-filename* 例： スイッチ(config)# **ip admission proxy http login expired page file disk1:expired.htm**	デフォルトのログイン失効ページの代わりに使用するカスタム HTML ファイルの場所を指定します。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

成功ログインに対するリダイレクション URL の指定

認証後に内部成功 HTML ページを効果的に置き換えユーザーのリダイレクト先となる URL を指定するためには、次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **configure terminal**
3. **ip admission proxy http success redirect** *url-string*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**ip admission proxy http success redirect** *url-string* 例： `スイッチ(config)# ip admission proxy http success redirect www.example.com`	デフォルトのログイン成功ページの代わりにユーザーをリダイレクトする URL を指定します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

Web ベース認証パラメータの設定

クライアントが待機時間中にウォッチリストに掲載されるまで許容される失敗ログイン試行の最大回数を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission max-login-attempts** *number*
4. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip admission max-login-attempts** *number* 例： `Device(config)# ip admission max-login-attempts`	失敗ログイン試行の最大回数を設定します。指定できる範囲は 1 ～ 2147483647 回です。デフォルトは 5 分です。

	コマンドまたはアクション	目的
	`10`	
ステップ 4	**exit** 例： `Device# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Web ベースの認証ローカル バナーの設定

Web 認証が設定されているスイッチにローカル バナーを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission auth-proxy-banner http** [*banner-text* | *file-path*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip admission auth-proxy-banner http** [*banner-text*	*file-path*] 例： `スイッチ(config)# ip admission auth-proxy-banner http C My Switch C`	ローカル バナーを有効にします。 （任意）*C banner-text C*（*C* は区切り文字）、またはバナーに表示されるファイル（たとえばロゴまたはテキストファイル）のファイルパスを入力して、カスタムバナーを作成します。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SVI を使用しない Web ベース認証の設定

ルーティング テーブルに IP アドレスを作成せずに、HTML のログイン ページがクライアントにリダイレクトする SVI 機能なしの Web ベースの認証を設定します。これらの手順は任意です。

手順の概要

1. **enable**
2. **configure terminal**
3. **parameter-map type webauth global**
4. **l2-webauth-enabled**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**parameter-map type webauth global** 例： スイッチ (config)# **parameter-map type webauth global**	パラメータ マップを作成し、parameter-map webauth コンフィギュレーション モードを開始します。グローバル キーワードで定義されたグローバル パラメータ マップでサポートされる特定のコンフィギュレーション コマンドは、parameter-map-name 引数で定義された名前付きパラメータ マップでサポートされるコマンドとは異なります。
ステップ 4	**l2-webauth-enabled** 例： スイッチ (config-params-parameter-map)# **l2-webauth-enabled**	SVI 機能なしの Web ベースの認証を有効にします
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーションファイルに設定を保存します。

VRF 認識による Web ベース認証の設定

HTML のログイン ページがクライアントにリダイレクトする VRF 認識による Web ベース認証を設定します。これらの手順は任意です。

手順の概要

1. **enable**

2. **configure terminal**
3. **parameter-map type webauth global**
4. **webauth-vrf-aware**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**parameter-map type webauth global** 例： スイッチ (config)# **parameter-map type webauth global**	パラメータ マップを作成し、parameter-map webauth コンフィギュレーション モードを開始します。グローバル キーワードで定義されたグローバル パラメータ マップでサポートされる特定のコンフィギュレーション コマンドは、parameter-map-name 引数で定義された名前付きパラメータ マップでサポートされるコマンドとは異なります。
ステップ 4	**webauth-vrf-aware** 例： スイッチ (config-params-parameter-map)# **webauth-vrf-aware**	SVI で Web ベース認証の VRF 認識機能を有効にします。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。

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

Web ベース認証キャッシュ エントリの削除

Web ベース認証キャッシュ エントリを削除するには、次の手順を実行します。

手順の概要

1. **enable**
2. **clear ip auth-proxy cache** {* | *host ip address*}
3. **clear ip admission cache** {* | *host ip address*}

手順の詳細

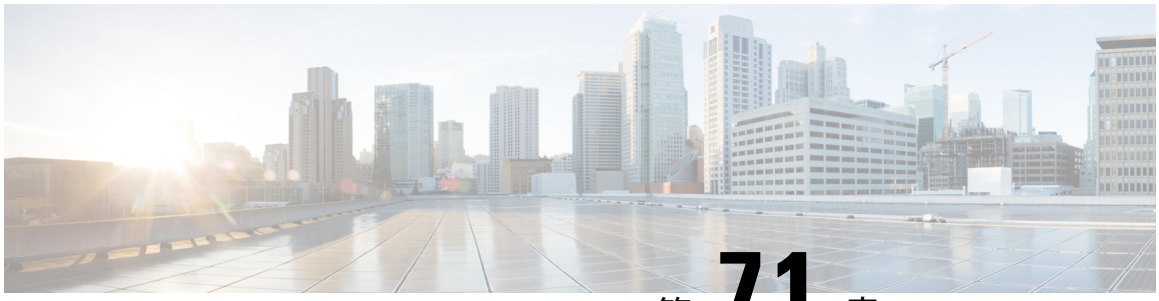
	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**clear ip auth-proxy cache** {*	*host ip address*} 例： スイッチ# **clear ip auth-proxy cache 192.168.4.5**	Delete 認証プロキシ エントリを削除します。キャッシュ エントリすべてを削除するには、アスタリスクを使用します。シングル ホストのエントリを削除するには、具体的な IP アドレスを入力します。
ステップ 3	**clear ip admission cache** {*	*host ip address*} 例： スイッチ# **clear ip admission cache 192.168.4.5**	Delete 認証プロキシ エントリを削除します。キャッシュ エントリすべてを削除するには、アスタリスクを使用します。シングル ホストのエントリを削除するには、具体的な IP アドレスを入力します。

Web ベース認証ステータスの確認

すべてのインターフェイスまたは特定のポートに対する Web ベース認証設定を表示するには、このトピックのコマンドを使用します。

表 *148 :* 特権 *EXEC* 表示コマンド

コマンド	目的
show authentication sessions method webauth	FastEthernet、ギガビットイーサネット、または10ギガビットイーサネットのすべてのインターフェイスに対する Web ベースの認証設定を表示します。
show wireless client mac-address *a.a.a* **detail**	セッション固有のワイヤレス情報とワイヤレス状態を表示します。
show authentication sessions interface *type slot/port*[**details**]	FastEthernet、ギガビットイーサネット、または10ギガビットイーサネットの特定のインターフェイスに対する Web ベースの認証設定を表示します。 セッション認識型ネットワーク モードでは、**show access-session interface** コマンドを使用します。



第 71 章

自動 ID

自動 ID 機能は、一連の組み込みポリシーをグローバル コンフィギュレーション モードとインターフェイス コンフィギュレーション モードで提供します。この機能は、Class-Based Policy Language（CPL）コントロール ポリシーと同等な新しいスタイルのモードでのみ使用できます。関連するすべての認証コマンドをそれらの CPL コントロールポリシーの同様のコマンドに変換するには、 **authentication convert-to new-style** コマンドを使用します。

このモジュールでは、その機能および設定方法について説明します。

自動 ID について

自動 ID の概要

Cisco Identity-Based Networking Services（IBNS）ソリューションは、エッジ デバイスが加入者に対して柔軟かつスケーラブルなサービスを提供できる、ポリシーとアイデンティティ ベースのフレームワークを提供します。IBNS では、IEEE 802.1x（dot1x）、MAC 認証バイパス（MAB）、および Web 認証方式を同時に実行することができます。これにより、1 つの加入者セッションに対して複数の認証方式を同時に呼び出すことができるようになります。これらの認証方式、dot1x、認証、認可、およびアカウンティング（AAA）、および RADIUS は、グローバル コンフィギュレーション モードとインターフェイス コンフィギュレーション モードで使用できます。

自動 ID 機能は、Cisco Common Classification Policy Language ベースの設定を使用します。これにより、認証方式とインターフェイス レベルのコマンドを設定するために使用するコマンドの数が大幅に削減されます。自動 ID 機能は、ポリシー マップ、クラス マップ、パラメータ マップ、およびインターフェイス テンプレートに基づいた一連の組み込みポリシーを提供します。

グローバル コンフィギュレーション モードでは、**source template AI_GLOBAL_CONFIG_TEMPLATE** コマンドで自動 ID 機能を有効にします。インターフェイス コンフィギュレーション モードでは、AI_MONITOR_MODE、AI_LOW_IMPACT_MODE、または AI_CLOSED_MODE インターフェイス テンプレートを設定し、インターフェイス上でこの機能を有効にします。

複数のテンプレートを設定できますが、**merge** コマンドを使用して、複数のテンプレートをまとめてバインドする必要があります。テンプレートをバインドしなかった場合は、最後に設定したテンプレートが使用されます。テンプレートをバインドする際に、2 つのテンプレートが異なる引数で繰り返された場合、最後に設定したコマンドが使用されます。

（注） また、グローバル コンフィギュレーション モードで **template** *name* コマンドを使用して設定したユーザー定義のテンプレートも有効にできます。

組み込みテンプレートに関する情報を表示するには、**show template interface** または **show template global** コマンドを使用します。組み込みテンプレートは編集できます。組み込みテンプレートを編集した場合は、**show running-config** コマンドの出力に、その組み込みテンプレートの情報が表示されます。編集した組み込みテンプレートを削除すると、その組み込みテンプレートはデフォルトに戻りますが、設定からは削除されません。ただし、ユーザー定義のテンプレートを削除した場合は設定から削除されます。

（注） テンプレートを削除する前に、デバイスに接続されていないことを確認します。

自動 ID グローバル テンプレート

グローバルテンプレートを有効にするには、**source template** *template-name* コマンドを設定します。

（注） RADIUS サーバー コマンドを設定する必要があります。これは、これらのコマンドはグローバル テンプレートが有効になっても、自動的に設定されないためです。

次に、グローバル テンプレートを有効にする例を示します。

```
Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE
Switch(config)# radius server ISE
Switch(config-radius-server)# address ipv4 172.20.254.4 auth-port 1645 acct-port 1646
Switch(config-radius-server)# key cisco
Switch(config-radius-server)# end
```

AI_GLOBAL_CONFIG_TEMPLATE は、次のコマンドを自動的に設定します。

```
dot1x system-auth-control
aaa new-model
aaa authentication dot1x default group radius
```

```
aaa authorization network default group radius
aaa authorization auth-proxy default group radius
aaa accounting identity default start-stop group radius
aaa accounting system default start-stop group radius
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 6 voice 1
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
```

自動 ID インターフェイス テンプレート

自動 ID 機能では、次のインターフェイス テンプレートを使用できます。

- AI_MONITOR_MODE：オープン モードで認証されているセッションを受動的に監視します。
- AI_LOW_IMPACT_MODE：モニター モードに似ていますが、ポート アクセス コントロール リスト（PACL）など、設定済みスタティック ポリシーを持ちます。
- AI_CLOSED_MODE：認証が完了するまで、データ トラフィックがネットワークに入ることを許可しないセキュア モードです。このモードがデフォルトです。

（注） マルチ認証ホストモードは、LAN Lite ライセンスではサポートされません。

次に、AI_MONITOR_MODE に組み込まれたコマンドを示します。

```
switchport mode access
 access-session port-control auto
 access-session host-mode multi-auth
 dot1x pae authenticator
 mab
 service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

次に、AI_LOW_IMPACT_MODE に組み込まれたコマンドを示します。

```
switchport mode access
 access-session port-control auto
 access-session host-mode multi-auth
 dot1x pae authenticator
 mab
 ip access-group AI_PORT_ACL in
 service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

次に、AI_CLOSED_MODE に組み込まれたコマンドを示します。

```
switchport mode access
 access-session closed
 access-session port-control auto
 access-session host-mode multi-auth
 dot1x pae authenticator
 mab
 service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

自動 ID 組み込みポリシー

自動 ID 機能では、次の 5 つの組み込みポリシーを使用できます。

- AI_DOT1X_MAB_AUTH：dot1x を使用してフレキシブル認証を有効にしてから、MAC アドレス バイパス（MAB）を有効にします。
- AI_DOT1X_MAB_POLICIES：dot1x を使用してフレキシブル認証を有効にしてから、MAB を有効にします。認証、認可、およびアカウンティング（AAA）サーバーに到達できない場合は、クリティカル VLAN を適用します。
- AI_DOT1X_MAB_WEBAUTH：dot1x を使用してフレキシブル認証を有効にしてから、Web 認証を有効にします。
- AI_NEXTGEN_AUTHBYBASS：IP 電話デバイスが検出された場合は認証をスキップします。デバイスを検出するには、**device classifier** コマンドをグローバル コンフィギュレーション モードで、**voice-vlan** コマンドをインターフェイス コンフィギュレーション モードで有効にします。これは参照ポリシー マップであり、ユーザーはこのポリシー マップの内容を別のポリシー マップにコピーできます。
- AI_STANDALONE_WEBAUTH：スタンドアロン Web 認証を定義します。

自動 ID クラス マップ テンプレート

次に、自動 ID 機能でサポートされている組み込みクラス マップを示します。

- AI_NRH：非応答ホスト（NRH）認証方式が有効であることを指定します。
- AI_WEBAUTH_METHOD：Web 認証方式が有効であることを指定します。
- AI_WEBAUTH_FAILED：Web 認証方式が認証に失敗したことを指定します。
- AI_WEBAUTH_NO_RESP：Web 認証クライアントが応答に失敗したことを指定します。
- AI_DOT1X_METHOD-dot1x：dot1x 方式が有効であることを指定します。
- AI_DOT1X_FAILED-dot1x：dot1x 方式が認証に失敗したことを指定します。
- AI_DOT1X_NO_RESP-dot1x：dot1x クライアントが応答に失敗したことを指定します。
- AI_DOT1X_TIMEOUT-dot1x：dot1x クライアントが最初の確認応答（ACK）要求後に応答を停止したことを指定します。
- AI_MAB_METHOD：MAC 認証バイパス（MAB）方式が有効であることを指定します。
- AI_MAB_FAILED-MAB：MAB 方式が認証に失敗したことを指定します。
- AI_AAA_SVR_DOWN_AUTHD_HOST：認証、認可、およびアカウンティング（AAA）サーバーがダウンし、クライアントが認可済みの状態になっていることを指定します。
- AI_AAA_SVR_DOWN_UNAUTHD_HOST-AAA：AAA サーバーがダウンし、クライアントが認可済みの状態になっていることを指定します。
- AI_IN_CRITICAL_AUTH：クリティカルな認証サービス テンプレートが適用されていることを指定します。
- AI_NOT_IN_CRITICAL_AUTH：クリティカルな認証サービス テンプレートが適用されていないことを指定します。
- AI_METHOD_DOT1X_DEVICE_PHONE：方式は dot1x であり、デバイス タイプが IP フォンであることを指定します。
- AI_DEVICE_PHONE：デバイス タイプが IP フォンであることを指定します。

自動 ID パラメータ マップ

次に、自動 ID 機能でサポートされている組み込みパラメータ マップ テンプレートを示します。

- AI_NRH_PMAP：非応答ホスト（NRH）認証を開始します。

  AI_WEBAUTH_PMAP：Web 認証を開始します。

自動 ID サービス テンプレート

サービス テンプレートは、組み込みポリシー マップ内で使用できます。次に、自動 ID 機能でサポートされている組み込みサービス テンプレートを示します。

- AI_INACTIVE_TIMER：非アクティビティ タイマーを起動するテンプレートです。
- AI_CRITICAL_ACL：ダミーテンプレートです。ユーザーはこのテンプレートを自分の要件に従って設定できます。

自動 ID の設定方法

自動 ID のグローバル設定

手順の概要

1. **enable**
2. **configure terminal**
3. **sourcetemplate** {**AI_GLOBAL_CONFIG_TEMPLATE** | *template-name*}
4. **aaa new-model**
5. **radius server** *name*
6. **address ipv4** {*hostname* | *ipv4-address*}
7. **key ipv4** { **0** *string* | **7** *string*} *string*
8. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**sourcetemplate** {**AI_GLOBAL_CONFIG_TEMPLATE** | *template-name*} 例： `Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE`	自動 ID テンプレートを設定します。 • **AI_GLOBAL_CONFIG_TEMPLATE** は組み込みのテンプレートです。 • *template-name* はユーザー定義のテンプレートです。
ステップ 4	**aaa new-model** 例： `Switch(config)# aaa new-model`	認証、認可、およびアカウンティング（AAA）アクセス コントロール モードを有効にします。
ステップ 5	**radius server** *name* 例： `Switch(config)# radius server ISE`	RADIUS サーバーの設定の名前を Protected Access Credential（PAC）のプロビジョニング用に指定し、RADIUS サーバー設定モードを開始します。
ステップ 6	**address ipv4** {*hostname* | *ipv4-address*} 例： `Switch(config-radius-server)# address ipv4 10.1.1.1`	RADIUS サーバーのアカウンティングおよび認証パラメータの IPv4 アドレスを設定します。 （注） このコマンドはグローバル テンプレートの一部ではないため、設定する必要があります。
ステップ 7	**key ipv4** { **0** *string* | **7** *string*} *string* 例： `Switch(config-radius-server)# key ipv4 cisco`	デバイスと RADIUS サーバーとの間におけるすべての RADIUS 通信用の認証および暗号キーを指定します。 （注） このコマンドはグローバル テンプレートの一部ではないため、設定する必要があります。
ステップ 8	**end** 例： `Switch(config-radius-server)# end`	RADIUS サーバ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

インターフェイス レベルでの自動 ID の設定

2 つのインターフェイス テンプレートを設定する場合は、**merge** キーワードを設定する必要があります。このキーワードを設定しない場合、最後に設定したテンプレートが使用されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*

4. **source template** {**AI_CLOSED_MODE** | **AI_LOW_IMPACT_MODE** | **AI_MONITOR_MODE** | *template-name*} [**merge**]
5. **source template** {**AI_CLOSED_MODE** | **AI_LOW_IMPACT_MODE** | **AI_MONITOR_MODE** | *template-name*} [**merge**]
6. **switchport access vlan** *vlan-id*
7. **switchport voice vlan** *vlan-id*
8. 自動 ID 機能を設定する必要があるすべてのインターフェイスで手順 4、6、および 7 を繰り返します。
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Switch(config)# interface gigabitethernet 1/0/1`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**source template** {**AI_CLOSED_MODE**	**AI_LOW_IMPACT_MODE**	**AI_MONITOR_MODE**	*template-name*} [**merge**] 例： `Switch(config-if)# source template AI_CLOSED_MODE`	インターフェイスにソース テンプレートを設定します。
ステップ 5	**source template** {**AI_CLOSED_MODE**	**AI_LOW_IMPACT_MODE**	**AI_MONITOR_MODE**	*template-name*} [**merge**] 例： `Switch(config-if)# source template AI_MONITOR_MODE merge`	（任意）インターフェイスのソース テンプレートを設定し、このテンプレートを以前に設定したテンプレートとマージします。 • 2 つのテンプレートを設定したときに **merge** キーワードを設定しなかった場合は、最後に設定したテンプレートが使用されます。
ステップ 6	**switchport access vlan** *vlan-id* 例： `Switch(config-if)# switchport access vlan 100`	インターフェイスがアクセス モードのときに VLAN を設定します。
ステップ 7	**switchport voice vlan** *vlan-id* 例： `Switch(config-if)# switchport voice vlan 101`	複数の VLAN アクセス ポートで音声 VLAN を設定します。

	コマンドまたはアクション	目的
ステップ 8	自動ID機能を設定する必要があるすべてのインターフェイスで手順 4、6、および 7 を繰り返します。	—
ステップ 9	**end** 例： `Switch(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

自動 ID の設定例

例：自動 ID のグローバル設定

```
Switch> enable
Switch# configure terminal
Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE
Switch(config)# aaa new-model
Switch(config)# radius server ISE
Switch(config-radius-server)# address ipv4 10.1.1.1
Switch(config-radius-server)# key ipv4 cisco
Switch(config-radius-server)# end
```

例：インターフェイス レベルでの自動 ID の設定

```
Switch> enable
Switch# configure terminal
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# source template AI_CLOSED_MODE
Switch(config-if)# source template AI_MONITOR_MODE merge
Switch(config-if)# switchport access vlan 100
Switch(config-if)# switchport voice vlan 101
Switch(config-if)# end
```

自動 ID の確認

ステップ 1 **enable**

例：

```
Switch> enable
```

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

ステップ 2　show template interface source built-in all

すべての設定済みの組み込みインターフェイス テンプレートを表示します。

例：

```
Switch# show template interface source built-in all

Template Name       : AI_CLOSED_MODE
Modified            : No
Template Definition :
 dot1x pae authenticator
 switchport mode access
 mab
 access-session closed
 access-session port-control auto
 service-policy type control subscriber AI_DOT1X_MAB_POLICIES
!
Template Name       : AI_LOW_IMPACT_MODE
Modified            : No
Template Definition :
 dot1x pae authenticator
 switchport mode access
 mab
 access-session port-control auto
 service-policy type control subscriber AI_DOT1X_MAB_POLICIES
 ip access-group AI_PORT_ACL in
!
Template Name       : AI_MONITOR_MODE
Modified            : No
Template Definition :
 dot1x pae authenticator
 switchport mode access
 mab
 access-session port-control auto
 service-policy type control subscriber AI_DOT1X_MAB_POLICIES
!
```

ステップ 3　show template global source built-in all

すべての設定済みのグローバル組み込みテンプレートを表示します。

例：

```
Switch# show template global source built-in all

Global Template Name       : AI_GLOBAL_CONFIG_TEMPLATE
Modified                   : No
Global Template Definition : global
 dot1x system-auth-control
 aaa new-model
 aaa authentication dot1x default group radius
 aaa authorization network default group radius
 aaa authorization auth-proxy default group radius
 aaa accounting identity default start-stop group radius
 aaa accounting system default start-stop group radius
 radius-server attribute 6 on-for-login-auth
 radius-server attribute 6 support-multiple
 radius-server attribute 6 voice 1
 radius-server attribute 8 include-in-access-req
 radius-server attribute 25 access-request include
```

```
!
```

ステップ 4 **show derived-config | include aaa |radius-server**

インターフェイスに適用されているすべてのコンフィギュレーション コマンドの複合された結果を表示します。これには、スタティック テンプレート、ダイナミック テンプレート、ダイヤライン インターフェイス、ならびに認証、認可、およびアカウンティング（AAA）のユーザー単位の属性など、送信元からのコマンドが含まれます。

例：

```
Switch# show derived-config | inc aaa| radius-server

aaa new-model
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa authorization auth-proxy default group radius
aaa accounting identity default start-stop group radius
aaa accounting system default start-stop group radius
aaa session-id common
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 6 voice 1
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
radius-server host 10.25.18.42 key cisco
```

ステップ 5 **show derived-config | interface** *type-number*

インターフェイスのすべての設定の複合された結果を表示します。

例：

```
Switch# show derived-config | interface gigabitethernet2/0/6

Building configuration...

Derived configuration : 267 bytes
!
interface GigabitEthernet2/0/6
 switchport mode access
 switchport voice vlan 100
 access-session closed
 access-session port-control auto
 mab
 dot1x pae authenticator
 spanning-tree portfast edge
 service-policy type control subscriber AI_DOT1X_MAB_POLICIES
end
```

ステップ 6 **show access-session | interface** *interface-type-number* **details**

インターフェイスに適用されているポリシーを表示します。

例：

```
Switch# show access-session interface gigabitethernet2/0/6 details
```

```
Interface:  GigabitEthernet2/0/6
          MAC Address:  c025.5c43.be00
         IPv6 Address:  Unknown
         IPv4 Address:  Unknown
            User-Name:  CP-9971-SEPC0255C43BE00
          Device-type:  Cisco-IP-Phone-9971
               Status:  Authorized
               Domain:  VOICE
       Oper host mode:  multi-auth
     Oper control dir:  both
      Session timeout:  N/A
    Common Session ID:  091A1C5B00000017002003EE
      Acct Session ID:  0x00000005
               Handle:  0xBB00000B
       Current Policy:  AI_DOT1X_MAB_POLICIES

Local Policies:

Server Policies:
           Vlan Group:  Vlan: 100
      Security Policy:  Must Not Secure
      Security Status:  Link Unsecure


Method status list:
       Method           State
       dot1x            Authc Success
```

ステップ 7 show running-config interface *type-number*

現在の実行コンフィギュレーション ファイルまたはインターフェイスの設定を表示します。

例：

```
Switch# show running-config interface gigabitethernet2/0/6

Building configuration...

Current configuration : 214 bytes
!
interface GigabitEthernet2/0/6
 switchport mode access
 switchport voice vlan 100
 access-session port-control auto
 spanning-tree portfast edge
 service-policy type control subscriber AI_NEXTGEN_AUTHBYPASS
end
```

ステップ 8 show lldp neighbor

Link Layer Discovery Protocol（LLDP）を使用して検出した 1 つまたはすべてのネイバー デバイスの情報を表示します。

例：

```
Switch# show lldp neighbor

Capability codes:
    (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
    (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
```

```
Device ID           Local Intf     Hold-time  Capability      Port ID
SEPC0255C43BE00     Gi2/0/6        180        B,T             C0255C43BE00:P1

Total entries displayed: 1
```

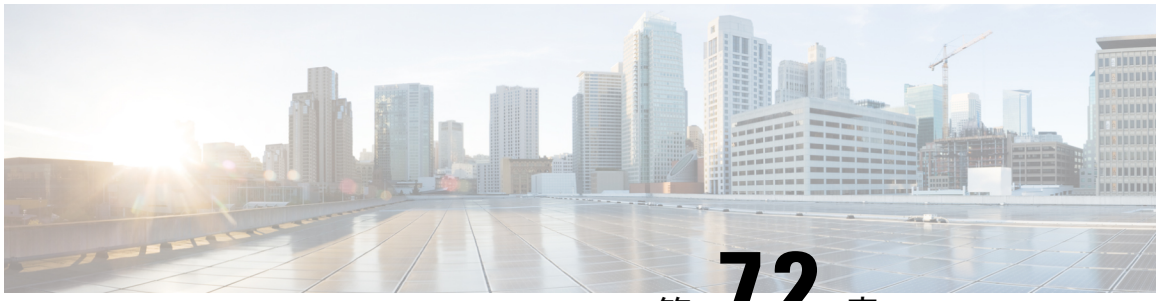
自動 ID の機能情報

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

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

表 *149* : 自動 *ID* の機能情報

機能名	リリース	機能情報
自動 ID	Cisco IOS リリース 15.2(4)E	自動ID機能は、一連の組み込みポリシーをグローバル コンフィギュレーション モードとインターフェイス コンフィギュレーション モードで提供します。この機能は、Class-Based Policy Language（CPL）コントロール ポリシーと同等な新しいスタイルのモードでのみ使用できます。 この機能は、Cisco IOS リリース 15.2(4)E で Cisco Catalyst 2960–X シリーズ スイッチ、Catalyst 3750–X シリーズ スイッチ、および Cisco Catalyst 4500E Supervisor Engine 7-E に実装されました。 次のコマンドが導入または変更されました。**source-template**



第 72 章

ポート単位のトラフィック制御の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

ストーム制御に関する情報

ストーム制御

ストーム制御は、物理インターフェイスの 1 つで発生したブロードキャスト、マルチキャスト、またはユニキャスト ストームによって LAN 上のトラフィックが混乱することを防ぎます。LAN ストームは、LAN にパケットがフラッディングした場合に発生します。その結果、トラフィックが極端に増えてネットワーク パフォーマンスが低下します。プロトコルスタックの実装エラー、ネットワーク構成の間違い、またはユーザによって引き起こされる DoS 攻撃もストームの原因になります。

ストーム コントロール（またはトラフィック抑制）は、インターフェイスからスイッチング バスを通過するパケットをモニタし、パケットがユニキャスト、マルチキャスト、またはブロードキャストのいずれであるかを判別します。スイッチは、1 秒間に受け取った特定のタイプのパケットの数をカウントして、事前に定義された抑制レベルのしきい値とその測定結果を比較します。

トラフィック アクティビティの測定方法

ストーム コントロールは、次のうちのいずれかをトラフィック アクティビティの測定方法に使用します。

- 帯域幅（ブロードキャスト、マルチキャスト、またはユニキャスト トラフィックが使用できるポートの総帯域幅の割合）。
- 秒単位で受信するパケット（ブロードキャスト、マルチキャスト、またはユニキャスト）のトラフィック レート
- 秒単位で受信するビット（ブロードキャスト、マルチキャスト、またはユニキャスト）のトラフィック レート

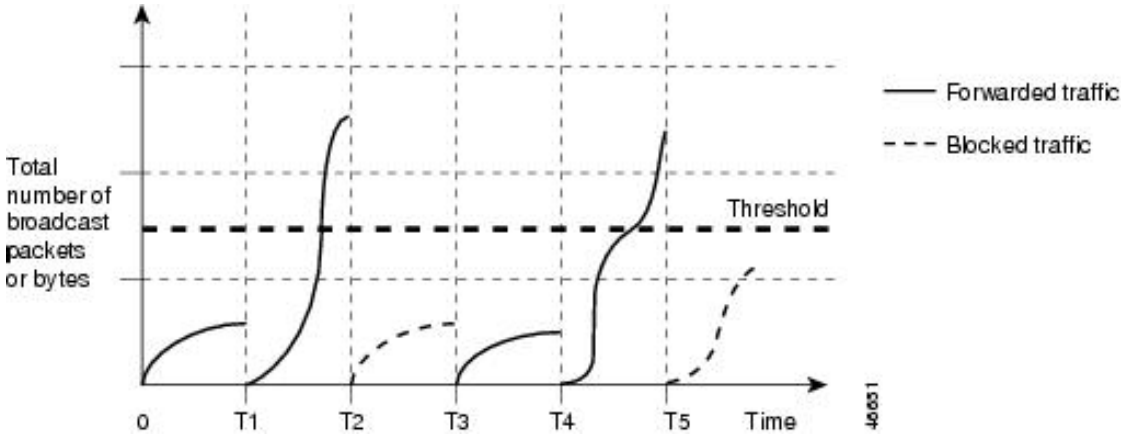
上記の方法のいずれを使用しても、しきい値に到達すると、ポートはトラフィックをブロックします。トラフィック レートが下限しきい値（指定されている場合）を下回らない限り、ポートはブロックされたままになり、その後、通常の転送が再開されます。下限抑制レベルが指定されていない場合、トラフィック レートが上限抑制レベルを下回らない限り、スイッチはすべてのトラフィックをブロックします。一般に、そのレベルが高ければ高いほど、ブロードキャスト ストームに対する保護効果は薄くなります。

（注） マルチキャストトラフィックのストーム制御しきい値に達した場合、ブリッジ プロトコル データ ユニット（BPDU）および Cisco Discovery Protocol フレームなどの制御トラフィック以外のマルチキャストトラフィックはすべてブロックされます。ただし、スイッチでは Open Shortest Path First（OSPF）などのルーティング アップデートと、正規のマルチキャスト データ トラフィックは区別されないため、両方のトラフィック タイプがブロックされます。

トラフィック パターン

図 *119:* ブロードキャスト ストーム制御の例

次の例は、一定時間におけるインターフェイス上のブロードキャスト トラフィック パターンを示しています。



T1 から T2、T4 から T5 のタイム インターバルで、転送するブロードキャスト トラフィックが設定されたしきい値を上回っています。指定のトラフィック量がしきい値を上回ると、次のインターバルで、そのタイプのトラフィックがすべてドロップされます。したがって、T2 と T5 の後のインターバルの間、ブロードキャスト トラフィックがブロックされます。その次のインターバル（たとえば、T3）では、しきい値を上回らない限り、ブロードキャスト トラフィックが再び転送されます。

ストーム制御抑制レベルと 1 秒間のインターバルを組み合わせて、ストーム制御アルゴリズムの動作を制御します。しきい値が高いほど、通過できるパケット数が多くなります。しきい値が 100% であれば、トラフィックに対する制限はありません。値を 0.0 にすると、そのポート上ではすべてのブロードキャスト、マルチキャスト、またはユニキャスト トラフィックがブロックされます。

（注） パケットは一定の間隔で届くわけではないので、トラフィック アクティビティを測定する 1 秒間のインターバルがストーム制御の動作を左右する可能性があります。

各トラフィックタイプのしきい値を設定するには、**storm-control** インターフェイス コンフィギュレーション コマンドを使用します。

ストーム制御の設定方法

ストーム制御およびしきい値レベルの設定

ポートにストーム制御を設定し、特定のトラフィック タイプで使用するしきい値レベルを入力します。

ただし、ハードウェアの制約とともに、さまざまなサイズのパケットをどのように数えるかという問題があるので、しきい値の割合はあくまでも近似値です。着信トラフィックを形成するパケットのサイズによって、実際に適用されるしきい値は設定されたレベルに対して、数 % の差異が生じる可能性があります。

（注） ストーム制御は、物理インターフェイスでサポートされています。また、EtherChannel でもストーム制御を設定できます。ストーム制御を EtherChannel で設定する場合、ストーム制御設定は EtherChannel 物理インターフェイスに伝播します。

ストーム制御としきい値レベルを設定するには、次の手順を実行します。

始める前に

ストーム制御は、物理インターフェイスでサポートされています。また、EtherChannel でもストーム制御を設定できます。ストーム制御を EtherChannel で設定する場合、ストーム制御設定は EtherChannel 物理インターフェイスに伝播します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}
5. **storm-control action** {**shutdown** | **trap**}
6. **end**
7. **show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]} 例： スイッチ(config-if)# **storm-control unicast level 87 65**	ブロードキャスト、マルチキャスト、またはユニキャスト ストーム制御を設定します。デフォルトでは、ストーム制御はディセーブルに設定されています。 キーワードの意味は次のとおりです。 • *level* には、ブロードキャスト、マルチキャスト、またはユニキャスト トラフィックの上限しきい値レベルを帯域幅のパーセンテージで指定します（小数点第 2 位まで）。上限しきい値に到達すると、ポートはトラフィックをブロックします。指定できる範囲は 0.00 ～ 100.00 です。 • （任意）*level-low* には、下限しきい値レベルを帯域幅のパーセンテージで指定します（小数点第 2 位まで）。この値は上限抑制値より小さいか、または等しくなければなりません。トラフィックがこのレベルを下回っていれば、ポートはトラフィックを転送します。下限抑制レベルを設定しない場合、上限抑制レベルの値に設定されます。指定できる範囲は 0.00 ～ 100.00 です。 しきい値に最大値（100%）を指定した場合、トラフィックの制限はなくなります。しきい値に 0.0 を設定すると、そのポート上のすべてのブ

	コマンドまたはアクション	目的
		ロードキャスト、マルチキャスト、またはユニキャスト トラフィックがブロックされます。 • **bps** *bps* には、ブロードキャスト、マルチキャスト、またはユニキャストトラフィックの上限しきい値レベルをビット/秒で指定します（小数点第 1 位まで）。上限しきい値に到達すると、ポートはトラフィックをブロックします。指定できる範囲は 0.0 ～ 10000000000.0 です。 •（任意）*bps-low* には、下限しきい値レベルをビット/秒で指定します（小数点第1位まで）。この値は上限しきい値レベル以下の値である必要があります。トラフィックがこのレベルを下回っていれば、ポートはトラフィックを転送します。指定できる範囲は 0.0 ～ 10000000000.0 です。 • **pps** *pps* には、ブロードキャスト、マルチキャスト、またはユニキャストトラフィックの上限しきい値レベルをパケット/秒で指定します（小数点第 1 位まで）。上限しきい値に到達すると、ポートはトラフィックをブロックします。指定できる範囲は 0.0 ～ 10000000000.0 です。 •（任意）*pps-low* には、下限しきい値レベルをパケット/秒で指定します（小数点第1位まで）。この値は上限しきい値レベル以下の値である必要があります。トラフィックがこのレベルを下回っていれば、ポートはトラフィックを転送します。指定できる範囲は **0.0 ～** 10000000000.0 です。 BPS および PPS の設定には、しきい値の数値を大きく設定できるように、サフィックスに測定記号（k、m、g など）を使用できます。
ステップ 5	**storm-control action** {**shutdown** | **trap**} 例： `スイッチ(config-if)# storm-control action trap`	ストーム検出時に実行するアクションを指定します。デフォルトではトラフィックにフィルタリングを実行し、トラップは送信しない設定です。 • ストーム中、ポートを errdisable の状態にするには、**shutdown** キーワードを選択します。 • ストームが検出された場合、SNMP トラップを生成するには、**trap** キーワードを選択します。

	コマンドまたはアクション	目的
ステップ 6	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**] 例： スイッチ# **show storm-control gigabitethernet1/0/1 unicast**	指定したトラフィック タイプについて、インターフェイスで設定したストーム制御抑制レベルを確認します。トラフィック タイプを入力しない場合は、すべてのトラフィック タイプ（ブロードキャスト、マルチキャスト、ユニキャスト）の詳細が表示されます。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

スモール フレーム到着レートの設定

67バイト未満の着信VLANタグ付きパケットは、小さいフレームと見なされます。このパケットはスイッチにより転送されますが、スイッチ ストーム制御カウンタを増加させません。

スイッチ上の小さいフレームの着信機能をグローバルにイネーブルにして、各インターフェイスのパケットの小さいフレームのしきい値を設定します。最小サイズよりも小さく、指定されたレート（しきい値）で着信するパケットは、ポートがディセーブルにされた後はドロップされます。

手順の概要

1. **enable**
2. **configure terminal**
3. **errdisable detect cause small-frame**
4. **errdisable recovery interval** *interval*
5. **errdisable recovery cause small-frame**
6. **interface** *interface-id*
7. **small-frame violation-rate** *pps*
8. **end**
9. **show interfaces** *interface-id*
10. **show running-config**
11. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**errdisable detect cause small-frame** 例： スイッチ(config)# **errdisable detect cause small-frame**	スイッチ上の小さいフレームの着信レート機能をイネーブルにします。
ステップ 4	**errdisable recovery interval** *interval* 例： スイッチ(config)# **errdisable recovery interval 60**	（任意）指定された errdisable ステートから回復する時間を指定します。
ステップ 5	**errdisable recovery cause small-frame** 例： スイッチ(config)# **errdisable recovery cause small-frame**	（任意）小さいフレームの着信によりポートが errdisable になった後、そのポートを自動的に再イネーブルにするリカバリ時間を設定します。 ストーム制御は、物理インターフェイスでサポートされています。また、EtherChannel でもストーム制御を設定できます。ストーム制御を EtherChannel で設定する場合、ストーム制御設定は EtherChannel 物理インターフェイスに伝播します。
ステップ 6	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/2**	インターフェイス コンフィギュレーション モードを開始し、設定するインターフェイスを指定します。
ステップ 7	**small-frame violation-rate** *pps* 例：	インターフェイスが着信パケットをドロップしてポートを errdisable にするようにしきい値レートを設定します。範囲は、1 ～ 10,000 パケット/秒（pps）です。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **small-frame violation rate 10000**	
ステップ 8	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 9	**show interfaces** *interface-id* 例： スイッチ# **show interfaces gigabitethernet1/0/2**	設定を確認します。
ステップ 10	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

保護ポートに関する情報

保護ポート

アプリケーションによっては、あるネイバーが生成したトラフィックが別のネイバーにわからないように、同一スイッチ上のポート間でレイヤ2トラフィックが転送されないように設定する必要があります。このような環境では、保護ポートを使用すると、スイッチ上のポート間でユニキャスト、ブロードキャスト、またはマルチキャストトラフィックの交換が確実になくなります。

保護ポートには、次の機能があります。

- 保護ポートは、同様に保護ポートになっている他のポートに対して、ユニキャスト、マルチキャスト、またはブロードキャスト トラフィックを転送しません。データ トラフィックはレイヤ 2 の保護ポート間で転送されません。PIM パケットなどは CPU で処理されてソフトウェアで転送されるため、このような制御トラフィックだけが転送されます。保護

ポート間を通過するすべてのデータ トラフィックは、レイヤ3デバイスを介して転送されなければなりません。

- 保護ポートと非保護ポート間の転送動作は、通常どおりに進みます。

スイッチ スタックは論理的には1つのスイッチを表しているため、レイヤ2トラフィックは、スタック内の同一スイッチか異なるスイッチかにかかわらず、スイッチ スタックの保護ポート間では転送されません。

保護ポートのデフォルト設定

デフォルトでは、保護ポートは定義されません。

保護ポートのガイドライン

保護ポートは、物理インターフェイス（GigabitEthernet ポート 1 など）または EtherChannel グループ（port-channel 5 など）に設定できます。ポート チャネルで保護ポートをイネーブルにした場合は、そのポート チャネル グループ内のすべてのポートでイネーブルになります。

保護ポートの設定方法

保護ポートの設定

始める前に

保護ポートは事前定義されていません。これは設定する必要があるタスクです。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport protected**
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport protected** 例： スイッチ(config-if)# **switchport protected**	インターフェイスを保護ポートとして設定します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show interfaces** *interface-id* **switchport** 例： スイッチ# **show interfaces gigabitethernet 1/0/1 switchport**	入力を確認します。
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

保護ポートの監視

表 *150 :* 保護ポートの設定を表示するコマンド

コマンド	目的
show interfaces [*interface-id*] **switchport**	すべてのスイッチング（非ルーティング）ポートまたは指定されたポートの管理ステータスまたは動作ステータスを、ポート ブロッキングおよびポート保護の設定を含めて表示します。

次の作業

-

ポート ブロッキングに関する情報

ポート ブロッキング

デフォルトでは、スイッチは未知の宛先 MAC アドレスが指定されたパケットをすべてのポートからフラッディングします。未知のユニキャストおよびマルチキャスト トラフィックが保護ポートに転送されると、セキュリティ上、問題になる可能性があります。未知のユニキャストおよびマルチキャスト トラフィックがあるポートから別のポートに転送されないようにするために、（保護または非保護）ポートをブロックし、未知のユニキャストまたはマルチキャスト パケットが他のポートにフラッディングされないようにします。

（注） マルチキャスト トラフィックでは、ポート ブロッキング機能は純粋なレイヤ 2 パケットだけをブロックします。ヘッダーに IPv4 または IPv6 の情報を含むマルチキャスト パケットはブロックされません。

ポート ブロッキングの設定方法

インターフェイスでのフラッディング トラフィックのブロッキング

始める前に

インターフェイスは物理インターフェイスまたはEtherChannelグループのいずれも可能です。ポート チャネルのマルチキャストまたはユニキャスト トラフィックをブロックすると、ポート チャネル グループのすべてのポートでブロックされます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport block multicast**
5. **switchport block unicast**
6. **end**
7. **show interfaces** *interface-id* **switchport**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**switchport block multicast** 例： `スイッチ(config-if)# switchport block multicast`	ポートからの未知のマルチキャストの転送をブロックします。
ステップ 5	**switchport block unicast** 例： `スイッチ(config-if)# switchport block unicast`	ポートからの未知のユニキャストの転送をブロックします。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show interfaces** *interface-id* **switchport** 例： `スイッチ# show interfaces gigabitethernet 1/0/1 switchport`	入力を確認します。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ポート ブロッキングの監視

表 *151 :* ポート ブロッキングの設定を表示するコマンド

コマンド	目的
show interfaces [*interface-id*] **switchport**	すべてのスイッチング（非ルーティング）ポートまたは指定されたポートの管理ステータスまたは動作ステータスを、ポート ブロッキングおよびポート保護の設定を含めて表示します。

ポート セキュリティの前提条件

（注） 最大値をインターフェイス上ですでに設定されているセキュア アドレスの数より小さい値に設定しようとすると、コマンドが拒否されます。

ポート セキュリティの制約事項

- スイッチまたはスイッチ スタックに設定できるセキュア MAC アドレスの最大数は、システムで許可されている MAC アドレスの最大数によって決まります。この数字はアクティブな Switch Database Management（SDM）テンプレートによって決められます。この値は、使用可能な MAC アドレス（その他のレイヤ 2 機能やインターフェイスに設定されたその他のセキュア MAC アドレスで使用される MAC アドレスを含む）の総数を表します。

ポート セキュリティの概要

ポート セキュリティ

ポート セキュリティ機能を使用すると、ポートへのアクセスを許可するステーションの MAC アドレスを制限および識別して、インターフェイスへの入力を制限できます。セキュア ポートにセキュア MAC アドレスを割り当てると、ポートは定義されたアドレス グループ以外の送信元アドレスを持つパケットを転送しません。セキュア MAC アドレス数を 1 つに制限し、単一のセキュア MAC アドレスを割り当てると、そのポートに接続されたワークステーションに、ポートの帯域幅全体が保証されます。

セキュア ポートとしてポートを設定し、セキュア MAC アドレスが最大数に達した場合、ポートにアクセスを試みるステーションの MAC アドレスが識別されたセキュア MAC アドレスのいずれとも一致しないので、セキュリティ違反が発生します。また、あるセキュア ポート上でセキュア MAC アドレスが設定または学習されているステーションが、別のセキュア ポートにアクセスしようとしたときにも、違反のフラグが立てられます。

セキュア MAC アドレスのタイプ

スイッチは、次のセキュア MAC アドレス タイプをサポートします。

- スタティック セキュア MAC アドレス：**switchport port-security mac-address** *mac-address* インターフェイス コンフィギュレーション コマンドを使用して手動で設定され、アドレス テーブルに保存された後、スイッチの実行コンフィギュレーションに追加されます。
- ダイナミック セキュア MAC アドレス：動的に設定されてアドレス テーブルにのみ保存され、スイッチの再起動時に削除されます。
- スティッキー セキュア MAC アドレス：動的に学習することも、手動で設定することもできます。アドレス テーブルに保存され、実行コンフィギュレーションに追加されます。このアドレスがコンフィギュレーション ファイルに保存されていると、スイッチの再起動時にインターフェイスはこれらを動的に再設定する必要がありません。

スティッキ セキュア MAC アドレス

スティッキー ラーニングをイネーブルにすると、ダイナミック MAC アドレスをスティッキー セキュア MAC アドレスに変換して実行コンフィギュレーションに追加するようにインターフェイスを設定できます。インターフェイスはスティッキ ラーニングがイネーブルになる前に学習したものを含め、すべてのダイナミック セキュア MAC アドレスをスティッキー セキュア MAC アドレスに変換します。すべてのスティッキー セキュア MAC アドレスは実行コンフィギュレーションに追加されます。

スティッキー セキュア MAC アドレスは、コンフィギュレーション ファイル（スイッチが再起動されるたびに使用されるスタートアップ コンフィギュレーション）に、自動的には反映されません。スティッキー セキュア MAC アドレスをコンフィギュレーション ファイルに保存すると、スイッチの再起動時にインターフェイスはこれらを再び学習する必要がありません。スティッキ セキュア アドレスを保存しない場合、アドレスは失われます。

スティッキ ラーニングがディセーブルの場合、スティッキ セキュア MAC アドレスはダイナミック セキュア アドレスに変換され、実行コンフィギュレーションから削除されます。

セキュリティ違反

次のいずれかの状況が発生すると、セキュリティ違反になります。

- 最大数のセキュア MAC アドレスがアドレス テーブルに追加されている状態で、アドレス テーブルに未登録の MAC アドレスを持つステーションがインターフェイスにアクセスしようとした場合。

- あるセキュア インターフェイスで学習または設定されたアドレスが、同一 VLAN 内の別のセキュア インターフェイスで使用された場合。
- ポートセキュリティが有効な状態で診断テストを実行しています。

違反が発生した場合の対処に基づいて、次の3種類の違反モードのいずれかにインターフェイスを設定できます。

- protect（保護）：セキュア MAC アドレスの数がポートで許可されている最大限度に達すると、最大値を下回るまで十分な数のセキュア MAC アドレスを削除するか、許可アドレス数を増やさないかぎり、未知の送信元アドレスを持つパケットはドロップされます。セキュリティ違反が起こっても、ユーザには通知されません。

（注） トランク ポートに protect 違反モードを設定することは推奨しません。保護モードでは、ポートが最大数に達していなくても VLAN が保護モードの最大数に達すると、ラーニングがディセーブルになります。

- restrict（制限）：セキュア MAC アドレスの数がポートで許可されている最大限度に達すると、最大値を下回るまで十分な数のセキュア MAC アドレスを削除するか、許可アドレス数を増やさないかぎり、未知の送信元アドレスを持つパケットはドロップされます。このモードでは、セキュリティ違反が発生したことが通知されます。SNMP トラップが送信されます。Syslog メッセージがロギングされ、違反カウンタが増加します。
- shutdown（シャットダウン）：ポート セキュリティ違反により、インターフェイスが error-disabled になり、ただちにシャットダウンされます。そのあと、ポートの LED が消灯します。セキュアポートが error-disabled 状態の場合は、**errdisable recovery cause** *psecure-violation* グローバル コンフィギュレーション コマンドを入力してこの状態を解消するか、**shutdown** および **no shut down** インターフェイス コンフィギュレーション コマンドを入力して手動で再度有効にできます。これは、デフォルトのモードです。
- shutdown vlan（VLAN シャットダウン）：VLAN 単位でセキュリティ違反モードを設定するために使用します。このモードで違反が発生すると、ポート全体ではなく、VLAN が errdisable になります。

次の表に、ポートセキュリティをインターフェイスに設定した場合の違反モードおよび対処について示します。

表 *152*：セキュリティ違反モードの処置

違反モード	トラフィックの転送 19	SNMP トラップの送信	Syslog メッセージの送信	エラー メッセージの表示 20	違反カウンタの増加	ポートのシャットダウン
protect	非対応	非対応	非対応	非対応	非対応	非対応

違反モード	トラフィックの転送 [19]	SNMP トラップの送信	Syslog メッセージの送信	エラー メッセージの表示 [20]	違反カウンタの増加	ポートのシャットダウン
restrict	非対応	対応	対応	非対応	対応	非対応
shutdown	非対応	非対応	非対応	非対応	対応	対応
shutdown vlan	非対応	非対応	対応	非対応	対応	非対応 [21]

19 十分な数のセキュアMACアドレスを削除するまで未知の送信元アドレスを持つパケットがドロップされます。
20 セキュリティ違反を引き起こすアドレスを手動で設定した場合、スイッチがエラー メッセージを返します。
21 違反が発生した VLAN のみシャットダウンします。

ポート セキュリティ エージング

ポート上のすべてのセキュア アドレスにエージング タイムを設定するには、ポート セキュリティ エージングを使用します。ポートごとに2つのタイプのエージングがサポートされています。

- absolute：指定されたエージング タイムの経過後に、ポート上のセキュア アドレスが削除されます。
- inactivity：指定されたエージング タイムの間、セキュア アドレスが非アクティブであった場合に限り、ポート上のセキュア アドレスが削除されます。

デフォルトのポート セキュリティ設定

表 *153*：デフォルトのポート セキュリティ設定

機能	デフォルト設定
ポート セキュリティ	ポート上でディセーブル
スティッキー アドレス ラーニング	ディセーブル
ポートあたりのセキュア MAC アドレスの最大数	1。
違反モード	shutdown。セキュアMACアドレスが最大数を上回ると、ポートがシャットダウンします。

機能	デフォルト設定
ポート セキュリティ エージング	ディセーブルエージング タイムは 0 スタティック エージングはディセーブル タイプは absolute

ポート セキュリティの設定時の注意事項

- ポート セキュリティを設定できるのは、スタティック アクセス ポートまたはトランク ポートに限られます。セキュア ポートをダイナミック アクセス ポートにすることはできません。
- セキュア ポートをスイッチド ポート アナライザ（SPAN）の宛先ポートにすることはできません。
- 音声 VLAN はアクセス ポートでのみサポートされており、設定可能であってもトランク ポートではサポートされていません。
- 音声 VLAN が設定されたインターフェイス上でポート セキュリティをイネーブルにする場合は、ポートの最大セキュア アドレス許容数を 2 に設定します。ポートを Cisco IP Phone に接続する場合は、IP Phone に MAC アドレスが 1 つ必要です。Cisco IP Phone のアドレスは音声 VLAN 上で学習されますが、アクセス VLAN 上では学習されません。1 台の PC を Cisco IP Phone に接続する場合、MAC アドレスの追加は必要ありません。複数の PC を Cisco IP Phone に接続する場合、各 PC と IP Phone に 1 つずつ使用できるように、十分な数のセキュア アドレスを設定する必要があります。
- トランクポートがポートセキュリティで設定され、データトラフィック用のアクセスVLANと音声トラフィック用の音声 VLAN に割り当てられている場合、**switchport voice** およびインターフェイス コンフィギュレーション コマンドを入力して **switchport priority extend** も効果はありません。

  接続装置が同じ MAC アドレスを使用してアクセス VLAN の IP アドレス、音声 VLAN の IP アドレスの順に要求すると、アクセス VLAN だけが IP アドレスに割り当てられます。
- インターフェイスの最大セキュアアドレス値を入力したときに、新しい値がそれまでの値より大きいと、それまで設定されていた値が新しい値によって上書きされます。新しい値が前回の値より小さく、インターフェイスで設定されているセキュアアドレス数が新しい値より大きい場合、コマンドは拒否されます。
- スイッチはスティッキセキュアMACアドレスのポートセキュリティエージングをサポートしていません。

次の表に、他のポートベース機能と互換性のあるポート セキュリティについてまとめます。

表 *154* : ポート セキュリティと他のポートベース機能との互換性

ポート タイプまたはポートの機能	ポート セキュリティとの互換性
DTP [22] ポート [23]	なし
トランク ポート	あり
ダイナミックアクセスポート[24]	なし
ルーテッド ポート	なし
SPAN 送信元ポート	あり
SPAN 宛先ポート	なし
EtherChannel	対応
トンネリング ポート	あり
保護ポート	あり
IEEE 802.1x ポート	あり
音声 VLAN ポート[25]	あり
IP ソース ガード	あり
ダイナミック アドレス解決プロトコル（ARP）インスペクション	あり
Flex Link	対応

22 DTP = Dynamic Trunking Protocol

23 **switchport mode dynamic** インターフェイス コンフィギュレーション コマンドで設定されたポート A。

24 **switchport access vlan dynamic** インターフェイス コンフィギュレーション コマンドで設定される VLAN Query Protocol（VQP）ポート。

25 ポートに最大限可能なセキュアなアドレスを設定します（アクセス VLAN で可能なセキュアなアドレスの最大数に 2 を加えた数）。

ポート セキュリティの設定方法

ポート セキュリティのイネーブル化および設定

始める前に

このタスクは、ポートにアクセスできるステーションの MAC アドレスを制限および識別して、インターフェイスへの入力を制約します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode** {**access** | **trunk**}
5. **switchport voice vlan** *vlan-id*
6. **switchport port-security**
7. **switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]]
8. **switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**}
9. **switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}]
10. **switchport port-security mac-address sticky**
11. **switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}]
12. **end**
13. **show port-security**
14. **show running-config**
15. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/1`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport mode** {**access** | **trunk**} 例： `スイッチ(config-if)# switchport mode access`	インターフェイス スイッチポート モードを access または trunk に設定します。デフォルト モード（dynamic auto）のインターフェイスは、セキュア ポートとして設定できません。
ステップ 5	**switchport voice vlan** *vlan-id* 例： `スイッチ(config-if)# switchport voice vlan 22`	ポート上で音声 VLAN をイネーブルにします。 vlan-id：音声トラフィックに使用する VLAN を指定します。
ステップ 6	**switchport port-security** 例： `スイッチ(config-if)# switchport port-security`	インターフェイス上でポートセキュリティをイネーブルにします。 （注） 特定の条件下では、スイッチ スタックのメンバー ポートでポートセキュリティが有効になっていると、DHCP および ARP パケットがドロップされます。これを解決するには、インターフェイスで shut と no shut を設定します。
ステップ 7	**switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]] 例： `スイッチ(config-if)# switchport port-security maximum 20`	（任意）インターフェイスの最大セキュア MAC アドレス数を設定します。スイッチまたはスイッチ スタックに設定できるセキュア MAC アドレスの最大数は、システムで許可されている MAC アドレスの最大数によって決まります。この値は、アクティブな SDM テンプレートによって決まります。この値は、使用可能な MAC アドレス（その他のレイヤ 2 機能やインターフェイスに設定されたその他のセキュア MAC アドレスで使用される MAC アドレスを含む）の総数を表します。 （任意）**vlan**：VLAN 当たりの最大値を設定します。 **vlan** キーワードを入力後、次のいずれかのオプションを入力します。 • *vlan-list*：トランク ポート上で、ハイフンで区切った範囲の VLAN、またはカンマで区切っ

	コマンドまたはアクション	目的
		た一連の VLAN における、VLAN 単位の最大値を設定できます。VLANを指定しない場合、VLAN ごとの最大値が使用されます。 • **access**：アクセス ポートで、VLANをアクセス VLAN として指定します。 • **voice**：アクセスポートで、VLANを音声VLANとして指定します。 （注） **voice** キーワードは、音声 VLANがポートに設定されていて、さらにそのポートがアクセス VLAN でない場合のみ有効です。インターフェイスに音声 VLAN が設定されている場合、セキュア MAC アドレスの最大数を 2 に設定します。
ステップ 8	**switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**} 例： `スイッチ(config-if)# switchport port-security violation restrict`	（任意）違反モードを設定します。セキュリティ違反が発生した場合に、次のいずれかのアクションを実行します。 • **protect**：ポート セキュア MAC アドレスの数がポートで許可されている最大限度に達すると、最大値を下回るまで十分な数のセキュア MAC アドレスを削除するか、許可アドレス数を増やさない限り、未知の送信元アドレスを持つパケットはドロップされます。セキュリティ違反が起こっても、ユーザには通知されません。 （注） トランク ポート上に保護モードを設定することは推奨できません。保護モードでは、ポートが最大数に達していなくても VLAN が保護モードの最大数に達すると、ラーニングがディセーブルになります。 • **restrict**：セキュアMACアドレス数がポートで許可されている最大数に到達した場合、不明な送信元アドレスのパケットはドロップされます。セキュアMACアドレス数を上限よりも少なくするか、許容できるアドレスの最大数を増やさない限り、この状態が続きます。SNMP トラップが送信されます。Syslog メッセージがロギングされ、違反カウンタが増加します。

	コマンドまたはアクション	目的
		• **shutdown**：違反が発生すると、インターフェイスが error-disabled になり、ポートの LED が消灯します。SNMP トラップが送信されます。Syslog メッセージがロギングされ、違反カウンタが増加します。 • **shutdown vlan**：VLAN 単位でセキュリティ違反モードを設定するために使用します。このモードで違反が発生すると、ポート全体ではなく、VLAN が errdisable になります。 （注） セキュア ポートが error-disabled ステートの場合は、**errdisable recovery cause psecure-violation** グローバル コンフィギュレーション コマンドを入力して、このステートから回復させることができます。手動で再びイネーブルにするには、**shutdown** および **no shutdown** インターフェイス コンフィギュレーション コマンドを入力するか、**clear errdisable interface vlan** 特権 EXEC コマンドを入力します。
ステップ 9	**switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}] 例： `スイッチ(config-if)# switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice`	（任意）インターフェイスのセキュアMACアドレスを入力します。このコマンドを使用すると、最大数のセキュアMACアドレスを入力できます。設定したセキュアMACアドレスが最大数より少ない場合、残りのMACアドレスは動的に学習されます。 （注） このコマンドの入力後にスティッキー ラーニングをイネーブルにすると、動的に学習されたセキュア アドレスがスティッキー セキュア MAC アドレスに変換されて実行コンフィギュレーションに追加されます。 （任意）**vlan**：VLAN 当たりの最大値を設定します。 **vlan** キーワードを入力後、次のいずれかのオプションを入力します。 • *vlan-id*：トランク ポートで、VLAN ID および MACアドレスを指定できます。VLAN ID を指定しない場合、ネイティブ VLAN が使用されます。

	コマンドまたはアクション	目的
		• **access**：アクセス ポートで、VLAN をアクセス VLAN として指定します。 • **voice**：アクセス ポートで、VLAN を音声 VLAN として指定します。 （注） **voice** キーワードは、音声 VLAN がポートに設定されていて、さらにそのポートがアクセス VLAN でない場合のみ有効です。インターフェイスに音声 VLAN が設定されている場合、セキュア MAC アドレスの最大数を 2 に設定します。
ステップ 10	**switchport port-security mac-address sticky** 例： `スイッチ(config-if)# switchport port-security mac-address sticky`	（任意）インターフェイス上でスティッキ ラーニングをイネーブルにします。
ステップ 11	**switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}] 例： `スイッチ(config-if)# switchport port-security mac-address sticky 00:A0:C7:12:C9:25 vlan voice`	（任意）スティッキー セキュア MAC アドレスを入力し、必要な回数だけコマンドを繰り返します。設定したセキュア MAC アドレスの数が最大数より少ない場合、残りの MAC アドレスは動的に学習されてスティッキー セキュア MAC アドレスに変換され、実行コンフィギュレーションに追加されます。 （注） このコマンドの入力前にスティッキー ラーニングをイネーブルにしないと、エラー メッセージが表示されてスティッキー セキュア MAC アドレスを入力できません。 （任意）**vlan**：VLAN 当たりの最大値を設定します。 **vlan** キーワードを入力後、次のいずれかのオプションを入力します。 • *vlan-id*：トランク ポートで、VLAN ID および MAC アドレスを指定できます。VLAN ID を指定しない場合、ネイティブ VLAN が使用されます。 • **access**：アクセス ポートで、VLAN をアクセス VLAN として指定します。

	コマンドまたはアクション	目的
		• **voice**：アクセス ポートで、VLAN を音声 VLAN として指定します。 （注） **voice** キーワードは、音声 VLAN がポートに設定されていて、さらにそのポートがアクセス VLAN でない場合のみ有効です。
ステップ 12	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 13	**show port-security** 例： `スイッチ# show port-security`	入力を確認します。
ステップ 14	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 15	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ポート セキュリティ エージングのイネーブル化および設定

この機能を使用すると、既存のセキュア MAC アドレスを手動で削除しなくても、セキュア ポート上のデバイスを削除および追加し、なおかつポート上のセキュア アドレス数を制限できます。セキュア アドレスのエージングは、ポート単位でイネーブルまたはディセーブルにできます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}}

5. **end**
6. **show port-security** [**interface** *interface-id*] [**address**]
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}} 例： `スイッチ(config-if)# switchport port-security aging time 120`	セキュア ポートのスタティック エージングをイネーブルまたはディセーブルにします。またはエージング タイムやタイプを設定します。 （注） スイッチは、スティッキー セキュア アドレスのポート セキュリティ エージングをサポートしていません。 このポートに、スタティックに設定されたセキュア アドレスのエージングをイネーブルにする場合は、**static** を入力します。 *time* には、このポートのエージング タイムを指定します。指定できる範囲は、0 ～ 1440 分です。 **type** には、次のキーワードのいずれか 1 つを選択します。 • **absolute**：（任意）エージング タイプを絶対エージングとして設定します。このポートのセキュア アドレスはすべて、指定した時間（分単位）が経過すると期限切れになり、セキュア アドレス リストから削除されます。

	コマンドまたはアクション	目的
		• **inactivity**：（任意）エージング タイプを非アクティブ エージングとして設定します。指定された time 期間中にセキュア送信元アドレスからのデータ トラフィックがない場合に限り、このポートのセキュア アドレスが期限切れになります。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show port-security** [**interface** *interface-id*] [**address**] 例： `スイッチ# show port-security interface gigabitethernet 1/0/1`	入力を確認します。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

ポート セキュリティの設定例

次に、ポート上でポート セキュリティをイネーブルにし、セキュア アドレスの最大数を 50 に設定する例を示します。違反モードはデフォルトです。スタティック セキュア MAC アドレスは設定せず、スティッキー ラーニングはイネーブルです。

```
スイッチ(config)# interface gigabitethernet 1/0/1
スイッチ(config-if)# switchport mode access
スイッチ(config-if)# switchport port-security
スイッチ(config-if)# switchport port-security maximum 50
スイッチ(config-if)# switchport port-security mac-address sticky
```

次に、ポートの VLAN 3 上にスタティック セキュア MAC アドレスを設定する例を示します。

```
スイッチ(config)# interface gigabitethernet 1/0/2
スイッチ(config-if)# switchport mode trunk
スイッチ(config-if)# switchport port-security
スイッチ(config-if)# switchport port-security mac-address 0000.0200.0004 vlan 3
```

次に、ポートのスティッキー ポート セキュリティをイネーブルにする例を示します。データ VLAN および音声 VLAN の MAC アドレスを手動で設定し、セキュア アドレスの総数を 20 に設定します（データ VLAN に 10、音声 VLAN に 10 を割り当てます）。

```
スイッチ(config)# interface tengigabitethernet 1/0/1
スイッチ(config-if)# switchport access vlan 21
スイッチ(config-if)# switchport mode access
スイッチ(config-if)# switchport voice vlan 22
スイッチ(config-if)# switchport port-security
スイッチ(config-if)# switchport port-security maximum 20
スイッチ(config-if)# switchport port-security violation restrict
スイッチ(config-if)# switchport port-security mac-address sticky
スイッチ(config-if)# switchport port-security mac-address sticky 0000.0000.0002
スイッチ(config-if)# switchport port-security mac-address 0000.0000.0003
スイッチ(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice

スイッチ(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
スイッチ(config-if)# switchport port-security maximum 10 vlan access
スイッチ(config-if)# switchport port-security maximum 10 vlan voice
```

プロトコル ストーム プロテクションに関する情報

プロトコル ストーム プロテクション

スイッチがアドレス解決プロトコル（ARP）または制御パケットでフラッドされると、CPUの高い使用率により CPU のオーバーロードが発生する可能性があります。これらの問題は、次のように発生します。

- プロトコル制御パケットが受信されず、ネイバーの隣接がドロップされるため、ルーティング プロトコルがフラップする場合があります。
- スパニングツリー プロトコル（STP）ブリッジ プロトコル データ ユニット（BPDU）が送受信されないため、STP が再収束します。
- CLI が遅くなるか応答しなくなります。

プロトコル ストーム プロテクションを使用すると、パケットのフロー レートの上限しきい値を指定して、制御パケットが送信されるレートを制御できます。サポートされるプロトコルは、ARP、ARP スヌーピング、Dynamic Host Configuration Protocol（DHCP）v4、DHCP スヌーピング、インターネット グループ管理プロトコル（IGMP）、およびIGMP スヌーピングです。

パケットのレートが定義されたしきい値を超えると、スイッチは指定されたポートに着信したすべてのトラフィックを 30 秒間ドロップします。パケット レートが再度計測され、必要な場合はプロトコル ストーム プロテクションが再度適用されます。

より強力な保護が必要な場合は、仮想ポートを手動で errdisable にし、その仮想ポートのすべての着信トラフィックをブロックできます。また、手動で仮想ポートをイネーブルにしたり、仮想ポートの自動再イネーブル化の時間間隔を設定したりすることもできます。

（注） 超過したパケットは、2 つ以下の仮想ポートにおいてドロップされます。

仮想ポートのエラー ディセーブル化は、EtherChannel インターフェイスと Flexlink インターフェイスではサポートされません。

デフォルトのプロトコル ストーム プロテクションの設定

プロトコル ストーム プロテクションはデフォルトでディセーブルです。これがイネーブルになると、仮想ポートの自動リカバリがデフォルトでディセーブルになります。

プロトコル ストーム プロテクションの設定方法

プロトコル ストーム プロテクションのイネーブル化

手順の概要

1. **enable**
2. **configure terminal**
3. **psp** {**arp** | **dhcp** | **igmp**} pps *value*
4. **errdisable detect cause psp**
5. **errdisable recovery interval** *time*
6. **end**
7. **show psp config** {**arp** | **dhcp** | **igmp**}

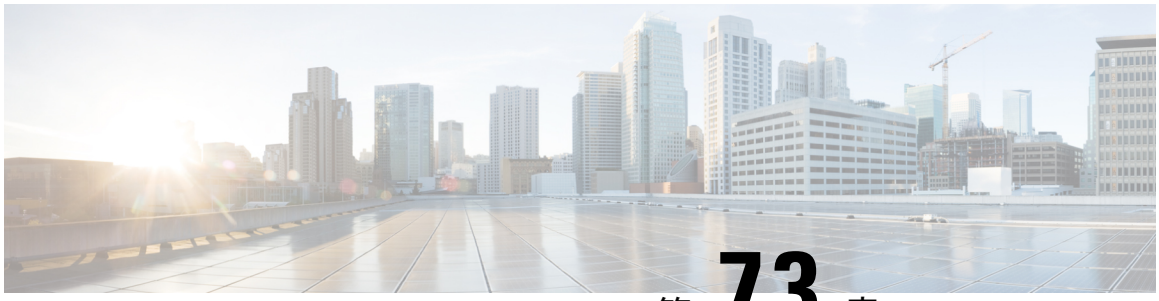
手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**psp** {**arp** | **dhcp** | **igmp**} pps *value* 例： `スイッチ(config)# psp dhcp pps 35`	ARP、IGMP、または DHCP に対してプロトコル ストーム プロテクションを設定します。 *value* には、1 秒あたりのパケット数のしきい値を指定します。トラフィックがこの値を超えると、プロトコル ストーム プロテクションが適用されます。範囲は毎秒 5 ～ 50 パケットです。
ステップ 4	**errdisable detect cause psp** 例： `スイッチ(config)# errdisable detect cause psp`	（任意）プロトコル ストーム プロテクションの errdisable 検出をイネーブルにします。この機能がイネーブルになると、仮想ポートが errdisable になります。この機能がディセーブルになると、そのポートは、ポートを errdisable にせずに超過したパケットをドロップします。
ステップ 5	**errdisable recovery interval** *time* 例： `スイッチ`	（任意）error-disabled の仮想ポートの自動リカバリ時間を秒単位で設定します。仮想ポートが error-disabled の場合、この時間を過ぎるとスイッチは自動的にリカバリします。指定できる範囲は 30 ～ 86400 秒です。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show psp config** {**arp** | **dhcp** | **igmp**} 例： `スイッチ# show psp config dhcp`	入力を確認します。

プロトコル ストーム プロテクションのモニタリング

コマンド	目的
show psp config {**arp** | **dhcp** | **igmp**}	入力内容を確認します。



第 73 章

IPv6 ファースト ホップ セキュリティの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

IPv6 でのファースト ホップ セキュリティの前提条件

- 必要な、IPv6 が有効になっている SDM テンプレートが設定されていること。

- **mls qos** コマンドを使用して CoPP ポリシーを設定する前に、スイッチで QoS を有効にする必要があります。

IPv6 でのファースト ホップ セキュリティの制約事項

- 次の制限は、FHS ポリシーを EtherChannel インターフェイスに適用する場合に該当します（ポート チャネル）。
  - FHS ポリシーがアタッチされた物理ポートは EtherChannel グループに参加することができません。
  - FHS ポリシーは、EtherChannel グループのメンバーである場合に物理ポートにアタッチすることができません。
- デフォルトでは、スヌーピング ポリシーにはセキュリティ レベルのガードがあります。そのようなスヌーピング ポリシーがアクセス スイッチに設定されると、ルータまたは DHCP サーバー/リレーに対応するアップリンク ポートが信頼できるポートとして設定されていても、IPv6（DHCPv6）サーバー パケットに対する外部 IPv6 ルータ アドバタイズメント(RA)または Dynamic Host Configuration Protocol はブロックされます。IPv6 RA または DHCPv6 サーバー メッセージを許可するには、次の手順を実行します。
  - IPv6 RA ガード ポリシー（RA の場合）または IPv6 DHCP ガード ポリシー（DHCP サーバー メッセージの場合）をアップリンク ポートに適用します。
  - 低いセキュリティ レベルでスヌーピング ポリシーを設定します（たとえば、glean や inspect など）。しかし、ファースト ホップ セキュリティ機能の利点が有効でないため、このようなスヌーピング ポリシーでは、低いセキュリティ レベルを設定することはお勧めしません。
- CSCvk32439 で報告された制限により、IPv6 SISF ベースのデバイス トラッキング ポリシーを使用した CoPP ポリシーには、次の制限が適用されます。
  - スイッチで IPv6 SISF ポリシーが設定されている場合、IPv6 NDP トラフィックを制限するには CoPP ポリシーが必要です。
  - NDP CoPP ポリシーが設定された後、制限されたトラフィックが CPU にヒットします。接続されているエンドポイントの合計に対応するには、NDP CoPP ポリシーの数を、スタック内の各スイッチに接続するユーザーの数よりわずかに多くする必要があります。スイッチに接続されているエンドポイントの数よりも少ない NDP CoPP ポリシーを設定すると、エンドポイントへの IP 割り当ては遅延しますが、完全に無視されるわけではありません。

（注） たとえば、5 つのスイッチのスタックに約 300 のユーザーがいる場合、NDP CoPP ポリシーは 300 を超える必要があります。

- DHCPv6（サーバーからクライアントおよびクライアントからサーバー）CoPP ポリシーは、Lightweight DHCPv6 リレーエージェント（LDRA）がスイッチの IPv6 SISF ベースのデバイス トラッキング ポリシーで設定されている場合にのみ必要です。

IPv6 でのファースト ホップ セキュリティに関する情報

IPv6 のファーストホップ セキュリティ（FHS IPv6）は、ポリシーを物理インターフェイス、または VLAN にアタッチできる一連の IPv6 セキュリティ機能です。IPv6 ソフトウェア ポリシー データベース サービスは、これらのポリシーを保存しアクセスします。ポリシーを設定または変更すると、ポリシー属性はソフトウェア ポリシー データベースに保存または更新され、その後指定したとおりに適用されます。次の IPv6 ポリシーが現在サポートされています。

- IPv6 スヌーピング ポリシー：IPv6 スヌーピング ポリシーは、IPv6 内の FHS で使用できるほとんどの機能を有効にできるコンテナ ポリシーとして機能します。
- IPv6 FHS バインディング テーブルの内容：スイッチに接続された IPv6 ネイバーのデータベース テーブルはネイバー探索（ND）プロトコル スヌーピングなどの情報ソースから作成されます。このデータベースまたはバインディング テーブルは、リンク層アドレス（LLA）、IPv4 または IPv6 アドレス、およびスプーフィングやリダイレクト攻撃を防止するためにネイバーのプレフィックス バインディングを検証するために、さまざまな IPv6 ガード機能（IPv6 ND 検査など）によって使用されます。
- IPv6 ネイバー探索検査：IPv6 ND 検査は、レイヤ 2 ネイバー テーブル内のステートレス自動設定アドレスのバインディングを学習し、保護します。IPv6 ND 検査は、信頼できるバインディング テーブル データベースを構築するためにネイバー探索メッセージを分析します。準拠していない IPv6 ネイバー探索メッセージは破棄されます。ND メッセージは、その IPv6 からメディア アクセス コントロール（MAC）へのマッピングが検証可能な場合に信頼できると見なされます。

  この機能によって、DAD、アドレス解決、ルータ ディスカバリ、ネイバー キャッシュに対する攻撃などの、ND メカニズムに固有の脆弱性のいくつかが軽減されます。
- IPv6 ルータ アドバタイズメント ガード：IPv6 ルータ アドバタイズメント（RA）ガード機能を使用すると、ネットワーク管理者は、ネットワーク スイッチ プラットフォームに到着した不要または不正な RA ガード メッセージをブロックまたは拒否できます。RA は、リンクで自身をアナウンスするためにルータによって使用されます。RA ガード機能は、これらの RA を分析して、未承認のルータによって送信された偽の RA をフィルタリングして除外します。ホスト モードでは、ポートではルータ アドバタイズメントとルータ リダイレクト メッセージはすべて許可されません。RA ガード機能は、レイヤ 2 デバイスの設定情報を、受信した RA フレームで検出された情報と比較します。レイヤ 2 デバイスは、RA フレームとルータ リダイレクト フレームの内容を設定と照らし合わせて検証した後で、RA をユニキャストまたはマルチキャストの宛先に転送します。RA フレームの内容が検証されない場合は、RA は破棄されます。
- IPv6 DHCP ガード：IPv6 DHCP ガード機能は、承認されない DHCPv6 サーバーおよびリレー エージェントからの返信およびアドバタイズメント メッセージをブロックします。

IPv6 DHCP ガードは、偽造されたメッセージがバインディング テーブルに入るのを防ぎ、DHCPv6 サーバーまたは DHCP リレーからデータを受信することが明示的に設定されていないポートで受信された DHCPv6 サーバー メッセージをブロックできます。この機能を使用するには、ポリシーを設定してインターフェイスまたは VLAN にアタッチします。DHCP ガード パケットをデバッグするには、**debug ipv6 snooping dhcp-guard** 特権 EXEC コマンドを使用します。

- IPv6 ソース ガード：IPv4 ソース ガードと同様、IPv6 ソース ガードは送信元アドレス スプーフィングを防ぐために、送信元アドレスまたはプレフィックスを検証します。

  ソース ガードでは、送信元または宛先アドレスに基づいてトラフィックを許可または拒否するようにハードウェアをプログラムします。ここでは、データ パケットのトラフィックのみを処理します。

  ソース ガード パケットをデバッグするには、debug ipv6 snooping source-guard 特権 EXEC コマンドを使用します。

  次の制約事項が適用されます。

  - FHS ポリシーは、EtherChannel グループのメンバーである場合に物理ポートにアタッチすることができません。
  - IPv6 ソース ガードがスイッチ ポートで有効になっている場合は、そのスイッチ ポートが属するインターフェイスで NDP または DHCP スヌーピングを有効にする必要があります。そうしないと、このポートからのすべてのデータ トラフィックがブロックされます。
  - IPv6 ソース ガード ポリシーを VLAN に適用することはできません。インターフェイス レベルのみでサポートされています。
  - インターフェイスで IPv4 および IPv6 のソース ガードを一緒に設定する場合は、**ip verify source** の代わりに **ip verify source mac-check** の使用を推奨します。2 つの異なるフィルタリング ルール（IPv4（IP フィルタ）用と IPv6（IP-MAC フィルタ）用）が設定されているため、特定のポートの IPv4 接続が切断される可能性があります。
  - IPv6 ソース ガードとプレフィックス ガードは同時に使用できません。ポリシーをインターフェイスに付加する際は、「アドレスを確認」するか「プレフィックスを確認」する必要はありますが、両方を確認する必要はありません。
  - PVLAN と送信元/プレフィックス ガードは同時に適用できません。

  IPv6 送信元ガードの詳細については、Cisco.comで『Cisco IOS IPv6 Configuration Guide Library』の「IPv6 Source Guard」の章を参照してください。

- IPv6 プレフィックス ガード：IPv6 プレフィックス ガードは、IPv6 送信元ガード機能内で動作し、デバイスがトポロジに不正なアドレスから発信されたトラフィックを拒否できるようにします。IPv6 プレフィックス ガードは、IPv6 プレフィックスが DHCP プレフィックス委任を使用してデバイス（ホーム ゲートウェイなど）に委任される場合によく使用されています。この機能は、リンクに割り当てられたアドレスの範囲を検出し、この範囲に入っていないアドレスを発信元とするトラフィックをブロックします。

IPv6 プレフィックス ガードの詳細については、Cisco.comで『Cisco IOS IPv6 Configuration Guide Library』の「IPv6 Prefix Guard」の章を参照してください。

- IPv6 宛先ガード：IPv6 宛先ガード機能は、IPv6 ネイバー探索で動作し、リンク上でアクティブであると認識されているアドレスについてのみ、デバイスがアドレスを解決します。アドレス グリーニング機能に依存して、リンク上でアクティブなすべての宛先をバインディング テーブルに挿入してから、バインディング テーブルで宛先が見つからなかったときに実行される解決をブロックします。

  IPv6 宛先ガードに関する詳細については、Cisco.comで『Cisco IOS IPv6 Configuration Guide Library』の「IPv6 Destination Guard」の章を参照してください。

- IPv6 ネイバー検索マルチキャスト抑制：IPv6 ネイバー検索マルチキャスト抑制機能は、IPv6 のスヌーピング機能で、スイッチまたはワイヤレス コントローラで実行し、適切なリンク動作に必要な制御トラフィック量を削減するために使用されます。

- DHCPv6 リレー：Lightweight DHCPv6 リレー エージェント：Lightweight DHCPv6 リレー エージェント機能を使用するとリンクレイヤブリッジング（非ルーティング）機能を実行するアクセス ノードによってリレーエージェント情報が挿入されます。Lightweight DHCPv6 リレー エージェント（LDRA）機能は、DSL アクセス マルチプレクサ（DSLAM）や IPv6 制御やルーティング機能をサポートしないイーサネット スイッチなどの既存のアクセス ノードに実装できます。LDRA を使用して、DHCP バージョン 6（DHCPv6）メッセージ交換にリレーエージェント オプションを挿入して、主にクライアント側のインターフェイスを特定します。LDRA 機能は、インターフェイスと VLAN でイネーブルにできます。

（注） LDRA デバイスがクライアントに直接接続されている場合は、サーバー側で特定のサブネットまたはリンク情報を取得するために、インターフェイスにプール設定が必要です。この場合、LDRA デバイスが異なるサブネットまたはリンクに存在する場合、サーバーは正しいサブネットを取得できない場合があります。インターフェイスでプール名を設定して、クライアントに適切なサブネットまたはリンクを選択できるようになりました。

DHCPv6 リレーの詳細については、『IP Addressing: DHCP Configuration Guide, Cisco IOS Release 15.1SG』の「DHCPv6 Relay—Lightweight DHCPv6 Relay Agent」の項を参照してください。

IPv6 スヌーピング ポリシーの設定方法

IPv6 スヌーピング ポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**

2. **ipv6 snooping policy** *policy-name*
3. {[**default**] | [**device-role** {**node** | **switch**}] | [**limit address-count** *value*] | [**no**] | [**protocol** {**dhcp** | **ndp**}] | [**security-level** {**glean** | **guard** | **inspect**}] | [**tracking** {**disable** [**stale-lifetime** [*seconds* | **infinite**] | **enable** [**reachable-lifetime** [*seconds* | **infinite**] }] | [**trusted-port**] }
4. **end**
5. **show ipv6 snooping policy** *policy-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ipv6 snooping policy** *policy-name* 例： スイッチ(config)# **ipv6 snooping policy example_policy**	スヌーピング ポリシーを作成し、IPv6 スヌーピング ポリシー コンフィギュレーション モードに移行します。
ステップ 3	{[**default**]	[**device-role** {**node**	**switch**}]	[**limit address-count** *value*]	[**no**]	[**protocol** {**dhcp**	**ndp**}]	[**security-level** {**glean**	**guard**	**inspect**}]	[**tracking** {**disable** [**stale-lifetime** [*seconds*	**infinite**]	**enable** [**reachable-lifetime** [*seconds*	**infinite**] }]	[**trusted-port**] } 例： スイッチ**(config-ipv6-snooping)# security-level inspect** 例： スイッチ**(config-ipv6-snooping)# trusted-port**	データ アドレス グリーニングを有効にし、さまざまな条件に対してメッセージを検証し、メッセージのセキュリティ レベルを指定します。 • （任意）**default**：すべてをデフォルト オプションに設定します。 • （任意）**device-role**{**node**]|**switch**}：ポートに接続されたデバイスの役割を指定します。デフォルトは **node** です。 • （任意）**limit address-count** *value*：ターゲットごとに許可されるアドレス数を制限します。 • （任意）**no**：コマンドを無効にするか、またはそのデフォルトに設定します。 • （任意）**protocol**{**dhcp**	**ndp**}：分析のために、スヌーピング機能にどのプロトコルをリダイレクトするかを指定します。デフォルトは、**dhcp** および **ndp** です。デフォルトを変更するには、**no protocol** コマンドを使用します。 • （任意）**security-level**{**glean**|**guard**|**inspect**}：この機能によって適用されるセキュリティのレベルを指定します。デフォルトは **guard** です。

	コマンドまたはアクション	目的
		glean：メッセージからアドレスを収集し、何も確認せずにバインディング テーブルに入力します。 **guard**：アドレスを収集し、メッセージを検査します。さらに、RA および DHCP サーバ メッセージを拒否します。これがデフォルトのオプションです。 **inspect**：アドレスを収集し、メッセージの一貫性と準拠を検証して、アドレスの所有権を適用します。 • （任意）**tracking** {**disable** | **enable**}：デフォルトの追跡動作を上書きし、追跡オプションを指定します。 • （任意）**trusted-port**：信頼できるポートを設定します。これにより、該当するターゲットに対するガードがディセーブルになります。信頼できるポートを経由して学習されたバインディングは、他のどのポートを経由して学習されたバインディングよりも優先されます。テーブル内にエントリを作成しているときに衝突が発生した場合、信頼できるポートが優先されます。
ステップ 4	**end** 例： `スイッチ(config-ipv6-snooping)# exit`	コンフィギュレーション モードから特権 EXEC モードに戻ります。
ステップ 5	**show ipv6 snooping policy** *policy-name* 例： `スイッチ#show ipv6 snooping policy example_policy`	スヌーピング ポリシー設定を表示します。

次のタスク

IPv6 スヌーピング ポリシーをインターフェイスまたは VLAN にアタッチします。

IPv6 スヌーピング ポリシーをインターフェイスにアタッチする方法

インターフェイスまたは VLAN に IPv6 スヌーピング ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**

2. **interface** Interface_type *stack/module/port*
3. **switchport**
4. **ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
5. **do show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** Interface_type *stack/module/port* 例： `スイッチ(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**switchport** 例： `スイッチ(config-if)# switchport`	switchport モードを開始します。 （注） インターフェイスがレイヤ 3 モードの場合に、レイヤ 2 パラメータを設定するには、パラメータを指定せずに switchport インターフェイス コンフィギュレーション コマンドを入力し、インターフェイスをレイヤ 2 モードにする必要があります。これにより、インターフェイスがいったんシャットダウンしてから再度有効になり、インターフェイスが接続しているデバイスに関するメッセージが表示されることがあります。レイヤ 3 モードのインターフェイスをレイヤ 2 モードにした場合、影響のあるインターフェイスに関連する以前の設定情報が消失する可能性があり、インターフェイスはデフォルト設定に戻ります。switchport コンフィギュレーション モードではコマンド プロンプトは（config-if）# と表示されます。
ステップ 4	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id*	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove** *vlan_ids*}]	**vlan** {*vlan_id*	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] 例：	インターフェイスまたはそのインターフェイス上の特定の VLAN にカスタム IPv6 スヌーピング ポリシーをアタッチします。デフォルト ポリシーをインターフェイスにアタッチするには、**attach-policy** キーワードを指定せずに **ipv6 snooping** コマンドを使用しま

	コマンドまたはアクション	目的
	スイッチ(config-if)# **ipv6 snooping** or スイッチ(config-if)# **ipv6 snooping attach-policy example_policy** or スイッチ(config-if)# **ipv6 snooping vlan 111,112** or スイッチ(config-if)# **ipv6 snooping attach-policy example_policy vlan 111,112**	す。デフォルト ポリシーをインターフェイス上の VLAN にアタッチするには、**ipv6 snooping vlan** コマンドを使用します。デフォルトポリシーは、セキュリティ レベル **guard**、デバイス ロール **node**、プロトコル **ndp** および **dhcp** です。
ステップ 5	**do show running-config** 例： スイッチ#(config-if)# **do show running-config**	インターフェイス コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 スヌーピング ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

EtherChannel インターフェイスまたは VLAN に IPv6 スヌーピング ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface range** *Interface_name* 例： スイッチ(config)# **interface range Po11**	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーションモードを開始します。 ヒント　インターフェイス名やタイプを簡単に参照するには **do show interfaces summary** コマンドを使用します。
ステップ 3	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]	IPv6 スヌーピング ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

	コマンドまたはアクション	目的
	例： スイッチ(config-if-range)# **ipv6 snooping attach-policy example_policy** or スイッチ(config-if-range)# **ipv6 snooping attach-policy example_policy vlan 222,223,224** **or** スイッチ(config-if-range)#**ipv6 snooping vlan 222, 223,224**	
ステップ 4	**do show running-config interface***portchannel_interface_name* 例： スイッチ#(config-if-range)# **do show running-config int po11**	コンフィギュレーションモードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 スヌーピング ポリシーを VLAN にグローバルにアタッチする方法

複数のインターフェイス上の VLAN に IPv6 スヌーピング ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 snooping** [**attach-policy** *policy_name*]
4. **do show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vlan configuration** *vlan_list* 例： スイッチ(config)# **vlan configuration 333**	VLAN インターフェイスのコンフィギュレーション モードを開始し、IPv6 スヌーピング ポリシーをアタッチする VLAN を指定します。

	コマンドまたはアクション	目的
ステップ 3	**ipv6 snooping** [**attach-policy** *policy_name*] 例： `スイッチ(config-vlan-config)#ipv6 snooping attach-policy example_policy`	すべてのスイッチおよびスタック インターフェイスで、IPv6 スヌーピング ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。デフォルト ポリシーは、セキュリティ レベル **guard**、デバイス ロール **node**、プロトコル **ndp** および **dhcp** です。
ステップ 4	**do show running-config** 例： `スイッチ#(config-if)# do show running-config`	インターフェイス コンフィギュレーション モードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 バインディング テーブルの内容を設定する方法

IPv6 バインディング テーブル コンテンツを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. [**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/port hw_address* [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**tracking**{ [default | disable] [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**enable** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] }]
3. [**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* | **port-limit** *number* [**mac-limit** *number*] | **vlan-limit** *number* [[**mac-limit** *number*] | [**port-limit** *number* [**mac-limit***number*]]]]
4. **ipv6 neighbor binding logging**
5. **exit**
6. **show ipv6 neighbor binding**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	[**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/port hw_address* [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**tracking**{ [default	disable] [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**enable**	バインディング テーブル データベースにスタティック エントリを追加します。

	コマンドまたはアクション	目的
	[**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] }] 例： `スイッチ(config)# ipv6 neighbor binding`	
ステップ 3	[**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* | **port-limit** *number* [**mac-limit** *number*] | **vlan-limit** *number* [[**mac-limit** *number*] | [**port-limit** *number* [**mac-limit***number*]]]] 例： `スイッチ(config)# ipv6 neighbor binding max-entries 30000`	バインディング テーブル キャッシュに挿入できるエントリの最大数を指定します。
ステップ 4	**ipv6 neighbor binding logging** 例： `スイッチ(config)# ipv6 neighbor binding logging`	バインディング テーブル メイン イベントのロギングを有効にします。
ステップ 5	**exit** 例： `スイッチ(config)# exit`	グローバル コンフィギュレーション モードを終了して、ルータを特権 EXEC モードにします。
ステップ 6	**show ipv6 neighbor binding** 例： `スイッチ# show ipv6 neighbor binding`	バインディング テーブルの内容を表示します。

IPv6 ネイバー探索検査ポリシーの設定方法

特権 EXEC モードから、IPv6 ND 検査ポリシーを設定するには、次の手順に従ってください。

手順の概要

1. **configure terminal**
2. [**no**]**ipv6 nd inspection policy** *policy-name*
3. **device-role** {**host** | **monitor** | **router** | **switch**}
4. **drop-unsecure**
5. **limit address-count** *value*
6. **sec-level minimum** *value*
7. **tracking** {**enable** [**reachable-lifetime** {*value* | **infinite**}] | **disable** [**stale-lifetime** {*value* | **infinite**}]}
8. **trusted-port**
9. **validate source-mac**

10. **no** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**}
11. **default** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**}
12. **do show ipv6 nd inspection policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	[**no**]**ipv6 nd inspection policy** *policy-name* 例： `スイッチ(config)# ipv6 nd inspection policy example_policy`	ND 検査ポリシー名を指定し、ND 検査ポリシー コンフィギュレーション モードを開始します。
ステップ 3	**device-role** {**host**	**monitor**	**router**	**switch**} 例： `スイッチ(config-nd-inspection)# device-role switch`	ポートに接続されているデバイスの役割を指定します。デフォルトは **host** です。
ステップ 4	**drop-unsecure** 例： `スイッチ(config-nd-inspection)# drop-unsecure`	オプションが指定されていないか無効なオプションが指定されているか、またはシグニチャが無効なメッセージをドロップします。
ステップ 5	**limit address-count** *value* 例： `スイッチ(config-nd-inspection)# limit address-count 1000`	1 ～ 10,000 を入力します。
ステップ 6	**sec-level minimum** *value* 例： `スイッチ(config-nd-inspection)# limit address-count 1000`	暗号化生成アドレス（CGA）オプションを使用する場合の最小のセキュリティ レベル パラメータ値を指定します。
ステップ 7	**tracking** {**enable** [**reachable-lifetime** {*value*	**infinite**}]	**disable** [**stale-lifetime** {*value*	**infinite**}]} 例： `スイッチ(config-nd-inspection)# tracking disable stale-lifetime infinite`	ポートのデフォルトのデバイス追跡ポリシーを上書きします。
ステップ 8	**trusted-port** 例： `スイッチ(config-nd-inspection)# trusted-port`	信頼できるポートにするポートを設定します。

	コマンドまたはアクション	目的
ステップ 9	**validate source-mac** 例： `スイッチ(config-nd-inspection)# validate source-mac`	送信元 Media Access Control（MAC）アドレスをリンク層アドレスと照合します。
ステップ 10	**no** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**} 例： `スイッチ(config-nd-inspection)# no validate source-mac`	このコマンドの **no** 形式を使用してパラメータの現在の設定を削除します。
ステップ 11	**default** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**} 例： `スイッチ(config-nd-inspection)# default limit address-count`	設定をデフォルト値に戻します。
ステップ 12	**do show ipv6 nd inspection policy** *policy_name* 例： `スイッチ(config-nd-inspection)# do show ipv6 nd inspection policy example_policy`	ND検査コンフィギュレーションモードを終了しないで ND 検査の設定を確認します。

IPv6 ネイバー探索検査ポリシーをインターフェイスにアタッチする方法

Cisco IOS XE Amsterdam 17.1.1 以降、IPv6 ND 検査機能は廃止され、SISF ベースのデバイス追跡機能に置き換えられます。対応する置き換えタスクについては、このドキュメントの「*SISF* ベースのデバイス追跡の設定」の章の「デバイス追跡ポリシーのインターフェイスへの適用」を参照してください。

インターフェイスまたはそのインターフェイス上の VLAN に IPv6 ND 検査ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** Interface_type *stack/module/port* 例： スイッチ(config)# **interface gigabitethernet 1/1/4**	インターフェイスのタイプおよび ID を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] 例： スイッチ(config-if)# **ipv6 nd inspection attach-policy example_policy** or スイッチ(config-if)# **ipv6 nd inspection attach-policy example_policy vlan 222,223,224** **or** スイッチ(config-if)# **ipv6 nd inspection vlan 222, 223,224**	ネイバー探索検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**do show running-config** 例： スイッチ#(config-if)# **do show running-config**	インターフェイス コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 ネイバー探索検査ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

Cisco IOS XE Amsterdam 17.1.1 以降、IPv6 ND 検査機能は廃止され、SISF ベースのデバイス追跡機能に置き換えられます。対応する置き換えタスクについては、このドキュメントの「*SISF* ベースのデバイス追跡の設定」の章の「デバイス追跡ポリシーのインターフェイスへの適用」を参照してください。

EtherChannel インターフェイスまたは VLAN に IPv6 ネイバー探索検査ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface range** *Interface_name* 例： `スイッチ(config)# interface Po11`	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーション モードを開始します。 ヒント　インターフェイス名やタイプを簡単に参照するには **do show interfaces summary** コマンドを使用します。
ステップ 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] 例： `スイッチ(config-if-range)# ipv6 nd inspection attach-policy example_policy` or `スイッチ(config-if-range)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224` **or** `スイッチ(config-if-range)#ipv6 nd inspection vlan 222, 223,224`	ND 検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**do show running-config interface***portchannel_interface_name* 例：	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

	コマンドまたはアクション	目的
	スイッチ#(config-if-range)# **do show running-config int po11**	

IPv6 ネイバー探索検査ポリシーを全体的に VLAN にアタッチする方法

Cisco IOS XE Amsterdam 17.1.1 以降、IPv6 ND 検査機能は廃止され、SISF ベースのデバイス追跡機能に置き換えられます。対応する置き換えタスクについては、このドキュメントの「*SISF* ベースのデバイス追跡の設定」の「デバイス追跡ポリシーの *VLAN* への適用」を参照してください。

複数のインターフェイス上の VLAN に IPv6 ND 探索ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 nd inspection** [**attach-policy** *policy_name*]
4. **do show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vlan configuration** *vlan_list* 例： スイッチ(config)# **vlan configuration 334**	VLAN インターフェイスのコンフィギュレーション モードを開始し、IPv6 スヌーピング ポリシーをアタッチする VLAN を指定します。
ステップ 3	**ipv6 nd inspection** [**attach-policy** *policy_name*] 例： スイッチ(config-vlan-config)#**ipv6 nd inspection attach-policy example_policy**	すべてのスイッチおよびスタック インターフェイスで、IPv6 ネイバー探索ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。 デフォルトのポリシーは、device-role **host**、no drop-unsecure、limit address-count disabled、sec-level minimum is disabled、tracking is disabled、no trusted-port、no validate source-mac です。

	コマンドまたはアクション	目的
ステップ 4	**do show running-config** 例： スイッチ#(config-if)# **do show running-config**	コンフィギュレーションモードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 ルータ アドバタイズメント ガード ポリシーの設定方法

IPv6 ルータ アドバタイズメント ポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. [**no**]**ipv6 nd raguard policy** *policy-name*
3. [**no**]**device-role** {**host** | **monitor** | **router** | **switch**}
4. [**no**]**hop-limit** {**maximum** | **minimum**} *value*
5. [**no**]**managed-config-flag** {**off** | **on**}
6. [**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*}
7. [**no**]**other-config-flag** {**on** | **off**}
8. [**no**]**router-preference maximum** {**high** | **medium** | **low**}
9. [**no**]**trusted-port**
10. **default** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**}
11. **do show ipv6 nd raguard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	[**no**]**ipv6 nd raguard policy** *policy-name* 例： スイッチ(config)# **ipv6 nd raguard policy example_policy**	RA ガード ポリシー名を指定し、RA ガード ポリシー コンフィギュレーション モードを開始します。
ステップ 3	[**no**]**device-role** {**host**	**monitor**	**router**	**switch**} 例： スイッチ(config-nd-raguard)# **device-role switch**	ポートに接続されているデバイスの役割を指定します。デフォルトは **host** です。

	コマンドまたはアクション	目的
ステップ 4	[**no**]**hop-limit** {**maximum** | **minimum**} *value* 例： スイッチ(config-nd-raguard)# **hop-limit maximum 33**	（1～255）最大および最小のホップ制限値の範囲。 ホップ制限値によるルータアドバタイズメントメッセージのフィルタリングを有効にします。不正RAメッセージは低いホップ制限値（IPv4 の Time to Live と同じ）を持つ可能性があるため、ホストによって受け入れられると、ホストが不正 RA メッセージ ジェネレータを超えて宛先にトラフィックを生成することができなくなります。指定されていないホップ制限値を持つRAメッセージはブロックされます。 設定されていない場合、このフィルタは無効になります。「**minimum**」を設定して、指定する値より低いホップ制限値を持つRAメッセージをブロックします。「**maximum**」を設定して、指定する値より高いホップ制限値を持つRAメッセージをブロックします。
ステップ 5	[**no**]**managed-config-flag** {**off** | **on**} 例： スイッチ(config-nd-raguard)# **managed-config-flag on**	管理アドレス設定（「M」フラグ）フィールドに基づいてルータアドバタイズメントメッセージのフィルタリングを有効にします。「M」フィールドが 1 の不正 RA メッセージの結果としてホストが不正 DHCPv6 サーバーを使用する場合があります。設定されていない場合、このフィルタは無効になります。 **On**：「M」値が 1 の RA メッセージを受け入れて転送し、0 のものをブロックします。 **Off**：「M」値が 0 の RA メッセージを受け入れて転送し、1 のものをブロックします。
ステップ 6	[**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*} 例： スイッチ(config-nd-raguard)# **match ipv6 access-list example_list**	指定したプレフィックス リストまたはアクセス リストと照合します。
ステップ 7	[**no**]**other-config-flag** {**on** | **off**} 例： スイッチ(config-nd-raguard)# **other-config-flag on**	その他の設定（「O」フラグ）フィールドに基づくルータ アドバタイズメント メッセージのフィルタリングを有効にします。「O」フィールドが 1 の不正 RA メッセージの結果としてホストが不正 DHCPv6 サーバーを使用する場合があります。設定されていない場合、このフィルタは無効になります。

	コマンドまたはアクション	目的
		On：「O」値が 1 の RA メッセージを受け入れて転送し、0 のものをブロックします。 **Off**：「O」値が 0 の RA メッセージを受け入れて転送し、1 のものをブロックします。
ステップ 8	[**no**]**router-preference maximum** {**high** | **medium** | **low**} 例： `スイッチ(config-nd-raguard)# router-preference maximum high`	「Router Preference」フラグを使用したルータ アドバタイズメント メッセージのフィルタリングを有効にします。設定されていない場合、このフィルタは無効になります。 • **high**：「Router Preference」が「high」、「medium」、または「low」に設定された RA メッセージを受け入れます。 • **medium**：「Router Preference」が「high」に設定された RA メッセージをブロックします。 • **low**：「Router Preference」が「medium」または「high」に設定された RA メッセージをブロックします。
ステップ 9	[**no**]**trusted-port** 例： `スイッチ(config-nd-raguard)# trusted-port`	信頼できるポートとして設定すると、すべての接続デバイスが信頼され、より詳細なメッセージ検証は実行されません。
ステップ 10	**default** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**} 例： `スイッチ(config-nd-raguard)# default hop-limit`	コマンドをデフォルト値に戻します。
ステップ 11	**do show ipv6 nd raguard policy** *policy_name* 例： `スイッチ(config-nd-raguard)# do show ipv6 nd raguard policy example_policy`	（任意）：RA ガード ポリシー コンフィギュレーション モードを終了しないで ND ガード ポリシー設定を表示します。

IPv6 ルータ アドバタイズメント ガード ポリシーをインターフェイスにアタッチする方法

インターフェイスまたはそのインターフェース上の VLAN に IPv6 ルータ アドバタイズメント ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** Interface_type *stack/module/port* 例： `スイッチ(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids*	**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}]	**vlan** [{*vlan_ids*	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] 例： `スイッチ(config-if)# ipv6 nd raguard attach-policy example_policy` or `スイッチ(config-if)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224` **or** `スイッチ(config-if)# ipv6 nd raguard vlan 222, 223,224`	ネイバー探索検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**do show running-config** 例： `スイッチ#(config-if)# do show running-config`	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6ルータアドバタイズメントガードポリシーをレイヤ2EtherChannelインターフェイスにアタッチする方法

EtherChannel インターフェイスまたは VLAN に IPv6 ルータ アドバタイズメント ガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface range** *Interface_name* 例： `スイッチ(config)# interface Po11`	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーション モードを開始します。 ヒント　インターフェイス名やタイプを簡単に参照するには **do show interfaces summary** コマンドを使用します。
ステップ 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] 例： `スイッチ(config-if-range)# ipv6 nd raguard attach-policy example_policy` or `スイッチ(config-if-range)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224` **or** `スイッチ(config-if-range)#ipv6 nd raguard vlan 222, 223,224`	RA ガード ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

	コマンドまたはアクション	目的
ステップ 4	**do show running-config interface***portchannel_interface_name* 例： `スイッチ#(config-if-range)# do show running-config int po11`	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 ルータ アドバタイズメント ガード ポリシーを VLAN にグローバルにアタッチする方法

インターフェイスに関係なく VLAN に IPv6 ルータ アドバタイズメント ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vlan configuration** *vlan_list* 例： `スイッチ(config)# vlan configuration 335`	VLAN インターフェイスのコンフィギュレーション モードを開始し、IPv6 RA ガード ポリシーをアタッチする VLAN を指定します。
ステップ 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] 例： `スイッチ(config-vlan-config)#ipv6 nd raguard attach-policy example_policy`	すべてのスイッチおよびスタック インターフェイスで、IPv6 RA ガード ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**do show running-config** 例： `スイッチ#(config-if)# do show running-config`	コンフィギュレーション モードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 DHCP ガード ポリシーの設定方法

IPv6 DHCP（DHCPv6）ガード ポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. [**no**]**ipv6 dhcp guard policy** *policy-name*
3. [**no**]**device-role** {**client** | **server**}
4. [**no**] **match server access-list** *ipv6-access-list-name*
5. [**no**] **match reply prefix-list** *ipv6-prefix-list-name*
6. [**no**]**preference**{ **max** *limit* | **min** *limit* }
7. [**no**] **trusted-port**
8. **default** {**device-role** | **trusted-port**}
9. **do show ipv6 dhcp guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	[**no**]**ipv6 dhcp guard policy** *policy-name* 例： `スイッチ(config)# ipv6 dhcp guard policy example_policy`	DHCPv6 ガード ポリシー名を指定し、DHCPv6 ガード ポリシー コンフィギュレーション モードを開始します。
ステップ 3	[**no**]**device-role** {**client** | **server**} 例： `スイッチ(config-dhcp-guard)# device-role server`	（任意）特定の役割のデバイスからのものではないポート上の DHCPv6 応答および DHCPv6 アドバタイズメントをフィルタします。デフォルトは **client** です。 • **client**：デフォルト値。アタッチされたデバイスがクライアントであることを指定します。サーバー メッセージにはこのポートで破棄されます。 • **server**：適用されたデバイスが DHCPv6 サーバーであることを指定します。このポートでは、サーバー メッセージが許可されます。

	コマンドまたはアクション	目的
ステップ 4	[**no**] **match server access-list** *ipv6-access-list-name* 例： `;;Assume a preconfigured IPv6 Access List as follows:` `スイッチ(config)# ipv6 access-list my_acls` `スイッチ(config-ipv6-acl)# permit host FE80::A8BB:CCFF:FE01:F700 any` `;;configure DCHPv6 Guard to match approved access list.` `スイッチ(config-dhcp-guard)# match server access-list my_acls`	（任意）。アドバタイズされたDHCPv6サーバーまたはリレーアドレスが認証されたサーバーのアクセスリストからのものであることの確認を有効にします（アクセス リストの宛先アドレスは「any」です）。設定されていない場合、このチェックは回避されます。空のアクセスリストは、permit allとして処理されます。
ステップ 5	[**no**] **match reply prefix-list** *ipv6-prefix-list-name* 例： `;;Assume a preconfigured IPv6 prefix list as follows:` `スイッチ(config)# ipv6 prefix-list my_prefix permit 2001:0DB8::/64 le 128` `;; Configure DCHPv6 Guard to match prefix` `スイッチ(config-dhcp-guard)# match reply prefix-list my_prefix`	（任意）DHCPv6応答メッセージ内のアドバタイズされたプレフィクスが設定された承認プレフィクスリストからのものであることの確認を有効にします。設定されていない場合、このチェックは回避されます。空のプレフィクスリストは、permitとして処理されます。
ステップ 6	[**no**]**preference** { **max** *limit* | **min** *limit* } 例： `スイッチ(config-dhcp-guard)# preference max 250` `スイッチ(config-dhcp-guard)#preference min 150`	**device-role** が **server** である場合に **max** および **min** を設定して、DHCPv6サーバーアドバタイズメント値をサーバー優先度値に基づいてフィルタします。デフォルトではすべてのアドバタイズメントが許可されます。 **max** *limit*：（0～255）（任意）アドバタイズされたプリファレンス（[preference]オプション内）が指定された制限未満であるかどうかの検証を有効にします。デフォルトは255です。設定されていない場合、このチェックは回避されます。 **min** *limit*：（0～255）（任意）アドバタイズされたプリファレンス（[preference]オプション内）が指定された制限を超過しているかどうかの検証を有効にします。デフォルトは0です。設定されていない場合、このチェックは回避されます。
ステップ 7	[**no**] **trusted-port** 例： `スイッチ(config-dhcp-guard)# trusted-port`	（任意）**trusted-port**：ポートを信頼モードに設定します。このポートでは、これ以上のポリシングは実行されません。 （注） 信頼できるポートを設定した場合、device-roleオプションは使用できません。

	コマンドまたはアクション	目的
ステップ 8	**default** {**device-role** | **trusted-port**} 例： スイッチ(config-dhcp-guard)# **default device-role**	（任意）**default**：コマンドをデフォルトに設定します。
ステップ 9	**do show ipv6 dhcp guard policy** *policy_name* 例： スイッチ(config-dhcp-guard)# **do show ipv6 dhcp guard policy example_policy**	（任意）コンフィギュレーションサブモードを終了せずに IPv6 DHCP のガード ポリシーの設定を表示します。*policy_name* 変数を省略すると、すべての DHCPv6 ポリシーが表示されます。

DHCPv6 ガード設定の例

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy pol1
 device-role server
 match server access-list acl1
 match reply prefix-list abc
 preference min 0
 preference max 255
 trusted-port
interface GigabitEthernet 0/2/0
 switchport
 ipv6 dhcp guard attach-policy pol1 vlan add 1
 vlan 1
  ipv6 dhcp guard attach-policy pol1
show ipv6 dhcp guard policy pol1
```

IPv6 DHCP ガード ポリシーをインターフェイスまたはインターフェイス上の VLAN にアタッチする方法

IPv6 バインディング テーブル コンテンツを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface** Interface_type *stack/module/port*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** Interface_type *stack/module/port* 例： スイッチ(config)# **interface gigabitethernet 1/1/4**	インターフェイスのタイプおよび ID を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] 例： スイッチ(config-if)# **ipv6 dhcp guard attach-policy example_policy** or スイッチ(config-if)# **ipv6 dhcp guard attach-policy example_policy vlan 222,223,224** **or** スイッチ(config-if)# **ipv6 dhcp guard vlan 222, 223,224**	DHCP ガード ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**do show running-config interface** Interface_type *stack/module/port* 例： スイッチ#(config-if)# **do show running-config gig 1/1/4**	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 DHCP ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

EtherChannel インターフェイスまたは VLAN に IPv6 DHCP ガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**

2. **interface range** *Interface_name*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface range** *Interface_name* 例： `スイッチ(config)# interface Po11`	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーション モードを開始します。 ヒント　インターフェイス名やタイプを簡単に参照するには **do show interfaces summary** コマンドを使用します。
ステップ 3	**ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] 例： `スイッチ(config-if-range)# ipv6 dhcp guard attach-policy example_policy` or `スイッチ(config-if-range)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224` **or** `スイッチ(config-if-range)#ipv6 dhcp guard vlan 222, 223,224`	DHCP ガード ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**do show running-config interface***portchannel_interface_name* 例： `スイッチ#(config-if-range)# do show running-config int po11`	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 DHCP ガード ポリシーを全体的に VLAN にアタッチする方法

複数のインターフェイス上の VLAN に IPv6 DHCP のガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vlan configuration** *vlan_list* 例： `スイッチ(config)# vlan configuration 334`	VLAN インターフェイスのコンフィギュレーション モードを開始し、IPv6 スヌーピング ポリシーをアタッチする VLAN を指定します。
ステップ 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] 例： `スイッチ(config-vlan-config)#ipv6 dhcp guard attach-policy example_policy`	すべてのスイッチおよびスタック インターフェイスで、IPv6 ネイバー探索ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。デフォルト ポリシーは、device-role **client**、**no** trusted-port です。
ステップ 4	**do show running-config** 例： `スイッチ#(config-if)# do show running-config`	コンフィギュレーション モードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 ソース ガードの設定方法

手順の概要

1. **enable**
2. **configure terminal**
3. [**no**] **ipv6 source-guard policy** *policy_name*
4. [**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }]
5. **end**

6. **show ipv6 source-guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	[**no**] **ipv6 source-guard policy** *policy_name* 例： `Device(config)# ipv6 source-guard policy example_policy`	IPv6 ソース ガード ポリシー名を指定し、IPv6 ソース ガード ポリシー コンフィギュレーション モードを開始します。
ステップ 4	[**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }] 例： `Device(config-sisf-sourceguard)# deny global-autoconf`	（任意）IPv6 ソース ガード ポリシーを定義します。 • **deny global-autoconf**：自動設定されたグローバル アドレスからのデータ トラフィックを拒否します。これは、リンク上のすべてのグローバル アドレスが DHCP によって割り当てられている際に、管理者が、自己設定されたアドレスを持つホストによるトラフィックの送信をブロックしたい場合に役立ちます。 • **permit link-local**：リンクローカル アドレスから送信されたすべてのデータ トラフィックを許可します。 （注） ソース ガード ポリシーでは trusted オプションはサポートされません。
ステップ 5	**end** 例： `Device(config-sisf-sourceguard)# end`	IPv6 ソース ガード ポリシー コンフィギュレーション モードを終了します。
ステップ 6	**show ipv6 source-guard policy** *policy_name* 例： `Device# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

次のタスク

インターフェイスに IPv6 ソース ガード ポリシーを適用します。

IPv6 ソース ガード ポリシーをインターフェイスにアタッチする方法

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** Interface_type *stack/module/port*
4. **ipv6 source-guard** [**attach-policy** <*policy_name*>]
5. **show ipv6 source-guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** Interface_type *stack/module/port* 例： `Device(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ipv6 source-guard** [**attach-policy** <*policy_name*>] 例： `Device(config-if)# ipv6 source-guard attach-policy example_policy`	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	**show ipv6 source-guard policy** *policy_name* 例： `Device#(config-if)# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 ソース ガードの設定方法

手順の概要

1. **enable**
2. **configure terminal**
3. [**no**] **ipv6 source-guard policy** *policy_name*
4. [**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }]
5. **end**
6. **show ipv6 source-guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	[**no**] **ipv6 source-guard policy** *policy_name* 例： `Device(config)# ipv6 source-guard policy example_policy`	IPv6 ソース ガード ポリシー名を指定し、IPv6 ソース ガード ポリシー コンフィギュレーション モードを開始します。
ステップ 4	[**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }] 例： `Device(config-sisf-sourceguard)# deny global-autoconf`	（任意）IPv6 ソース ガード ポリシーを定義します。 • **deny global-autoconf**：自動設定されたグローバル アドレスからのデータ トラフィックを拒否します。これは、リンク上のすべてのグローバル アドレスが DHCP によって割り当てられている際に、管理者が、自己設定されたアドレスを持つホストによるトラフィックの送信をブロックしたい場合に役立ちます。 • **permit link-local**：リンクローカル アドレスから送信されたすべてのデータ トラフィックを許可します。 （注） ソース ガード ポリシーでは trusted オプションはサポートされません。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `Device(config-sisf-sourceguard)# end`	IPv6 ソース ガード ポリシー コンフィギュレーション モードを終了します。
ステップ 6	**show ipv6 source-guard policy** *policy_name* 例： `Device# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

次のタスク

インターフェイスに IPv6 ソース ガード ポリシーを適用します。

IPv6 ソース ガード ポリシーをインターフェイスにアタッチする方法

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** Interface_type *stack/module/port*
4. **ipv6 source-guard** [**attach-policy** <*policy_name*>]
5. **show ipv6 source-guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** Interface_type *stack/module/port* 例： `Device(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ipv6 source-guard** [**attach-policy** <*policy_name*>] 例：	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用し

	コマンドまたはアクション	目的
	`Device(config-if)# ipv6 source-guard attach-policy example_policy`	ない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	**show ipv6 source-guard policy** *policy_name* 例： `Device#(config-if)# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 ソース ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number*
4. **ipv6 source-guard**［**attach-policy** <*policy_name*> ］
5. **show ipv6 source-guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface port-channel** *port-channel-number* 例： `Device (config)# interface Po4`	インターフェイスのタイプとポート番号を指定し、スイッチをポート チャネル コンフィギュレーション モードにします。
ステップ 4	**ipv6 source-guard**［**attach-policy** <*policy_name*> ］ 例： `Device(config-if) # ipv6 source-guard attach-policy example_policy`	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

	コマンドまたはアクション	目的
ステップ 5	**show ipv6 source-guard policy** *policy_name* 例： `Device(config-if) #show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 プレフィックス ガードの設定方法

（注） プレフィックス ガードが適用されている場合にリンクローカル アドレスから送信されたルーティング プロトコル制御パケットを許可するには、ソースガード ポリシー コンフィギュレーション モードで permit link-local コマンドを有効にします。

手順の概要

1. **enable**
2. **configure terminal**
3. ［**no**］**ipv6 source-guard policy** *source-guard-policy*
4. ［ **no** ］**validate address**
5. **validate prefix**
6. **exit**
7. **show ipv6 source-guard policy**［*source-guard-policy*］

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	［**no**］**ipv6 source-guard policy** *source-guard-policy* 例： `Device(config)# ipv6 source-guard policy my_snooping_policy`	IPv6 ソースガード ポリシー名を定義して、スイッチ統合セキュリティ機能のソースガード ポリシー コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	［ **no** ］**validate address** 例： `Device(config-sisf-sourceguard)# no validate address`	アドレス検証機能を無効にし、IPv6プレフィックスガード機能を設定できるようにします。
ステップ 5	**validate prefix** 例： `Device(config-sisf-sourceguard)# validate prefix`	IPv6 プレフィックスガード動作を実行するよう、IPv6 ソースガードを有効にします。
ステップ 6	**exit** 例： `Device(config-sisf-sourceguard)# exit`	スイッチ統合セキュリティ機能のソースガードポリシー コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 7	**show ipv6 source-guard policy** ［*source-guard-policy*］ 例： `Device# show ipv6 source-guard policy policy1`	IPv6 ソースガード ポリシー設定を表示します。

IPv6プレフィックスガードポリシーをインターフェイスにアタッチする方法

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** Interface_type *stack/module/port*
4. **ipv6 source-guard attach-policy** *policy_name*
5. **show ipv6 source-guard policy** *policy_name*

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface** Interface_type *stack/module/port* 例： `Device(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**ipv6 source-guard attach-policy** *policy_name* 例： `Device(config-if)# ipv6 source-guard attach-policy example_policy`	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	**show ipv6 source-guard policy** *policy_name* 例： `Device(config-if)# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 プレフィックス ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

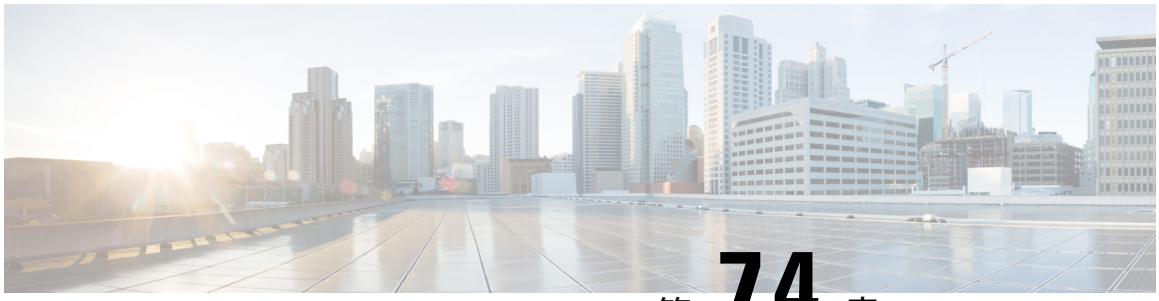
手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number*
4. **ipv6 source-guard** [**attach-policy** <*policy_name*>]
5. **show ipv6 source-guard policy** *policy_name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface port-channel** *port-channel-number* 例： `Device (config)# interface Po4`	インターフェイスのタイプとポート番号を指定し、スイッチをポート チャネル コンフィギュレーション モードにします。

	コマンドまたはアクション	目的
ステップ 4	**ipv6 source-guard** [**attach-policy** <*policy_name*>] 例： `Device(config-if)# ipv6 source-guard attach-policy example_policy`	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 5	**show ipv6 source-guard policy** *policy_name* 例： `Device(config-if)# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。



第 74 章

FIPS の設定

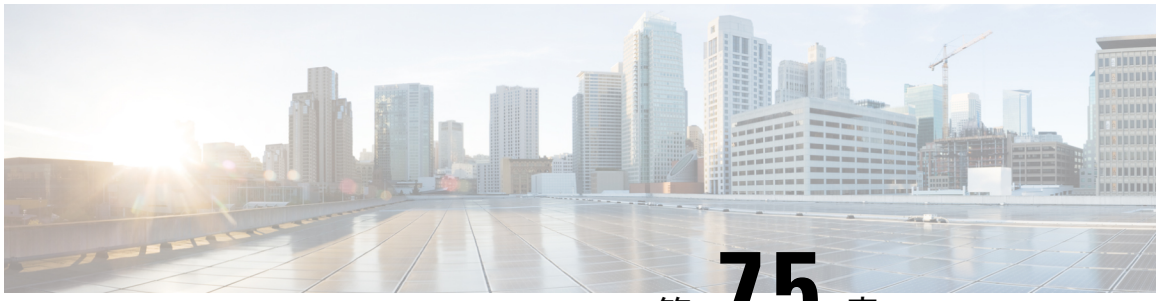
FIPS および共通基準に関する情報

Cisco Catalyst シリーズ スイッチに対する、連邦情報処理標準（FIPS）認証ドキュメントは、次の Web サイトで公開されています。

http://www.cisco.com/web/strategy/government/security_certification/net_business_benefit_seccert_fips140.html

統合された検証証明書とセキュリティ ポリシーのドキュメントを表示するには、証明書の列のリンクをクリックします。セキュリティ ポリシーのドキュメントでは、FIPS の実装、ハードウェアの設置、ファームウェア初期化、および FIPS 操作のためのソフトウェア設定手順について説明します。

Common Criteria はコンピュータ セキュリティ証明書向け国際基準（ISO/IEC 15408）です。この規格は一連の要件、テスト、評価方法から成り、評価のターゲットが特定の保護プロファイルまたはカスタム セキュリティ ターゲットに準拠していることを保証します。詳細については、特定のモデルおよびIOS リリースに対応するセキュリティを目的としたマニュアルを次の URL で参照してください。
http://www.niap-ccevs.org/CCEVS_Products/pcl.cfm?tech_name=Network+Switch



第 75 章

コントロール プレーン ポリシングの設定

コントロール プレーン ポリシングの制約事項

コントロール プレーン ポリシング設定時には次の制約事項が適用されます。

- 次のプロトコルのうち 6 つのみを同時に設定できます：**rip**、**ospf-v6**、**eigrp-v6**、**rip-v6**、**dhcp-snoop-client-to-server**、**dhcp-snoop-server-to-client**、**ndp-router-solicitation**、**ndp-router-advertisement**、**ndp-redirect**、**dhcpv6-client-to-server**、**dhcpv6-server-to-client**、**igrp**。

- **ospf**、**eigrp**、および**ripv2**プロトコルの場合、「**reserve-multicast-group**」オプションを一緒に使用すると、ルータのマルチキャスト MAC 宛ての制御パケットはポリシングされます。

コントロール プレーン ポリシング

事前に設定した一連のプロトコルでコントロール プレーン ポリシング（CoPP）機能を設定し、CPU に着信するトラフィックのフローを制御します。CoPP を使用すると、特定のプロトコル パケットにレート制限を設定できます。これらのパケットに、ポリシーが適用され、定義されているレート制限に準拠するパケットがCPUに着信することがを許可されますCOPPは、スイッチやネットワークのパフォーマンスに影響を与える可能性がある、好ましくないレートでCPUにパケットがルーティングされないように保護します。さらに、CoPP は、サービス拒否（DoS）攻撃からCPUを保護し、ルーティングの安定性、到達可能性、およびパケットの配信を確保します。マルチレイヤ スイッチング QoS CLI を使用すると、レート制限とポリシング パラメータを特定のプロトコルに設定できます。

（注） CoPPは、LANベース、IP Lite、およびIPサービス ライセンスでのみサポートされます。

コントロール プレーン ポリシングの設定

CPUに着信するトラフィックのフローを制御するため、事前に設定した一連のプロトコルでコントロール プレーン ポリシング（CoPP）機能を設定します。

手順の概要

1. **enable**
2. **configure terminal**
3. **mls qos copp protocol** { **autorp-announce** | **autorp-discovery** | **bgp** | **cdp** | **cgmp** | **dai** | **dhcp-snoop-client-to-server** | **dhcp-snoop-server-to-client** | **dhcpv6-client-to-server** | **dhcpv6-server-to-client** | **eigrp** | **eigrp-v6** | **energy-wise** | **igmp-gs-query** | **igmp-leave** | **igmp-query** | **igmp-report** | **igrp** | **ipv6-pimv2** | **lldp** | **mld-gs-query** | **mld-leave** | **mld-query** | **mld-report** | **ndp-redirect** | **ndp-router-advertisement** | **ndp-router-solicitation** | **ospf** | **ospf-v6** | **pimv1** | **pxe** | **rep-hfl** | **reserve-multicast-group** | **rip** | **rip-v6** | **rsvp-snoop** | **stp** } **police** {*pps* | *bps*} **police** *rate*
4. **end**
5. **show mls qos copp protocols**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mls qos copp protocol** { **autorp-announce**	**autorp-discovery**	**bgp**	**cdp**	**cgmp**	**dai**	**dhcp-snoop-client-to-server**	**dhcp-snoop-server-to-client**	**dhcpv6-client-to-server**	**dhcpv6-server-to-client**	**eigrp**	**eigrp-v6**	**energy-wise**	**igmp-gs-query**	**igmp-leave**	**igmp-query**	**igmp-report**	**igrp**	**ipv6-pimv2**	**lldp**	**mld-gs-query**	**mld-leave**	**mld-query**	**mld-report**	**ndp-redirect**		指定したプロトコルのパケット ポリサーを設定します。 さまざまなパラメータについては、『*Consolidated Platform Command Reference, Cisco IOS Release 15.2(4)E*』を参照してください。

	コマンドまたはアクション	目的
	ndp-router-advertisement | **ndp-router-solicitation** | **ospf** | **ospf-v6** | **pimv1** | **pxe** | **rep-hfl** | **reserve-multicast-group** | **rip** | **rip-v6** | **rsvp-snoop** | **stp** } **police** {*pps* | *bps*} **police** *rate* 例： スイッチ (config)# **mls qos copp protocol cdp police bps 10000** スイッチ(config)# **mls qos copp protocol cdp police pps 500**	
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos copp protocols** 例： スイッチ# **show mls qos copp protocols**	設定されているすべてのプロトコルの CoPP パラメータおよびカウンタを表示します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

CoPP の統計情報をクリアするには、**clear copp counters** コマンドを使用します。

例：CoPP の設定

次に、特定のプロトコルのコントロール プレーン ポリシング（CoPP）を有効にする例を示します。

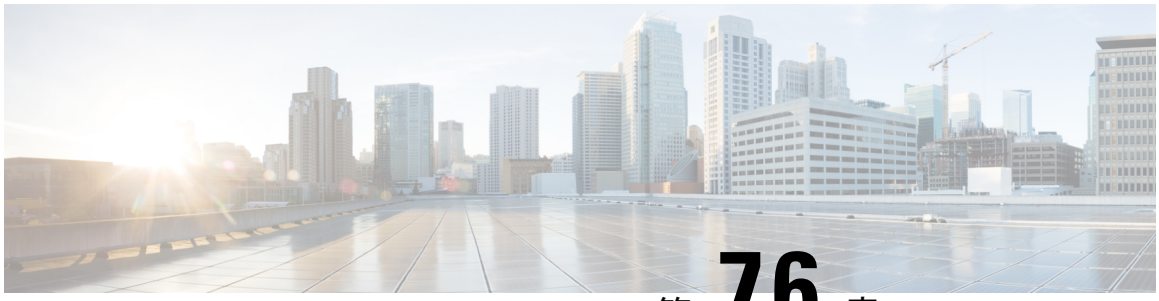
```
Switch (config)# mls qos copp protocol cdp police bps ?
  <8000-2000000000> Bits per second (postfix k, m, g optional; decimal point allowed)
Switch (config)# mls qos copp protocol cdp police bps 10000
Switch(config)# mls qos copp protocol cdp police pps ?
  <100-100000> Packet per second
Switch(config)# mls qos copp protocol cdp police pps 500
```

次に、設定されているすべてのプロトコルの CoPP パラメータとカウンタを表示する例を示します。

```
Switch# show running-config | inc copp
Switch#show running-config | inc copp
mls qos copp protocol rep-hfl police pps 5600
mls qos copp protocol lldp police bps 908900
mls qos copp protocol cdp police pps 3434

/* Copp detailed output */
Switch#show mls qos copp protocols
-------------------------------------------------------------------------------
Protocol                      Mode       PolicerRate         PolicerBurst
InProfilePackets    OutProfilePackets   InProfileBytes      OutProfileBytes
-------------------------------------------------------------------------------
rep-hfl                       pps        5600                5600
0                   0                   0                   0

lldp                          bps        908900              908900
0                   0                   0                   0

cdp                           pps        3434                3434
45172               0                   2891008             0
```

第 X 部

システム管理



第 76 章

システムの管理

デバイスの管理に関する情報

システム日時の管理

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

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

システム クロック

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

システム クロックは、次のソースにより設定できます。

- RTC
- NTP
- 手動設定

システム クロックは、次のサービスに時刻を提供します。

- user **show** コマンド

- ログおよびデバッグ メッセージ

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

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

Real Time Clock（リアルタイム クロック）

リアルタイム クロック（RTC）は、スイッチの現在時刻を追跡します。スイッチはクロッキング パラメータを再設定するまでは GMT 時間に設定された RTC を装備しています。

RTC の利点は次のとおりです。

- RTC はバッテリ電源式です。
- システム時刻は、停電時およびシステム リブート時に保持されます。

RTC と NTP クロックはスイッチに統合されます。NTP を有効にすると、RTC 時間が NTP クロックと定期的に同期化され、精度が保たれます。

Network Time Protocol

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

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

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

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

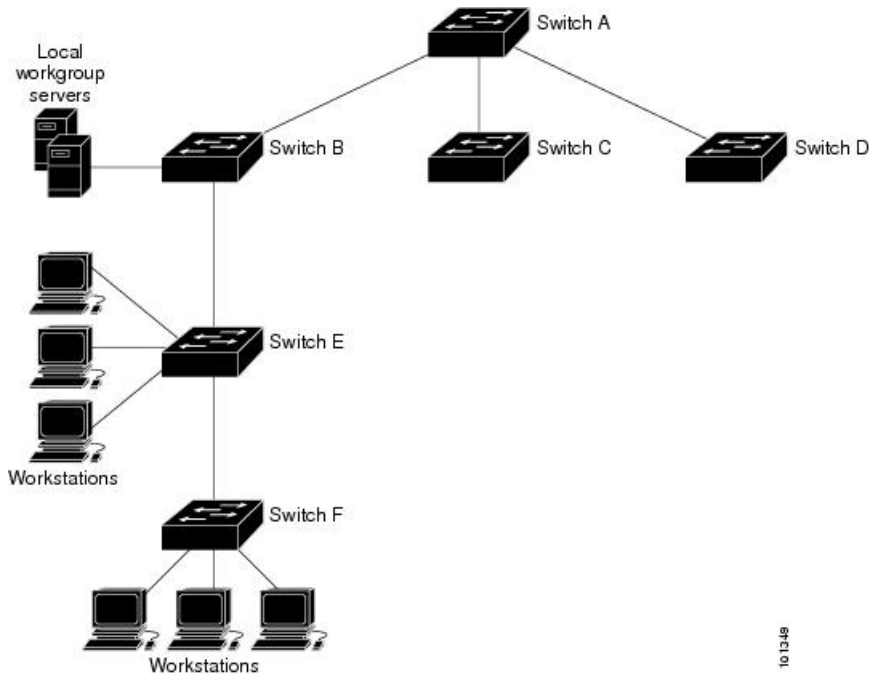
NTPが稼働するデバイス間の通信（アソシエーション）は、通常静的に設定されます。各デバイスには、アソシエーションを作成すべきすべてのデバイスのIPアドレスが与えられます。アソシエーションのペアとなるデバイス間でNTPメッセージを交換することによって、正確な時刻の維持が可能になります。ただし、LAN環境では、代わりにIPブロードキャストメッセージを使用するようにNTPを設定できます。各デバイスを、単にブロードキャストメッセージを送受信するように設定すればよいので、この代替手段によって設定の複雑さが緩和されます。ただし、情報の流れは一方向に限られます。

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

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

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

図 *120*：一般的な *NTP* ネットワークの構成



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

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

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

NTP ストラタム

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

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

NTP アソシエーション

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

NTP セキュリティ

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

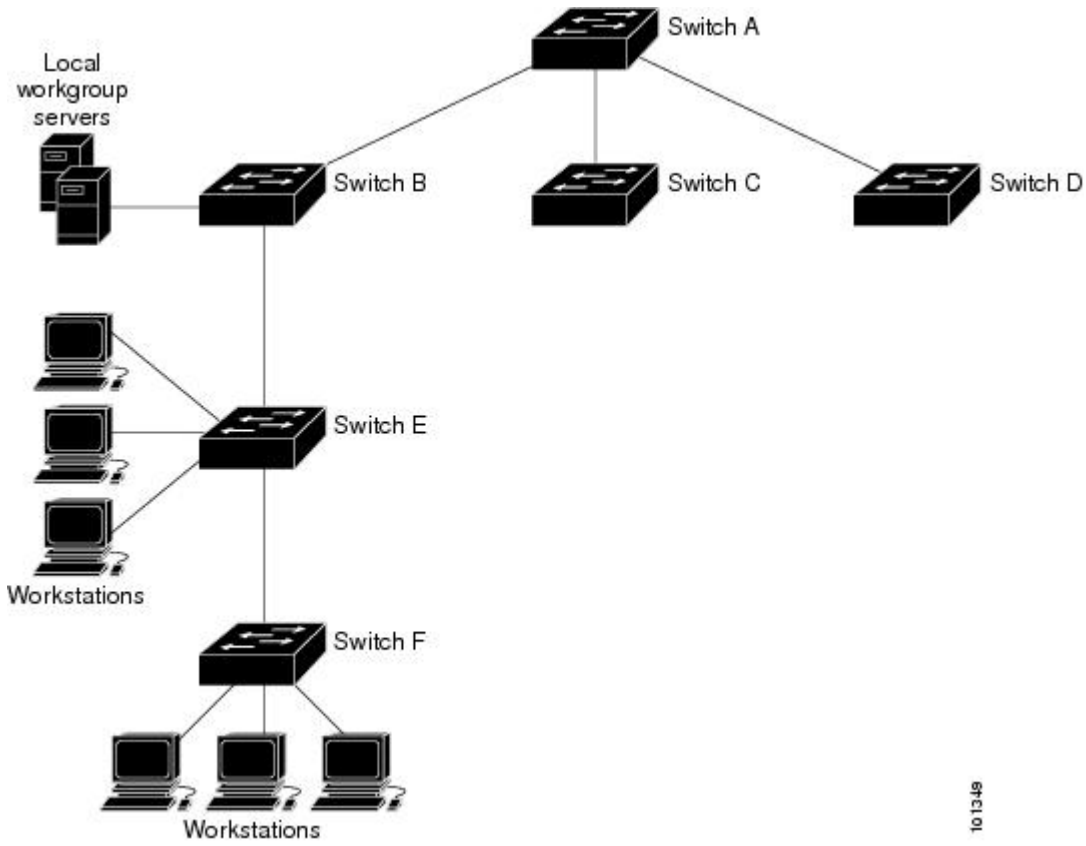
NTP の実装

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

図 *121 :* 一般的な *NTP* ネットワークの構成

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

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



す。

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

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

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

NTP バージョン 4

deviceには、NTP バージョン 4 が実装されています。NTPv4 は NTP バージョン 3 の拡張版です。NTPv4 は IPv4 と IPv6 の両方をサポートし、NTPv3 との下位互換性があります。

NTPv4 は次の互換性を提供します。

- IPv6 のサポート。
- NTPv3 よりさらに向上したセキュリティ。NTPv4 プロトコルは、公開キー暗号化および標準 X509 認証に基づくセキュリティ フレームワークを提供します。

- ネットワークに対する時間分布ヒエラルキーの自動計算。特定のマルチキャスト グループを使用して、NTPv4は、最も低い帯域幅コストで最高の時間精度を達成するサーバのヒエラルキーを自動的に設定します。この機能では、サイトローカル IPv6 マルチキャスト アドレスが活用されます。

NTPv4 の設定の詳細については、『*Cisco IOS IPv6 Configuration Guide, Release 12.4T*』の「*Implementing NTPv4 in IPv6*」の章を参照してください。

システム名およびシステム プロンプト

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

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

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

デフォルトのシステム名とプロンプトの設定

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

DNS

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

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

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

DNS のデフォルト設定値

表 *155 : DNS* のデフォルト設定値

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

ログイン バナー

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

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

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

バナーのデフォルト設定

MoTD およびログイン バナーは設定されません。

MAC アドレス テーブル

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

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

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

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

MAC アドレス テーブルの作成

すべてのポートでサポートされる複数のMACアドレスを使用して、他のネットワーク デバイスにdevice上のすべてのポートを接続できます。deviceは、各ポートで受信するパケットの送信元アドレスを取得し、アドレステーブルにアドレスとそれに関連付けられたポート番号を追加することによって、動的なアドレス指定を行います。ネットワークでデバイスの追加または削除が行われると、deviceによってアドレステーブルが更新され、新しいダイナミック アドレスが追加され、使用されていないアドレスは期限切れになります。

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

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

MAC アドレスおよび VLAN

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

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

MAC アドレス テーブルのデフォルト設定

次の表に、MAC アドレス テーブルのデフォルト設定を示します。

表 *156 : MAC* アドレスのデフォルト設定

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

ARP テーブルの管理

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

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

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

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

デバイスを管理する方法

手動による日付と時刻の設定

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

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

システム クロックの設定

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

システム クロックを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. 次のいずれかを使用します。
   - **clock set** *hh:mm:ss day month year*
   - **clock set** *hh:mm:ss month day year*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	次のいずれかを使用します。 • **clock set** *hh:mm:ss day month year* • **clock set** *hh:mm:ss month day year* 例： スイッチ# **clock set 13:32:00 23 March 2013**	次のいずれかの書式を使ってシステム クロックを手動で設定します。 • *hh:mm:ss*：時間（24 時間形式）、分、秒を指定します。指定された時刻は、設定されたタイム ゾーンに基づきます。 • *day*：月の日で日付を指定します。 • *month*：月を名前で指定します。 • *year*：年を指定します（略式表記で指定しないでください）。

タイム ゾーンの設定

タイム ゾーンを手動で設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **clock timezone** *zone hours-offset* [*minutes-offset*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**clock timezone** *zone hours-offset* [*minutes-offset*] 例： スイッチ(config)# **clock timezone AST -3 30**	時間帯を設定します。 内部時間は、協定世界時（UTC）で維持されるため、このコマンドは表示専用で、時刻を手動で設定するときだけに使用されます。 • *zone*：標準時が適用されているときに表示されるタイムゾーンの名前を入力します。デフォルトは UTC です。 • *hours-offset*：UTC からのオフセット時間数を入力します。 •（任意）*minutes-offset*：UTC からのオフセット分数を入力します。ローカル タイム ゾーンが UTC と 1 時間の差の割合である場合に指定できます。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

夏時間の設定

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

手順の概要

1. **enable**

2. **configure terminal**
3. **clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]]
4. **clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]] 例： スイッチ(config)# **clock summer-time PDT date 10 March 2013 2:00 3 November 2013 2:00**	毎年指定された日に開始および終了する夏時間を設定します。
ステップ 4	**clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]] 例： スイッチ(config)# **clock summer-time PDT recurring 10 March 2013 2:00 3 November 2013 2:00**	毎年指定された日に開始および終了する夏時間を設定します。すべての時刻は、現地のタイム ゾーンを基準にしています。開始時間は標準時を基準にしています。 終了時間は夏時間を基準にしています。夏時間はデフォルトでディセーブルに設定されています。パラメータなしで **clock summer-time** *zone* **recurring** を指定すると、夏時間のルールは米国のルールにデフォルト設定されます。 開始月が終了月より後の場合は、システムでは南半球にいると見なされます。 • *zone*：夏時間が有効な場合に表示される時間帯名（PDT など）を指定します。 • （任意）*week*：月の週（1 ～ 4、**first**、または **last**）を指定します。

	コマンドまたはアクション	目的
		• （任意）*day*：曜日（Sunday、Monday など）を指定します。 • （任意）*month*：月（January、February など）を指定します。 • （任意）*hh:mm*：時および分単位で時間（24 時間形式）を指定します。 • （任意）*offset*：夏時間中に追加する分数を指定します。デフォルトは 60 です。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

ユーザーの居住地域の夏時間が定期的なパターンに従わない（次の夏時間のイベントの正確な日時を設定する）場合は、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **clock summer-time** *zone* **date**[*month date year hh:mm month date year hh:mm* [*offset*]]or**clock summer-time** *zone* **date** [*date month year hh:mm date month year hh:mm* [*offset*]]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**clock summer-time** *zone* **date**[*month date year hh:mm month date year hh:mm* [*offset*]]or**clock summer-time** *zone* **date** [*date month year hh:mm date month year hh:mm* [*offset*]]	最初の日付で夏時間開始の日付を、2 番めの日付で終了の日付を設定します。 夏時間はデフォルトでディセーブルに設定されています。 • *zone* には、夏時間が施行されているときに表示されるタイム ゾーンの名前（たとえばPDT）を入力します。 • （任意）*week* には、月の何週目かを指定します（1 ～ 5、または last）。 • （任意）*day* には、曜日を指定します（Sunday、Monday など）。 • （任意）*month* には、月を指定します（January、February など）。 • （任意）*hh:mm* には、時刻を時間（24 時間形式）と分で指定します。 • （任意）*offset* には、夏時間の間、追加する分数を指定します。デフォルトは 60 です。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例：	入力を確認します。

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

システム名の設定

システム名を手動で設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **hostname** *name*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**hostname** *name* 例： スイッチ(config)# **hostname remote-users**	システム名を設定します。システム名を設定すると、システム プロンプトとしても使用されます。 デフォルト設定は Switch です。 名前は ARPANET ホスト名のルールに従う必要があります。このルールではホスト名は文字で始まり、文字または数字で終わり、その間には文字、数字、

	コマンドまたはアクション	目的
		またはハイフンしか使用できません。名前には 63 文字まで使用できます。
ステップ 4	**end** 例： `remote-users(config)#end` `remote-users#`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

DNS の設定

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

DNS を使用するようにスイッチを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip domain-name** *name*
4. **ip name-server** *server-address1* [*server-address2 ... server-address6*]
5. **ip domain-lookup** [**nsap** | **source-interface** *interface*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip domain-name** *name* 例： スイッチ(config)# **ip domain-name Cisco.com**	非完全修飾ホスト名（ドット付き 10 進表記ドメイン名のない名前）を完成させるためにソフトウェアが使用する、デフォルトのドメイン名を定義します。 ドメイン名を未修飾の名前から区切るために使用される最初のピリオドは入れないでください。 ブート時にはドメイン名は設定されていませんが、deviceの設定が BOOTP または Dynamic Host Configuration Protocol（DHCP）サーバーから行われている場合、BOOTP または DHCP サーバーによってデフォルトのドメイン名が設定されることがあります（この情報がサーバーに設定されている場合）。
ステップ 4	**ip name-server** *server-address1* [*server-address2 ... server-address6*] 例： スイッチ(config)# **ip name-server 192.168.1.100 192.168.1.200 192.168.1.300**	名前とアドレスの解決に使用する 1 つまたは複数のネーム サーバのアドレスを指定します。 最大 6 つのネーム サーバを指定できます。各サーバ アドレスはスペースで区切ります。最初に指定されたサーバが、プライマリ サーバです。deviceは、最初にプライマリ サーバーへ DNS クエリを送信します。そのクエリが失敗した場合は、バックアップ サーバにクエリが送信されます。
ステップ 5	**ip domain-lookup** [**nsap** | **source-interface** *interface*] 例： スイッチ(config)# **ip domain-lookup**	（任意） device上で、DNS に基づくホスト名からアドレスへの変換をイネーブルにします。この機能は、デフォルトでイネーブルにされています。 ユーザのネットワークデバイスが、名前の割り当てを制御できないネットワーク内のデバイスと接続する必要がある場合、グローバルなインターネットのネーミング方式（DNS）を使用して、ユーザのデバ

	コマンドまたはアクション	目的
		イスを一意に識別するデバイス名を動的に割り当てることができます。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

Message-of-the-Day ログイン バナーの設定

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

MOTD ログイン バナーを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **banner motd** *c message c*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

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

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

ログイン バナーの設定

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

ログイン バナーを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **banner login** *c message c*

4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

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

MAC アドレス テーブルの管理

アドレス エージング タイムの変更

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

手順の概要

1. **enable**
2. **configure terminal**
3. **mac address-table aging-time** [*0* | *10-1000000*] [**routed-mac** | **vlan** *vlan-id*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

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

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

MAC アドレス変更通知トラップの設定

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

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server host** *host-addr community-string notification-type* { **informs** | **traps** } {**version** {**1** | **2c** | **3**}} { **vrf** *vrf instance name*}
4. **snmp-server enable traps mac-notification change**
5. **mac address-table notification change**
6. **mac address-table notification change** [**interval** *value*] [**history-size** *value*]
7. **interface** *interface-id*
8. **snmp trap mac-notification change** {**added** | **removed**}
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**snmp-server host** *host-addr community-string notification-type* { **informs** | **traps** } {**version** {**1** | **2c** | **3**}} { **vrf** *vrf instance name*} 例： スイッチ(config)# **snmp-server host 172.20.10.10 traps private mac-notification**	トラップ メッセージの受信側を指定します。 • *host-addr*：NMS の名前またはアドレスを指定します。 • **traps**（デフォルト）：ホストに SNMP トラップを送信します。 • **informs**：ホストに SNMP 情報を送信します。 • **version**：サポートする SNMP バージョンを指定します。informs にはバージョン 1（デフォルト）を使用できません。 • *community-string*：通知処理で送信する文字列を指定します。**snmp-server host** コマンドを使用してこの文字列を設定できますが、この文字列を定義するには、**snmp-server community** グローバル コンフィギュレーション コマンドを使用してから、**snmp-server host** コマンドを使用することを推奨します。 • *notification-type*：**mac-notification** キーワードを使用します。 • **vrf** *vrf* インスタンス名：このホストの VPN ルーティング/転送インスタンスを指定します。
ステップ 4	**snmp-server enable traps mac-notification change** 例： スイッチ(config)# **snmp-server enable traps mac-notification change**	device が MAC アドレス変更通知トラップを送信できるようにします。
ステップ 5	**mac address-table notification change** 例： スイッチ(config)# **mac address-table notification change**	MAC アドレス変更通知機能をイネーブルにします。
ステップ 6	**mac address-table notification change** [**interval** *value*] [**history-size** *value*] 例：	トラップ インターバル タイムと履歴テーブルのサイズを入力します。

	コマンドまたはアクション	目的
	スイッチ(config)# **mac address-table notification change interval 123** スイッチ(config)#**mac address-table notification change history-size 100**	• （任意）**interval** *value*：NMS に生成されるトラップの各セット間の通知トラップインターバルを秒単位で指定します。指定できる範囲は0～2147483647秒です。デフォルトは1秒です。 • （任意）**history-size** *value*：MAC 通知履歴テーブルの最大エントリ数を指定します。指定できる範囲は 0 ～ 500 です。デフォルトは 1 です。
ステップ 7	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/2**	インターフェイス コンフィギュレーション モードを開始し、SNMP MAC アドレス通知トラップをイネーブルにするレイヤ2インターフェイスを指定します。
ステップ 8	**snmp trap mac-notification change** {**added** | **removed**} 例： スイッチ(config-if)# **snmp trap mac-notification change added**	インターフェイス上で MAC アドレス変更通知トラップをイネーブルにします。 • MAC アドレスがインターフェイスに**added**された場合にトラップをイネーブルにします。 • MAC アドレスがインターフェイスに**removed**された場合にトラップをイネーブルにします。
ステップ 9	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 10	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。
ステップ 11	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

MAC アドレス移動通知トラップの設定

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

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

手順の概要

1. **enable**
2. **configure terminal**
3. **snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*
4. **snmp-server enable traps mac-notification move**
5. **mac address-table notification mac-move**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**snmp-server host** *host-addr* {**traps**	**informs**} {**version** {**1**	**2c**	**3**}} *community-string notification-type* 例： `スイッチ(config)# snmp-server host 172.20.10.10 traps private mac-notification`	トラップ メッセージの受信側を指定します。 • *host-addr*：NMS の名前またはアドレスを指定します。 • **traps**（デフォルト）：ホストに SNMP トラップを送信します。 • **informs**：ホストに SNMP 情報を送信します。 • **version**：サポートする SNMP バージョンを指定します。informs にはバージョン 1（デフォルト）を使用できません。 • *community-string*：通知処理で送信する文字列を指定します。**snmp-server host** コマンドを使用してこの文字列を設定できますが、この文字列を定義するには、**snmp-server community** グローバル コンフィギュレーション コマンドを使用

	コマンドまたはアクション	目的
		してから、**snmp-server host** コマンドを使用することを推奨します。 • *notification-type*：**mac-notification** キーワードを使用します。
ステップ 4	**snmp-server enable traps mac-notification move** 例： `スイッチ(config)# snmp-server enable traps mac-notification move`	deviceが NMS に MAC アドレス移動通知トラップを送信できるようにします。
ステップ 5	**mac address-table notification mac-move** 例： `スイッチ(config)# mac address-table notification mac-move`	MACアドレス移動通知機能をイネーブルにします。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

次のタスク

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

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

MAC しきい値通知トラップの設定

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

手順の概要

1. **configure terminal**
2. **snmp-server host** *host-addr* { **traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*
3. **snmp-server enable traps mac-notification threshold**
4. **mac address-table notification threshold**
5. **mac address-table notification threshold** [**limit** *percentage*] | [**interval** *time*]
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**snmp-server host** *host-addr* { **traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type* 例： `スイッチ(config)# snmp-server host 172.20.10.10 traps private mac-notification`	トラップ メッセージの受信側を指定します。 • *host-addr*：NMS の名前またはアドレスを指定します。 • **traps**（デフォルト）：ホストにSNMP トラップを送信します。 • **informs**：ホストに SNMP 情報を送信します。 • **version**：サポートするSNMP バージョンを指定します。informs にはバージョン 1（デフォルト）を使用できません。 • *community-string*：通知処理で送信する文字列を指定します。**snmp-server host** コマンドを使用してこの文字列を設定できますが、この文字列を定義するには、**snmp-server community** グローバル コンフィギュレーション コマンドを使用してから、**snmp-server host** コマンドを使用することを推奨します。 • *notification-type*：**mac-notification** キーワードを使用します。

	コマンドまたはアクション	目的
ステップ 3	**snmp-server enable traps mac-notification threshold** 例： `スイッチ(config)# snmp-server enable traps mac-notification threshold`	NMS への MAC しきい値通知トラップをイネーブルにします。
ステップ 4	**mac address-table notification threshold** 例： `スイッチ(config)# mac address-table notification threshold`	MAC アドレスしきい値通知機能をイネーブルにします。
ステップ 5	**mac address-table notification threshold** [**limit** *percentage*]	[**interval** *time*] 例： `スイッチ(config)# mac address-table notification threshold interval 123` `スイッチ(config)# mac address-table notification threshold limit 78`	MAC アドレスしきい値使用状況モニタリングのしきい値を入力します。 • （任意）**limit** *percentage*：MAC アドレステーブルの使用率を指定します。有効値は 1 ～ 100% ですデフォルト値は 50% です。 • （任意）**interval** *time*：通知の間隔を指定します。有効値は 120 秒以上です。デフォルトは 120 秒です。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

スタティック アドレス エントリの追加および削除

スタティック アドレスを追加するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* 例： `Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/0/1`	MAC アドレス テーブルにスタティック アドレスを追加します。 • *mac-addr*：アドレス テーブルに追加する宛先 MAC ユニキャスト アドレスを指定します。この宛先アドレスを持つパケットが指定した VLAN に着信すると、指定したインターフェイスに転送されます。 • *vlan-id*：指定されたMACアドレスを持つパケットを受信するVLANを指定します。指定できるVLAN ID の範囲は 1 ～ 4094 です。 • *interface-id*：受信パケットが転送されるインターフェイスを指定します。有効なインターフェイスは、物理ポートまたはポート チャネルです。スタティック マルチキャスト アドレスの場合、複数のインターフェイス ID を入力できます。スタティック ユニキャスト アドレスの場合、インターフェイスは同時に 1 つしか入力できません。ただし、同じMACアドレスおよびVLAN IDを指定すると、コマンドを複数回入力できます。
ステップ 4	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。
ステップ 5	**show running-config** 例：	入力を確認します。

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

ユニキャスト MAC アドレス フィルタリングの設定

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

手順の概要

1. **enable**
2. **configure terminal**
3. **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** 例： スイッチ(config)# **mac address-table static c2f3.220a.12f4 vlan 4 drop**	ユニキャスト MAC アドレス フィルタリングをイネーブルにし、deviceが指定した送信元または宛先ユニキャスト スタティック アドレスを持つパケットをドロップするように設定します。

	コマンドまたはアクション	目的
		• *mac-addr*：送信元または宛先ユニキャストMACアドレス（48ビット）を指定します。このMACアドレスを持つパケットはドロップされます。 • *vlan-id*：指定されたMACアドレスを持つパケットを受信するVLANを指定します。指定できるVLAN IDの範囲は1 ～ 4094です。
ステップ4	**end** 例： `スイッチ(config)# end`	特権EXECモードに戻ります。
ステップ5	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

デバイスのモニターリングおよび保守の管理

コマンド	目的
clear mac address-table dynamic	すべてのダイナミックエントリを削除します。
clear mac address-table dynamic address *mac-address*	特定のMACアドレスを削除します。
clear mac address-table dynamic interface *interface-id*	指定された物理ポートまたはポートチャネル上のすべてのアドレスを削除します。
clear mac address-table dynamic vlan *vlan-id*	指定されたVLAN上のすべてのアドレスを削除します。
show clock [*detail*]	時刻と日付の設定を表示します。

コマンド	目的
show ip igmp snooping groups	すべての VLAN または指定された VLAN に対するレイヤ 2 マルチキャスト エントリを表示します。
show mac address-table address *mac-address*	指定された MAC アドレスの MAC アドレス テーブル情報を表示します。
show mac address-table aging-time	すべての VLAN または指定された VLAN のエージング タイムを表示します。
show mac address-table count	すべての VLAN または指定された VLAN で存在しているアドレス数を表示します。
show mac address-table dynamic	ダイナミック MAC アドレス テーブル エントリのみを表示します。
show mac address-table interface *interface-name*	指定されたインターフェイスの MAC アドレス テーブル情報を表示します。
show mac address-table move update	MAC アドレステーブル移動更新情報を表示します。
show mac address-table multicast	マルチキャストの MAC アドレスのリストを表示します。
show mac address-table notification {change | mac-move | threshold}	MAC 通知パラメータおよび履歴テーブルを表示します。
show mac address-table secure	セキュア MAC アドレスを表示します。
show mac address-table static	スタティック MAC アドレス テーブル エントリだけを表示します。
show mac address-table vlan *vlan-id*	指定された VLAN の MAC アドレス テーブル情報を表示します。

デバイス管理の設定例

例：システム クロックの設定

次の例は、システム クロックを手動で設定する方法を示しています。

```
スイッチ# clock set 13:32:00 23 July 2013
```

例：サマー タイムの設定

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

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

次に、サマータイムの開始日と終了日を設定する例を示します。

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

例：MOTD バナーの設定

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

```
スイッチ(config)# banner motd #

This is a secure site. Only authorized users are allowed.
For access, contact technical support.

#

スイッチ(config)#
```

次に、前の設定により表示されたバナーの例を示します。

```
Unix> telnet 192.0.2.15

Trying 192.0.2.15...

Connected to 192.0.2.15.

Escape character is '^]'.

This is a secure site. Only authorized users are allowed.

For access, contact technical support.

User Access Verification

Password:
```

例：ログイン バナーの設定

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

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

例：MAC アドレス変更通知トラップの設定

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

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

例：MAC しきい値通知トラップの設定

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

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

例：MAC アドレス テーブルへのスタティック アドレスの追加

次の例では、MAC アドレス テーブルにスタティック アドレス c2f3.220a.12f4 を追加する方法を示します。VLAN 4 でこの MAC アドレスを宛先アドレスとして持つパケットを受信すると、パケットは指定されたポートに転送されます。

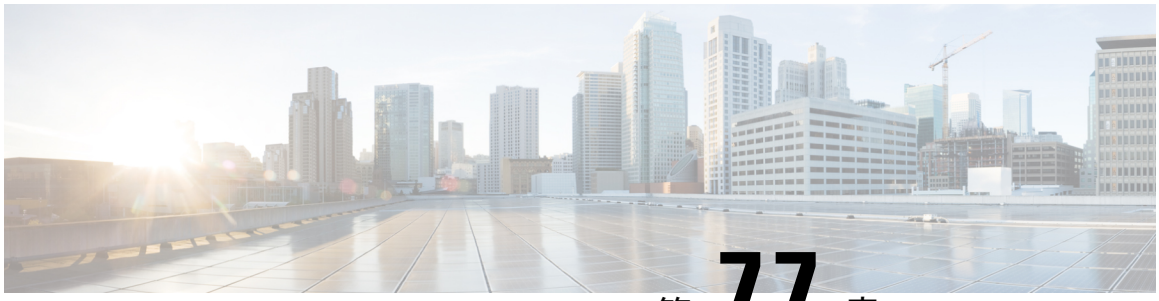
（注） 複数のインターフェイスに同じ静的MACアドレスを関連付けることはできません。コマンドを別のインターフェイスで再度実行すると、新しいインターフェイス上で静的 MAC アドレスが上書きされます。

```
スイッチ(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet
 1/1/1
```

例：ユニキャスト MAC アドレス フィルタリングの設定

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

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



第 77 章

デバイスのセットアップ設定の実行

デバイスセットアップ設定の実行に関する情報

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

ブート プロセス

deviceを起動するには、スタートアップガイドやハードウェア設置ガイドの手順に従い、deviceを設置して電源をオンにし、deviceの初期設定（IP アドレス、サブネットマスク、デフォルトゲートウェイ、シークレット、Telnet パスワードなど）を行う必要があります。

ブートローダ ソフトウェアは、通常の起動プロセスを実行します。これには、次のアクティビティが含まれています。

- バンドルまたはインストール パッケージ セットでブート可能（基本）パッケージを検索します。
- 下位レベルの CPU 初期化を行います。CPU レジスタを初期化することにより、物理メモリがマッピングされる場所、容量、速度などを制御します。
- CPU サブシステムの電源投入時セルフ テスト（POST）を実行し、システム DRAM をテストします。
- システム ボード上のファイル システムを初期化します。
- デフォルトのオペレーティング システム ソフトウェア イメージをメモリにロードし、deviceを起動します。

ブートローダによってフラッシュ ファイル システムにアクセスしてから、オペレーティング システムをロードします。ブートローダの使用目的は通常、オペレーティング システムのロード、展開、および起動に限定されます。オペレーティング システムが CPU を制御できるようになると、ブートローダは、次にシステムがリセットされるか電源が投入されるまでは非アクティブになります。

また、オペレーティング システムが使用不可能になるほどの重大な障害が発生した場合は、ブートローダはシステムにトラップドアからアクセスします。トラップドアからシステムへアクセスすることで、必要に応じて、フラッシュ ファイル システムのフォーマット、XMODEM プロトコルを使用したオペレーティング システムのソフトウェア イメージの再インストール、失われたパスワードの回復、そして最終的にオペレーティング システムの再起動ができます。

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

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

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

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

Devices 情報の割り当て

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

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

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

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

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

deviceの設定手順を熟知している経験豊富なユーザーの場合は、deviceを手動で設定してください。それ以外のユーザは、「ブート プロセス」で説明したセットアップ プログラムを使用してください。

デフォルトのスイッチ情報

表 *157:* デフォルトのスイッチ情報

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

DHCP ベースの自動設定の概要

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

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

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

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

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

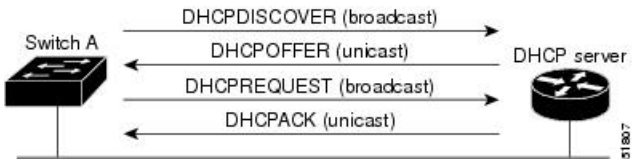
DHCP ベースの自動設定は、deviceの BOOTP クライアント機能に代わるものです。

DHCP クライアントの要求プロセス

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

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

図 *122 : DHCP* クライアント/サーバ間のメッセージ交換



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

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

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

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

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

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

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

クライアントにデフォルトのホスト名がある場合（**hostname** *name* グローバル コンフィギュレーション コマンドを設定していないか、**no hostname** グローバル コンフィギュレーション コマンドを使用してホスト名を削除していない場合）は、**ip address dhcp** インターフェイス コンフィギュレーション コマンドを入力すると、DHCPのホスト名オプションがパケットに含まれません。この場合、インターフェイスの IP アドレスを取得中にクライアントが DHCP との相互作用で DHCP ホスト名オプションを受信した場合、クライアントは DHCP ホスト名オプションを受け入れて、システムに設定済みのホスト名があることを示すフラグが設定されます。

DHCP ベースの自動設定およびイメージ アップデート

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

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

DHCP ベースの自動設定の制約事項

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

DHCP 自動設定

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

DHCP 自動イメージ アップデート

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

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

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

device をネットワークに設置すると、自動イメージ アップデート機能が開始します。ダウンロードされたコンフィギュレーション ファイルは device の実行コンフィギュレーションに保存

され、新しいイメージがダウンロードされてdeviceにインストールされます。deviceを再起動すると、このコンフィギュレーションがdeviceのコンフィギュレーションに保存されます。

DHCP サーバ設定時の注意事項

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

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

DNS サーバの目的

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

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

DNS サーバーは、device と同じ LAN 上に配置することも、別の LAN 上に配置することもできます。DNS サーバーが別の LAN 上に存在する場合、deviceはルータを介して DNS サーバーにアクセスできなければなりません。

コンフィギュレーション ファイルの入手方法

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

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

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

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

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

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

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

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

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

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

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

環境変数の制御方法

通常動作の device では、コンソール接続のみを通じてブートローダモードを開始します。スイッチの電源コードを取り外してから、もう一度電源コードを接続します。ブート ローダ スイッチのプロンプトが表示されるまで [MODE] を押し続けます。

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

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

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

環境変数には 2 種類のデータが保存されます。

- Cisco IOS コンフィギュレーション ファイルを読み取らないコードを制御するデータ。たとえば、ブートローダの機能を拡張したり、パッチを適用したりするブートローダ ヘルパー ファイルの名前は、環境変数として保存できます。
- Cisco IOS コンフィギュレーション ファイルを読み取るコードを制御するデータ。たとえば、Cisco IOS コンフィギュレーション ファイル名は環境変数として保存できます。

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

一般的な環境変数

この表では、最も一般的な環境変数の機能について説明します。

表 *158 :* 一般的な環境変数

変数	ブートローダ コマンド	Cisco IOS グローバル コンフィギュレーション コマンド
BOOT	**set BOOT** *filesystem* **:/** *file-url ...* 自動起動時にロードして実行を試みる、セミコロンで区切られた実行可能ファイルのリスト。BOOT 環境変数が設定されていない場合、システムは、フラッシュ ファイル システム全体に再帰的な縦型検索を行って、最初に検出された実行可能イメージをロードして実行を試みます。BOOT 変数が設定されていても、指定されたイメージをロードできなかった場合、システムはフラッシュ ファイル システムで最初に検出した起動可能なファイルを起動しようとします。	**boot system** {*filesystem* **:** */file-url ...* 次の起動時にロードする Cisco IOS イメージと、イメージがロードされるスタック メンバーを指定します。このコマンドは、BOOT 環境変数の設定を変更します。

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

変数	ブートローダ コマンド	Cisco IOS グローバル コンフィギュレーション コマンド
BAUD	**set BAUD** *baud-rate*	**line console 0** **speed***speed-value* ボー レートを設定します。
ENABLE_BREAK	**set ENABLE_BREAK yes/no**	**boot enable-break switch yes/no** このコマンドは、**ENABLE_BREAK** が **yes** に設定されている場合にフラッシュ ファイルシステムを初期化するときに発行できます。

TFTP の環境変数

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

表 *159 : TFTP* の環境変数

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

ソフトウェア イメージのリロードのスケジューリング

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

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

リロード オプションには以下のものがあります。

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

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

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

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

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

デバイスセットアップ設定の実行方法

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

DHCP 自動設定（コンフィギュレーション ファイルだけ）の設定

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

手順の概要

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **exit**
8. **tftp-server flash**:*filename.text*
9. **interface** *interface-id*
10. **no switchport**
11. **ip address** *address mask*
12. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**ip dhcp pool** *poolname* 例： `スイッチ(config)# ip dhcp pool pool`	DHCP サーバ アドレス プールの名前を作成し、DHCP プール コンフィギュレーション モードを開始します。
ステップ 3	**boot** *filename* 例： `スイッチ(dhcp-config)# boot config-boot.text`	ブートイメージとして使用されるコンフィギュレーション ファイルの名前を指定します。
ステップ 4	**network** *network-number mask prefix-length* 例： `スイッチ(dhcp-config)# network 10.10.10.0 255.255.255.0`	DHCP アドレス プールのサブネット ネットワーク番号およびマスクを指定します。 （注） プレフィックス長は、アドレス プレフィックスを構成するビット数を指定します。プレフィックスは、クライアントのネットワーク マスクを指定する二者択一の方法です。プレフィックス長は、スラッシュ（/）で開始する必要があります。

	コマンドまたはアクション	目的
ステップ 5	**default-router** *address* 例： `スイッチ(dhcp-config)# default-router 10.10.10.1`	DHCP クライアントのデフォルト ルータの IPアドレスを指定します。
ステップ 6	**option 150** *address* 例： `スイッチ(dhcp-config)# option 150 10.10.10.1`	TFTP サーバの IP アドレスを指定します。
ステップ 7	**exit** 例： `スイッチ(dhcp-config)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 8	**tftp-server flash**:*filename.text* 例： `スイッチ(config)# tftp-server flash:config-boot.text`	TFTP サーバ上のコンフィギュレーション ファイルを指定します。
ステップ 9	**interface** *interface-id* 例：	コンフィギュレーション ファイルを受信するクライアントのアドレスを指定します。
ステップ 10	**no switchport** 例： `スイッチ(config-if)# no switchport`	インターフェイスをレイヤ 3 モードにします。
ステップ 11	**ip address** *address mask* 例： `スイッチ(config-if)# ip address 10.10.10.1 255.255.255.0`	IP アドレスとインターフェイスのマスクを指定します。
ステップ 12	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。

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

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

始める前に

最初にdeviceにアップロードするテキスト ファイル（たとえば、autoinstall_dhcp）を作成します。テキストファイルに、ダウンロードするイメージの名前を指定します（たとえば、c3750e-ipservices-mz.122-44.3.SE.tarc3750x-ipservices-mz.122-53.3.SE2.tar）。 このイメージは、bin ファイルでなく、tar ファイルである必要があります。

手順の概要

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **option 125** *hex*
8. **copy tftp flash** *filename.txt*
9. **copy tftp flash** *imagename.bin*
10. **exit**
11. **tftp-server flash:** *config.text*
12. **tftp-server flash:** *imagename.bin*
13. **tftp-server flash:** *filename.txt*
14. **interface** *interface-id*
15. **no switchport**
16. **ip address** *address mask*
17. **end**
18. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**ip dhcp pool** *poolname* 例： `スイッチ(config)# ip dhcp pool pool1`	DHCP サーバ アドレス プールの名前を作成し、DHCP プール コンフィギュレーション モードを開始します。
ステップ 3	**boot** *filename* 例： `スイッチ(dhcp-config)# boot config-boot.text`	ブート イメージとして使用されるファイルの名前を指定します。
ステップ 4	**network** *network-number mask prefix-length* 例： `スイッチ(dhcp-config)# network 10.10.10.0 255.255.255.0`	DHCP アドレス プールのサブネット ネットワーク番号およびマスクを指定します。 （注） プレフィックス長は、アドレス プレフィックスを構成するビット数を指定します。プレフィックスは、クライアントのネットワーク マスクを指定する二者択一の方法です。プレフィックス長は、スラッシュ（/）で開始する必要があります。
ステップ 5	**default-router** *address* 例： `スイッチ(dhcp-config)# default-router 10.10.10.1`	DHCP クライアントのデフォルト ルータの IPアドレスを指定します。
ステップ 6	**option 150** *address* 例： `スイッチ(dhcp-config)# option 150 10.10.10.1`	TFTP サーバの IP アドレスを指定します。
ステップ 7	**option 125** *hex* 例： `スイッチ(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370`	イメージ ファイルのパスを記述したテキスト ファイルのパスを指定します。
ステップ 8	**copy tftp flash** *filename.txt* 例：	deviceに、テキスト ファイルをアップロードします。

	コマンドまたはアクション	目的
	スイッチ(config)# **copy tftp flash image.bin**	
ステップ 9	**copy tftp flash** *imagename.bin* 例： スイッチ(config)# **copy tftp flash image.bin**	deviceに、新しいイメージの tar ファイルをアップロードします。
ステップ 10	**exit** 例： スイッチ(dhcp-config)# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 11	**tftp-server flash:** *config.text* 例： スイッチ(config)# **tftp-server flash:config-boot.text**	TFTP サーバ上の Cisco IOS コンフィギュレーション ファイルを指定します。
ステップ 12	**tftp-server flash:** *imagename.bin* 例： スイッチ(config)# **tftp-server flash:image.bin**	TFTP サーバ上のイメージ名を指定します。
ステップ 13	**tftp-server flash:** *filename.txt* 例： スイッチ(config)# **tftp-server flash:boot-config.text**	ダウンロードするイメージ ファイルの名前を記述したテキスト ファイルを指定します。
ステップ 14	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/4**	コンフィギュレーション ファイルを受信するクライアントのアドレスを指定します。
ステップ 15	**no switchport** 例： スイッチ(config-if)# **no switchport**	インターフェイスをレイヤ 3 モードにします。

	コマンドまたはアクション	目的
ステップ 16	**ip address** *address mask* 例： スイッチ(config-if)# **ip address 10.10.10.1 255.255.255.0**	IP アドレスとインターフェイスのマスクを指定します。
ステップ 17	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 18	**copy running-config startup-config** 例： スイッチ(config-if)# **end**	（任意）コンフィギュレーション ファイルに設定を保存します。

DHCP サーバからファイルをダウンロードするクライアントの設定

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

手順の概要

1. **configure terminal**
2. **boot host dhcp**
3. **boot host retry timeout** *timeout-value*
4. **banner config-save ^C** *warning-message* **^C**
5. **end**
6. **show boot**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**boot host dhcp** 例： `スイッチ(conf)# boot host dhcp`	保存されているコンフィギュレーションで自動設定をイネーブルにします。
ステップ 3	**boot host retry timeout** *timeout-value* 例： `スイッチ(conf)# boot host retry timeout 300`	（任意）システムがコンフィギュレーションファイルをダウンロードしようとする時間を設定します。 （注） タイムアウトを設定しないと、システムは無期限に DHCP サーバから IP アドレスを取得しようとします。
ステップ 4	**banner config-save ^C** *warning-message* **^C** 例： `スイッチ(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to No longer Automatically Download Configuration Files at Reboot^C`	（任意）コンフィギュレーション ファイルを NVRAM に保存しようとするときに表示される警告メッセージを作成します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show boot** 例： `スイッチ# show boot`	設定を確認します。

複数の SVI への IP 情報の手動割り当て

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

手順の概要

1. **configure terminal**
2. **interface vlan** *vlan-id*
3. **ip address** *ip-address subnet-mask*
4. **exit**
5. **ip default-gateway** *ip-address*

6. **end**
7. **show interfaces vlan** *vlan-id*
8. **show ip redirects**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface vlan** *vlan-id* 例： `スイッチ(config)# interface vlan 99`	インターフェイス コンフィギュレーション モードを開始し、IP 情報を割り当てる VLAN を入力します。指定できる範囲は 1 ～ 4094 です。
ステップ 3	**ip address** *ip-address subnet-mask* 例： `スイッチ(config-vlan)# ip address 10.10.10.2 255.255.255.0`	IP アドレスとサブネット マスクを入力します。
ステップ 4	**exit** 例： `スイッチ(config-vlan)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 5	**ip default-gateway** *ip-address* 例： `スイッチ(config)# ip default-gateway 10.10.10.1`	deviceに直接接続しているネクスト ホップのルータ インターフェイスのIPアドレスを入力します。このスイッチにはデフォルト ゲートウェイが設定されています。デフォルト ゲートウェイは、deviceスイッチから宛先 IP アドレスを取得していない IP パケットを受信します。 デフォルト ゲートウェイが設定されると、deviceは、ホストが接続する必要のあるリモート ネットワークに接続できます。 （注） IP でルーティングするようにdeviceを設定した場合、デフォルト ゲートウェイの設定は不要です。

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

NVRAM バッファ サイズの設定

デフォルトの NVRAM バッファ サイズは 512 KB です。コンフィギュレーション ファイルが大きすぎて NVRAM に保存できない場合があります。一般的に、この状態はスイッチ スタック内に多くのスイッチがある場合に発生します。より大きいコンフィギュレーションファイルをサポートできるように、NVRAM バッファのサイズを設定できます。新しい NVRAM バッファ サイズは、現在および新しいすべてのメンバ スイッチに同期されます。

（注） NVRAMバッファ サイズを設定後、スイッチまたはスイッチスタックをリロードします。

スイッチをスタックに追加し、NVRAMサイズが異なる場合、新しいスイッチはスタックに同期化し、自動的にリロードされます。

手順の概要

1. **configure terminal**
2. **boot buffersize** *size*
3. **end**
4. **show boot**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**boot buffersize** *size* 例： `スイッチ(config)# boot buffersize 524288`	NVRAM のバッファ サイズを KB 単位で設定します。*size* の有効な範囲は、4096 ～ 1048576 です。
ステップ 3	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show boot** 例： `スイッチ# show boot`	設定を確認します。

デバイスのスタートアップ コンフィギュレーションの変更

システム コンフィギュレーションを読み書きするためのファイル名の指定

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

始める前に

このタスクではスタンドアロンの device を使用します。

手順の概要

1. **configure terminal**
2. **boot config-file** file name
3. **end**
4. **show boot**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**boot config-file** file name 例： `Switch(config)# boot config-file config.text`	次回の起動時に読み込むコンフィギュレーション ファイルを指定します。 *file-url*：パス（ディレクトリ）およびコンフィギュレーション ファイル名。 ファイル名およびディレクトリ名は、大文字と小文字を区別します。
ステップ 3	**end** 例： `Switch(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show boot** 例： `Switch# show boot`	入力を確認します。 **boot** グローバル コンフィギュレーション コマンドによって、CONFIG_FILE 環境変数の設定が変更されます。
ステップ 5	**copy running-config startup-config** 例： `Switch# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

スイッチの手動による起動

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

始める前に

このタスクのスタンドアロン スイッチを使用します。

手順の概要

1. **configure terminal**
2. **boot manual**
3. **end**

4. **show boot**
5. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**boot manual** 例： `スイッチ(config)# boot manual`	次回の起動時に、スイッチを手動で起動できるようにします。
ステップ 3	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show boot** 例： `スイッチ# show boot`	入力を確認します。 **boot manual** グローバルコマンドは、MANUAL_BOOT 環境変数の設定を変更します。 次回、システムを再起動した際には、スイッチはブートローダ モードになり、ブートローダ モードであることが *switch:* プロンプトによって示されます。システムを起動するには、**boot** *filesystem:/file-url* ブートローダコマンドを使用します。 • *filesystem*：システム ボードのフラッシュ デバイスに flash: を使用します。 `Switch: boot flash:` • *file-url*：パス（ディレクトリ）および起動可能なイメージの名前を指定します。 ファイル名およびディレクトリ名は、大文字と小文字を区別します。
ステップ 5	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

ソフトウェア イメージのリロードのスケジュール設定

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

手順の概要

1. **configure terminal**
2. **copy running-config startup-config**
3. **reload in** [*hh*:]*mm* [*text*]
4. **reload at** *hh*: *mm* [*month day* | *day month*] [*text*]
5. **reload cancel**
6. **show reload**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**copy running-config startup-config** 例： **copy running-config startup-config**	**reload** コマンドを使用する前に、device の設定情報をスタートアップ コンフィギュレーションに保存します。
ステップ 3	**reload in** [*hh*:]*mm* [*text*] 例： スイッチ(config)# **reload in 12** System configuration has been modified. Save? [yes/no]: **y**	指定した分数、または時間および分数が経過したときに、ソフトウェアがリロードされるようにスケジュールを設定します。リロードは、約 24 日以内に実行する必要があります。最大 255 文字で、リロードの理由を指定できます。
ステップ 4	**reload at** *hh*: *mm* [*month day*	*day month*] [*text*] 例：	リロードを実行する時間を、時間数と分数で指定します。

	コマンドまたはアクション	目的
	スイッチ(config)# **reload at 14:00**	（注） **at** キーワードを使用するのは、device システムクロックが（ネットワークタイムプロトコル、ハードウェアカレンダー、または手動で）設定されている場合だけです。時刻は、deviceに設定されたタイム ゾーンに基づきます。リロードが複数のdevicesで同時に行われるようにスケジューリングするには、各deviceの時間が NTP と同期している必要があります。
ステップ 5	**reload cancel** 例： スイッチ(config)# **reload cancel**	以前にスケジューリングされたリロードをキャンセルします。
ステップ 6	**show reload** 例： **show reload**	以前deviceにスケジューリングされたリロードに関する情報、またはリロードがスケジューリングされているかを表示します。

デバイスのセットアップ設定のモニターリング

例：デバイス実行コンフィギュレーションの確認

```
スイッチ# show running-config
Building configuration...

Current configuration: 1363 bytes
!
version 12.4
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Stack1
!
enable secret 5 $1$ej9.$DMUvAUnZOAmvmgqBEzIxE0
!
.
<output truncated>
.
interface gigabitethernet6/0/2
mvr type source

<output truncated>

...!
```

```
interface VLAN1
 ip address 172.20.137.50 255.255.255.0
 no ip directed-broadcast
!
ip default-gateway 172.20.137.1 !
!
snmp-server community private RW
snmp-server community public RO
snmp-server community private@es0 RW
snmp-server community public@es0 RO
snmp-server chassis-id 0x12
!
end
```

例：ソフトウェア インストールの表示

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

```
switch# boot flash:/c3560cx-universalk9-mz.152-3.E/c3560cx-universalk9-tar.152-3.E.bin
```

デバイス のセットアップを実行する場合の設定例

例：DHCP サーバーとしてのデバイスの設定

```
スイッチ# configure terminal
スイッチ(config)# ip dhcp pool pool1
スイッチ(dhcp-config)# network 10.10.10.0 255.255.255.0
スイッチ(dhcp-config)# boot config-boot.text
スイッチ(dhcp-config)# default-router 10.10.10.1
スイッチ(dhcp-config)# option 150 10.10.10.1
スイッチ(dhcp-config)# exit
スイッチ(config)# tftp-server flash:config-boot.text
スイッチ(config)# interface gigabitethernet 1/0/4
スイッチ(config-if)# no switchport
スイッチ(config-if)# ip address 10.10.10.1 255.255.255.0
スイッチ(config-if)# end
```

例：DHCP 自動イメージ アップデートの設定

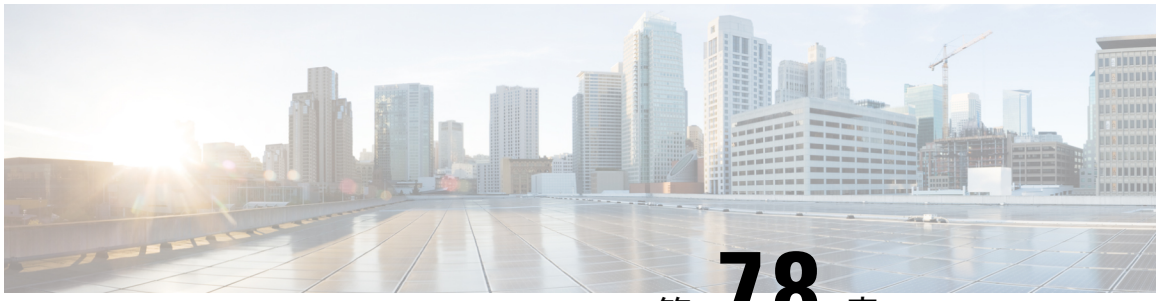
例：DHCP サーバーから設定をダウンロードするための デバイス の設定

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

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

例：NVRAM バッファ サイズの設定

```
スイッチ# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
スイッチ(config)# boot buffersize 600000
スイッチ(config)# end
スイッチ# show boot
BOOT path-list      :
Config file         : flash:/config.text
Private Config file : flash:/private-config.text
Enable Break        : no
Manual Boot         : no
HELPER path-list    :
Auto upgrade        : yes
Auto upgrade path   :
NVRAM/Config file
      buffer size:   600000
Timeout for Config
          Download:    300 seconds
Config Download
       via DHCP:       enabled (next boot: enabled)
スイッチ#
```



第 78 章

RTU ライセンスの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

RTU ライセンスの設定に関する制約事項

次に、RTU ライセンスの設定および使用に関する制約事項を示します。

- AP-Count ライセンスは注文が可能で、スイッチ上で事前にアクティブ化できます。
- イメージ ベースのライセンスは、アップグレードできます。AP-Count ライセンスは非アクティブ化したり、スイッチとコントローラとの間で移動したりできます。
- ライセンスをアクティブ化するには、新しいライセンス レベルを設定した後にスイッチを再起動する必要があります。AP-Count ライセンスをアクティブ化するために再起動する必要はありません。

- 再起動後に、期限が切れた評価ライセンスを再びアクティブ化することはできません。
- スイッチ スタックのスタック メンバは同一のライセンス レベルを実行する必要があります。ライセンス レベルが異なる場合、レベルを変更してスタックのアクティブなスイッチから再起動するまでは、スイッチはスタックに参加しません。
- 追加 AP-Count ライセンスは、工場出荷時にインストールされます。

RTU ライセンスの設定に関する情報

Right-To-Use ライセンス

Right-To-Use（RTU）ライセンスでは、特定のライセンス タイプおよびレベルを注文してアクティブ化し、ライセンスの使用状況をスイッチで管理することができます。注文できる期間別のライセンスのタイプは次のとおりです。

- 永久ライセンス：特定の機能を備え、有効期限のないライセンスを購入できます。
- 評価ライセンス ：スイッチに事前にインストールされています。使用有効期間は 90 日です。

永久ライセンスまたは評価ライセンスをアクティブ化するには、エンドユーザ ライセンス契約（EULA）を承認する必要があります。

永久ライセンスは1つのデバイスから別のデバイスに移動できます。ライセンスをアクティブ化するには、スイッチを再起動する必要があります。

評価ライセンスは、アクティブ化してから 90 日後に有効期限が切れます。評価ライセンスはスイッチのマニュファクチャリング イメージであり、別のスイッチに移動できません。このタイプのライセンスは、いったんアクティブ化すると、有効期限が切れるまで非アクティブ化できません。評価期間が満了すると、次のリロード時にスイッチのイメージのライセンスはデフォルトに戻るため、ネットワーク運用に影響はありません。

Right-To-Use イメージ ベースのライセンス

Right-To-Use イメージ ライセンスは、特定のイメージ ベースのライセンスに基づき、次の一連の機能をサポートします。

- LAN Base：レイヤ 2 の機能。
- IP Base：レイヤ 2 およびレイヤ 3 の機能。
- IP Services：レイヤ 2、レイヤ 3、IPv6 の機能（スイッチにのみ適用され、コントローラには適用されません）。

スイッチのデフォルトのイメージ ライセンスは次のとおりです。

- Catalyst 2960-CX スイッチ：LAN Base
- Catalyst 3560-CX スイッチ：IP Base

Right-To-Use ライセンスの状態

特定のライセンス タイプとレベルを設定した後は、ライセンスの状態をモニタすることでライセンスを管理できます。

表 *160 : RTU* ライセンスの状態

License State	説明
Active, In Use	EULA が承認され、デバイス再起動後にライセンスが使用されています。
Active, Not In Use	EULA が承認され、ライセンスが有効になった時点で、スイッチを使用する準備が整っています。
非アクティブ化	EULA が承認されませんでした。

イメージ ベースのライセンスの状態をモニタする場合のガイドラインは次のとおりです。

- 購入した永久ライセンスは、スイッチの再起動後のみに *Active, In Use* 状態に設定されます。
- 複数のライセンスを購入した場合は、再起動すると最も高い機能セットのライセンスがアクティブ化されます。たとえば、IP Services ライセンスがアクティブ化され、LAN Base ライセンスはアクティブ化されません。
- スイッチの再起動後も、購入済みの残りのライセンスは **Active, Not In Use** 状態のままです。

（注） AP-Count ライセンスの場合に状態を「Active, In Use」に変更するには、まず、評価 AP-Count ライセンスが非アクティブ化されているようにする必要があります。

モビリティ コントローラ モード

AP-Count ライセンスは、スイッチがモビリティ コントローラ モードになっている場合にのみ使用します。MC は、AP-CountAP-Count ライセンスをトラッキングするゲートキーパであり、アクセス ポイント参加を許可または拒否できます。

AP-Count ライセンスはを、CLI で設定可能なモビリティ コントローラ モードで実行して管理します。

Right-To-Use Adder AP-Count 再ホスト ライセンス

あるデバイスのライセンスを無効にして、別のデバイスにインストールする操作を再ホストと呼びます。デバイスの目的を変更するために、ライセンスのリホストが必要になる場合があります。

ライセンスを再ホストするには、あるデバイスの Adder AP-Count ライセンスを非アクティブ化し、別のデバイスで同じライセンスをアクティブ化します。

評価ライセンスを再ホストすることはできません。

RTU ライセンスの設定方法

イメージ ベース ライセンスのアクティブ化

イメージベースのライセンスをアクティブ化するには、次のタスクを実行します。

手順の概要

1. **license right-to-use activate** {**ipbase** | **ipservices** | **lanbase**} [**all** | **evaluation** | **slot***slot-number*] [**acceptEULA**]
2. **reload** [*LINE* | **at** | **cancel** | **in** | **slot** *stack-member-number* | **standby-cpu**]
3. **show license right-to-use usage** [**slot** *slot-number*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**license right-to-use activate** {**ipbase** | **ipservices** | **lanbase**} [**all** | **evaluation** | **slot***slot-number*] [**acceptEULA**] 例： `スイッチ# license right-to-use activate ipservices all acceptEULA`	ライセンス レベルをアクティブにします。すべてのスイッチ上でアクティブ化され、EULA への同意が含まれることもあります。 （注） EULA に同意しない場合は、変更した設定はリロード後に反映されません。デフォルトのライセンス（または非アクティブ化されたライセンス）がリロード後にアクティブになります。
ステップ 2	**reload** [*LINE* | **at** | **cancel** | **in** | **slot** *stack-member-number* | **standby-cpu**] 例： `スイッチ# reload slot 1` `Proceed with reload? [confirm] y`	特定のスタック メンバをリロードし、RTU 追加 AP-Count ライセンスのアクティブ化プロセスを完了します。 （注） これまでに同意していなかった場合は、リロード後に同意を促すメッセージが表示されます。

	コマンドまたはアクション	目的
		ライセンス レベルを変更する場合は、設定を保存する必要はありません。ただし、リロードする前にすべての設定が適切に保存されていることを確認することをお勧めします。再起動時に高いライセンス レベルから低いライセンス レベルに変更すると、適用できない CLI は削除されます。アクティブに使用される低いライセンス レベルの機能はすべて削除されないようにしてください。
ステップ 3	**show license right-to-use usage** [**slot** *slot-number*] 例： スイッチ# **show license right-to-use usage** Slot# License Name Type usage-duration(y:m:d) In-Use EULA 1 ipservices Permanent 0 :10:27 yes yes 1 ipservices Evaluation 0 :0 :0 no no 1 ipbase Permanent 0 :0 :9 no yes 1 ipbase Evaluation 0 :0 :0 no no 1 lanbase Permanent 0 :11:12 no yes Switch#	詳細な使用状況に関する情報を表示します。

ap-count ライセンスのアクティブ化

手順の概要

1. **license right-to-use activate** {**apcount** *ap-number* **slot** *slot-num*} | **evaluation**} [**acceptEULA**]
2. **show license right-to-use usage** [**slot** *slot-number*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**license right-to-use activate** {**apcount** *ap-number* **slot** *slot-num*}	**evaluation**} [**acceptEULA**] 例： スイッチ# **license right to use activate apcount 5 slot 1 acceptEULA**	1 つ以上の追加 AP-Count ライセンスをアクティブ化し、ただちに EULA に同意します。

	コマンドまたはアクション	目的
ステップ 2	**show license right-to-use usage** [**slot** *slot-number*] 例： スイッチ# **show license right-to-use usage** Slot# License Name Type usage-duration(y:m:d) In-Use EULA 1 ipservices permanent 0 :3 :29 yes yes 1 ipservices evaluation 0 :0 :0 no no 1 ipbase permanent 0 :0 :0 no no 1 ipbase evaluation 0 :0 :0 no no 1 lanbase permanent 0 :0 :0 no no 1 apcount evaluation 0 :3 :11 no no 1 apcount base 0 :0 :0 no yes 1 apcount adder 0 :0 :17 yes yes Switch#	詳細な使用状況に関する情報を表示します。

アップグレード ライセンスまたはキャパシティ Adder ライセンスの取得

キャパシティ Adder ライセンスを使用すれば、deviceがサポートするアクセスポイントの数が増やせます。

手順の概要

1. **license right-to-use** {**activate** | **deactivate**} **apcount** {*ap-number* | **evaluation** } **slot** *slot-num* [**acceptEULA**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**license right-to-use** {**activate**	**deactivate**} **apcount** {*ap-number*	**evaluation** } **slot** *slot-num* [**acceptEULA**] 例： スイッチ# **license right to use activate apcount 5 slot 2 acceptEULA**	1 つ以上の追加 AP-Count ライセンスをアクティブ化し、ただちに EULA に同意します。

ライセンスの再ホスト

ライセンスを再ホストするには、1つのdeviceのライセンスを非アクティブ化し、別のdeviceで同じライセンスをアクティブ化します。

手順の概要

1. **license right-to-use deactivate [license-level] apcount** *ap-number* **slot** *slot-num*
2. **license right-to-use activate [license-level] slot** *slot-num* [**acceptEULA**]

手順の詳細

	コマンドまたはアクション	目的
ステップ1	**license right-to-use deactivate [license-level] apcount** *ap-number* **slot** *slot-num* 例： スイッチ# **license right-to-use deactivate apcount 1 slot 1**	1つのdeviceのライセンスを非アクティブ化します。この例では、IP Base ライセンスレベルがスロット1から非アクティブ化されています。
ステップ2	**license right-to-use activate [license-level] slot** *slot-num* [**acceptEULA**] 例： スイッチ# **license right to use activate ipbase slot 2 acceptEULA**	別のdeviceをアクティブ化します。この例では、IP Base ライセンスレベルがスロット2に再ホストされています。

RTU ライセンスのモニタリングおよびメンテナンス

コマンド	目的
show license right-to-use default	デフォルトのライセンス情報を表示します。
show license right-to-use detail	スイッチ スタック内のすべてのライセンスの詳細情報を表示します。
show license right-to-use eula {evaluation | permanent}	エンドユーザ ライセンス契約を表示します。
show license right-to-use mismatch	一致しないライセンス情報を表示します。
show license right-to-use slot *slot-number*	スイッチ スタック内の特定のスロットのライセンス情報を表示します。
show license right-to-use summary	スイッチ スタック全体のライセンス情報の要約を表示します。

コマンド	目的
show license right-to-use usage [**slot** *slot-number*]	スイッチ スタック内のすべてのライセンスの使用状況に関する詳細情報を表示します。
show switch	ライセンスのステータスを含むスイッチスタック内のすべてのメンバの詳細情報を表示します。

RTU ライセンスの設定例

例：RTU イメージ ベースのライセンスのアクティブ化

次に、IP Servicesイメージライセンスをアクティブ化し、特定のスロットのEULAを受け入れる例を示します。

```
Switch# license right-to-use activate ipservices slot 1 acceptEULA
% switch-1:stack-mgr:Reboot the switch to invoke the highest activated License level
```

次に、評価用ライセンスをアクティブ化する例を示します。

```
Switch# license right-to-use activate ipservices evaluation acceptEULA
% switch-1:stack-mgr:Reboot the switch to invoke the highest activated License level
```

例：RTU ライセンス情報の表示

例：RTU ライセンスの詳細の表示

次に、スロット 1 の RTU ライセンスのすべての詳細情報の例を示します。

例：RTU ライセンスの不一致の表示

この例では、スタック内のスイッチのライセンス情報と、メンバスイッチの不一致ステータスを示します。メンバ スイッチがアクティブ スイッチと一致している必要があります。

```
Switch# show switch

Switch/Stack Mac Address : 1c1d.8625.7700 - Local Mac Address
                                                 H/W   Current
Switch#   Role        Mac Address     Priority Version  State
-----------------------------------------------------------------------------
*1       Active   1c1d.8625.7700     15     V02     Ready
```

```
2       Standby  bc16.f55c.ab80     7      V04     Ready
3       Member   580a.2095.da00     1      V03     Lic-Mismatch
```

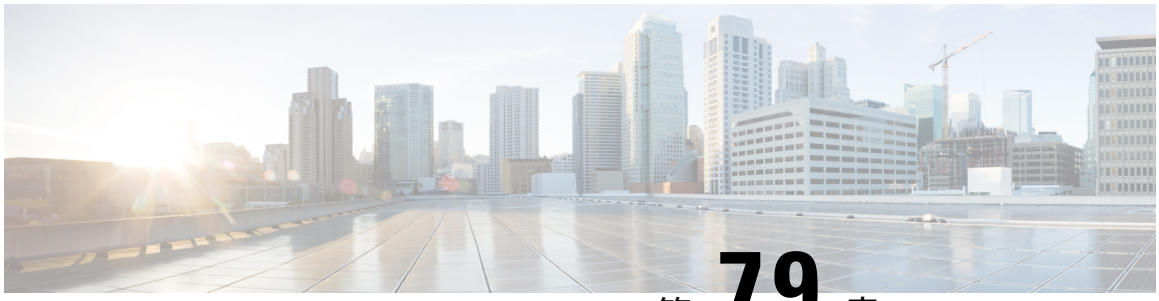
（注） ライセンスの不一致を解決するには、まず、RTU ライセンスのサマリーを確認します。

```
Switch# show license right-to-use
```

次に、アクティブ スイッチと同じライセンス レベルとなるように、一致していないスイッチのライセンス レベルを変更します。この例では、アクティブ スイッチと一致するように IP Base ライセンスをメンバ スイッチに対してアクティブ化したことを示します。

```
Switch# license right-to-use activate ipbase slot 3 acceptEULA
```

例：RTU ライセンス使用状況の表示



第 79 章

スイッチのクラスタリング

スイッチ クラスタの概要

スイッチクラスタは、最大16個の接続されたクラスタ対応Catalystスイッチで、単一エンティティとして管理されます。クラスタ内のスイッチは、スイッチクラスタ化テクノロジーによって、単一の IP アドレスから異なる Catalyst デスクトップ スイッチ プラットフォームで構成されたグループを設定したり、トラブルシューティングを行ったりできます。

スイッチクラスタでは、1 台のスイッチがクラスタコマンドスイッチとして動作する必要があり、最大15台の他のスイッチがクラスタメンバースイッチとして動作できます。1つのクラスタ内のスイッチの総数は、16 台のスイッチを超えることはできません。クラスタコマンドスイッチは、クラスタメンバースイッチの設定、管理、およびモニターに使用する、一元化されたアクセスポイントです。クラスタ メンバは、一度に1つのクラスタにしか所属できません。

（注） スイッチクラスタはスイッチスタックとは異なります。スイッチスタックとは、スタックポート経由で接続された Catalyst 3750-X、Catalyst 3750-E、または Catalyst 3750 スイッチのセットです。

スイッチのクラスタ化には次のような利点があります。

- 相互接続メディアや物理的な場所に左右されずCatalystスイッチの管理ができます。スイッチは同じ場所に設置することも、レイヤ 2 またはレイヤ 3 ネットワークを介して設置することもできます（Catalyst 3560、Catalyst 3750、Catalyst 3560-E、Catalyst 3750-E、Catalyst 3560-X、または Catalyst 3750-X スイッチを、クラスタのレイヤ 2 スイッチの間に設置するレイヤ 3 のルータとして使用している場合）。
- クラスタコマンドスイッチに冗長性を持たせることで、コマンドスイッチに障害が発生した場合でも対応できます。1 つまたは複数のスイッチをスタンバイ クラスタ コマンド ス

イッチに指定すると、クラスタメンバー間の競合を回避できます。クラスタスタンバイグループは、スタンバイ クラスタ コマンド スイッチのグループです。

- さまざまな Catalyst スイッチを 1 つの IP アドレスで管理できます。これは、特に IP アドレスの数が限られている場合に効果があります。スイッチクラスタとの通信はすべてクラスタコマンドスイッチの IP アドレスで行われます。

下の表に、スイッチのクラスタ化に対応している Catalyst スイッチを示します。クラスタコマンドスイッチになれるスイッチおよびクラスタメンバースイッチにしかなれないスイッチ、さらに、それらに必要なソフトウェアバージョンも示します。

表 *161 :* スイッチ ソフトウェアおよびクラスタへの対応性

Switch	Cisco IOS Release	Cluster Capability
Catalyst 3750-X	12.2(53)SE2 以降	メンバまたはコマンド スイッチ
Catalyst 3750-E	12.2(35)SE2 以降	メンバまたはコマンド スイッチ
Catalyst 3750	12.1(11)AX 以降	メンバまたはコマンド スイッチ
Catalyst 3560-X	12.2(53)SE1 以降	メンバまたはコマンド スイッチ
Catalyst 3560-E	12.2(35)SE2 以降	メンバまたはコマンド スイッチ
Catalyst 3560	12.1(19)EA1b 以降	メンバまたはコマンド スイッチ
Catalyst 3550	12.1(4)EA1 以降	メンバまたはコマンド スイッチ
Catalyst 2970	12.1(11)AX 以降	メンバまたはコマンド スイッチ
Catalyst 2960	12.2(25)FX 以降	メンバまたはコマンド スイッチ
Catalyst 2955	12.1(12c)EA1 以降	メンバまたはコマンド スイッチ
Catalyst 2950	12.0(5.2)WC(1) 以降	メンバまたはコマンド スイッチ
Catalyst 2950 LRE	12.1(11)JY 以降	メンバまたはコマンド スイッチ
Catalyst 2940	12.1(13)AY 以降	メンバまたはコマンド スイッチ
Catalyst 3500 XL	12.0(5.1)XU 以降	メンバまたはコマンド スイッチ
Catalyst 2900 XL（8 MB スイッチ）	12.0(5.1)XU 以降	メンバまたはコマンド スイッチ
Catalyst 2900 XL（4 MB スイッチ）	11.2(8.5)SA6（推奨）	メンバ スイッチのみ
Catalyst 1900 および Catalyst 2820	9.00（-A または -EN）以降	メンバ スイッチのみ

クラスタ コマンド スイッチの特性

クラスタコマンドスイッチは、次の要件を満たしている必要があります。

- サポート対象のソフトウェア リリースを実行している。
- IP アドレスが指定されている。
- Cisco Discovery Protocol（CDP）バージョン 2 がイネーブル（デフォルト）に設定されている。
- 別のクラスタのコマンドまたはクラスタメンバースイッチではない。
- 管理 VLAN を介してスタンバイ クラスタ コマンド スイッチに、および共通 VLAN を介してクラスタメンバースイッチに接続されている。

スタンバイ クラスタ コマンド スイッチの特性

スタンバイ クラスタ コマンド スイッチは、次の要件を満たしている必要があります。

- サポート対象のソフトウェア リリースを実行している。
- IP アドレスが指定されている。
- CDP バージョン 2 がイネーブルに設定されている。
- 管理 VLAN を介してコマンドスイッチに接続されていて、なおかつ他のスタンバイコマンドスイッチに接続されている。
- 共通 VLAN を介して（クラスタコマンドおよびスタンバイコマンドスイッチを除く）他のすべてのクラスタメンバースイッチに接続されている。
- クラスタメンバースイッチとの接続能力を維持するために、クラスタに冗長接続されている。
- 別のクラスタのコマンドまたはメンバースイッチではない。

（注） スタンバイ クラスタ コマンド スイッチは、クラスタコマンドスイッチと同タイプのスイッチでなければなりません。たとえば、クラスタコマンドスイッチが Catalyst 3750-E スイッチの場合、スタンバイ クラスタ コマンド スイッチも Catalyst 3750-E スイッチにする必要があります。スタンバイ クラスタ コマンド スイッチの要件については、他のクラスタ対応スイッチのスイッチのコンフィギュレーション ガイドを参照してください。

候補スイッチおよびクラスタ メンバ スイッチの特性

候補スイッチとは、クラスタにまだ追加されていないクラスタ対応スイッチおよびスイッチスタックです。クラスタメンバースイッチは、スイッチクラスタにすでに追加されているスイッチおよびスイッチスタックです。候補またはクラスタメンバースイッチには独自の IP アドレスおよびパスワードがありますが、必須ではありません。

クラスタに加入するには、候補スイッチが次の要件を満たしている必要があります。

- クラスタ対応のソフトウェアが稼働している。

- CDP バージョン 2 がイネーブルに設定されている。
- 別のクラスタのコマンドまたはクラスタメンバースイッチではない。
- クラスタスタンバイグループが存在する場合、少なくとも 1 つの共通 VLAN を介して、すべてのスタンバイ クラスタ コマンド スイッチに接続されている。各スタンバイ クラスタ コマンド スイッチに対応する VLAN は、異なる場合があります。
- スイッチで **ip http** サーバー グローバル コンフィギュレーション コマンドを設定する必要がある。
- 少なくとも 1 つの共通 VLAN を介して、クラスタコマンドスイッチに接続されている。

（注） Catalyst 1900、Catalyst 2820、Catalyst 2900 XL、Catalyst 2940、Catalyst 2950、および Catalyst 3500 XL 候補およびクラスタメンバースイッチは、管理 VLAN を介してクラスタコマンドスイッチおよびスタンバイ クラスタ コマンド スイッチに接続する必要があります。スイッチクラスタ環境におけるこれらのスイッチの詳細情報は、該当するスイッチのソフトウェア コンフィギュレーション ガイドを参照してください。Catalyst 2960、Catalyst 2970、Catalyst 3550、Catalyst 3560、Catalyst 3560-E、Catalyst 3750、Catalyst 3750-E、Catalyst 3650-X、または Catalyst 3750-X クラスタ コマンド スイッチを使用する場合、この要件は当てはまりません。候補およびクラスタメンバースイッチは、クラスタコマンドスイッチと共通の任意の VLAN を介して接続できます。

スイッチ クラスタのプランニング

複数のスイッチをクラスタで管理する場合、予想される競合や互換性の問題解決に重点を置きます。ここでは、クラスタを作成する前に理解すべき注意事項、要件、および警告について説明します。

クラスタに対応している Catalyst スイッチについては、各スイッチのリリース ノートを参照してください。リリース ノートでは、クラスタ コマンド スイッチになれるスイッチとクラスタ メンバー スイッチにしかなれないスイッチ、また、それらに必要なソフトウェア バージョンやブラウザだけでなく、Java プラグインの設定も参照できます。

クラスタ候補およびメンバの自動検出

クラスタ コマンド スイッチは Cisco Discovery Protocol（CDP）を使用して、複数の VLAN の中からクラスタ メンバ スイッチ、候補スイッチ、ネイバー スイッチクラスタ、エッジデバイスを検出します。また、スター型のトポロジやカスケード型のトポロジ内からも検出できます。

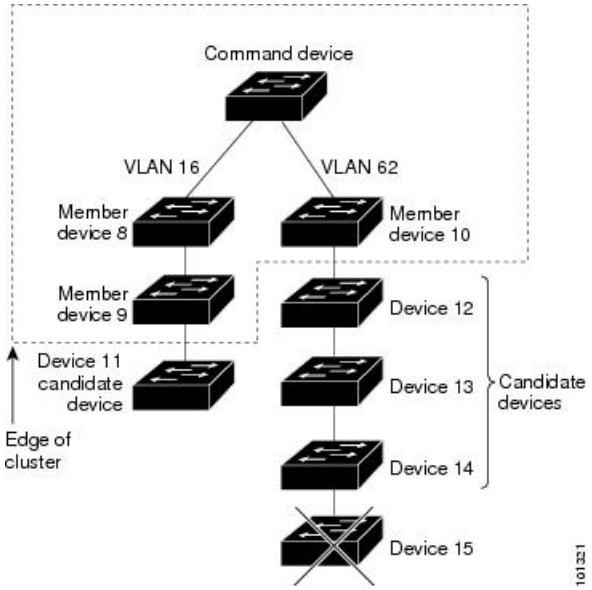
（注） クラスタ コマンド スイッチを使用してクラスタに対応したスイッチを検出する場合、クラスタ コマンド スイッチ、クラスタ メンバ、またはクラスタ対応スイッチの CDP を無効にしないでください。

CDP ホップによる検出

クラスタ コマンド スイッチは CDP を使用して、クラスタ エッジから最大 7 CDP ホップ（デフォルトは 3 ホップ）までスイッチを検出できます。クラスタ エッジは、クラスタや候補スイッチに接続している最後のクラスタ スイッチの部分を指します。たとえば、図のクラスタ メンバ スイッチ 9 と 10 はクラスタのエッジにあります。

この図では、クラスタ コマンド スイッチには VLAN 16 と 62 に割り当てられたポートがあります。CDP ホップのカウントは 3 です。クラスタ エッジから 3 ホップ以内にあるので、クラスタ コマンド スイッチはスイッチ 11、12、13、14 を検出します。スイッチ 15 はクラスタ エッジから 4 ホップ先なので検出されません。

図 *123 : CDP* ホップによる検出

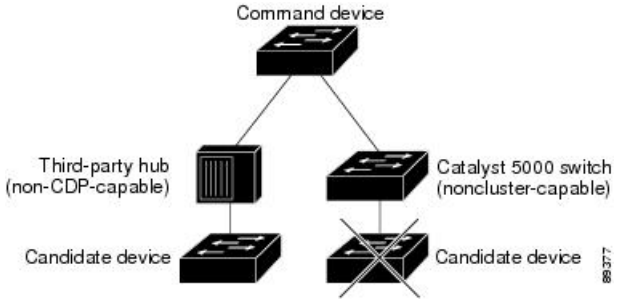


CDP 非対応デバイスおよびクラスタ非対応デバイスからの検出

クラスタ コマンド スイッチを *CDP* 非対応のサードパーティ製のハブ（他社製のハブなど）に接続している場合、そのサードパーティ製のハブを介して接続しているクラスタ対応デバイスを検出できます。ただし、クラスタ コマンド スイッチをクラスタ非対応のシスコ デバイスに接続している場合、クラスタ非対応のシスコデバイスより先にあるクラスタ対応のデバイスは検出できません。

下の図に、サードパーティ製のハブに接続したスイッチを検出するクラスタ コマンド スイッチを示します。ただし、クラスタ コマンド スイッチは Catalyst 5000 スイッチに接続しているスイッチは検出しません。

図 *124 : CDP* 非対応デバイスおよびクラスタ非対応デバイスからの検出



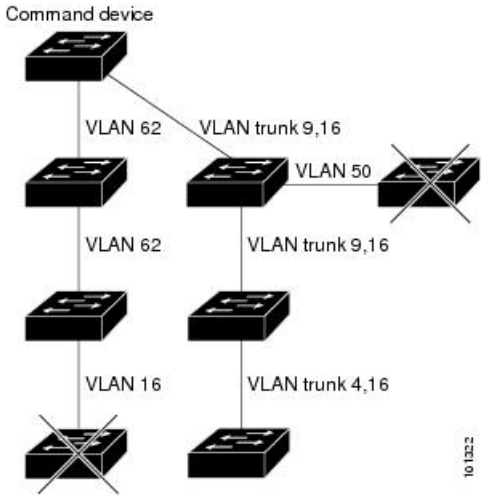
異なる VLAN からの検出

クラスタ コマンドスイッチが、Catalyst 3560-E、Catalyst 3750-E、Catalyst 3560-X、または Catalyst 3750-X スイッチの場合、クラスタは、異なる VLAN にあるスイッチをクラスタ メンバにすることができます。クラスタ メンバ スイッチとして、Catalyst スイッチもクラスタ コマンド スイッチと共通の VLAN に少なくとも 1 つは接続している必要があります。図のクラスタ コマンド スイッチのポートは VLAN 9、16、62 に割り当てられているため、これらの VLAN のスイッチが検出されます。VLAN 50 にあるスイッチは検出されません。また、最初の列の VLAN 16 にあるスイッチも、クラスタ コマンド スイッチに VLAN が接続されていないため検出されません。

Catalyst 2900 XL、Catalyst 2950、および Catalyst 3500 XL のクラスタ メンバ スイッチは、それぞれの管理 VLAN を介してクラスタ コマンド スイッチに接続している必要があります。

（注） スイッチスタックにある VLAN のその他の考慮事項については、「スイッチクラスタおよびスイッチスタック」セクションを参照してください。

図 *125 :* 異なる *VLAN* からの検出



異なる管理 VLAN からの検出

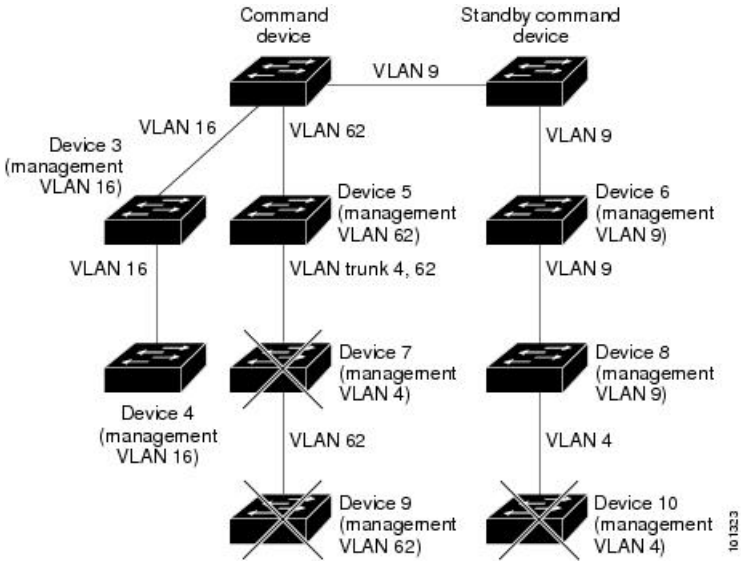
Catalyst 2960、Catalyst 2970、Catalyst 3550、Catalyst 3560、Catalyst 3560-E、Catalyst 3750、Catalyst 3750-E、Catalyst 3560-X、または Catalyst 3750-X クラスタ コマンド スイッチは、異なる VLAN や管理 VLAN のクラスタ メンバ スイッチを検出して管理できます。クラスタ メンバ スイッチとして、Catalyst スイッチもクラスタ コマンド スイッチと共通の VLAN に少なくとも 1 つは接続している必要があります。ただし、管理 VLAN を介してクラスタ コマンド スイッチに接続する必要はありません。デフォルトの管理 VLAN は VLAN 1 です。

（注） スイッチ クラスタに Catalyst 3750-E スイッチ、Catalyst 3750-X スイッチまたはスイッチ スタックがある場合、スイッチまたはスイッチ スタックをクラスタ コマンド スイッチにする必要があります。

図に示されているクラスタ コマンド スイッチおよびスタンバイ コマンド スイッチ（Catalyst 2960、Catalyst 2970、Catalyst 3550、Catalyst 3560、Catalyst 3560-E、Catalyst 3750、Catalyst 3750-E、Catalyst 3560-X、または Catalyst 3750-X と想定します）のポートには、VLAN 9、16、および 62 が割り当てられています。クラスタ コマンド スイッチの管理 VLAN は VLAN 9 です。各クラスタ コマンド スイッチは、次の例外を除き、異なる管理 VLAN のスイッチを検出します。

- スイッチ 7 およびスイッチ 10（管理 VLAN 4 のスイッチ）。クラスタ コマンド スイッチと共通の VLAN（VLAN 62 および VLAN 9）に接続していないため検出されません。
- スイッチ 9。自動検出は非候補デバイス（スイッチ 7）より先は検出できないため、検出されません。

図 *126 :* レイヤ *3* クラスタ コマンド スイッチを使用した異なる管理 *VLAN* からの検出

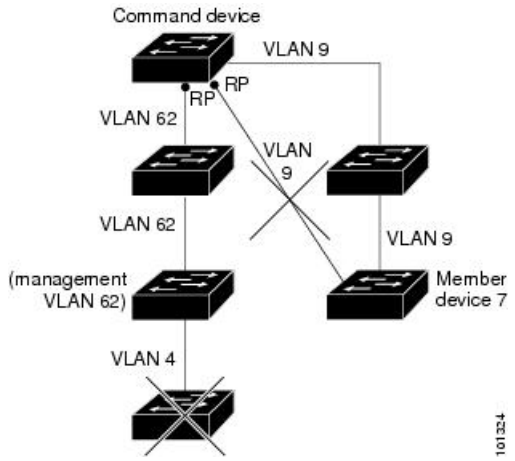


ルーテッド ポートからの検出

ルーテッド ポート（RP）が設定されているクラスタ コマンド スイッチは、RP と同じ VLAN 内の候補スイッチおよびクラスタ メンバ スイッチだけを検出します。

図のレイヤ 3 クラスタ コマンド スイッチにより、VLAN 9 および 62 のスイッチは検出されますが、VLAN 4 のスイッチは検出されません。クラスタ コマンド スイッチとクラスタ メンバ スイッチ 7 間の RP パスが損失している場合、VLAN 9 を介する冗長パスがあるため、クラスタ メンバ スイッチ 7 との接続は維持されます。

図 *127 :* ルーテッド ポートからの検出



新しく設置したスイッチの検出

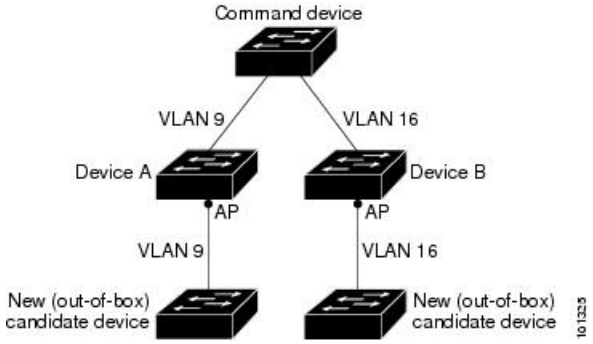
新しいアウトオブボックス スイッチをクラスタに加入させるには、アクセスポートの 1 つを介してクラスタに接続する必要があります。アクセス ポート（AP）は 1 つの VLAN にのみ属し、そのトラフィックを転送します。デフォルトでは、新しいスイッチとそのアクセス ポートが VLAN 1 に割り当てられます。

新しいスイッチがクラスタに加入すると、デフォルトの VLAN は即座にアップストリーム ネイバーの VLAN に変わります。また、新しいスイッチも自身のアクセス ポートを変更して、そのネイバーの VLAN に加わります。

図のクラスタ コマンド スイッチは、VLAN 9 および 16 に加入しています。新しいクラスタ対応のスイッチがクラスタに加入すると、次の処理が行われます。

- 1 つのクラスタ対応のスイッチとそのアクセス ポートが VLAN 9 に割り当てられます。
- 他のクラスタ対応のスイッチとそのアクセス ポートが管理 VLAN 16 に割り当てられます。

図 *128 :* 新しく設置したスイッチの検出



HSRP およびスタンバイ クラスタ コマンド スイッチ

スイッチは Hot Standby Router Protocol（HSRP）をサポートしているため、スタンバイ クラスタ コマンド スイッチのグループを設定できます。クラスタ コマンド スイッチは、すべての通信の転送と、すべてのクラスタ メンバ スイッチの設定情報を管理しているため、次のような環境設定を推奨します。

- クラスタ コマンドのスイッチ スタックには、スイッチ スタック全体に障害が発生する場合に備えて、スタンバイ クラスタ コマンド スイッチが必要です。ただし、コマンド スイッチのスタック マスターだけに障害が発生した場合は、スイッチ スタックで新しいスタック マスターを選出し、クラスタ コマンド スイッチ スタックとしての機能を引き継がせることができます。
- スタンドアロンのクラスタ コマンド スイッチの場合、プライマリ クラスタ コマンド スイッチの障害に備え、スタンバイ クラスタ コマンド スイッチを設定してその機能を引き継がせるようにします。

クラスタ スタンバイ グループは、「スタンバイ クラスタ コマンド スイッチの特性」の項で説明している要件を満たしたコマンド対応スイッチのグループです。クラスタごとに、1 つのクラスタ スタンバイ グループのみ割り当てることができます。

（注） クラスタ スタンバイ グループは HSRP グループです。HSRP をディセーブルにすると、クラスタ スタンバイ グループがディセーブルになります。

クラスタ スタンバイ グループのスイッチは、HSRP プライオリティに基づいてランク付けされています。グループ内でプライオリティが最も高いスイッチは、アクティブ クラスタ コマンド スイッチ（AC）です。グループ内で次にプライオリティの高いスイッチは、スタンバイ クラスタ コマンド スイッチ（SC）です。クラスタ スタンバイ グループの他のスイッチは、パッシブ クラスタ コマンド スイッチ（PC）です。アクティブ クラスタ コマンド スイッチおよびスタンバイ クラスタ コマンド スイッチが同時にディセーブルになった場合、パッシブ クラスタ コマンド スイッチの中でプライオリティが一番高いものがアクティブ クラスタ コマンド スイッチになります。クラスタスタンバイグループのメンバーおよびルータ冗長グループのメン

バーのプライオリティの変更には、同じ HSRP **standby priority** インターフェイス コンフィギュレーション コマンドを使用します。

（注） HSRP のスタンバイ中止間隔は、hello タイム間隔の 3 倍以上必要です。デフォルトの HSRP スタンバイ中止間隔は 10 秒です。デフォルトの HSRP スタンバイ hello タイム インターバルは 3 秒です。

仮想 IP アドレス

クラスタ スタンバイ グループには、一意の仮想 IP アドレス、グループ番号、グループ名を割り当てる必要があります。この情報は、特定の VLAN またはアクティブ クラスタ コマンド スイッチのルーテッド ポートで設定します。アクティブ クラスタ コマンド スイッチは、仮想 IP アドレス宛てのトラフィックを受信します。クラスタを管理するには、コマンドスイッチの IP アドレスからではなく、仮想 IP アドレスからアクティブ クラスタ コマンド スイッチにアクセスする必要があります。（アクティブ クラスタ コマンド スイッチの IP アドレスがクラスタ スタンバイ グループの仮想 IP アドレスと異なる場合）。

アクティブ クラスタ コマンド スイッチに障害が発生すると、スタンバイ クラスタ コマンド スイッチが仮想 IP アドレスを使用して、アクティブ クラスタ コマンド スイッチになります。クラスタ スタンバイ グループのパッシブ スイッチは、それぞれ割り当てられたプライオリティを比較し、新しいスタンバイ クラスタ コマンド スイッチを選出します。その後、プライオリティの一番高いパッシブ スタンバイ スイッチがスタンバイ クラスタ コマンド スイッチになります。前回アクティブ クラスタ コマンド スイッチだったスイッチが再びアクティブになると、アクティブ クラスタ コマンド スイッチの役割を再開します。そのため、現在アクティブ クラスタ コマンド スイッチを担当しているスイッチは再びスタンバイ クラスタ コマンド スイッチになります。スイッチ クラスタの IP アドレスの詳細については、「IP アドレス」の項を参照してください。

クラスタ スタンバイ グループに関する他の考慮事項

次の要件も満たす必要があります。

- スタンバイ クラスタ コマンド スイッチは、クラスタ コマンド スイッチと同タイプのスイッチでなければなりません。たとえば、クラスタ コマンド スイッチが Catalyst 3750-E または Catalyst 3750-X スイッチの場合、スタンバイ クラスタ コマンド スイッチも Catalyst 3750-E か Catalyst 3750-X スイッチにする必要があります。スタンバイ クラスタ コマンド スイッチの要件については、他のクラスタ対応スイッチのコンフィギュレーション ガイドを参照してください。

  スイッチ クラスタに Catalyst 3750-X スイッチまたはスイッチ スタックが含まれている場合、それをクラスタ コマンド スイッチにする必要があります。含まれていない場合、クラスタに Catalyst 3750-E スイッチまたはスイッチ スタックがあれば、そのスイッチをクラスタ コマンド スイッチにします。

- クラスタごとに、1 つのクラスタ スタンバイ グループのみ割り当てることができます。ルータ冗長スタンバイ グループは複数作成できます。

1 つの HSRP グループをクラスタ スタンバイ グループとルータ冗長構成グループの両方にすることができます。ただし、ルータ冗長構成グループがクラスタ スタンバイ グループになった場合、そのグループ上でのルータ冗長構成はディセーブルになります。CLI を使用すれば、冗長構成を再びイネーブルにすることができます。

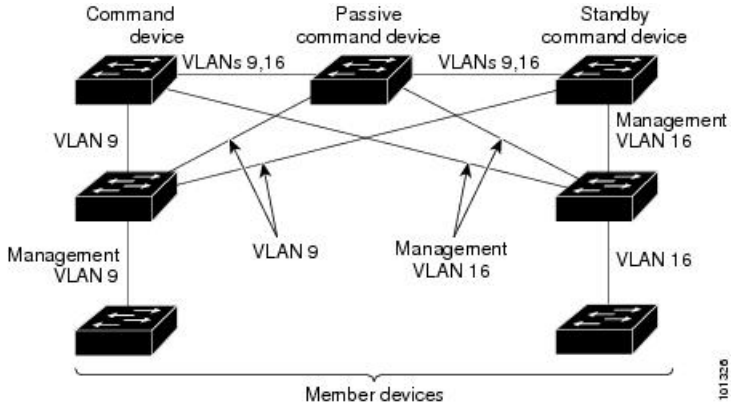
- すべてのスタンバイグループ メンバはそのクラスタのメンバである必要があります。

（注） スタンバイ クラスタ コマンド スイッチとして割り当てることができるスイッチ数に制限はありません。ただし、クラスタのスイッチの総数（アクティブ クラスタ コマンド スイッチ、スタンバイ グループ メンバ、およびクラスタ メンバ スイッチを含む）は 16 以内にする必要があります。

- 各スタンバイグループのメンバ（下の図を参照）は、同じ VLAN を介してクラスタ コマンド スイッチに接続されている必要があります。この例では、クラスタ コマンド スイッチとスタンバイ クラスタ コマンド スイッチが Catalyst 3560-E、Catalyst 3750-E、Catalyst 3560-X、または Catalyst 3750-X クラスタ コマンド スイッチです。各スタンバイグループのメンバも、スイッチ クラスタと同じ VLAN を最低 1 つは介在させて、冗長性を持たせながら相互接続する必要があります。

  Catalyst 1900、Catalyst 2820、Catalyst 2900 XL、Catalyst 2950、Catalyst 3500 XL クラスタ メンバ スイッチは、それぞれの管理 VLAN を介してクラスタ スタンバイ グループに接続する必要があります。

図 *129*：スタンバイグループ メンバとクラスタ メンバ間の *VLAN* 接続



クラスタ設定の自動回復

アクティブ クラスタ コマンド スイッチは、クラスタ設定情報をスタンバイ クラスタ コマンド スイッチに継続的に送信します（デバイス設定情報は送信しません）。アクティブ クラスタ コマンド スイッチに障害が発生した場合は、この情報をもとに、スタンバイ クラスタ コマンド スイッチが即座にクラスタを引き継ぎます。

自動検出には次のような制限があります。

- この制限は、Catalyst 2950、Catalyst 2960、Catalyst 2970、Catalyst 3550、Catalyst 3560、Catalyst 3560-E、Catalyst 3560-X、Catalyst 3750、Catalyst 3750-E、および Catalyst 3750-X コマンド スイッチおよびスタンバイ クラスタ コマンド スイッチを備えたクラスタだけに該当します。アクティブ クラスタ コマンド スイッチおよびスタンバイ クラスタ コマンド スイッチが同時にディセーブルになった場合、パッシブ クラスタ コマンド スイッチの中でプライオリティが一番高いものがアクティブ クラスタ コマンド スイッチになります。ただし、パッシブ スタンバイ クラスタ コマンド スイッチだったため、以前のクラスタ コマンド スイッチはクラスタ設定情報を送信していません。アクティブ クラスタ コマンド スイッチは、スタンバイ クラスタ コマンド スイッチにクラスタ設定情報のみ送信します。そのため、クラスタを再設定する必要があります。
- クラスタ スタンバイ グループに複数のスイッチを持つアクティブ クラスタ コマンド スイッチに障害が発生した場合、新しいクラスタ コマンド スイッチは、いかなる Catalyst 1900、Catalyst 2820、および Catalyst 2916M XL のクラスタ メンバ スイッチも検出しません。これらのクラスタ メンバ スイッチをクラスタにもう一度追加する必要があります。
- アクティブ クラスタ コマンド スイッチに障害が発生してダウンした後、再びアクティブになった場合、そのスイッチはいかなる Catalyst 1900、Catalyst 2820、および Catalyst 2916M XL クラスタ メンバ スイッチも検出しません。これらのクラスタ メンバ スイッチをクラスタにもう一度追加する必要があります。

以前アクティブ クラスタ コマンド スイッチだったスイッチが再びアクティブになった場合、そのスイッチは最新のクラスタ設定のコピー（ダウン中に追加されたメンバを含む）をアクティブ クラスタ コマンド スイッチから受信します。アクティブ クラスタ コマンド スイッチは、クラスタ スタンバイ グループにクラスタ設定のコピーを送信します。

IP Addresses

IP 情報をクラスタ コマンド スイッチに割り当てる必要があります。クラスタ コマンド スイッチには複数の IP アドレスを割り当てることができます。クラスタには、これらのコマンドスイッチの IP アドレスを介してアクセスできます。クラスタ スタンバイ グループを設定する場合、アクティブ クラスタ コマンド スイッチからスタンバイグループの仮想 IP アドレスを使用して、クラスタを管理する必要があります。仮想 IP アドレスを使用すると、アクティブ クラスタ コマンド スイッチに障害が発生してスタンバイ クラスタ コマンド スイッチがアクティブ クラスタ コマンド スイッチになった場合でも、クラスタへの接続を確保できます。

アクティブ クラスタ コマンド スイッチに障害が発生してスタンバイ クラスタ コマンド スイッチがその役割を引き継いだ場合、クラスタのアクセスには、スタンバイグループの仮想 IP アドレスも、新しいアクティブ クラスタ コマンド スイッチで使える IP アドレスも使用できます。

必須ではありませんが、IP アドレスはクラスタ対応のスイッチにも割り当てることができます。クラスタ メンバ スイッチは、コマンドスイッチの IP アドレスを使用して管理され、他のクラスタ メンバ スイッチと通信します。IP アドレスが割り当てられていないクラスタ メンバ スイッチがそのクラスタを離れる場合、スタンドアロン スイッチとして管理する IP アドレスを割り当てる必要があります。

ホスト名

クラスタ コマンド スイッチと対象のクラスタ メンバにはホスト名を割り当てる必要はありません。ただし、クラスタ コマンド スイッチに割り当てられたホスト名は、スイッチ クラスタを識別するのに役立ちます。スイッチのデフォルトのホスト名は *Switch* です。

クラスタに加入するスイッチにホスト名がない場合、クラスタ コマンド スイッチは一意のメンバ番号を自身のホスト名に追加し、そのスイッチに割り当てます。この処理はクラスタに加入するスイッチごとに順番に行われます。ここでいう番号とは、スイッチがクラスタに追加された順番を指します。たとえば、*eng-cluster* という名前のクラスタ コマンド スイッチでは、*5* 番目のクラスタ メンバとして *eng-cluster-5* という名前が割り当てられます。

スイッチにホスト名がある場合、クラスタへの加入時もクラスタからの脱退時もその名前が使用されます。

クラスタ脱退時、または新しいクラスタへの加入時にそのメンバ番号（*5* など）を確保するため、クラスタ コマンド スイッチからスイッチにホスト名を送信した場合、それを受信したスイッチは、新しいクラスタのクラスタ コマンド スイッチのホスト名（*mkg-cluster-5* など）で古いホスト名（*eng-cluster-5* など）を上書きします。新しいクラスタではスイッチのメンバ番号を変更する場合（*3* など）、スイッチは前回の名前（*eng-cluster-5*）を確保します。

パスワード

クラスタのメンバになるスイッチにはパスワードを割り当てる必要はありません。スイッチはコマンドスイッチのパスワードを継承してクラスタに加入し、脱退する際もその情報を保有したまま離れます。コマンドスイッチのパスワードが設定されていない場合、クラスタ メンバ スイッチはヌル パスワードを代わりに継承します。クラスタ メンバ スイッチが継承するのはコマンドスイッチのパスワードのみです。

コマンドスイッチのパスワードと異なるメンバスイッチのパスワードを指定してその設定を保存してしまうと、クラスタ コマンド スイッチからそのスイッチを管理できなくなります。この状態はメンバスイッチのパスワードをコマンドスイッチのパスワードに戻すまで続きます。メンバスイッチを再起動しても、パスワードは元のコマンドスイッチパスワードには戻りません。スイッチをクラスタに加入させた後は、メンバスイッチパスワードを変更しないことを推奨します。

Catalyst 1900 および Catalyst 2820 スイッチ固有のパスワードの考慮事項については、これらのスイッチのインストレーションおよびコンフィギュレーション ガイドを参照してください。

SNMP コミュニティ ストリング

クラスタ メンバ スイッチは、次のように *@esN* をコミュニティ ストリングの後ろに追加してコマンドスイッチの Read-Only（RO）と Read-Write（RW）のコミュニティ ストリングを継承します。

- *command-switch-readonly-community-string@esN*（N はメンバスイッチ番号）
- *command-switch-readwrite-community-string@esN*（N はメンバスイッチ番号）

クラスタ コマンド スイッチに複数の Read-Only または Read-Write コミュニティ ストリングがある場合、クラスタ メンバ スイッチには最初の Read-Only または Read-Write ストリングのみ伝播されます。

スイッチのコミュニティ ストリング数とその長さには制限がありません。

Catalyst 1900 および Catalyst 2820 スイッチ固有の SNMP の考慮事項については、これらのスイッチのインストレーション コンフィギュレーション ガイドを参照してください。

TACACS+ および RADIUS

Terminal Access Controller Access Control System Plus（TACACS+）をクラスタ メンバに設定する場合、すべてのクラスタ メンバに設定する必要があります。同様に、RADIUS をクラスタ メンバに設定する場合、すべてのクラスタ メンバに設定する必要があります。また、TACACS+ を設定したメンバと RADIUS を設定した他のメンバを同じスイッチ クラスタには追加できません。

LRE プロファイル

スイッチ クラスタに、個人のプロファイルと公開プロファイルの両方を使用した Long-Reach Ethernet（LRE）スイッチがある場合、設定の競合が発生します。クラスタの 1 つの LRE スイッチに公開プロファイルが割り当てられている場合、クラスタ内のすべての LRE スイッチにも同じプロファイルを割り当てる必要があります。LRE スイッチをクラスタに追加する前に、クラスタ内の他の LRE スイッチが同じ公開プロファイルを使用しているかどうかを確認してください。

クラスタ内に異なる個人プロファイルを使用している LRE スイッチを混在させることはできます。

CLI を使用したスイッチ クラスタの管理

クラスタ コマンド スイッチにログインすることにより、CLI からクラスタ メンバ スイッチを設定できます。**rcommand** ユーザー EXEC コマンドおよびクラスタメンバースイッチ番号を入力して、（コンソールまたは Telnet 接続を経由して）Telnet セッションを開始し、クラスタメンバースイッチの CLI にアクセスします。コマンド モードが変更され、通常どおりに Cisco IOS コマンドを使用できるようになります。クラスタメンバースイッチで **exit** 特権 EXEC コマンドを入力すると、コマンドスイッチの CLI に戻ります。

次に、コマンド スイッチの CLI からメンバ スイッチ 3 にログインする例を示します。

```
switch# rcommand 3
```

メンバースイッチ番号が不明の場合は、クラスタコマンドスイッチで **show cluster members** 特権 EXEC コマンドを入力します。**rcommand** コマンドおよび他のすべてのクラスタコマンドの詳細については、スイッチ コマンド リファレンスを参照してください。

Telnet セッションは、クラスタ コマンド スイッチと同じ権限レベルでメンバ スイッチの CLI にアクセスします。その後、Cisco IOS コマンドを通常どおりに使用できます。

（注） CLI により、最大 16 までのスイッチ クラスタの作成と管理がサポートされます。

SNMP を使用したスイッチ クラスタの管理

スイッチの最初の起動時にセットアップ プログラムを使用して IP 情報を入力し、提示されたコンフィギュレーションを採用した場合、SNMP はイネーブルに設定されています。セットアップ プログラムを使用して IP 情報を入力していない場合は、SNMP はイネーブルではありません。その場合は、「SNMP の設定」の説明に従って、SNMP をイネーブルに設定します。Catalyst 1900 および Catalyst 2820 スイッチでは、SNMP はデフォルトでイネーブルに設定されています。

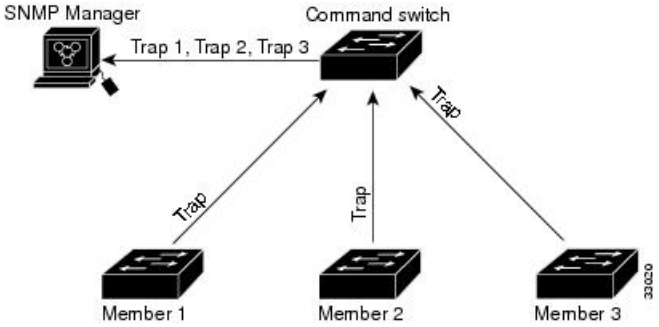
クラスタを作成すると、クラスタ コマンド スイッチがクラスタ メンバ スイッチと SNMP アプリケーション間のメッセージ交換を管理します。クラスタ コマンド スイッチ上のクラスタ ソフトウェアは、クラスタ コマンド スイッチ上で最初に設定された Read-Write および Read-Only コミュニティ ストリングにクラスタ メンバ スイッチ番号（*@esN*、N はスイッチ番号）を追加し、これらのストリングをクラスタ メンバ スイッチに送信します。クラスタ コマンド スイッチは、このコミュニティ ストリングを使用して、SNMP 管理ステーションとクラスタ メンバ スイッチ間で、get、set、および get-next メッセージの転送を制御します。

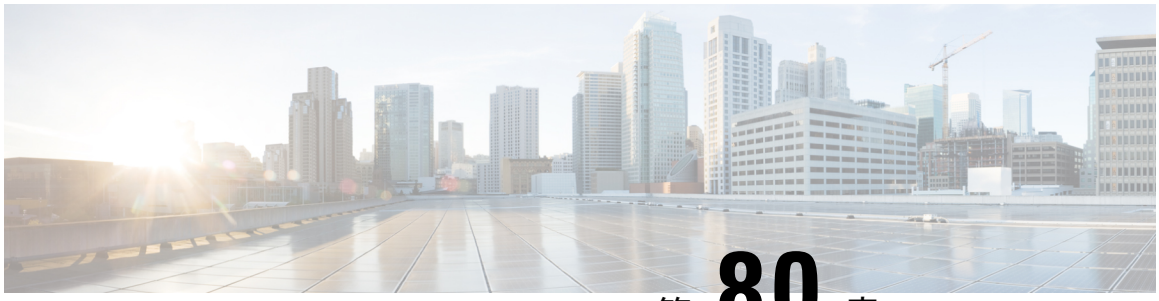
（注） クラスタ スタンバイ グループを設定すると、ユーザが気付かないうちにクラスタ コマンド スイッチが変更される場合があります。クラスタにクラスタ スタンバイ グループを設定している場合は、クラスタ コマンド スイッチとの通信には、最初に設定された Read-Write および Read-Only コミュニティ ストリングを使用してください。

クラスタ メンバ スイッチに IP アドレスが割り当てられていない場合、図に示すように、クラスタ コマンド スイッチはクラスタ メンバ スイッチからのトラップを管理ステーションにリダイレクトします。クラスタ メンバ スイッチに専用の IP アドレスおよびコミュニティ ストリングが割り当てられている場合、そのクラスタ メンバ スイッチはクラスタ コマンド スイッチを経由せず、管理ステーションに直接トラップを送信できます。

クラスタ メンバ スイッチに専用の IP アドレスとコミュニティ ストリングが割り当てられている場合、クラスタ コマンド スイッチによるアクセスの他に、その IP アドレスとコミュニティ ストリングも使用できます。

図 *130 : SNMP* によるクラスタ管理





第 80 章

DNS-AS を使用した AVC の設定

DNS-AS を使用した AVC に関する前提条件

- DNS-AS を使用するために、Cisco ONE for Access を所有している。
- マルチレイヤ スイッチ（MLS）の Quality of Service（QoS）が有効になっている。
- DNS-AS を使用した AVC を有効にする前に、権威 DNS サーバー内にデータを維持しており、到達可能である。
- DNS-AS クライアントがホストから開始されるフォワードルックアップ要求をスヌーピングできる。
- DNS パケットのロギングとスヌーピングを確実に実行するため、**service-policy input** コマンドを使用してインターフェイスにポリシーマップを付加している。

DNS-AS を使用した AVC の制約事項およびガイドライン

- この機能は Cisco Catalyst 3560-CX シリーズ スイッチ上でのみサポートされています。Cisco Catalyst 2960-CX シリーズ スイッチではサポートされていません。
- フォワード ルックアップのみがサポートされています。
- 2 台の DNS サーバーがサポートされます（フェールオーバーの場合）。1 台がプライマリ DNS サーバー、もう 1 台がセカンダリ DNS サーバーと見なされます。

- IPv6 はサポートされていません。AAA 要求、および IPv6 DNS サーバーはサポートされていません。
- DNS-AS を使用した AVC は、物理インターフェイス上の入力方向でのみサポートされています。
- Virtual Routing and Forwarding（VRF）はサポートされていません。
- TCAM（Ternary Content Addressable Memory）に影響するため、バインディング テーブル内の DNS-AS を使用した AVC アプリケーションは最大で 300 個までにすることを推奨します。アプリケーションを追加することによって TCAM にどのように影響するかについては、この章の「DNS-AS を使用した AVC のトラブルシューティング」の項を参照してください。

DNS-AS を使用した AVC について

信頼できるソースとしてのドメイン ネーム システム（DNS-AS）機能を使用した Application Visibility Control（AVC）（DNS-AS を使用した AVC）は、組織内の信頼ネットワーク トラフィックの識別と分類を制御する一元化された手段を提供します。これは、対象のドメインに対して権威のある DNS サーバーに格納されたネットワーク メタデータを使用することで行われ、アプリケーションを識別し、サービス品質（QoS）によって対応するトラフィックを分類して適切なポリシーを適用し、Flexible Netflow（FNF）によって、アプリケーション情報を監視して外部コレクタにエクスポートします。

この機能は以下を提供します。

- アプリケーションの可視性：アプリケーションの可視性を向上させます。

  DNS-AS メカニズムは要求をスヌーピングします。これには、CPU 集約型のディープ パケット インスペクション（DPI）は必要ありません。トラフィックの分類は、DPI ではなく、DNS 要求によるものであるため、この機能はネットワーク トラフィックが暗号化されているシナリオに適しています。

- メタデータ駆動：アプリケーションに関する情報を使用します。

  ネットワークを全体的にプログラムできるため、自動運転車のように動作します。トラフィックの暗号化の有無に関わらず、ネットワーク内の必要なアプリケーションすべてに関する情報を入手できます。

- 一元管理：クロスドメイン アプリケーションを対象にしたポリシー コントローラを使用します。

  この機能は、一般的に使用可能な既存のクエリ/応答メカニズムを活用して、権威サーバーとして機能するように組織内のローカル DNS サーバーを有効にし、アプリケーション分類情報をエンタープライズ ネットワーク内の DNS-AS クライアントに伝播させます。

- 管理アクセスなしの制御：コントローラベースのアプローチに代わる手段を提供します。

この機能は、ネットワークがクラウド内にあり、クラウドの所有者ではないという状況もサポートします。この場合も、これらのデバイスに対して管理制御を行えなくても、インターネットを通じてネットワーク デバイスを制御できます。

DNS-AS を使用した AVC の概要

プロセスは、ネットワーク トラフィックの管理と制御に関連する組織の要件で開始します。ネットワーク内のさまざまなホスト（電話機、PC など）で実行するソフトウェア アプリケーション、このようなデバイスがアクセスするドメイン（Web サイト）およびアプリケーション、ならびに組織内のこれらのドメインやアプリケーションのビジネス関連性を評価することから始めます。

この評価は、組織が「信頼」しているドメインやアプリケーションのリストを作成し、残りのドメインやアプリケーションはすべて信頼できないと指定するのに役立ちます。

ネットワーク上でDNS-ASを有効にし、信頼ドメインのリストを使用することで、ネットワーク内のネットワーキング デバイスや DNS-AS は、ネットワーク トラフィックが属するアプリケーションや、要求されているドメインを識別します。トラフィックが信頼リストに含まれている限り、スイッチはDNS サーバーにメタデータや IP の情報を要求します。この要求はDNS クエリの形式で送信されます。受信されるとすぐに、そのリソース レコードの存続可能時間（TTL）が切れるまで、応答がローカルにキャッシュされます。応答はトラフィックにバインドされ、DNS-AS クライアントが適切にトラフィックを識別、分類、転送できるようになります。

DNS-AS を使用した AVC の主要概念

概念	意味または定義
メタデータ（RFC6759）	DNS-AS 機能を使用した AVC では、メタデータとしてトラフィック分類情報、アプリケーション識別情報、およびビジネス関連性情報が含まれます。 メタデータはTXT レコード形式で維持されます。次に、所定の形式のメタデータ例を示します： `CISCO-CLS=app-name:example|app-class:TD|business:YES|app-id:CU/28202`
フォワード ルックアップ	ホストから発信される IP アドレスの要求、または「A」レコードの要求。 DNS-AS機能を使用したACVには、ネットワーク トラフィック内でこれらのフォワードルックアップをスヌーピングできる必要があります。
ホスト	ユーザーがソフトウェアアプリケーションを実行すると、PC やモバイルは Web サイトなどにアクセスします。 フォワード ルックアップ要求はホストから開始されます。

概念	意味または定義
クライアントまたは DNS-AS クライアント	ネットワーク全体に存在するネットワーキングデバイス。ホスト トラフィックは常にこのようなクライアントを通じてルーティングされます。 （注） この章では、アクセス スイッチとしてのみ導入されている Cisco Catalyst スイッチ上の DNS-AS を使用した AVC の設定について説明します。このドキュメント全体を通じて、「クライアント」および「DNS-AS クライアント」という用語は、DNS-AS を使用した AVC が有効になっているスイッチのことを指します。 DNS-AS クライアントは権威 DNS サーバーからメタデータを受信し、この情報のデータベースをレコード形式で維持します。クライアントのデータベースにレコードが維持される期間は、レコードの TTL によって決定されます。
バインディング テーブル	DNS-AS クライアントに存在し、解析済み DNS サーバー応答（TXT レコードと「A」レコード）のデータベースとして機能するテーブル。 各 DNS-AS クライアントには各自のバインディング テーブルがあります。 信頼ドメイン リストは信頼ドメインのみ含むリストです。このバインディング テーブルと混同しないでください。
「A」レコード	ドメイン名と IP アドレス情報（IPv4 アドレスのみ）を含むレコード。これは DNS サーバー応答の 1 つであり（もう 1 つは TXT レコード）、期限が事前に定義されています。 ホストからのフォワード ルックアップ要求は、「A」レコードの要求です。
TXT DNS-AS リソース レコードまたは TXT レコード	メタデータを含むレコード。これは、DNS サーバー応答の 1 つであり（もう 1 つは「A」レコード）、期限が事前に定義されています。 TXT レコードは 255 文字までに制限されています。 DNS-AS を使用した AVC の場合、TXT 属性は常に CISCO-CLS です。CISCO CLS= で始まるすべての TXT レコードは、DNS-AS を使用した AVC メッセージとして認識できます。このメッセージの形式は次のとおりです。 `CISCO-CLS =<option>:<val>{|<option>:<val>}*`

概念	意味または定義
存続可能時間（TTL）	バインディング テーブル内の「A」レコードと TXT レコードの期限。 TTL 値は DNS サーバー上で設定されます。 TTLは、TXT レコードと「A」レコードの両方に適用されますが、DNS クライアントはDNS サーバーからの「A」レコード応答にのみ従います。
権威 DNS サーバー	すべてのクライアント メタデータおよび「A」レコード要求で使用される DNS サーバー。 どの DNS ドメインにも、権威 DNS サーバーが 1 つのみ存在します。 このサーバーがアプリケーション メタデータのレコードを TXT レコードの形式で維持し、必要な形式で維持されているドメイン名に関するクエリにのみ、応答を返します。 次に、所定の形式のメタデータ例を示します： `CISCO-CLS=app-name:example|app-class:TD|business:YES|app-id:CU/28202`

DNS-AS プロセス フローを使用した AVC

DNS-AS を使用した AVC の動作には、DNS スヌーピング プロセスと DNS-AS クライアント プロセスが含まれます。この両方は緩やかに結びついていますが、独立したプロセスです。

DNS スヌーピング プロセス

ステップ 1 ホストが「A」レコード要求を開始します。

組織のユーザーはオフィス ビル内の会議室にいます。ここでは、関連付けられた DNS-AS クライアントはスイッチです（この会議室からのネットワーク トラフィックはこのスイッチを通じてルーティングされます）。ユーザーが Web サイトの www.example.com を検索し、それにより「A」レコードの要求が開始されます。

ステップ 2 権威 DNS サーバーが、「A」レコード応答で応答します。

DNS-AS クライアント プロセス

ステップ 1 DNS-AS クライアントは権威 DNS サーバーに DNS クエリ（TXT 要求）を送信します。

DNS-AS クライアントは、（信頼ドメインリストのエントリに対応する）要求を継続的にスヌーピングし、ホストのフォワード ルックアップ要求を検索します。DNS-AS クライアントはスヌーピングの結果に基づいて TXT 要求を権威 DNS サーバーに送信します。

（注） DNS-AS クライアントはホストの「A」レコード要求のコピーを受信しますが、ホストの元の要求をいかなる方法でも変更しません。

ステップ 2 権威 DNS サーバーは TXT レコード応答で応答します。

ステップ 3 TXT 応答の成功後に「A」レコード要求が続きます。

ステップ 4 権威 DNS サーバーが、「A」レコード応答で応答します。

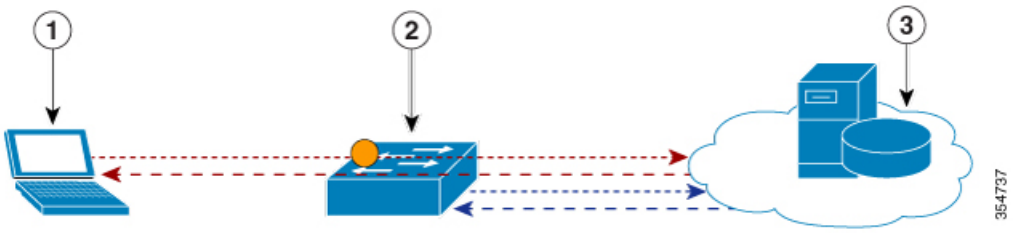
ステップ 5 DNS-AS クライアントは応答を解析し、バインディング テーブルに保存します。

DNS-AS クライアントは TXT レコードと「A」レコードをバインディング テーブルに保存します。応答は、「A」レコードの TTL で指定された期間、バインディング テーブルに保存されたままとなります。システムによって、バインディング テーブル内の完全修飾ドメイン名の重複エントリが自動的に確認され、防止されます。

DNS-AS クライアントは、（DNS サーバーから）受信したメタデータを使用して、QoS ポリシーを適用する必要があるかどうかを決定します。

DNS-AS クライアントは識別したアプリケーションに関する情報を FNF に転送し、この情報をエクスポートできるようにします。

図：DNS-AS プロセス フローを使用した AVC



1	ホスト	2	DNS-AS クライアント	3	権威 DNS サーバー

パート 1：DNS スヌーピング プロセス			
.....	ホストから DNS サーバーへの「A」レコード要求	-----	DNS サーバーからホストへの「A」レコード応答
パート II：DNS-AS クライアント プロセス			
●	DNS-AS クライアントが保存する「A」レコード要求のコピー	-	-

	DNS-AS クライアントから DNS サーバーへの TXT レコードと「A」レコード要求		DNS サーバーから DNS-AS クライアントへの TXT レコードと「A」レコード応答

DNS-AS を使用した AVC 用のデフォルト設定

DNS-AS は無効になっています。

DNS-AS を使用した AVC の設定方法

メタデータ ストリームの生成

アプリケーション メタデータは、ローカルの権威 DNS サーバーで設定され、保存されます。信頼ドメインごとに、既定の形式（メタデータ ストリーム）で、アプリケーション分類情報を設定します。これは、アプリケーション メタデータを照会されたときにサーバーがスイッチに伝達する情報です。スイッチがアプリケーションに関する TXT クエリを送信すると、DNS サーバーが TXT 応答で関連メタデータを送信します。

メタデータ ストリームを生成するには、次のタスクを実行します。

手順の概要

1. AVC リソース レコード ジェネレータに移動します。
2. メタデータ ストリームを生成するオプションのいずれかをクリックします。
   - Generate predefined
   - Generate custom
3. 信頼ドメインとしてマークした DNS ドメインを担う DNS サーバーの対応する TXT リソース レコードにメタデータをコピーします。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	AVC リソース レコード ジェネレータに移動します。 例： `CISCO-CLS=app-name:example|app-class:TD|business:YES|app-id:CU/28202`	これは、アプリケーションやドメインに TXT レコード形式でメタデータ ストリームを生成するのに役立ちます。 次のメタデータ フィールドを指定できます。 • （任意）ドメイン名

	コマンドまたはアクション	目的
		• （必須）アプリケーション名：値が必須です。既存のアプリケーション名またはカスタムアプリケーション名を使用できます。 • 既存のアプリケーション名（**app-name:**）：標準アプリケーションのリストから選択します。 • （任意）カスタム アプリケーション名（**app-name:**）：カスタム アプリケーション名を入力する場合は、メタデータ ストリーム内にもトラフィック クラスとビジネス関連性情報を維持する必要があります。 • （任意）セレクタ ID（**app-id:**）：分類エンジン ID（最初の 8 ビット）とセレクタ ID（次の 24 ビット）から構成されます。 • エンジン ID または分類エンジン ID：セレクタ ID のコンテキストを定義します。次のエンジン ID のみが使用できます。 L3：IANA レイヤ 3 のプロトコル番号 L4：IANA レイヤ 4 のウェルノウン ポート番号 L7：シスコのグローバルアプリケーション ID CU：カスタム プロトコルこのエンジン ID をカスタムアプリケーション名に使用します。 • セレクタ ID：所定の分類エンジン ID のアプリケーション ID。1 ～ 65535 の数値を入力します。 （注） 既存のアプリケーション名にエンジン ID とセレクタ ID を入力する場合は、Network Based Application Recognition（NBAR）の標準に適合させる必要があります。適合させた後でのみ、FNF エクスポータが共通の ID を一貫した方法で報告します。 • （任意）ポート範囲（**server-port:**）

	コマンドまたはアクション	目的
		• （任意）トラフィック クラス（**app-class:**） • （任意）ビジネス関連性（**business:**）：yes または no を選択しなかった場合、ビジネス関連性の値は app-class または app-name に基づき、その優先順位の順序で設定されます。 ここでトラフィック クラスとビジネス関連性フィールドが QoS トラフィック分類にマッピングされる方法については、「アプリクラスと QoS トラフィックのマッピング」を参照してください。
ステップ 2	メタデータ ストリームを生成するオプションのいずれかをクリックします。 • Generate predefined • Generate custom **例**： `Generate predefined`	**Generate predefined**：既知のアプリケーションに事前に定義されたメタデータ ストリームを、ベスト プラクティスのデフォルト値を使用して生成します。 **Generate custom**：独自のアプリケーションのカスタム メタデータ ストリームをカスタム値を使用して生成します。
ステップ 3	信頼ドメインとしてマークした DNS ドメインを担う DNS サーバーの対応する TXT リソース レコードにメタデータをコピーします。	メタデータ ストリームを Web サイトからコピーし、使用している権威 DNS サーバーに貼り付けます。

権威サーバーとしての DNS サーバーの設定

すべての DNS クエリを 1 台の権威 DNS サーバーに送信するように、ネットワーク内のすべての DNS-AS クライアントを設定する必要があります。Cisco Catalyst スイッチで次のタスクを実行します。

手順の概要

1. **configure terminal**
2. **ip name-server***server-address*

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**ip name-server***server-address* 例： スイッチ(config)# **ip name-server server-address 192.0.2.1 192.0.2.2**	権威 DNS サーバーの IP アドレスを指定します。ポート番号は常に 53 です。 フェールオーバーに備えて最大2 台の DNS サーバーを設定できます。 （注） このコマンドを使用すると、最大 6 台のネーム サーバー（IPv4 および IPv6）を設定できます。シーケンス内の最初の 2 つ以上の IP アドレスを IPv4 アドレスにします。これは、DNS-AS 機能を使用した AVC がこれらのみを使用するためです。次の例では、最初の 2 つのアドレスが IPv4（192.0.2.1 および 192.0.2.2）、3 番目のアドレス（2001:DB8::1）は IPv6 アドレスです。DNS-AS を使用した AVC は最初の 2 つを使用します。 スイッチ(config)# **ip name-server 192.0.2.1 192.0.2.2 2001:DB8::1**

DNS-AS を使用した AVC の有効化

DNS-AS はデフォルトで無効になっています。Cisco Catalyst スイッチで機能を有効にするには、次のタスクを実行します。

手順の概要

1. **configure terminal**
2. ［**no**］**avc dns-as client enable**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	［**no**］**avc dns-as client enable** 例： スイッチ(config)# **avc dns-as client enable**	スイッチで DNS-AS を使用した AVC（DNS-AS クライアント）を有効にします。 次に、システムによりバインディング テーブルが作成されます。このバインディング テーブルでは、解

	コマンドまたはアクション	目的
		析した DNS サーバー応答が TTL が期限切れになるまで保存されます。 （注） DNS パケットロギングやスヌーピングを確実に実行するには、（トラフィッククラスを決定する関連クラスマップを含んでいる）ポリシーマップを **service-policy input** コマンドを使用してインターフェイスに付加する必要があります。詳細については、DNS-AS を使用した AVC 用 QoS の設定（1910 ページ）を参照してください。

信頼ドメインのリストの維持

信頼ドメインは、DNS-AS を使用した AVC が有効になっている DNS-AS クライアントごとに保存されます。DNS-AS クライアントで機能が最初に有効になった時点では、リストは空です。スイッチで信頼すべきドメインを入力する必要があります。スイッチは、このリストに維持されているネットワーク トラフィックのみをスヌーピングします。信頼ドメイン リストにエントリを作成するには、次のタスクを実行します。

手順の概要

1. **configure terminal**
2. ［**no**］ **avc dns-as client trusted-domains**
3. ［**no**］**domain** *domain-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	［**no**］ **avc dns-as client trusted-domains** 例： `スイッチ(config)# avc dns-as client trusted-domains`	信頼ドメイン コンフィギュレーション モードを開始します。
ステップ 3	［**no**］**domain** *domain-name* 例： `スイッチ(config-trusted-domains)# domain www.example.com` OR	信頼ドメイン リストに追加するドメイン名を入力します。これにより、DNS-AS クライアントの信頼ドメイン リストの一部が形成されます。残りのすべてのドメインは無視され、デフォルトの転送動作に従います。

	コマンドまたはアクション	目的
	スイッチ(config-trusted-domains)# **domain *example.com**	最大 50 ドメインを入力できます。 ドメイン名の照合には、正規表現を使用できます。たとえば、組織のすべてのドメインを表現するために Switch(config-trusted-domains)# domain *.example.*, を入力した場合、DNS-AS クライアントは www.example.com、ftp.example.org、および、組織「example」に関連するその他のすべてのドメインを照合します。ただし、このようなエントリは自身の裁量で使用してください。バインディングテーブルのサイズを大幅に拡大させる可能性があります。

DNS-AS を使用した AVC 用 QoS の設定

信頼できるトラフィックをメタデータ ストリームに定義されているように分離し、分類するには、クラスマップを作成し（トラフィック クラスごとに1つ）、トラフィッククラスの一致基準とビジネス関連性の一致基準を定義し、ポリシーマップを作成し、クラスマップを追加し、アクションを設定し、ポリシーマップをインターフェイスに付加する必要があります。詳細については、このガイドの「*QoS* の設定」の章の「」「」「分類の概要」の項を参照してください。

簡単な QoS モデルのクラス マップの設定

プロビジョニングする必要があるトラフィック クラスの数を特定するには、12 クラスの簡単な QoS モデルを使用します。このモデルは、統一された標準ベースの推奨事項を提供し、QoS 設計と導入の組織全体にわたる均一性と一貫性を保証するのに役立ちます。次の出力例では、12 クラスの簡単な QoS モデルに従い、トラフィック クラスとビジネス関連性のクラス マップ設定が表示されています。

（注） DNS-AS 機能でのみ、各クラスに 2 つの一致属性を指定できます。

```
class-map match-all VOICE
match protocol attribute traffic-class voip-telephony
match protocol attribute business-relevance business-relevant
class-map match-all BROADCAST-VIDEO
match protocol attribute traffic-class broadcast-video
match protocol attribute business-relevance business-relevant
class-map match-all REAL-TIME-INTERACTIVE
match protocol attribute traffic-class real-time-interactive
match protocol attribute business-relevance business-relevant
class-map match-all MULTIMEDIA-CONFERENCING
match protocol attribute traffic-class multimedia-conferencing
match protocol attribute business-relevance business-relevant
class-map match-all MULTIMEDIA-STREAMING
match protocol attribute traffic-class multimedia-streaming
match protocol attribute business-relevance business-relevant
```

```
class-map match-all SIGNALING
match protocol attribute traffic-class signaling
match protocol attribute business-relevance business-relevant
class-map match-all NETWORK-CONTROL
match protocol attribute traffic-class network-control
match protocol attribute business-relevance business-relevant
class-map match-all NETWORK-MANAGEMENT
match protocol attribute traffic-class ops-admin-mgmt
match protocol attribute business-relevance business-relevant
class-map match-all TRANSACTIONAL-DATA
match protocol attribute traffic-class transactional-data
match protocol attribute business-relevance business-relevant
class-map match-all BULK-DATA
match protocol attribute traffic-class bulk-data
match protocol attribute business-relevance business-relevant
class-map match-all SCAVENGER
match protocol attribute business-relevance business-irrelevant
```

簡単な QoS モデルのポリシー マップの定義

次の出力例では、ポリシー マップの定義と、12 クラスの簡単な QoS モデルですべてのトラフィック クラスをマーキングするトラフィック属性が表示されています。

```
policy-map MARKING
class VOICE
set dscp ef
class BROADCAST-VIDEO
set dscp cs5
class REAL-TIME-INTERACTIVE
set dscp cs4
class MULTIMEDIA-CONFERENCING
set dscp af41
class MULTIMEDIA-STREAMING
set dscp af31
class SIGNALING
set dscp cs3
class NETWORK-CONTROL
set dscp cs6
class NETWORK-MANAGEMENT
set dscp cs2
class TRANSACTIONAL-DATA
set dscp af21
class BULK-DATA
set dscp af11
class SCAVENGER
set dscp cs1
class class-default
set dscp default
```

アプリクラスと QoS トラフィックのマッピング

次の表に、メタデータ ストリーム マップの `app-class` フィールドをトラフィック分類の 12 クラスの簡単な QoS モデルにマッピングする方法を示します。

アプリクラスと QoS トラフィックのマッピング

アプリケーションクラスの長いテキスト	アプリケーションクラスの短いテキスト	対応する Qos トラフィック クラス名とビジネス関連性
VOIP-TELEPHONY	VO	トラフィック クラス = voip-telephony ビジネス関連性 = YES
BROADCAST-VIDEO	BV	トラフィック クラス = broadcast-video ビジネス関連性 = YES
REALTIME-INTERACTIVE	RTI	トラフィック クラス = real-time-interactive ビジネス関連性 = YES
MULTIMEDIA-CONFERENCING	MMC	トラフィック クラス = multimedia-conferencing ビジネス関連性 = YES
MULTIMEDIA-STREAMING	MMS	トラフィック クラス = multimedia-streaming ビジネス関連性 = YES
NETWORK-CONTROL	NC	トラフィック クラス = network-control ビジネス関連性 = YES
SIGNALING	CS	トラフィック クラス = Signaling ビジネス関連性 = Yes
OPS-ADMIN-MGMT	OAM	トラフィック クラス = ops-admin-mgmt ビジネス関連性 = YES
TRANSACTIONAL-DATA	TD	トラフィック クラス = Transactional-Data ビジネス関連性 = YES
BULK-DATA	BD	トラフィック クラス = bulk-data ビジネス関連性 = YES

アプリケーションクラスの長いテキスト	アプリケーションクラスの短いテキスト	対応する Qos トラフィック クラス名とビジネス関連性
BEST-EFFORT	BE	トラフィック クラス = <no change> ビジネス関連性 = default
SCAVENGER	SCV	トラフィック クラス = <no change> ビジネス関連性 = NO

ネットワーク制御トラフィックの分類

次に、ネットワーク制御トラフィックを分類する例を示します。維持する必要がある対応するメタデータは `CISCO-CLS=app-name:example|app-class:NC|business:YES` です。

1. クラス マップを作成し、属性を一致させます。

```
スイッチ# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
スイッチ(config)# class-map NETWORK-CONTROL
スイッチ(config-cmap)# match protocol attribute traffic-class network-control
スイッチ(config-cmap)# match protocol attribute business-relevance business-relevant
スイッチ(config-cmap)# end
```

2. ポリシー マップを作成し、それにクラス マップを付加して、優先順位を指定します。

```
スイッチ# configure terminal
スイッチ configuration commands, one per line. End with CNTL/Z.
スイッチ(config)# policy-map MARKING
スイッチ(config-pmap)# class NETWORK-CONTROL
スイッチ(config-pmap-c)# set dscp ef
スイッチ(config-pmap-c)# end
```

3. インターフェイスにポリシー マップを付加します。

```
スイッチ# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
スイッチ(config)# interface tengigabitethernet 1/0/1
スイッチ(config-if)# service-policy input MARKING
スイッチ(config-if)# end
```

DNS-AS を使用した AVC 用 FNF の設定

FNF を使用すると、ネットワーク上で実行されているアプリケーションについての可視性が得られ、FNF オプション テンプレートを使用してアプリケーションの ID、説明、および属性の情報をエクスポートできます。DNS-AS クライアント上では、次の FNF 設定を行う必要があります。

- 非キーフィールド **application-name**、キーフィールドの **ipv4 source address** と **ipv4 destination address** を収集するためのフローレコードを設定します。
- フロー エクスポータと 2 つのオプション テンプレートを設定します。オプション テンプレートは、アプリケーションの情報を取得します。

  オプションテンプレート **application-table**：DNS-AS クライアントによって解決されたアプリケーションのみをエクスポートします。つまり、バインディングテーブルからアプリケーション ID と名前のみがエクスポートされます。対応するアプリケーションの記述は、標準的なアプリケーションの Network Based Application Recognition（NBAR）からのものです。構築化されたヘルプ文字列は、カスタム アプリケーションに使用されます。

  オプションテンプレート **application-attributes** は、アプリケーション名にマッピングすることによって属性情報を取得します。標準的なアプリケーション名を使用した場合、オプション テンプレートは標準的な Network Based Application Recognition（NBAR）属性の定義が使用されます。カスタムアプリケーション名が使用された場合、ユーザー定義のアプリケーションと特定の属性フィールドのみが値を確実に伝達します。
- フロー モニターを設定し、それをインターフェイスに適用することで、ネットワーク トラフィックの監視を有効にします。

DNS-AS を使用した FNF インタラクション：フローテーブルで作成されたすべてのフローで、DNS-AS クライアントは宛先 IP アドレスまたは送信元 IP アドレス（使用できない場合）を使用し、フローのアプリケーション名を解決します（バインディングテーブルにエントリが存在する場合）。

FNF は、対応するアプリケーションにマッピングされているオプション テンプレート データを設定された間隔（デフォルトでは 600 秒）で定期的に外部コレクタにエクスポートします。

オプション テンプレート

application-table および **application-attributes** オプションテンプレートがサポートされています。オプションテンプレートにより、外部コレクタにエクスポートする情報が決定されます。

option application-table

このテンプレートは、アプリケーション名、アプリケーション タグ、および説明を外部コレクタにエクスポートします。

DNS-AS を使用した AVC が有効になっているデバイスでは、DNS-AS クライアントによって解決されたアプリケーションのみがエクスポートされます。ただし、永続的な機能として、application-table テンプレートは、この機能が有効になっているかどうかにかかわらず、**unclassified** と **unknown** のアプリケーションをエクスポートします。

- アプリケーション名：カスタム アプリケーションおよび標準アプリケーションの場合、この情報はバインディング テーブルに保存されている TXT 応答（**app-name:**）から抽出されます。
- アプリケーション タグ：DNS-AS 機能を使用した AVC では、アプリケーション ID と同じです。エンジン ID とセレクタ ID で構成されています。

- エンジン ID または分類エンジン ID：セレクタ ID のコンテキストを定義します。次の値のみがサポートされています。
  - L3：IANA レイヤ 3 プロトコル番号（IANA_L3_STANDARD、ID：1）
  - L4：IANA レイヤ 4 のウェルノウン ポート番号（IANA_L4_STANDARD、ID：3）
  - L7：シスコのグローバル アプリケーション ID（CISCO_L7_GLOBAL、ID：13）
  - CU：カスタム プロトコル（NBAR_CUSTOM、ID：6）
- セレクタ ID：アプリケーションまたは分類を一意に識別します。

標準アプリケーションの場合、アプリケーション タグ情報は次の送信元から記載されている順に抽出されます。

1. TXT 応答（**app-id:**）
2. 標準的なアプリケーションの NBAR 定義（TXT 応答が値を持たない場合）

カスタム アプリケーションの場合は、アプリケーション タグ情報に次が適用されます。

- TXT 応答（**app-id:**）からのみ抽出されます。
- エンジン ID の場合、DNS-AS クライアントが自動的に CU（カスタム プロトコル）を使用します（NBAR_CUSTOM、ID：6）。
- セレクタ ID の場合、DNS-AS クライアントがカスタム セレクタ ID を割り当てます。最大 120 のカスタム アプリケーションがサポートされます。その中の 110 のカスタム アプリケーションを DNS-AS クライアントに使用できます。セレクタ ID 値 243 以降は、降順で ID が割り当てられます。割り当てる ID がなくなった場合、エントリはバインディング テーブルに保存されません。

- 説明：この情報は、標準アプリケーションの NBAR 定義から抽出されます。カスタム アプリケーションの場合、DNS-AS クライアントはユーザー定義のプロトコル <app-name> を使用します。

option application-attributes

このテンプレートは、コレクタが属性にアプリケーション名を（オプションの application-table から）マッピングできるようにします。属性は、プロトコルまたはアプリケーションごとに静的に割り当てられ、トラフィックには依存しません。このテンプレートでは、次の属性がサポートされています。

標準アプリケーションの場合：

- アプリケーション タグ：上記の option application-table セクションのアプリケーション タグの情報を参照してください。ここでも同じことが当てはまります。

- カテゴリ：一致基準として、各プロトコルのカテゴリ化の最初のレベルに基づいてアプリケーションをグループ化します。類似したアプリケーションが1つのカテゴリにまとめてグループ化されます。たとえば、電子メール カテゴリには、Internet Mail Access Protocol（IMAP）、Simple Mail Transfer Protocol（SMTP）、Lotus Notes などのすべての電子メール アプリケーションが含まれます。
- サブカテゴリ：一致基準として、各プロトコルのカテゴリ化の2番目のレベルに基づいてアプリケーションをグループ化します。たとえば、clearcase、dbase、rda、mysql、その他のデータベース アプリケーションはデータベース グループにグループ化されます。
- アプリケーション グループ：同じネットワー キング アプリケーションをまとめてグループ化します。たとえば、Example-Messenger、Example-VoIP-messenger、および Example-VoIP-over-SIP を example-messenger-group の下にまとめてグループ化します。
- ピアツーピア（p2p）：p2p テクノロジーを使用するかどうかに基づいてプロトコルをグループ化します。
- トンネル：プロトコルが他のプロトコルのトラフィックをトンネルするかどうかに基づいてプロトコルを分類します。NBARが値を指定しないプロトコルは、未割り当てのトンネル グループに分類されます。たとえば、レイヤ 2 トンネリング プロトコル（L2TP）などです。
- 暗号化：アプリケーションの暗号化と非暗号化のステータスに基づいてアプリケーションをグループ化します。NBARが値を指定しないプロトコルは、未割り当ての暗号化グループに分類されます。
- トラフィックのクラス：所属するトラフィック クラスに基づいてアプリケーションとプロトコルを分類します。たとえば、トラフィック クラス TD のすべてのアプリケーションが挙げられます。トラフィック クラス情報は、次の送信元から記載されている順に抽出されます。

  1. TXT 応答（**app-class:**）
  2. 標準的なアプリケーションの NBAR 定義（TXT 応答が値を持たない場合）

- ビジネスの関連性：ビジネスに関連があるとマークされているかどうかに基づいてアプリケーションをグループ化します。たとえば、ビジネス関連性が YES のすべてのアプリケーションが挙げられます。ビジネス関連性情報は、次の送信元から記載されている順に抽出されます。

  1. TXT 応答（**business**:）
  2. 標準的なアプリケーションの NBAR 定義（TXT 応答が値を持たない場合）

カスタム アプリケーションの場合：

application-attributes オプション テンプレートの次の属性のみが値を伝送することが保証されています。

- アプリケーション タグ：上記の option application-table セクションのアプリケーション タグの情報を参照してください。ここでも同じことが当てはまります。

- トラフィック クラス：この情報は TXT 応答（**app-class:**）から抽出されます。
- ビジネスの関連性：この情報は TXT 応答（**business:**）から抽出されます。

DNS-AS を使用した AVC 用 FNF 設定の例

次に、DNS-AS を使用した AVC 用 FNF を設定する例を示します。

パート 1：フロー レコードを作成します。例に示すように設定する必要があります。

- アプリケーション名を解決するための、key フィールドとしての送信元と宛先の IP アドレス。
- フロー レコード内の nonkey フィールドとしてのアプリケーション名の使用。

さらに、フロー内のバイトまたはパケット数を nonkey フィールドとして設定して、コレクタに送信するアプリケーションの数を表示することもできます（オプション）。

```
スイッチ# configure terminal
スイッチ(config)# flow record example-record1
スイッチ(config-flow-record)# match ipv4 source address
スイッチ(config-flow-record)# match ipv4 destination address
スイッチ(config-flow-record)# collect application name
スイッチ(config-flow-record)# collect counter packets
スイッチ(config-flow-record)# exit

スイッチ# show flow record example-record1
flow record example-record1
 match ipv4 source address
 match ipv4 destination address
 collect application name
 collect counter packets
```

パート 2：フロー エクスポータを作成します。

また、**application-table** オプション テンプレートと **application-attributes** オプション テンプレートもエクスポータ内に設定します。オプション テンプレートを使用しないと、コレクタは意味のあるアプリケーション情報を取得できません。少なくとも、application-table オプションを設定することを推奨します。属性情報の場合は、application-attribute オプションも設定します。

また、テンプレートをエクスポートする頻度を秒単位で変更することもできます（許容範囲は 1 ～ 86400 秒、デフォルト値は 600 秒）。

```
スイッチ(config)# flow exporter example-exporter1
スイッチ(config-flow-exporter)# option application-table
スイッチ(config-flow-exporter)# option application-attributes
スイッチ(config-flow-exporter)# template data timeout 500
スイッチ(config-flow-exporter)# exit

スイッチ# show flow exporter example-exporter1
Flow Exporter example-exporter1:
  Description:              User defined
  Export protocol:          NetFlow Version 9
  Transport Configuration:
```

```
    Destination IP address: 192.0.1.254
    Source IP address:      192.51.100.2
    Transport Protocol:     UDP
    Destination Port:       9995
    Source Port:            54964
    DSCP:                   0x0
    TTL:                    255
    Output Features:        Not Used
  Options Configuration:
    application-table (timeout 500 seconds)
    application-attributes (timeout 500 seconds)

スイッチ# show flow exporter example-exporter1 statistics
Flow Exporter example-exporter1:
  Packet send statistics (last cleared 00:00:48 ago):
    Successfully sent:         2                     (924 bytes)

  Client send statistics:
    Client: Option options application-name
      Records added:           4
        - sent:                4
      Bytes added:             332
        - sent:                332

    Client: Option options application-attributes
      Records added:           2
        - sent:                2
      Bytes added:             388
        - sent:                388
```

パート 3：フロー モニターを作成します。

フローモニターをインターフェイスに適用し、ネットワーク トラフィックの監視を実行します。

また、同じインターフェイスに QoS ポリシーも適用できます。次のれいでは、同じ QoS 設定の一部として作成された QoS ポリシーを適用しています（ネットワーク制御トラフィックの分類 （1913 ページ））。

```
スイッチ# configure terminal
スイッチ(config)# flow monitor example-monitor1
スイッチ(config-flow-monitor)# record example-record1
スイッチ(config-flow-monitor)#  exporter exporter-export1
スイッチ(config-flow-monitor)# exit
スイッチ(config)# interface tengigabitethernet 1/0/1
スイッチ(config-if)# switchport acceess vlan 100
スイッチ(config-if)# switchport mode access
スイッチ(config-if)# ip flow monitor example-monitor1 input
スイッチ(config-if)# service-policy input MARKING
スイッチ(config-if)# end

スイッチ# show flow monitor
flow monitor example-monitor1
 record example-record1
 exporter example-exporter1
!
スイッチ# show interface tengigabitethernet1/0/1
interface tengigabitethernet1/0/1
 switchport access vlan 100
```

```
 switchport mode access
ip flow monitor example-monitor1 input

スイッチ# show flow monitor example-monitor1 cache
  Cache type:                               Normal
  Cache size:                                16640
  Current entries:                               3
  High Watermark:                                3

  Flows added:                                   6
  Flows aged:                                    3
    - Active timeout      (  1800 secs)          0
    - Inactive timeout    (    30 secs)          3
    - Event aged                                 0
    - Watermark aged                             0
    - Emergency aged                             0

IPV4 SOURCE ADDRESS:       192.0.1.254
IPV4 DESTINATION ADDRESS:  192.51.100.2
counter packets long:      7479
application name:          appexample1

IPV4 SOURCE ADDRESS:       192.51.100.11
IPV4 DESTINATION ADDRESS:  203.0.113.125
counter packets long:      445
application name:          appexample2

IPV4 SOURCE ADDRESS:       192.51.51.51
IPV4 DESTINATION ADDRESS: 203.0.113.100
counter packets long:      14325
application name:          appexample3
Switch#
```

パート 4：その他の関連 **show** コマンド

```
スイッチ# show avc dns-as client binding-table detail
DNS-AS generated protocols:
 Max number of protocols      :50
 Customization interval [min] :N/A

 Age            :   The amount of time that the entry is active
 TTL            :   Time to live which was learned from DNS-AS server
 Time To Expire :   Entry expiration time in case device does not see DNS traffic for
the entry host

Protocol-Name        : appexample1
 VRF                 : <default>
 Host                : www.appexample1.com
 Age[min]            : 2
 TTL[min]            : 60
 Time To Expire[min] : 58
 TXT Record          : app-name:appexample1|app-class:VO|business:YES
 Traffic Class       : voip-telephony
 Business Relevance  : business relevant
 IP                  : 192.0.1.254

Protocol-Name        : appexample2
 VRF                 : <default>
 Host                : www.appexample2.com
 Age[min]            : 2
 TTL[min]            : 60
 Time To Expire[min] : 58
```

```
 TXT Record          : app-name:appexample2|app-class:VO|business:YES
 Traffic Class       : voip-telephony
 Business Relevance  : business relevant
 IP                  : 192.51.100.11

<output truncated>


スイッチ# show flow exporter option application engines
Engine: prot (IANA_L3_STANDARD, ID: 1)
Engine: port (IANA_L4_STANDARD, ID: 3)
Engine: NBAR (NBAR_CUSTOM, ID: 6)
Engine: cisco (CISCO_L7_GLOBAL, ID: 13)


スイッチ# show flow exporter option application table
Engine: prot (IANA_L3_STANDARD, ID: 1)
appID   Name    Description
-----   ----    -----------

Engine: port (IANA_L4_STANDARD, ID: 3)
appID   Name    Description
-----   ----    -----------

Engine: NBAR (NBAR_CUSTOM, ID: 6)
appID   Name    Description
-----   ----    -----------
6:28202 appexample1 User defined protocol appexample1

Engine: cisco (CISCO_L7_GLOBAL, ID: 13)
appID   Name    Description
-----   ----    -----------
13:0  unclassified Unclassified traffic
13:1   unknown     Unknown application
13:518  appexample2 appexample2, social web application and service
```

DNS-AS を使用した AVC の監視

設定した、さまざまな DNS-AS を使用した AVC 設定を表示するには、特権 EXEC モードで次のコマンドを使用します。

表 *162 : DNS-AS* を使用した *AVC* の監視コマンド

コマンド	目的	出力の例
show avc dns-as client status	DNS-AS クライアントの現在のステータスを表示します。このコマンドを使用すると、DNS-AS を使用した AVC が有効になっているかどうかを知ることができます。	例：show avc dns-as client status
show avc dns-as client trusted-domains	バインディング テーブルに維持されている信頼ドメインのリストを表示します。	例：show avc dns-as client trusted-domains

コマンド	目的	出力の例
show avc dns-as client binding-table および **show avc dns-as client binding-table detail**	信頼ドメインと解決済みエントリのリスト用の DNS-AS を使用した AVC のメタデータを表示します。アプリケーション名やドメイン名などで、出力をフィルタリングできます。 どちらのコマンドも、異なる形式で同じ情報を表示します。	例：show avc dns-as client binding-table
show avc dns-as client statistics	パケット ロギング情報（送信した DNS クエリの数と受信した応答の数）を表示します。	例：show avc dns-as client statistics
show avc dns-as client name-server brief	メタデータ要求の送信先の DNS サーバーに関する情報を表示します。	例：show avc dns-as client name-server brief
show ip name-server	維持されているすべてのネームサーバーの IP アドレスを表示します。	例：show ip name-server
show platform tcam utilization	TCAM の可用性に関する情報を表示します。	例：show platform tcam utilization

例：show avc dns-as client status

```
スイッチ# show avc dns-as client status
DNS-AS client is enabled
```

表 162 : DNS-AS を使用した AVC の監視コマンドに戻る

例：show avc dns-as client trusted-domains

```
スイッチ# show avc dns-as client trusted-domains
Id   | Trusted domain
-----------------------------------------------------
    1| example.com
    2| www.example.com
    3| example.net
    4| www.example.net
    5| example.org
    6| www.example.org
```

表 162 : DNS-AS を使用した AVC の監視コマンドに戻る

例：show avc dns-as client binding-table

```
スイッチ# show avc dns-as client binding-table
 Switch# show avc dns-as client binding-table detailed
DNS-AS generated protocols:
Max number of protocols :50
Customization interval [min] :N/A

Age : The amount of time that the entry is active
TTL : Time to live which was learned from DNS-AS server
Time To Expire : Entry expiration time in case device does not see DNS traffic for the
entry host

Protocol-Name : example
VRF : <default>
Host : www.example.com
Age[min] : 2
TTL[min] : 60
Time To Expire[min] : 58
TXT Record : app-name:example|app-class:VO|business:YES
Traffic Class : voip-telephony
Business Relevance : business relevant
IP : 192.0.2.121
: 192.0.2.254
: 198.51.100.1
: 198.51.100.254
: 192.51.100.12
: 203.0.113.125
<output truncated>
```

表 162 : DNS-AS を使用した AVC の監視コマンドに戻る

例：show avc dns-as client statistics

（注） この例では、2 つの DNS サーバーが設定されます。

```
スイッチ# show avc dns-as client statistics
Server details: vrf-id = 0 vrf-name = <default> ip = 192.0.2.1
AAAA Query Error packets 0
AAAA Query TX packets 0
AAAA Response RX packets 0
TXT Query Error packets 0
TXT Query TX packets 8
TXT Response RX packets 0
A Query Error packets 0
A Query TX packets 6
A Response RX packets 0
Server details: vrf-id = 0 vrf-name = <default> ip = 192.0.2.2
AAAA Query Error packets 0
AAAA Query TX packets 0
AAAA Response RX packets 0
TXT Query Error packets 0
TXT Query TX packets 2
TXT Response RX packets 2
A Query Error packets 0
A Query TX packets 4
A Response RX packets 2
Total Drop packets 0

avc_dns_as_pkts_logged = 2
avc_dns_as_q_pkts_processed = 2
```

表 162 : DNS-AS を使用した AVC の監視コマンドに戻る

例：show avc dns-as client name-server brief

```
スイッチ# show avc dns-as client name-server brief

Server-IP | Vrf-name
-----------------------------------------------------
192.0.2.1 | <default>
192.0.2.2 | <default>
```

表 162 : DNS-AS を使用した AVC の監視コマンドに戻る

例：show ip name-server

```
スイッチ# show ip name-server
192.0.2.1
192.0.2.2
2001:DB8::1
```

表 162 : DNS-AS を使用した AVC の監視コマンドに戻る

例：show platform tcam utilization

（注） 関連する TCAM エントリは `IPv4 qos aces:` です。

```
スイッチ# show platform tcam utilization
CAM Utilization for ASIC# 0 Max Used
Masks/Values Masks/values

Unicast mac addresses: 16604/16604 24/24
IPv4 IGMP groups + multicast routes: 1072/1072 3/3
IPv4 unicast directly-connected routes: 4096/4096 4/4
IPv4 unicast indirectly-connected routes: 1280/1280 40/40
IPv6 Multicast groups: 1072/1072 18/18
IPv6 unicast directly-connected routes: 4096/4096 1/1
IPv6 unicast indirectly-connected routes: 1280/1280 32/32
IPv4 policy based routing aces: 512/512 14/14
IPv4 qos aces: 512/512 51/51
IPv4 security aces: 1024/1024 78/78
IPv6 policy based routing aces: 256/256 8/8
IPv6 qos aces: 256/256 44/44
IPv6 security aces: 512/512 18/18

Note: Allocation of TCAM entries per feature uses
a complex algorithm. The above information is meant
to provide an abstract view of the current TCAM utilization
```

表 162 : DNS-AS を使用した AVC の監視コマンドに戻る

DNS-AS を使用した AVC のトラブルシューティング

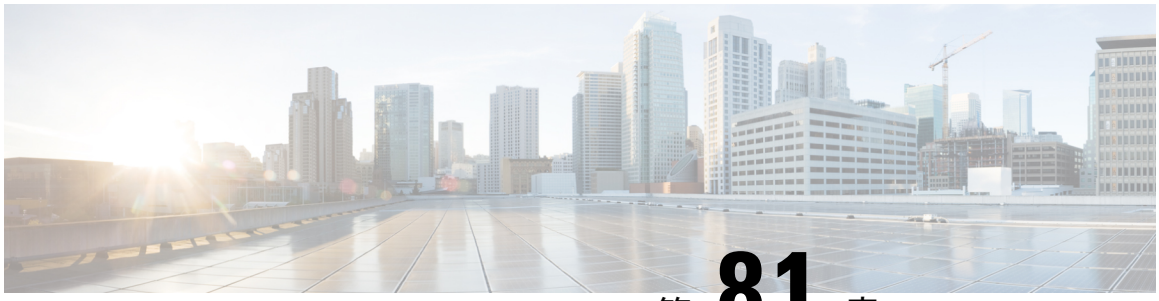
問題	考えられる原因と解決策
バインディング テーブルにエントリがない。	バインドテーブルが、次の理由のいずれか、または両方によって空になっている可能性があります。 • DNS サーバーでメタデータが維持されていない — 次のタスクを完了してください：メタデータ ストリームの生成（1905 ページ） • 信頼ドメインリストでエントリが維持されていない — 次のタスクを完了してください：信頼ドメインのリストの維持（1909 ページ）
DNS スヌーピングまたはパケット ロギングに失敗する。	DNS スヌーピングおよびパケット ロギングを確実に実行するには、ポリシー マップ（トラフィック クラスを決定する関連クラス マップが含まれている）をインターフェイスに付加する必要があります。次の例を参照してください：DNS-AS を使用した AVC 用 QoS の設定（1910 ページ）
DNS サーバーが不正な値を返す。	正しい DNS-AS メタデータが DNS システムに維持されていることを確認します。 • Linux の dig を次のように使用します。 `dig TXT +short www.example.org [dns-server-ip]` `"CISCO-CLS=app-name:example|app-class:TD|business:YES|app-id:CU/28202"` • Windows nslookup を次のように使用します。 `C:\Windows\system32>NSLookup.exe -q=TXT www.example.org [dns-server-ip]` `www.example.org text =` `"CISCO-CLS=app-name:example|app-class:TD|business:YES|app-id:CU/28202"`
適用した QoS ポリシーがポートから削除されている。	DNS-AS クライアントがアプリケーションを認識し、「A」レコード応答がバインディング テーブルに保存されると、システムは TCAM を使用してそのアプリケーションの IP アドレスを保存します。事実上、単一のアプリケーションが複数の IP アドレスを持つことができ、各アプリケーションが TCAM のスペースをさらに使用します。TCAM が枯渇すると、QoS ポリシーは適用を停止します。 この問題を回避するには、定期的に TCAM の使用率を監視します。TCAM の可用性に関する情報を表示するには、**show platform tcam utilisation** コマンドを特権 EXEC モードで入力します。

問題	考えられる原因と解決策
DNS-AS クライアントが、定義した QoS マッピングを無視し、デフォルトの転送動作を適用します。	次の場合に、DNS-AS クライアントが、QoS マッピングを無視し、デフォルトの転送動作を適用します。 • トラフィック クラスとビジネス関連性に指定した一致属性が、メタデータ ストリームに定義したものと一致しない場合、必要に応じて修正してください。 • バインディング テーブル エントリがアクティブでなくなっている場合、これはエントリの経過時間を意味します。エントリの経過時間を表示するには、**show avc dns-as client binding-table** コマンドを使用します。

DNS-AS を使用した AVC の機能履歴および情報

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

リリース	変更内容
Cisco IOS リリース 15.2(5)E1	この機能が導入されました。 このリリース以降、この機能は Cisco Catalyst 3560-CX シリーズ スイッチでのみサポートされ、Cisco Catalyst 2960-CX シリーズ スイッチではサポートされません。
Cisco IOS リリース 15.2(5)E2	DNS-AS を使用した AVC 向けに Flexible Netflow（FnF）が導入され、FNF を使用してアプリケーション情報をエクスポートできるようになりました。



第 81 章

SDM テンプレートの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

SDM テンプレートの設定に関する情報

SDM テンプレートの制約事項

次に、SDM テンプレートを使用している場合の制約事項を示します。

SDM テンプレート

Switch Database Management（SDM）テンプレートを使用してシステム リソースを設定し、特定の機能に対するサポートをネットワーク内でのデバイスの使用方法に応じて最適化することができます。

Ternary CAM（TCAM）リソースをさまざまな用途に割り当てるために、スイッチ SDM テンプレートはシステム リソースにプライオリティを設定して、特定の機能のサポートを最適化します。デバイスでサポートされているテンプレートは次のとおりです。

- デフォルト：デフォルト テンプレートは、すべての機能に均等にリソースを割り当てます。

（注）

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

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

- SDM テンプレートは VLAN を作成しません。SDM テンプレートにコマンドを追加する前に、VLAN を作成する必要があります。

Catalyst 2960-CX のデフォルト テンプレート

Catalyst 2960-CX スイッチのテンプレートには LAN Base ライセンスが適用されます。

表 *163 :* テンプレートで許容される機能リソースの概算

リソース	デフォルト
ユニキャスト MAC アドレス	16 K
アクティブ VLAN/VLAN ID	255/4096
NetFlow エントリ	16 K
スタックあたりの EtherChannel グループ数	6
IPv4 IGMP または IPv6 グループ	1K IPv4 1K IPv6
直接ルート	2K IPv4 2K IPv6

リソース	デフォルト
間接ルート	1K IPv4 1K IPv6 （16 スタティック ルートのみ）
IPv4 または IPv6 ポリシーベース ルーティング ACE	0（IPv4 PBR） 0（IPv6 PBR）
IPv4 または IPv6 MAC QoS ACE	0.375K（IPv4 QoS） 0.25K（IPv6 QoS）
IPv4 または IPv6 ポートあるいは MAC Security ACE	0.375K（IPv4 ACL） 0.375K（IPv6 ACL）

Catalyst 3560-CX のデフォルト テンプレート

Catalyst 3560-CX スイッチのテンプレートには IP Base および IP Services のライセンスが適用されます。

表 *164* : テンプレートで許容される機能リソースの概算

リソース	デフォルト
ユニキャスト MAC アドレス	16 K
アクティブ VLAN/VLAN ID	1K/4096
スタックあたりの EtherChannel グループ数	6
IPv4 IGMP または IPv6 グループ	1K IPv4 1K IPv6
直接ルート	4K IPv4 4K IPv6
間接ルート	1K IPv4 1K IPv6
IPv4 または IPv6 ポリシーベース ルーティング ACE	0.25K（IPv4 PBR） 0.25K（IPv6 PBR）
IPv4 または IPv6 QoS ACE	0.375K（IPv4 QoS） 0.25K（IPv6 QoS）

リソース	デフォルト
IPv4 または IPv6 ポートあるいは MAC Security ACE	0.375K（IPv4 ACL） 0.375K（IPv6 ACL）

SDM テンプレートの設定方法

SDM テンプレートの設定

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

手順の概要

1. **enable**
2. **configure terminal**
3. **sdm prefer** { **advanced** | **vlan** }
4. **sdm prefer** { **default** }
5. **end**
6. **reload**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**sdm prefer** { **advanced**	**vlan** } 例： `スイッチ(config)# sdm prefer advanced`	スイッチで使用する SDM テンプレートを指定します。キーワードの意味は次のとおりです。 • **advanced**：NetFlow などの高度な機能をサポートします。 • **vlan**：ハードウェアでのルーティングをサポートしないスイッチでの VLAN 設定を最適化します。

	コマンドまたはアクション	目的
		（注） **no sdm prefer** コマンドとデフォルトテンプレートはサポートされません。
ステップ 4	**sdm prefer** { **default** } 例： スイッチ(config)# **sdm prefer lanbase-routing**	スイッチで使用する SDM テンプレートを指定します。キーワードの意味は次のとおりです。 • **default**デフォルト テンプレートでレイヤ 2、IPv4、および IPv6 の機能をすべて均衡化します。 スイッチをデフォルトテンプレートに設定するには、**no sdm prefer** コマンドを使用します。デフォルトテンプレートはシステムリソースを均等に割り当てます。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**reload** 例： スイッチ# **reload**	オペレーティング システムをリロードします。

SDM テンプレートの設定例

例：SDM テンプレートの表示

次に、デフォルトのテンプレート情報を表示した出力例を示します。

これは、Catalyst 3560-CX スイッチのデフォルト テンプレート情報を表示した出力例です。

```
Device# show sdm prefer

The current template is "default" template.
The selected template optimizes the resources in
the switch to support this level of features for
8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  16K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    5K
  number of directly-connected IPv4 hosts:          4K
  number of indirect IPv4 routes:                   1K
```

```
  number of IPv6 multicast groups:                  1K
  number of IPv6 unicast routes:                    5K
  number of directly-connected IPv6 addresses:      4K
  number of indirect IPv6 unicast routes:           1K
  number of IPv4 policy based routing aces:         0.25K
  number of IPv4/MAC qos aces:                      0.375k
  number of IPv4/MAC security aces:                 0.375k
  number of IPv6 policy based routing aces:         0.25K
  number of IPv6 qos aces:                          0.25K
  number of IPv6 security aces:                     0.375k
```

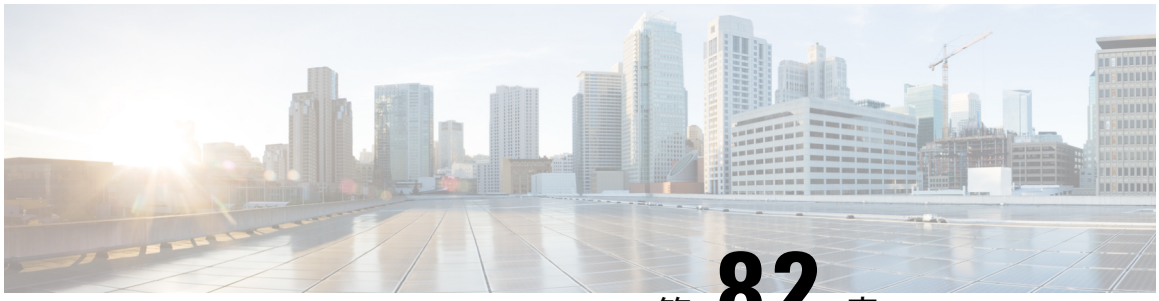
これは、Catalyst 2960-CX スイッチのデフォルト テンプレート情報を表示した出力例です。

```
Device# show sdm prefer

The current template is "default" template.
 The selected template optimizes the resources in
 the switch to support this level of features for
 0 routed interfaces and 255 VLANs.

  number of unicast mac addresses:                  16K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    3K
  number of directly-connected IPv4 hosts:          2K
  number of indirect IPv4 routes:                   1K
  number of IPv6 multicast groups:                  1K
  number of IPv6 unicast routes:                    3K
  number of directly-connected IPv6 addresses:      2K
  number of indirect IPv6 unicast routes:           1K
  number of IPv4 policy based routing aces:         0
  number of IPv4/MAC qos aces:                      0.375k
  number of IPv4/MAC security aces:                 0.375k
  number of IPv6 policy based routing aces:         0
  number of IPv6 qos aces:                          0.25K
  number of IPv6 security aces:                     0.375k
```

例：SDM テンプレートの設定

次に、VLAN テンプレートの設定方法の例を示します。

```
スイッチ(config)# sdm prefer lanbase-routing
スイッチ(config)# exit
スイッチ# reload
  Proceed with reload? [confirm]
```



第 82 章

システム メッセージ ログの設定

システム メッセージ ログの設定に関する制約事項

logging discriminator コマンドを設定すると、デバイスにメモリリークまたはクラッシュが発生する可能性があります。通常これは、大量の syslog またはデバッグが出力されているときに発生します。メモリ リークのレートは、生成されるログの数によって異なります。極端なケースでは、デバイスがクラッシュすることもあります。回避するには、**no logging discriminator** コマンドを使用して、ロギングディスクリミネータを無効にします。

システム メッセージ ログの設定に関する情報

システム メッセージ ロギング

スイッチはデフォルトで、システム メッセージおよび **debug** 特権 EXEC コマンドの出力をロギング プロセスに送信します。。ロギング プロセスはログ メッセージを各宛先（設定に応じて、ログ バッファ、端末回線、UNIX Syslog サーバなど）に配信する処理を制御します。ロギング プロセスは、コンソールにもメッセージを送信します。

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

メッセージの重大度を設定して、コンソールおよび各宛先に表示されるメッセージのタイプを制御できます。ログ メッセージにタイム スタンプを設定したり、Syslog 送信元アドレスを設

定したりして、リアルタイムのデバッグ機能および管理機能を強化できます。表示されるメッセージについては、このリリースに対応するシステム メッセージ ガイドを参照してください。

ロギングされたシステム メッセージにアクセスするには、スイッチのコマンドライン インターフェイス（CLI）を使用するか、または適切に設定された Syslog サーバにこれらのシステム メッセージを保存します。スイッチ ソフトウェアは、Syslog メッセージをスタンドアロン スイッチ上の内部バッファに保存します。スタンドアロン スイッチに障害が発生すると、ログをフラッシュ メモリに保存していなかった場合、ログは失われます。

システム メッセージをリモートで監視するには、Syslog サーバー上でログを表示するか、あるいは Telnet、コンソール ポート、またはイーサネット管理ポート経由でスイッチにアクセスします。

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

システム ログ メッセージのフォーマット

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

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

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

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

表 *165 :* システム ログ メッセージの要素

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

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

デフォルトのシステム メッセージ ロギングの設定

表 *166* : デフォルトのシステム メッセージ ロギングの設定

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

Syslog トラップ メッセージの有効化

Syslog トラップは、**snmp-server enable traps syslog** コマンドを使用してイネーブルにすることができます。

Syslog トラップをイネーブルにしたら、トラップ メッセージ重大度を指定する必要があります。**logging snmp-trap** コマンドを使用して、トラップレベルを指定します。デフォルトでは、このコマンドは重大度 0 から 4 をイネーブルにします。すべての重大度レベルをイネーブルにするには、**logging snmp-trap 0 7** コマンドを設定します。

個々のトラップ レベルをイネーブルにするには、次のコマンドを設定します。

- **logging snmp-trap emergencies**：重大度 0 のトラップのみをイネーブルにします。
- **logging snmp-trap alert** 重大度 1 のトラップのみをイネーブルにします。

Syslog トラップと一緒に、Syslog 履歴にも適用されることに注意してください。これが設定されていないと、Syslog トラップは送信されません。

logging history informational コマンドを使用して、Syslog 履歴をイネーブルにします。

システム メッセージ ログの設定方法

メッセージ表示宛先デバイスの設定

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

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **logging buffered** [*size*]
3. **logging** *host*
4. **logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*]
5. **end**
6. **terminal monitor**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**logging buffered** [*size*] 例： `スイッチ(config)# logging buffered 8192`	スイッチ上、またはスタンドアロンスイッチ上、あるいはスイッチスタックの場合はアクティブスイッチ上で、ログメッセージを内部バッファに保存します。指定できる範囲は 4096 ～ 2147483647 バイトです。デフォルトのバッファ サイズは 4096 バイトです。 スタンドアロンスイッチまたはアクティブスイッチに障害が発生すると、ログファイルをフラッシュメモリに保存していなかった場合、ログファイルは失われます。ステップ 4 を参照してください。 （注） バッファ サイズを大きすぎる値に設定しないでください。他の作業に使用するメモリが不足することがあります。スイッチ上の空きプロセッサメモリを表示するには、**show memory** 特権 EXEC コマンドを使用します。ただし、表示される値は使用できる最大値であるため、バッファ サイズをこの値に設定しないでください。
ステップ 3	**logging** *host* 例： `スイッチ(config)# logging 125.1.1.100`	UNIX Syslog サーバ ホストにメッセージを保存します。 *host* には、syslog サーバとして使用するホストの名前または IP アドレスを指定します。 ログ メッセージを受信する Syslog サーバのリストを作成するには、このコマンドを複数回入力します。
ステップ 4	**logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*] 例： `スイッチ(config)# logging file flash:log_msg.txt 40960 4096 3`	スタンドアロンスイッチ上か、または、スイッチスタックの場合はアクティブスイッチ上で、フラッシュメモリにあるファイルにログメッセージを保存します。 • *filename*：ログ メッセージのファイル名を入力します。 • （任意）**max-file-size** — には、ログファイルの最大サイズを指定します。指定できる範囲は 4096 ～ 2147483647 です。デフォルトは 4096 バイトです。 • （任意）*min-file-size*：ログ ファイルの最小サイズを指定します。指定できる範囲は 1024 ～

	コマンドまたはアクション	目的
		2147483647 です。デフォルトは 2048 バイトです。 • （任意）*severity-level-number*	*type*：ロギングの重大度またはロギング タイプを指定します。重大度に指定できる範囲は 0 ～ 7 です。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**terminal monitor** 例： スイッチ# **terminal monitor**	現在のセッション間、非コンソール端末にメッセージを保存します。 端末パラメータ コンフィギュレーション コマンドはローカルに設定され、セッションの終了後は無効になります。デバッグ メッセージを表示する場合は、セッションごとにこのステップを実行する必要があります。

ログ メッセージの同期化

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

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

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **line** [**console** | **vty**] *line-number* [*ending-line-number*]
3. **logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*]
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**line** [**console** | **vty**] *line-number* [*ending-line-number*] 例： スイッチ(config)# **line console**	メッセージの同期ロギングに設定する回線を指定します。 • **console**：スイッチ コンソール ポートまたはイーサネット管理ポートでの設定を指定します。 • **line vty** *line-number*：どの vty 回線の同期ロギングをイネーブルにするかを指定します。Telnet セッションを介して行われる設定には、vty 接続を使用します。回線番号に指定できる範囲は 0 ～ 15 です。 16 個の vty 回線の設定をすべて一度に変更するには、次のように入力します。 line vty 0 15 また、現在の接続に使用されている 1 つの vty 回線の設定を変更することもできます。たとえば、vty 回線 2 の設定を変更するには、次のように入力します。 line vty 2 このコマンドを入力すると、ライン コンフィギュレーション モードになります。
ステップ 3	**logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*] 例： スイッチ(config)# **logging synchronous level 3 limit 1000**	メッセージの同期ロギングをイネーブルにします。 • （任意）**level** *severity-level*：メッセージの重大度レベルを指定します。重大度がこの値以上であるメッセージは、非同期に出力されます。値が小さいほど重大度は大きく、値が大きいほど重大度は小さくなります。デフォルトは 2 です。 • （任意）**level all**：重大度に関係なく、すべてのメッセージが非同期に出力されます。 • （任意）**limit** *number-of-buffers*：キューイングされる端末のバッファ数を指定します。これを超える新しいメッセージは廃棄されます。指定

	コマンドまたはアクション	目的
		できる範囲は 0 ～ 2147483647 です。デフォルトは 20 です。
ステップ 4	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

メッセージ ロギングのディセーブル化

メッセージロギングはデフォルトでイネーブルに設定されています。コンソール以外のいずれかの宛先にメッセージを送信する場合は、メッセージロギングをイネーブルにする必要があります。メッセージ ロギングがイネーブルの場合、ログ メッセージはロギング プロセスに送信されます。ロギングプロセスは、メッセージを生成元プロセスと同期しないで指定場所に記録します。

ロギングプロセスをディセーブルにすると、メッセージがコンソールに書き込まれるまでプロセスは処理続行を待機する必要があるため、スイッチの処理速度が低下することがあります。ロギングプロセスがディセーブルの場合、メッセージは生成後すぐに（通常はコマンド出力に割り込む形で）コンソールに表示されます。

logging synchronous グローバル コンフィギュレーション コマンドも、コンソールへのメッセージ表示に影響します。このコマンドをイネーブルにすると、Return を押さなければメッセージが表示されません。

メッセージ ロギングをディセーブルにした後に再びイネーブルにするには、**logging on** グローバル コンフィギュレーション コマンドを使用します。

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **no logging console**
3. **end**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**no logging console** 例： `スイッチ(config)# no logging console`	メッセージ ロギングをディセーブルにします。
ステップ 3	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。

ログ メッセージのタイム スタンプのイネーブル化およびディセーブル化

デフォルトでは、ログ メッセージにはタイム スタンプが適用されません。

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. 次のいずれかのコマンドを使用します。
   - **service timestamps log uptime**
   - **service timestamps log datetime**[**msec** | **localtime** | **show-timezone**]
3. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	次のいずれかのコマンドを使用します。 • **service timestamps log uptime** • **service timestamps log datetime**[**msec** | **localtime** | **show-timezone**] 例：	ログのタイムスタンプをイネーブルにします。 • **log uptime**：ログ メッセージのタイムスタンプをイネーブルにして、システムの再起動以降の経過時間を表示します。 • **log datetime**：ログ メッセージのタイムスタンプをイネーブルにします。選択したオプション

	コマンドまたはアクション	目的
	スイッチ(config)# **service timestamps log uptime** または スイッチ(config)# **service timestamps log datetime**	に応じて、ローカル タイム ゾーンを基準とした日付、時間（ミリ秒）、タイム ゾーン名をタイム スタンプとして表示できます。
ステップ 3	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

ログメッセージのシーケンス番号のイネーブル化およびディセーブル化

タイムスタンプが同じログ メッセージが複数ある場合、これらのメッセージを表示するには、シーケンス番号を使用してメッセージを表示できます。デフォルトでは、ログ メッセージにシーケンス番号は表示されません。

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **service sequence-numbers**
3. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**service sequence-numbers** 例： スイッチ(config)# **service sequence-numbers**	シーケンス番号をイネーブルにします。
ステップ 3	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	

メッセージ重大度の定義

メッセージの重大度を指定して、選択したデバイスに表示されるメッセージを制限します。

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **logging console** *level*
3. **logging monitor** *level*
4. **logging trap** *level*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**logging console** *level* 例： スイッチ(config)# **logging console 3**	コンソールに保存するメッセージを制限します。 デフォルトで、コンソールはデバッグ メッセージ、および数値的により低いレベルのメッセージを受信します。
ステップ 3	**logging monitor** *level* 例： スイッチ(config)# **logging monitor 3**	端末回線に出力するメッセージを制限します。 デフォルトで、端末はデバッグ メッセージ、および数値的により低いレベルのメッセージを受信します。
ステップ 4	**logging trap** *level* 例： スイッチ(config)# **logging trap 3**	Syslog サーバに保存するメッセージを制限します。 デフォルトで、Syslog サーバは通知メッセージ、および数値的により低いレベルのメッセージを受信します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。

履歴テーブルおよび SNMP に送信される syslog メッセージの制限

このタスクでは、履歴テーブルおよび SNMP に送信される syslog メッセージを制限する方法について説明します。

このタスクはオプションです。

手順の概要

1. **configure terminal**
2. **logging history** *level*
3. **logging history size** *number*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**logging history** *level* 例： スイッチ(config)# **logging history 3**	履歴ファイルに保存され、SNMP サーバに送信される syslog メッセージのデフォルト レベルを変更します。 デフォルトでは **warnings**、**errors**、**critical**、**alerts**、および **emergencies** メッセージは送信されません。
ステップ 3	**logging history size** *number* 例： スイッチ(config)# **logging history size 200**	履歴テーブルに保存できる Syslog メッセージの数を指定します。 デフォルトでは 1 つのメッセージが格納されます。指定できる範囲は 0 ～ 500 です。
ステップ 4	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	

UNIX Syslog デーモンへのメッセージのロギング

このタスクはオプションです。

（注） 最新バージョンの UNIX Syslog デーモンの中には、デフォルトでネットワークからの Syslog パケットを受け入れないものがあります。このようなシステムの場合に、Syslog メッセージのリモートロギングをイネーブルにするには、Syslog コマンドラインに追加または削除する必要があるオプションを、UNIX の **man syslogd** コマンドを使用して判別します。

始める前に

- root としてログインします。
- システム ログ メッセージを UNIX Syslog サーバに送信する前に、UNIX サーバ上で Syslog デーモンを設定する必要があります。

手順の概要

1. /etc/syslog.conf ファイルに次の行を追加します。
2. UNIX シェル プロンプトに次のコマンドを入力します。
3. Syslog デーモンに新しい設定を認識させます。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	/etc/syslog.conf ファイルに次の行を追加します。 **例**： **local7.debug /usr/adm/logs/***cisco.log*	• **local7**：ロギング機能を指定します。 • **debug**：syslog レベルを指定します。このファイルは、syslog デーモンに書き込み権限がある既存ファイルである必要があります。
ステップ 2	UNIX シェル プロンプトに次のコマンドを入力します。 **例**： $ **touch /var/log/***cisco.log* $ **chmod 666 /var/log/***cisco.log*	ログ ファイルを作成します。syslog デーモンは、このレベルまたはこのファイルのより高い重大度レベルでメッセージを送信します。

	コマンドまたはアクション	目的
ステップ 3	Syslog デーモンに新しい設定を認識させます。 例： `$ kill -HUP `cat /etc/syslog.pid``	詳細については、ご使用の UNIX システムの **man syslog.conf** および **man syslogd** コマンドを参照してください。

システム メッセージ ログのモニタリングおよびメンテナンス

コンフィギュレーション アーカイブ ログのモニタリング

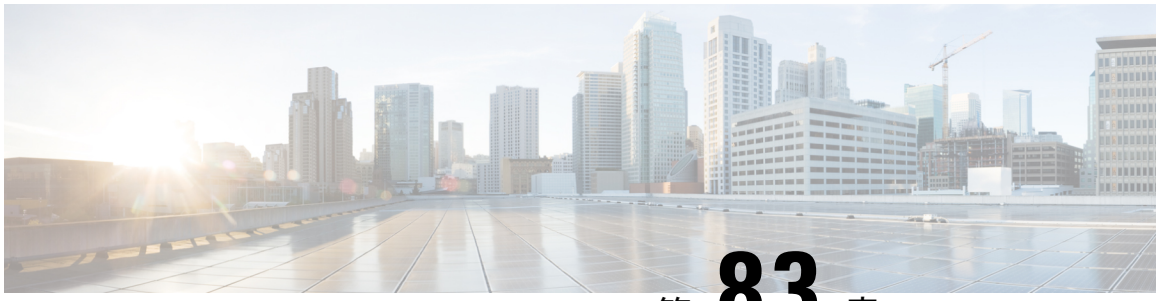
コマンド	目的
show archive log config {**all** | **number** [*end-number*] | **user** *username* [**session** *number*] *number* [*end-number*] | **statistics**} [**provisioning**]	コンフィギュレーション ログ全体、または指定されたパラメータのログを表示します。

システム メッセージ ログの設定例

例：スイッチ システム メッセージ

次に、スイッチ上のスイッチ システム メッセージの一部を示します。

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```



第 83 章

オンライン診断の設定

オンライン診断の設定に関する情報

オンライン診断

オンライン診断では、デバイスが稼働中のネットワークに接続している間に、デバイスのハードウェア機能をテストし、確認できます。

オンライン診断には、異なるハードウェア コンポーネントをチェックするパケット交換テストが含まれ、データ パスおよび制御信号が確認されます。

オンライン診断では、次の領域の問題が検出されます。

- ハードウェア コンポーネント
- インターフェイス（イーサネット ポートなど）
- はんだ接合

オンライン診断は、オンデマンド診断、スケジュール診断、ヘルスモニタリング診断に分類できます。オンデマンド診断は、CLI から実行されます。スケジュールされた診断は、動作中のネットワークに デバイス が接続されているときに、ユーザーが指定した間隔または指定した時刻に実行されます。ヘルスモニターリングは、バックグラウンドでユーザーが指定した間隔で実行されます。デフォルトでは、30 秒ごとにヘルス モニタリング テストが実行されます。

オンライン診断を設定したあと、手動で診断テストを開始したり、テスト結果を表示したりできます。また、デバイスまたはスイッチスタックに設定されているテストの種類、およびすでに実行された診断テスト名を確認できます。

オンライン診断の設定方法

オンライン診断テストの開始

スイッチで実行する診断テストを設定したあと、**diagnostic start** 特権 EXEC コマンドを使用して診断テストを開始します。

テストを開始したら、テスト プロセスの停止はできません。

手動でオンライン診断テストを開始するには、次の特権 EXEC コマンドを使用します。

手順の概要

1. **diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** }

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** } 例： `スイッチ# diagnostic start switch 2 test basic`	診断テストを開始します。 **switch** *number* キーワードは、スタック構成スイッチだけでサポートされます。有効な範囲は 1 ～ 8 です。 次のいずれかのオプションを使用してテストを指定できます。 • *name*：テストの名前を入力します。 • *test-id*：テストの ID 番号を入力します。 • *test-id-range*：カンマとハイフンで区切ってテスト ID の範囲を整数で入力します。 • **all**：すべてのテストを開始します。 • **basic**：基本テスト スイートを開始します。 • **non-disruptive**：ノンディスラプティブ テスト スイートを開始します。

オンライン診断の設定

診断モニタリングをイネーブルにする前に、障害しきい値およびテストの間隔を設定する必要があります。

オンライン診断のスケジューリング

特定のスイッチについて指定した時間、または日、週、月単位でオンライン診断をスケジューリングできます。スケジューリングを削除するには、コマンドの **no** 形式を入力します。

手順の概要

1. **configure terminal**
2. **diagnostic schedule switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** |} {**daily** | **on** *mm dd yyyy hh:mm* | **weekly** *day-of-week hh:mm*}

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**diagnostic schedule switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** |} {**daily** | **on** *mm dd yyyy hh:mm* | **weekly** *day-of-week hh:mm*} 例： `スイッチ(config)# diagnostic schedule switch 1 test 1-5 on July 3 2013 23:10`	特定日時のオンデマンド診断テストをスケジュールします。 **switch** *number* キーワードは、スタック構成スイッチだけでサポートされます。有効な範囲は 1 ～ 8 です。 スケジュールするテストを指定する場合は、次のオプションを使用します。 • *name*：**show diagnostic content** コマンドの出力に表示されるテストの名前です。 • *test-id*：**show diagnostic content** コマンドの出力に表示されるテストの ID 番号です。 • *test-id-range*：**show diagnostic content** コマンドの出力に表示されるテストの ID 番号です。 • **all**：すべてのテスト ID • **basic**：基本的なオンデマンドの診断テストを開始します。 • **non-disruptive**：ノンディスラプティブ テスト スイートを開始します。 テストは次のようにスケジュールできます。 • 毎日：**daily** *hh:mm* パラメータを使用します。

	コマンドまたはアクション	目的
		• 特定日時：**on** *mm dd yyyy hh:mm* パラメータを使用します。 • 毎週：**weekly** *day-of-week hh:mm* パラメータを使用します。

ヘルス モニタリング診断の設定

デバイスが稼働中のネットワークに接続されている間に、スイッチに対しヘルスモニタリング診断テストを設定できます。各ヘルスモニタリングテストの実行間隔を設定したり、デバイスをイネーブルにし、テスト失敗時のsyslogメッセージを生成したり、特定のテストをイネーブルにできます。

テストをディセーブルにするには、コマンドの **no** 形式を入力します。

デフォルトでは、ヘルスモニターリングはディセーブルですが、デバイスはテストの失敗時にSyslog メッセージを生成します。

ヘルスモニターリング診断テストを設定し、イネーブルにするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **diagnostic monitor interval switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day*
4. **diagnostic monitor syslog**
5. **diagnostic monitor threshold switch** *number number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count*
6. **diagnostic monitor switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**diagnostic monitor interval switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day* 例： スイッチ(config)# **diagnostic monitor interval switch 2 test 1 12:30:00 750 5**	指定のテストに対し、ヘルスモニターリングの実行間隔を設定します。 テストを指定する場合は、次のいずれかのパラメータを使用します。 • *name*：**show diagnostic content** コマンドの出力に表示されるテストの名前です。 • *test-id*：**show diagnostic content** コマンドの出力に表示されるテストの ID 番号です。 • *test-id-range*：**show diagnostic content** コマンドの出力に表示されるテストの ID 番号です。 • **all**：すべての診断テスト。 間隔を指定する場合は、次のパラメータを設定します。 • *hh:mm:ss*：モニタリング間隔（時間、分、秒）。指定できる範囲は *hh* が 0 ~ 24、*mm* および *ss* が 0 ~ 60 です。 • *milliseconds*：モニタリング間隔（ミリ秒（ms））。指定できる範囲は 0 ～ 999 です。 • *day*：モニタリング間隔（日数）。指定できる範囲は 0 ～ 20 です。
ステップ 4	**diagnostic monitor syslog** 例： スイッチ(config)# **diagnostic monitor syslog**	（任意）ヘルス モニタリング テストの失敗時にスイッチが Syslog メッセージを生成するように設定します。
ステップ 5	**diagnostic monitor threshold switch** *number number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count* 例： スイッチ(config)# **diagnostic monitor threshold switch 2 test 1 failure count 20**	（任意）ヘルス モニターリング テストの失敗しきい値を設定します。 **switch** *number* キーワードは、スタック構成スイッチだけでサポートされます。有効な範囲は 1 ～ 8 です。 テストを指定する場合は、次のいずれかのパラメータを使用します。

	コマンドまたはアクション	目的
		• *name*：**show diagnostic content** コマンドの出力に表示されるテストの名前です。 • *test-id*：**show diagnostic content** コマンドの出力に表示されるテストの ID 番号です。 • *test-id-range*：**show diagnostic content** コマンドの出力に表示されるテストの ID 番号です。 • **all**：すべての診断テスト。 失敗しきい値 *count* に指定できる範囲は 0 ～ 99 です。
ステップ 6	**diagnostic monitor switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} 例： `スイッチ(config)# diagnostic monitor switch 2 test 1`	指定のヘルス モニタリング テストをイネーブルにします。 **switch** *number* キーワードは、スタック構成スイッチだけでサポートされます。有効な範囲は 1 ～ 8 です。 テストを指定する場合は、次のいずれかのパラメータを使用します。 • *name*：**show diagnostic content** コマンドの出力に表示されるテストの名前です。 • *test-id*：**show diagnostic content** コマンドの出力に表示されるテストの ID 番号です。 • *test-id-range*：**show diagnostic content** コマンドの出力に表示されるテストの ID 番号です。 • **all**：すべての診断テスト。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	スイッチ# **copy running-config startup-config**	

次のタスク

no diagnostic monitor interval test*test-id* | *test-id-range* } グローバル コンフィギュレーション コマンドを使用して、間隔をデフォルトの値またはゼロに変更します。**no diagnostic monitor syslog** コマンドを使用し、ヘルスモニタリングテストが失敗した場合の Syslog メッセージの生成をディセーブルにします。**diagnostic monitor threshold test***test-id* | *test-id-range* }**failure count** コマンドを使用し、失敗しきい値を削除します。

オンライン診断のモニタリングおよびメンテナンス

オンライン診断テストとテスト結果の表示

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

表 *167 :* 診断テストの設定および結果用のコマンド

コマンド	目的
show diagnostic content switch [*number* | **all**]	スイッチに対して設定されたオンライン診断を表示します。
show diagnostic status	現在実行中の診断テストを表示します。
show diagnostic result switch [*number* | **all**] [**detail** | **test** {*name* | *test-id* | *test-id-range* | **all**} [**detail**]]	オンライン診断テストの結果を表示します。
show diagnostic switch [*number* | **all**] [**detail**]	オンライン診断テストの結果を表示します。
show diagnostic schedule switch [*number* | **all**]	オンライン診断テストのスケジュールを表示します。
show diagnostic post	POST 結果を表示します（出力は **show post** コマンドの出力と同じ）。

オンライン診断テストの設定例

オンライン診断テストの開始

スイッチで実行する診断テストを設定したあと、**diagnostic start** 特権 EXEC コマンドを使用して診断テストを開始します。

テストを開始したら、テスト プロセスの停止はできません。

手動でオンライン診断テストを開始するには、次の特権 EXEC コマンドを使用します。

手順の概要

1. **diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** }

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** } 例： `スイッチ# diagnostic start switch 2 test basic`	診断テストを開始します。 **switch** *number* キーワードは、スタック構成スイッチだけでサポートされます。有効な範囲は 1 ～ 8 です。 次のいずれかのオプションを使用してテストを指定できます。 • *name*：テストの名前を入力します。 • *test-id*：テストの ID 番号を入力します。 • *test-id-range*：カンマとハイフンで区切ってテスト ID の範囲を整数で入力します。 • **all**：すべてのテストを開始します。 • **basic**：基本テスト スイートを開始します。 • **non-disruptive**：ノンディスラプティブ テスト スイートを開始します。

例：ヘルス モニタリング テストの設定

次に、ヘルス モニタリング テストを設定する例を示します。

```
スイッチ(config)# diagnostic monitor threshold switch 1 test 1 failure count 50
```

```
スイッチ(config)# diagnostic monitor interval switch 1 test TestPortAsicStackPortLoopback
```

例：診断テストのスケジューリング

次に、特定のスイッチに対して、特定の日時に診断テストを実行するようにスケジューリングする例を示します。

```
スイッチ(config)# diagnostic schedule test DiagThermalTest on June 3 2013  22:25
```

次の例では、指定されたスイッチで毎週特定の時間に診断テストを実行するようにスケジューリングする方法を示します。

```
スイッチ(config)# diagnostic schedule switch 1 test 1,2,4-6 weekly saturday 10:30
```

オンライン診断の表示：例

次の例では、特定のスイッチのオンライン診断の詳細情報を表示する方法を示します。

```
スイッチ# show diagnostic switch 1 detail

Switch 1: SerialNo :

Overall Diagnostic Result for Switch 1 : UNTESTED

Test results: (. = Pass, F = Fail, U = Untested)

_____________________________________________________________________________

1) TestPortAsicStackPortLoopback ---> U

Error code ------------------> 3 (DIAG_SKIPPED)
Total run count -------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ---------> n/a
Total failure count ---------> 0
Consecutive failure count ---> 0
_____________________________________________________________________________

2) TestPortAsicLoopback ------------> U

Error code ------------------> 3 (DIAG_SKIPPED)
Total run count -------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ---------> n/a
Total failure count ---------> 0
Consecutive failure count ---> 0
_____________________________________________________________________________
```

```
3) TestPortAsicCam ------------------> U

Error code --------------------> 3 (DIAG_SKIPPED)
Total run count --------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ----------> n/a
Total failure count ----------> 0
Consecutive failure count ---> 0
_____________________________________________________________________________

4) TestPortAsicMem ------------------> U

Error code --------------------> 3 (DIAG_SKIPPED)
Total run count --------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ----------> n/a
Total failure count ----------> 0
Consecutive failure count ---> 0
_____________________________________________________________________________

5) TestInlinePwrCtlr ----------------> U

Error code --------------------> 3 (DIAG_SKIPPED)
Total run count --------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ----------> n/a
Total failure count ----------> 0
Consecutive failure count ---> 0
_____________________________________________________________________________
```

次の例では、特定のスイッチに設定されているオンライン診断を表示する方法を示します。

```
スイッチ# show diagnostic content switch 3

Switch 1:
Diagnostics test suite attributes:
    B/* - Basic ondemand test / NA
  P/V/* - Per port test / Per device test / NA
  D/N/* - Disruptive test / Non-disruptive test / NA
    S/* - Only applicable to standby unit / NA
    X/* - Not a health monitoring test / NA
    F/* - Fixed monitoring interval test / NA
    E/* - Always enabled monitoring test / NA
    A/I - Monitoring is active / Monitoring is inactive
    R/* - Switch will reload after test list completion / NA
    P/* - will partition stack / NA

                                                      Test Interval   Thre-
ID   Test Name                           Attributes    day hh:mm:ss.ms shold
==== ================================== ============  =============== =====
```

```
  1) TestPortAsicStackPortLoopback ---> B*N****I**      not configured  n/a
  2) TestPortAsicLoopback ------------> B*D*X**IR*      not configured  n/a
  3) TestPortAsicCam -----------------> B*D*X**IR*      not configured  n/a
  4) TestPortAsicRingLoopback --------> B*D*X**IR*      not configured  n/a
  5) TestMicRingLoopback -------------> B*D*X**IR*      not configured  n/a
  6) TestPortAsicMem -----------------> B*D*X**IR*      not configured  n/a
```

次の例では、スイッチのオンライン診断結果を表示する方法を示します。

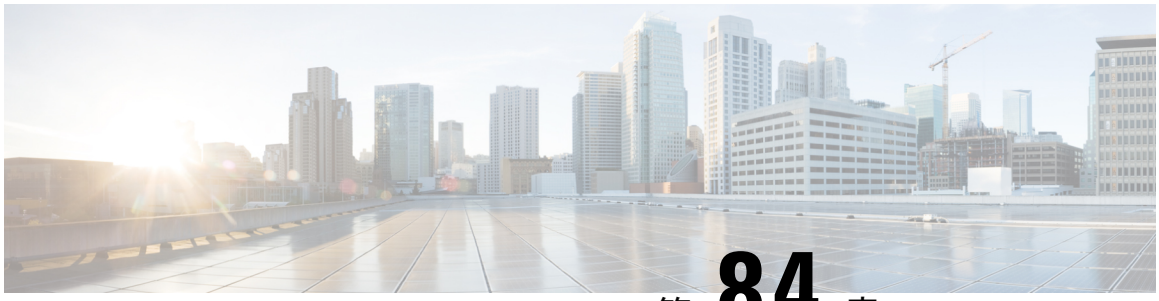
```
スイッチ# show diagnostic result

Switch 1: SerialNo :
Overall diagnostic result: PASS
Test results: (. = Pass, F = Fail, U = Untested)
1) TestPortAsicStackPortLoopback ---> .
2) TestPortAsicLoopback ------------> .
3) TestPortAsicCam -----------------> .
4) TestPortAsicRingLoopback --------> .
5) TestMicRingLoopback -------------> .
6) TestPortAsicMem -----------------> .
```

次の例では、オンライン診断テストのステータスを表示する方法を示します。

```
スイッチ# show diagnostic status

<BU> - Bootup Diagnostics, <HM> - Health Monitoring Diagnostics,
<OD> - OnDemand Diagnostics, <SCH> - Scheduled Diagnostics
====== ================================= ================================ ======
Card   Description                       Current Running Test             Run by
------ --------------------------------- -------------------              ------
1                                         N/A                               N/A
2                                         TestPortAsicStackPortLoopback     <OD>
                                          TestPortAsicLoopback              <OD>
                                          TestPortAsicCam                   <OD>
                                          TestPortAsicRingLoopback          <OD>
                                          TestMicRingLoopback               <OD>
                                          TestPortAsicMem                   <OD>
3                                         N/A                               N/A
4                                         N/A                               N/A
====== ================================= ================================ ======
Switch#
```

次の例では、スイッチのオンライン診断のテスト スケジュールを表示する方法を示します。

```
スイッチ# show diagnostic schedule switch 1

Current Time = 14:39:49 PST Tue May 5 2013
Diagnostic for Switch 1:
Schedule #1:
To be run daily 12:00
Test ID(s) to be executed: 1.
```



第 84 章

データのサニタイズ

このモジュールは、デバイスからデータをサニタイズする方法に関する情報を提供します。

データのサニタイズ

単純で非侵襲的なデータリカバリ技術または最先端の実験技術によってデータをリカバリ不能にレンダリングする米国国立標準技術研究所（NIST）のパージメソッドを使用します。

（注） 特に明記されていない限り、データのサニタイズ手順は、単純な非侵襲的なデータ リカバリ技術から保護するために、ユーザーがアドレス指定可能なストレージの場所で NIST 800-88 の明確なサニタイズ技術を提供し、最先端の実験室技術を使用してデータ リカバリを実行不可能にする技術を提供しません。

次の手順に従って、フラッシュ ドライブからファイルを削除します。

ステップ 1 **factory-reset all secure**

例：

```
Device> factory-reset all secure
```

フラッシュ上のデータをパージします。

ステップ 2 TFTP を使用してイメージをフラッシュにコピーします。

詳細については、「TFTP を使用したイメージファイルのコピー」を参照してください。

ステップ 3 **reload**

例：

```
Device> reload
```

デバイスがリロードされます。

（注） イメージをフラッシュドライブにコピーした場合（手順2）、スイッチは自動的に再起動します。

ステップ 4 show platform software factory-reset secure log

例：

```
Device> show platform software factory-reset secure log
```

データのサニタイズレポートを表示します。

例: データのサニタイズ

次の例は、デバイスからすべてのデータをリセットする方法を示しています。

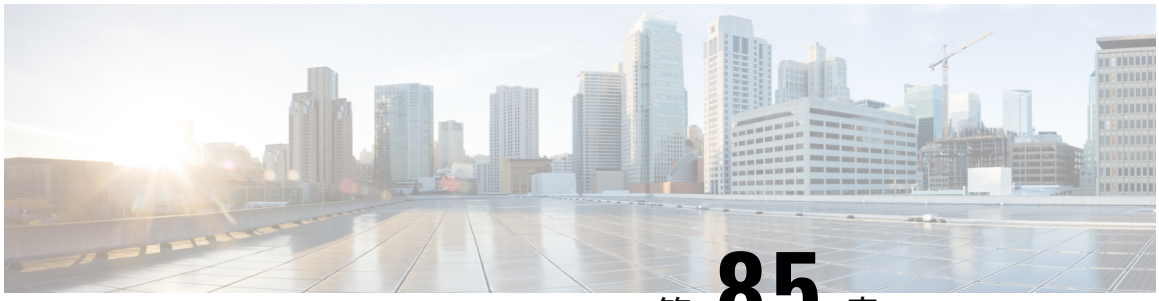
```
Device# factory-reset all secure

The factory reset operation is irreversible for all operations. Are you sure? [confirm]

 The following will be deleted as a part of factory reset: NIST-SP-800-88-R1

 1: Crash info and logs
 2: User data, startup and running configuration
 3: All IOS images, including the current boot image
 4: User added rommon variables
 5: OBFL logs
 6: License usage log files

Note:

 1. You are advised to COPY an IOS image via TFTP after factory-reset and before reloading

    the box (OPTIONAL)
 2. Then, Reload the box for factory-reset to complete

 DO NOT UNPLUG THE POWER OR INTERRUPT THE OPERATION

 Are you sure you want to continue?

[confirm]

% factory-reset: started.
% Format of nvram start..
% Format of nvram end...

*Sep 20 11:36:14.980: %SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvram

% Erase of obfl0 start...

.............................

% Erase of obfl0 end...

% Validating obfl0 partition...

00000000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

.............................................................

003FFFF0: **
```

```
.

% Format of obfl0 start
% Format of obfl0 complete
% Erase of rsvd start...

.......

% Erase of rsvd end...
% Validating rsvd partition...

00000000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

.............

000DFFF0: **

.

% Erase of flash start...

..........................................................................................................................................

% Erase of flash end...

% Validating flash partition...

00000000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

..........................................................................................................................................

0E9FFFF0: **

.

% Format of flash start
% Format of flash complete
% Format of vb: start...
% Format of vb: end...
% act2 erase started...

------ USER 1 ------------------

 ObjectID   ObjectType  ObjectSize

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

0xBA7E1F05   0x01        0x00DC

% act2 erase completed...

#CISCO C1000-48T-4G-L DATA SANITIZATION REPORT#

START : 2022-09-20 11:36:11
END   : 2022-09-20 11:37:28
PNM   : NAND
MNM   : IS34/35ML02G084
MID   : 0x00
DID   : 0xDAC8
NIST : PURGE SUCCESS

% factory-reset: logging success...
% FACTORY-RESET - Secure Successfull...
```

```
1. You are advised to COPY an IOS image via TFTP before reloading the box (OPTIONAL)
2. Then, Reload the box for factory-reset to complete
```

次に、デバイスの安全な工場出荷時のリセット後の show platform software factory-reset secure log コマンドからの出力例を示します。

```
Device# show platform software factory-reset secure log

#CISCO C1000-48T-4G-L DATA SANITIZATION REPORT#
START : 2022-07-13 10:50:29
END   : 2022-07-13 10:51:45
PNM   : NAND
MNM   : IS34/35ML02G084
MID   : 0x00
DID   : 0xDAC8
NIST : PURGE SUCCESS
```



第 85 章

ソフトウェア設定のトラブルシューティング

この章では、スイッチが稼働する Cisco IOS ソフトウェアに関連する問題を特定し、解決する方法について説明します。問題の性質に応じて、コマンドライン インターフェイス（CLI）、デバイス マネージャ、または Network Assistant を使用して、問題を特定し解決できます。

LED の説明など、トラブルシューティングの詳細については、ハードウェア インストレーション ガイドを参照してください。

ソフトウェア設定のトラブルシューティングに関する情報

スイッチのソフトウェア障害

スイッチ ソフトウェアがアップグレード中に破損する原因として、誤ったファイルがスイッチにダウンロードされた場合やイメージ ファイルが削除された場合があります。いずれの場合にも、スイッチは電源投入時自己診断テスト（POST）に失敗し、接続できなくなります。

のパスワードを紛失したか忘れた場合 デバイス

deviceのデフォルト設定では、deviceに物理的にアクセスしているエンドエンド ユーザーは、スイッチの電源投入中に起動プロセスを中断して新しいパスワードを入力することにより、パスワードを紛失した状態から回復できます。ここで紹介する回復手順を実行するには、deviceに物理的にアクセスする必要があります。

（注） これらのdevicesでは、システム管理者は、デフォルト設定に戻すことに同意した場合に限り、エンドユーザーによるパスワードのリセットを許可することによって、この機能の一部を無効化できます。パスワード回復がディセーブルになっている場合に、エンドユーザーがパスワードをリセットしようとすると、ステータス メッセージで回復プロセスの間はデフォルトの設定に戻すように指示されます。

（注） Cisco WLC の設定を複数の Cisco WLC 間でコピーすると、暗号化パスワード キーを回復できなくなります（RMA の場合）。

Power over Ethernet（PoE）ポート

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

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

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

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

電力消失によるポートの障害

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

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

PoE ポート ステータスのモニタリング

- **show controllers power inline** 特権 EXEC コマンド

- **show power inline** EXEC コマンド
- **debug ilpower** 特権 EXEC コマンドを使用します。

不正リンク アップによるポート障害

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

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

ping

デバイス は IP の ping をサポートしており、これを使用してリモートホストへの接続をテストできます。pingはアドレスにエコー要求パケットを送信し、応答を待ちます。pingは次のいずれかの応答を返します。

- 正常な応答：正常な応答（*hostname* が存在する）は、ネットワーク トラフィックにもよりますが、1 ～ 10 秒以内で発生します。
- 宛先の応答なし：ホストが応答しない場合、*no-answer* メッセージが返されます。
- 不明なホスト：ホストが存在しない場合、*unknown host* メッセージが返されます。
- 宛先到達不能：デフォルト ゲートウェイが指定されたネットワークに到達できない場合、*destination-unreachable* メッセージが返されます。
- ネットワークまたはホストへの到達不能：ルートテーブルにホストまたはネットワークのエントリがない場合、*network or host unreachable* メッセージが返されます。

レイヤ 2 トレースルート

レイヤ2 トレースルート機能により、パケットが通過する送信元デバイスから宛先デバイスまでの物理パスを識別できます。レイヤ2 トレースルートは、ユニキャストの送信元および宛先 MAC アドレスだけをサポートします。transroute は、パス内にある デバイス の MAC アドレス テーブルを使用してパスを識別します。デバイス がレイヤ2 パス内でレイヤ2 トレースルートをサポートしていないデバイスを検知した場合、デバイス はレイヤ2 トレースクエリを送信し続け、タイムアウトにします。

デバイス は、送信元デバイスから宛先デバイスへのパスのみを識別できます。パケットが通過する、送信元ホストから送信元デバイスまで、または宛先デバイスから宛先ホストまでのパスは識別できません。

レイヤ 2 の traceroute のガイドライン

- ネットワーク内のすべてのデバイスで、Cisco Discovery Protocol（CDP）をイネーブルにする必要があります。レイヤ 2 traceroute が適切に動作するために、CDP を無効にしないでください。

  物理パス内のデバイスが CDP に対して透過的な場合、スイッチはこれらのデバイスを通過するパスを識別できません。

- **ping** 特権 EXEC コマンドを使用して接続をテストできれば、この デバイス は別の デバイス から到達可能であると定義できます。物理パス内のすべての デバイス は、他のスイッチから相互に到達可能でなければなりません。

- パス内で識別可能な最大ホップ数は 10 です。

- 送信元デバイスと宛先デバイスの間の物理パス内にないデバイス で、**traceroute mac** または **traceroute mac ip** の特権 EXEC コマンドを実行できます。パス内のすべての デバイス は、このスイッチから到達可能でなければなりません。

- 指定された送信元および宛先アドレスが同じ VLAN にある場合、**traceroute mac** コマンド出力はレイヤ 2 パスを表示します。指定した送信元および宛先 MAC アドレスが、それぞれ異なる VLAN に属している場合は、レイヤ 2 パスは識別されず、エラー メッセージが表示されます。

- マルチキャストの送信元または宛先 MAC アドレスを指定すると、パスは識別されず、エラー メッセージが表示されます。

- 送信元または宛先 MAC アドレスが複数の VLAN に属する場合は、送信元および宛先 MAC アドレスの両方が属している VLAN を指定する必要があります。VLAN を指定しないと、パスは識別されず、エラー メッセージが表示されます。

- 指定された送信元および宛先の IP アドレスが同一のサブネット内にある場合、**traceroute mac ip** コマンド出力はレイヤ 2 パスを表示します。IP アドレスを指定した場合、デバイス は Address Resolution Protocol（ARP）を使用し、IP アドレスとそれに対応する MAC アドレスおよび VLAN ID を対応させます。

  - 指定の IP アドレスの ARP のエントリが存在している場合、デバイス は関連付けられた MAC アドレスを使用し、物理パスを識別します。

  - ARP のエントリが存在しない場合、デバイス は ARP クエリを送信し、IP アドレスを解決しようと試みます。IP アドレスが解決されない場合は、パスは識別されず、エラー メッセージが表示されます。

- 複数のデバイスがハブを介して 1 つのポートに接続されている場合（たとえば複数の CDP ネイバーがポートで検出された場合）、レイヤ 2 traceroute 機能はサポートされません。複数の CDP ネイバーが 1 つのポートで検出された場合、レイヤ 2 パスは特定されず、エラー メッセージが表示されます。

- この機能は、トークンリング VLAN ではサポートされません。

- レイヤ 2 トレースルートは、ユーザ データグラム プロトコル（UDP）ポート 2228 でリスニングソケットを開きます。このポートは、任意の IPv4 アドレスを使用してリモートからアクセスでき、認証は必要ありません。この UDP ソケットにより、VLAN 情報、リンク、特定の MAC アドレスの存在、および CDP ネイバー情報をデバイスから読み取ることができます。この情報を使用することにより、最終的にレイヤ 2 ネットワークトポロジの全体像を構築できます。
- レイヤ 2 トレースルートはデフォルトで有効になっており、グローバル コンフィギュレーション モードで **no l2 traceroute** コマンドを実行することによって無効にできます。レイヤ 2 トレースルートを再度有効にするには、グローバル コンフィギュレーション モードで **l2 traceroute** コマンドを使用します。

IP トレースルート

IP traceroute を使用すると、ネットワーク上でパケットが通過するパスをホップバイホップで識別できます。このコマンドを実行すると、トラフィックが宛先に到達するまでに通過するルータなどのすべてのネットワーク層（レイヤ 3）デバイスが表示されます。

デバイスは、**traceroute** 特権 EXEC コマンドの送信元または宛先として指定できます。また、**traceroute** コマンドの出力でホップとして表示される場合があります。デバイス を traceroute の宛先とすると、スイッチは、traceroute の出力で最終の宛先として表示されます。中間 デバイス が同じ VLAN 内でポート間のパケットのブリッジングだけを行う場合、traceroute の出力に中間スイッチは表示されません。ただし、中間 デバイス が特定のパケットをルーティングするマルチレイヤ デバイス の場合、この デバイス は traceroute の出力にホップとして表示されます。

traceroute 特権 EXEC コマンドは、IP ヘッダーの存続可能時間（TTL）フィールドを使用して、ルータおよびサーバで特定のリターンメッセージが生成されるようにします。traceroute の実行は、ユーザ データグラム プロトコル（UDP）データグラムを、TTL フィールドが 1 に設定されている宛先ホストへ送信することから始まります。ルータで TTL 値が 1 または 0 であることを検出すると、データグラムをドロップし、インターネット制御メッセージプロトコル（ICMP）time-to-live-exceeded メッセージを送信元に送信します。traceroute は、ICMP time-to-live-exceeded メッセージの送信元アドレス フィールドを調べて、最初のホップのアドレスを判別します。

ネクスト ホップを識別するために、traceroute は TTL 値が 2 の UDP パケットを送信します。1 番めのルータは、TTL フィールドの値から 1 を差し引いて次のルータにデータグラムを送信します。2 番めのルータは、TTL 値が 1 であることを確認すると、このデータグラムを廃棄し、time-to-live-exceeded メッセージを送信元へ返します。このように、データグラムが宛先ホストに到達するまで（または TTL の最大値に達するまで）TTL の値は増分され、処理が続けられます。

データグラムが宛先に到達したことを学習するために、traceroute は、データグラムの UDP 宛先ポート番号を、宛先ホストが使用する可能性のない大きな値に設定します。ホストが、ローカルで使用されない宛先ポート番号を持つ自分自身宛てのデータグラムを受信すると、送信元に ICMP ポート到達不能エラーを送信します。ポート到達不能エラーを除くすべてのエラーは

中間ホップから送信されるため、ポート到達不能エラーを受信するということは、このメッセージが宛先ポートから送信されたことを意味します。

Time Domain Reflector ガイドライン

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

TDR は 10/100/1000 の銅線イーサネット ポート上でだけサポートされます。10 ギガビット イーサネット ポートまたは SFP モジュール ポートではサポートされません。

TDR は次のケーブル障害を検出します。

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

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

次の状況で TDR を使用して、ケーブル障害を診断および解決してください。

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

TDR の実行時、次の場合に デバイス は正確な情報をレポートします。

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

TDR の実行時、次の場合に デバイス は正確な情報をレポートしません。

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

debug コマンド

⚠

注意 デバッグ出力は CPU プロセスで高プライオリティが割り当てられているため、デバッグ出力を行うとシステムが使用できなくなることがあります。したがって、**debug** コマンドを使用するのは、特定の問題のトラブルシューティング時、またはシスコのテクニカルサポート担当者とともにトラブルシューティングを行う場合に限定してください。ネットワークトラフィック量やユーザ数が少ない期間に **debug** コマンドを使用することをお勧めします。デバッギングをこのような時間帯に行うと、**debug** コマンド処理のオーバーヘッドの増加によりシステムの使用に影響が及ぶ可能性が低くなります。

debug コマンドはすべて特権 EXEC モードで実行します。ほとんどの **debug** コマンドは引数を取りません。

スイッチのオンボード障害ロギング

オンボード障害ロギング（OBFL）機能を使用すれば、デバイス に関する情報を収集できます。この情報には稼働時間、温度、電圧などの情報が含まれており、シスコのテクニカルサポート担当者が デバイス の問題をトラブルシューティングする際に役立ちます。OBFL はイネーブルにしておき、フラッシュ メモリに保存されたデータは消さないようにすることを推奨します。

OBFL は、デフォルトでイネーブルになっています。デバイスおよびSmall Form-Factor Pluggable（SFP）モジュールに関する情報が収集されます。デバイスは、次の情報をフラッシュメモリに保存します。

- CLI コマンド：スタンドアロン デバイスに入力された OBFL CLI コマンドの記録。
- 環境データ： スタンドアロン デバイスおよび接続されているすべての FRU デバイスの一意のデバイス ID（UDI）情報、製品 ID（PID）、バージョン ID（VID）、およびシリアル番号。
- メッセージ：スタンドアロンデバイスにより生成されたハードウェア関連のシステムメッセージの記録。
- Power over Ethernet（PoE）：スタンドアロン デバイス の PoE ポートの消費電力の記録。
- 温度：スタンドアロン デバイス の温度。
- 稼働時間：スタンドアロン デバイス が起動された際の時刻、デバイス が再起動された理由、および デバイス が最後に再起動されて以来の稼働時間。
- 電圧：スタンドアロン デバイス のシステム電圧。

システム時計は、手動で時刻を設定するか、またはネットワーク タイム プロトコル（NTP）を使用するように設定します。

デバイス の稼働中には、**show logging onboard** 特権 EXEC コマンドを使用することにより、OBFL データを取得できます。デバイス に障害が発生した場合のデータの取得方法については、お客様担当のシスコテクニカルサポート担当者にお問い合わせください。

OBFL がイネーブルになっている デバイス が再起動された場合、新しいデータの記録が開始するまでに 10 分間の遅延があります。

CPU 使用率が高い場合に起こりうる症状

CPU使用率が高すぎることで次の症状が発生する可能性がありますが、他の原因で発生する場合もあります。

（注） Cisco Catalyst 4500E Supervisor Engine 8-E をワイヤレス モードで使用すると、システムのメモリ使用率が上がることがあります。

- スパニングツリー トポロジの変更
- 通信が切断されたために EtherChannel リンクがダウンした
- 管理要求（ICMP ping、SNMP のタイムアウト、低速な Telnet または SSH セッション）に応答できない
- UDLD フラッピング
- SLA の応答が許容可能なしきい値を超えたことによる IP SLA の失敗
- スイッチが要求を転送しない、または要求に応答しない場合の DHCP または IEEE 802.1x の処理の失敗

レイヤ 3 スイッチの場合：

- ソフトウェアでルーティングされるパケットのドロップまたは遅延の増加
- BGP または OSPF ルーティング トポロジの変更
- HSRP フラッピング

ソフトウェア設定のトラブルシューティング方法

ソフトウェア障害からの回復

アップグレード中にスイッチ ソフトウェアが破損する状況としては、スイッチに誤ったファイルをダウンロードした場合や、イメージファイルを削除した場合などが考えられます。いずれの場合にも、スイッチは電源投入時自己診断テスト（POST）に失敗し、接続できなくなります。

次の手順では、XMODEMプロトコルを使用して、破損したイメージファイルまたは間違ったイメージファイルを回復します。XMODEMプロトコルをサポートするソフトウェアパッケージは多数あり、使用するエミュレーション ソフトウェアによって、この手順は異なります。

ここで紹介する回復手順を実行するには、スイッチを直接操作する必要があります。

ステップ 1 PC 上で、Cisco.com から tar 形式のソフトウェア イメージ ファイル（*image_filename.tar*）をダウンロードします。Cisco IOS イメージは、tar ファイルのディレクトリ内に bin ファイルとして格納されます。Cisco.com 上のソフトウェア イメージ ファイルの検索方法については、リリース ノートを参照してください。

ステップ 2 tar ファイルから bin ファイルを抽出します。Windows を使用している場合は、tar ファイルの読み取り機能を備えた zip プログラムを使用します。zip プログラムを使用して移動します。Windows を使用している場合は、tar ファイルの読み取り機能を備えた zip プログラムを使用します。zip プログラムを使用して移動します。UNIX を使用している場合は、次の手順に従ってください。

a) **tar -tvf** <*image_filename.tar*> UNIX コマンドを使用して、tar ファイルの内容を表示します。

例：

```
unix-1% tar -tvf image_filename.tar
```

b) **tar -xvf** <*image_filename.tar*> <*image_filename.bin*> UNIX コマンドを使用して、bin ファイルを特定し、抽出します。

例：

```
unix-1% tar -xvf image_filename.tar image_filename.bin
x c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin, 2928176 bytes, 5720
tape blocks
```

c) **ls -l** <*image_filename.bin*> UNIX コマンドを使用して、bin ファイルが抽出されたことを確認します。

例：

```
unix-1% ls -l image_filename.bin
-rw-r--r--   1 boba      2928176 Apr 21 12:01
c2960x-universalk9-mz.150-2.0.66.UCP/c2960x-universalk9-mz.150-2.0.66.UCP.bin
```

ステップ 3 XMODEM プロトコルをサポートする端末エミュレーション ソフトウェアを備えた PC を、スイッチのコンソール ポートに接続します。

ステップ 4 エミュレーション ソフトウェアの回線速度を 9600 ボーに設定します。

ステップ 5 スイッチの電源コードを取り外します。

ステップ 6 [Mode] ボタンを押しながら、電源コードを再度スイッチに接続します。ポート 1 の上の LED が消灯してから 1 ～ 2 秒後に、[Mode] ボタンを放します。ソフトウェアに関する数行分の情報と指示が表示されます。

例：

```
The system has been interrupted prior to initializing the flash file system. The following
commands will initialize the flash file system, and finish loading the operating system
software#
```

```
flash_init

load_helper

boot
```

ステップ 7 フラッシュ ファイル システムを初期化します。

例：

```
switch: flash_init
```

ステップ 8 コンソール ポートの速度を 9600 以外に設定していた場合、9600 にリセットされます。エミュレーション ソフトウェアの回線速度をスイッチのコンソール ポートに合わせて変更します。

ステップ 9 ヘルパー ファイルがある場合にはロードします。

例：

```
switch: load_helper
```

ステップ 10 XMODEM プロトコルを使用して、ファイル転送を開始します。

例：

```
switch: copy xmodem: flash:image_filename.bin
```

ステップ 11 XMODEM 要求が表示されたら、端末エミュレーション ソフトウェアに適切なコマンドを使用して、転送を開始し、ソフトウェア イメージをフラッシュ メモリにコピーします。

ステップ 12 新規にダウンロードされた Cisco IOS イメージを起動します。

例：

```
switch: boot flash:image_filename.bin
```

ステップ 13 **archive download-sw** 特権 EXEC コマンドを使用して、スイッチまたはスイッチスタックにソフトウェア イメージをダウンロードします。

ステップ 14 **reload** 特権 EXEC コマンドを使用してスイッチを再起動し、新しいソフトウェア イメージが適切に動作していることを確認します。

ステップ 15 スイッチから、**flash**:*image_filename.bin* ファイルを削除します。

パスワードを忘れた場合の回復

スイッチのデフォルト設定では、スイッチを直接操作するエンドユーザが、スイッチの電源投入時に起動プロセスを中断して新しいパスワードを入力することにより、パスワードを紛失した状態から回復できます。ここで紹介する回復手順を実行するには、スイッチを直接操作してください。

（注） これらのスイッチでは、システム管理者はデフォルト設定に戻す場合に限りエンド ユーザによるパスワードのリセットを許可することによって、この機能の一部をディセーブルにできます。パスワード回復がディセーブルになっている場合に、エンド ユーザがパスワードをリセットしようとすると、回復プロセスの間、ステータス メッセージにその旨が表示されます。

パスワードの回復をイネーブルまたはディセーブルにするには、**service password-recovery** グローバル コンフィギュレーション コマンドを使用します。

ステップ 1 端末または PC をスイッチに接続します。

- 端末または端末エミュレーション ソフトウェアが稼働している PC をスイッチのコンソール ポートに接続します。

  または

- PC をイーサネット管理ポートに接続します。

ステップ 2 エミュレーション ソフトウェアの回線速度を 9600 ボーに設定します。

ステップ 3 スイッチの電源を切断します。

ステップ 4 スイッチに電源コードを再接続します。15 秒以内に **Mode** ボタンを押します。このときシステム LED はグリーンに点滅しています。すべてのシステム LED が点灯した状態になるまで、**Mode** ボタンを押し続けます。その後、**Mode** ボタンを放します。

ソフトウェアについての情報および指示が数行表示され、パスワード回復手順がディセーブルであるかどうかが示されます。

- 次のステートメントで始まるメッセージが表示された場合

```
The system has been interrupted prior to initializing the flash file system. The following
commands will initialize the flash file system
```

「パスワード回復がイネーブルになっている場合の手順」セクションに記載されている手順を実行します。

- 次のステートメントで始まるメッセージが表示された場合

```
The password-recovery mechanism has been triggered, but is currently disabled.
```

「パスワード回復がディセーブルになっている場合の手順」に記載されている手順を実行します。

ステップ 5 パスワードが回復したら、スイッチをリロードします。

スイッチの場合

```
Switch> reload
Proceed with reload? [confirm] y
```

パスワード回復がイネーブルになっている場合の手順

パスワード回復動作がイネーブルになっている場合は、次のメッセージが表示されます。

```
The system has been interrupted prior to initializing the flash file system. The following
 commands will initialize the flash file system, and finish loading the operating system
 software:

flash_init
load_helper
boot
```

ステップ 1 フラッシュ ファイル システムを初期化します。

スイッチ: **flash_init**

ステップ 2 コンソール ポートの速度を 9600 以外の値に設定していた場合、9600 にリセットされます。エミュレーション ソフトウェアの回線速度をスイッチのコンソール ポートに合わせて変更します。

ステップ 3 ヘルパー ファイルがある場合にはロードします。

スイッチ: **load_helper**

ステップ 4 フラッシュ メモリの内容を表示します。

```
スイッチ: dir: flash:
Directory of flash:
   13  drwx         192   Mar 01 2013 22:30:48
c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin
   11  -rwx        5825   Mar 01 2013 22:31:59  config.text

16128000 bytes total (10003456 bytes free)
```

ステップ 5 コンフィギュレーション ファイルの名前を config.text.old に変更します。

このファイルには、パスワード定義が収められています。

スイッチ: **rename flash:** ***config.text*** **flash:** ***config.text.old***

ステップ 6 システムを起動します。

スイッチ: **boot**

セットアップ プログラムを起動するように求められます。プロンプトに **N** を入力します。

```
Continue with the configuration dialog?? [yes/no]: No
```

ステップ 7 スイッチ プロンプトで、特権 EXEC モードを開始します。

```
スイッチ> enable
Switch#
```

ステップ 8 コンフィギュレーション ファイルを元の名前に戻します。

```
スイッチ# rename flash: config.text.old flash: config.text
```

（注） ステップ 9 に進む前に、接続されているすべてのスタック メンバの電源を入れ、それらが完全に初期化されるまで待ちます。このステップに従わなかった場合は、deviceの設定によっては設定を失う可能性もあります。

ステップ 9 コンフィギュレーション ファイルをメモリにコピーします。

```
スイッチ# copy flash: config.text system: running-config
Source filename [config.text]?
Destination filename [running-config]?
```

確認を求めるプロンプトに、Return を押して応答します。これで、コンフィギュレーション ファイルがリロードされ、パスワードを変更できるようになります。

ステップ 10 グローバル コンフィギュレーション モードを開始します。

```
スイッチ# configure terminal
```

ステップ 11 パスワードを変更します。

```
スイッチ(config)# enable secret password
```

シークレット パスワードは 1 ～ 25 文字の英数字です。数字で始めることができます。大文字と小文字が区別され、スペースを使用できますが、先行スペースは無視されます。

ステップ 12 特権 EXEC モードに戻ります。

```
スイッチ(config)# exit
Switch#
```

ステップ 13 実行コンフィギュレーションをスタートアップ コンフィギュレーション ファイルに書き込みます。

```
スイッチ# copy running-config startup-config
```

新しいパスワードがスタートアップ コンフィギュレーションに組み込まれました。

（注） 上記の手順を実行すると、スイッチの仮想インターフェイスがシャットダウン ステートになることがあります。このステートになっているインターフェイスを調べるには、**show running-config** 特権 EXEC コマンドを入力します。インターフェイスを再びイネーブルにするには、**interface vlan** *vlan-id* グローバル コンフィギュレーション コマンドを入力して、シャットダウン インターフェイスの VLAN ID を指定します。スイッチがインターフェイス コンフィギュレーション モードの状態で、**no shutdown** コマンドを入力します。

ステップ 14 フラッシュの *packages.conf* ファイルを使用して、deviceを起動します。

```
スイッチ: boot flash:packages.conf
```

ステップ 15 スイッチスタックをリロードします。

```
スイッチ# reload
```

パスワード回復がディセーブルになっている場合の手順

パスワード回復メカニズムがディセーブルの場合、次のメッセージが表示されます。

```
The password-recovery mechanism has been triggered, but
is currently disabled.  Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point.  However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.

Would you like to reset the system back to the default configuration (y/n)?
```

注意 デバイスをデフォルト設定に戻すと、既存の設定がすべて失われます。システム管理者に問い合わせて、バックアップデバイスと VLAN（仮想 LAN）コンフィギュレーション ファイルがあるかどうかを確認してください。

- **n**（no）を入力すると、Mode ボタンを押さなかった場合と同様に、通常のブート プロセスが継続されます。ブートローダ プロンプトにはアクセスできません。したがって、新しいパスワードを入力できません。次のメッセージが表示されます。

  ```
  Press Enter to continue........
  ```

- **y**（yes）を入力すると、フラッシュ メモリ内のコンフィギュレーション ファイルおよび VLAN データベース ファイルが削除されます。デフォルト設定がロードされるときに、パスワードをリセットできます。

ステップ 1 パスワード回復手順の継続を選択すると、既存の設定が失われます。

```
Would you like to reset the system back to the default configuration (y/n)? Y
```

ステップ 2 フラッシュ メモリの内容を表示します。

```
スイッチ: dir flash:
```

デバイスのファイル システムが表示されます。

ステップ 3 システムを起動します。

```
スイッチ: boot
```

セットアップ プログラムを起動するように求められます。パスワード回復手順を継続するには、プロンプトに **N** を入力します。

```
Continue with the configuration dialog? [yes/no]: N
```

ステップ 4 デバイスプロンプトで、特権 EXEC モードを開始します。

```
スイッチ> enable
```

ステップ 5 グローバル コンフィギュレーション モードを開始します。

```
スイッチ# configure terminal
```

ステップ 6 パスワードを変更します。

```
スイッチ(config)# enable secret password
```

シークレット パスワードは 1 ～ 25 文字の英数字です。数字で始めることができます。大文字と小文字が区別され、スペースを使用できますが、先行スペースは無視されます。

ステップ 7 特権 EXEC モードに戻ります。

```
スイッチ(config)# exit
スイッチ#
```

（注） ステップ 9 に進む前に、接続されているすべてのスタック メンバの電源を入れ、それらが完全に初期化されるまで待ちます。

ステップ 8 実行コンフィギュレーションをスタートアップ コンフィギュレーション ファイルに書き込みます。

```
スイッチ# copy running-config startup-config
```

新しいパスワードがスタートアップ コンフィギュレーションに組み込まれました。

ステップ 9 ここで、デバイスを再設定する必要があります。システム管理者によって、バックアップデバイスとVLAN コンフィギュレーション ファイルが使用可能に設定されている場合は、これらを使用します。

コマンド スイッチで障害が発生した場合の回復

ここでは、コマンドスイッチで障害が発生した場合の回復手順について説明します。ホットスタンバイ ルータ プロトコル（HSRP）を使用すると、冗長コマンド スイッチ グループを設定できます。

（注） この機能は、Cisco IOS リリース 15.2(5)E2 で導入されました。

スタンバイ コマンド スイッチが未設定で、かつコマンド スイッチで電源故障などの障害が発生した場合には、メンバ スイッチとの管理接続が失われるので、新しいコマンド スイッチに交換する必要があります。ただし、接続されているスイッチ間の接続は影響を受けません。また、メンバ スイッチも通常どおりにパケットを転送します。メンバ スイッチは、コンソール ポートを介してスタンドアロンのスイッチとして管理できます。また、IPアドレスが与えられている場合は、他の管理インターフェイスを使用して管理できます。

コマンド対応メンバ スイッチまたは他のスイッチにIP アドレスを割り当て、コマンド スイッチのパスワードを書き留め、メンバ スイッチと交換用コマンド スイッチ間の冗長接続が得られるようにクラスタを配置することにより、コマンドスイッチ障害に備えます。ここでは、故障したコマンド スイッチの交換方法を 2 通り紹介します。

- 故障したコマンド スイッチをクラスタ メンバーと交換する場合
- 故障したコマンド スイッチを他のスイッチと交換する場合

ここで紹介する回復手順を実行するには、スイッチを直接操作してください。コマンド対応スイッチについては、リリース ノートを参照してください。

故障したコマンド スイッチをクラスタ メンバーと交換する場合

故障したコマンドスイッチを同じクラスタ内のコマンド対応メンバスイッチに交換するには、次の手順に従ってください。

ステップ 1 メンバ スイッチからコマンド スイッチを外し、クラスタからコマンド スイッチを物理的に取り外します。

ステップ 2 故障したコマンドスイッチの代わりに新しいメンバスイッチを取り付け、コマンドスイッチとクラスタ メンバ間の接続を復元します。

ステップ 3 新しいコマンド スイッチでCLI セッションを開始します。

CLI にはコンソール ポートを使用してアクセスできます。また、スイッチにIP アドレスが割り当てられている場合は、Telnet を使用してアクセスできます。コンソールポートの使用方法の詳細については、『*Catalyst 3560-CX* および *2960-CX* スイッチ ハードウェア設置ガイド

ステップ 4 スイッチ プロンプトで、特権 EXEC モードを開始します。

例：

```
Switch> enable
Switch#
```

ステップ 5 故障したコマンド スイッチのパスワードを入力します。

ステップ 6 グローバル コンフィギュレーション モードを開始します。

例：

```
Switch# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.
```

ステップ 7 クラスタからメンバスイッチを削除します。

例：

```
Switch(config)# no cluster commander-address
```

ステップ 8 特権 EXEC モードに戻ります。

例：

```
Switch(config)# end
Switch#
```

ステップ 9 セットアッププログラムを使用して、スイッチの IP 情報を設定します。IP アドレス情報およびパスワードを入力するように要求されます。特権 EXEC モードで **setup** と入力し、[Return] キーを押します。

例：

```
Switch# setup

 --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
Would you like to enter basic management setup? [yes/no]:
```

ステップ 10 最初のプロンプトに **Y** を入力します。

例：

```
The prompts in the setup program vary depending on the member switch that you selected to be the
 command switch:
Continue with configuration dialog? [yes/no]: y

or

Configuring global parameters:
```

このプロンプトが表示されなければ、**enable** と入力し、Return を押してください。セットアップ プログラムを開始するには、**setup** と入力し、Return を押してください。

ステップ 11 セットアップ プログラムの質問に応答します。

ホスト名を入力するように要求された場合、メンバ スイッチで入力できる文字数は 28 ~ 31 文字に制限されます。どのスイッチでも、ホスト名の最終文字として *-n*（*n* は数字）を使用しないでください。Telnet（仮想端末）パスワードを入力するように要求された場合、パスワードには 1 ～ 25 文字の英数字を使用でき、大文字と小文字が区別され、スペースを使用できますが、先行スペースは無視されます。

ステップ 12 **enable secret** および **enable** パスワードを入力するように要求された場合、故障したコマンド スイッチのパスワードを再び入力します。

ステップ 13 要求された場合は、スイッチをクラスタ コマンドスイッチとしてイネーブルにすることを確認し、Return を押します。

ステップ 14 要求された場合は、クラスタに名前を指定し、Return を押します。

クラスタ名には 1 ～ 31 文字の英数字、ダッシュ、または下線を使用できます。

ステップ 15 初期設定が表示されたら、アドレスが正しいことを確認してください。

ステップ 16 表示された情報が正しい場合は、**Y** を入力し、Return を押します。

情報に誤りがある場合には、**N** を入力し、[Return] キーを押して、ステップ 9 からやり直します。

ステップ 17 ブラウザを起動し、新しいコマンド スイッチの IP アドレスを入力します。

ステップ 18 クラスタ メニューから、[Add to Cluster] を選択し、クラスタへ追加する候補スイッチの一覧を表示します。

故障したコマンド スイッチを他のスイッチと交換する場合

故障したコマンドスイッチを、クラスタに組み込まれていないコマンド対応スイッチと交換する場合、次の手順に従ってください。

ステップ 1 故障したコマンドスイッチの代わりに新しいスイッチを取り付け、コマンドスイッチとクラスタ メンバ間の接続を復元します。

ステップ 2 CLI にはコンソール ポートを使用してアクセスできます。また、スイッチに IP アドレスが割り当てられている場合は、Telnet を使用してアクセスできます。コンソール ポートの詳しい使用方法については、スイッチのハードウェア インストレーション ガイドを参照してください。

ステップ 3 スイッチ プロンプトで、特権 EXEC モードを開始します。

例：

```
Switch> enable
Switch#
```

ステップ 4 故障したコマンド スイッチのパスワードを入力します。

ステップ 5 セットアッププログラムを使用して、スイッチの IP 情報を設定します。IP アドレス情報およびパスワードを入力するように要求されます。特権 EXEC モードで **setup** と入力し、[Return] キーを押します。

例：

```
Switch# setup

 --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
```

```
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
Would you like to enter basic management setup? [yes/no]:
```

ステップ 6 最初のプロンプトに **Y** を入力します。

例：

```
The prompts in the setup program vary depending on the member switch that you selected to be the
 command switch:
Continue with configuration dialog? [yes/no]: y

or

Configuring global parameters:
```

このプロンプトが表示されなければ、**enable** と入力し、Return を押してください。セットアップ プログラムを開始するには、**setup** と入力し、Return を押してください。

ステップ 7 セットアップ プログラムの質問に応答します。

ホスト名を入力するように要求された場合、メンバ スイッチで入力できる文字数は 28 ~ 31 文字に制限されます。どのスイッチでも、ホスト名の最終文字として *-n*（*n* は数字）を使用しないでください。Telnet（仮想端末）パスワードを入力するように要求された場合、パスワードには 1 ～ 25 文字の英数字を使用でき、大文字と小文字が区別され、スペースを使用できますが、先行スペースは無視されます。

ステップ 8 **enable secret** および **enable** パスワードを入力するように要求された場合、故障したコマンド スイッチのパスワードを再び入力します。

ステップ 9 要求された場合は、スイッチをクラスタ コマンドスイッチとしてイネーブルにすることを確認し、Return を押します。

ステップ 10 要求された場合は、クラスタに名前を指定し、Return を押します。

クラスタ名には 1 ～ 31 文字の英数字、ダッシュ、または下線を使用できます。

ステップ 11 初期設定が表示されたら、アドレスが正しいことを確認してください。

ステップ 12 表示された情報が正しい場合は、**Y** を入力し、Return を押します。

情報に誤りがある場合には、**N** を入力し、[Return] キーを押して、ステップ 9 からやり直します。

ステップ 13 ブラウザを起動し、新しいコマンド スイッチの IP アドレスを入力します。

ステップ 14 クラスタ メニューから、[Add to Cluster] を選択し、クラスタへ追加する候補スイッチの一覧を表示します。

自動ネゴシエーションの不一致の防止

IEEE 802.3ab 自動ネゴシエーション プロトコルは速度（10 Mbps、100 Mbps、および SFP モジュール ポート以外の 1000 Mbps）およびデュプレックス（半二重または全二重）に関するスイッチの設定を管理します。このプロトコルは設定を適切に調整しないことがあり、その場合はパフォーマンスが低下します。不一致は次の条件で発生します。

- 手動で設定した速度またはデュプレックスのパラメータが、接続ポート上で手動で設定された速度またはデュプレックスのパラメータと異なっている場合。
- ポートを自動ネゴシエーションに設定したが、接続先ポートは自動ネゴシエーションを使用しない全二重に設定されている場合。

スイッチのパフォーマンスを最大限に引き出してリンクを確保するには、次のいずれかの注意事項に従って、デュプレックスおよび速度の設定を変更してください。

- 速度とデュプレックスの両方について、両方のポートで自動ネゴシエーションを実行させます。
- 接続の両側でポートの速度とデュプレックスのパラメータを手動で設定します。

（注） 接続先装置が自動ネゴシエーションを実行しない場合は、2つのポートのデュプレックス設定を一致させます。速度パラメータは、接続先のポートが自動ネゴシエーションを実行しない場合でも自動調整が可能です。

SFP モジュールのセキュリティと識別に関するトラブルシューティング

シスコの Small Form-Factor Pluggable（SFP）モジュールは、モジュールのシリアル番号、ベンダー名とベンダー ID、一意のセキュリティ コード、および巡回冗長検査（CRC）が格納されたシリアル EEPROM（電気的に消去可能でプログラミング可能な ROM）を備えています。デバイスに SFP モジュールを装着すると、デバイス ソフトウェアは、EEPROM を読み取ってシリアル番号、ベンダー名、およびベンダー ID を確認し、セキュリティコードと CRC を再計算します。シリアル番号、ベンダー名、ベンダー ID、セキュリティ コード、または CRC が無効な場合、ソフトウェアは、セキュリティ エラー メッセージを生成し、インターフェイスを errdisable ステートにします。

（注） セキュリティ エラー メッセージは、GBIC_SECURITY 機能を参照します。デバイスは、SFP モジュールをサポートしていますが、GBIC（ギガビット インターフェイス コンバータ）モジュールはサポートしていません。エラー メッセージテキストは、GBIC インターフェイスおよびモジュールを参照しますが、セキュリティ メッセージは、実際は SFP モジュールおよびモジュール インターフェイスを参照します。

他社の SFP モジュールを使用している場合、デバイス から SFP モジュールを取り外し、シスコのモジュールに交換します。シスコの SFP モジュールを装着したら、**errdisable recovery cause gbic-invalid** グローバル コンフィギュレーション コマンドを使用してポートのステータスを確認し、error-disabled 状態から回復する時間間隔を入力します。この時間間隔が経過すると、デバイス は error-disabled 状態からインターフェイスを回復させ、操作を再試行します。**errdisable recovery** コマンドの詳細については、このリリースに対応するコマンドリファレンスを参照してください。

モジュールがシスコ製SFPモジュールとして識別されたにもかかわらず、システムがベンダーデータ情報を読み取ってその情報が正確かどうかを確認できないと、SFP モジュール エラー メッセージが生成されます。この場合、SFPモジュールを取り外して再び装着してください。それでも障害が発生する場合は、SFP モジュールが不良品である可能性があります。

SFP モジュール ステータスのモニタリング

show interfaces transceiver 特権 EXEC コマンドを使用すると、SFP モジュールの物理または動作ステータスを確認できます。このコマンドは、温度や特定のインターフェイス上の SFP モジュールの現状などの動作ステータスと、アラーム ステータスを表示します。また、このコマンドを使用してSFPモジュールの速度およびデュプレックス設定も確認できます。詳細については、このリリースに対応するコマンドリファレンスにある **show interfaces transceiver** コマンドを参照してください。

ping の実行

別の IP サブネットワーク内のホストに ping を実行する場合は、ネットワークへのスタティック ルートを定義するか、またはこれらのサブネット間でルーティングされるようにIPルーティングを設定する必要があります。

IP ルーティングは、デフォルトではすべてのデバイスでディセーブルになります。

（注） **ping** コマンドでは、他のプロトコルキーワードも使用可能ですが、このリリースではサポートされていません。

このコマンドは、デバイスからネットワーク上の他のデバイスに ping を実行する目的で使用します。

コマンド	目的
ping ip *host* | *address* `スイッチ# ping 172.20.52.3`	IP またはホスト名やネットワーク アドレスを指定してリモート ホストに ping を実行します。

温度のモニタリング

デバイス は温度条件をモニターし、温度情報を使用してファンを制御します。

温度の値、状態、しきい値を表示するには、**show env temperature status** 特権 EXEC コマンドを使用します。温度の値は、デバイス内の温度です（外部温度ではありません）。**system env temperature threshold yellow** *value*グローバル コンフィギュレーション コマンドを使用して、イエローのしきい値（摂氏）のみを設定し、イエローのしきい値とレッドのしきい値の差を設定できます。グリーンまたはレッドのしきい値は設定できません。詳細については、このリリースのコマンド リファレンスを参照してください。

物理パスのモニタリング

次のいずれかの特権EXECコマンドを使用して、パケットが通過する、送信元デバイスから宛先デバイスへの物理パスをモニタできます。

表 *168 :* 物理パスのモニタリング

コマンド	目的
tracetroute mac [**interface** *interface-id*] {*source-mac-address*} [**interface** *interface-id*] {*destination-mac-address*} [**vlan** *vlan-id*] [**detail**]	指定の送信元MACアドレスから、指定の宛先MACアドレスまでをパケットが通過するレイヤ 2 パスを表示します。
tracetroute mac ip {*source-ip-address* | *source-hostname*}{*destination-ip-address* | *destination-hostname*} [**detail**]	指定の送信元 IP アドレスまたはホスト名から、指定の宛先 IP アドレスまたはホスト名を通過するパケットのレイヤ 2 パスを表示します。

IP traceroute の実行

（注） **traceroute** 特権 EXEC コマンドでは、他のプロトコルキーワードも使用可能ですが、このリリースではサポートされていません。

コマンド	目的
traceroute ip *host* `スイッチ# traceroute ip 192.51.100.1`	ネットワーク上でパケットが通過するパスを追跡します。

TDR の実行および結果の表示

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

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

デバッグおよびエラー メッセージ出力のリダイレクト

デフォルトでは、ネットワークサーバが **debug** コマンドからの出力とシステムエラーメッセージをコンソールに送信します。このデフォルトの設定を使用する場合は、コンソールポートまたはイーサネット管理ポートに接続する代わりに、仮想端末接続によってデバッグ出力をモニタできます。

指定できる宛先として、コンソール、仮想端末、内部バッファ、およびsyslogサーバを実行している UNIX ホストがあります。Syslog フォーマットは、4.3 BSD UNIX およびそのバリエーションと互換性があります。

（注） デバッグの出力先がシステムのオーバーヘッドに影響を与えることがないように注意してください。メッセージをコンソールに記録すると、非常に高いオーバーヘッドが発生します。仮想端末にメッセージを記録すると、発生するオーバーヘッドは低くなります。Syslog サーバでメッセージ ロギングを行うと、オーバーヘッドはさらに小さくなり、内部バッファであれば最小限ですみます。

システム メッセージのロギングに関する詳細については、「システム メッセージ ロギングの設定」を参照してください。

show platform forward コマンドの使用

show platform forward 特権 EXEC コマンドの出力からは、インターフェイスに着信するパケットがシステムを介して送信された場合の転送結果に関する有意義な情報がいくつか得られます。パケットに関して入力されたパラメータに応じて、参照テーブル結果、転送宛先の計算に使用されるポート マップ、ビットマップ、および出力側の情報が表示されます。

このコマンドで出力される情報のほとんどは、主に、デバイスの特定用途向け集積回路（ASIC）に関する詳細情報を使用するテクニカルサポート担当者に役立つものです。ただし、パケット転送情報はトラブルシューティングにも役立ちます。

OBFL の設定

⚠

注意 OBFL はディセーブルにせず、フラッシュ メモリに保存されたデータは削除しないことを推奨します。

- OBFL をイネーブルにするには、**hw-switch switch** [*switch-number*] **logging onboard** [**message level** *level*] グローバル コンフィギュレーション コマンドを使用します。スイッチでは、*switch-number* に指定できる範囲は 1 ～ 9 です。スイッチが生成してフラッシュメモリに保存するハードウェア関連のメッセージの重大度を指定するには、**message level** *level* パラメータを使用します。
- OBFL データをローカルネットワークまたは特定のファイルシステムにコピーするには、**copy onboard switch** *switch-number* **url** *url-destination* 特権 EXEC コマンドを使用します。
- OBFL をイネーブルにするには、**no hw-switch switch** [*switch-number*] **logging onboard** [**message level**] グローバル コンフィギュレーション コマンドを使用します。
- フラッシュメモリ内の稼働時間と CLI コマンド情報以外のすべての OBFL データをクリアするには、**clear onboard switch** *switch-number* 特権 EXEC コマンドを使用します。
- アクティブスタックのメンバースイッチの OBFL をイネーブルまたはディセーブルにできます。

ここで説明した各コマンドの詳細については、このリリースのコマンドリファレンスを参照してください。

ソフトウェア設定のトラブルシューティングの確認

OBFL 情報の表示

表 *169 : OBFL* 情報を表示するためのコマンド

コマンド	目的
show logging onboard [**module**[*switch-number*]]**clilog** `スイッチ# show logging onboard  1 clilog`	スタンドアロンスイッチまたはで入力された OBFL CLI コマンドを表示します。
show logging onboard [**module**[*switch-number*]] **environment** `スイッチ# show logging onboard 1 environment`	スタンドアロンスイッチおよび接続されているすべての FRU デバイスの UDI 情報、PID、VID、およびシリアル番号を表示します。

コマンド	目的
show logging onboard [**module**[*switch-number*]] **message** スイッチ# show logging onboard 1 message	スタンドアロンスイッチによって生成されたハードウェア関連のメッセージを表示します。
show logging onboard [**module**[*switch-number*]] **poe** スイッチ# show logging onboard 1 poe	スタンドアロンスイッチのPoEポートの消費電力を表示します。
show logging onboard [**module**[*switch-number*]] **temperature** スイッチ# show logging onboard 1 temperature	スタンドアロンスイッチの温度を表示します。
show logging onboard [**module**[*switch-number*]] **uptime** スイッチ# show logging onboard 1 uptime	スタンドアロンスイッチまたは指定されたスタックメンバが起動した時刻、スタンドアロンスイッチまたは指定されたスタックメンバが再起動された理由、およびスタンドアロンスイッチが最後に再起動されて以来の稼働時間を表示します。
show logging onboard [**module**[*switch-number*]] **voltage** スイッチ# show logging onboard 1 voltage	スタンドアロンスイッチのシステム電圧を表示します。
show logging onboard [**module**[*switch-number*]] **continuous** スイッチ# show logging onboard 1 continuous	連続ファイルのデータを表示します。
show logging onboard [**module**[*switch-number*]] **detail** スイッチ# show logging onboard 1 detail	連続データおよびサマリーデータの両方を表示します。
show logging onboard [**module**[*switch-number*]] **end***hh:mm:ss* スイッチ# show logging onboard 1 end 13:00:15 jul 2013	スタンドアロンスイッチの終了日時を表示します。
show logging onboard [**module**[*switch-number*]] スイッチ# show logging onboard 1	システム内で指定されているスイッチに関するOBFL情報を表示します。

コマンド	目的
show logging onboard [**module**[*switch-number*]] **raw** スイッチ# show logging onboard 1 raw	スタンドアロンスイッチの raw 情報を表示します。
show logging onboard [**module**[*switch-number*]] **start** スイッチ# show logging onboard 1 start 13:00:10 jul 2013	スタンドアロンスイッチの開始日時を表示します。
show logging onboard [**module**[*switch-number*]] **status** スイッチ# show logging onboard 1 status	スタンドアロンスイッチのステータス情報を表示します。
show logging onboard [**module**[*switch-number*]] **summary** スイッチ# show logging onboard 1 summary	サマリーファイルの両方のデータを表示します。

例：高い CPU 使用率に関する問題と原因の確認

CPU 使用率が高いことが問題となっているかどうか判別するには、**show processes cpu sorted** 特権 EXEC コマンドを入力します。出力例の 1 行目にある下線が付いた部分に注目してください。

```
スイッチ# show processes cpu sorted
CPU utilization for five seconds: 8%/0%; one minute: 7%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
309 42289103 752750 56180 1.75% 1.20% 1.22% 0 RIP Timers
140 8820183 4942081 1784 0.63% 0.37% 0.30% 0 HRPC qos request
100 3427318 16150534 212 0.47% 0.14% 0.11% 0 HRPC pm-counters
192 3093252 14081112 219 0.31% 0.14% 0.11% 0 Spanning Tree
143 8 37 216 0.15% 0.01% 0.00% 0 Exec
...
<output truncated>
```

この例は、正常な CPU 使用率を示しています。この出力によると、最後の 5 秒間の使用率が 8%/0% となっていますが、この意味は次のとおりです。

- Cisco IOS の処理時間と割り込みの処理にかかった時間を合わせた CPU の合計の使用率は全体の 8%
- 割り込みの処理にかかった時間は全体の 0%

表 *170 : CPU* 使用率に関する問題のトラブルシューティング

問題のタイプ	原因	修正措置
割り込みのパーセント値が合計の CPU 使用率の値とほぼ同程度に高い	CPU がネットワークから受信するパケット数が多すぎる。	ネットワーク パケットのソースを判別する。データの流れを遮断するか、スイッチの設定を変更します。「Analyzing Network Traffic（ネットワーク トラフィックの解析）」の項を参照してください。
割り込みの所要時間は最小限であったにもかかわらず CPU の合計使用率が 50% を超える	CPU 時間を過度に消費する Cisco IOS 処理が 1 つ以上存在する。これは通常、処理をアクティブ化するイベントによって始動されます。	異常なイベントを特定して根本的な原因を解消する。「Debugging Active Processes（アクティブなプロセスのデバッグ）」のセクションを参照してください。

ソフトウェア設定のトラブルシューティングのシナリオ

Power over Ethernet（PoE）に関するトラブルシューティングのシナリオ

表 *171 : Power over Ethernet* に関するトラブルシューティングのシナリオ

症状または問題	考えられる原因と解決法
PoE がないポートは 1 つに限りません。 1 つのスイッチ ポートに限り問題が発生する。このポートでは PoE 装置と PoE 非対応の装置のいずれも動作しないが、他のポートでは動作します。	

症状または問題	考えられる原因と解決法
	この受電デバイスが他のPoEポートで動作するかを確認する。 **show run** または **show interface status** ユーザ EXEC コマンドを使用して、ポートがシャットダウンしていないか、または error-disabled になっていないかを確認します。 （注） ほとんどのスイッチはポートがシャットダウンしているときはポートの電力供給をオフにします。これは、IEEE 仕様でこれがオプションに指定されている場合も同様です。 該当するインターフェイスまたはポートに **power inline never** が設定されていないことを確認します。 受電デバイスからスイッチポートまでのイーサネットケーブルの動作が正常であることを確認します。具体的には、既知の正常なPoE非対応のイーサネット装置とイーサネットケーブルを接続して、受電デバイスがリンクを確立し他のホストとトラフィックを交換することを確認します。 （注） シスコ受電装置は、ストレート ケーブルでのみ機能します。クロスオーバ ケーブルでは機能しません。 スイッチのフロントパネルから受電デバイスまでのケーブル長の合計が 100 メートル以下であることを確認します。 スイッチ ポートからイーサネット ケーブルを外します。短いイーサネットケーブルを使用して、既知の正常なイーサネット装置を、スイッチのフロント パネルの（パッチ パネルではない）このポートに直接接続します。これによってイーサネット リンクが確立され他のホストとトラフィックを交換できることを確認します。あるいは、ポートのVLAN SVI で ping を実行してください。次に、受電デバイスをこのポートに接続し、電源がオンになることを確認します。 パッチ コードをスイッチ ポートに接続しても受電デバイスの電源がオンにならない場合、接続する受電デバイスの合計数とスイッチの電力バジェット（使用可能な PoE）とを比較してください。**show power inline** コマンドを使用して、利用可能な電力量を確認します。

症状または問題	考えられる原因と解決法
すべてのポートまたは1つのポート グループで PoE が機能しない。 すべてのスイッチ ポートで問題が発生する。電力が供給されていないイーサネット装置がどのポートでもイーサネット リンクを確立できず、PoE装置の電源がオンになりません。	

症状または問題	考えられる原因と解決法
	電力に関するアラームが継続的に発生する、断続的に発生する、または再発する場合は、可能であれば電源モジュールを交換します（現場交換可能ユニットです）。そうでない場合はスイッチを交換してください。 連続する複数のポートで問題があるものの、すべてのポートで問題が発生するわけではない場合、電源の故障ではないと考えられ、スイッチのPoEレギュレータに関連した異常の可能性があります。 PoEの状況やステータスの変更について過去に報告されているアラームまたはシステムメッセージがないか、**show log** 特権 EXEC コマンドを使用して調べます。 アラームがない場合は、**show interface status** コマンドを使用して、ポートがシャットダウンしていないか、またはerror-disabledになっていないかを確認します。ポートがerror-disabledの場合、**shut** および **no shut** インターフェイス コンフィギュレーション コマンドを使用して、ポートを再度有効にします。 **show env power** および **show power inline** 特権 EXEC コマンドを使用して、PoEのステータスおよび電力バジェット（使用可能なPoE）を調べます。 実行コンフィギュレーションを調べて、**power inline never** がこのポートに設定されていないことを確認します。 受電していないイーサネット装置をスイッチポートに直接接続します。接続には短いパッチ コードだけを使用します。既存の配線ケーブルは使用しないでください。**shut** および **no shut** インターフェイス コンフィギュレーション コマンドを入力し、イーサネットリンクが確立されていることを確認します。正しく接続している場合、短いパッチコードを使用して受電デバイスをこのポートに接続し、電源がオンになることを確認します。装置の電源がオンになったら、すべての中間パッチパネルが正しく接続されているか確認してください。 1本を除くすべてのイーサネットケーブルをスイッチポートから抜きます。短いパッチ コードを使用して、1つのPoEポートにだけ受電デバイスを接続します。スイッチポートからの受電に比較して、受電デバイスが多くの電力を必要としないことを確認してください。 **show power inline** 特権 EXEC コマンドを使用して、ポートがシャットダウンされていない場合に、受電デバイスに電力が供給されることを確認します。あるいは、受電デバイ

症状または問題	考えられる原因と解決法
	スを観察して電源がオンになることを確認してください。 1 台の受電デバイスだけがスイッチに接続している際に電力が供給される場合、残りのポートで **shut** および **no shut** インターフェイス コンフィギュレーション コマンドを入力してから、イーサネットケーブルをスイッチの PoE ポートに 1 本ずつ再接続してください。**show interface status** および **show power inline** 特権 EXEC コマンドを使用して、インラインパワーの統計情報とポートのステータスをモニタします。 すべてのポートで、まだ PoE が機能しない場合は、電源装置の PoE セクションでヒューズを開くことができる場合があります。この場合、アラームが生成されるのが一般的です。過去にシステム メッセージでアラームが報告されていないか、ログをもう一度チェックしてください。
シスコ先行標準受電装置は、切断またはリセットされます。 正常に動作した後で、Cisco phone が断続的にリロードしたり、PoE から切断されたりします。	スイッチから受電デバイスまでのすべての電気系統を確認してください。信頼性の低い接続は、電力供給の中断や受電デバイスの機能が不安定になる原因となり、受電デバイスの断続的な切断やリロードなどが発生します。 スイッチ ポートから受電デバイスまでのケーブル長が 100 メートル以下であることを確認してください。 スイッチが配置されている場所で電気環境にどのような変化があるか、切断時に、受電デバイスに何が起きるかについて注意してください。 切断と同時にエラー メッセージが表示されたか注意します。**show log** 特権 EXEC コマンドを使用して、エラーメッセージを確認します。 リロードの発生直前に IP Phone から Call Manager へのアクセスが失われていないか確認してください（PoE の障害ではなくネットワークに問題が発生している場合があります）。 受電デバイスを PoE 非対応の装置に交換し、装置が正しく動作することを確認します。PoE 非対応の装置にリンク障害または高いエラー率がある場合、スイッチ ポートと受電デバイスを接続する信頼性の低いケーブル接続が問題の可能性があります。

症状または問題	考えられる原因と解決法
IEEE 802.3af 準拠または IEEE 802.3at 準拠の受電装置は、Cisco PoE スイッチでは機能しません。 シスコ PoE スイッチに接続するシスコ以外の受電デバイスに電源が供給されないか、電源投入後すぐに電源が切れます。PoE 非対応装置は正常に動作します。	**show power inline** コマンドを使用して、受電デバイスの接続前後に、スイッチの電力バジェット（使用可能な PoE）が枯渇していないか確認します。受電デバイスを接続する前に、このタイプの装置に十分な電力が使用可能であることを確認します。 **show interface status** コマンドを使用して、接続されている受電デバイスがスイッチに検出されることを確認します。 **show log** コマンドを使用して、ポートの過電流状態を報告したシステムメッセージがないか確認します。症状を正確に特定してください。最初に電力が受電デバイスに供給され、その後、切断される状態ですか。その場合は、問題は最初のサージ電流（突入電流）が原因で、ポートの電流上限しきい値が超過した可能性があります。

ソフトウェアのトラブルシューティングの設定例

例：IP ホストの ping

次に、IP ホストに ping を実行する例を示します。

```
スイッチ# ping 172.20.52.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
スイッチ#
```

表 *172 : ping* の出力表示文字

文字	Description
!	感嘆符 1 個につき 1 回の応答を受信したことを示します。
.	ピリオド 1 個につき応答待ちの間にネットワーク サーバのタイムアウトが 1 回発生したことを示します。
U	宛先到達不能エラー PDU を受信したことを示します。
C	輻輳に遭遇したパケットを受信したことを示します。
I	ユーザによりテストが中断されたことを示します。

文字	Description
?	パケット タイプが不明です。
&	パケットの存続時間を超過したことを示します。

ping セッションを終了するには、エスケープ シーケンス（デフォルトでは Ctrl+^ X）を入力してください。Ctrl キー、Shift キー、および 6 キーを同時に押してから放し、その後 X キーを押します。

例：IP ホストに対する traceroute の実行

次に、IP ホストに **traceroute** を実行する例を示します。

```
スイッチ# traceroute ip 192.0.2.10

Type escape sequence to abort.
Tracing the route to 192.0.2.10

  1 192.0.2.1 0 msec 0 msec 4 msec
  2 192.0.2.203 12 msec 8 msec 0 msec
  3 192.0.2.100 4 msec 0 msec 0 msec
  4 192.0.2.10 0 msec 4 msec 0 msec
```

ディスプレイには、送信される 3 つのプローブごとに、ホップ カウント、ルータの IP アドレス、およびラウンドトリップ タイム（ミリ秒単位）が表示されます。

表 *173 : traceroute* の出力表示文字

文字	Description
*	プローブがタイムアウトになりました。
?	パケット タイプが不明です。
A	管理上、到達不能です。通常、この出力は、アクセス リストがトラフィックをブロックしていることを表しています。
H	ホストが到達不能です。
N	ネットワークが到達不能です。
P	プロトコルが到達不能です。
Q	発信元。
U	ポートが到達不能です。

実行中の追跡を終了するには、エスケープ シーケンス（デフォルトではCtrl+^ X）を入力してください。Ctrl キー、Shift キー、および 6 キーを同時に押してから放し、その後 X キーを押します。

例：すべてのシステム診断をイネーブルにする

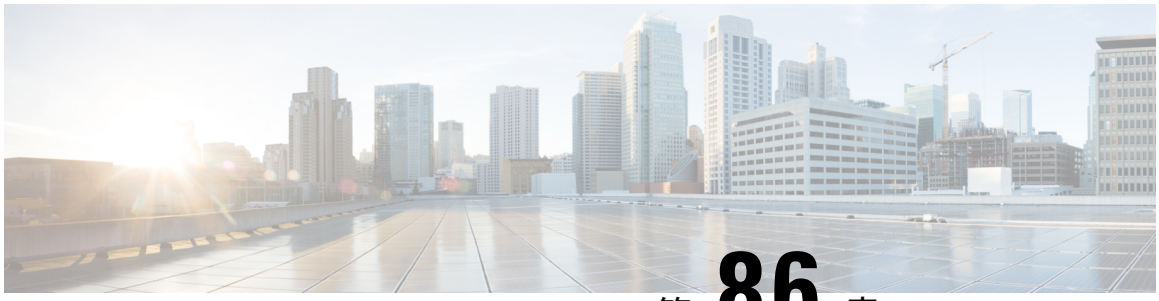
注意 デバッグ出力は他のネットワークトラフィックより優先され、**debug all** 特権 EXEC コマンドは他の **debug** コマンドより出力が大量になるので、スイッチのパフォーマンスが極度に低下したり、場合によっては使用不能になったりすることがあります。状況にかかわらず、特定性の高い **debug** コマンドを使用するのが原則です。

このコマンドは、すべてのシステム診断をディセーブルにします。

```
スイッチ# debug all
```

no debug all 特権 EXEC コマンドを使用すると、すべての診断出力がディセーブルになります。いずれかの **debug** コマンドが誤ってイネーブルのままにならないようにするには、**no debug all** コマンドを使用すると便利です。

例：すべてのシステム診断をイネーブルにする



第 86 章

ライセンシングについての情報

ライセンスの設定に関する制約事項

- スイッチスタックのメンバーでは、同じライセンスレベル（基本ライセンスレベルとアドオン）を実行する必要があります。基本ライセンスが一致せずライセンスレベルが異なる場合、レベルを変更してアクティブスタックから再起動するまでは、スイッチはスタックに参加しません。アドオンライセンスが一致していない場合は、アクティブスタックによって自動的に同期されます。
- 永久ライセンスは 1 つのデバイスから別のデバイスに移動できます。ライセンスをアクティブ化するには、スイッチを再起動する必要があります。
- 再起動後に、期限が切れた評価ライセンスを再びアクティブ化することはできません。

ライセンシングについての情報

ライセンスレベルの概要

スイッチのソフトウェア機能は、基本（機能セットとも呼ばれます）およびアドオンライセンスレベルで使用できます。有効期間によってライセンスタイプが決まります。

- スイッチの**基本ライセンスレベル**は、スイッチのモデル番号で示されます。常に期限日のない永久ライセンスです。

- **アドオンライセンスレベル**では、スイッチだけでなく Cisco Digital Network Architecture Center（Cisco DNA Center）でもシスコのイノベーションとなる機能を得られます。アドオンライセンスは、3、5、または 7 年間のライセンスタイプでのみ注文できます。

基本ライセンス

Cisco Catalyst 3560-CX シリーズ コンパクト スイッチには、IP Base ライセンスが付属しており、IP Services ライセンスを使用するようにアップグレードできます。Cisco Catalyst 2960-CX シリーズ コンパクト スイッチには、LAN Base ライセンスレベルが付属しています。

（注） Cisco Catalyst 2960-CX シリーズでは、基本ライセンスレベルはハードウェアモデルにバインドされており、変更できません。

アドオン ライセンス

Cisco Catalyst 3560-CX シリーズ コンパクト スイッチでは、次のアドオンライセンスを使用できます。

- DNA Essentials
- DNA Advantage

Cisco Catalyst 2960-CX シリーズ コンパクト スイッチでは、Cisco DNA Essentials アドオンライセンスを使用できます

アドオンライセンスには次のガイドラインが適用されます。

- アドオンライセンスを設定する場合、再起動は必要ありません。
- アドオンライセンスは、3 年、5 年、または 7 年単位で注文できます。
- 日単位で電子メールアラートを受信し、アドオンライセンスの更新期限通知を受け取るには、Cisco SSM を設定する必要があります。
- Cisco Catalyst 3560-CX シリーズの場合：IP Base + Cisco DNA Advantage の組み合わせを注文すると、SDA で使用できる仮想ネットワークは 3 つだけです。
- Cisco Catalyst 2960-CX シリーズの場合：Cisco DNA Essentials アドオンライセンスのみが利用可能です。（CLI には表示されますが、Cisco DNA Advantage ライセンスレベルは使用できません）。

ライセンスの状態

特権 EXEC モードで **show license** コマンドを使用して、ライセンス情報にアクセスすることもできます。

表 *174 :* 使用権ライセンスの状態

License State	説明
Active, In Use	EULA が承認され、デバイス再起動後にライセンスが使用されています。
Active, Not In Use	EULA が承認され、ライセンスが有効になった時点で、スイッチを使用する準備が整っています。
非アクティブ化	EULA が承認されませんでした。

次に、スイッチのライセンスレベルを表示する例を示します。この例では、LAN Base がアクティブかつ使用中のライセンスとして示されています。

```
Switch# show license

Index 1
 License Name    : lanlite
 Period left     : 0  minute  0  second
 License Type: Permanent
 License State: Inactive
Index 2
 License Name    : lanbase
 Period left     : 0  minute  0  second
 License Type: Permanent
 License State: Active, In use

Index 3
 License Name    : dna-essentials
 Period left     : CSSM Managed
 License Type    : Subscription
 License State   : Active, In use

Index 4
 License Name    : dna-advantage
 Period left     : CSSM Managed
 License Type    : Subscription
 License State   : Not Activated
```

イメージ ベースのライセンスの状態をモニタする場合のガイドラインは次のとおりです。

- 購入した永久ライセンスは、スイッチの再起動後のみに Active, In Use 状態に設定されます。
- 複数のライセンスを購入した場合は、再起動すると最も高い機能セットのライセンスがアクティブ化されます。たとえば、LAN Base ライセンスがアクティブ化され、LAN Lite ライセンスはアクティブ化されません。
- スイッチの再起動後も、残りの購入済みライセンスはアクティブで未使用の状態のままです。

ライセンスタイプのガイドライン

ライセンスは、永久タイプまたは期間タイプのみです。

- 永久：ライセンスレベル、有効期限なし。スイッチの基本ライセンスタイプはモデルによって決まり、常に無期限です。
- 有効期間付き：ライセンスレベル、3 年、5 年、または 7 年の期間。アドオンライセンス（DNA Essentials および DNA Advantage）の注文は、有効期間付きライセンスタイプのみとなります。

スマートアカウントでの発注

スマートアカウントを使用してデバイスとライセンスを注文することをお勧めします。スマートアカウントでは、一元化された 1 つの Web サイトから、スイッチ、ルータ、ファイアウォール、アクセスポイント、ツールのすべてのソフトウェアライセンスを管理できます。スマートアカウントを作成するには、Cisco Smart Software Manager（Cisco SSM）を使用します。

（注） 有効期間付きライセンスの期限切れに関する情報は Cisco SSM の Web サイトを通じてのみ利用可能であるため、これは有効期間付きライセンスを注文する場合に特に役立ちます。

Cisco SSM の詳細については、http://www.cisco.com/c/en/us/buy/smart-accounts/software-licensing.html を参照してください。

スイッチ スタックのライセンスのアクティブ化

LAN Base モデルは、LAN Base モデルのみとスタックできます。

アクティブスタックは、そのアクティブコンソールからライセンスを使用してアクティブ化します。スタック内のメンバーのライセンスレベルも同時にアクティブ化できます。

スタックケーブルが接続されている場合、ライセンスレベルを変更する際に、新たに追加されたスタックメンバーを切断しないでください。代わりに、アクティブコンソールを使用して新しいメンバーのライセンスレベルをアクティブスタックと同じレベルに設定してから、新しいメンバーを再起動すると、新規メンバーがスタックに参加します。

基本ライセンスの場合にのみ再起動が必要です。アドオンライセンスを設定する際には必要ありません。

アドオンライセンスレベルの設定方法

ここでは、アドオンライセンスレベルの設定方法について説明します。

イメージベースのアドオンライセンスのアクティブ化

次の手順を実行すると、イメージベースのライセンスをアクティブ化できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **license boot level addon** *addon-license*
4. **license accept end user agreement force**
5. **show license right-to-use usage**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**license boot level addon** *addon-license* 例： `Device(config)# license boot level addon dna-essentials`	アドオンライセンスレベルを指定します。次のオプションを使用できます。 • DNA Essentials • DNA Advantage
ステップ 4	**license accept end user agreement force** 例： `Device(config)# license accept end user agreement force`	エンドユーザーライセンス契約（EULA）の承認を有効にします。 （注） アドオンライセンス契約（EULA）の承認は必須ではありませんが、この手順を完了するまでは、DNAC機能を使用または設定することはできません。
ステップ 5	**show license right-to-use usage** 例： `Device(config)# show license right-to-use usage`	詳細な使用状況に関する情報を表示します。 **show license right-to-use command**で他のオプションを使用できます。

Cisco Catalyst 3560-CX シリーズでのイメージベースのアドオンライセンスのアクティブ化

次の手順を実行すると、イメージベースのライセンスをアクティブ化できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **license right-to-use** [**activate** | **deactivate**] [**addon** {**dna-essentials** | **dna-advantage** }] {**subscription** | **evaluation**} [**acceptEULA**]
4. **show license right-to-use usage**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**license right-to-use** [**activate**	**deactivate**] [**addon** {**dna-essentials**	**dna-advantage** }] {**subscription**	**evaluation**} [**acceptEULA**] 例： `Device(config)# license right-to-use activate ipbase acceptEULA` `Device(config)# license right-to-use activate addon dna-essentials subscription acceptEULA`	スイッチで指定されたライセンスレベルをアクティブ化し、エンドユーザーライセンス契約（EULA）への同意を有効にします。 アドオンライセンスを構成するために、EULA への同意は必須ではありません。
ステップ 4	**show license right-to-use usage** 例： `Device(config)# show license right-to-use usage`	詳細な使用状況に関する情報を表示します。 **show license right-to-use command**で他のオプションを使用できます。

ライセンスの再ホスト

ライセンスを再ホストするには、1 つのデバイスのライセンスを非アクティブ化し、別のデバイスで同じライセンスをアクティブ化します。次の手順を使用して、ライセンスを再ホストできます。

手順の概要

1. **enable**
2. **configure terminal**
3. **license right-to-use deactivate [license-level] slot***slot-num*
4. **license right-to-use activate [license-level]***slot-num* [**acceptEULA**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**license right-to-use deactivate [license-level] slot***slot-num* 例： `Device(config)# license right-to-use deactivate dna-essentials slot 1`	1 つのデバイスのライセンスを非アクティブ化します。
ステップ 4	**license right-to-use activate [license-level]***slot-num* [**acceptEULA**] 例： `Device(config)# license right-to-use activate dna-essentials slot 2`	別のデバイスのライセンスをアクティブ化します。

ライセンスのモニタリング

ライセンス情報をモニタリングするには、特権 EXEC モードで次のコマンドを使用します。

コマンド	目的
show license right-to-use default	デフォルトのライセンス情報を表示します。
show license right-to-use detail	スイッチ スタック内のすべてのライセンスの詳細情報を表示します。
show license right-to-use eula	エンドユーザ ライセンス契約を表示します。
show license right-to-use slot slot-number	スイッチ スタック内の特定のスロットのライセンス情報を表示します。

コマンド	目的
show license right-to-use summary	スイッチ スタック全体のライセンス情報の要約を表示します。
show license right-to-use usage [slot slot-number]	スイッチ スタック内のすべてのライセンスの使用状況に関する詳細情報を表示します。

ライセンスレベルの設定例

ここでは、ライセンスレベルの設定例を示します。

参照先

-

例：ライセンスの詳細情報の表示

次に、**show license right-to-use detail** コマンドを使用してスタック内にあるすべてのライセンスの詳細情報を表示する例を示します。

```
Device# show license right-to-use detail
Index 1
  License Name    : Advanced Enterprise Services
  Period left     : Lifetime
  License Type    : permanent
  License State   : Active, In use
Index 2
  License Name    : dna-essentials
  Period left     : CSSM Managed
  License Type    : Subscription
  License State   : Not Activated
Index 3
  License Name    : dna-advantage
  Period left     : CSSM Managed
  License Type    : Subscription
  License State   : Active, In use
```

例：ライセンスの要約情報の表示

次に、**show license right-to-use summary** コマンドを使用して、ライセンスの要約情報を表示する例を示します。

```
Device# show license right-to-use summary
License Name                  Type          Period left
-------------------------------------------------------
lanlite                       Permanent     0  minute  0  second
lanbase                       Permanent     0  minute  0  second
dna-essentials                Subscription  CSSM Managed
-------------------------------------------------------
```

```
License Level In Use: lanbase  addon: dna-essentials
License Level on Reboot: lanbase  addon: dna-essentials

Example: show license right-to-use usage

FEX-0#show license right-to-use usage
slot         License Name                  Type          In-use   EULA
------------------------------------------------------------------------
0            lanlite                       Permanent     yes      yes
0            lanbase                       Permanent     yes      yes
             dna-essentials                Subscription  yes      yes
             dna-advantage                 Subscription  no       yes
```

例：エンドユーザーライセンス契約の表示

次に、エンドユーザーライセンス契約を表示する例を示します。

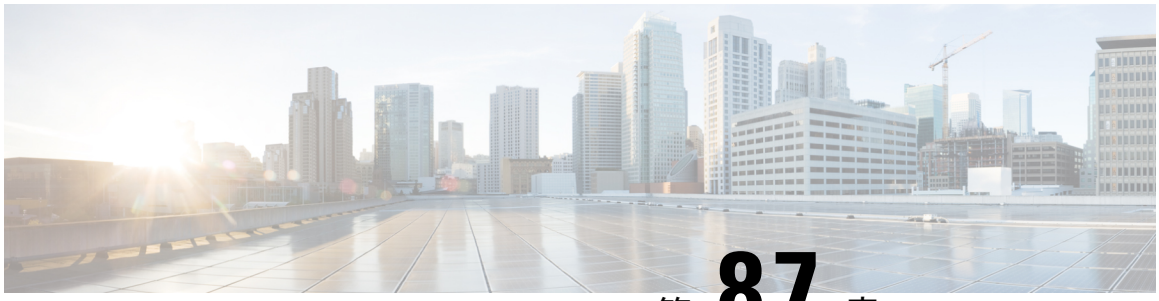
```
Device# show license right-to-use eula subscription
Feature name             EULA Accepted
------------             -------------
dna-essentials           yes
dna-advantage            no
PLEASE READ THE FOLLOWING TERMS  CAREFULLY. INSTALLING THE LICENSE OR
LICENSE KEY PROVIDED FOR ANY CISCO SOFTWARE PRODUCT, PRODUCT FEATURE,
AND OR SUBSEQUENTLY PROVIDED SOFTWARE FEATURES (COLLECTIVELY, THE ?SOFTWARE?),
USING SUCH SOFTWARE, AND/OR ACTIVATION OF THE SOFTWARE COMMAND LINE INTERFACE
CONSTITUTES  YOUR  FULL ACCEPTANCE  OF  THE FOLLOWING TERMS.YOU MUST NOT PROCEED
FURTHER IF YOU ARE NOT WILLING TO BE BOUND BY ALL THE TERMS SET FORTH HEREIN.

Your use of the Software is subject to the Cisco End User License Agreement (EULA)
and any relevant supplemental terms (SEULA) found at
http://www.cisco.com/c/en/us/about/legal/cloud-and-software/software-terms.html.
You hereby acknowledge and  agree  that certain Software and/or  features are licensed
for a particular term, that the license to such Software and/or features is valid only
for the  applicable term and that such  Software and/or features  may be shut down or
otherwise terminated by Cisco after expiration of the applicable license term  (e.g.,
90-day trial period). Cisco reserves the right to terminate any such Software feature
electronically or by any other means available. While Cisco may provide alerts, it is
your  sole  responsibility to  monitor your usage of any such term Software feature to
ensure that your systems and networks are prepared for a shutdown of the Software feature.
To memorialize your acceptance of these terms and activate your license to use the
Software,
please execute the command "license accept end user agreement force".
```

ライセンスの機能の履歴

リリース	変更内容
Cisco IOS リリース 15.2(6)E1	この機能が導入されました。

第 XI 部

組み込まれている Event Manager



第 87 章

Embedded Event Manager Overview

Embedded Event Manager（EEM）は、イベント検出と回復を Cisco IOS 内部で直接行うための分散型でカスタマイズされた手法です。EEM はイベントをモニターし、モニター対象イベントが発生したり、しきい値に達したりすると、情報提供や訂正などの必要な EEM 処理を実行します。EEM ポリシーは、イベントおよびイベントが発生した場合に行う処理を定義するエンティティです。

この章では、EEM の技術的概要を説明します。EEM は単体でも使用できますが、ネットワークのモニターとメンテナンスのための他のネットワーク管理テクノロジーと合わせて使用することもできます。EEM の実装を開始する前に、このモジュールに示す情報を理解することが重要です。

Embedded Event Manager について

組み込まれている Event Manager

従来、イベント トラッキングおよびイベント管理はネットワーキング デバイスの外部のデバイスによって実施されてきました。Embedded Event Manager（EEM）は、イベント管理を Cisco IOS デバイス内で直接実施できるように設計されました。障害によってはデバイスと外部ネットワーク管理デバイスとの通信が損なわれることがあるため、デバイス外ですべてのイベント管理ができるわけではないことから、EEM のデバイス上での予防的なイベント管理機能は有用です。このような状況でデバイスの状態をキャプチャすることは、迅速な回復アクションの実行、および根本原因の分析実施のための情報収集に非常に役立ちます。ルーティング デバイスを完全にリブートすることなしに自動回復アクションが実施されれば、ネットワーク可用性も向上します。

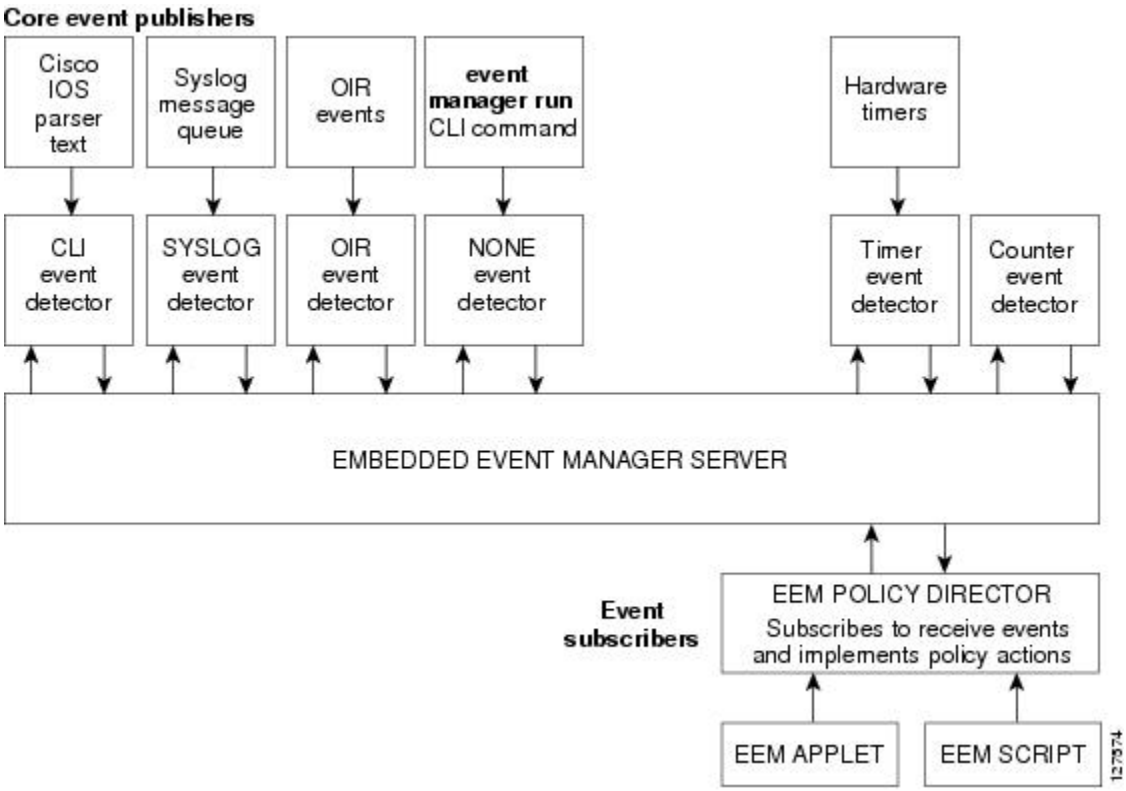
EEM は、イベント ディテクタと呼ばれるソフトウェア エージェントを使用してシステム内の異なるコンポーネントのモニターリングをサポートする、柔軟でポリシードリブンのフレーム

組み込まれている Event Manager

ワークです。次の図に、EEMサーバー、コアイベントパブリッシャ（イベントディテクタ）、およびイベントサブスクライバ（ポリシー）の関係を示します。基本的に、イベントパブリッシャはイベントをスクリーニングして、イベントサブスクライバから提供されたイベント仕様に一致したときにイベントをパブリッシュします。イベントディテクタは、注目するイベントが発生したときにEEMサーバーに通知します。Ciscoコマンドラインインターフェイス（CLI）を使用して設定されたEEMポリシーは、現在のシステムの状態と、該当するイベントのポリシーで指定されたアクションに基づいて回復を実施します。

EEMでは、イベントをモニターし、イベント発生が検出されたとき、およびしきい値を超えたときに情報通知や是正アクションを実施できます。EEMポリシーは、イベントおよびイベントが発生した場合に行う処理を定義するエンティティです。EEMポリシーにはアプレットとスクリプトの2つのタイプがあります。アプレットは、CLI設定に定義された、ポリシーの単純な形式です。スクリプトは、Tool Command Language（Tcl）で記述されたポリシーの形式です。

図 *131 : Embedded Event Manager* コア イベント ディテクタ



（注） ネットワークにEEMの上位バージョンがある場合、そのバージョンは以前のリリースのEEMバージョンを含みます。

Embedded Event Manager 1.0

EEM 1.0 は次のイベント ディテクタを追加しました。

- SNMP：簡易ネットワーク管理プロトコル（SNMP）イベント ディテクタによって、標準 SNMP MIB オブジェクトを監視し、オブジェクトが指定された値と一致するとき、または指定されたしきい値を超えたときにイベントを生成することができます。
- Syslog：syslog イベント ディテクタは、正規表現パターン マッチに対して syslog メッセージをスクリーニングできます。

EEM 1.0 は、次のアクションを追加しました。

- 優先化された syslog メッセージの生成。
- Cisco Networking Services（CNS）デバイスによるアップストリーム処理に対し CNS イベントの生成。
- シスコのソフトウェアをリロードします。
- 完全冗長ハードウェア構成におけるセカンダリ プロセッサへのスイッチング。

Embedded Event Manager 2.0

EEM 2.0 で、いくつかの新機能が導入されました。EEM 2.0 では次のイベント ディテクタが追加されました。

- Application-Specific：Application-Specific イベント ディテクタによって、Embedded Event Manager ポリシーはイベントをパブリッシュできます。
- Counter：Counter イベント ディテクタは、名前付きカウンタが指定されたしきい値を超えたときにイベントをパブリッシュします。
- Interface Counter：Interface Counter イベント ディテクタは、指定されたインターフェイスの汎用 Cisco IOS インターフェイス カウンタが定義されたしきい値を超えたときにイベントをパブリッシュします。
- Timer：Timer イベント ディテクタは、absolute-time-of-day、countdown、watchdog、および CRON の 4 種類のタイマーのイベントをパブリッシュします。
- Watchdog System Monitor（IOSWDSysMon）：Cisco IOS Watchdog System Monitor イベント ディテクタは、Cisco IOS プロセスの CPU またはメモリの使用率がしきい値を超えたときにイベントをパブリッシュします。

EEM 2.0 では次のアクションが追加されました。

- 名前付きカウンタの設定または変更。
- アプリケーション特有のイベントのパブリッシュ
- SNMP トラップの生成。

Tool Command Language（Tcl）を使用して記述された、Cisco 定義のサンプル ポリシー実行機能が追加されました。システム ポリシー ディレクトリに格納可能なサンプル ポリシーが提供されました。

Embedded Event Manager 2.1

EEM 2.1 は次の新しいイベント ディテクタを追加しました。

- CLI：CLI イベント ディテクタは、正規表現と一致するコマンドライン インターフェイス（CLI）コマンドをスクリーニングします。
- None：None イベントディテクタは、Cisco IOS **event manager run** コマンドが EEM ポリシーを実行したときに、イベントをパブリッシュします。
- OIR：Online Insertion and Removal（OIR）イベント ディテクタは、特定のハードウェアの挿入または削除のイベント発生時にイベントをパブリッシュします。

EEM 2.1 は次のアクションを追加しました。

- Cisco CLI コマンドの実行。
- イベント発生時のシステム情報要求。
- ショート メールの送信。
- 手動による EEM ポリシーの実行。

EEM 2.1 は、新しい **event manager scheduler script** コマンドを使用した、複数の同時実行ポリシーの実行も許可します。SNMP イベント ディテクタ比率ベース イベントのサポートは、Tool Command Language（Tcl）を使用してポリシーを作成する機能として提供されます。

Embedded Event Manager 2.1（ソフトウェア モジュール方式）

EEM 2.1（ソフトウェア モジュール）は、Cisco ソフトウェア モジュラリティ イメージでサポートされます。EEM 2.1（ソフトウェア モジュール方式）は、次のイベント ディテクタを追加しました。

- GOLD：Generic Online Diagnostics（GOLD）イベント ディテクタは、GOLD 障害イベントが指定されたカードおよびサブカードで検出されたときにイベントをパブリッシュします。
- System Manager：System Manager イベント ディテクタは、Cisco IOS ソフトウェア モジュール方式プロセスの開始、通常停止、異常停止、および再起動のイベントに対してイベントを生成します。System Manager によって生成されたイベントによって、ポリシーはプロセス再起動のデフォルトの動作を変更できます。
- Watchdog System Monitor（WDSysMon）：Cisco Software Modularity Watchdog System Monitor イベント ディテクタは、Cisco IOS ソフトウェア モジュール方式プロセスにおける無限ループ、デッドロック、メモリリークを検出します。

EEM 2.1 ソフトウェア モジュール方式では、プロセスに対する EEM 信頼性メトリック データの表示機能が追加されました。

（注） EEM 2.1 ソフトウェア モジュール方式イメージは、Resource イベント ディテクタおよび RF イベント ディテクタを EEM 2.2 に追加しましたが、EOT イベント ディテクタ、またはトラッキング対象オブジェクトの読み込みおよび設定のアクションをサポートしません。

Embedded Event Manager 2.2

EEM 2.2 で、いくつかの新機能が導入されました。EEM 2.2 では次のイベント ディテクタが追加されました。

- Enhanced Object Tracking：Enhanced Object Tracking イベント ディテクタは、トラッキング対象オブジェクトが変更されたときにイベントをパブリッシュします。拡張オブジェクト トラッキングは、トラッキング対象オブジェクトと、トラッキング対象オブジェクトが変更されたときにクライアントが実施するアクションとを全面的に分離します。
- Resource：Resource イベント ディテクタは、Embedded Resource Manager（ERM）が、指定されたポリシーのイベントをレポートしたときにイベントをパブリッシュします。
- RF：Redundancy Framework（RF）イベント ディテクタは、デュアル ルートプロセッサ（RP）システムにおける同期の間に、1 つ以上の RF イベントが発生したときにイベントをパブリッシュします。RF イベント ディテクタは、デュアル RP システムが一方の RP からもう一方の RP に継続的にスイッチしている（ピンポン状態と呼ばれる）ときもイベントを検出できます。

EEM 2.2 では次のアクションが追加されました。

- トラッキング対象オブジェクトの状態の読み取り。
- トラッキング対象オブジェクトの状態の設定。

Embedded Event Manager 2.3

EEM 2.3 は、Cisco Catalyst 6500 シリーズ スイッチでサポートされ、その製品での汎用オンライン診断（GOLD）イベント ディテクタのための拡張が追加されています。

- **event gold** コマンドは、GOLD テスト失敗および条件への対応を改善するための **action-notify**、**testing-type**、**test-name**、**test-id**、**consecutive-failure**、**platform-action**、および **maxrun** キーワードが追加され、拡張されました。
- 次のプラットフォーム全体の GOLD イベント ディテクタ情報には、新しい読み込み専用 EEM 組み込み環境変数を通じてアクセスできます。
  - 起動診断レベル

- カード インデックス、名前、シリアル番号
- ポート数
- テスト数

- 次のテスト固有 GOLD イベント ディテクタ情報は、新しい読み込み専用 EEM 組み込み環境変数（EEM アプレットだけが利用可能）を通じてアクセスできます。
  - テスト名、属性、総実行回数
  - テストごと、ポートごと、またはデバイスごとのテスト結果
  - 合計障害カウント、最終障害時間
  - エラー コード
  - 連続的障害の発生

これらの拡張の結果、オートメーションと障害検出が改善され、平均修復時間（MTTR）が削減され、可用性が向上しました。

Embedded Event Manager 2.4

EEM 2.4 は次のイベント ディテクタを追加しました。

- SNMP Notification：SNMP 通知イベント ディテクタには、デバイスが受信した SNMP トラップおよび SNMP インフォーム メッセージを代行受信する機能があります。SNMP 通知イベントは、受信 SNMP トラップまたは SNMP インフォーム メッセージが指定された値に一致するか、指定されたしきい値を超えたときに生成されます。
- RPC：リモート プロシージャ コール（RPC）イベント ディテクタには、EEM ポリシーをセキュア シェル（SSH）を使用して暗号化された接続経由でデバイスの外から起動する機能があります。RPC イベント ディテクタは、XML ベースのメッセージ交換に Simple Object Access Protocol（SOAP）データ エンコーディングを使用します。このイベント ディテクタは、EEM ポリシーの実行および SOAP XML フォーマット化された応答内の出力の受信に使用できます。

EEM 2.4 は、次のイベント ディテクタに拡張を追加しました。

- インターフェイス カウンタ 比率ベース トリガー：この機能によって、インターフェイス イベントが期間中の変更の比率に基づいてトリガーされる機能が追加されました。entry 値または exit 値の両方に対して比率が指定できます。この機能は、現在、SNMP イベント ディテクタに存在する比率ベースの機能をコピーします。
- SNMP デルタ値：モニターリング期間の開始時のモニター対象オブジェクト識別子（OID）の値と、イベントがパブリッシュされた時点での実際の OID の差が、SNMP イベント ディテクタとインターフェイス カウンタ イベント ディテクタの両方の **event reqinfo** データで提供されます。

EEM 2.4 は次のアクションを追加しました。

- 複数イベントのサポート：複数のイベントを実行する機能が導入されました。さらに、**show event manager** コマンドは複数のイベントを表示するように拡張されました。

- パラメータのサポート：パラメータ引数が **event manager run** コマンドに追加されました。最大 15 個のパラメータを使用できます。
- ジョブ ID と完了ステータスの表示：**show event manager** のコマンドの一部が、ジョブ ID と完了ステータスを表示するように拡張されました。
- バイトコードのサポート：Tcl 8 は、特殊なバイトコード言語（BCL）を定義し、Tcl スクリプトを BCL に変換する Just-In-Time コンパイラを備えています。バイト シーケンスが「virtual machine」、Tcl_ExecuteByteCode()、または TEBC によって、必要に応じて短縮して実行されます。現在、EEM は、ユーザー ポリシーのファイル拡張子として、*.tcl を、また、システム ポリシーのファイル拡張子として *.tm を認めています。bytecode スクリプトの Tcl 標準拡張子は、*.tbc です。現在、EEM は *.tbc を有効な EEM ポリシーとして認めます。
- 登録置換拡張：単一の環境変数によるイベント登録文での複数のパラメータの置換をサポートします。
- Tcl パッケージのサポート

Embedded Event Manager 3.0

EEM 3.0 は次の新しいイベント ディテクタを追加しました。

- Custom CLI：Custom CLI イベント ディテクタは、既存の CLI コマンド構文を追加、拡張するためにイベントをパブリッシュします。
- Routing：Routing イベント ディテクタは、ルーティング情報ベース（RIB）のルート エントリが変化したときにイベントをパブリッシュします。
- NetFlow：NetFlow イベント ディテクタは、NetFlow イベントがトリガーされたときにイベントをパブリッシュします。
- IP SLA：IP SLA イベント ディテクタは、IP SLA 応答がトリガーされたときにイベントをパブリッシュします。

EEM 3.0 では、次の新機能が導入されました。

- クラスベース スケジューリング：EEM ポリシーは、登録されるときに **class** キーワードを使用してクラスが割り当てられます。クラスなしで登録された EEM ポリシーは、デフォルト クラスに割り当てられます。
- 高パフォーマンス Tcl ポリシー：**event_completion**、**event_wait**、および **event_completion_with_wait** の 3 つの新しい Tcl コマンドが導入されました。
- インタラクティブ CLI サポート：同期アプレットが、ローカル コンソール（TTY）とのインタラクションをサポートするように拡張されました。2 つの新しい IOS コマンド、**action gets** と **action puts,** が導入され、ユーザーがコンソールに直接入力し、それを表示できるようになりました。

- アプレット用の可変ロジック：EEM アプレット用の可変ロジック機能は、EEM アプレット内に条件付きロジックを適用する機能を追加します。条件付きロジックは、アプレット内のアクションのフローを条件式に従って変更する制御構造を追加します。
- デジタル署名サポート：新しい API は、Tcl スクリプトの実行の前に、スクリプトが Cisco によって署名されていることを確認するために、Tcl スクリプトのデジタル署名検証を実行します。
- 電子メールサーバーの認証のサポート：オプションのユーザー名とパスワードを含めるように、**action mail** コマンドが変更されました。
- SMTP IPv6 サポート：Tcl 電子メールテンプレートに、IPv6 または IPv4 アドレスのいずれかを指定するためのキーワード **sourceaddr** が追加されました。
- SNMP ライブラリの機能拡張：EEM アプレット **action info** と Tcl **sys_reqinfo_snmp** コマンドが拡張され、SNMP getid、inform、trap、および set-type 動作用の機能が組み込まれました。
- SNMP 通知 IPv6 サポート：送信元 IP アドレスと宛先 IP アドレスでの IPv6 アドレスがサポートされます。
- CLI Library XML-PI サポート：異なるシスコ製品間で矛盾のない方法で、IOS コマンドライン インターフェイス（CLI）show コマンドを XML 形式にカプセル化した、プログラム可能なインターフェイスを提供します。XML-PI を使用する場合は、既知のキーワードを使用して IOS show コマンドの出力を Tcl スクリプトから解析できます。正規表現サポートを使用する必要はありません。

Embedded Event Manager 3.1

EEM 3.1 は、新しいイベント ディテクタを追加しました。

- SNMP Object：簡易ネットワーク管理プロトコル（SNMP）Object Trap イベント ディテクタは、指定された SNMP オブジェクト ID（OID）を持つ SNMP トラップが特定のインターフェイスまたはアドレスで発生したときに、値を置き換えるように拡張されました。

EEM 3.1 は、次のイベント ディテクタに拡張を追加しました。

- SNMP Notification：SNMP 通知イベント ディテクタは、出力 SNMP トラップおよび SNMP インフォームを待ち、代行受信できるようになりました。

EEM 3.1 は、次のアクションに拡張機能を追加しました。

- ファシリティの指定：**action syslog** コマンドが拡張され、syslog ファシリティを指定できるようになりました。

EEM 3.1 は、次の機能を追加しました。

- 登録されたポリシーの簡単な説明を作成するための機能を提供：ポリシーを簡単な説明とともに Cisco IOS CLI と Tcl ポリシーに登録するための新しい **description** コマンドが導入されました。**show event manager policy available** コマンドと **show event manager policy**

registered コマンドが拡張され、登録されたアプレットの説明を表示するための **description** キーワードが追加されました。

- EEM ポリシーでの AAA 認証のバイパスが可能：**event manager application** コマンドが拡張され、承認を提供し、AAA を無効にするキーワードをバイパスできるようになりました。
- CLI ライブラリ拡張機能の導入：CLI ライブラリに 2 つの新しいコマンドが提供されました：**cli_run** および **cli_run_interactive**。

Embedded Event Manager 3.2

EEM 3.2 は次の新しいイベント ディテクタを追加しました。

- ネイバー探索：ネイバー探索イベントディテクタによって、次の場合に自動ネイバー検出に応答するポリシーをパブリッシュできます。
  - Cisco Discovery Protocol（CDP）のキャッシュ エントリが追加、削除、または更新された場合。
  - リンク層検出プロトコル（LLDP）のキャッシュ エントリが追加、削除、または更新された場合。
  - インターフェイスのリンク ステータスが変更された場合。
  - インターフェイスのライン ステータスが変更された場合。
- ID：ID イベントディテクタは、AAA の許可および認証が成功した場合、障害が発生した場合、またはポート上で通常のユーザー トラフィックの送信が許可された後にイベントを生成します。
- Mac-Address-Table：Mac-Address-Table イベント ディテクタは、MAC アドレスが MAC アドレス テーブルで学習された場合にイベントを生成します。

（注） Mac-Address-Table イベント検出器は、スイッチ プラットフォームでだけサポートされており、MAC アドレスが学習されたレイヤ 2 インターフェイスだけで使用できます。レイヤ 3 インターフェイスはアドレスを学習しません。デバイスは通常、学習された MAC アドレスの EEM を通知する必要がある mac-address-table インフラストラクチャをサポートしません。

EEM 3.2 では、新しいイベントディテクタで動作するアプレットをサポートするための新しい CLI コマンドも導入されています。

Embedded Event Manager 4.0

EEM 4.0 では、次の新機能が導入されます。

- EEM 電子メール アクションの機能

- SMTP 電子メールアクションの TLS サポート：新しいオプションの **secure** キーワードが、**tls** および **none** キーワードとともに、**action mail** CLI に追加されました。対応する Tcl ポリシーには更新はありません。
- SMTP 電子メールアクションのカスタムポート：新しいオプションの **port** キーワードが **action mail** CLI に追加されました。Tcl ポリシーでは、電子メール テンプレートに行を追加することでポート番号を指定できます。

- EEM セキュリティの機能拡張
  - チェックサムベースのスクリプトの整合性：デジタル署名がサポートされていない、または使用できない場合、ユーザーは引き続き Unix コマンド **openssl sha1** を使用して、TCL ポリシーにいくつかの基本的な整合性チェックを適用できます。新しいオプションの **checksum**、**md5**、および **sha-1** キーワードが **event manager policy** コマンドに追加されました。
  - サードパーティのデジタル署名のサポート：署名を確認するには、Tcl セキュア モードとトラストポイントを TCL スクリプトに関連付ける必要があります。
  - スクリプト所有者の識別：ポリシーが正常にデジタル署名に登録されている場合は、**show event manager policy registered** コマンドを使用して、show 出力で **Dsig** キーワードを確認することで、ポリシーの所有者（または署名者）を識別できます。
  - リモート Tcl ポリシーの登録：新しいオプションの **remote** キーワードが **event manager policy** コマンドに追加されました。

- EEM リソース管理
  - リソース消費量のスロットリング：新しいオプションの **resource-limit** キーワードが **event manager scheduler** コマンドに追加されました。
  - イベントごとにトリガーされるポリシーのレート制限：新しいオプションの **rate-limit** キーワードが **event syslog** コマンドに追加されました。

- EEM ユーザービリティの機能拡張
  - EEM アプレットアクションでのファイル操作：新しい CLI **action file** が追加され、ファイルを選択できるようになりました。
  - **show event manager statistics** EXEC コマンドを使用して、キューサイズ、ドロップされたイベント、および実行時間の統計情報を追跡するために、EEM に新しいフィールドが追加されました。イベントマネージャのキューカウンタをクリアするために、一連の新しい clear コマンド（**clear event manager detector counters** および **clear event manager server counters**）が導入されました。

- EEM イベント ディテクタの機能拡張
  - CLI イベント ディテクタの機能拡張：ユーザーがイベント CLI コマンドを入力したセッションを検出する機能を提供します。4 つの新しいキーワードと組み込み環境変数：**username**、**host**、**privilege**、および **tty** が **event cli** アプレットに、event_reqinfo アレイ名が **event_register_cli** イベントディテクタに追加されました。**show event manager detector** EXEC コマンドも、この機能強化を反映するように変更されました。
  - Syslog イベント ディテクタの機能拡張：特定のログ メッセージ フィールドで文字列の照合を実行するオプションを提供します。4 つの新しいキーワード（**facility**、

mnemonic、**sequence**、および **timestamp** キーワード）が、**action syslog** コマンド、**event syslog** コマンド、および **event_register_syslog** イベントディテクタに追加されました。**show event manager detector** EXEC コマンドも、この機能強化を反映するように変更されました。

Cisco IOS Release ごとの利用可能な EEM イベント ディテクタ

EEM は、イベントディテクタと呼ばれるソフトウェア プログラムを使用して、EEM イベントの発生したときを判断します。一部のイベントディテクタは、すべての Cisco IOS Release で利用できますが、イベントディテクタの多くは、特定のリリースに導入されています。次の表を使用して、特定の Cisco IOS リリースで使用可能なイベント ディテクタを特定します。ブランク エントリ（--）は、そのイベント ディテクタが利用できないことを示します。「Yes」の文字はイベント ディテクタが利用できることを示します。この表に示されているイベント ディテクタは、同じ Cisco IOS リリース トレインの最新のリリースでサポートされています。各イベントディテクタの詳細については、「Embedded Event Manager の概要」の章のイベントディテクタの概念を参照してください。

表 *175 : Cisco IOS Release* ごとのイベント ディテクタの可用性

イベント ディテクタ	12.2(25)S	12.3(14)T 12.2(18)SXF5 12.2(28)SB 12.2(33)SRA	12.4(2)T 12.2(31)SB3 12.2(33)SRB	12.2(18)SXF4 Cisco IOS ソフトウェアモジュール方式	12.2(33)SXH	12.4(20)T 12.2(33)SXI	12.4(22)T 12.2(33)SRE	15.0(1)M 15.1(3)T	15.1(2)SY	15 E XE 3E
Application-Specific	対応	対応	対応	対応	対応	対応	対応	対応	対応	対応
CLI	--	対応	対応	対応	対応	対応	対応	対応	--	可
Counter	対応	対応	対応	対応	対応	対応	対応	対応	対応	対応
Custom CLI	--	--	--	--	--	--	対応	対応	--	--
Enhanced Object Tracking	--	--	可	--	対応	対応	対応	対応	--	--
Environmental	--	--	--	--	--	--	--	--	--	可
GOLD	--	--	--	対応	対応	対応	対応	対応	--	可
Identity	--	--	--	--	--	--	--	対応	対応	対応
Interface Counter	対応	対応	対応	対応	対応	対応	対応	対応	--	可
IPSLA	--	--	--	--	--	--	対応	対応	--	可
Mac-Address-Table	--	--	--	--	--	--	--	対応	対応	対応

イベント ディテクタ	12.2(25)S	12.3(14)T 12.2(18)SXF5 12.2(28)SB 12.2(33)SRA	12.4(2)T 12.2(31)SB3 12.2(33)SRB	12.2(18)SXF4 Cisco IOS ソフトウェアモジュール方式	12.2(33)SXH	12.4(20)T 12.2(33)SXI	12.4(22)T 12.2(33)SRE	15.0(1)M 15.1(3)T	15.1(2)SY	15 E XE 3E
Neighbor Discovery	--	--	--	--	--	--	--	対応	対応	対応
NF	--	--	--	--	--	--	対応	対応	--	--
なし	--	対応	対応	対応	対応	対応	対応	対応	対応	対応
OIR	--	対応	対応	対応	対応	対応	対応	対応	対応	対応
Resource	--	--	対応	対応	対応	対応	対応	対応	--	--
RF	--	--	対応	対応	対応	対応	対応	対応	--	可
Routing	--	--	--	--	--	--	対応	対応	--	可
RPC	--	--	--	--	--	対応	対応	対応	対応	--
SNMP	対応	対応	対応	対応	対応	対応	対応	対応	--	可
SNMP Proxy	--	--	--	--	--	--	--	--	可	--
SNMP Notification	--	--	--	--	--	対応	対応	対応	--	可
SNMP Object	--	--	--	--	--	--	--	可	--	可
Syslog	対応	対応	対応	対応	対応	対応	対応	対応	対応	対応
System Manager	--	--	--	対応	対応	対応	対応	対応	対応	--
Timer	対応	対応	対応	対応	対応	対応	対応	対応	対応	対応
IOSWDSysMon（Cisco IOS Watchdog）	対応	対応	対応	対応	対応	対応	対応	対応	--	可
WDSysMon（Cisco IOS Software Modularity Watchdog）	--	--	--	可	--	--	--		--	--

イベント検出器

Embedded Event Manager（EEM）は、イベント ディテクタと呼ばれるソフトウェア プログラムを使用して、EEMイベントの発生したときを判断します。イベントディテクタは、モニターされるエージェント（たとえば、簡易ネットワーク管理プロトコル（SNMP））と、アクションが実施される EEM ポリシーの間のインターフェイスを提供する、独立したシステムです。一部のイベント ディテクタは、すべての Cisco IOS Release で利用できますが、イベント ディテクタの多くは、特定のリリースに導入されています。各 Cisco IOS Release でサポートされるイベント ディテクタの詳細については、「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」または「Tcl を使用した Embedded Event Manager ポリシーの記述」の章の Cisco IOS Release ごとの利用可能な EEM イベント ディテクタについての記述を参照してください。EEM には次のイベント ディテクタがあります。

Application-Specific イベント ディテクタ

Application-Specific イベント ディテクタによって、任意の Embedded Event Manager ポリシーがイベントをパブリッシュできます。EEM ポリシーがイベントをパブリッシュするとき、任意のイベントタイプで、EEMサブシステム番号798を使用する必要があります。既存のポリシーがサブシステム798と指定されたイベントタイプに対して登録されている場合、同じイベントタイプの別のポリシーは、指定されたイベントがパブリッシュされたときに第1のポリシーをトリガーして実行します。

CLI イベント ディテクタ

CLIイベントディテクタは、コマンドライン インターフェイス（CLI）コマンドを正規表現に一致するかスクリーニングします。一致が見つかったとき、イベントがパブリッシュされます。コマンドが正常に解析されたあと、コマンドが実施される前に、完全に展開されたCLIコマンドで一致ロジックが実施されます。CLIイベントディテクタは次の3種類のパブリッシュモードをサポートします。

- CLIイベントの同期パブリッシング：CLIコマンドは、EEMポリシーが終了するまで実行されません。EEMポリシーは、コマンドが実行されるかどうかをコントロールできます。読み取り/書き込み変数 _exit_status では、ポリシー 終了時に同期イベントからトリガーされたポリシーの終了ステータスを設定できます。_exit_status が 0 の場合、コマンドはスキップされ、_exit_status が 1 の場合はコマンドが実行されます。
- CLI イベントの非同期パブリッシング：CLI イベントは、パブリッシュされ、続いて CLI コマンドが実行されます。
- CLI イベントの非同期パブリッシングかつコマンド スキップ：CLI イベントがパブリッシュされますが、CLI コマンドは実行されません。

Counter イベント ディテクタ

Counter イベント ディテクタは、名前付きカウンタが指定されたしきい値を超えたときにイベントをパブリッシュします。カウンタ処理に影響を与える関係タスクが 2 つ以上あります。Counterイベントディテクタは、カウンタを変更でき、1つ以上のサブスクライバは、イベント

をパブリッシュする条件を定義します。カウンタイベントがパブリッシュされた後、カウンタモニターリングロジックをリセットして、すぐにカウンタの監視を開始できます。また、別のしきい値（exit 値と呼ばれる）を超えたときにリセットすることもできます。

Custom CLI イベント ディテクタ

Custom CLI イベント ディテクタは、既存の CLI コマンド構文を追加、拡張するためにイベントをパブリッシュします。特別なパーサー キャラクタである Tab、?（疑問符）、および Enter が入力された場合、パーサーは処理のために入力を Custom CLI イベント ディテクタに送信します。（疑問符）、および Enter が入力された場合、パーサーは処理のために入力を Custom CLI イベント ディテクタに送信します。続いて Custom CLI イベント ディテクタは、この入力を登録された文字列と比較して、新しい、または拡張された CLI コマンドかどうかを判断します。一致すると、カスタム CLI イベント ディテクタが適切なアクションを実行します。たとえば、? が入力された場合はコマンドのヘルプを表示する、タブが入力された場合はコマンド全体を表示する、Enter が入力された場合はコマンドを実行するなどです。一致しなかった場合は、パーサーはコントロールを回復し、通常どおりに情報を処理します。

Enhanced Object Tracking イベント ディテクタ

Enhanced Object Tracking（EOT）イベント ディテクタは、トラッキング対象のオブジェクトのステータスが変更されたときイベントをパブリッシュします。オブジェクト トラッキングは、当初、ユーザーがインターフェイスのラインプロトコル ステートをトラッキングできるだけの単純なトラッキングメカニズムとして、ホットスタンバイ ルータ プロトコル（HSRP）に導入されました。インターフェイスのラインプロトコル ステートがダウンになった場合、デバイスの HSRP 優先度は削減され、より高い優先度のもう 1 つの HSRP デバイスがアクティブになることができます。

オブジェクト トラッキングはトラッキング対象オブジェクトと、トラッキング対象オブジェクトが変更されたときにクライアントが実施するアクションとを全面的に分離するように拡張されました。したがって、HSRP、仮想ルータ冗長プロトコル（VRRP）、または Gateway Load Balancing Protocol（GLBP）などの複数のクライアントが、トラッキング プロセスの対象を登録でき、同一オブジェクトをトラッキング可能であり、さらに、オブジェクト変更時に異なるアクションを実行できます。各トラッキング対象オブジェクトは、トラッキング コマンドライン インターフェイス（CLI）で指定された一意の番号で識別されます。クライアント プロセスは、この番号を使用して特定のオブジェクトを追跡します。トラッキング プロセスは定期的に、トラッキング対象オブジェクトをポーリングし、値の変更を確認します。トラッキング対象オブジェクトの変更は、すぐに、または指定された遅延後に、対象のクライアント プロセスに通知されます。オブジェクトの値は、アップまたはダウンとして報告されます。

拡張オブジェクト トラッキングが EEM と統合され、EEM は追跡対象オブジェクトのステータス変更を報告して、拡張オブジェクト トラッキングが EEM オブジェクトを追跡できるようになりました。新しいタイプのトラッキング オブジェクト、スタブ オブジェクトが作成されます。現在追跡対象オブジェクトを操作できるようにしている既存の CLI コマンドを使用して、スタブ オブジェクトを操作できます。

Generic Online Diagnostics（GOLD）イベント ディテクタ

GOLD イベント ディテクタは、GOLD 障害イベントが指定されたカードおよびサブカードで検出されたときにイベントをパブリッシュします。

Interface Counter イベント ディテクタ

Interface Counter イベント ディテクタは、指定されたインターフェイスの汎用 Cisco IOS インターフェイス カウンタが、定義されたしきい値を超えたときにイベントをパブリッシュします。しきい値は絶対値か増分値で指定できます。たとえば、増分値を 50 に設定した場合、インターフェイス カウンタが 50 増えると、イベントがパブリッシュされます。

インターフェイス カウンタ イベントがパブリッシュされた後、インターフェイス カウンタ モニターリング ロジックは 2 つの方法でリセットされます。インターフェイス カウンタは、別のしきい値（exit 値と呼ばれる）を超えたとき、または、期間の経過が発生したときにリセットされます。

IP SLA イベント ディテクタ

IP SLA イベント ディテクタは、IP SLA 応答がトリガーされたときにイベントをパブリッシュします。

NetFlow イベント ディテクタ

NetFlowイベントディテクタは、NetFlowイベントがトリガーされたときにイベントをパブリッシュします。

None イベント ディテクタ

None イベントディテクタは、Cisco IOS **event manager run** CLI コマンドが EEM ポリシーを実行すると、イベントをパブリッシュします。EEM は、ポリシーそのものに含まれるイベント仕様に基づいてポリシーをスケジューリングし、実行します。EEM ポリシーは識別される必要があり、手動での実行が許可されるように、**event manager run** コマンドが実行される前に登録される必要があります。

OIR イベント ディテクタ

活性挿抜（OIR）イベントディテクタは、次のハードウェアの挿入または削除のいずれかのイベント発生時にイベントをパブリッシュします。

- カードが削除されました。
- カードが挿入されました。

ルートプロセッサ（RP）、ラインカード、またはフィーチャ カードは、OIR イベントでモニターできます。

Resource イベント ディテクタ

Resource イベント ディテクタは、Embedded Resource Manager（ERM）が指定されたポリシーのイベントをレポートしたときにイベントをパブリッシュします。ERMインフラストラクチャは、プロセス間およびシステム内のリソースの枯渇とリソースの依存関係を追跡し、さまざまなエラー状態を処理します。エラー状態は、さまざまなアプリケーション間でリソースを等分に共有することで処理されます。ERM フレームワークは、リソース エンティティに通信メカニズムを提供して、さまざまなロケーションからこれらのリソースエンティティ間での通知が行えるようにします。ERM フレームワークは、CPU およびメモリ関連の問題のデバッグにも役立ちます。ERM は、CPU、バッファ、およびメモリなどのリソースに対してユーザーがしきい値を設定できるようにすることで、スケーラビリティ ニーズを理解するためにシステム リソース使用率をモニターリングします。ERMイベントディテクタは、Cisco ソフトウェアのリソースを監視するためのより望ましい方法ですが、ERM イベント ディテクタはソフトウェア モジュラリティ イメージをサポートしません。ERM の詳細については、「Embedded Resource Manager」の章を参照してください。

RF イベント ディテクタ

Redundancy Framework（RF）イベント ディテクタは、デュアル ルート プロセッサ（RP）システムにおける同期の間に、1 つ以上の RF イベントが発生したときにイベントをパブリッシュします。RF イベント ディテクタは、デュアル RP システムが一方の RP からもう一方の RP に継続的にスイッチしている（ピンポン状態と呼ばれる）ときもイベントを検出できます。

Remote Procedure Call（RPC）イベント ディテクタ

リモート プロシージャ コール（RPC）イベント ディテクタには、EEM ポリシーをセキュア シェル（SSH）を使用して暗号化された接続経由でデバイスの外から起動する機能があります。RPC イベント ディテクタは、XML ベースのメッセージ交換に Simple Object Access Protocol（SOAP）データ エンコーディングを使用します。このイベント ディテクタは、EEM ポリシーの実行および SOAP XML フォーマット化された応答内の出力の受信に使用できます。

ルーティング イベント ディテクタ

ルーティング イベント ディテクタは、ルーティング情報ベース（RIB）のルート エントリが変化したときにイベントをパブリッシュします。

SNMP イベント ディテクタ

SNMP イベント ディテクタによって、標準 SNMP MIB オブジェクトを監視し、オブジェクトが指定された値と一致するとき、または指定されたしきい値を超えたときにイベントを生成することができます。

SNMP 通知イベント ディテクタ

SNMP 通知イベント ディテクタには、デバイスが受信した SNMP トラップおよび SNMP インフォーム メッセージを代行受信する機能があります。SNMP 通知イベントは、受信または送信 SNMP トラップまたは SNMP インフォーム メッセージが指定された値に一致するか、指定さ

れたしきい値を超えたときに生成されます。SNMPイベントディテクタは、送信SNMPトラップおよびSNMPインフォームを待ち、代行受信できます。

SNMP Object イベント ディテクタ

簡易ネットワーク管理プロトコル（SNMP）Object Trapイベント ディテクタは、指定されたSNMPオブジェクトID（OID）を持つSNMPトラップが特定のインターフェイスまたはアドレスで発生したときに、値を置き換えるように拡張されました。

syslog イベント ディテクタ

syslogイベントディテクタは、正規表現パターンマッチに対してsyslogメッセージをスクリーニングできます。選別されたメッセージをさらに限定し、指定された時間内に特定の回数の発生を記録するように要求できます。指定されたイベント基準での一致により、設定されたポリシー処理がトリガーされます。

System Manager イベント ディテクタ

System Managerイベント ディテクタは、Cisco IOSソフトウェア モジュール方式プロセスの開始、通常停止、異常停止、および再起動のイベントに対してイベントを生成します。System Managerによって生成されたイベントによって、ポリシーはプロセス再起動のデフォルトの動作を変更できます。

Timer イベント ディテクタ

timerイベント ディテクタは、次の4種類のタイマーのイベントをパブリッシュします。

- absolute-time-of-day タイマーは、指定された絶対的な日時が発生したとき、イベントをパブリッシュします。
- countdown タイマーは、タイマーがカウント ダウンしてゼロ（0）になったときにイベントをパブリッシュします。
- watchdog タイマーは、タイマーがカウント ダウンしてゼロ（0）になったときにイベントをパブリッシュし、自動的にタイマーを初期値にリセットして、再びカウントダウンを開始します。
- CRON タイマーは、UNIX標準CRON仕様を使用してイベントをパブリッシュするときを指定して、イベントをパブリッシュします。CRON タイマーは、1分間にイベントを複数回パブリッシュすることはありません。

Cisco IOS の Watchdog System Monitor（IOSWDSysMon）イベント ディテクタ

Cisco IOS Watchdog System Monitorイベント ディテクタは、次のいずれかが発生したときにイベントをパブリッシュします。

- Cisco IOSタスクのCPU使用率がしきい値を超えたとき。
- Cisco IOSタスクのメモリ使用率がしきい値を超えたとき。

（注） Cisco IOS プロセスは、現在、Cisco IOS ソフトウェア モジュール方式プロセスから区別するために、タスクと呼ばれています。

同時に2つのイベントがモニターリングされることがあります。指定されたしきい値を超えるために1つのイベントを必要とするか、両方のイベントを必要とするかを、イベントパブリッシング基準で指定できます。

Cisco IOS Software Modularity の Watchdog System Monitor (WDSysMon) イベント ディテクタ

Cisco IOS Software Modularity Watchdog System Monitor イベント ディテクタは、Cisco IOS ソフトウェア モジュラリティ プロセスにおける無限ループ、デッドロック、メモリ リークを検出します。

各 Cisco IOS リリースで利用可能な EEM アクション

イベントディテクタがイベントを報告したときに実行される是正アクションはCLIベースで、強力なオンデバイスのイベント管理メカニズムを実現します。一部のアクションは、すべてのCisco IOS Release で利用できますが、アクションの多くは、特定のリリースに導入されています。次の表を使用して、特定の Cisco IOS リリースで使用可能なアクションを特定します。ブランク エントリ（--）は、そのアクションが使用できないことを示します。「Yes」のテキストはそのアクションが使用できることを示します。この表に示されているアクションは、同じCisco IOS リリース トレインの最新のリリースでサポートされています。各アクションの詳細については、「Embedded Event Manager Overview」の章の Embedded Event Manager アクションの概念を参照してください。

表 *176 :* 各 *Cisco IOS* リリースで利用可能なアクション

アクション	12.2(25)S	12.3(14)T 12.2(18)SXF5 12.2(28)SB 12.2(33)SRA	12.4(2)T 12.2(31)SB3 12.2(33)SRB	12.2(18)SXF4 Cisco IOS ソフトウェア モジュール方式	12.2(33)SXH	12.4(20)T	12.4(22)T	15.0(1)M	15E XE 3E
CLI コマンドの実行	--	対応	対応	対応	対応	対応	対応	対応	対応
CNS イベントの生成	対応	対応	対応	対応	対応	対応	対応	対応	対応
優先化された syslog メッセージの生成	対応	対応	対応	対応	対応	対応	対応	対応	対応
SNMP トラップの生成	対応	対応	対応	対応	対応	対応	対応	対応	対応
手動による EEM ポリシーの実行	--	対応	対応	対応	対応	対応	対応	対応	対応

アクション	12.2(25)S	12.3(14)T 12.2(18)SXF5 12.2(28)SB 12.2(33)SRA	12.4(2)T 12.2(31)SB3 12.2(33)SRB	12.2(18)SXF4 Cisco IOS ソフトウェア モジュール方式	12.2(33)SXH	12.4(20)T	12.4(22)T	15.0(1)M	15E XE 3E
アプリケーション固有のイベントのパブリッシュ	対応	対応	対応	対応	対応	対応	対応	対応	対応
トラッキング対象オブジェクトの状態の読み取り	--	--	可	--		対応	対応	対応	対応
シスコのソフトウェアのリロード	対応	対応	対応	対応	対応	対応	対応	対応	対応
システム情報の要求	--	対応	対応	対応	対応	対応	対応	対応	対応
ショート メールの送信	--	対応	対応	対応	対応	対応	対応	対応	対応
名前付きカウンタの設定または変更	対応	対応	対応	対応	対応	対応	対応	対応	対応
トラッキング対象オブジェクトの状態の設定	--	--	可	--		対応	対応	対応	対応
セカンダリ RP へのスイッチ	対応	対応	対応	対応	対応	対応	対応	対応	対応

Embedded Event Manager のアクション

イベントディテクタがイベントを報告したときに実行される是正アクションはCLIベースで、強力なオンデバイスのイベント管理メカニズムを実現します。一部の EEM アクションは、すべての Cisco IOS Release で利用できますが、EEM アクションの多くは、特定のリリースに導入されています。各 Cisco IOS Release でサポートされる EEM アクションの詳細については、「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」または「Writing Embedded Event Manager Policies Using Tcl」の章の Cisco IOS Release ごとの利用可能な EEM アクションについての記述を参照してください。EEM がサポートするアクションは、次のとおりです。

- Cisco IOS コマンドライン インターフェイス（CLI）コマンドの実行。
- Cisco CNS デバイスによるアップストリーム処理に対し CNS イベントの生成。
- 名前付きカウンタの設定または変更。
- 完全冗長ハードウェア構成におけるセカンダリ プロセッサへのスイッチング。

- イベント発生時のシステム情報要求。
- ショート メールの送信。
- 手動による EEM ポリシーの実行。
- アプリケーション特有のイベントのパブリッシュ。
- シスコのソフトウェアをリロードします。
- SNMP トラップの生成。
- 優先化された syslog メッセージの生成。
- トラッキング対象オブジェクトの状態の読み取り。
- トラッキング対象オブジェクトの状態の設定。

EEM アクション CLI コマンドには、任意の文字列値が可能で一意の ID である EEM アクション ラベルが含まれます。アクションは、ラベルをソート キーとして使用して、英数字のキーの昇順（辞書順）にソートされ、実行されます。ラベルとして数字を使用している場合は、英数字ソートは、10.0は1.0よりも後ですが、2.0よりも前になることに注意してください。このような場合、01.0、02.0 のような数字を使用する、または頭文字の後に同様の数字を続けることを推奨します。

Embedded Event Manager の環境変数

EEM では、EEM ポリシーに環境変数を使用できます。Tool Command Language（Tcl）では、Tcl スクリプト内のすべてのプロシージャで既知のグローバル変数を定義できます。EEM では、CLI コマンドの **event manager environment** コマンドを使用して、EEM ポリシー内で使用するための環境変数を定義できます。EEM環境変数は、Tclスクリプトの実行前に、Tclグローバル変数に自動的に割り当てられます。Embedded Event Managerに関連する環境変数には次の3 種類があります。

- ユーザー定義：ユーザーが記述したポリシー内の環境変数を作成する場合にユーザーが定義できます。
- シスコ定義：特定のサンプル ポリシーのためにシスコが定義しました。
- シスコ組み込み（EEM アプレット内で利用可能）：シスコが定義し、読み取り専用、または読み取り/書き込み可能です。読み取り専用変数は、アプレットの実行開始前にシステムによって設定されます。単一の読み取りと書き込みの変数 _exit_status では、同期イベントからトリガーされたポリシーの終了ステータスを設定できます。

シスコ定義環境変数（次の表を参照）およびシスコ システム定義環境変数は、1 つの特定イベント ディテクタまたはすべてのイベント ディテクタに適用できます。ユーザー定義の環境変数、またはサンプルポリシーでCiscoによって定義された環境変数は、**event manager environment** コマンドを使用して設定されます。EEM ポリシーで使用される変数は、ポリシーを登録する前に定義する必要があります。Tcl ポリシーには、ポリシーの実行前に必要な環境変数がすべ

て定義されているかどうかを確認するために定義される「Environment Must Define」と呼ばれるセクションがあります。

シスコ組み込み環境変数は、シスコ定義の環境変数のサブセットです。組み込み変数は、EEM アプレットでだけ利用できます。組み込み変数は、読み込み専用であるか、または読み込みおよび書き込み用のいずれかです。これらの変数は、1 個の特定のイベント ディテクタまたはすべてのイベント ディテクタに適用されます。シスコ システム定義変数の詳細と、一覧表については、「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」の章を参照してください。

（注） シスコ定義環境変数は、アンダースコア（_）で始まります。付ける名前の競合を防止するため、ユーザー間での同じ命名規則の使用は避けることを強く推奨します。

次の表に、サンプル EEM ポリシーで使用されるシスコ定義変数の説明を示します。一部の環境変数は、対応サンプル ポリシーで実行のために指定される必要はありません。これらは任意として示されています。

表 *177*：シスコ定義環境変数と例

環境変数	説明	例
_config_cmd1	実行される 1 番めのコンフィギュレーション コマンド。	**interface Ethernet1/0**
_config_cmd2	（任意）実行される 2 番めのコンフィギュレーション コマンド。	**no shutdown**
_crash_reporter_debug	（任意）tm_crash_reporter.tcl のデバッグ情報がイネーブルであるかどうかを決定する値。	1
_crash_reporter_url	クラッシュ レポートが送信される URL 位置。	http://www.yourdomain.com/fm/interface_tm.cgi
_cron_entry	ポリシーが実行されるときを決定する CRON 仕様。cron エントリを指定する方法の詳細については、「Tcl を使用した Embedded Event Manager ポリシーの記述」の章を参照してください。	0-59/1 0-23/1 * * 0-7

環境変数	説明	例
_email_server	Eメール送信に使用されるシンプル メール転送プロトコル（SMTP）メール サーバー。	mailserver.yourdomain.com
_email_to	Eメールの送信先アドレス。	engineer@yourdomain.com
_email_from	Eメールの送信元アドレス。	devtest@yourdomain.com
_email_cc	Eメールのコピーの送信先アドレス。	manager@yourdomain.com
_email_ipaddr	受信者の送信元 IP アドレス。	209.165.201.1 または（IPv6 アドレス）2001:0DB8::1
_info_snmp_oid	SNMP オブジェクト ID。	1.3.6.1.2.1.2 または iso.internet.mgmt.mib-2.interfaces
_info_snmp_value	割り当てられた SNMP データ エレメントの値文字列。	
_show_cmd	ポリシーの実行時に実行される CLI **show** コマンド。	**show version**
_syslog_pattern	ポリシー実行時を決定するために syslog メッセージを比較するために使用する正規表現パターン マッチ文字列。	.*UPDOWN.*FastEthernet 0/0.*
_tm_fsys_usage_cron	（オプション）**event_register** キーワード拡張機能で使用される CRON 仕様。指定されない場合、_tm_fsys_usage.tcl ポリシーが 1 分に 1 回、トリガーされます。	0-59/1 0-23/1 * * 0-7

環境変数	説明	例
_tm_fsys_usage_debug	（任意）この変数が値 1 に設定された場合、システムのすべてのエントリのディスク使用率情報が表示されます。	1
_tm_fsys_usage_freebytes	（任意）システムまたは特定のプレフィックスの空きバイト数しきい値。空きスペースが所定の値を下回ると、警告が表示されます。	disk2:98000000
_tm_fsys_usage_percent	（任意）システムまたは特定のプレフィックスのディスク使用割合しきい値。ディスク使用割合が所定の割合を超えると、警告が表示されます。指定されない場合、すべてのシステムのデフォルトのディスク使用割合は、80% です。	nvram:25 disk2:5

Embedded Event Manager ポリシーの作成

EEM は、Cisco ソフトウェア システムで障害またはその他のイベントが発生したときに EEM ポリシー エンジンが通知を受け取るポリシー ドリブン プロセスです。Embedded Event Manager ポリシーは、システムの現在の状態に基づいて回復を実行し、該当するイベントのポリシーに指定されたアクションを実行します。回復アクションはポリシーが実行されたときにトリガーされます。

いくつかの EEM CLI 設定と **show** コマンドはありますが、EEM はポリシーの作成を通じて実装されます。EEM ポリシーは、イベントおよびイベントが発生した場合に行う処理を定義するエンティティです。EEM ポリシーにはアプレットとスクリプトの 2 つのタイプがあります。アプレットは、CLI 設定に定義された、ポリシーの単純な形式です。スクリプトは、Tcl で記述された、ポリシーの形式です。

EEM ポリシーの作成には次の項目が含まれます。

- ポリシーが実行されるイベントの選択。
- イベントの記録およびイベントへの対応に関連付けられたイベント ディテクタ オプションの定義。

- 必要に応じて、環境変数の定義。
- イベント発生時に実行されるアクションの選択。

EEM ポリシーの作成には 2 つの方法があります。第 1 の方法は、CLI コマンドを使用してアプレットを記述する方法で、第 2 の方法は、Tcl スクリプトを記述する方法です。シスコは、Tcl に EEM ポリシー開発を促進する Tcl コマンド拡張機能を加えました。スクリプトは、ネットワーキング デバイスの外部で ASCII エディタを使用して定義します。続いてスクリプトはネットワーキング デバイスにコピーされ EEM に登録されます。Embedded Event Manager にポリシーが登録されると、ソフトウェアはポリシーを調べ、指定されたイベントの発生時に起動するために登録します。ポリシーは、未登録または中断にできます。両方のタイプのポリシーとも、ネットワークの EEM 実装に使用できます。

Cisco IOS CLI を使用して EEM ポリシーを記述する方法については、「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」の章を参照してください。

Tcl を使用して EEM ポリシーを記述する方法の詳細については、「Writing Embedded Event Manager Policies Using Tcl」の章を参照してください。

次の作業

- Cisco IOS CLI を使用して EEM ポリシーを記述するには、「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」の章を参照してください。
- Tcl を使用して EEM ポリシーを記述する方法については、「Writing Embedded Event Manager Policies Using Tcl」の章を参照してください。

Embedded Event Manager 4.0 の機能情報の概要

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

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

表 *178 : Embedded Event Manager 4.0* の機能情報の概要

機能名	リリース	機能情報
Embedded Event Manager 4.0	IOS 15.2(5)E1	この機能は、c2960cx にのみ導入され、サポートされています。

その他の参考資料

EEM に関連する参考資料については、次の各項を参照してください。

関連資料

関連項目	マニュアル タイトル
EEM コマンド：コマンド構文の詳細、デフォルト、コマンドモード、コマンド履歴、使用上の注意事項、および例	Cisco IOS Embedded Event Manager のコマンド リファレンス
CLI を使用して Embedded Event Manager ポリシーを記述する	「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」の章
Tcl を使用して Embedded Event Manager ポリシーを記述する	「Tcl を使用した Embedded Event Manager ポリシーの記述」の章
Embedded Resource Manager	「Embedded Resource Manager」の章

標準

標準	タイトル
新しい規格または変更された規格はサポートされていません。また、既存の規格に対するサポートに変更はありません。	--

MIB

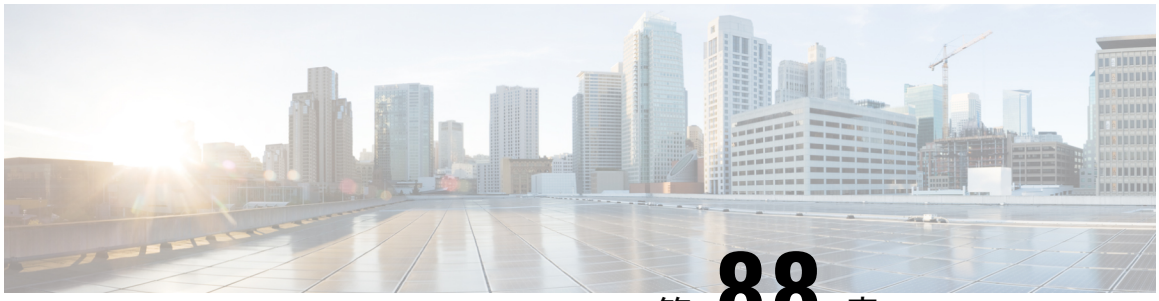
MIB	MIB のリンク
CISCO-EMBEDDED-EVENT-MGR-MIB	選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
新しい RFC または変更された RFC はサポートされていません。また、既存の RFC に対するサポートに変更はありません。	--

シスコのテクニカル サポート

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



第 88 章

Cisco IOS CLI を使用した EEM ポリシーの記述について

Cisco IOS CLI を使用した EEM ポリシーの記述に関する前提条件

- EEM ポリシーを記述する前に、「Embedded Event Manager の概要」の章で説明されている概念を十分に理解しておく必要があります。
- **action cns-event** コマンドを使用する場合は、Cisco Networking Services（CNS）イベントゲートウェイへのアクセスを設定する必要があります。
- **action force-switchover** コマンドを使用する場合は、デバイスでセカンダリプロセッサを設定する必要があります。
- **action snmp-trap** コマンドを使用した場合、**snmp-server enable traps event-manager** コマンドを有効にして、SNMP トラップが Cisco IOS デバイスから SNMP サーバーに送信されることを許可する必要があります。その他の関連する **snmp-server** コマンドを設定する必要もあります。詳細については、**action snmp-trap** コマンドのページを参照してください。

Cisco IOS CLI を使用した EEM ポリシーの記述について

Embedded Event Manager ポリシー

EEM では、イベントを監視し、監視対象のイベントが発生したときやしきい値を超えたときに情報通知や是正アクションを実施できます。EEM ポリシーは、イベントおよびイベントが発生した場合に行う処理を定義するエンティティです。EEM ポリシーにはアプレットとスクリプトの 2 つのタイプがあります。アプレットは、CLI 設定に定義された、ポリシーの単純な形式です。スクリプトは、Tool Command Language（Tcl）で記述されたポリシーの形式です。

EEM アプレット

EEM アプレットは、イベント スクリーニング基準とイベント発生時に実行するアクションを定義する簡潔な方法です。アプレット コンフィギュレーション モードでは、3 種類のコンフィギュレーション ステートメントがサポートされています。**event** コマンドを使用して実行するアプレットをトリガーするイベント基準を指定し、**action** コマンドを使用して、EEM アプレットがトリガーされるときに実行されるアクションを指定し、**set** コマンドを使用して EEM アプレット変数の値を設定します。現在、_exit_status 変数だけが、**set** コマンドでサポートされます。

アプレット コンフィギュレーション内では、**event** コンフィギュレーション コマンドを 1 つだけが使用できます。アプレット コンフィギュレーション モードが終了し、**event** コマンドが存在しない場合は、このアプレットにイベントが関連付けられていないことを示す警告が表示されます。イベントが指定されない場合、このアプレットは登録されたと見なされません。このアプレットにアクションが割り当てられない場合、イベントはトリガーされますが、アクションは実行されません。1 つのアプレット コンフィギュレーション内で複数の **action** コンフィギュレーション コマンドが使用できます。登録済みのアプレットを表示するには、**show event manager policy registered** コマンドを使用します。

EEM アプレットを修正する前に、アプレット コンフィギュレーション モードを終了するまで既存のアプレットを置き換えられないことに注意してください。アプレット コンフィギュレーション モードでアプレットを修正中であっても、既存のアプレットを実行できます。アプレットを登録解除することなく修正することが安全な方法です。アプレット コンフィギュレーション モードを終了すると、古いアプレットが登録解除され、新しいバージョンが登録されます。

action コンフィギュレーション コマンドは、*label* 引数を使用して一意に識別できます。この引数には任意の文字列値が使用できます。アクションは *label* 引数を使用してソート キーとして、英数字のキーの昇順に並べ替えられ、この順序で実行されます。

Embedded Event Manager は、ポリシーそのものに含まれるイベント仕様に基づいてポリシーをスケジューリングし、実行します。アプレット コンフィギュレーション モードが終了するとき、EEM は、入力された **event** コマンドと **action** コマンドを検査し、指定されたイベントの発生時に実行されるようにアプレットを登録します。

EEM スクリプト

スクリプトは、ネットワーキング デバイスの外部で ASCII エディタを使用して定義します。続いてスクリプトはネットワーキング デバイスにコピーされ EEM に登録されます。Tcl スクリプトは EEM でサポートされます。

EEM では、Tcl を使用して独自のポリシーを記述、実装できます。EEM ポリシーの記述には、次の作業が含まれます。

- ポリシーが実行されるイベントの選択。
- イベントの記録およびイベントへの対応に関連付けられたイベント ディテクタ オプションの定義。
- イベント発生後に実行されるアクションの選択。

シスコは、Tcl に EEM ポリシー開発を促進するキーワード拡張機能の形式を加えました。キーワードの主要なカテゴリでは、検出されたイベント、後続のアクション、ユーティリティ情報、カウンタの値、システム情報が特定されます。Tcl を使用して EEM ポリシーを記述する方法については、「Tcl を使用した Embedded Event Manager ポリシーの記述」の章を参照してください。

EEM アプレットに使用される Embedded Event Manager 組み込み環境変数

EEM 組み込み環境変数は、シスコ定義の環境変数のサブセットです。組み込み変数は、EEM アプレットでだけ利用できます。組み込み変数は、読み込み専用であるか、または読み込みおよび書き込み用のいずれかです。これらの変数は、1 個の特定のイベント ディテクタまたはすべてのイベント ディテクタに適用されます。次の表に、イベント ディテクタおよびサブイベントごとの読み込み専用のシスコ組み込み環境変数の一覧をアルファベット順に示します。

表 *179 : EEM* 組み込み環境変数（読み取り専用）

環境変数	説明
すべてのイベント	
_event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
_event_type	イベントのタイプ。
_event_type_string	イベントをトリガーしたイベントの種類を識別する ASCII 文字列。

環境変数	説明
_event_pub_sec _event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
_event_severity	イベントの重大度。
Application-Specific イベント ディテクタ	
_application_component_id	イベント アプリケーション コンポーネント ID。
_application_data1	イベントがパブリッシュされたときにアプリケーション固有のイベントに渡される、環境変数の値、文字テキスト、またはその両方の組み合わせ。
_application_data2	イベントがパブリッシュされたときにアプリケーション固有のイベントに渡される、環境変数の値、文字テキスト、またはその両方の組み合わせ。
_application_data3	イベントがパブリッシュされたときにアプリケーション固有のイベントに渡される、環境変数の値、文字テキスト、またはその両方の組み合わせ。
_application_data4	イベントがパブリッシュされたときにアプリケーション固有のイベントに渡される、環境変数の値、文字テキスト、またはその両方の組み合わせ。
_application_sub_system	イベント アプリケーション サブシステム番号。
_application_type	アプリケーションのタイプ。
CLI イベント ディテクタ	
_cli_msg	CLIイベントをトリガーした、完全に展開されたメッセージ。
_cli_msg_count	イベントがパブリッシュされる前にメッセージ一致が発生した回数。
Counter イベント ディテクタ	
_counter_name	カウンタの名前。
_counter_value	カウンタの値。
Enhanced Object Tracking イベント ディテクタ	
_track_number	トラッキング対象オブジェクトの数。

環境変数	説明
_track_state	トラッキング対象オブジェクトの状態（ダウン、またはアップ）。
Generic Online Diagnostics（GOLD）イベント ディテクタ	
_action_notify	GOLD イベント フラグのアクション通知情報（False または True）。
_event_severity	イベントの重大度（Normal、Minor、または Major）。
_gold_bl	起動診断レベル（次のいずれかの値）。 • 0：完全診断 • 1：最小診断 • 2：バイパス診断
_gold_card	GOLD 障害イベントが検出されたカード。
_gold_cf *testnum*	連続的な障害。*testnum* はテスト番号。たとえば、**_gold_cf3** は、テスト 3 の連続的な障害の EEM 組み込み環境変数です。
_gold_ci	カード インデックス。
_gold_cn	カードの名前。
_gold_ec *testnum*	テスト エラー コード。*testnum* はテスト番号。たとえば、**_gold_ec3** は、テスト 3 のエラー コードの EEM 組み込み環境変数です。
_gold_lf *testnum*	最終障害時間。*testnum* はテスト番号。たとえば、**_gold_lf3** は、テスト 3 の最終障害時間の EEM 組み込み環境変数です。 タイムスタンプの形式は *mmm dd yyyy hh*:*mm*:*ss* です。例：Mar 11 2005 08:47:00。
_gold_new_failure	GOLD イベント フラグの新しいテスト障害情報（False または True）。

環境変数	説明
_gold_overall_result	総合診断結果、次のいずれかの値である。 • 0：OK • 3：マイナー エラー • 4：メジャー エラー • 14：結果不明
_gold_pc	ポート数。
_gold_rc *testnum*	テスト総実行回数。*testnum* はテスト番号。たとえば、**_gold_rc3** は、テスト 3 の総実行回数の EEM 組み込み変数です。
_gold_sn	カード シリアル番号。
_gold_sub_card	GOLD 障害イベントが検出されたサブカード。
_gold_ta *testnum*	テスト属性名。*testnum* はテスト番号。たとえば、**_gold_ta3** は、テスト 3 の属性の EEM 組み込み環境変数です。
_gold_tc	テスト数。
_gold_tf *testnum*	合計障害回数。*testnum* はテスト番号。たとえば、**_gold_tf3** は、テスト 3 の合計障害回数の EEM 組み込み変数です。
_gold_tn *testnum*	テストの名前。*testnum* はテスト番号。たとえば、**_gold_tn3** は、テスト 3 の名前の EEM 組み込み環境変数です。
_gold_tr *testnum*	テストの結果。*testnum* はテスト番号。たとえば、**_gold_tr6** はテスト 6 用の EEM 組み込み変数です。テスト 6 はポート単位のテストでも、デバイス単位のテストでもありません。 テスト結果は、次の値のうちのいずれかです。 • P：診断結果 Pass • F：診断結果 Fail • U：診断結果 Unknown

環境変数	説明
_gold_tr *testnum* **d** *devnum*	デバイスごとのテスト結果。*testnum* はテスト番号で、*devnum* はデバイス番号です。たとえば、**_gold_tr3d20** は、テスト 3、デバイス 20 のテスト結果の EEM 組み込み環境変数です。 テスト結果は、次の値のうちのいずれかです。 • P：診断結果 Pass • F：診断結果 Fail • U：診断結果 Unknown
_gold_tr *testnum* **p** *portnum*	ポートごとのテスト結果。*testnum* はテスト番号で、*portnum* はポート番号です。たとえば、**_gold_tr5p20** は、テスト 5、ポート 20 のテスト結果の EEM 組み込み環境変数です。 テスト結果は、次の値のうちのいずれかです。 • P：診断結果 Pass • F：診断結果 Fail • U：診断結果 Unknown
_gold_tt	テストのタイプ。次のうちのいずれかです。 • 1：起動診断 • 2：オンデマンド診断 • 3：スケジュール診断 • 4：モニターリング診断
Interface Counter イベント ディテクタ	
_interface_is_increment	現在のインターフェイスカウンタ値が、絶対値（0）か増分値（1）かを示す値。
_interface_name	モニターされるインターフェイスの名前。
_interface_parameter	モニターされるインターフェイス カウンタの名前。
_interface_value	現在のインターフェイスカウンタ値と比較される値。
None イベント ディテクタ	

環境変数	説明
_event_id	1 であれば挿入イベントを示し、2 であれば削除イベントを示す値。
_none_argc **_none_arg1** **_none_arg2** **_none_arg3** **_none_arg4** **_none_arg5** **_none_arg6** **_none_arg7** **_none_arg8** **_none_arg9** **_none_arg10** **_none_arg11** **_none_arg12** **_none_arg13** **_none_arg14** **_none_arg15**	Extensible Markup Language（XML）Simple Object Access Protocol（SOAP）コマンドからスクリプトに渡されるパラメータ。
OIR イベント ディテクタ	
_oir_event	1 であれば挿入イベントを示し、2 であれば削除イベントを示す値。
_oir_slot	OIR イベントのスロット番号。
Resource イベント ディテクタ	
_resource_configured_threshold	設定されている ERM しきい値。
_resource_current_value	ERM によって報告された、現在の値。
_resource_dampen_time	ERM 減衰時間、ナノ秒単位。
_resource_direction	ERM イベント方向。イベント方向は、アップ、ダウン、または、変更なしのうちのいずれかです。
_resource_level	ERM イベント レベル。イベント レベルは、Normal、Minor、Major、および Critical の 4 つです。
_resource_notify_data_flag	ERM 通知データ フラグ。

環境変数	説明
_resource_owner_id	ERM リソース オーナー ID。
_resource_policy_id	ERM ポリシー ID。
_resource_policy_violation_flag	ERM ポリシー違反フラグ（False または True）。
_resource_time_sent	ERM イベント時間、ナノ秒単位。
_resource_user_id	ERM リソース ユーザー ID。
RF イベント ディテクタ	0 であれば RF イベントでないことを示し、1 であれば RF イベントであることを示す値。
_rf_event	
Remote Procedure Call（RPC）イベント ディテクタ	値 0 はエラーがないことを示し、値 1 ～ 83 はエラーを示します。
_rpc_event	
_rpc_argc **_rpc_arg0** **_rpc_arg1** **_rpc_arg2** **_rpc_arg3** **_rpc_arg4** **_rpc_arg5** **_rpc_arg6** **_rpc_arg7** **_rpc_arg8** **_rpc_arg9** **_rpc_arg10** **_rpc_arg11** **_rpc_arg12** **_rpc_arg13** **_rpc_arg14**	XML SOAP コマンドからアプレットに渡されるパラメータ。
SNMP イベント ディテクタ	0 であれば exit イベントでないことを示し、1 であれば exit イベントであることを示す値。
_snmp_exit_event	

環境変数	説明
_snmp_oid	パブリッシュされるイベントの原因となった SNMP オブジェクト ID。
_snmp_oid_delta_val	現在の SNMP オブジェクト ID の値と、イベントが最後にトリガーされたときの実際の増分差異。
_snmp_oid_val	イベントがパブリッシュされたときの SNMP オブジェクト ID 値。
SNMP 通知イベント ディテクタ	
_snmp_notif_oid	ユーザー指定オブジェクト ID。
_snmp_notif_oid_val	ユーザー指定オブジェクト ID 値。
_snmp_notif_src_ip_addr	SNMP プロトコル データ ユニット（PDU）の発信元 IP アドレス。
_snmp_notif_dest_ip_addr	SNMP PDU の宛先の IP アドレス。
_x_x_x_x_x_x_x(varbinds)	SNMP PDU varbind 情報。
_snmp_notif_trunc_vb_buf	バッファの領域不足から varbind 情報が切り捨てられているかどうかを示します。
syslog イベント ディテクタ	
_syslog_msg	パブリッシュされるイベントの原因となる syslog メッセージ。
System Manager（Process）イベント ディテクタ	
_process_dump_count	Posix プロセスがダンプされた回数。
_process_exit_status	終了時の Posix プロセスの状態。
_process_fail_count	Posix プロセスが失敗した回数。
_process_instance	Posix プロセスのインスタンス数。
_process_last_respawn	最後に再生成された Posix プロセス。
_process_node_name	Posix プロセスのノード名。
_process_path	Posix プロセスのパス。
_process_process_name	Posix プロセスの名前。
_process_respawn_count	Posix プロセスが再生成された回数。

環境変数	説明
Timer イベント ディテクタ	
_timer_remain	タイマーの期限が切れるまでの使用可能時間。 （注） この環境変数は、CRON タイマーには使用できません。
_timer_time	最後のイベントがトリガーされた時刻。
_timer_type	タイマーのタイプ。
Watchdog System Monitor（IOSWDSysMon）イベント ディテクタ	
_ioswd_node	ルート プロセッサ（RP）レポーティング ノードのスロット番号。
_ioswd_num_subs	存在するサブイベントの数。
全 Watchdog System Monitor（IOSWDSysMon）サブイベント	
_ioswd_sub1_present **_ioswd_sub2_present**	サブイベント 1 またはサブイベント 2 の存在を示す値。値1は、サブイベントが存在することを示し、値 0 はサブイベントが存在しないことを示します。
_ioswd_sub1_type _ioswd_sub2_type	イベントのタイプ（cpu_proc、または mem_proc）。
Watchdog System Monitor（IOSWDSysMon）cpu_proc サブイベント	
_ioswd_sub1_path _ioswd_sub2_path	サブイベントのプロセス名。
_ioswd_sub1_period **_ioswd_sub2_period**	サブイベントの測定に使用される時間間隔（秒単位、オプションでミリ秒単位）。
_ioswd_sub1_pid _ioswd_sub2_pid	サブイベントのプロセス ID。
_ioswd_sub1_taskname **_ioswd_sub2_taskname**	サブイベントのタスク名。
_ioswd_sub1_value _ioswd_sub2_value	パーセンテージで測定されたサブイベントの CPU 使用率。
Watchdog System Monitor（IOSWDSysMon）mem_proc サブイベント	

環境変数	説明
_ioswd_sub1_diff _ioswd_sub2_diff	イベントをトリガーした差のパーセンテージの値。 （注） この変数は、**_ioswd_sub1_is_percent** 変数または **_ioswd_sub2_is_percent** 変数が 1 である場合に限って設定されます。
_ioswd_sub1_is_percent **_ioswd_sub2_is_percent**	値がパーセンテージであるかどうかを識別する番号。0 であれば値がパーセンテージではないことを意味し、1 であれば値がパーセンテージであることを意味します。
_ioswd_sub1_path _ioswd_sub2_path	サブイベントのプロセス名。
_ioswd_sub1_pid _ioswd_sub2_pid	サブイベントのプロセス ID。
_ioswd_sub1_taskname **_ioswd_sub2_taskname**	サブイベントのタスク名。
_ioswd_sub1_value _ioswd_sub2_value	パーセンテージで測定されたサブイベントの CPU 使用率。
Watchdog System Monitor（WDSysMon）イベント ディテクタ	
_wd_sub1_present _wd_sub2_present	サブイベント 1 またはサブイベント 2 の存在を示す値。値1は、サブイベントが存在することを示し、値 0 はサブイベントが存在しないことを示します。
_wd_num_subs	存在するサブイベントの数。
_wd_sub1_type _wd_sub2_type	イベントのタイプ（cpu_proc、cpu_tot、deadlock、dispatch_mgr、mem_proc、mem_tot_avail、または mem_tot_used）。
Watchdog System Monitor（WDSysMon）cpu_proc サブイベント	
_wd_sub1_node _wd_sub2_node	サブイベント RP レポーティング ノードのスロット番号。
_wd_sub1_period _wd_sub2_period	サブイベントの測定に使用される時間間隔（秒単位、オプションでミリ秒単位）。
_wd_sub1_procname **_wd_sub2_procname**	サブイベントのプロセス名。
_wd_sub1_value _wd_sub2_value	パーセンテージで測定されたサブイベントの CPU 使用率。

環境変数	説明
Watchdog System Monitor（WDSysMon）cpu_tot サブイベント	
_wd_sub1_node _wd_sub2_node	サブイベント RP レポーティング ノードのスロット番号。
_wd_sub1_period _wd_sub2_period	サブイベントの測定に使用される時間間隔（秒単位、オプションでミリ秒単位）。
_wd_sub1_value _wd_sub2_value	パーセンテージで測定されたサブイベントの CPU 使用率。
Watchdog System Monitor（WDSysMon）deadlock サブイベント	
_wd_sub1_entry_ [1-N]**_b_node** **_wd_sub2_entry_**[1-N]**_b_node**	サブイベント RP レポーティング ノードのスロット番号。
_wd_sub1_entry_ [1-N]**_b_pid** **_wd_sub2_entry_**[1-N]**_b_pid**	サブイベントのプロセス ID。
_wd_sub1_entry_ [1-N]**_b_procname** **_wd_sub2_entry_**[1-N]**_b_procname**	サブイベントのプロセス名。
_wd_sub1_entry_ [1-N]**_b_tid** **_wd_sub2_entry_**[1-N]**_b_tid**	サブイベントの時間 ID。
_wd_sub1_entry_ [1-N]**_node** **_wd_sub2_entry_**[1-N]**_node**	サブイベント RP レポーティング ノードのスロット番号。
_wd_sub1_entry_ [1-N]**_pid** **_wd_sub2_entry_**[1-N]**_pid**	サブイベントのプロセス ID。
_wd_sub1_entry_ [1-N]**_procname** **_wd_sub2_entry_**[1-N]**_procname**	サブイベントのプロセス名。
_wd_sub1_entry_ [1-N]**_state** **_wd_sub2_entry_**[1-N]**_state**	サブイベントの時間 ID。
_wd_sub1_entry_ [1-N]**_tid** **_wd_sub2_entry_**[1-N]**_tid**	サブイベントの時間 ID。
_wd_sub1_num_entries **_wd_sub2_num_entries**	サブイベントの数。
Watchdog System Monitor（WDSysMon）dispatch manager サブイベント	
_wd_sub1_node _wd_sub2_node	サブイベント RP レポーティング ノードのスロット番号。

環境変数	説明
_wd_sub1_period _wd_sub2_period	サブイベントの測定に使用される時間間隔（秒単位、オプションでミリ秒単位）。
_wd_sub1_procname **_wd_sub2_procname**	サブイベントのプロセス名。
_wd_sub1_value _wd_sub2_value	パーセンテージで測定されたサブイベントの CPU 使用率。
Watchdog System Monitor（WDSysMon）mem_proc サブイベント	
_wd_sub1_diff _wd_sub2_diff	イベントをトリガーした差のパーセンテージの値。 （注） この変数は、**_wd_sub1_is_percent** 変数または **_wd_sub2_is_percent** 変数が 1 である場合に限って設定されます。
_wd_sub1_is_percent **_wd_sub2_is_percent**	値がパーセンテージであるかどうかを識別する番号。0 であれば値がパーセンテージではないことを意味し、1 であれば値がパーセンテージであることを意味します。
_wd_sub1_node _wd_sub2_node	サブイベント RP レポーティング ノードのスロット番号。
_wd_sub1_period _wd_sub2_period	サブイベントの測定に使用される時間間隔（秒単位、オプションでミリ秒単位）。
_wd_sub1_pid _wd_sub2_pid	サブイベントのプロセス ID。
_wd_sub1_procname **_wd_sub2_procname**	サブイベントのプロセス名。
_wd_sub1_value _wd_sub2_value	パーセンテージで測定されたサブイベントの CPU 使用率。
Watchdog System Monitor（WDSysMon）mem_tot_avail and mem_tot_used サブイベント	
_wd_sub1_avail _wd_sub2_avail	サブイベントに使用可能なメモリ。
_wd_sub1_diff _wd_sub2_diff	イベントをトリガーした差のパーセンテージの値。 （注） この変数は、**_wd_sub1_is_percent** 変数または **_wd_sub2_is_percent** 変数が 1 である場合に限って設定されます。

環境変数	説明
_wd_sub1_is_percent **_wd_sub2_is_percent**	値がパーセンテージであるかどうかを識別する番号。0 であれば値がパーセンテージではないことを意味し、1 であれば値がパーセンテージであることを意味します。
_wd_sub1_node　_wd_sub2_node	サブイベント RP レポーティング ノードのスロット番号。
_wd_sub1_period　_wd_sub2_period	サブイベントの測定に使用される時間間隔（秒単位、オプションでミリ秒単位）。
_wd_sub1_value　_wd_sub2_value	パーセンテージで測定されたサブイベントの CPU 使用率。
_wd_sub1_used　_wd_sub2_used	サブイベントが使用したメモリ。

Cisco IOS CLI を使用した EEM ポリシーの記述方法

Embedded Event Manager アプレットの登録と定義

アプレットを Embedded Event Manager に登録し、Cisco IOS CLI **event** コマンドと **action** コマンドを使用して定義するには、次の作業を実行します。EEM アプレットでは、**event** コマンドが 1 つだけ許可されます。**action** コマンドは複数許可されます。**event** コマンドと **action** コマンドが指定されていない場合、コンフィギュレーション モードの終了時にアプレットが削除されます。

この作業で使用する SNMP イベントディテクタと syslog **action** コマンドは、任意のイベントディテクタと **action** コマンドを表しています。他のイベントディテクタや **action** コマンドの使用例については、Embedded Event Manager アプレットの設定例 （2099 ページ） を参照してください。

EEM 環境変数

EEM ポリシーの EEM 環境変数は、EEM **event manager environment** コンフィギュレーション コマンドを使用して定義されます。慣例として、すべてのシスコ EEM 環境変数は、「_」で始まります。将来的な競合を避けるため、「_」で始まる新しい変数を定義しないことを推奨します。

show event manager environment 特権 EXEC コマンドを使用して、システムの EEM 環境変数セットを表示できます。

たとえば、イベント発生時に E メールを送信する EEM ポリシーを作成できます。次の表に、EEM ポリシーで使用できる電子メール特有の環境変数の説明を示します。

表 *180 : EEM* 電子メール固有の環境変数

環境変数	説明	例
_email_server	E メール送信に使用されるシンプル メール転送プロトコル（SMTP）メール サーバー。	電子メール サーバー名（Mailservername）は、次のテンプレート形式のいずれかを使用できます。 • username:password@host • username@host • ホスト
_email_to	E メールの送信先アドレス。	engineering@example.com
_email_from	E メールの送信元アドレス。	devtest@example.com
_email_cc	E メールのコピーの送信先アドレス。	manager@example.com

EEM アクション ラベルのアルファベット順

EEM アクション ラベルは一意の ID で、任意の文字列値が可能です。アクションは、ラベルをソート キーとして使用して、英数字のキーの昇順（辞書順）にソートされ、実行されます。ラベルとして数字を使用している場合は、英数字ソートは、10.0 は 1.0 よりも後ですが、2.0 よりも前になることに注意してください。このような場合、01.0、02.0 のような数字を使用する、または頭文字の後に同様の数字を続けることを推奨します。

手順の概要

1. **enable**
2. **show event manager environment** [**all**| *variable-name*]
3. **configure terminal**
4. **event manager environment** *variable-name string*
5. EEM アクション ラベルのアルファベット順 を、必要なすべての環境変数に繰り返します。
6. **event manager applet** *applet-name*
7. 次のいずれかを実行します。
   - **event snmp oid** *oid-value* **get-type** {**exact**| **next**} **entry-op** *operator* **entry-val** *entry-value*[**exit-comb**| **and**}] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*
8. **action** *label* **cli command** *cli-string* [**pattern** *pattern-string*]
9. **action** *label* **syslog** [**priority** *priority-level*] **msg** *msg-text* **facility** *string*
10. **action** *label* **mail server** *server-address* **to** *to-address* **from** *from-address* [**cc** *cc-address*] **subject** *subject* **body** *body-text*

11. 必要に応じて action コマンドを追加します。
12. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager environment** [**all**| *variable-name*] 例： `Device# show event manager environment all`	（任意）EEM 環境変数の名前と値を表示します。 • オプションの **all** キーワードは、すべての EEM 環境変数を表示します。 • オプションの *variable-name* 引数は、指定された環境変数に関する情報を表示します。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**event manager environment** *variable-name string* 例： `Device(config)# event manager environment _email_to engineering@example.com`	指定された EEM 環境変数の値を設定します。 • この例では、E メール送信先の E メール アドレスを保持する環境変数は、engineering@example.com に設定されます。
ステップ 5	EEM アクション ラベルのアルファベット順 を、必要なすべての環境変数に繰り返します。	EEM アクション ラベルのアルファベット順 を繰り返して、EEM アクション ラベルのアルファベット順 で登録されるポリシーに必要なすべての環境変数を設定します。
ステップ 6	**event manager applet** *applet-name* 例： `Device(config)# event manager applet memory-fail`	Embedded Event Manager（EEM）にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 7	次のいずれかを実行します。 • **event snmp oid** *oid-value* **get-type** {**exact**| **next**} **entry-op** *operator* **entry-val** *entry-value*[**exit-comb**| **and**}] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*	EEM アプレットの実行の原因となる、イベント基準を指定します。 • この例では、空きメモリの値が 5120000 を下回ったときに EEM イベントがトリガーされます。

	コマンドまたはアクション	目的
	例： `Device(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val 5120000 poll-interval 90`	• 終了基準はオプションです。指定されない場合、イベントのモニターリングは、すぐに再び有効になります。
ステップ 8	**action** *label* **cli command** *cli-string* [**pattern** *pattern-string*] 例： `Device(config-applet)# action 1.0 cli command "enable"` 例： `Device(config-applet)# action 2.0 cli command "clear counters Ethernet0/1" pattern "confirm"` 例： `Device(config-applet)# action 3.0 cli command "y"`	EEM アプレットがトリガーされたときに Cisco IOS CLI コマンドを実行するアクションを指定します。 **pattern** キーワードはオプションで、コマンド文字列が入力を求める場合にだけ使用します。**action cli** コマンドは、オプションの **pattern** キーワードで指定されているとおりの応答プロンプトを受信した時点で終了します。次の応答プロンプトに一致する正規表現パターンを指定する必要があります。正しくないパターンを指定すると、**action cli** コマンドが、maxrun タイマー期限切れによるアプレット実行タイムアウトまで、待ち続けることになります。 • 実行されるアクションは、**pattern** キーワードが **clear counters Ethernet0/1** コマンドの *confirm* 引数を指定するときに実行される EEM アプレットを指定するためのものです。この場合、コマンド文字列は「confirm」という入力を要求します。その入力は、「yes」または「no」で完了する必要があります。
ステップ 9	**action** *label* **syslog** [**priority** *priority-level*] **msg** *msg-text* **facility** *string* 例： `Device(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current available memory is $_snmp_oid_val bytes"` 例： `Device(config-applet)# action 1.0 syslog priority errors facility EEM-FAC message "TEST MSG"`	EEM アプレットがトリガーされたときに実行されるアクションを指定します。 この例では、実行されるアクションは syslog にメッセージを書き込むことです。 • オプションの **priority** キーワードは syslog メッセージの優先度レベルを指定します。選択した場合は、*priority-level* 引数を定義する必要があります。 • *msg-text* 引数は、文字テキスト、環境変数、またはその両方の組み合わせが可能です。 • **facility** キーワードは生成したメッセージの場所を指定します。 • *string* 引数は、キャラクタ テキスト、環境変数、またはその両方の組み合わせが可能です。

	コマンドまたはアクション	目的
ステップ 10	**action** *label* **mail server** *server-address* **to** *to-address* **from** *from-address* [**cc** *cc-address*] **subject** *subject* **body** *body-text* 例： `Device(config-applet)# action 2.0 mail server 192.168.1.10 to engineering@example.com from devtest@example.com subject "Memory failure" body "Memory exhausted; current available memory is $_snmp_oid_val bytes"`	EEM アプレットがトリガーされたときにショートメールを送信するアクションを指定します。 • *server-address* 引数は、電子メールの転送に使用する電子メール サーバーの完全修飾ドメイン名を指定します。 • *to-address* 引数は、電子メールの送信先の電子メール アドレスを指定します。 • *from-address* 引数は、電子メール送信元の電子メール アドレスを指定します。 • *subject* 引数は、英数字の文字列で、電子メールのサブジェクト ラインの内容を指定します。 • *body-text* 引数は、英数字の文字列で、電子メールのテキストの内容を指定します。
ステップ 11	必要に応じて action コマンドを追加します。	--
ステップ 12	**end** 例： `Device(config-applet)# end`	アプレット コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

トラブルシューティングのヒント

特権 EXEC モードで **debug event manager** コマンドを使用して、EEM コマンド操作のトラブルシューティングを行います。debugging コマンドは注意して使用してください。生成される出力量によってデバイスの動作が遅くなったり、停止したりすることがあります。シスコ エンジニアの管理下に限ってこのコマンドを使用することを推奨します。

EEM Tcl スクリプトの登録と定義

環境変数を設定し、EEM ポリシーを登録するには、この作業を実行します。EEM は、ポリシーそのものに含まれるイベント仕様に基づいてポリシーをスケジューリングし、実行します。EEM ポリシーが登録されると、ソフトウェアによって、ポリシーが調べられ、指定されたイベントの発生時に実行されるよう、登録されます。

始める前に

Tcl スクリプト言語で記述されたポリシーが使用できる状態である必要があります。サンプル ポリシーを示します。使用している Cisco IOS リリースのイメージで使用可能なポリシーについては、EEM サンプル ポリシー（2139 ページ）を参照してください。これらのサンプル ポリシーは、システム ポリシー ディレクトリに保存されています。

手順の概要

1. **enable**
2. **show event manager environment** [**all**| *variable-name*]
3. **configure terminal**
4. **event manager environment** *variable-name string*
5. EEM Tcl スクリプトの登録と定義 を繰り返して、EEM Tcl スクリプトの登録と定義 で登録されるポリシーに必要なすべての環境変数を設定します。
6. **event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]
7. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager environment** [**all**| *variable-name*] 例： `Device# show event manager environment all`	（任意）EEM 環境変数の名前と値を表示します。 • オプションの **all** キーワードは、すべての EEM 環境変数を表示します。 • オプションの *variable-name* 引数は、指定された環境変数に関する情報を表示します。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**event manager environment** *variable-name string* 例： `Device(config)# event manager environment _cron_entry 0-59/2 0-23/1 * * 0-6`	指定された EEM 環境変数の値を設定します。 • この例では、ソフトウェアによって、CRON タイマー環境変数が、毎日、毎時の 2 分目に設定されます。
ステップ 5	EEM Tcl スクリプトの登録と定義 を繰り返して、EEM Tcl スクリプトの登録と定義 で登録されるポリシーに必要なすべての環境変数を設定します。	--
ステップ 6	**event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**] 例： `Device(config)# event manager policy tm_cli_cmd.tcl type system`	ポリシー内で定義された指定イベントが発生した場合に、EEM ポリシーを実行するよう、定義します。 • **system** キーワードを使用して、シスコ定義のシステムポリシーを登録します。

	コマンドまたはアクション	目的
		• **user** キーワードを使用して、ユーザー定義のシステムポリシーを登録します。 • **trap** キーワードを使用して、ポリシーがトリガーされた場合の SNMP トラップを生成します。 • この例では、tm_cli_cmd.tcl という名前の EEM サンプル ポリシーが、システム ポリシーとして定義されます。
ステップ 7	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、**show event manager environment** 特権 EXEC コマンドを使用して、すべての EEM 環境変数の名前と値を表示する例を示します。

```
Device# show event manager environment all
No.  Name                           Value
1    _cron_entry                    0-59/2 0-23/1 * * 0-6
2    _show_cmd                      show ver
3    _syslog_pattern                .*UPDOWN.*Ethernet1/0.*
4    _config_cmd1                   interface Ethernet1/0
5    _config_cmd2                   no shut
```

Embedded Event Manager ポリシーの登録解除

EEM ポリシーを実行コンフィギュレーション ファイルから削除するには、次の作業を実行します。ポリシーの実行はキャンセルされます。

手順の概要

1. **enable**
2. **show event manager policy registered** [**description** [*policy-name*] | **detailed** *policy-filename* [**system** | **user**] | [**event-type** *event-name*] [**system** | **user**] [**time-ordered** | **name-ordered**]]
3. **configure terminal**
4. **no event manager policy** *policy-filename*
5. **exit**
6. ステップ 2 を繰り返して、ポリシーが削除されたことを確認します。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager policy registered** [**description** [*policy-name*] | **detailed** *policy-filename* [**system** | **user**] | [**event-type** *event-name*] [**system** | **user**] [**time-ordered** | **name-ordered**]] 例： `Device# show event manager policy registered`	（任意）現在登録されている EEM ポリシーを表示します。 • オプションの **system** キーワードおよび **user** キーワードは登録されているシステム ポリシーおよびユーザー ポリシーを表示します。 • キーワードが指定されない場合は、すべてのイベント タイプに対する登録された EEM ポリシーが時間順に表示されます。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**no event manager policy** *policy-filename* 例： `Device(config)# no event manager policy IPSLAping1`	ポリシーを登録解除するために EEM ポリシーを設定から削除します。
ステップ 5	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	ステップ 2 を繰り返して、ポリシーが削除されたことを確認します。 例： `Device# show event manager policy registered`	--

例

次に、**show event manager policy registered** 特権 EXEC コマンドを使用して、現在登録されている 2 個の EEM アプレットを表示する例を示します。

```
Device# show event manager policy registered
No.  Class   Type    Event Type          Trap  Time Registered          Name
```

```
1    applet  system  snmp                Off   Fri Aug 12 17:42:52 2005  IPSLAping1
 oid {1.3.6.1.4.1.9.9.42.1.2.9.1.6.4} get-type exact entry-op eq entry-val {1}
 exit-op eq exit-val {2} poll-interval 90.000
 action 1.0 syslog priority critical msg "Server IPecho Failed: OID=$_snmp_oid_val"
 action 1.1 snmp-trap strdata "EEM detected server reachability failure to 10.1.88.9"
 action 1.2 publish-event sub-system 88000101 type 1 arg1 "10.1.88.9" arg2 "IPSLAEcho"
arg3 "fail"
 action 1.3 counter name _IPSLA1F op inc value 1
2    applet  system  snmp                Off   Thu Sep 15 05:57:16 2005  memory-fail
 oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
 action 1.0 syslog priority critical msg Memory exhausted; current available memory is
$_snmp_oid_val bytes
 action 2.0 force-switchover
```

次の例では、**show event manager policy registered** 特権 EXEC コマンドを使用して、アプレット IPSLAping1 が **no event manager policy** コマンドの入力後に削除されていることを示します。

```
Device# show event manager policy registered
No.  Class   Type    Event Type          Trap  Time Registered           Name
1    applet  system  snmp                Off   Thu Sep 15 05:57:16 2005  memory-fail
 oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
 action 1.0 syslog priority critical msg Memory exhausted; current available memory is
$_snmp_oid_val bytes
 action 2.0 force-switchover
```

すべての Embedded Event Manager ポリシーの実行の一時停止

すべての EEM ポリシーの実行をただちに一時停止するには、次の作業を実行します。一時的なパフォーマンスまたはセキュリティ面での理由から、ポリシーの登録解除ではなく一時停止が必要なことがあります。

手順の概要

1. **enable**
2. **show event manager policy registered** [**description** [*policy-name*] | **detailed** *policy-filename* [**system** | **user**] | [**event-type** *event-name*] [**system** | **user**] [**time-ordered** | **name-ordered**]]
3. **configure terminal**
4. **event manager scheduler suspend**
5. **exit**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**show event manager policy registered** [**description** [*policy-name*] | **detailed** *policy-filename* [**system** | **user**] | [**event-type** *event-name*] [**system** | **user**] [**time-ordered** | **name-ordered**]] 例： `Device# show event manager policy registered`	（任意）現在登録されている EEM ポリシーを表示します。 • オプションの **system** キーワードおよび **user** キーワードは登録されているシステム ポリシーおよびユーザー ポリシーを表示します。 • キーワードが指定されない場合は、すべてのイベント タイプに対する登録された EEM ポリシーが時間順に表示されます。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**event manager scheduler suspend** 例： `Device(config)# event manager scheduler suspend`	すべての EEM ポリシーの実行がすぐに一時停止されます。
ステップ 5	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Embedded Event Manager 履歴データの表示

履歴テーブルのサイズを変更し、EEM 履歴データを表示するには、次の任意の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager history size** {**events** | **traps**} [*size*]
4. **exit**
5. **show event manager history events** [**detailed**] [**maximum** *number*]
6. **show event manager history traps** {**server** | **policy**}

手順の詳細

ステップ 1 enable

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 configure terminal

グローバル コンフィギュレーション モードを開始します。

例：

```
Device# configure terminal
```

ステップ 3 event manager history size {**events** | **traps**} [*size*]

このコマンドを使用して、EEM イベント履歴テーブルのサイズ、または、EEM SNMP トラップ履歴テーブルのサイズを変更します。次に、EEM イベント履歴テーブルのサイズを 30 エントリに変更する例を示します。

例：

```
Device(config)# event manager history size events 30
```

ステップ 4 exit

グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例：

```
Device(config)# exit
```

ステップ 5 show event manager history events [**detailed**] [**maximum** *number*]

このコマンドを使用して、各 EEM イベントの詳細情報を表示します。次に例を示します。

例：

```
Device# show event manager history events
No.  Time of Event             Event Type          Name
1    Fri Aug13  21:42:57 2004  snmp                applet: SAAping1
2    Fri Aug13  22:20:29 2004  snmp                applet: SAAping1
3    Wed Aug18  21:54:48 2004  snmp                applet: SAAping1
4    Wed Aug18  22:06:38 2004  snmp                applet: SAAping1
5    Wed Aug18  22:30:58 2004  snmp                applet: SAAping1
6    Wed Aug18  22:34:58 2004  snmp                applet: SAAping1
7    Wed Aug18  22:51:18 2004  snmp                applet: SAAping1
8    Wed Aug18  22:51:18 2004  application         applet: CustApp1
```

ステップ 6 show event manager history traps {**server** | **policy**}

このコマンドを使用して、EEM サーバーまたは EEM ポリシーのいずれかから送信された EEM SNMP トラップを表示します。次に、EEM ポリシー内からトリガーされた EEM SNMP トラップが表示される例を示します。

例：

```
Device# show event manager history traps policy
```

```
No.  Time                      Trap Type           Name
1    Wed Aug18  22:30:58 2004  policy              EEM Policy Director
2    Wed Aug18  22:34:58 2004  policy              EEM Policy Director
3    Wed Aug18  22:51:18 2004  policy              EEM Policy Director
```

Embedded Event Manager 登録済みポリシーの表示

登録済みの EEM ポリシーを表示するには、次の任意の作業を実行します。

手順の概要

1. **enable**
2. **show event manager policy registered** [**event-type** *event-name*] [**time-ordered**| **name-ordered**]

手順の詳細

ステップ 1 enable

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 show event manager policy registered [event-type event-name] [time-ordered| name-ordered]

このコマンドを **time-ordered** キーワードとともに使用して、現在登録されているポリシーの情報を時間でソートして表示します。次に例を示します。

例：

```
Device# show event manager policy registered time-ordered
No.  Type    Event Type          Time                    Registered Name
1    applet  snmp                Thu May30 05:57:16 2004 memory-fail
 oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val
{5120000} poll-interval 90
action 1.0 syslog priority critical msg "Memory exhausted; current available memory
is $_snmp_oid_val bytes"
 action 2.0 force-switchover
2    applet  syslog              Wed Jul16 00:05:17 2004 intf-down
 pattern {.*UPDOWN.*Ethernet1/0.*}
 action 1.0 cns-event msg "Interface state change: $_syslog_msg"
```

このコマンドを **name-ordered** キーワードとともに使用して、現在登録されているポリシーの情報を名前でソートして表示します。次に例を示します。

例：

```
Device# show event manager policy registered name-ordered
No.  Type    Event Type          Time Registered          Name
1    applet  syslog              Wed Jul16  00:05:17 2004 intf-down
 pattern {.*UPDOWN.*Ethernet1/0.*}
 action 1.0 cns-event msg "Interface state change: $_syslog_msg"
```

```
2    applet  snmp                Thu May30 05:57:16 2004   memory-fail
 oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val
{5120000} poll-interval 90
 action 1.0 syslog priority critical msg "Memory exhausted; current available memory
is $_snmp_oid_val bytes"
 action 2.0 force-switchover
```

このコマンドを **event-type** キーワードとともに使用して、*event-name* 引数で指定されたイベントタイプの現在登録されているポリシーに関する情報を表示します。次に例を示します。

例：

```
Device# show event manager policy registered event-type syslog
No.  Type    Event Type          Time Registered           Name
1    applet  syslog              Wed Jul16  00:05:17 2004 intf-down
 pattern {.*UPDOWN.*Ethernet1/0.*}
 action 1.0 cns-event msg "Interface state change: $_syslog_msg"
```

イベント SNMP 通知の設定

SNMP 通知を設定するには、次の作業を実行します。

始める前に

- SNMP イベントマネージャは、**snmp-server manager** コマンドを使用して設定する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event** [**tag** *event-tag*] **snmp-notification oid** *oid-string* **oid-val** *comparison-value* **op** *operator* [**maxrun** *maxruntime-number*] [**src-ip-address** *ip-address*] [**dest-ip-address** *ip-address*] [**default** *seconds*] [**direction** {**incoming** | **outgoing**}] [**msg-op** {**drop** | **send**}]
5. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	`Device# configure terminal`	
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet snmp`	Event Manager にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**event** [**tag** *event-tag*] **snmp-notification oid** *oid-string* **oid-val** *comparison-value* **op** *operator* [**maxrun** *maxruntime-number*] [**src-ip-address** *ip-address*] [**dest-ip-address** *ip-address*] [**default** *seconds*] [**direction** {**incoming** | **outgoing**}] [**msg-op** {**drop** | **send**}] 例： `Device(config-applet)# event snmp-notification dest-ip-address 192.168.1.1 oid 1 op eq oid-val 10`	簡易ネットワーク管理プロトコル（SNMP）通知のサンプリングによって実行される Embedded Event Manager（EEM）アプレットのイベント基準を指定します。
ステップ 5	**end** 例： `Device(config-applet)# end`	アプレット コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

複数イベント サポートの設定

複数イベントサポート機能は、EEMサーバーに複数のイベントを登録する機能を追加します。複数イベントサポートには、1個以上のイベントの発生、1個以上のトラッキング対象オブジェクトの状態、および、発生するイベントの時間間隔が含まれます。イベント パラメータは、CLIコマンドで指定されます。複数イベントを扱うためのデータ構造には、複数のイベントIDと相関関係ロジックが含まれます。このデータは、EEM サーバーに複数のイベントを登録するために使用されます。

イベント設定パラメータの設定

trigger コマンドは、トリガー アプレット コンフィギュレーション モードを開始し、EEM アプレットの複数イベント設定ステートメントを指定します。トリガー ステートメントは、各イベント文に指定される *tag* 引数を使用して複数イベント ステートメントを関連付けます。イベントは指定されたパラメータに基づいて発生します。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*

4. **event** [**tag** *event-tag*] **cli pattern** *regular-expression* **sync** {**yes** | **no skip** {**yes** | **no**}} [**occurs** *num-occurrences*] [**period** *period-value*] [**maxrun** *maxruntime-number*]
5. **trigger** [**occurs** *occurs-value*] [**period** *period-value*] [**period-start** *period-start-value*] [**delay** *delay-value*]
6. **correlate** {**event** *event-tag* | **track** *object-number*} [*boolean-operator* **event** *event-tag*]
7. **attribute tag** *event-tag* [**occurs** *occurs-value*]
8. **action** *label* **cli command** *cli-string*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet EventInterface`	EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**event** [**tag** *event-tag*] **cli pattern** *regular-expression* **sync** {**yes**	**no skip** {**yes**	**no**}} [**occurs** *num-occurrences*] [**period** *period-value*] [**maxrun** *maxruntime-number*] 例： `Device(config-applet)# event tag 1.0 cli pattern "show bgp all" sync yes occurs 32 period 60 maxrun 60`	Cisco IOS コマンドライン インターフェイス（CLI）コマンドの一致によって実行される EEM アプレットのイベント基準を指定します。
ステップ 5	**trigger** [**occurs** *occurs-value*] [**period** *period-value*] [**period-start** *period-start-value*] [**delay** *delay-value*] 例： `Device(config-applet)# trigger occurs 1 period-start "0 8 * * 1-5" period 60`	EEM アプレットの複雑なイベント設定パラメータを指定します。
ステップ 6	**correlate** {**event** *event-tag*	**track** *object-number*} [*boolean-operator* **event** *event-tag*] 例：	EEM アプレットのトリガー モードで複雑なイベント関連付けを指定します。

	コマンドまたはアクション	目的
	`Device(config-applet)# correlate event 1.0 or event 2.0`	（注） 「and」を使用して、トラップやsyslogメッセージなどのイベントをグループ化した場合、デフォルトのトリガー発生時間枠は3分です。
ステップ 7	**attribute tag** *event-tag* [**occurs** *occurs-value*] 例： `Device(config-applet)# attribute tag 1.0 occurs 1`	EEMアプレットの複雑なイベントをビルドする最大8個の属性文を指定します。
ステップ 8	**action** *label* **cli command** *cli-string* 例： `Device(config-applet)# action 1.0 cli command "show pattern"`	EEM アプレットがトリガーされたときに CLI コマンドを実行するアクションを指定します。

例

次に、**show bgp all** CLI コマンドと「COUNT」文字列を含む syslog メッセージが 60 秒以内に発生した場合にアプレットが実行される例を示します。

```
event manager applet delay_50
 event tag 1.0 cli pattern "show bgp all" sync yes occurs 32 period 60 maxrun 60
 event tag 2.0 syslog pattern "COUNT"
 trigger occurs 1 delay 50
  correlate event 1.0 or event 2.0
  attribute tag 1.0 occurs 1
  attribute tag 2.0 occurs 1
 action 1.0 cli command "show pattern"
 action 2.0 cli command "enable"
 action 3.0 cli command "config terminal"
 action 4.0 cli command " ip route 192.0.2.0 255.255.255.224 192.0.2.12"
 action 91.0 cli command "exit"
 action 99.0 cli command "show ip route | incl 192.0.2.5"
```

EEM クラスベース スケジューリングの設定

Embedded Event Manager（EEM）ポリシーをスケジュールし、ポリシー スケジュール オプションを設定するには、次の作業を実行します。このタスクでは、2 個の EEM 実行スレッドが作成され、デフォルト クラスに割り当てられたアプレットが実行されます。

EEM ポリシーは、登録時に **class** キーワードを使用して、クラスに割り当てられます。クラスなしで登録された EEM ポリシーは、デフォルト クラスに割り当てられます。デフォルト クラスを保持するスレッドは、スレッドが作業に利用可能であるとき、デフォルト クラスをサービスします。特定のクラス文字に割り当てられたスレッドは、スレッドが作業に利用可能であるとき、クラス文字が一致する任意のポリシーをサービスします。

EEM 実行スレッドが、指定されたクラスのポリシー実行に利用可能でない場合で、クラスのスケジューラルールが設定されている場合は、ポリシーは該当クラスのスレッドが実行可能になるまで待ちます。同じ入力イベントからトリガーされた同期ポリシーは、同一の実行スレッドにスケジュールされなければなりません。

手順の概要

1. **enable**
2. **configure terminal**
3. {| |} クラスオプションスレッド **event manager scheduler appletaxpcall-homethread class** *class-options* **number** 番号
4. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	{| |} クラスオプションスレッド **event manager scheduler appletaxpcall-homethread class** *class-options* **number** 番号 例： `Device(config)# event manager scheduler applet thread class default number 2`	EEM ポリシーをスケジュールし、ポリシー スケジューリング オプションを設定します。 • この例では、2 個の EEM 実行スレッドが作成され、デフォルト クラスに割り当てられたアプレットが実行されます。
ステップ 4	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

スケジュール済み EEM ポリシー イベントまたはイベント キューの保留

EEM スケジューラで、スケジュールされた EEM ポリシー イベントまたはイベント キューをホールドするには、次の作業を実行します。このタスクでは、すべての保留 EEM ポリシーが表示されます。ジョブ ID 2 を使用して特定されるポリシーは、EEM スケジューラでホールドされています。最初のステップは、ジョブ ID 2 のポリシーは、状態が Pending から Held に変更されていることを示しています。

手順の概要

1. **enable**
2. **show event manager policy pending** [**queue-type**{**applet** | **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]
3. **event manager scheduler hold** {**all**| **policy** *job-id* | **queue-type** {**applet** | **call-home** | **axp** | **script**} **class** *class-options*} [**processor** {**rp_primary**| **rp_standby**}]
4. **show event manager policy pending** [**queue-type**{**applet** | **call-home** | **axp** | **script**} **class** *class-options* | **detailed**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager policy pending** [**queue-type**{**applet**	**call-home**	**axp**	**script**} **class** *class-options*	**detailed**] 例： `Device# show event manager policy pending`	保留 EEM ポリシーを表示します。
ステップ 3	**event manager scheduler hold** {**all**| **policy** *job-id*	**queue-type** {**applet**	**call-home**	**axp**	**script**} **class** *class-options*} [**processor** {**rp_primary**| **rp_standby**}] 例： `Device# event manager scheduler hold policy 2`	EEM スケジューラで、スケジュールされた EEM ポリシー イベントまたはイベント キューをホールドします。 • この例では、ジョブ ID 2 のポリシーがホールドされます。
ステップ 4	**show event manager policy pending** [**queue-type**{**applet**	**call-home**	**axp**	**script**} **class** *class-options*	**detailed** 例： `Device# show event manager policy pending`	他の保留ポリシーとともに、手順 3 でホールドされた EEM ポリシーのステータスが Held と表示されます。

例

次に、すべての保留 EEM ポリシーの表示方法とジョブ ID 2 の EEM ポリシーをホールドする例を示します。

```
Device# show event manager policy pending
no. job id status time of event             event type     name
1   1      pend   Thu Sep 7  02:54:04 2006  syslog         applet: one
2   2      pend   Thu Sep 7  02:54:04 2006  syslog         applet: two
3   3      pend   Thu Sep 7  02:54:04 2006  syslog         applet: three
Device# event manager scheduler hold policy 2
```

```
Device# show event manager policy pending

no. job id status time of event             event type     name
1   1      pend   Thu Sep 7  02:54:04 2006  syslog         applet: one
2  2      held   Thu Sep 7  02:54:04 2006  syslog         applet: two
3   3      pend   Thu Sep 7  02:54:04 2006  syslog         applet: three
```

EEM ポリシー イベントまたはイベント キューの実行の再開

EEM ポリシー イベントまたはイベント キューの実行を再開するには、次の作業を実行します。このタスクでは、スケジュール済み EEM ポリシー イベントまたはイベント キューの保留で保留状態となっていたポリシーは、実行を再開できるようになっています。

手順の概要

1. **enable**
2. **show event manager policy pending**
3. **event manager scheduler release** {**all** | **policy** *policy-id* | **queue-type** {**applet** | **call-home** | **axp** | **script**}} **class** *class-options* [**processor** {*rp_primary* | *rp_standby*}]
4. **show event manager policy pending**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager policy pending** 例： `Device# show event manager policy pending`	保留およびホールドされた EEM ポリシーを表示します。 （注） この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS Network Management Command Reference』を参照してください。
ステップ 3	**event manager scheduler release** {**all** | **policy** *policy-id* | **queue-type** {**applet** | **call-home** | **axp** | **script**}} **class** *class-options* [**processor** {*rp_primary* | *rp_standby*}] 例： `Device# event manager scheduler release policy 2`	指定された EEM ポリシーの実行を再開します。 • 例では、ジョブ ID 2 のポリシーの実行を再開する方法を示しています。
ステップ 4	**show event manager policy pending** 例：	他の保留ポリシーとともに、手順 3 で再開された EEM ポリシーの状態が pending と表示されます。

	コマンドまたはアクション	目的
	`Device# show event manager policy pending`	（注） この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS Network Management Command Reference』を参照してください。

例

次に、すべての保留 EEM ポリシーの表示方法、および実行を再開するポリシーを指定する方法、ポリシーが保留状態に戻っていることを確認する例を示します。

```
Device# show event manager policy pending

no. job id status time of event             event type     name
1   1      pend   Thu Sep 7  02:54:04 2006  syslog         applet: one
2  2      held   Thu Sep 7  02:54:04 2006  syslog         applet: two
3   3      pend   Thu Sep 7  02:54:04 2006  syslog         applet: three
Rotuer# event manager scheduler release policy 2
Rotuer# show event manager policy pending
no. job id status time of event             event type     name
1   1      pend   Thu Sep 7  02:54:04 2006  syslog         applet: one
2   2      pend   Thu Sep 7  02:54:04 2006  syslog         applet: two
3   3      pend   Thu Sep 7  02:54:04 2006  syslog         applet: three
```

保留 EEM ポリシー イベントまたはイベント キューのクリア

実行中または実行を保留中の EEM ポリシー イベントをクリアするには、次の作業を実行します。このタスクでは、ジョブ ID 2 のポリシーが保留キューからクリアされます。ポリシーがクリアされる前後に保留中のポリシーを表示するには、**show event manager policy pending** コマンドを使用します。

手順の概要

1. **enable**
2. **show event manager policy pending**
3. **event manager scheduler clear** {**all** | **policy** *job-id* | **queue-type** {**applet** | **call-home** | **axp** | **script**} **class** *class-options*} [**processor** {**rp_primary** | **rp_standby**}]
4. **show event manager policy pending**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**show event manager policy pending** 例： `Device# show event manager policy pending`	保留 EEM ポリシーを表示します。 （注） この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS Network Management Command Reference』を参照してください。
ステップ 3	**event manager scheduler clear** {**all** | **policy** *job-id* | **queue-type** {**applet** | **call-home** | **axp** | **script**} **class** *class-options*} [**processor** {**rp_primary** | **rp_standby**}] 例： `Device# event manager scheduler clear policy 2`	実行中または実行を保留中の EEM ポリシーをクリアします。 • この例では、ジョブ ID 2 のポリシーが保留キューからクリアされます。
ステップ 4	**show event manager policy pending** 例： `Device# show event manager policy pending`	手順 3 でクリアされたポリシーを除く、保留中のすべての EEM ポリシーを表示します。 （注） この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS Network Management Command Reference』を参照してください。

例

次に、実行を保留されたジョブ ID 2 のポリシーをクリアする例を示します。ポリシーがクリアされる前後に保留中のポリシーを表示するには、**show** コマンドを使用します。

```
Device# show event manager policy pending
no. job id status time of event             event type     name
1   1      pend   Thu Sep 7  02:54:04 2006  syslog         applet: one
2   2      pend   Thu Sep 7  02:54:04 2006  syslog         applet: two
3   3      pend   Thu Sep 7  02:54:04 2006  syslog         applet: three

Device# event manager scheduler clear policy 2
Device# show event manager policy pending

no. job id status time of event             event type     name
1   1      pend   Thu Sep 7  02:54:04 2006  syslog         applet: one
3   3      pend   Thu Sep 7  02:54:04 2006  syslog         applet: three
```

EEM ポリシー イベントまたはイベント キューのスケジューリング パラメータの変更

EEM ポリシー イベントのスケジューリング パラメータを変更するには、次の作業を実行します。**show event manager policy pending** コマンドは、B またはデフォルトクラスに割り当てられているポリシーを表示します。現在保留されているすべてのポリシーがクラス A に変更され

ます。設定変更後、**show event manager policy pending** コマンドはクラス A として割り当てられているすべてのポリシーを表示します。

手順の概要

1. **enable**
2. **show event manager policy pending**
3. **event manager scheduler modify** {**all** | **policy** *job-id* | **queue-type** {**applet** | **call-home** | **axp** | **script**} | **class** *class-options*} [**queue-priority** {**high** | **last** | **low** | **normal**}][**processor** {**rp_primary** | **rp_standby**}]
4. **show event manager policy pending**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager policy pending** 例： `Device# show event manager policy pending`	保留 EEM ポリシーを表示します。 （注） この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS Network Management Command Reference』を参照してください。
ステップ 3	**event manager scheduler modify** {**all** | **policy** *job-id* | **queue-type** {**applet** | **call-home** | **axp** | **script**} | **class** *class-options*} [**queue-priority** {**high** | **last** | **low** | **normal**}][**processor** {**rp_primary** | **rp_standby**}] 例： `Device# event manager scheduler modify all class A`	EEM ポリシーのスケジューリング パラメータを変更します。 • この例では、現時点での保留 EEM ポリシーはすべてクラス A に割り当てられています。
ステップ 4	**show event manager policy pending** 例： `Device# show event manager policy pending`	他の保留ポリシーとともに、手順 3 で変更された EEM ポリシーが表示されます。 （注） この例では、このタスクに適用可能な構文だけが使用されています。詳細については、『Cisco IOS Network Management Command Reference』を参照してください。

例

次に、EEMポリシーのスケジューリングパラメータを変更する例を示します。この例では、**show event manager policy pending** コマンドは、B またはデフォルトクラスに割り当てられているポリシーを表示します。現在保留されているすべてのポリシーがクラス A に変更されます。設定変更後、**show event manager policy pending** コマンドはクラス A として現在割り当てられているすべてのポリシーを確認します。

```
Device# show event manager policy pending
no. class   status time of event             event type     name
1   default pend   Thu Sep 7  02:54:04 2006  syslog         applet: one
2   default pend   Thu Sep 7  02:54:04 2006  syslog         applet: two
3   B       pend   Thu Sep 7  02:54:04 2006  syslog         applet: three

Device# event manager scheduler modify all class A
Device# show event manager policy pending

no. class status time of event             event type     name
1   A     pend   Thu Sep 7  02:54:04 2006  syslog         applet: one
2   A     pend   Thu Sep 7  02:54:04 2006  syslog         applet: two
3   A     pend   Thu Sep 7  02:54:04 2006  syslog         applet: three
```

クラスベースのアクティブ EEM ポリシーの確認

アクティブな EEM ポリシーか、または実行中の EEM ポリシーを確認するには、**show event manager policy active** コマンドを使用します。

手順の概要

1. **show event manager policy active** [**queue-type** {**applet**| **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]

手順の詳細

show event manager policy active [**queue-type** {**applet**| **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]

このコマンドは、実行中の EEM ポリシーだけを表示します。このコマンドには、オプションの **class** キーワード、**detailed** キーワード、および **queue-type** キーワードが含まれています。次に、このコマンドの出力例を示します。

例：

```
Device# show event manager policy active
no. job id p s status time of event event type name
1 12598 N A running Mon Oct29 20:49:37 2007 timer watchdog loop.tcl
2 12609 N A running Mon Oct29 20:49:42 2007 timer watchdog loop.tcl
3 12620 N A running Mon Oct29 20:49:46 2007 timer watchdog loop.tcl
4 12650 N A running Mon Oct29 20:49:59 2007 timer watchdog loop.tcl
5 12842 N A running Mon Oct29 20:51:13 2007 timer watchdog loop.tcl
default class - 6 applet events
no. job id p s status time of event event type name
1 15852 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
```

```
2 15853 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
3 15854 N A running Mon Oct29 21:11:10 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
4 15855 N A running Mon Oct29 21:11:10 2007 timer watchdog WDOG_SYSLG_CNTR_TRACK_INTF_APPL
5 15856 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
6 15858 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
```

クラスベースのアクティブ EEM ポリシーの確認

アクティブな EEM ポリシーか、または実行中の EEM ポリシーを確認するには、**show event manager policy active** コマンドを使用します。

手順の概要

1. **show event manager policy active** [**queue-type** {**applet**| **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]

手順の詳細

show event manager policy active [**queue-type** {**applet**| **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]

このコマンドは、実行中の EEM ポリシーだけを表示します。このコマンドには、オプションの **class** キーワード、**detailed** キーワード、および **queue-type** キーワードが含まれています。次に、このコマンドの出力例を示します。

例：

```
Device# show event manager policy active
no. job id p s status time of event event type name
1 12598 N A running Mon Oct29 20:49:37 2007 timer watchdog loop.tcl
2 12609 N A running Mon Oct29 20:49:42 2007 timer watchdog loop.tcl
3 12620 N A running Mon Oct29 20:49:46 2007 timer watchdog loop.tcl
4 12650 N A running Mon Oct29 20:49:59 2007 timer watchdog loop.tcl
5 12842 N A running Mon Oct29 20:51:13 2007 timer watchdog loop.tcl
default class - 6 applet events
no. job id p s status time of event event type name
1 15852 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
2 15853 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
3 15854 N A running Mon Oct29 21:11:10 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
4 15855 N A running Mon Oct29 21:11:10 2007 timer watchdog WDOG_SYSLG_CNTR_TRACK_INTF_APPL
5 15856 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
6 15858 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
```

保留 EEM ポリシーの確認

実行が保留中の EEM ポリシーを確認するには、**show event manager policy pending** コマンドを使用します。EEM クラス ベースのスケジュール オプションを指定するには、オプションのキーワードを使用します。

手順の概要

1. **show event manager policy pending** [**queue-type** {**applet**| **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]

手順の詳細

show event manager policy pending [**queue-type** {**applet**| **call-home** | **axp** | **script**} **class** *class-options* | **detailed**]

このコマンドは、保留中の EEM ポリシーのみを表示します。このコマンドには、オプションの **class** キーワード、**detailed** キーワード、および **queue-type** キーワードが含まれています。次に、このコマンドの出力例を示します。

例：

```
Device# show event manager policy pending
no. job id p s status time of event event type name
1    12851   N A    pend    Mon Oct29   20:51:18 2007   timer watchdog   loop.tcl
2    12868   N A    pend    Mon Oct29   20:51:24 2007   timer watchdog   loop.tcl
3    12873   N A    pend    Mon Oct29   20:51:27 2007   timer watchdog   loop.tcl
4    12907   N A    pend    Mon Oct29   20:51:41 2007   timer watchdog   loop.tcl
5    13100   N A    pend    Mon Oct29   20:52:55 2007   timer watchdog   loop.tcl
```

EEM アプレット（インタラクティブ CLI）サポートの設定

同期アプレットは、2 つのコマンド、**action gets** および **action puts** を使用してローカルコンソール（tty）との連携をサポートするように拡張されました。これらのコマンドによってコンソールへの直接入力と表示が可能です。同期アプレットの出力は、System Logger をバイパスします。ローカル コンソールは、アプレットによって開かれ、対応する同期イベント ディテクタ pty によってサービスされます。同期出力は、開かれたコンソールに向けられます。

同期 EEM アプレットのアクティブ コンソールからの入力の読み取りと書き込み

次のタスクを使用して、EEM アプレットのインタラクティブ CLI サポートを実装します。

アクティブなコンソールからの入力の読み取り

同期ポリシーがトリガーされたとき、関連するコンソールがパブリッシュ情報仕様に格納されます。ポリシー ディテクタは、この情報を event_reqinfo コール内で問い合わせ、**action gets** コマンドで使用するために与えられたコンソール情報を格納します。

action gets コマンドは、アクティブコンソールからの入力の 1 行を読み、入力を変数に格納します。後続の改行文字は戻されません。

手順の概要

1. **enable**
2. **configure terminal**

3. **event manager applet** *applet-name*
4. **event none**
5. **action** *label* **gets** *variable*
6. **action** *label* **syslog** [**priority** *priority-level*] **msg** *msg-text*
7. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet action`	EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**event none** 例： `Device(config-applet)# event none`	EEM に登録して手動で起動される EEM ポリシーを指定します。
ステップ 5	**action** *label* **gets** *variable* 例： `Device(config-applet)# action label2 gets input`	EEM アプレットがトリガーされたときに、同期アプレットのローカル コンソールから入力を取得し、与えられた変数に値を格納します。
ステップ 6	**action** *label* **syslog** [**priority** *priority-level*] **msg** *msg-text* 例： `Device(config-applet)# action label3 syslog msg "Input entered was \"$input\""`	EEM アプレットがトリガーされたときに実行されるアクションを指定します。 • この例では、実行されるアクションは手順 5 で指定された変数の値を syslog に書き込むことです。
ステップ 7	**exit** 例： `Device(config-applet)# exit`	アプレット コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、同期アプレットのローカルttyから入力を取得して値を格納する例を示します。

```
Device(config)# event manager applet action
Device(config-applet)# event none
Device(config-applet)# action label2 gets input
Device(config-applet)#  action label3 syslog msg "Input entered was \"$input\""
```

アクティブなコンソールへの入力の書き込み

同期ポリシーがトリガーされたとき、関連するコンソールがパブリッシュ情報仕様に格納されます。ポリシーディテクタは、この情報をevent_reqinfoコール内で問い合わせ、**action puts**コマンドで使用するために与えられたコンソール情報を格納します。

action puts コマンドは、アクティブコンソールに文字列を書き込みます。**nonewline** キーワードが指定されない限り、改行文字が表示されます。同期アプレットの**action puts**コマンドからの出力は、直接コンソールに表示され、System Logger をバイパスします。非同期アプレットの **action puts** コマンドの出力は、System Logger に向けられます。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event none**
5. **action** *label* **regexp** *string-pattern string-input* [*string-match* [*string-submatch1*] [*string-submatch2*] [*string-submatch3*]]
6. **action** *label* **puts** [**nonewline**] *string*
7. **exit**
8. **event manager run** *applet-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例：	EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`Device(config)# event manager applet action`	
ステップ 4	**event none** 例： `Device(config-applet)# event none`	EEM に登録して手動で起動される EEM ポリシーを指定します。
ステップ 5	**action** *label* **regexp** *string-pattern string-input* [*string-match* [*string-submatch1*] [*string-submatch2*] [*string-submatch3*]] 例： `Device(config-applet)# action 1 regexp "(.*) (.*) (.*)" "one two three" _match _sub1`	EEM アプレットがトリガーされたときに入力文字列の正規表現パターンと比較するアクションを指定します。
ステップ 6	**action** *label* **puts** [**nonewline**] *string* 例： `Device(config-applet)# action 2 puts "match is $_match"`	EEM アプレットがトリガーされたときにデータを直接ローカルコンソールに出力するアクションを指定します。 • **nonewline** キーワードはオプションであり、改行文字を表示しないために使用します。
ステップ 7	**exit** 例： `Device(config-applet)# exit`	アプレット コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 8	**event manager run** *applet-name* 例： `Device# event manager run action`	登録された EEM ポリシーを手動で実行します。 • この例では、手順 3 で登録されたポリシーがトリガーされ、手順 5 および手順 6 で指定された、関連付けられたアクションが実行されます。

例

次に、**action puts** コマンドがデータを直接ローカルコンソールに出力する例を示します。

```
Device(config-applet)# event manager applet puts
Device(config-applet)# event none
Device(config-applet)# action 1 regexp "(.*) (.*) (.*)" "one two three" _match _sub1
Device(config-applet)# action 2 puts "match is $_match"
Device(config-applet)# action 3 puts "submatch 1 is $_sub1"
Device# event manager run puts
match is one two three
submatch 1 is one
```

SNMP ライブラリ拡張の設定

リリースに応じて、SNMP ライブラリ拡張機能で次の設定を実行できます。

前提条件

この機能を使用するには、Cisco IOS Release12.4(22)T 以降のリリースを実行している必要があります

SNMP Get および Set オペレーション

SNMP ライブラリ拡張機能により、EEM アプレットの **action info** コマンドと Tcl の **sys_reqinfo_snmp** コマンドが拡張され、SNMP の get-one、get-next、getid および set-any オペレーションのための機能が追加されます。

SNMP Get オペレーション

SNMP イベント マネージャは SNMP get オペレーションを実行して、管理対象オブジェクトの 1 つ以上の変数を取得します。**action info type snmp oid get-type** コマンドと **action info type snmp getid** コマンドを使用すると、取得する変数とエージェントの IP アドレスを指定して SNMP get 要求を送信するように SNMP イベントマネージャを設定できます。

たとえば、OID の値が 1.3.6.1.2.1.1.1 である変数を取得する場合、変数値、1.3.6.1.2.1.1.1 を指定する必要があります。指定された値が一致しない場合、トラップが生成され、エラーメッセージが syslog 履歴に書き込まれます。

action info type snmp oid get-type コマンドは、実行する get オペレーションのタイプを指定します。正確な変数を取得するには、get オペレーションのタイプを **exact** に指定する必要があります。指定された OID 値の辞書順での後続値を取得するには、get オペレーションのタイプを **next** に設定する必要があります。

次の表に、SNMP get オペレーションから取得された値が保存される組み込み変数を示します。

表 *181 : action info type snmp oid* コマンドの組み込み変数

組み込み変数	説明
_info_snmp_oid	SNMP オブジェクト ID。
_info_snmp_value	割り当てられた SNMP データ エレメントの値文字列。

GetID の動作

action info type snmp getid コマンドは SNMP エンティティから次の変数を取得します。

- sysDescr.0
- sysObjectID.0
- sysUpTime.0
- sysContact.0

- sysName.0
- sysLocation.0

次の表に、SNMP getID オペレーションから取得された値が保存される組み込み変数を示します。

表 *182 : action info type snmp getid* コマンドの組み込み変数

組み込み変数	説明
_info_snmp_syslocation_oid	sysLocation 変数の OID 値。
_info_snmp_syslocation_value	sysLocation 変数の値文字列。
_info_snmp_sysdescr_oid	sysDescr 変数の OID 値。
_info_snmp_sysdescr_value	sysDescr 変数の値文字列。
_info_snmp_sysobjectid_oid	sysObjectID 変数の OID 値。
_info_snmp_sysobjectid_value	sysObjectID 変数の値文字列。
_info_snmp_sysuptime_oid	sysUptime 変数の OID 値。
_info_snmp_sysuptime_value	sysUptime 変数の値文字列。
_info_snmp_syscontact_oid	sysContact 変数の OID 値。
_info_snmp_syscontact_value	sysContact 変数の値文字列。

get オペレーション要求は、ローカル ホストとリモート ホストの両方に送信できます。

SNMP Set オペレーション

MIB ビューでは、すべての SNMP 変数にデフォルト値が割り当てられています。SNMP イベント マネージャは、set オペレーションによってこれらの MIB 変数の値を変更できます。set オペレーションは、読み取りと書き込みアクセスが許可されたシステムでだけ実行できます。

set オペレーションを実行するには、変数のタイプと変数に割り当てられる値を指定する必要があります。

次の表に、有効な OID タイプと各 OID タイプの値を示します。

表 *183 : set* オペレーションの *OID* タイプおよび値

OID タイプ	説明
counter32	最小値が 0 の 32 ビットの数値。最大値に到達すると、カウンタが 0 にリセットされます。0 ～ 4294967295 の範囲の整数値が有効です。

OID タイプ	説明
gauge	最小値が 0 の 32 ビットの数値。たとえば、gauge オブジェクト タイプを使用して、デバイス上のインターフェイスの速度を測定できます。0 ～ 4294967295 の範囲の整数値が有効です。
integer	管理対象オブジェクトのコンテキスト内の番号が付けられたタイプを指定する場合は、32 ビットの数字が使用されます。たとえば、デバイス インターフェイスの動作ステータスを 1 に設定した場合はアップ、2 に設定した場合はダウンを示します。0 ～ 4294967295 の範囲の整数値が有効です。
ipv4	IP バージョン 4 アドレス。ドット付き 10 進表記の IPv4 アドレスが有効です。
octet string	物理アドレスを表すために使用される、16 進表記のオクテット文字列。テキスト文字列が有効です。
string	テキスト文字列を表すために使用される、テキスト表記のオクテット文字列。テキスト文字列が有効です。
unsigned32	10 進の値を表すために使用される、32 ビットの数値。0 ～ 4294967295 の範囲の符号なし整数値が有効です。

set オペレーションは、ローカル ホストとリモート ホストの両方で実行できます。

SNMP トラップ要求および通知要求

トラップは、SNMP マネージャまたは NMS にネットワーク状態を警告する SNMP 通知です。

SNMP インフォーム要求は、SNMP マネージャにネットワーク状態を警告する SNMP 通知を参照し、SNMP マネージャからの受信の確認を要求します。

SNMP イベントは、SNMP MIB オブジェクト ID 値がサンプリングされたとき、または、SNMP カウンタが定義されたしきい値を超えたときに発生します。通知がイネーブルであり、該当するイベントが設定されている場合、SNMP トラップまたはインフォーム メッセージが生成されます。イベント マネージャ サーバーによって SNMP トラップまたはインフォーム メッセージが受信されたとき、SNMP 通知イベントがトリガーされます。

Embedded Event Manager（EEM）アプレットがトリガーされたときに SNMP トラップまたは通知メッセージを送信するには、 **action info type snmp trap** コマンドと **action info type snmp**

inform コマンドを使用します。CISCO-EMBEDDED-EVENT-MGR-MIB.my を使用して、トラップおよびインフォーム メッセージが定義されます。

SNMP Get および Set オペレーションの EEM Applet 設定

ポリシーをイベント マネージャ サーバーに登録する一方で、SNMP イベントに関連付けられたアクションを設定できます。

SNMP set および get オペレーションの EEM アプレットを設定するには、次の作業を実行します。

始める前に

- SNMP イベントマネージャは、**snmp-server manager** コマンドを使用して設定する必要があります。
- SNMP エンティティへのアクセスを有効にするためには、**snmp-server community** コマンドを使用して、SNMP コミュニティストリングを設定する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. 次のいずれかを実行します。
   - **event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** *operator* **entry-val** *entry-value*[**exit-comb** | **and**}] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*
5. **action label info type snmp oid** *oid-value* **get-type** {**exact**| **next**} [**community** *community-string*] [**ipaddr** *ip-address*]
6. **action label info type snmp oid** *oid-value* **set-type** *oid-type* *oid-type-value* **community** *community-string* [**ipaddr** *ip-address*]
7. **action label info type snmp getid** *oid-value* [**community** *community-string*] [**ipaddr** *ip-address*]
8. **exit**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet snmp`	Event Manager にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	次のいずれかを実行します。 • **event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** *operator* **entry-val** *entry-value*[**exit-comb** | **and**}] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value* 例： `Device(config-applet)# event snmp oid` 例： `1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact` 例： `entry-op lt entry-val 5120000 poll-interval 90`	EEM アプレットの実行の原因となる、イベント基準を指定します。 • この例では、空きメモリの値が 5120000 を下回ったときに EEM イベントがトリガーされます。 • 終了基準はオプションです。指定されない場合、イベントのモニターリングは、すぐに再び有効になります。
ステップ 5	**action label info type snmp oid** *oid-value* **get-type** {**exact**| **next**} [**community** *community-string*] [**ipaddr** *ip-address*] 例： `Device(config-applet)# action 1.3 info type` 例： `snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type` 例： `exact community public ipaddr 172.17.16.69`	実行する get オペレーションのタイプを指定します。 • この例では、get オペレーションのタイプが exact と指定され、コミュニティ ストリングが public と指定されます。
ステップ 6	**action label info type snmp oid** *oid-value* **set-type** *oid-type oid-type-value* **community** *community-string* [**ipaddr** *ip-address*] 例： `Device(config-applet)# action 1.4 info type` 例： `snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 set-type` 例：	（任意）設定される変数を指定します。 • この例では、sysName.0 変数が set オペレーションに指定され、コミュニティ ストリングに rw が指定されます。 （注） set オペレーションの場合、SNMP コミュニティ ストリングを指定する必要があります。

	コマンドまたはアクション	目的
	`integer 42220 sysName.0 community rw ipaddr` 例： `172.17.16.69`	
ステップ 7	**action label info type snmp getid** *oid-value* [**community** *community-string*] [**ipaddr** *ip-address*] 例： `Device(config-applet)# action 1.3 info type` 例： `snmp getid community public ipaddr 172.17.16.69`	（任意）個々の変数が getid オペレーションによって取得される必要があるかどうかを指定します。
ステップ 8	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

SNMP OID 通知の EEM アプレットの設定

SNMP 通知を設定するには、次の作業を実行します。

始める前に

- SNMP イベントマネージャを、**snmp-server manager** コマンドを使用して設定し、SNMP エージェントが EEM ポリシーのために生成された SNMP トラップを送受信するように設定する必要があります。
- SNMP トラップとインフォームを **snmp-server enable traps event-manager** および **snmp-server enable traps** コマンドを使用して有効にして、トラップ要求とインフォーム要求をデバイスからイベントマネージャサーバーに送信できるようにする必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. 次のいずれかを実行します。
   - **event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** *operator* **entry-val** *entry-value*[**exit-comb** | **and**}] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*
5. **action label info type snmp var** *variable-name* **oid** *oid-value* *oid-type oid-type-value*

6. **action label info type snmp trap enterprise-oid** *enterprise-oid-value* **generic-trapnum** *generic-trap-number* **specific-trapnum** *specific-trap-number* **trap-oid** *trap-oid-value* **trap-var** *trap-variable*
7. **action label info type snmp inform trap-oid** *trap-oid-value* **trap-var** *trap-variable* **community** *community-string* **ipaddr** *ip-address*
8. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet snmp`	Event Manager にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	次のいずれかを実行します。 • **event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** *operator* **entry-val** *entry-value*[**exit-comb** | **and**}] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value* 例： `Device(config-applet)# event snmp oid` 例： `1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact` 例： `entry-op lt entry-val 5120000 poll-interval 90`	EEM アプレットの実行の原因となる、イベント基準を指定します。 • この例では、空きメモリの値が 5120000 を下回ったときに EEM イベントがトリガーされます。 • 終了基準はオプションです。指定されない場合、イベントのモニターリングは、すぐに再び有効になります。
ステップ 5	**action label info type snmp var** *variable-name* **oid** *oid-value* *oid-type oid-type-value* 例： `Device(config-applet)# action 1.3 info type` 例：	管理対象オブジェクトのインスタンスとその値を指定します。 • この例では、sysDescr.0 変数が使用されています。

	コマンドまたはアクション	目的
	`snmp var sysDescr.0 oid` 例： `1.3.6.1.4.1.9.9.48.1.1.1.6.1 integer 4220`	
ステップ 6	**action label info type snmp trap enterprise-oid** *enterprise-oid-value* **generic-trapnum** *generic-trap-number* **specific-trapnum** *specific-trap-number* **trap-oid** *trap-oid-value* **trap-var** *trap-variable* 例： `Device(config-applet)# action 1.4 info type` 例： `snmp trap enterprise-oid 1.3.6.1.4.1.1` 例： `generic-trapnum 4 specific-trapnum 7 trap-oid` 例： `1.3.6.1.4.1.226.0.2.1 trap-var sysUpTime.0`	EEM アプレットがトリガーされたときに SNMP トラップを生成します。 • この例では、authenticationFailure トラップが生成されます。 （注） 固有のトラップ番号は、enterprise イベントが発生したときに生成される enterprise-specific トラップを示します。標準トラップ番号が 6 に設定されていない場合、指定した固有のトラップ番号がトラップの生成に使用されます。
ステップ 7	**action label info type snmp inform trap-oid** *trap-oid-value* **trap-var** *trap-variable* **community** *community-string* **ipaddr** *ip-address* 例： `Device(config-applet)# action 1.4 info type` 例： `snmp inform trap-oid 1.3.6.1.4.1.1.226.0.2.1` 例： `trap-var sysUpTime.0 community public ipaddr` 例： `172.69.16.2`	EEM アプレットがトリガーされたときに SNMP インフォーム要求を生成します。 • この例では、sysUpTime.0 変数のインフォーム要求が生成されます。
ステップ 8	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権モードに戻ります。

EEM アプレットの可変ロジックの設定

EEM アプレットの可変ロジック機能は、EEM アプレット内に条件付きロジックを適用する機能を追加します。アプレットには、可変ロジックが導入される前は、イベントがトリガーされたときに各アクションが設定された順に実行されるリニア構造だけがありました。条件付きロジックは、アプレット内のアクションのフローを条件式に従って変更する制御構造を追加します。各制御構造には、ループ アクションや、構造を実行するかどうかを決定する if/else アクションを含むアプレット アクションのリストが含まれます。

アプレットコンフィギュレーションモードの情報は、action コマンドの内容を設定するための背景として示されます。

Tool Command Language（Tcl）とアプレット（CLI）ベースの EEM ポリシーの間で一貫したユーザー インターフェイスを実現するには、次の基準に従います。

- Tcl ベースの実装では、イベント仕様基準は TCL で記述されます。
- アプレット ベースの実装では、イベント仕様データは CLI アプレット サブモード コンフィギュレーション文を使用して記述されます。

アプレット コンフィギュレーション モードは、event manager applet コマンドを使用して開始します。アプレットコンフィギュレーションモードでは、config プロンプトが、(config-applet)# に変わります。アプレット コンフィギュレーション モードでは、2 種類のコンフィギュレーション文がサポートされます。

- event：アプレットが実行される原因となるイベント基準を指定するために使用します。
- action：実行する組み込みアクションを指定するために使用します。

1 つのアプレットコンフィギュレーション内で複数の **action** アプレットコンフィギュレーション コマンドを使用できます。**action** アプレット コンフィギュレーション コマンドが存在しない場合は、終了時に、このアプレットに文が割り当てられていないことを示す警告が表示されます。このアプレットに文が割り当てられない場合、イベントはトリガーされますが、アクションは実行されません。アプレット コンフィギュレーション モードでコマンドが指定されない場合は、終了時にアプレットが削除されます。exit applet config コマンドは、アプレットコンフィギュレーション モードを終了するために使用されます。

リリースに応じて、EEM アプレットの可変ロジック機能は次の設定を実行できます。

前提条件

この機能を使用するには、Cisco IOS Release12.4(22)T 以降のリリースを実行している必要があります

EEM アプレットの可変ロジックの設定

EEM 3.0 は、アプレット内で単純な可変ロジックを可能にするための新しいアプレット action コマンドを追加しました。

action コマンドを使用して可変ロジックを設定するには、次の作業を実行します。

条件付きブロックのループの指定

EEM アプレットがトリガーされたときに、条件付きブロックのループを指定するには、次の作業を実行します。次のタスクでは、変数の値が 10 よりも小さいかどうかを確認するために、条件付きループが設定されます。変数の値が 10 よりも小さい場合は、メッセージ「i is $_i」が syslog に書き込まれます。

（注） リリースに応じて、**set**（EEM）コマンドは **action set** コマンドに置き換えられます。詳細については、**action** *label* **set** コマンドを参照してください。特定のリリースで set（EEM）コマンドを入力した場合、IOS パーサーは **set** コマンドを **action** *label* **set** コマンドに変換します。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action** *label* **set**
5. **action** *label* **while** *string_op1 operator string_op2*
6. 必要に応じてアクションを追加します。
7. **action** *label* **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet condition`	Embedded Event Manager（EEM）にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**action** *label* **set** 例： `Device(config-applet)#action 1.0 set i 2`	イベントに対するアクションを設定します。 • この例では、変数 i の値が 2 に設定されます。

	コマンドまたはアクション	目的
ステップ 5	**action** *label* **while** *string_op1 operator string_op2* 例： `Device(config-applet)# action 2 while $i lt 10`	条件付きブロックのループを指定します。 • この例では、変数 i の値が 10 よりも小さいかどうかを確認するために、ループが設定されます。
ステップ 6	必要に応じてアクションを追加します。 例： `Device(config-applet)# action 3 syslog msg "i is $i"`	action コマンドで指示されたアクションを実行します。 • この例では、メッセージ「i is $_i」が syslog に書き込まれます。
ステップ 7	**action** *label* **end** 例： `Device(config-applet)# action 3 end`	実行中のアクションを終了します。

if else 条件付きブロックの指定

if 条件付き文の開始とそれに続く else 条件付き文を指定するには、次の作業を実行します。if 条件付き文と else 条件付き文は、それぞれを結合して使用することも、別々に使用することもできます。このタスクでは、変数の値が 5 に設定されます。次に、変数の値が 10 よりも小さいかどうかを確認するために、if 条件付きブロックが指定されます。if 条件付きブロックが満たされる場合にメッセージ「x is less than 10」を出力する action コマンドが指定されます。

if 条件付きブロックに続いて、else 条件付き部位ロックが指定されます。if 条件付きブロックが満たされない場合にメッセージ「x is greater than 10」を出力する action コマンドが指定されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action** *label* **set** *variable-name variable-value*
5. **action** *label* **if** [*stringop1*] {**eq** | **gt** | **ge** | **lt** | **le** | **ne**} [*stringop2*]
6. 必要に応じてアクションを追加します。
7. **action** *label* **else**
8. 必要に応じてアクションを追加します。
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet ifcondition`	Embedded Event Manager（EEM）にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**action** *label* **set** *variable-name variable-value* 例： `Device(config-applet)# action 1.0 set x 5`	イベントに対するアクションを設定します。 • この例では、変数 x の値が 5 に設定されます。
ステップ 5	**action** *label* **if** [*stringop1*] {**eq** | **gt** | **ge** | **lt** | **le** | **ne**} [*stringop2*] 例： `Device(config-applet)# action 2.0 if $x lt 10`	if 条件付き文を指定します。 • この例では、if条件付き文は変数の値が 10 よりも小さいかどうかを確認します。
ステップ 6	必要に応じてアクションを追加します。 例： `Device(config-applet)# action 3.0 puts "$x is less than 10"`	action コマンドで指示されたアクションを実行します。 • この例では、メッセージ「5 is less than 10」が画面に表示されます。
ステップ 7	**action** *label* **else** 例： `Device(config-applet)# action 4.0 else`	else 条件付きステートメントを指定します。
ステップ 8	必要に応じてアクションを追加します。 例： `Device(config-applet)# action 5.0`	action コマンドで指示されたアクションを実行します。 • この例では、メッセージ「5 is greater than 10」が画面に表示されます。

	コマンドまたはアクション	目的
ステップ 9	**end** 例： `Device(config-applet)# end`	実行中のアクションを終了します。

foreach 反復文の指定

デリミタをトークン化パターンとして使用して入力文字列上で繰り返す条件付き文を指定するには、次の作業を実行します。foreach反復文は目的の情報を取得するためにコレクションを使用して繰り返すために使用されます。デリミタは、正規表現パターン文字列です。各反復で見つかったトークンは、与えられた iterator 変数に割り当てられます。すべての算術演算は、長整数としてオーバーフローのチェックなしで実行されます。この例では、変数 x の値が 5 に設定されます。反復文は、入力文字列 red、blue、green、orange の間、実行するように設定されます。入力文字列の各エレメントに対して、対応するメッセージが画面に表示されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action** *label* **foreach** [*string-iterator*] [*string-input*] [*string-delimiter*]
5. 任意の action コマンドを指定します。
6. **action** *label* **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet iteration`	Embedded Event Manager（EEM）にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 4	**action** *label* **foreach** [*string-iterator*] [*string-input*] [*string-delimiter*] 例： `Device(config-applet)# action 2.0 foreach iterator "red blue green orange"`	デリミタをトークン化パターンとして使用して、入力文字列上で繰り返します。 • この例では、入力のエレメント、red、blue、green、および、orangeの間、反復が実行されます。
ステップ 5	任意の action コマンドを指定します。 例： `Device(config-applet)# action 3.0 puts "Iterator is $iterator"`	action コマンドで指示されたアクションを実行します。 • この例では、次のメッセージが画面に表示されます。 Iterator is red Iterator is blue Iterator is green Iterator is orange
ステップ 6	**action** *label* **end** 例： `Device(config-applet)# action 4.0 end`	実行中のアクションを終了します。

正規表現の使用

正規表現パターンを入力文字列と比較するには、次の作業を実行します。正規表現を使用すると、比較される文字列として可能性のある文字列のセットを表す規則を指定できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action** *label* **regexp** *string-pattern string-input* [*string-match* [*string-submatch1*] [*string-submatch2*] [*string-submatch3*]]

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet regexp`	Embedded Event Manager（EEM）にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**action** *label* **regexp** *string-pattern string-input* [*string-match* [*string-submatch1*] [*string-submatch2*] [*string-submatch3*]] 例： `Device(config-applet)# action 2.0 regexp "(.*) (.*) (.*)" "red blue green" _match _sub1`	入力文字列と比較する表現パターンを指定します。 • この例では、「red blue green」の入力文字列が指定されます。表現パターンが入力文字列と一致すると、**red blue green** の結果全体が変数の **_match** に格納され、部分一致の **red** は変数の **_sub1** に格納されます。

変数の値の増加

変数の値を増加させるには、次の作業を実行します。このタスクでは変数の値が 20 に設定され、次に値が 12 だけ増加します。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action** *label* **set**
5. **action** *label* **increment** *variable-name long-integer*

手順の詳細

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

	コマンドまたはアクション	目的
	`Device# configure terminal`	
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet increment`	Embedded Event Manager（EEM）にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**action** *label* **set** 例： `Device(config-applet)#action 1.0 set varname 20`	イベントに対するアクションを設定します。 • この例では、変数の値が 20 に設定されます。
ステップ 5	**action** *label* **increment** *variable-name long-integer* 例： `Device(config-applet)#action 2.0 increment varname 12`	変数の値が指定された長整数だけ増加します。 • この例では、変数の値が 12 だけ増加します。

イベント SNMP オブジェクトの設定

SNMP オブジェクトのサンプリングによって実行される Embedded Event Manager（EEM）アプレットの簡易ネットワーク管理プロトコル（SNMP）オブジェクトイベントを登録するには、次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event snmp-object oid** *oid-value* **type** *value* **sync** {**yes** | **no**} **skip** {**yes** | **no**} **istable** {**yes** | **no**} [**default** *seconds*] [**maxrun** *maxruntime-number*]
5. **exit**

手順の詳細

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

	コマンドまたはアクション	目的
	`Device# configure terminal`	
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet manual-policy`	Embedded Event Manager にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**event snmp-object oid** *oid-value* **type** *value* **sync** {**yes** | **no**} **skip** {**yes** | **no**} **istable** {**yes** | **no**} [**default** *seconds*] [**maxrun** *maxruntime-number*] 例： `Device(config-applet)# event snmp-object oid 1.9.9.9.9 type gauge sync yes` 例： `action 1 syslog msg "oid = $_snmp_oid"` 例： `action 2 syslog msg "request = $_snmp_request"` 例： `action 3 syslog msg "request_type = $_snmp_request_type"`	Embedded Event Manager（EEM）アプレット用の簡易ネットワーク管理プロトコル（SNMP）オブジェクトイベントを登録し、オブジェクトの SNMP GET および SET 要求を代行受信します。 デフォルトでは、このコマンドは設定されていません。このコマンドが設定されると、デフォルトは構文オプションの説明と同一になります。 • **oid** キーワードは、SNMP オブジェクト識別子（object ID）を指定します。 • *oid-value* 引数は、SNMP ドット付き表記のデータ エレメントのオブジェクト ID 値です。OID は、関連する MIB（CISCO-EMBEDDED-EVENT-MGR-MIB）内にタイプとして定義され、各タイプはオブジェクト値を保持します。 • **istable** キーワードは、OID が SNMP テーブルかどうかを指定します。 • sync キーワードは、アプレットを同期モードで実行するよう指定します。アプレットからの戻りコードは、SNMP 要求に応答するかどうかを示します。コード 0 は「要求に応答しない」、コード 1 は「要求に応答する」を意味します。アプレットからの戻りコードが要求に応答すると、**action snmp-object-value** コマンドを使用して、オブジェクトのアプレットで値が指定されます。 • **type** キーワードは、オブジェクトのタイプを指定します。 • *value* 引数はオブジェクトの値です。 • skip キーワードは、CLI コマンドの実行をスキップするかどうかを指定します

	コマンドまたはアクション	目的
		• **default** キーワードは、アプレットが通常処理する SET 要求または GET 要求の時間を指定します。**default** キーワードが指定されない場合は、デフォルトの時間が 30 秒に設定されます。 • *milliseconds* 引数は、SNMP オブジェクト イベントディテクタがポリシーの終了を待つ時間です。 • **maxrun** キーワードは、アプレットの最大ランタイムを指定します。**maxrun** キーワードを指定した場合、*maxruntime-number* 値を指定する必要があります。**maxrun** キーワードが指定されていない場合、デフォルトのアプレットランタイムは 20 秒です。 • *milliseconds* 引数は、ミリ秒単位のアプレットの最大ランタイムです。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
ステップ 5	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

AAA 認証の無効化

トリガーされたときに、EEM ポリシーが AAA 認証をバイパスするようにするには、次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name* [**authorization bypass**] [**class** *class-options*] [**trap**]
4. **exit**

手順の詳細

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

	コマンドまたはアクション	目的
	`Device> enable`	
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* [**authorization bypass**] [**class** *class-options*] [**trap**] 例： `Device(config)# event manager applet one class A authorization bypass`	Embedded Event Manager（EEM）にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**exit** 例： `Device(config-aaplet)# exit`	デバイス コンフィギュレーション アプレット モードを終了し、特権 EXEC モードに戻ります。

Embedded Event Manager アプレットの説明の設定

EEM アプレットについて記述するには、次の作業を実行します。アプレットの説明は、他のアプレット設定の前でも後でも、任意の順序で追加できます。すでに説明があるアプレットに新しい説明を設定すると、現在の説明が上書きされます。アプレットの説明はオプションです。

アプレットに新しい説明を設定するには、次の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **description** *line*
5. **event syslog pattern** *regular-expression*
6. **action** *label* **syslog msg** *msg-text*
7. **end**

手順の詳細

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

	コマンドまたはアクション	目的
	`Device> enable`	
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**event manager applet** *applet-name* 例： `Device(config)# event manager applet increment`	EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	**description** *line* 例： `Device(config-applet)# description "This applet looks for the word count in syslog messages"`	簡易ネットワーク管理プロトコル（SNMP）のサンプリングによって実行される EEM アプレットの説明を追加または変更します。
ステップ 5	**event syslog pattern** *regular-expression* 例： `Device(config-applet)# event syslog pattern "count"`	syslog メッセージの一致によって実行される Embedded Event Manager（EEM）アプレットのイベント基準を指定します。
ステップ 6	**action** *label* **syslog msg** *msg-text* 例： `Device(config-applet)# action 1 syslog msg hi`	EEM アプレットがトリガーされたときに実行されるアクションを指定します。 • この例では、実行されるアクションは syslog にメッセージを書き込むことです。 • *msg-text* 引数は、文字テキスト、環境変数、またはその両方の組み合わせが可能です。
ステップ 7	**end** 例： `Device(config-applet)# end`	アプレット コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Tcl を使用した Embedded Event Manager（EEM）ポリシー記述の設定例

Embedded Event Manager アプレットの設定例

次に、一部の EEM イベント ディテクタの EEM アプレット作成例を示します。次の例は、Embedded Event Manager アプレットの登録と定義 （2051 ページ）で説明した手順に従っています。

Application-Specific イベント ディテクタ

次に、EventPublish_A という名前のポリシーが、20 秒ごとに実行され、番号が 1 のイベント タイプを、番号 798 のサブシステムにパブリッシュする例を示します。サブシステムの値、798 は、イベントのパブリッシュが EEM ポリシーから発生することを指定します。EventPublish_B という名前の別のポリシーは、EEM イベント タイプ 1 が発生したときに実行されるように、subsystem 798 に登録されます。EventPublish_B ポリシーは実行されるときに、EventPublish_A から引数として渡されたデータを含むメッセージを syslog に送信します。

```
event manager applet EventPublish_A
 event timer watchdog time 20.0
 action 1.0 syslog msg "Applet EventPublish_A"
 action 2.0 publish-event sub-system 798 type 1 arg1 twenty
 exit
event manager applet EventPublish_B
 event application sub-system 798 type 1
 action 1.0 syslog msg "Applet EventPublish_B arg1 $_application_data1"
```

CLI イベント ディテクタ

次に、Cisco IOS **write memory** CLI コマンドが実行されたときに実行する EEM アプレットを指定する例を示します。アプレットは、このイベントが syslog メッセージによって生成した通知を提供します。この例では、**sync** キーワードが yes 引数とともに設定されています。これは、このポリシーの実行が完了したときに、イベントディテクタに通知されることを意味します。ポリシーの終了状態が、CLI コマンドが実行されるかどうかを決定します。この例では、ポリシーの終了状態は 1 に設定され、CLI コマンドは実行されます。

```
event manager applet cli-match
 event cli pattern "write mem.*" sync yes
 action 1.0 syslog msg "$_cli_msg Command Executed"
 set 2.0 _exit_status 1
```

次に、**cli pattern** と test 引数を照合するアプレットの例を示します。**show access-list test** が入力されると、CLI イベントディテクタは、test 引数を照合し、アプレットがトリガーされます。**debug event manager detector cli** 出力が追加され、num_matches が 1 に設定されていることが示されます。

```
!
```

```
event manager applet EEM-PIPE-TEST
 event cli pattern "test" sync yes
 action 1.0 syslog msg "Pattern matched!"
!
*Aug 23 23:19:59.827: check_eem_cli_policy_handler: command_string=show access-lists
test
*Aug 23 23:19:59.827: check_eem_cli_policy_handler: num_matches = 1, response_code = 4
*Aug 23 23:19:59.843: %HA_EM-6-LOG: EEM-PIPE-TEST: Pattern matched!
```

（注） CLI イベント ディテクタによる機能は、有効な IOS CLI コマンドでの正規表現パターン比較機能だけです。これには、リダイレクションが使用される場合のパイプ記号（|）以降のテキストは含まれません。

次に、**show version | include test** が入力された場合にアプレットがトリガーされなかった例を示します。CLI イベントディテクタでパイプ（|）文字の後ろに入力された文字との一致がなく、**debug event manager detector cli** 出力で num_matches がゼロと表示されいるためにトリガーされませんでした。

```
*Aug 23 23:20:16.827: check_eem_cli_policy_handler: command_string=show version
*Aug 23 23:20:16.827: check_eem_cli_policy_handler: num_matches = 0, response_code = 1
```

Counter イベント ディテクタおよび Timer イベント ディテクタ

次に、EventCounter_A ポリシーが 1 分に 1 回実行されるように設定され、既知のカウンタ critical_errors を増加させる例を示します。2 番目のポリシー、EventCounter_B は、critical_errors がという既知のカウンタがしきい値 3 を超えたときにトリガーされるように登録されます。EventCounter_B ポリシーが実行されたとき、カウンタは 0 にリセットされます。

```
event manager applet EventCounter_A
 event timer watchdog time 60.0
 action 1.0 syslog msg “EventCounter_A”
 action 2.0 counter name critical_errors op inc value 1
 exit
event manager applet EventCounter_B
 event counter name critical_errors entry-op gt entry-val 3 exit-op lt exit-val 3
 action 1.0 syslog msg “EventCounter_B”
 action 2.0 counter name critical_errors op set value 0
```

Interface Counter イベント ディテクタ

次に、EventInterface という名前のポリシーが、ファスト イーサネット インターフェイス 0/0 の receive_throttle カウンタが 5 ずつ増加するたびに、トリガーされる例を示します。カウンタをチェックするポーリング間隔は、90 秒ごとに 1 回実行するように指定されます。

```
event manager applet EventInterface
 event interface name FastEthernet0/0 parameter receive_throttle entry-op ge entry-val
5
 entry-val-is-increment true poll-interval 90
 action 1.0 syslog msg "Applet EventInterface"
```

Resource イベント ディテクタ

次に、CPU 使用率が高い場合に報告するように定義された、ポリシーの ERM イベント レポートに基づいてイベント基準を指定する例を示します。

```
event manager applet policy-one
 event resource policy cpu-high
 action 1.0 syslog msg "CPU high at $_resource_current_value percent"
```

RF イベント ディテクタ

RF イベント ディテクタは、デュアル ルート プロセッサ（RP）を備えたネットワーキング デバイスでだけ利用できます。次に、RF 状態変化通知に基づいてイベント基準を指定する例を示します。

```
event manager applet start-rf
 event rf event rf_prog_initialization
 action 1.0 syslog msg "rf state rf_prog_initialization reached"
```

Remote Procedure Call（RPC）イベント ディテクタ

RPC イベント ディテクタによって、外部エンティティがデバイスに対して Simple Object Access Protocol（SOAP）要求を作成でき、定義された EEM ポリシーまたはスクリプトを実行できます。次に、Event_RPC という名前の EEM アプレットが EEM スクリプトを実行するように登録されている例を示します。

```
event manager applet Event_RPC
 event rpc
 action print puts "hello there"
```

次に、SOAP 要求と返信メッセージの形式の例を示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://www.cisco.com/eem.xsd">
  <SOAP:Body>
    <run_eemscript>
      <script_name>Event_RPC</script_name>
    </run_eemscript>
  </SOAP:Body>
</SOAP:Envelope>
]]>]]>
<?xml version="1.0" encoding="UTF-8"?><SOAP:Envelope
xmlns:SOAP="http://www.cisco.com/eem.xsd"><SOAP:Body>
<run_eemscript_response><return_code>0</return_code><output></output></run_eemscript_response></SOAP:Body></SOAP:Envelope>]]>]]>
```

SNMP イベント ディテクタ

次に、CPU 使用率が 75% を上回ったときに実行する EEM アプレットを指定する例を示します。EEM アプレットを実行すると、CLI コマンドの **enable** と **show cpu processes** が実行され、**show cpu processes** コマンドの結果が含まれている電子メールがエンジニアに送信されます。

```
event manager applet snmpcpuge75
 event snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.1.3.1 get-type exact entry-op ge entry-val 75
```

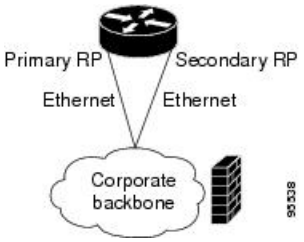
```
poll-interval 10
 action 1.0 cli command "enable"
 action 2.0 cli command "show process cpu"
 action 3.0 mail server "192.168.1.146" to "engineer@cisco.com" from "devtest@cisco.com"
subject "B25 PBX Alert" body "$_cli_result"
```

次の例はより複雑で、プライマリ ルート プロセッサ（RP）がメモリ不足で実行されているときに、セカンダリ（冗長）RP に切り替えるように EEM アプレットを設定する例を示します。

次に、メモリ リークの原因となるソフトウェア障害に対する予防措置を実施する例を示します。ここで実行されるアクションは、メモリ リークの可能性が検出されたときに、冗長 RP へ切り替えることによってダウンタイムを削減することを意図しています。

次の図は、EEM イメージを実行しているデュアル RP デバイスを示しています。EEM アプレットは、**event manager applet** コマンドを使用して CLI によって登録されています。プライマリ RP の使用可能なメモリが、指定されたしきい値 5,120,000 バイトを下回ったときに、アプレットは実行されます。アプレットのアクションは、利用可能なメモリのバイト数を示すメッセージを syslog に書き込み、セカンダリ RP へスイッチします。

図 *132*：デュアル *RP* トポロジ



ポリシーの登録に使用されるコマンドは、次のとおりです。

```
event manager applet memory-demo
 event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val 5120000
poll-interval 90
 action 1.0 syslog priority critical msg "Memory exhausted; current available memory is
$_snmp_oid_val bytes"
 action 2.0 force-switchover
```

登録済みのアプレットは、**show event manager policy registered** コマンドを使用して表示できます。

```
Device# show event manager policy registered
No.  Type    Event Type          Time Registered           Name
1    applet  snmp                Thu Jan30  05:57:16 2003  memory-demo
 oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
 action 1.0 syslog priority critical msg "Memory exhausted; current available memory is
$_snmp_oid_val bytes"
 action 2.0 force-switchover
```

この例を示すため、デバイスでメモリを強制的に枯渇させ、一連の **show memory** コマンドを実行させてメモリの枯渇を監視します。

```
Device# show memory
                Head    Total(b)     Used(b)     Free(b)   Lowest(b)  Largest(b)
Processor   53585260   212348444   119523060    92825384    92825384    92365916
```

```
Fast        53565260      131080       70360       60720       60720       60668
Device# show memory
                Head    Total(b)     Used(b)     Free(b)   Lowest(b)  Largest(b)
Processor   53585260   212364664   164509492    47855172    47855172    47169340
Fast        53565260      131080       70360       60720       60720       60668
Device# show memory
                Head    Total(b)     Used(b)     Free(b)   Lowest(b)  Largest(b)
Processor   53585260   212369492   179488300    32881192    32881192    32127556
Fast        53565260      131080       70360       60720       60720       60668
```

しきい値に達したときに、EEM イベントがトリガーされます。memory-demo という名前のアプレットが実行され、これによって、syslog メッセージがコンソールに出力され、セカンダリ RP へのスイッチが発生します。次のメッセージが記録されます。

```
00:08:31: %HA_EM-2-LOG: memory-demo: Memory exhausted; current available memory is
4484196 bytes
00:08:31: %HA_EM-6-FMS_SWITCH_HARDWARE: fh_io_msg: Policy has requested a hardware
switchover
```

次に、プライマリ RP とセカンダリ（冗長）RP の両方での **show running-config** コマンドの出力の一部を示します。

```
redundancy
 mode sso
.
.
!
event manager applet memory-demo
 event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val
5120000 poll-interval 90
 action 1.0 syslog priority critical msg "Memory exhausted; current available memory
is $_snmp_oid_val bytes"
 action 2.0 force-switchover
```

SNMP 通知イベント ディテクタ

次に、**event snmp-notification** を設定する前に、**snmp-server community** パブリック RW コマンドと **snmp-server manager** コマンドを設定する例を示します。

```
snmp-server community public RW
 snmp-server manager
```

次に、値が 10 であるオブジェクト ID 1 の宛先 IP アドレス 192.168.1.1 でデバイスが SNMP 通知を受け取ったときに、EEM スクリプトを実行するように SNMP_Notification という名前の EEM アプレットを登録する例を示します。

```
event manager applet SNMP_Notification
 event snmp-notification dest_ip_address 192.168.1.1 oid 1 op eq oid-value 10
 action 1 policy eem_script
```

syslog イベント ディテクタ

次に、syslog がイーサネット インターフェイス 1/0 のダウンを認識したときに実行する EEM アプレットを指定する例を示します。アプレットはインターフェイスに関するメッセージを syslog に送信します。

```
event manager applet interface-down
 event syslog pattern "˜.*UPDOWN.*Ethernet1/0.*" occurs 4
 action 1.0 syslog msg "Ethernet interface 1/0 changed state 4 times"
```

Embedded Event Manager アプレットの設定例

ID イベント ディテクタの例

次に、「EventIdentity」というポリシーが、ファストイーサネット インターフェイス 0 で認証が成功するたびにトリガーされる例を示します。

```
event manager applet EventIdentity
 event identity interface FastEthernet0 authc success
action 1.0 syslog msg "Applet EventIdentity"
```

MAT イベント ディテクタの例

次に、「EventMat」というポリシーが、mac-address-table で MAC アドレスが学習されるたびにトリガーされる例を示します。

```
event manager applet EventMat
 event mat interface FastEthernet0
action 1.0 syslog msg "Applet EventMat"
```

ネイバー検出イベント ディテクタの例

次に、「EventNeighbor」というポリシーが、Cisco Discovery Protocol（CDP）キャッシュ エントリが変化するときにトリガーされる例を示します。

```
event manager applet EventNeighbor
 event neighbor-discovery interface FastEthernet0 cdp all
action 1.0 syslog msg "Applet EventNeighbor"
```

Embedded Event Manager の手動によるポリシー実行の例

次に、手動で実行する EEM ポリシー（アプレットまたはスクリプト）の設定に None イベント ディテクタを使用する例を示します。

イベント マネージャ run コマンドの使用

次に、**event manager run** コマンドを使用して、手動でポリシーを実行する例を示します。ポリシーはアプレット コンフィギュレーション モードで **event none** コマンドを使用して登録されてから、グローバル コンフィギュレーション モードで **event manager run** コマンドを使用して実行されます。

```
event manager applet manual-policy
 event none
 action 1.0 syslog msg "Manual-policy triggered"
 end
```

```
!
event manager run manual-policy
```

action policy コマンドの使用

次に、**action policy** コマンドを使用して、手動でポリシーを実行する例を示します。ポリシーはアプレット コンフィギュレーション モードで **event none** コマンドを使用して登録されてから、アプレット コンフィギュレーション モードで **action policy** コマンドを使用して実行されます。

```
event manager applet manual-policy
 event none
 action 1.0 syslog msg "Manual-policy triggered"
 exit
!
event manager applet manual-policy-two
 event none
 action 1.0 policy manual-policy
 end
!
event manager run manual-policy-two
```

Embedded Event Manager Watchdog System Monitor（Cisco IOS）イベント ディテクタの設定例

次に、Cisco IOS watchdog system monitor (IOSWDSysMon) イベント ディテクタの動作を具体的に表示する 3 個の EEM アプレットの設定例を示します。

Watchdog System Monitor サンプル 1 ポリシー

第 1 のポリシーは、IP Input という名前のプロセスの平均 CPU 使用率が 10 秒間 1% 以上になったときにアプレットをトリガーします。

```
event manager applet IOSWD_Sample1
 event ioswdsysmon sub1 cpu-proc taskname "IP Input" op ge val 1 period 10
 action 1.0 syslog msg "IOSWD_Sample1 Policy Triggered"
```

Watchdog System Monitor サンプル 2 ポリシー

第 2 のポリシーは、Net Input という名前のプロセスによる合計メモリ使用量が 100 kb を超えたときアプレットをトリガーします。

```
event manager applet IOSWD_Sample2
 event ioswdsysmon sub1 mem-proc taskname "Net Input" op gt val 100 is-percent false
 action 1.0 syslog msg "IOSWD_Sample2 Policy Triggered"
```

Watchdog System Monitor サンプル 3 ポリシー

第 3 のポリシーは、IP RIB Update という名前のプロセスによる合計メモリ使用量が、60 秒のサンプリング時間全体で、50% を超えて増加したときにアプレットをトリガーします。

```
event manager applet IOSWD_Sample3
 event ioswdsysmon sub1 mem-proc taskname "IP RIB Update" op gt val 50 is-percent true
period 60
 action 1.0 syslog msg "IOSWD_Sample3 Policy Triggered"
```

3 個のポリシーが設定され、複数のワークステーションからネットワーキング デバイスに対して繰り返し大量の ping が実行されます。そのためネットワーキング デバイスは一定の利用量を記録します。これにより、ポリシー 1 およびポリシー 2 がトリガーされ、コンソールに次のメッセージが表示されます。

```
00:42:23: %HA_EM-6-LOG: IOSWD_Sample1: IOSWD_Sample1 Policy Triggered
00:42:47: %HA_EM-6-LOG: IOSWD_Sample2: IOSWD_Sample2 Policy Triggered
```

登録したポリシーを表示するには、**show event manager policy registered** コマンドを使用します。

```
Device# show event manager policy registered
No.  Class   Type    Event Type          Trap  Time Registered           Name
1    applet  system  ioswdsysmon         Off   Fri Jul 23 02:27:28 2004  IOSWD_Sample1
 sub1 cpu_util {taskname {IP Input} op ge val 1 period 10.000 }
 action 1.0 syslog msg "IOSWD_Sample1 Policy Triggered"
2    applet  system  ioswdsysmon         Off   Fri Jul 23 02:23:52 2004  IOSWD_Sample2
 sub1 mem_used {taskname {Net Input} op gt val 100 is_percent FALSE}
 action 1.0 syslog msg "IOSWD_Sample2 Policy Triggered"
3    applet  system  ioswdsysmon         Off   Fri Jul 23 03:07:38 2004  IOSWD_Sample3
 sub1 mem_used {taskname {IP RIB Update} op gt val 50 is_percent TRUE period 60.000 }
 action 1.0 syslog msg "IOSWD_Sample3 Policy Triggered"
```

SNMP ライブラリ拡張の設定例

SNMP get オペレーションの例

次に、get 要求をローカル ホストに送信する例を示します。

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact
community
 public
Device(config-applet)# action 1.3 info type snmp oid
 1.3.6.1.2.1.1.4.0 get-type next community
public
```

次のログ メッセージが SNMP イベント マネージャ ログに書き込まれます。

```
1d03h:%HA_EM-6-LOG: lg: 1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgn: 1.3.6.1.2.1.1.5.0
```

次に、get 要求をリモート ホストに送信する例を示します。

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
 90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 get-type next community
 public ipaddr
 172.17.16.69
Device(config-applet)# action 1.3 info type snmp getid
 1.3.6.1.2.1.1.1.0 community
 public ipaddr
 172.17.16.69
```

次のログ メッセージが SNMP イベント マネージャ ログに書き込まれます。

```
1d03h:%HA_EM-6-LOG: lg: 1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgn: 1.3.6.1.2.1.1.5.0
```

SNMP GetID オペレーションの例

次に、getid 要求をローカル ホストに送信する例を示します。

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val
5120000 poll-interval
 90
Device(config-applet)# action 1.3 info type snmp getid
 community
 public
```

次のログ メッセージが SNMP イベント マネージャ ログに書き込まれます。

```
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_oid=1.3.6.1.2.1.1.5.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_value=jubjub.cisco.com
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_oid=1.3.6.1.2.1.1.6.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_value=
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysdescr_oid=1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_oid=1.3.6.1.2.1.1.2.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_value=products.222
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=1.3.6.1.2.1.1.3.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=10131676
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_oid=1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_value=YYY
```

次に、getid 要求をリモート ホストに送信する例を示します。

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
 90
Device(config-applet)# action 1.3 info type snmp getid
 1.3.6.1.2.1.1.1.0 community
```

```
 public ipaddr
 172.17.16.69
```

次のログ メッセージが SNMP イベント マネージャ ログに書き込まれます。

```
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_oid=1.3.6.1.2.1.1.5.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_value=jubjub.cisco.com
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_oid=1.3.6.1.2.1.1.6.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_value=
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysdescr_oid=1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_oid=1.3.6.1.2.1.1.2.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_value=products.222
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=1.3.6.1.2.1.1.3.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=10131676
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_oid=1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_value=YYY
```

set オペレーションの例

次に、set オペレーションをローカル ホストで実行する例を示します。

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
 90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 set-type
 integer
 5 sysName.0 community
 public
```

次のログ メッセージが SNMP イベント マネージャ ログに書き込まれます。

```
1d04h:%HA_EM-6-LOG: lset: 1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lset: XXX
```

次に、set オペレーションをリモート ホストで実行する例を示します。

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 set-type integer
 5 sysName.0 community
 public ipaddr
172.17.16.69
```

次のログ メッセージが SNMP イベント マネージャ ログに書き込まれます。

```
1d04h:%HA_EM-6-LOG: lset: 1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lset: XXX
```

SNMP 通知の生成の例

次に、sysUpTime.0 変数の SNMP トラップを設定する例を示します。

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
lt entry-val
 5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp var
 sysUpTime.0 oid
1.3.6.1.4.1.9.9.43.1.1.6.1.3.41 integer
 2
Device(config-applet)# action 1.4 info type snmp trap
 enterprise-oid
 ciscoSyslogMIB.2 generic-trapnum
6 specific-trapnum
1 trap-oid
 1.3.6.1.4.1.9.9.41.2.0.1 trap-var
sysUpTime.0
```

debug snmp packets コマンドがイネーブルにされている場合、次の出力が生成されます。

```
Device# debug snmp packets
1d04h: SNMP: Queuing packet to 172.69.16.2
1d04h: SNMP: V1 Trap, ent ciscoSyslogMIB.2, addr 172.19.rap 1
clogHistoryEntry.3 = 4
clogHistoryEntry.6 = 9999
1d04h: SNMP: Queuing packet to 172.19.208.130
1d04h: SNMP: V1 Trap, ent ciscoSyslogMIB.2, addr 172.19.rap 1
clogHistoryEntry.3 = 4
clogHistoryEntry.6 = 9999
1d04h: SNMP: Packet sent via UDP to 172.69.16.2
1d04h: SNMP: Packet sent via UDP to 172.69.16.2
infra-view10:
Packet Dump:
30 53 02 01 00 04 04 63 6f 6d 6d a4 48 06 09 2b
06 01 04 01 09 09 29 02 40 04 ac 13 d1 17 02 01
06 02 01 01 43 04 00 9b 82 5d 30 29 30 12 06 0d
2b 06 01 04 01 09 09 29 01 02 03 01 03 02 01 04
30 13 06 0d 2b 06 01 04 01 09 09 29 01 02 03 01
06 02 02 27 0f
Received SNMPv1 Trap:
Community: comm
Enterprise: ciscoSyslogMIBNotificationPrefix
Agent-addr: 172.19.209.23
Enterprise Specific trap.
Enterprise Specific trap: 1
Time Ticks: 10191453
clogHistSeverity = error(4)
clogHistTimestamp = 9999
```

次に、sysUpTime.0 変数の SNMP インフォーム要求を設定する例を示します。

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
 90
```

```
Device(config-applet)# action 1.3 info type snmp var
 sysUpTime.0 oid
1.3.6.1.4.1.9.9.43.1.1.6.1.3.41 integer
 2
Device(config-applet)# action 1.4 info type snmp inform
trap-oid
 1.3.6.1.4.1.9.9.43.2.0.1 trap-var
 sysUpTime.0 community
 public ipaddr
 172.19.209.24
```

debug snmp packets コマンドがイネーブルにされている場合、次の出力が生成されます。

```
Device# debug snmp packets
1d04h: SNMP: Inform request, reqid 24, errstat 0, erridx 0
sysUpTime.0 = 10244391
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.40 = 1
1d04h: SNMP: Packet sent via UDP to 172.19.209.24.162
1d04h: SNMP: Packet received via UDP from 172.19.209.24 on FastEthernet0/0
1d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
1d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
1d04h: SNMP: Inform request, reqid 25, errstat 0, erridx 0
sysUpTime.0 = 10244396
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.41 = 2
1d04h: SNMP: Packet sent via UDP to 172.19.209.24.162
1d04h: SNMP: Packet received via UDP from 172.19.209.24 on FastEthernet0/0
1d04h: SNMP: Response, reqid 25, errstat 0, erridx 0
1d04h: SNMP: Response, reqid 25, errstat 0, erridx 0
Device# debug snmp packets
5d04h: SNMP: Packet received via UDP from 172.19.209.23 on FastEthernet0/0
5d04h: SNMP: Inform request, reqid 24, errstat 0, erridx 0
sysUpTime.0 = 10244391
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.40 = 1
5d04h: dest if_index = 1
5d04h: dest ip addr= 172.19.209.24
5d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
5d04h: SNMP: Packet sent via UDP to 172.19.209.23.57748
5d04h: SNMP: Packet received via UDP from 172.19.209.23 on FastEthernet0/0
5d04h: SNMP: Inform request, reqid 25, errstat 0, erridx 0
```

EEM アプレットの可変ロジックの設定例

このセクションでは、一部の選択された action コマンドの例を示します。アプレット内の可変ロジックをサポートするすべての action コマンドについては、次の表を参照してください。

この例では、条件付きのループである **while**、**if** および **foreach** を使用してデータを出力します。**action divide**、**action increment** および **action puts** のようなその他のアクションコマンドは、条件が満たされている場合に実行されるアクションを定義するために使用します。

```
event manager applet printdata
event none
action 100 set colors "red green blue"
action 101 set shapes "square triangle rectange"
action 102 set i "1"
action 103 while $i lt 6
action 104  divide $i 2
```

```
action 105  if $_remainder eq 1
action 106   foreach _iterator "$colors"
action 107    puts nonewline "$_iterator "
action 108   end
action 109   puts ""
action 110  else
action 111   foreach _iterator "$shapes"
action 112    puts nonewline "$_iterator "
action 113   end
action 114   puts ""
action 115  end
action 116  increment i
action 117 end
```

イベント マネージャ アプレット ex が実行されると、次の出力が得られます。

```
event manager run printdata
red green blue
square triangle rectange
red green blue
square triangle rectange
red green blue
```

次の例では、2 個の環境変数、poll_interface と max_rx_rate が、それぞれ、F0/0 と 3 に設定されます。30 秒ごとに、インターフェイスで rx 比率の調査が行われます。rx 比率がしきい値を上回った場合は、syslog メッセージが表示されます。

このアプレットは、インターフェイスの調査に foreach 条件付き文を使用します。また、RXPS に属する値を EEM 環境変数に設定された max_rx_rate と比較するために、if 条件付きブロックを使用します。

```
event manager environment poll_interfaces F0/0
event manager environment max_rx_rate 3
ev man app check_rx_rate
ev timer watchdog name rx_timer time 30
action 100 foreach int $poll_interfaces
action 101  cli command "en"
action 102  cli command "show int $int summ | beg ------"
action 103  foreach line $_cli_result "\n"
action 105  regexp ".*[0-9]+\s+[0-9]+\s+[0-9]+\s+[0-9]+\s+[0-9]+\s+([0-9]+)\s+.*" $line
 junk rxps
action 106   if $_regexp_result eq 1
action 107    if $rxps gt $max_rx_rate
action 108     syslog msg "Warning rx rate for $int is > than threshold. Current value
is $rxps
 (threshold is $max_rx_rate)"
action 109    end
action 110   end
action 111  end
action 112 end
```

syslog メッセージ例：

```
Oct 16 09:29:26.153: %HA_EM-6-LOG: c: Warning rx rate for F0/0 is > than threshold.
 Current value is 4 (threshold is 3)
The output of show int F0/0 summ is of the format:

#show int f0/0 summ

 *: interface is up
 IHQ: pkts in input hold queue     IQD: pkts dropped from input queue
```

```
OHQ: pkts in output hold queue    OQD: pkts dropped from output queue
RXBS: rx rate (bits/sec)          RXPS: rx rate (pkts/sec)
TXBS: tx rate (bits/sec)          TXPS: tx rate (pkts/sec)
TRTL: throttle count

  Interface               IHQ   IQD  OHQ   OQD  RXBS RXPS  TXBS TXPS TRTL
------------------------------------------------------------------------
* FastEthernet0/0          0 87283    0     0     0    0     0    0    0
```

（注） アプレット内の可変ロジックをサポートするその他の action コマンドを使用するには、次の表にあるコマンドを使用してください。

表 *184 :* 使用できる *action* コマンド

Action コマンド	**目的**
action add	EEM アプレットがトリガーされたときに 2 つの変数の値を足します。
action append	EEM アプレットがトリガーされたときに、与えられた値に変数の現在の値をアペンドします。
action break	EEM アプレットがトリガーされたときに、すぐにアクションのループを終了します。
action comment	EEM アプレットがトリガーされたときにアプレットにコメントを追加します。
action context retrieve	EEM アプレットがトリガーされたときに、与えられたコンテキスト名キーのセットで特定される変数を取得します。
action context save	EEM アプレットがトリガーされたときに、複数のポリシー トリガー全体の情報を保存します。
action continue	EEM アプレットがトリガーされたときに、アクションのループを継続します。
action decrement	EEM アプレットがトリガーされたときに、変数の値をデクリメントします。
action divide	EEM アプレットがトリガーされたときに、与えられた序数の値で非除数を割ります。
action else	EEM アプレットがトリガーされたときに、if / else 条件付きアクション ブロックの else 条件付きアクションブロックの開始を指定します。

Action コマンド	目的
action elseif	EEM アプレットがトリガーされたときに、else / if 条件付きアクション ブロックの else 条件付きアクション ブロックの開始を特定します。
action end	EEM アプレットがトリガーされたときに、if / else および while 条件付きアクション ブロックの条件付きアクション ブロック終了の ID を指定します。
action exit	EEM アプレットがトリガーされたときに、すぐに実行中のアプレット コンフィギュレーションを終了することを指定します。
action foreach	EEM アプレットがトリガーされたときに、デリミタをトークン化されたパターンとして使用した入力文字列の反復を指定します。
action gets	EEM アプレットがトリガーされたときに、同期アプレットのローカル TTY から入力を取得し、与えられた変数に値を格納します。
action if	EEM アプレットがトリガーされたときに、if 条件付きブロック開始の ID を指定します。
action if goto	EEM アプレットがトリガーされたときに、指定された条件が True であればアプレットが与えられたラベルにジャンプすることを指示します。
action increment	EEM アプレットがトリガーされたときに、変数の値を増加させます。
action info type interface-names	EEM アプレットがトリガーされたときに、インターフェイス名を取得するアクションを指定します。
action info type snmp getid	SNMP get オペレーション中に簡易ネットワーク管理プロトコル（SNMP）エンティティから各変数を取得します。
action info type snmp inform	EEM アプレットがトリガーされたときに、SNMP インフォーム要求を送信します。

Action コマンド	目的
action info type snmp oid	EEM アプレットがトリガーされたときに、SNMP get オペレーションのタイプ、および、SNMP set オペレーション中に取得するオブジェクトを指定します。
action info type snmp trap	EEM アプレットがトリガーされたときに、SNMP trap 要求を送信します。
action info type snmp var	SNMP オブジェクト ID（OID）の変数、およびその値を EEM アプレットから作成します。
action multiply	EEM アプレットがトリガーされたときに、変数値に指定された整数値を掛けるアクションを指定します。
action puts	EEM アプレットがトリガーされたときにデータを直接ローカル TTY に出力するアクションを有効にします。
action regexp	EEM アプレットがトリガーされたときに入力文字列の正規表現パターンと比較するアクションを指定します。
action set（EEM）	EEM アプレットがトリガーされたときに変数の値を設定するアクションを指定します。
action string compare	EEM アプレットがトリガーされたときに 2 個の等しくない文字列を比較するアクションを指定します。
action string equal	EEM アプレットがトリガーされたときに 2 個の文字列が等しいかどうかを検証するアクションを指定します。
action string first	EEM アプレットがトリガーされたときに string2 内に最初に string1 が見つかったインデックスを返すアクションを指定します。
action string index	EEM アプレットがトリガーされたときに与えられたインデックス値で指定される文字を返すアクションを指定します。
action string last	EEM アプレットがトリガーされたときに string 2 内に最後に string1 が見つかったインデックスを返すアクションを指定します。

Action コマンド	目的
action string length	EEM アプレットがトリガーされたときに文字列の文字数を返すアクションを指定します。
action string match	EEM アプレットがトリガーされたときに文字列がパターンに一致すれば、$_string_result に 1 を返すアクションを指定します。
action string range	EEM アプレットがトリガーされたときに文字列の文字の範囲を格納するアクションを指定します。
action string replace	EEM アプレットがトリガーされたときに指定された文字列の文字の範囲を置き換えることで新しい文字列を格納するアクションを指定します。
action string tolower	EEM アプレットがトリガーされたときに文字列の特定の範囲の文字を小文字で格納するアクションを指定します。
action string toupper	EEM アプレットがトリガーされたときに文字列の特定の範囲の文字を大文字で格納するアクションを指定します。
action string trim	EEM アプレットがトリガーされたときに文字列をトリムするアクションを指定します。
action string trimleft	EEM アプレットがトリガーされたときに、ある文字列の文字を別の文字列の左端からトリムするアクションを指定します。
action string trimright	EEM アプレットがトリガーされたときに、ある文字列の文字を別の文字列の右端からトリムするアクションを指定します。
action subtract	EEM アプレットがトリガーされたときに、別の値から変数の値を引きます。
action while	EEM アプレットがトリガーされたときに条件付きブロックのループの開始を特定するアクションを指定します。

イベント SNMP オブジェクトの設定例

次の例は、SET オペレーション、および、設定される値が $_snmp_value にありスクリプトで管理されることを示します。次の例は、oid とその値を、後で取得されるコンテキストとして格納します。

```
event manager applet snmp-object1
 description "APPLET SNMP-OBJ-1"
 event snmp-object oid 1.3.6.1.2.1.31.1.1.1.18 type string sync no skip no istable yes
default 0
 action 1 syslog msg "SNMP-OBJ1:TRIGGERED" facility "SNMP_OBJ"
 action 2 context save key myoid variable "_snmp_oid"
 action 3 context save key myvalue variable "_snmp_value"
```

EEM アプレットの説明の設定例

次に、簡易ネットワーク管理プロトコル（SNMP）のサンプリングによって実行される Embedded Event Manager（EEM）アプレットの説明を追加または変更する例を示します。

```
event manager applet test
 description "This applet looks for the word count in syslog messages"
 event syslog pattern "count"
 action 1 syslog msg hi
```

その他の参考資料

ここでは、Cisco IOS CLI を使用した EEM ポリシーの記述に関する関連資料について説明します。

関連資料

関連項目	マニュアル タイトル
EEM コマンド：コマンド構文の詳細、デフォルト、コマンドモード、コマンド履歴、使用上の注意事項、および例	Cisco IOS Embedded Event Manager のコマンド リファレンス
Embedded Event Manager 概要	「Embedded Event Manager の概要」の章
Tcl を使用して Embedded Event Manager ポリシーを記述する	「Tcl を使用した Embedded Event Manager ポリシーの記述」の章
拡張オブジェクト トラッキングの設定	「Configuring Enhanced Object Tracking」の章

標準

標準	タイトル
新しい規格または変更された規格はサポートされていません。また、既存の規格に対するサポートに変更はありません。	--

MIB

MIB	MIB のリンク
CISCO-EMBEDDED-EVENT-MGR-MIB	選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
新しい RFC または変更された RFC はサポートされていません。また、既存の RFC に対するサポートに変更はありません。	--

シスコのテクニカル サポート

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

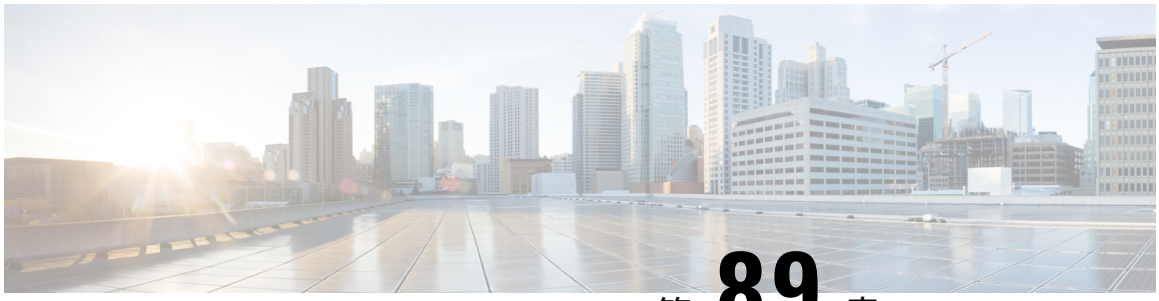
Cisco IOS CLI を使用した EEM 4.0 ポリシーの記述の機能情報

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

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

表 *185 : Cisco IOS CLI* を使用した *EEM 4.0* ポリシーの記述の機能情報

機能名	リリース	機能情報
Embedded Event Manager 4.0	15.2(5)E1	この機能は、c2960cx プラットフォームにのみ導入され、サポートされています。



第 89 章

Writing Embedded Event Manager Policies Using Tcl

この章では、ソフトウェア開発者が Tool command language（Tcl）スクリプトを使用して Embedded Event Manager（EEM）ポリシーを記述およびカスタマイズし、Cisco ソフトウェアの障害とイベントを処理できるようにする方法について説明します。EEM は、定義済みの Application Programming Interface（API）を介してレポートされる Cisco ソフトウェア システムの障害による、ポリシー方式のプロセスです。EEM ポリシー エンジンは、障害およびその他のイベントが発生したときに通知を受け取ります。EEM ポリシーは、システムの現在の状態に基づいて回復を実行し、該当するイベントのポリシーに指定されたアクションを実行します。回復アクションはポリシーが実行されたときにトリガーされます。

Tcl を使用した Embedded Event Manager ポリシーの記述に関する前提条件

- EEM ポリシーを記述するには、その前に「Embedded Event Manager Overview」の章を理解しておく必要があります。
- コマンドライン インターフェイス（CLI）コマンドを使用して EEM ポリシーを記述するときは、「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」の章をよく理解しておいてください。

Tcl を使用した Embedded Event Manager ポリシー記述について

EEM ポリシー

EEM では、イベントをモニターし、イベント発生が検出されたとき、およびしきい値を超えたときに、情報通知や是正などの任意のアクションを実施できます。EEM ポリシーは、イベントおよびイベントが発生した場合に行う処理を定義するエンティティです。EEM ポリシーにはアプレットとスクリプトの2つのタイプがあります。アプレットは、コマンドラインインターフェイス（CLI）設定に定義された、ポリシーの単純な形式です。スクリプトは、Tool Command Language（Tcl）で記述されたポリシーの形式です。

EEM アプレット

EEM アプレットは、イベント スクリーニング基準とイベント発生時に実行するアクションを定義する簡潔な方法です。EEM アプレット コンフィギュレーション モードでは、3 種類のコンフィギュレーション文がサポートされます。event コマンドを使用して実行するアプレットをトリガーするイベント基準を指定し、action コマンドを使用して、EEM アプレットがトリガーされるときに実行されるアクションを指定し、**set** コマンドを使用して EEM アプレット変数の値を設定します。現在、_exit_status 変数だけが、**set** コマンドでサポートされます。

アプレット コンフィギュレーションでは、event コンフィギュレーション コマンドを 1 つだけ使用できます。アプレット コンフィギュレーション サブモードが終了し、event コマンドが存在しない場合は、アプレットにイベントが割り当てられていないことを示す警告が表示されます。イベントが指定されない場合、アプレットは登録されたと見なされません。アプレットにアクションが割り当てられない場合、イベントはトリガーされますが、アクションは実行されません。1 つのアプレット コンフィギュレーション内で複数の action コンフィギュレーション コマンドが使用できます。登録済みのアプレットを表示するには、**show event manager policy registered** コマンドを使用します。

EEM アプレットを修正する前に、アプレット コンフィギュレーション モードを終了するまで既存のアプレットを置き換えられないことに注意してください。アプレット コンフィギュレーション モードでアプレットを修正中であっても、既存のアプレットを実行できます。変更は一時ファイルに書き込まれるため、登録を解除しないでアプレットを変更するのが安全です。アプレット コンフィギュレーション モードを終了すると、古いアプレットが登録解除され、新しいバージョンが登録されます。

アプレット内の action コンフィギュレーション コマンドは、*label* 引数を使用することで一意に識別できます。label 引数には任意の文字列値が使用できます。アクションは、*label* 引数をソート キーとして、アプレット内で英数字のキーの昇順に並べ替えられ、この順序で実行されます。同じ *label* 引数を異なるアプレットで使用できます。ラベルは 1 つのアプレット内でのみ一意にする必要があります。

Embedded Event Manager は、ポリシーそのものに含まれるイベント仕様に基づいてポリシーをスケジューリングし、実行します。アプレット コンフィギュレーション モードが終了するとき、EEM は、入力された event コマンドと action コマンドを検査し、指定されたイベントの発生時に実行されるようにアプレットを登録します。

Cisco IOS CLI を使用して EEM ポリシーを記述する方法については、「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」の章を参照してください。

EEM スクリプト

すべての Embedded Event Manager スクリプトは、Tcl で記述されます。Tcl は文字列ベースのコマンド言語で、実行時に解釈されます。Tclがサポートされるバージョンは、Tclバージョン 8.3.4 に、スクリプト サポートが追加されたものです。スクリプトは、ネットワーキング デバイスではなく、別のデバイスで、ASCIIエディタを使用して定義されます。続いてスクリプトはネットワーキング デバイスにコピーされ EEM に登録されます。Tcl スクリプトは EEM でサポートされます。強制適用される規則としての Embedded Event Manager ポリシーは、経過時間 20 秒未満で解釈および実行される必要がある、存続時間の短い実行時ルーチンです。20 秒よりも長い経過時間が必要な場合、event_register 文で maxrun パラメータを使用して、必要な値を指定する必要があります。

EEM ポリシーでは、すべての Tcl 言語機能が使用されます。ただし、シスコでは、EEM ポリシーの記述に活用できる Tcl コマンド拡張の形式で、Tcl 言語の機能を拡張しています。Tcl コマンド拡張のキーワードの主要なカテゴリでは、検出されたイベント、後続のアクション、ユーティリティ情報、カウンタの値、システム情報が特定されます。

EEM では、Tcl を使用して独自のポリシーを記述、実装できます。EEM スクリプトの記述には、次の作業が含まれます。

- ポリシーの実行時に決定に使用される基準を確立する、イベント Tcl コマンド拡張の選択。
- イベントの検出に関連付けられているイベント ディテクタ オプションの定義。
- 検出されたイベントのリカバリまたは検出されたイベントに対する応答を実装するアクションを選択すること。

EEM ポリシーの Tcl コマンド拡張のカテゴリ

EEM ポリシーの Tcl コマンド拡張には、さまざまなカテゴリがあります。

（注） すべての EEM ポリシーで使用するこれらの各カテゴリで使用可能な Tcl コマンドは、この資料の以降の項で説明します。

表 *186 : EEM* ポリシーの *Tcl* コマンド拡張のカテゴリ

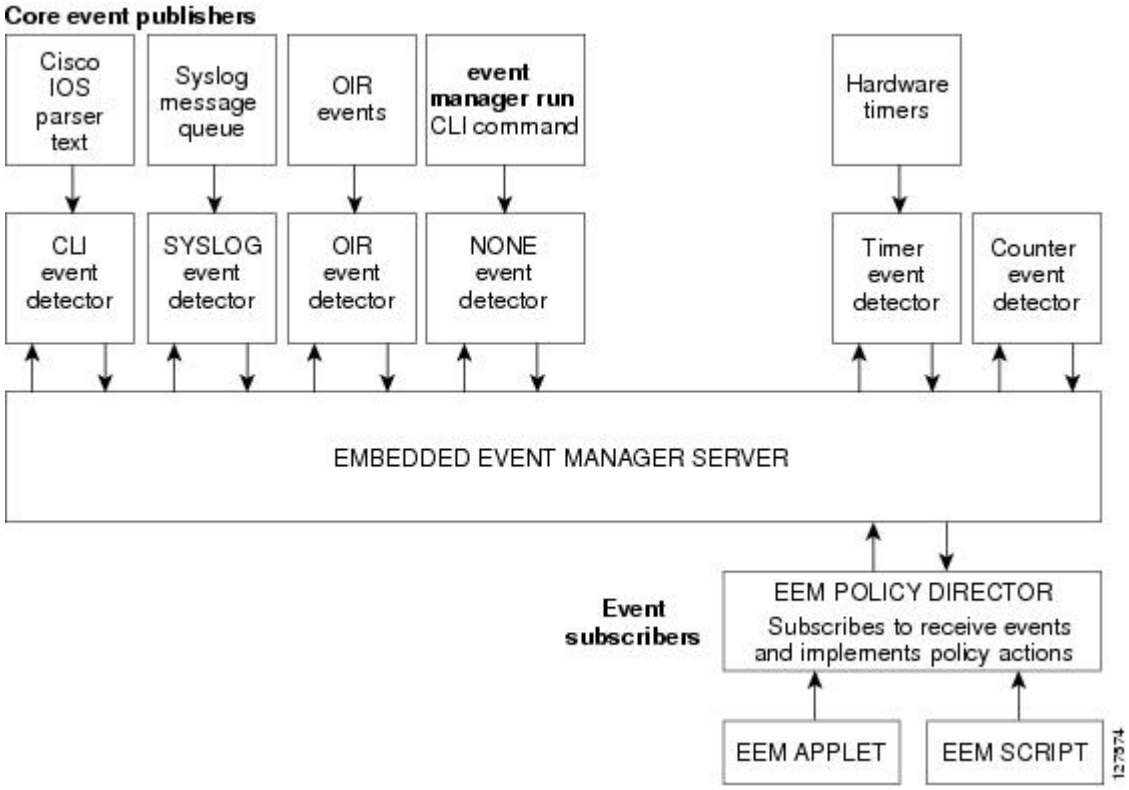
カテゴリ	定義
EEM イベントの Tcl コマンド拡張（イベント情報、イベント登録、イベント パブリッシュの 3 タイプ）	このカテゴリは、イベント固有のコマンドの **event_register_** *xxx* ファミリによって表されます。このカテゴリには、別のイベント情報 Tcl コマンド拡張の **event_reqinfo** もあります。これは、イベントについての情報を EEM に問い合わせるためにポリシーで使用されるコマンドです。アプリケーション固有のイベントをパブリッシュする、EEM イベントパブリッシュ Tcl コマンド拡張 **event_publish**> もあります。
EEM アクションの Tcl コマンド拡張	これらの Tcl コマンド拡張（たとえば、**action_syslog** など）は、イベントまたは障害への応答か、あるいは、イベントまたは障害からの回復のために、ポリシーによって使用されます。これらの拡張に加え、開発者は、Tcl 言語を使用して、必要なアクションを実装できます。
EEM ユーティリティの Tcl コマンド拡張	これらの Tcl コマンド拡張は、アプリケーション情報、カウンタ、またはタイマーの取得、保存、設定、または変更に使用されます。
EEM システム情報の Tcl コマンド拡張	このカテゴリは、システム固有の情報コマンドの **sys_reqinfo _** *xxx* ファミリによって表されます。これらのコマンドは、システム情報を収集する目的で、ポリシーによって使用されます。
EEM コンテキストの Tcl コマンド拡張	これらの Tcl コマンド拡張は、Tcl コンテキスト（可視変数およびその値）の保存および取得に使用されます。

EEM イベントの検出および回復の一般的なフロー

EEM は、イベント ディテクタと呼ばれるソフトウェア エージェントを使用してシステム内の異なるコンポーネントのモニターリングをサポートする、柔軟でポリシードリブンのフレームワークです。次の図に、EEM サーバー、コアイベントパブリッシャ（イベントディテクタ）、およびイベント サブスクライバ（ポリシー）の関係を示します。基本的に、イベントパブリッシャはイベントをスクリーニングして、イベント サブスクライバから提供されたイベント仕様に一致したときにイベントをパブリッシュします。イベントディテクタは、注目するイベントが発生したときに EEM サーバーに通知します。

イベントまたは障害が検出されると、Embedded Event Manager によって、たとえば次の図の OIR イベント パブリッシャなどのイベント パブリッシャから、検出された障害またはイベントの登録が発生しているかどうかが判断されます。EEM によって、イベント登録情報が、イベント データそのものと、照会されます。ポリシーによって、検出されたイベントが Tcl コマンド拡張 **event_register _** *xxx* で登録されます。イベント情報 Tcl コマンド拡張 **event_reqinfo** は、検出されたイベントに関する情報について Embedded Event Manager に問い合わせるために、ポリシーで使用されます。

図 *133 : Embedded Event Manager* コア イベント ディテクタ



Safe-Tcl

Safe-Tcl は、安全モードで作成されたインタープリタで、非信頼 Tcl スクリプトを実行できる、安全メカニズムです。安全インタープリタには、一部のシステム リソースへのアクセスや、ホストおよび他のアプリケーションに害が及ぼされることを防ぐ、制限されたコマンドのセットがあります。たとえば、コマンドは、重要な Cisco IOS ファイル システム ディレクトリにはアクセスできません。

シスコ定義のスクリプトはフル Tcl モードで実行されますが、ユーザー定義のスクリプトは Safe-Tcl モードで実行されます。Safe-Tcl を使用すると、シスコでは、個々の Tcl コマンドのディセーブルまたはカスタマイズを行えます。Tcl コマンドの詳細については、http://www.tcl.tk/man/ を参照してください。

次のリストにある Tcl コマンドは、一部の例外によって制約されます。各コマンドまたはコマンド キーワードに対する制約事項は、次のとおりです。

- **cd** ：制約付きの Cisco ディレクトリ名の 1 つへのディレクトリ移動はできません。
- **-- encoding** コマンド **names** 、 **encoding** 、 **convertfrom**および **encoding** が許可されます **convertto** 。 **encoding** 引数のない **encoding system** コマンドは許可されています が、**?encoding?** キーワードを使用した **encoding system** コマンドは使用できません。
- **exec** ：使用できません。

- **fconfigure** ：使用できます。
- **file** ：以下は使用できます。
  - **file dirname**
  - **file exists**
  - **file extension**
  - **file isdirectory**
  - **file join**
  - **file pathtype**
  - **file rootname**
  - **file split**
  - **file stat**
  - **file tail**
- **file** ：以下は使用できません。
  - **file atime**
  - **file attributes**
  - **file channels**
  - **file copy**
  - **file delete**
  - **file executable**
  - **file isfile**
  - **file link**
  - **file lstat**
  - **file mkdir**
  - **file mtime**
  - **file nativename**
  - **file normalize**
  - **file owned**
  - **file readable**
  - **file readlink**
  - **file rename**
  - **file rootname**
  - **file separator**
  - **file size**
  - **file system**
  - **file type**
  - **file volumes**
  - **file writable**
- **glob** ：制約付きの Cisco ディレクトリの 1 つで検索する場合、**glob** コマンドは使用できません。これ以外の場合は使用できます。
- **load** ：ユーザー ポリシー ディレクトリまたはユーザー ライブラリ ディレクトリにあるファイルのみがロードできます。静的パッケージ（たとえば、C コードで構成されるライブラリ）は、**load** コマンドではロードできません。

- **open** ：**open** コマンドは、制約付きの Cisco ディレクトリの 1 つにあるファイルでは使用できません。
- **pwd** ：**pwd** コマンドは使用できません。
- **socket** ：**socket** コマンドは使用できます。
- **source** ：**source** コマンドは、ユーザーポリシーディレクトリまたはユーザー ライブラリディレクトリにあるファイルで使用できます。

EEM 2.4 のバイトコード サポート

EEM 2.4 で、標準バイトコード スクリプト拡張子 .tbc のファイルを受け付けることによって、Bytecode Language（BCL）サポートが導入されています。Tcl バージョン 8.3.4 では、BCL が定義され、Tcl スクリプトが BCL に変換されるコンパイラが含まれています。EEM 2.4 のユーザー ポリシーおよびシステム ポリシーで有効な EEM ポリシーのファイル拡張子は、.tcl（Tcl テキスト ファイル）と .tbc（Tcl バイトコード ファイル）です。

バイトコードの Tcl スクリプトを格納すると、ポリシーの実行速度が向上します。これは、コードが事前にコンパイルされ、ポリシーサイズが小さくなり、コードを隠蔽するためです。難読化はスクリプトの変更を若干難しくし、論理を隠して知的財産権を保護します。

サポート コードおよび信頼済みコードのリリースのために別のオプションを提供するため、バイトコードのサポートが追加されています。十分に理解しているソフトウェア、信頼できるソフトウェア、またはサポートされているソフトウェアのみをネットワーク デバイスで実行することを推奨します。IOS EEM サポートの Tcl バイトコードを生成するには、TclPro バージョン 1.4 または 1.5 を使用します。

Tcl スクリプトをバイトコードに変換するには、procomp、Free TclPro Compiler の一部、または Active State Tcl Development Kit を使用できます。Tcl スクリプトを procomp を使用してコンパイルする場合、コードはスクランブルされ、.tbc ファイルが生成されます。バイトコード ファイルはプラットフォームに依存せず、Windows、Linux、および UNIX などの、TclPro を使用できるすべてのオペレーティング システムで生成できます。Procomp は TclPro の一部であり、http://www.tcl.tk/software/tclpro で入手できます。

登録の置き換え

通常の Tcl の置き換えの他に、EEM 2.3 では、EEM イベント登録ステートメントの行内の個別のパラメータを環境変数に置き換えることができます。

EEM 2.4 では、イベント登録ステートメントの行にある複数パラメータを 1 つの環境変数で置き換える機能が導入されています。

（注） 1 つめの環境変数のみで、複数パラメータの置き換えがサポートされます。個別のパラメータを指定することも引き続き可能です。それを行うには最初の変数の後に追加の環境変数を追加します。

置き換えを示すために、1 つの環境変数 $_eem_syslog_statement が次のとおりに設定されています。

```
::cisco::eem::event_register_syslog pattern COUNT
```

登録の置き換えを使用すると、$_eem_syslog_statement 環境変数が、次の EEM ユーザー ポリシーで使用されます。

```
$_eem_syslog_statement occurs $_eem_occurs_val
action_syslog "this is test 3"
```

環境変数は、それらを使用するポリシーを登録する前に定義しておく必要があります。$_eem_syslog_statement 環境変数を定義するには、次を実行します。

```
Device(config)# event manager environment eem_syslog_statement
::cisco::eem::event_register_syslog pattern COUNT
Device(config)# event manager environment eem_occurs_val 2
```

EEM 用のシスコ ファイル命名規則

すべての Embedded Event Manager ポリシー名、ポリシー サポート ファイル（たとえば、E メール テンプレート ファイル）、およびライブラリ ファイル名は、シスコのファイル命名規則に従う必要があります。このため、Embedded Event Manager ポリシー ファイル名は、次の仕様に従っています。

- オプションのプレフィックス Mandatory. がある場合、これは、システム ポリシーがまだ登録されていない場合に、自動的に登録される必要があるシステム ポリシーであることを示します。たとえば、Mandatory.sl_text.tcl などです。
- 指定された 1 つめのイベントの 2 文字の省略形が含まれるファイル名の本体部（下の表を参照）、下線部、および、ポリシーをさらに示す説明フィールド部。
- ファイル名拡張子部は .tcl と定義されます。

Embedded Event Manager の E メール テンプレート ファイルは、email_template のファイル名のプレフィックスと、後続の E メール テンプレートの使用状況を示す省略形で構成されます。

Embedded Event Manager ライブラリ ファイル名は、ライブラリの使用状況を示す説明フィールドを含むファイル名の本体部と、後続の _lib、および .tcl というファイル名拡張子で構成されます。

表 *187 : 2* 文字の省略形の指定

ap	event_register_appl
cl	event_register_cli
ct	event_register_counter
go	event_register_gold
if	event_register_interface

io	event_register_ioswdsysmon
la	event_register_ipsla
nf	event_register_nf
no	event_register_none
oi	event_register_oir
pr	event_register_process
rf	event_register_rf
rs	event_register_resource
rt	event_register_routing
rp	event_register_rpc
sl	event_register_syslog
sn	event_register_snmp
st	event_register_snmp_notification
so	event_register_snmp_object
tm	event_register_timer
tr	event_register_track
ts	event_register_timer_subscriber
wd	event_register_wdsysmon

Tcl を使用した Embedded Event Manager ポリシーの記述方法

EEM Tcl スクリプトの登録と定義

環境変数を設定し、EEMポリシーを登録するには、この作業を実行します。EEMは、ポリシーそのものに含まれるイベント仕様に基づいてポリシーをスケジューリングし、実行します。EEM ポリシーが登録されると、ソフトウェアによって、ポリシーが調べられ、指定されたイベントの発生時に実行されるよう、登録されます。

始める前に

Tcl スクリプト言語で記述されたポリシーが使用できる状態である必要があります。サンプルポリシーを示します。使用している Cisco IOS リリースのイメージで使用可能なポリシーについては、EEM サンプル ポリシー（2139 ページ）を参照してください。これらのサンプル ポリシーは、システム ポリシー ディレクトリに保存されています。

手順の概要

1. **enable**
2. **show event manager environment** [**all**| *variable-name*]
3. **configure terminal**
4. **event manager environment** *variable-name string*
5. EEM Tcl スクリプトの登録と定義 を繰り返して、EEM Tcl スクリプトの登録と定義 で登録されるポリシーに必要なすべての環境変数を設定します。
6. **event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]
7. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager environment** [**all**| *variable-name*] 例： `Device# show event manager environment all`	（任意）EEM 環境変数の名前と値を表示します。 • オプションの **all** キーワードは、すべての EEM 環境変数を表示します。 • オプションの *variable-name* 引数は、指定された環境変数に関する情報を表示します。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**event manager environment** *variable-name string* 例： `Device(config)# event manager environment _cron_entry 0-59/2 0-23/1 * * 0-6`	指定された EEM 環境変数の値を設定します。 • この例では、ソフトウェアによって、CRON タイマー環境変数が、毎日、毎時の 2 分目に設定されます。
ステップ 5	EEM Tcl スクリプトの登録と定義 を繰り返して、EEM Tcl スクリプトの登録と定義 で登録されるポリシーに必要なすべての環境変数を設定します。	--

	コマンドまたはアクション	目的
ステップ 6	**event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**] 例： `Device(config)# event manager policy tm_cli_cmd.tcl type system`	ポリシー内で定義された指定イベントが発生した場合に、EEM ポリシーを実行するよう、定義します。 • **system** キーワードを使用して、シスコ定義のシステムポリシーを登録します。 • **user** キーワードを使用して、ユーザー定義のシステムポリシーを登録します。 • **trap** キーワードを使用して、ポリシーがトリガーされた場合の SNMP トラップを生成します。 • この例では、tm_cli_cmd.tcl という名前の EEM サンプル ポリシーが、システム ポリシーとして定義されます。
ステップ 7	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、**show event manager environment** 特権 EXEC コマンドを使用して、すべての EEM 環境変数の名前と値を表示する例を示します。

```
Device# show event manager environment all
No.  Name                           Value
1    _cron_entry                    0-59/2 0-23/1 * * 0-6
2    _show_cmd                      show ver
3    _syslog_pattern                .*UPDOWN.*Ethernet1/0.*
4    _config_cmd1                   interface Ethernet1/0
5    _config_cmd2                   no shut
```

登録済みの EEM ポリシーの表示

登録済みの EEM ポリシーを表示するには、次の任意の作業を実行します。

手順の概要

1. **enable**
2. **show event manager policy registered** [**event-type** *event-name*] [**time-ordered**| **name-ordered**] [**detailed** *policy-filename*]

手順の詳細

ステップ 1 **enable**

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 **show event manager policy registered** [**event-type** *event-name*] [**time-ordered**| **name-ordered**] [**detailed** *policy-filename*]

このコマンドを **time-ordered** キーワードとともに使用して、現在登録されているポリシーの情報を時間でソートして表示します。次に例を示します。

例：

```
Device# show event manager policy registered time-ordered
No.  Type    Event Type          Trap  Time Registered           Name
1    system  timer cron          Off   Wed May11  01:43:18 2005  tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240
2    system  syslog              Off   Wed May11  01:43:28 2005  sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90
3    system  proc abort          Off   Wed May11  01:43:38 2005  pr_cdp_abort.tcl
 instance 1 path {cdp2.iosproc}
 nice 0 priority normal maxrun 20
```

このコマンドを **name-ordered** キーワードとともに使用して、現在登録されているポリシーの情報を名前でソートして表示します。次に例を示します。

例：

```
Device# show event manager policy registered name-ordered
No.  Type    Event Type          Trap  Time Registered           Name
1    system  proc abort          Off   Wed May11  01:43:38 2005  pr_cdp_abort.tcl
 instance 1 path {cdp2.iosproc}
 nice 0 priority normal maxrun 20
2    system  syslog              Off   Wed May11  01:43:28 2005  sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90
3    system  timer cron          Off   Wed May11  01:43:18 2005  tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240
```

このコマンドを **event-type** キーワードとともに使用して、*event-name* 引数で指定されたイベントタイプの現在登録されているポリシーに関する情報を表示します。次に例を示します。

例：

```
Device# show event manager policy registered event-type syslog
No.  Type    Event Type          Time Registered           Name
1    system  syslog              Wed May11  01:43:28 2005  sl_intf_down.tcl
```

```
occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
nice 0 priority normal maxrun 90
```

EEM ポリシーの登録解除

EEM ポリシーを実行コンフィギュレーション ファイルから削除するには、次の作業を実行します。ポリシーの実行はキャンセルされます。

手順の概要

1. **enable**
2. **show event manager policy registered** [**event-type** *event-name*][**system**| **user**] [**time-ordered**| **name-ordered**] [**detailed** *policy-filename*]
3. **configure terminal**
4. **no event manager policy** *policy-filename*
5. **exit**
6. EEM ポリシーの登録解除 を繰り返して、ポリシーが削除されたことを確認します。

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager policy registered** [**event-type** *event-name*][**system**| **user**] [**time-ordered**| **name-ordered**] [**detailed** *policy-filename*] 例： `Device# show event manager policy registered`	（任意）現在登録されている EEM ポリシーを表示します。 • オプションの **system** キーワードまたは **user** キーワードによって、登録済みのシステム ポリシーまたはユーザー ポリシーが表示されます。 • キーワードが指定されない場合は、すべてのイベント タイプに対する登録された EEM ポリシーが時間順に表示されます。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**no event manager policy** *policy-filename* 例：	ポリシーを登録解除するために EEM ポリシーを設定から削除します。

	コマンドまたはアクション	目的
	`Device(config)# no event manager policy pr_cdp_terminate.tcl`	• この例では、コマンドの **no** 形式を使用して、指定されたポリシーの登録が解除します。
ステップ 5	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 6	EEM ポリシーの登録解除 を繰り返して、ポリシーが削除されたことを確認します。 例： `Device# show event manager policy registered`	--

例

次に、**show event manager policy registered** 特権 EXEC コマンドを使用して、現在登録されている 3 個の EEM ポリシーを表示する例を示します。

```
Device# show event manager policy registered
No.  Type    Event Type          Trap  Time Registered           Name
1    system  timer cron          Off   Tue Oct11  01:43:18 2005 tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240.000
2    system  syslog              Off   Tue Oct11  01:43:28 2005 sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90.000
3    system  proc abort          Off   Tue Oct11  01:43:38 2005 pr_cdp_terminate.tcl
 instance 1 path {cdp2.iosproc}
 nice 0 priority normal maxrun 20.000
```

現在のポリシーが表示されたら、**no** 形式の **event manager policy** コマンドを使用して pr_cdp_terminate.tcl ポリシーの削除が決定されます。

```
Device# configure terminal
Device(config)# no event manager policy pr_cdp_terminate.tcl
Device(config)# exit
```

show event manager policy registered 特権 EXEC コマンドを再度入力すると、現在登録されている EEM ポリシーが表示されます。ポリシー pr_cdp_terminate.tcl は、登録されていません。

```
Device# show event manager policy registered
No.  Type    Event Type          Trap  Time Registered           Name
1    system  timer cron          Off   Tue Oct11  01:45:17 2005 tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240.000
2    system  syslog              Off   Tue Oct11  01:45:27 2005 sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90.000
```

EEM ポリシー実行の一時停止

すべての EEM ポリシーの実行をただちに一時停止するには、次の作業を実行します。一時的なパフォーマンスまたはセキュリティ面での理由から、ポリシーの登録解除ではなく一時停止が必要なことがあります。

手順の概要

1. **enable**
2. **show event manager policy registered** [**event-type** *event-name*][**system**| **user**] [**time-ordered**| **name-ordered**] [**detailed** *policy-filename*]
3. **configure terminal**
4. **event manager scheduler suspend**
5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager policy registered** [**event-type** *event-name*][**system**| **user**] [**time-ordered**| **name-ordered**] [**detailed** *policy-filename*] 例： `Device# show event manager policy registered`	（任意）現在登録されている EEM ポリシーを表示します。 • オプションの **system** キーワードまたは **user** キーワードによって、登録済みのシステム ポリシーまたはユーザー ポリシーが表示されます。 • キーワードが指定されない場合は、すべてのイベント タイプに対する登録された EEM ポリシーが時間順に表示されます。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**event manager scheduler suspend** 例： `Device(config)# event manager scheduler suspend`	すべての EEM ポリシーの実行がすぐに一時停止されます。
ステップ 5	**exit** 例：	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

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

例

次に、**show event manager policy registered** 特権 EXEC コマンドを使用して、EEM のすべての登録済みポリシーを表示する例を示します。

```
Device# show event manager policy registered
No.  Type    Event Type          Trap  Time Registered           Name
1    system  timer cron          Off   Sat Oct11  01:43:18 2003  tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240.000
2    system  syslog              Off   Sat Oct11  01:43:28 2003  sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90.000
3    system  proc abort          Off   Sat Oct11  01:43:38 2003  pr_cdp_abort.tcl
 instance 1 path {cdp2.iosproc}
 nice 0 priority normal maxrun 20.000
```

すべての EEM ポリシーの実行をすぐに一時停止するには、**event manager scheduler suspend** コマンドを入力します。

```
Device# configure terminal
Device(config)# event manager scheduler suspend
*Nov  2 15:34:39.000: %HA_EM-6-FMS_POLICY_EXEC: fh_io_msg: Policy execution has been
suspended
```

EEM ポリシーの管理

ユーザー ライブラリ ファイルまたはユーザー定義 EEM ポリシーの保存に使用するディレクトリを指定するには、この作業を実行します。

（注） この作業は、Tcl スクリプトを使用して記述される EEM ポリシーのみに適用されます。

手順の概要

1. **enable**
2. **show event manager directory user** [**library**| **policy**]
3. **configure terminal**
4. **event manager directory user** {**library** *path*| **policy** *path*}
5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show event manager directory user** [**library**| **policy**] 例： `Device# show event manager directory user library`	（任意）EEM ユーザー ライブラリまたはポリシー ファイルの保存に使用するディレクトリを表示します。 • オプションの **library** キーワードによって、ユーザーライブラリファイルに使用されるディレクトリが表示されます。 • オプションの **policy** キーワードによって、ユーザー定義 EEM ポリシーに使用されるディレクトリが表示されます。
ステップ 3	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**event manager directory user** {**library** *path*| **policy** *path*} 例： `Device(config)# event manager directory user library disk0:/user_library` `Device(config)# event manager directory user library bootflash:/user_library`	ユーザー ライブラリ ファイルまたはユーザー定義 EEMポリシーの保存に使用するディレクトリを指定します。 • ユーザーディレクトリへの絶対パス名を指定するには、*path* 引数を指定します。
ステップ 5	**exit** 例： `Device(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、**show event manager directory user** 特権 EXEC コマンドを使用して、EEM ユーザーライブラリファイルの保存に使用されるディレクトリがある場合に、そのディレクトリを表示する例を示します。

```
Device# show event manager directory user library
disk0:/user_library
```

```
Device# show event manager directory user library
bootflash:/user_library
```

履歴テーブル サイズの変更と EEM 履歴データの表示

履歴テーブルのサイズを変更し、EEM 履歴データを表示するには、次の任意の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **event manager history size** {**events** | **traps**} [*size*]
4. **exit**
5. **show event manager history events** [**detailed**] [**maximum** *number*]
6. **show event manager history traps** [**server** | **policy]**

手順の詳細

ステップ 1 enable

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 configure terminal

グローバル コンフィギュレーション モードを開始します。

例：

```
Device# configure terminal
```

ステップ 3 event manager history size {events | traps} [size]

このコマンドを使用して、EEM イベント履歴テーブルのサイズ、または、EEM SNMP トラップ履歴テーブルのサイズを変更します。次に、EEM イベント履歴テーブルのサイズを 30 エントリに変更する例を示します。

例：

```
Device(config)# event manager history size events 30
```

ステップ 4 exit

グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例：

```
Device(config)# exit
```

ステップ 5 **show event manager history events** [**detailed**] [**maximum** *number*]

このコマンドを使用して、トリガーされた各 EEM イベントについての情報を表示します。

例：

```
Device# show event manager history events
No.  Time of Event             Event Type          Name
1    Fri Sep  9 13:48:40 2005  syslog              applet: one
2    Fri Sep  9 13:48:40 2005  syslog              applet: two
3    Fri Sep  9 13:48:40 2005  syslog              applet: three
4    Fri Sep  9 13:50:00 2005  timer cron          script: tm_cli_cmd.tcl
5    Fri Sep  9 13:51:00 2005  timer cron          script: tm_cli_cmd.tcl
```

ステップ 6 **show event manager history traps** [**server** | **policy]**

このコマンドを使用して、EEM サーバーまたは EEM ポリシーのいずれかから送信された EEM SNMP トラップを表示します。

例：

```
Device# show event manager history traps
No.  Time                      Trap Type           Name
1    Fri Sep  9 13:48:40 2005  server              applet: four
2    Fri Sep  9 13:57:03 2005  policy              script: no_snmp_test.tcl
```

EEM を使用したソフトウェア モジュール方式プロセスの信頼性メトリック

Cisco IOS ソフトウェアモジュール方式プロセスの信頼性メトリックを表示するには、この任意の作業を実行します。この **show event manager metric processes** コマンド拡張は、ソフトウェアモジュール方式イメージでのみサポートされます。

手順の概要

1. **enable**
2. **show event manager metric process** {**all**| *process-name*}

手順の詳細

ステップ 1 **enable**

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 **show event manager metric process** {**all**| *process-name*}

このコマンドを使用して、プロセスの信頼性メトリック データを表示します。システムでは、プロセスの開始時と終了時にレコードが保存され、このデータが、信頼性分析の基本データとして使用されます。この部分の例では、システムに挿入されているすべてのカード上でのプロセスのメトリック データを示す、最初と最後のエントリが表示されます。

例：

```
Device# show event manager metric process all
====================================
process name: devc-pty, instance: 1
sub_system id: 0, version: 00.00.0000
--------------------------------
last event type: process start
recent start time: Fri Oct10  20:34:40 2005
recent normal end time: n/a
recent abnormal end time: n/a
number of times started: 1
number of times ended normally: 0
number of times ended abnormally: 0
most recent 10 process start times:
-------------------------
Fri Oct10  20:34:40 2005
-------------------------
most recent 10 process end times and types:
cumulative process available time: 6 hours 30 minutes 7 seconds 378 milliseconds
cumulative process unavailable time: 0 hours 0 minutes 0 seconds 0 milliseconds
process availability:  0.100000000
number of abnormal ends within the past 60 minutes (since reload): 0
number of abnormal ends within the past 24 hours (since reload): 0
number of abnormal ends within the past 30 days (since reload): 0
.
.
.
====================================
process name: cdp2.iosproc, instance: 1
sub_system id: 0, version: 00.00.0000
--------------------------------
last event type: process start
recent start time: Fri Oct10  20:35:02 2005
recent normal end time: n/a
recent abnormal end time: n/a
number of times started: 1
number of times ended normally: 0
number of times ended abnormally: 0
most recent 10 process start times:
-------------------------
Fri Oct10  20:35:02 2005
-------------------------
most recent 10 process end times and types:

cumulative process available time: 6 hours 29 minutes 45 seconds 506 milliseconds
cumulative process unavailable time: 0 hours 0 minutes 0 seconds 0 milliseconds
process availability:  0.100000000
number of abnormal ends within the past 60 minutes (since reload): 0
number of abnormal ends within the past 24 hours (since reload): 0
number of abnormal ends within the past 30 days (since reload): 0
```

トラブルシューティングのヒント

特権 EXEC モードで **debug event manager** コマンドを使用して、EEM コマンド操作のトラブルシューティングを行います。デバッグ コマンドは注意して使用してください。生成される出力量によってデバイスの動作が遅くなったり、停止したりすることがあります。シスコ エンジニアの管理下に限ってこのコマンドを使用することを推奨します。

EEM サンプル ポリシーの変更

サンプル ポリシーの 1 つを変更するには、この作業を実行します。Cisco ソフトウェアには、Embedded Event Manager が含まれるイメージに、いくつかのサンプル ポリシーが含まれています。EEM ポリシーの開発者は、ポリシーが実行されるイベントと、イベントの記録および応答に関連付けられているオプションを、カスタマイズすることによって、これらのポリシーを変更できます。さらに、開発者は、ポリシーの実行時に実装されるアクションを選択できます。

EEM サンプル ポリシー

シスコには、次の表に示されているように、サンプル ポリシーのセットが含まれています。ユーザーは、サンプル ポリシーをユーザー ディレクトリにコピーし、ポリシーを変更するか、または、独自にポリシーを記述することができます。現時点でポリシー作成のためにシスコでサポートされているスクリプト言語は、Tcl だけです。Tcl ポリシーは、Emacs などのテキスト エディタを使用して変更できます。ポリシーは、定義されている経過時間の秒数以内で実行する必要があり、時間変数はポリシー内で設定できます。現在のデフォルト値は 20 秒です。

次の表で、サンプル EEM ポリシーについて説明します。

表 *188 : EEM* サンプル ポリシーの説明

ポリシーの名前	説明
pr_cdp_abort.tcl	Cisco ソフトウェア モジュラリティ イメージを使用して導入されました。このポリシーでは、cdp2.iosproc プロセスの終了イベントがモニターされます。SYSLOG にメッセージが記録され、終了の詳細が E メールで送信されます。
pr_crash_reporter.tcl	Cisco ソフトウェア モジュラリティ イメージを使用して導入されました。このポリシーでは、すべてのプロセスの終了イベントがモニターされます。イベントが発生すると、ポリシーによって、クラッシュ ダンプ ファイルを含むクラッシュ情報が、CGI スクリプトによってデータが処理される指定された URL に、送信されます。
pr_iprouting_abort.tcl	Cisco ソフトウェア モジュラリティ イメージを使用して導入されました。このポリシーでは、iprouting.iosproc プロセスの終了イベントがモニターされます。SYSLOG にメッセージが記録され、終了の詳細が E メールで送信されます。

ポリシーの名前	説明
sl_intf_down.tcl	このポリシーは、設定可能な Syslog メッセージが記録されるときに実行されます。設定可能な CLI コマンドが実行され、結果が E メールで送信されます。
tm_cli_cmd.tcl	このポリシーは、設定可能な CRON エントリを使用して実行されます。設定可能な CLI コマンドが実行され、結果が E メールで送信されます。
tm_crash_history.tcl	Cisco ソフトウェア モジュラリティ イメージを使用して導入されました。このポリシーは、毎日夜中に実行され、指定された E メール アドレスにプロセス クラッシュ履歴レポートが E メールで送信されます。
tm_crash_reporter.tcl	このポリシーは、登録後 5 秒間実行されます。ポリシーが設定に保存される場合、デバイスがリロードされるたびにも実行されます。ポリシーによって、リロード理由を示すプロンプトが表示されます。クラッシュが原因でリロードされる場合、ポリシーによって最新の crashinfo ファイルが検索され、この情報が指定された URL に送信されます。
tm_fsys_usage.tcl	Cisco ソフトウェア モジュラリティ イメージを使用して導入されました。このポリシーは、設定可能な CRON エントリを使用して実行され、ディスク領域の使用状況がモニターされます。ディスク領域の使用状況が、設定可能なしきい値を超えると、Syslog メッセージが表示されます。
wd_mem_reporter.tcl	Cisco ソフトウェア モジュラリティ イメージを使用して導入されました。使用可能なメモリ容量が、使用可能な初期システム メモリの 20% を下回った場合、このポリシーによって、システム メモリ低下の状態がレポートされます。Syslog メッセージが表示され、オプションで、E メールが送信されます。

使用可能なサンプル ポリシーおよびその実行方法についての詳細は、EEM イベント ディテクタのデモの例 （2160 ページ）を参照してください。

手順の概要

1. **enable**
2. **show event manager policy available detailed** *policy-filename*
3. 画面に表示されたサンプル ポリシーの内容を、テキスト エディタにカット アンド ペーストします。
4. ポリシーを編集し、新しいファイル名で保存します。
5. 新しいファイルを、デバイスのフラッシュ メモリにコピーして戻します。
6. **configure terminal**
7. **event manager directory user** {**library** *path*| **policy** *path*}
8. **event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]

手順の詳細

ステップ 1 **enable**

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 **show event manager policy available detailed** *policy-filename*

ポリシーによって使用される環境変数と、ポリシーの実行方法の説明の詳細を含む、指定された実際のサンプル ポリシーを表示します。**detailed** キーワードが **show event manager policy available** コマンドと **show event manager policy registered** コマンドに導入されました。お使いのリリースによっては、2 つの Tcl スクリプトのいずれかをこのドキュメントの設定例セクションからコピーしなければならない場合があります（Tcl のサンプル スクリプトを使用したポリシーのプログラミングの例 （2169 ページ）を参照）。次に、サンプル ポリシー tm_cli_cmd.tcl についての詳細が画面上に表示される例を示します。

例：

```
Device# show event manager policy available detailed tm_cli_cmd.tcl
```

ステップ 3 画面に表示されたサンプル ポリシーの内容を、テキスト エディタにカット アンド ペーストします。

編集機能とコピー機能を使用して、デバイスから別のデバイス上のテキスト エディタに、内容を移動します。

ステップ 4 ポリシーを編集し、新しいファイル名で保存します。

テキスト エディタを使用して、ポリシーを Tcl スクリプトとして変更します。ファイルの命名規則については、EEM 用のシスコ ファイル命名規則 （2126 ページ）を参照してください。

ステップ 5 新しいファイルを、デバイスのフラッシュ メモリにコピーして戻します。

デバイスのフラッシュ ファイル システム（通常は disk0:）にファイルをコピーします。ファイルのコピーの詳細については、『*Configuration Fundamentals Configuration Guide*』の「Using the Cisco IOS File System」の章を参照してください。

デバイスのフラッシュ ファイル システム（通常は bootflash:）にファイルをコピーします。ファイルのコピーの詳細については、『*Configuration Fundamentals Configuration Guide*』の「Using the Cisco IOS File System」の章を参照してください。

ステップ 6 **configure terminal**

グローバル コンフィギュレーション モードを開始します。

例：

```
Device# configure terminal
```

ステップ 7 **event manager directory user** {**library** *path*| **policy** *path*}

ユーザー ライブラリ ファイルまたはユーザー定義 EEM ポリシーの保存に使用するディレクトリを指定します。次に、disk0 の user_library ディレクトリが、ユーザー ライブラリ ファイルを保存するディレクトリとして指定されます。

ユーザー ライブラリ ファイルまたはユーザー定義 EEM ポリシーの保存に使用するディレクトリを指定します。次に、bootflash の user_library ディレクトリが、ユーザー ライブラリ ファイルを保存するディレクトリとして指定されます。

例：

```
Device(config)# event manager directory user library disk0:/user_library

Device(config)# event manager directory user library bootflash:/user_library
```

ステップ 8 **event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]

ポリシー内で定義された指定イベントが発生した場合に、EEM ポリシーを実行するよう、定義します。次に、test.tcl という名前の EEM ポリシーが、ユーザー定義ポリシーとして登録される例を示します。

例：

```
Device(config)# event manager policy test.tcl type user
```

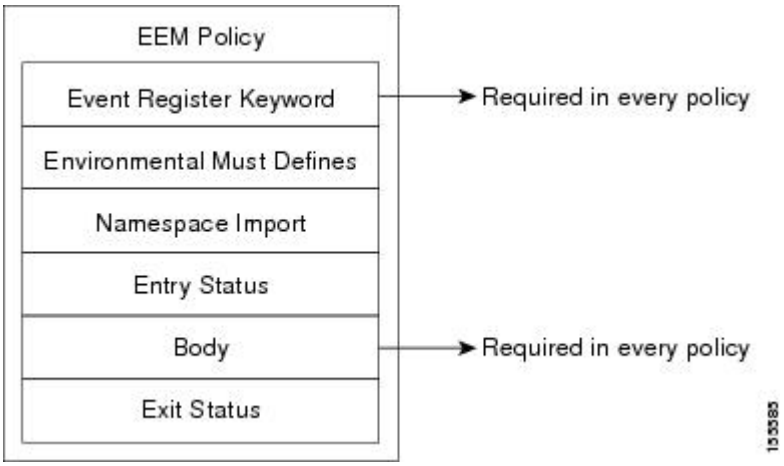
Tcl を使用した EEM ポリシーのプログラミング

Tcl コマンド拡張を使用してポリシーをプログラムするには、この作業を実行します。既存のポリシーをコピーし、変更することを推奨します。EEM Tcl ポリシーには、**event_register** Tcl コマンド拡張と本体の 2 つの必須部分が存在する必要があります。Tcl ポリシーの構造と要件（2142 ページ） の概念にある他のすべてのセクションは、オプションです。

Tcl ポリシーの構造と要件

すべての EEM ポリシーでは、次の図に示されているように、同じ構造が共有されます。EEM ポリシーには、**event_register** Tcl コマンド拡張と本体という 2 つの必須部分が存在します。ポリシーの残りの部分の、環境定義必須、名前空間のインポート、開始ステータス、および終了ステータスは、オプションです。

図 *134 : Tcl* ポリシーの構造と要件



各ポリシーの開始部分では、**event_register** Tcl コマンド拡張を使用して検出するイベントを記述し登録する必要があります。ポリシーのこの部分によって、ポリシーの実行がスケジュールされます。次に、**event_register_timer** Tcl コマンド拡張を登録する Tcl コードの例を示します。

```
::cisco::eem::event_register_timer cron name crontimer2 cron_entry $_cron_entry maxrun
240
```

環境定義必須セクションはオプションで、環境変数の定義が含まれます。次に、一部の環境変数をチェックし、定義する Tcl コードの例を示します。

```
# Check if all the env variables that we need exist.
# If any of them does not exist, print out an error msg and quit.
if {![info exists _email_server]} {
    set result \
      "Policy cannot be run: variable _email_server has not been set"
    error $result $errorInfo
}
if {![info exists _email_from]} {
    set result \
      "Policy cannot be run: variable _email_from has not been set"
    error $result $errorInfo
}
if {![info exists _email_to]} {
    set result \
      "Policy cannot be run: variable _email_to has not been set"
    error $result $errorInfo
```

名前空間のインポートセクションはオプションで、コードライブラリが定義されます。次に、名前空間インポート セクションを設定する Tcl コードの例を示します。

```
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
```

ポリシーの本体は必須の構造で、次のものを含める必要があります。

- 検出されたイベントに関する情報の EEM への問い合わせに使用される **event_reqinfo** イベント情報の Tcl コマンド拡張。

- EEM 特有のアクションの指定に使用される、**action_syslog** などのアクション Tcl コマンド拡張。
- 一般的なシステム情報の取得に使用される、**sys_reqinfo_routername** などのシステム情報の Tcl コマンド拡張。
- ポリシーからの、SMTP ライブラリ（電子メール通知を送信）または CLI ライブラリ（CLI コマンドを実行）の使用。
- 他のポリシーによって使用される Tcl 変数の保存に使用される **context_save** および **context_retrieve** の Tcl コマンド拡張。

次に、イベントを問い合わせ、本体セクションの一部としてメッセージを記録するコードの Tcl コードの例を示します。

```
# Query the event info and log a message.
array set arr_einfo [event_reqinfo]

if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}

global timer_type timer_time_sec
set timer_type $arr_einfo(timer_type)
set timer_time_sec $arr_einfo(timer_time_sec)

# Log a message.
set msg [format "timer event: timer type %s, time expired %s" \
        $timer_type [clock format $timer_time_sec]]

action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
      $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

EEM 開始ステータス

EEM ポリシーの開始ステータスの部分は、前のポリシーが同じイベントに対して実行されたかどうかや、前のポリシーの終了ステータスを特定するために、使用されます。_entry_status 変数が定義されている場合、このイベントに対して前のポリシーがすでに実行されています。_entry_status 変数の値によって、前のポリシーの戻りコードが特定されます。

開始ステータス指定には、0（前のポリシーが正常終了した）、Not=0（前のポリシーに障害が発生した）、および Undefined（実行された前のポリシーがない）の、3 つの値のうちいずれか 1 つを使用できます。

EEM 終了ステータス

ポリシーでそのコードの実行を終了すると、終了値が設定されます。終了値は、Embedded Event Manager によって使用され、このイベントのデフォルトアクションがある場合に、それが適用

されたかどうかが判断されます。ゼロの値は、デフォルトアクションが実行されていないことを意味します。ゼロではない値は、デフォルトアクションが実行されたことを意味します。終了ステータスは、同じイベントで実行される後続ポリシーに渡されます。

EEM ポリシーと Cisco エラー番号

一部の EEM Tcl コマンド拡張によって、Cisco エラー番号の Tcl グローバル変数の _cerrno が設定されます。_cerrno が設定されるたびに、他の 4 つの Tcl グローバル変数が _cerrno から分岐し、それとともに設定されます（_cerr_sub_num、_cerr_sub_err、_cerr_posix_err、および _cerr_str）。

たとえば、次の例の **action_syslog** コマンドでは、コマンド実行の副次的な影響としてこれらのグローバル変数が設定されます。

```
action_syslog priority warning msg "A sample message generated by action_syslog"
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

_cerrno：32 ビット エラー戻り値

コマンドによって設定された _cerrno は、次の形式の 32 ビットの整数を表す場合があります。

```
XYSSSSSSSSSSSSSEEEEEEEEPPPPPPPP
```

たとえば、次のエラー戻り値は、EEM Tcl コマンド拡張から戻される場合があります。

```
862439AE
```

この数字は、次の 32 ビット値として解釈されます。

```
10000110001001000011100110101110
```

この 32 ビットの整数は、次の表に示されているように、5 つの変数に分けられます。

表 *189 : _cerrno*：*32* ビット エラー戻り値の変数

変数	説明
XY	エラー クラス（エラーの重大度を示します）。この変数は、32 ビットのエラー戻り値の最初の 2 ビットに対応しています。前述のケースの 10 は、CERR_CLASS_WARNING を示します。 この変数固有の 4 つのエラー クラス エンコーディングについては、次の表を参照してください。
SSSSSSSSSSSSS	最新のエラーが生成されたサブシステム番号（13 ビット = 値 8192）。これは、32 ビット シーケンスの次の 13 ビットで、その整数値は $_cerr_sub_num に含まれています。
変数	説明

変数	説明
EEEEEEEE	サブシステム固有のエラー番号（8 ビット＝値 256）。このセグメントは、32 ビット シーケンスの次の 8 ビットで、このエラー番号に対応する文字列は、$_cerr_sub_err に含まれています。
PPPPPPPP	パススルー POSIX エラー コード（9 ビット＝値 512）。これは、32 ビット シーケンスの最後で、このエラー コードに対応する文字列は、$_cerr_posix_err に含まれています。

XY のエラー クラス エンコーディング

最初の変数 XY は、次の表に示されているように、エラー クラス エンコーディングを参照しています。

表 *190* : エラー クラス エンコーディング

00	CERR_CLASS_SUCCESS
01	CERR_CLASS_INFO
10	CERR_CLASS_WARNING
11	CERR_CLASS_FATAL

ゼロのエラー戻り値は、SUCCESS を示します。

手順の概要

1. **enable**
2. **show event manager policy available detailed** *policy-filename*
3. 画面に表示されたサンプル ポリシーの内容を、テキスト エディタにカット アンド ペーストします。
4. 必要な **event_register** Tcl コマンド拡張を定義します。
5. 適切な名前空間を、::cisco 階層構造に追加します。
6. Must Define セクションをプログラムし、このポリシーで使用される各環境変数をチェックします。
7. スクリプトの本体をプログラムします。
8. 開始ステータスをチェックし、ポリシーがこのイベントに対して前に実行されたかどうかを判断します。
9. 終了ステータスをチェックし、デフォルト アクションが存在する場合に、このイベントのデフォルト アクションが適用されたかどうかを判断します。
10. Cisco エラー番号（_cerrno）の Tcl グローバル変数を設定します。
11. 新しいファイル名で Tcl スクリプトを保存し、Tcl スクリプトをデバイスにコピーします。
12. **configure terminal**
13. **event manager directory user** {**library** *path*| **policy** *path*}
14. **event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]

15. ポリシーを実行し、ポリシーを観察します。
16. ポリシーが正しく実行されていない場合、デバッグのテクニックを使用します。

手順の詳細

ステップ 1 **enable**

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 **show event manager policy available detailed** *policy-filename*

ポリシーによって使用される環境変数と、ポリシーの実行方法の説明の詳細を含む、指定された実際のサンプル ポリシーを表示します。**detailed** キーワードが **show event manager policy available** コマンドと **show event manager policy registered** コマンドに導入されました。お使いのリリースによっては、2 つの Tcl スクリプトのいずれかをこのドキュメントの設定例セクションからコピーする必要があります（Tcl のサンプル スクリプトを使用したポリシーのプログラミングの例（2169 ページ）を参照）。次に、サンプル ポリシー tm_cli_cmd.tcl についての詳細が画面上に表示される例を示します。

例：

```
Device# show event manager policy available detailed tm_cli_cmd.tcl
```

ステップ 3 画面に表示されたサンプル ポリシーの内容を、テキスト エディタにカット アンド ペーストします。

編集機能とコピー機能を使用して、デバイスから別のデバイス上のテキスト エディタに、内容を移動します。テキスト エディタを使用して、ポリシーを Tcl スクリプトとして編集します。

ステップ 4 必要な **event_register** Tcl コマンド拡張を定義します。

検出するイベントについて、適切な **event_register** Tcl コマンド拡張を次の表から選択し、ポリシーに追加します。

表 *191 : EEM* イベント登録の *Tcl* コマンド拡張

イベント登録の Tcl コマンド拡張
event_register_appl
event_register_cli
event_register_counter
event_register_gold
event_register_interface
event_register_ioswdsysmon
event_register_ipsla

イベント登録の Tcl コマンド拡張
event_register_nf
event_register_none
event_register_oir
event_register_process
event_register_resource
event_register_rf
event_register_routing
event_register_rpc
event_register_snmp
event_register_snmp_notification
event_register_snmp_object
event_register_syslog
event_register_timer
event_register_timer_subscriber
event_register_track
event_register_wdsysmon

ステップ 5 適切な名前空間を、::cisco 階層構造に追加します。

ポリシーの開発者は、Cisco IOS EEM によって使用されるすべての拡張をグループ化するため、Tcl ポリシーで新しい名前空間 ::cisco を使用できます。::cisco 階層構造の下には、2 つの名前空間があります。次の表に、各名前空間の下に属する EEM Tcl コマンド拡張のカテゴリを示します。

表 *192 : Cisco IOS EEM* 名前空間グルーピング

Namespace	Tcl コマンド拡張のカテゴリ
::cisco::eem	EEM イベント登録
	EEM イベント情報
	EEM イベント パブリッシュ
	EEM アクション
	EEM ユーティリティ
	EEM コンテキスト ライブラリ
	EEM システム情報
	CLI ライブラリ
::cisco::lib	SMTP ライブラリ

（注） 前述のコマンドの使用時に、適切な名前空間をインポートするか、または、認定コマンド名を使用します。

ステップ 6 Must Define セクションをプログラムし、このポリシーで使用される各環境変数をチェックします。

この手順は任意です。Must Define は、ポリシーによって必要とされるすべての EEM 環境変数が、回復アクションの実行前に定義されているかどうかをテストする、ポリシーのセクションです。ポリシーによって EEM 環境変数が使用されない場合、Must Define セクションは不要です。EEM スクリプトの EEM 環境変数は、ポリシーの実行前にポリシーに対して外部定義された Tcl グローバル変数です。EEM 環境変数を定義するには、Embedded Event Manager コンフィギュレーション コマンド **event manager environment** CLI コマンドを使用します。規則として、すべてのシスコ EEM 環境変数の先頭は、「_」（アンダースコア）になっています。将来的な競合を避けるため、「_」で始まる新しい変数を定義しないことを推奨します。

（注） **show event manager environment** 特権 EXEC コマンドを使用して、システムの Embedded Event Manager 環境変数セットを表示できます。

たとえば、サンプル ポリシーで定義されている Embedded Event Manager 環境変数には、E メール変数が含まれます。適切に動作させるためには、電子メールを送信するサンプル ポリシーに、次の表に示す変数が設定されている必要があります。

次の表で EEM サンプル ポリシーで使用される電子メール特有の環境変数について説明します。

表 *193 :* サンプル ポリシーで使用される電子メール特有の環境変数

環境変数	説明	例
_email_server	E メール送信に使用されるシンプル メール転送プロトコル（SMTP）メールサーバー。	E メール サーバー名は、次のテンプレート形式のいずれかで使用できます。 • username:password@host • username@host • ホスト
_email_to	E メールの送信先アドレス。	engineering@example.com
_email_from	E メールの送信元アドレス。	devtest@example.com
_email_cc	E メールのコピーの送信先アドレス。	manager@example.com

次に、E メール特有の環境変数のプログラムをチェックする Must Define セクションの例を示します。

Must Define の例

例：

```
if {![info exists _email_server]} {
    set result \
        "Policy cannot be run: variable _email_server has not been set"
    error $result $errorInfo
}
if {![info exists _email_from]} {
    set result \
        "Policy cannot be run: variable _email_from has not been set"
    error $result $errorInfo
}
if {![info exists _email_to]} {
    set result \
        "Policy cannot be run: variable _email_to has not been set"
    error $result $errorInfo
}
if {![info exists _email_cc]} {
    set result \
        "Policy cannot be run: variable _email_cc has not been set"
    error $result $errorInfo
}
```

ステップ 7 スクリプトの本体をプログラムします。

スクリプトのこのセクションでは、次のいずれかを定義できます。

- 検出されたイベントに関する情報の EEM への問い合わせに使用される **event_reqinfo** イベント情報の Tcl コマンド拡張。
- EEM 特有のアクションの指定に使用される、**action_syslog** などのアクション Tcl コマンド拡張。
- 一般的なシステム情報の取得に使用される、**sys_reqinfo_routername** などのシステム情報の Tcl コマンド拡張。

- 他のポリシーによって使用される Tcl 変数の保存に使用される **context_save** および **context_retrieve** の Tcl コマンド拡張。
- ポリシーからの、SMTP ライブラリ（電子メール通知を送信）または CLI ライブラリ（CLI コマンドを実行）の使用。

ステップ 8 開始ステータスをチェックし、ポリシーがこのイベントに対して前に実行されたかどうかを判断します。

前のポリシーが正常終了した場合、現在のポリシーは、実行が必要な場合と、実行が不要な場合があります。開始ステータス指定には、0（前のポリシーが正常終了した）、Not=0（前のポリシーに障害が発生した）、および Undefined（実行された前のポリシーがない）の、3 つの値のうちいずれか 1 つを使用できます。

ステップ 9 終了ステータスをチェックし、デフォルトアクションが存在する場合に、このイベントのデフォルトアクションが適用されたかどうかを判断します。

ゼロの値は、デフォルト アクションが実行されていないことを意味します。ゼロではない値は、デフォルト アクションが実行されたことを意味します。終了ステータスは、同じイベントで実行される後続ポリシーに渡されます。

ステップ 10 Cisco エラー番号（_cerrno）の Tcl グローバル変数を設定します。

一部の EEM Tcl コマンド拡張によって、Cisco エラー番号の Tcl グローバル変数の _cerrno が設定されます。_cerrno が設定されるたびに、他の 4 つの Tcl グローバル変数が _cerrno から分岐し、それとともに設定されます（_cerr_sub_num、_cerr_sub_err、_cerr_posix_err、および _cerr_str）。

たとえば、次の例の **action_syslog** コマンドでは、コマンド実行の副次的な影響としてこれらのグローバル変数が設定されます。

例：

```
action_syslog priority warning msg "A sample message generated by action_syslog
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

ステップ 11 新しいファイル名で Tcl スクリプトを保存し、Tcl スクリプトをデバイスにコピーします。

Embedded Event Manager ポリシー ファイル名は、次の仕様に従っています。

- オプションのプレフィックス Mandatory. がある場合、これは、システム ポリシーがまだ登録されていない場合に、自動的に登録される必要があるシステムポリシーであることを示します。たとえば、Mandatory.sl_text.tcl などです。
- 指定された 1 つめのイベントの 2 文字の省略形が含まれるファイル名の本体部（EEM ポリシーと Cisco エラー番号 （2145 ページ）を参照）、下線文字部、および、ポリシーをさらに示す説明フィールド部。
- ファイル名拡張子部は .tcl と定義されます。

詳細については、EEM 用のシスコ ファイル命名規則 （2126 ページ）を参照してください。

デバイスのフラッシュ ファイル システム（通常は disk0:）にファイルをコピーします。ファイルのコピーの詳細については、『Cisco IOS Configuration Fundamentals Configuration Guide』の「Using the Cisco IOS File System」の章を参照してください。

デバイスのフラッシュ ファイル システム（通常は bootflash:）にファイルをコピーします。ファイルのコピーの詳細については、『Cisco IOS Configuration Fundamentals Configuration Guide』の「Using the Cisco IOS File System」の章を参照してください。

ステップ 12 **configure terminal**

グローバル コンフィギュレーション モードを開始します。

例：

```
Device# configure terminal
```

ステップ 13 **event manager directory user** {**library** *path*| **policy** *path*}

ユーザー ライブラリ ファイルまたはユーザー定義 EEM ポリシーの保存に使用するディレクトリを指定します。次に、disk0 の user_library ディレクトリが、ユーザー ライブラリ ファイルを保存するディレクトリとして指定されます。

ユーザー ライブラリ ファイルまたはユーザー定義 EEM ポリシーの保存に使用するディレクトリを指定します。次に、bootflash の user_library ディレクトリが、ユーザー ライブラリ ファイルを保存するディレクトリとして指定されます。

例：

```
Device(config)# event manager directory user library disk0:/user_library

Device(config)# event manager directory user library bootflash:/user_library
```

ステップ 14 **event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]

ポリシー内で定義された指定イベントが発生した場合に、EEM ポリシーを実行するよう、定義します。次に、cl_mytest.tcl という名前の EEM ポリシーが、ユーザー定義ポリシーとして登録される例を示します。

例：

```
Device(config)# event manager policy cl_mytest.tcl type user
```

ステップ 15 ポリシーを実行し、ポリシーを観察します。

ポリシーの実行をテストするには、ポリシーが実行される原因となる条件を生成し、ポリシーが想定どおりに実行されていることを確認します。

ステップ 16 ポリシーが正しく実行されていない場合、デバッグのテクニックを使用します。

Cisco IOS **debug event manager** CLI コマンドをそのさまざまなキーワードとともに使用して、問題をデバッグします。Tcl 特有のキーワード使用の詳細については、トラブルシューティングのヒント （2153 ページ）を参照してください。

トラブルシューティングのヒント

- Tcl 拡張コマンドの問題をデバッグするには、**debug event manager tcl commands** コマンドを使用します。イネーブルの場合、このコマンドによって、CLI のやり取りを処理する TTY セッションに渡され、TTY セッションから読み戻される、すべてのデータが表示されます。このデータを使用すると、ユーザーが CLI に渡すコマンドが有効になります。
- CLI ライブラリを使用すると、ユーザーは、CLI コマンドを実行し、Tcl のコマンドの出力を取得できます。**debug event manager tcl cli-library** CLI コマンドを使用して、CLI ライブラリの問題をデバッグします。
- SMTP ライブラリを使用すると、ユーザーは、SMTP E メール サーバーへ、E メール メッセージを送信できます。**debug event manager tcl smtp_library** CLI コマンドを使用して、SMTP ライブラリの問題をデバッグします。イネーブルの場合、このコマンドによって、SMTP ライブラリ ルーチンに渡され、SMTP ライブラリ ルーチンから読み戻される、すべてのデータが表示されます。このデータを使用すると、ユーザーが SMTP ライブラリに渡すコマンドが有効になります。
- Tcl は、コマンドを上書きできる融通性のある言語です。たとえば、**set** コマンドを変更し、スカラ変数が設定されたときにメッセージを表示する **set** コマンドのバージョンを作成します。ポリシーに **set** コマンドが入力されると、スカラ変数が設定されたときにはいつでもメッセージが表示され、スカラ変数をデバッグする方法が示されます。このデバッグ テクニックの例を参照するには、Tcl set コマンド操作のトレースの例 （2181 ページ）を参照してください。

これらのデバッグ テクニックのいくつかの例を参照するには、Embedded Event Manager ポリシーのデバッグの例 （2179 ページ）を参照してください。

EEM ユーザー Tcl ライブラリ索引の作成

Tcl ファイルのライブラリに含まれているすべての手順のディレクトリが含まれている、索引ファイルを作成するには、この作業を実行します。この作業によって、EEM Tcl でライブラリ サポートをテストできます。この作業では、Tcl ライブラリ ファイルが含まれるライブラリ ディレクトリが作成され、ファイルがディレクトリにコピーされ、ライブラリ ファイルにあるすべての手順のディレクトリが含まれる索引 tclIndex が作成されます。索引が作成されない場合、Tcl 手順を参照する EEM ポリシーを実行するときに、Tcl 手順は見つかりません。

手順の概要

1. ワークステーション（UNIX、Linux、PC、または Mac）で、ライブラリ ディレクトリを作成し、Tcl ライブラリ ファイルをディレクトリにコピーします。
2. **tclsh**
3. **auto_mkindex** *directory_name* ***.tcl**
4. ターゲット デバイス上のユーザー ライブラリ ファイルの保存に使用されるディレクトリに Tcl ライブラリ ファイルと tclIndex ファイルをコピーします。
5. Tcl で記述されたユーザー定義 EEM ポリシー ファイルを、ターゲット デバイス上でユーザー定義 EEM ポリシーの保存に使用されるディレクトリにコピーします。

6. **enable**
7. **configure terminal**
8. **event manager directory user library** *path*
9. **event manager directory user policy** *path*
10. **event manager policy** *policy-name* [**type** {**system** | **user**} [**trap]**
11. **event manager run** *policy-name*

手順の詳細

ステップ 1 ワークステーション（UNIX、Linux、PC、または Mac）で、ライブラリ ディレクトリを作成し、Tcl ライブラリ ファイルをディレクトリにコピーします。

次の例ファイルを使用すると、Tcl シェルが実行されているワークステーション上で、tclIndex を作成できます。

lib1.tcl

例：

```
proc test1 {} {
    puts "In procedure test1"
}

proc test2 {} {
    puts "In procedure test2"
}
```

lib2.tcl

例：

```
proc test3 {} {
    puts "In procedure test3"
}
```

ステップ 2 **tclsh**

このコマンドを使用して、Tcl シェルを開始します。

例：

```
workstation% tclsh
```

ステップ 3 **auto_mkindex** *directory_name* ***.tcl**

auto_mkindex コマンドを使用して、tclIndex ファイルを作成します。すべての手順のディレクトリが含まれる tclIndex ファイルは、Tcl ライブラリ ファイルに含まれていました。どのディレクトリにも 1 つの tclIndex ファイルのみを存在させることができ、他の Tcl ファイルはグループ化しておくことが可能であるため、ディレクトリ内で auto_mkindex を実行することを推奨します。ディレクトリ内で auto_mkindex を実行すると、特定の tclIndex を使用してどの Tcl ソース ファイルを索引化できるかが判断されます。

例：

```
workstation% auto_mkindex eem_library *.tcl
```

lib1.tcl ファイルと lib2.tcl ファイルがライブラリ ファイル ディレクトリにあり、**auto_mkindex** コマンドが実行されたときに、次の例に示す tclIndex が作成されます。

tclIndex

例：

```
# Tcl autoload index file, version 2.0
# This file is generated by the "auto_mkindex" command
# and sourced to set up indexing information for one or
# more commands.  Typically each line is a command that
# sets an element in the auto_index array, where the
# element name is the name of a command and the value is
# a script that loads the command.

set auto_index(test1) [list source [file join $dir lib1.tcl]]
set auto_index(test2) [list source [file join $dir lib1.tcl]]
set auto_index(test3) [list source [file join $dir lib2.tcl]]
```

ステップ 4 ターゲット デバイス上のユーザー ライブラリ ファイルの保存に使用されるディレクトリに Tcl ライブラリ ファイルと tclIndex ファイルをコピーします。

ステップ 5 Tcl で記述されたユーザー定義 EEM ポリシー ファイルを、ターゲット デバイス上でユーザー定義 EEM ポリシーの保存に使用されるディレクトリにコピーします。

ユーザー定義 EEM ポリシーを保存するディレクトリは、ステップ 4 で使用されるディレクトリと同じディレクトリを使用できます。次に、EEM でサポートされる Tcl ライブラリのテストに、ユーザー定義 EEM ポリシーを使用できる例を示します。

libtest.tcl

例：

```
::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

global auto_index auto_path

puts [array names auto_index]

if { [catch {test1} result]} {
    puts "calling test1 failed result = $result $auto_path"
}

if { [catch {test2} result]} {
    puts "calling test2 failed result = $result $auto_path"
}
if { [catch {test3} result]} {
    puts "calling test3 failed result = $result $auto_path"
}
```

ステップ 6 **enable**

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 7 **configure terminal**

グローバル コンフィギュレーション モードをイネーブルにします。

例：

```
Device# configure terminal
```

ステップ 8 **event manager directory user library** *path*

このコマンドを使用して、EEM ユーザー ライブラリ ディレクトリを指定します。これは、ファイルがコピーされたディレクトリです。

例：

```
Device(config)# event manager directory user library disk2:/eem_library
```

ステップ 9 **event manager directory user policy** *path*

このコマンドを使用して、EEM ユーザー ポリシー ディレクトリを指定します。これは、ファイルがコピーされたディレクトリです。

例：

```
Device(config)# event manager directory user policy disk2:/eem_policies
```

ステップ 10 **event manager policy** *policy-name* [**type** {**system** | **user**} [**trap]**

このコマンドを使用して、ユーザー定義 EEM ポリシーを登録します。この例では、libtest.tcl という名前のポリシーが登録されます。

例：

```
Device(config)# event manager policy libtest.tcl
```

ステップ 11 **event manager run** *policy-name*

このコマンドを使用して、手作業で EEM ポリシーを実行します。この例では、libtest.tcl という名前のポリシーが実行され、EEM の Tcl サポートがテストされます。次に、EEM の Tcl サポートが正常終了した出力例を示します。

例：

```
Device(config)# event manager run libtest.tcl
The following output is displayed:
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test1
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test2
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test3
```

EEM ユーザー Tcl パッケージ索引の作成

すべての Tcl パッケージのディレクトリと、Tcl パッケージ ファイルのライブラリに含まれるバージョン情報が含まれる、Tcl パッケージの索引ファイルを作成するには、この作業を実行

します。使用しているリリースによっては、Tcl **package** キーワードを使用することで Tcl パッケージがサポートされます。

Tcl パッケージは、EEM システム ライブラリ ディレクトリまたは EEM ユーザー ライブラリ ディレクトリのいずれかにあります。**package require** Tcl コマンドが実行されると、ユーザー ライブラリ ディレクトリで、まず、pkgIndex.tcl ファイルが検索されます。pkgIndex.tcl ファイルがユーザー ディレクトリで見つからない場合、システム ライブラリ ディレクトリが検索されます。この作業では、**pkg_mkIndex** コマンドを使用して、適切なライブラリディレクトリに Tcl パッケージディレクトリ（pkgIndex.tcl ファイル）が作成され、バージョン情報とともに、ディレクトリ内にあるすべての Tcl パッケージについての情報が含められます。索引が作成されない場合、**package require** Tcl コマンドが含まれる、EEM ポリシーが実行されたときに、Tcl パッケージは見つかりません。

EEM で Tcl パッケージ サポートを使用すると、ユーザーは、Tcl の XML_RPC などのパッケージにアクセスできます。Tcl パッケージ インデックスが作成されるとき、Tcl スクリプトは、外部エンティティに対する XML-RPC 呼び出しを容易に行うことができます。

（注） C プログラミング コードで実装されるパッケージは、EEM ではサポートされません。

手順の概要

1. ワークステーション（UNIX、Linux、PC、または Mac）で、ライブラリ ディレクトリを作成し、Tcl パッケージ ファイルをディレクトリにコピーします。
2. **tclsh**
3. **pkg_mkindex** *directory_name* ***.tcl**
4. ターゲット デバイス上のユーザー ライブラリ ファイルの保存に使用されるディレクトリに Tcl ライブラリ ファイルと pkgIndex ファイルをコピーします。
5. Tcl で記述されたユーザー定義 EEM ポリシー ファイルを、ターゲット デバイス上でユーザー定義 EEM ポリシーの保存に使用されるディレクトリにコピーします。
6. **enable**
7. **configure terminal**
8. **event manager directory user library** *path*
9. **event manager directory user policy** *path*
10. **event manager policy** *policy-name* [**type** {**system** | **user**} [**trap**]
11. **event manager run** *policy-name*

手順の詳細

ステップ 1 ワークステーション（UNIX、Linux、PC、または Mac）で、ライブラリ ディレクトリを作成し、Tcl パッケージ ファイルをディレクトリにコピーします。

ステップ 2 **tclsh**

このコマンドを使用して、Tcl シェルを開始します。

例：

```
workstation% tclsh
```

ステップ 3 **pkg_mkindex** *directory_name* ***.tcl**

pkg_mkindex コマンドを使用して、pkgIndex ファイルを作成します。すべてのパッケージのディレクトリが含まれる pkgIndex ファイルは、Tcl ライブラリ ファイルに含まれていました。どのディレクトリにも 1 つの pkgIndex ファイルのみを存在させることができ、他の Tcl ファイルはグループ化しておくことが可能であるため、ディレクトリ内で pkg_mkindex を実行することを推奨します。ディレクトリ内で pkg_mkindex を実行すると、特定の pkgIndex を使用してどの Tcl パッケージ ファイルを索引化できるかが判断されます。

例：

```
workstation% pkg_mkindex eem_library *.tcl
```

次に、いくつかの Tcl パッケージがライブラリ ファイル ディレクトリにあり、**pkg_mkindex** コマンドが実行されたときに、pkgIndex が作成される例を示します。

pkgIndex

例：

```
# Tcl package index file, version 1.1
# This file is generated by the "pkg_mkIndex" command
# and sourced either when an application starts up or
# by a "package unknown" script. It invokes the
# "package ifneeded" command to set up package-related
# information so that packages will be loaded automatically
# in response to "package require" commands. When this
# script is sourced, the variable $dir must contain the
# full path name of this file's directory.
package ifneeded xmlrpc 0.3 [list source [file join $dir xmlrpc.tcl]]
```

ステップ 4 ターゲット デバイス上のユーザー ライブラリ ファイルの保存に使用されるディレクトリに Tcl ライブラリ ファイルと pkgIndex ファイルをコピーします。

ステップ 5 Tcl で記述されたユーザー定義 EEM ポリシー ファイルを、ターゲット デバイス上でユーザー定義 EEM ポリシーの保存に使用されるディレクトリにコピーします。

ユーザー定義 EEM ポリシーを保存するディレクトリは、ステップ 4 で使用されるディレクトリと同じディレクトリを使用できます。次に、EEM でサポートされる Tcl パッケージのテストに、ユーザー定義 EEM ポリシーを使用できる例を示します。

packagetest.tcl

例：

```
::cisco::eem::event_register_none maxrun 1000000.000
#
# test if xmlrpc available
#
#
# Namespace imports
#
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
#
```

```
package require xmlrpc
puts "Did you get an error?"
```

ステップ 6 **enable**

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 7 **configure terminal**

グローバル コンフィギュレーション モードをイネーブルにします。

例：

```
Device# configure terminal
```

ステップ 8 **event manager directory user library** *path*

このコマンドを使用して、EEM ユーザー ライブラリ ディレクトリを指定します。これは、ファイルがコピーされたディレクトリです。

例：

```
Device(config)# event manager directory user library disk2:/eem_library
```

ステップ 9 **event manager directory user policy** *path*

このコマンドを使用して、EEM ユーザー ポリシー ディレクトリを指定します。これは、ファイルがコピーされたディレクトリです。

例：

```
Device(config)# event manager directory user policy disk2:/eem_policies
```

ステップ 10 **event manager policy** *policy-name* [**type** {**system** | **user**} [**trap**]

このコマンドを使用して、ユーザー定義 EEM ポリシーを登録します。この例では、packagetest.tcl という名前のポリシーが登録されます。

例：

```
Device(config)# event manager policy packagetest.tcl
```

ステップ 11 **event manager run** *policy-name*

このコマンドを使用して、手作業で EEM ポリシーを実行します。この例では、packagetest.tcl という名前のポリシーが実行され、EEM の Tcl パッケージ サポートがテストされます。

例：

```
Device(config)# event manager run packagetest.tcl
```

Tcl を使用した Embedded Event Manager（EEM）ポリシー記述の設定例

Tcl セッションへのユーザー名割り当ての例

次に、Tcl セッションに関連付けられるユーザー名を設定する例を示します。認証、認可、カウンティング（AAA）セキュリティを使用し、コマンドベースで認可を実装する場合、**event manager session cli username** コマンドを使用して、Tcl セッションに関連付けられるユーザー名を設定する必要があります。Tcl ポリシーによって CLI コマンドが実行されるときに、ユーザー名が使用されます。TACACS+ では、ポリシーを実行している Tcl セッションに関連付けられているユーザー名を使用して、各 CLI コマンドが確認されます。ポリシーを登録するには、デバイスが特権 EXEC モードである必要があるため、Tcl ポリシーからのコマンドは、通常、確認されません。この例では、ユーザー名は yourname で、これは、CLI コマンドセッションが EEM ポリシー内から開始されるたびに使用されるユーザー名です。

```
configure terminal
 event manager session cli username yourname
 end
```

EEM イベント ディテクタのデモの例

EEM サンプル ポリシーの説明

この設定例では、一部の EEM サンプル ポリシーについて説明します。

- ap_perf_test_base_cpu.tcl：EEM ポリシーの CPU パフォーマンスを測定するために実行されます。
- no_perf_test_init.tcl：EEM ポリシーの CPU パフォーマンスを測定するために実行されます。
- sl_intf_down.tcl：設定可能な syslog メッセージが記録されるときに実行されます。最大で 2 つまでの CLI コマンドを実行し、結果が E メールで送信されます。
- tm_cli_cmd.tcl：設定可能な CRON エントリを使用して実行されます。設定可能な CLI コマンドが実行され、結果が電子メールで送信されます。
- tm_crash_reporter.tcl：登録後の 5 秒間と、デバイスの起動後の 5 秒間に実行されます。トリガーされると、スクリプトによって、リロード原因の検索が試行されます。リロードの原因がクラッシュの場合、ポリシーによって、関連する crashinfo ファイルが検索され、環境変数 _crash_reporter_url でユーザーによって指定された URL へ、この情報が送信されます。

- tm_fsys_usage.tcl：このポリシーは、設定可能なCRONエントリを使用して実行され、ディスク領域の使用状況を監視します。ディスク領域の使用状況が、設定可能なしきい値を超えると、Syslog メッセージが表示されます。

サンプル ポリシーのイベント マネージャ環境変数

イベント マネージャ環境変数は、ポリシーの登録および実行の前に EEM ポリシーに対して外部定義された Tcl グローバル変数です。サンプル ポリシーでは、3 つの電子メール環境変数が設定されている必要があります。_email_cc のみが省略可能です。他の必須および任意の変数設定については、次の表で説明します。

次の表に、ap_perf_test_base_cpu.tcl サンプル ポリシーの実行前に設定する必要がある EEM 環境変数を示します。

表 *194 : ap_perf_test_base_cpu.tcl* ポリシーで使用される環境変数

環境変数	説明	例
_perf_iterations	測定を反復する回数。	**100**
_perf_cmd1	測定テストの一部として実行される最初の非インタラクティブ CLI コマンド。この変数は任意で、指定する必要はありません。	**enable**
_perf_cmd2	測定テストの一部として実行される 2 番目の非インタラクティブ CLI コマンド。_perf_cmd2 を使用するには、_perf_cmd1 を定義する必要があります。この変数は任意で、指定する必要はありません。	**show version**
_perf_cmd3	測定テストの一部として実行される 3 番目の非インタラクティブ CLI コマンド。_perf_cmd3 を使用するには、_perf_cmd1 を定義する必要があります。この変数は任意で、指定する必要はありません。	**show interface counters protocol status**

次の表に、no_perf_test_init.tcl サンプル ポリシーの実行前に設定する必要がある EEM 環境変数を示します。

表 *195 : no_perf_test_init.tcl* ポリシーで使用される環境変数

環境変数	説明	例
_perf_iterations	測定を反復する回数。	**100**
_perf_cmd1	測定テストの一部として実行される最初の非インタラクティブ CLI コマンド。この変数は任意で、指定する必要はありません。	**enable**

環境変数	説明	例
_perf_cmd2	測定テストの一部として実行される 2 番目の非インタラクティブ CLI コマンド。_perf_cmd2 を使用するには、_perf_cmd1 を定義する必要があります。この変数は任意で、指定する必要はありません。	**show version**
_perf_cmd3	測定テストの一部として実行される 3 番目の非インタラクティブ CLI コマンド。_perf_cmd3 を使用するには、_perf_cmd1 を定義する必要があります。この変数は任意で、指定する必要はありません。	**show interface counters protocol status**

次の表に、sl_intf_down.tcl サンプル ポリシーの実行前に設定する必要がある EEM 環境変数を示します。

表 *196 : sl_intf_down.tcl* ポリシーで使用される環境変数

環境変数	説明	例
_config_cmd1	実行される 1 番めのコンフィギュレーション コマンド。	**interface Ethernet1/0**
_config_cmd2	実行される 2 番めのコンフィギュレーション コマンド。この変数は任意で、指定する必要はありません。	**no shutdown**
_syslog_pattern	ポリシー実行時を決定するために syslog メッセージを比較するために使用する正規表現パターン マッチ文字列。	.*UPDOWN.*FastEthernet0/0.*

次の表に、tm_cli_cmd.tcl サンプル ポリシーの実行前に設定する必要がある EEM 環境変数を示します。

表 *197 : tm_cli_cmd.tcl* ポリシーで使用される環境変数

環境変数	説明	例
_cron_entry	ポリシーが実行されるときを決定する CRON 仕様。	0-59/1 0-23/1 * * 0-7
_show_cmd	ポリシーの実行時に実行される CLI コマンド。	**show version**

次の表に、tm_crash_reporter.tcl サンプル ポリシーの実行前に設定する必要がある EEM 環境変数を示します。

表 *198 : tm_crash_reporter.tcl* ポリシーで使用される環境変数

環境変数	説明	例
_crash_reporter_debug	tm_crash_reporter.tcl のデバッグ情報がイネーブルであるかどうかを決定する値。この変数は任意で、指定する必要はありません。	1
_crash_reporter_url	クラッシュ レポートが送信される URL 位置。	http://www.example.com/fm/interface_tm.cgi

次の表に、tm_fsys_usage.tcl サンプル ポリシーの実行前に設定する必要がある EEM 環境変数を示します。

表 *199 : tm_fsys_usage.tcl* ポリシーで使用される環境変数

環境変数	説明	例
_tm_fsys_usage_cron	**event_register** Tcl コマンド拡張で使用される CRON 仕様。指定されない場合、tm_fsys_usage.tcl ポリシーが 1 分に 1 回トリガーされます。この変数は任意で、指定する必要はありません。	0-59/1 0-23/1 * * 0-7
_tm_fsys_usage_debug	この変数が値 1 に設定された場合、システムのすべてのエントリのディスク使用率情報が表示されます。この変数は任意で、指定する必要はありません。	1
_tm_fsys_usage_freebytes	システムまたは特定のプレフィックスの空きバイト数しきい値。空きスペースが所定の値を下回ると、警告が表示されます。この変数は任意で、指定する必要はありません。	disk2:98000000
_tm_fsys_usage_percent	システムまたは特定のプレフィックスのディスク使用割合しきい値。ディスク使用割合が所定の割合を超えると、警告が表示されます。指定されない場合、すべてのシステムのデフォルトのディスク使用割合は、80% です。この変数は任意で、指定する必要はありません。	nvram:25 disk2:5

一部の EEM ポリシーの登録

ポリシーの登録後に EEM 環境変数が変更された場合、一部の EEM ポリシーは、登録を解除し、再登録する必要があります。ポリシーの開始時に表示される event_register_*xxx* ステートメントには、一部の EEM 環境変数が含まれ、このステートメントは、ポリシーが実行される条件の確立に使用されます。ポリシーの登録後に環境変数が変更された場合、条件は無効になり

ます。いかなるエラーも回避するには、ポリシーの登録を解除し、再登録する必要があります。次の変数に影響が及ぼされます。

- _cron_entry in the tm_cli_cmd.tcl policy
- _syslog_pattern in the sl_intf_down.tcl policy

すべてのサンプル ポリシーの基本設定の詳細

Embedded Event Manager から電子メールを送信できるようにするには、**hostname** コマンドと **ip domain-name** コマンドを設定する必要があります。EEM 環境変数も設定する必要があります。Cisco IOS イメージのブート後、次の初期設定を使用し、ネットワークで適切な値を置き換えます。tm_fsys_usage サンプル ポリシーの環境変数（上の表を参照）はすべて任意で、ここではそのリストは示されていません。

```
hostname cpu
ip domain-name example.com
event manager environment _email_server ms.example.net
event manager environment _email_to username@example.net
event manager environment _email_from engineer@example.net
event manager environment _email_cc projectgroup@example.net
event manager environment _cron_entry 0-59/2 0-23/1 * * 0-7
event manager environment _show_cmd show event manager policy registered
event manager environment _syslog_pattern .*UPDOWN.*FastEthernet0/0
event manager environment _config_cmd1 interface Ethernet1/0
event manager environment _config_cmd2 no shutdown
event manager environment _crash_reporter_debug 1
event manager environment _crash_reporter_url
http://www.example.com/fm/interface_tm.cgi
end
```

サンプル ポリシーの使用

ここでは、次の設定シナリオを使用して、Tcl サンプル ポリシーを使用する方法について説明します。

Mandatory.go_*.tcl サンプル ポリシーの実行

GOLD EEM ポリシーの一部として実行される各テストに GOLD TCL スクリプトがあります。この TCL スクリプトをテスト用に変更したり、連続障害回数を指定することができ、また、デフォルトの是正アクションを変更することもできます。たとえば、他の CLI ベースのアクションをリセットするのではなく、ラインカードの電源を切ることができます。

登録済みのテストごとにデフォルトの TCL スクリプトを使用できます。このスクリプトはシステムに登録し、デフォルトのアクションと一致させることができます。これは、これらのスクリプトによってオーバーライドできます。

次の表は、GOLD が EEM にインストールした必須ポリシーのリストです。各ポリシーが、カードのリセットやポートの無効化といった何らかのアクションを実行します。

GOLD Tcl スクリプト	テスト
Mandatory.go_asicsync.tcl	TestAsicSync

GOLD Tcl スクリプト	テスト
Mandatory.go_bootup.tcl	すべての ブートアップ テストに共通。
Mandatory.go_fabric.tcl	TestFabricHealth
Mandatory.go_fabrich0.tcl	TestFabricCh0Health
Mandatory.go_fabrich1.tcl	TestFabricCh1Health
Mandatory.go_ipsec.tcl	TestIPSecEncrypDecrypPkt
Mandatory.go_mac.tcl	TestMacNotification
Mandatory.go_nondislp.tcl	TestNonDisruptiveLoopback
Mandatory.go_scratchreg.tcl	TestScratchRegister
Mandatory.go_sprping.tcl	TestSPRPInbandPing

次に、このポリシーの使用方法を示すサンプル設定について説明します。ユーザーEXECモードを開始し、デバイスプロンプトで**enable**コマンドを入力します。デバイスは特権EXECモードを開始します。このモードで **show event manager policy registered** コマンドを入力すると、現在登録されているポリシーがないことを確認できます。次のコマンドはどのポリシーがインストールできるかを表示する **show event manager policy available** コマンドです。**configure terminal** コマンドを入力してグローバル コンフィギュレーション モードが開始されたら、**event manager policy** コマンドを使用して mandatory.go_*.tcl ポリシーを EEM に登録できます。グローバル コンフィギュレーション モードを終了し、もう一度 **show event manager policy registered** コマンドを入力してポリシーが登録されていることを確認します。

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy Mandatory.go_spuriousisr.tcl
 end
show event manager policy registered
show event manager environment
```

ap_perf_test_base_cpu.tcl および no_perf_test_init.tcl サンプル ポリシーの実行

これらのサンプル ポリシーは、EEM ポリシーの CPU パフォーマンスを測定します。これらのポリシーは、各 EEM ポリシーの標準実行時間の検出に役立ち、CLI ライブラリ コマンドを使用して EEM 環境変数の perf_cmd1（任意で _perf_cmd2 および _perf_cmd3）で指定されているコンフィギュレーション コマンドを実行します。

次に、このポリシーの使用方法を示すサンプル設定について説明します。ユーザーEXECモードを開始し、デバイスプロンプトで**enable**コマンドを入力します。デバイスは特権EXECモードを開始します。このモードで **show event manager policy registered** コマンドを入力すると、現在登録されているポリシーがないことを確認できます。次のコマンドはどのポリシーがインストールできるかを表示する **show event manager policy available** コマンドです。**configure terminal** コマンドを入力してグローバル コンフィギュレーション モードが開始された後に

service timestamps debug datetime msec コマンドを入力すると、**event manager policy** コマンドを使用して EEM に ap_perf_test_base_cpu.tcl ポリシーと no_perf_test_init.tcl ポリシーを登録できます。グローバル コンフィギュレーション モードを終了し、もう一度 **show event manager policy registered** コマンドを入力してポリシーが登録されていることを確認します。

ポリシー ap_perf_test_base_cpu.tcl および no_perf_test_init.tcl はセットで実行されるので、一緒に登録する必要があります。no_perf_test_init.tcl ポリシーを実行し、テストを開始することができます。反復ごとに返ってくる syslog メッセージを使用して結果を分析します。反復の総回数は、変数 _perf_iterations で指定します。時間の差を測り、反復の総回数で除算して、各 EEM ポリシーの平均実行時間を計算します。

```
enable
show event manager policy registered
show event manager policy available
show event manager environment
configure terminal
 service timestamps debug datetime msec
 event manager environment _perf_iterations 100
 event manager policy ap_perf_test_base_cpu.tcl
 event manager policy no_perf_test_init.tcl
 end
show event manager policy registered
show event manager policy available
show event manager environment
event manager run no_perf_test_init.tcl
```

no_perf_test_init.tcl サンプル ポリシーの実行

このサンプル ポリシーでは、EEM ポリシーの CPI パフォーマンスを測定します。このポリシーは、各 EEM ポリシーの標準実行時間の検出に役立ち、CLI ライブラリ コマンドを使用して EEM 環境変数の perf_cmd1（任意で _perf_cmd2 および _perf_cmd3）で指定されているコンフィギュレーション コマンドを実行します。

次に、このポリシーの使用方法を示すサンプル設定について説明します。ユーザー EXEC モードを開始し、デバイス プロンプトで **enable** コマンドを入力します。デバイスは特権 EXEC モードを開始します。このモードで **show event manager policy registered** コマンドを入力すると、現在登録されているポリシーがないことを確認できます。次のコマンドはどのポリシーがインストールできるかを表示する **show event manager policy available** コマンドです。**configure terminal** コマンドを入力してグローバル コンフィギュレーション モードが開始されたら、**event manager policy** コマンドを使用して no_perf_test_init.tcl ポリシーを EEM に登録できます。グローバル コンフィギュレーション モードを終了し、もう一度 **show event manager policy registered** コマンドを入力してポリシーが登録されていることを確認します。

反復ごとに返ってくる syslog メッセージを使用して結果を分析します。反復の総回数は、変数 _perf_iterations で指定します。時間の差を測り、反復の総回数で除算して、各 EEM ポリシーの平均実行時間を計算します。

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy no_perf_test_init.tcl
 end
```

```
show event manager policy registered
show event manager environment
```

sl_intf_down.tcl サンプル ポリシーの実行

このサンプル ポリシーでは、特定のパターンで Syslog メッセージが記録されるときに設定を変更する機能について説明します。ポリシーでは、イベントについての詳細情報が収集され、CLI ライブラリを使用して、EEM 環境変数 _config_cmd1 と、任意で _config_cmd2 で指定された、コンフィギュレーション コマンドが実行されます。CLI コマンドの結果とともに、電子メール メッセージが送信されます。

次に、このポリシーの使用方法を示すサンプル設定について説明します。ユーザーEXECモードを開始し、デバイスプロンプトで**enable**コマンドを入力します。デバイスは特権EXECモードを開始します。このモードで **show event manager policy registered** コマンドを入力すると、現在登録されているポリシーがないことを確認できます。次のコマンドはどのポリシーがインストールできるかを表示する **show event manager policy available** コマンドです。**configure terminal**コマンドを入力してグローバルコンフィギュレーションモードが開始されたら、**event manager policy** コマンドを使用して sl_intf_down.tcl ポリシーを EEM に登録できます。グローバル コンフィギュレーション モードを終了し、もう一度 **show event manager policy registered** コマンドを入力してポリシーが登録されていることを確認します。

インターフェイスがダウンするときに、ポリシーが実行されます。**show event manager environment** コマンドを入力して現在の環境変数の値を表示します。_syslog_pattern EEM 環境変数で指定されたインターフェイスのケーブルを取り外します（またはシャットダウンを設定します）。インターフェイスがダウンし、インターフェイスがダウンしていることについてのSyslogメッセージを記録するSyslogデーモンのプロンプトが表示されて、Syslogイベントディテクタが呼び出されます。

Syslog イベント ディテクタによって、未解決のイベント仕様が見直され、インターフェイス ステータス変更に対する一致が検索されます。EEM サーバーに通知され、サーバーでは、このイベント sl_intf_down.tcl を処理するために登録されたポリシーが実行されます。

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy sl_intf_down.tcl
 end
show event manager policy registered
show event manager environment
```

tm_cli_cmd.tcl サンプル ポリシーの実行

このサンプル ポリシーでは、定期的に CLI コマンドを実行し、結果を E メールで送信する機能について説明します。CRON 仕様「0-59/2 0-23/1 * * 0-7」を使用すると、このポリシーは、毎時 2 分目に実行されます。ポリシーでは、イベントについての詳細情報が収集され、CLI ライブラリを使用して、EEM 環境変数 _show_cmd で指定された、コンフィギュレーション コマンドが実行されます。CLIコマンドの結果とともに、電子メールメッセージが送信されます。

次に、このポリシーの使用方法を示すサンプル設定について説明します。ユーザーEXECモードを開始し、デバイスプロンプトで**enable**コマンドを入力します。デバイスは特権EXECモー

ドを開始します。このモードで **show event manager policy registered** コマンドを入力すると、現在登録されているポリシーがないことを確認できます。次のコマンドはどのポリシーがインストールできるかを表示する **show event manager policy available** コマンドです。**configure terminal** コマンドを入力してグローバル コンフィギュレーション モードが開始されたら、**event manager policy** コマンドを使用して tm_cli_cmd.tcl ポリシーを EEM に登録できます。グローバル コンフィギュレーション モードを終了し、**show event manager policy registered** コマンドを入力してポリシーが登録されていることを確認します。

EEM 環境変数 _cron_entry に設定されている CRON 文字列に従って、タイマー イベント ディテクタによって、定期的にこのケースのイベントがトリガーされます。EEM サーバーに通知され、サーバーでは、このイベント tm_cli_cmd.tcl を処理するために登録されたポリシーが実行されます。

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy tm_cli_cmd.tcl
 end
show event manager policy registered
```

tm_crash_reporter.tcl サンプル ポリシーの実行

このサンプル ポリシーでは、ある URL へ HTTP 形式のクラッシュ レポートを送信する機能について説明します。ポリシー登録がスタートアップ コンフィギュレーション ファイルに保存されている場合、ポリシーは、ブートの 5 秒後にトリガーされます。トリガーされると、スクリプトによって、リロード原因の検索が試行されます。リロードの原因がクラッシュの場合、ポリシーによって、関連する crashinfo ファイルが検索され、環境変数 _crash_reporter_url でユーザーによって指定された URL へ、この情報が送信されます。CGI スクリプト interface_tm.cgi は、tm_crash_reporter.tcl ポリシーから URL を受け取るために作成され、ターゲット URL マシン上のローカル データベースにクラッシュ情報が保存されます。

Perl CGI スクリプト interface_tm.cgi が作成され、HTTP サーバーが含まれているマシン上で実行するために設計され、tm_crash_reporter.tcl ポリシーが実行されているデバイスからアクセスできます。interface_tm.cgi スクリプトによって、tm_crash_reporter.tcl から渡されたデータが解析され、テキストファイルの末尾にクラッシュ情報が追加され、これによって、システムのすべてのクラッシュの履歴が作成されます。さらに、各クラッシュの詳細情報は、ユーザーが指定したクラッシュ データベース ディレクトリの 3 つのファイルに保存されます。別の Perl CGI スクリプト crash_report_display.cgi は、interface_tm.cgi スクリプトによって作成されたデータベースに保存されている情報を表示するために作成されました。crash_report_display.cgi スクリプトは、interface_tm.cgi が含まれているマシンと同じマシンに置く必要があります。そのマシンでは、Internet Explorer または Netscape などのブラウザが実行されている必要があります。crash_report_display.cgi スクリプトが実行されると、読み取り可能な形式でクラッシュ情報が表示されます。

次に、このポリシーの使用方法を示すサンプル設定について説明します。ユーザー EXEC モードを開始し、デバイス プロンプトで **enable** コマンドを入力します。デバイスは特権 EXEC モードを開始します。このモードで **show event manager policy registered** コマンドを入力すると、現在登録されているポリシーがないことを確認できます。次のコマンドはどのポリシーがイン

ストールできるかを表示する **show event manager policy available** コマンドです。**configure terminal** コマンドを入力してグローバル コンフィギュレーション モードが開始されたら、**event manager policy** コマンドを使用して tm_crash_reporter.tcl ポリシーを EEM に登録できます。グローバル コンフィギュレーション モードを終了し、**show event manager policy registered** コマンドを入力してポリシーが登録されていることを確認します。

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy tm_crash_reporter.tcl
 end
show event manager policy registered
```

tm_fsys_usage.tcl サンプル ポリシーの実行

このサンプルポリシーでは、ディスク領域の使用状況を定期的にモニターし、値が設定可能なしきい値に近くなったときに Syslog を介してレポートする機能について説明します。

次に、このポリシーの使用方法を示すサンプル設定について説明します。ユーザー EXEC モードを開始し、デバイス プロンプトで **enable** コマンドを入力します。デバイスは特権 EXEC モードを開始します。このモードで **show event manager policy registered** コマンドを入力すると、現在登録されているポリシーがないことを確認できます。次のコマンドはどのポリシーがインストールできるかを表示する **show event manager policy available** コマンドです。**configure terminal** コマンドを入力してグローバル コンフィギュレーション モードが開始されたら、**event manager policy** コマンドを使用して tm_fsys_usage.tcl ポリシーを EEM に登録できます。グローバル コンフィギュレーション モードを終了し、もう一度 **show event manager policy registered** コマンドを入力してポリシーが登録されていることを確認します。tm_fsys_usage.tcl ポリシーで使用されるオプション環境変数のいずれかを設定した場合、**show event manager environment** コマンドによって、設定された変数が表示されます。

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy tm_fsys_usage.tcl
 end
show event manager policy registered
show event manager environment
```

Tcl のサンプル スクリプトを使用したポリシーのプログラミングの例

ここでは、EEM システム ポリシーとして含まれているいくつかのサンプル ポリシーについて説明します。これらのポリシーの詳細については、EEMイベントディテクタのデモの例（2160ページ）を参照してください。

Mandatory.go_ipsec.tcl サンプル ポリシー

次のサンプルポリシーは、TestIPSecEncrypDecrypPkt テスト用です。

```
::cisco::eem::event_register_gold card all testing_type monitoring test_name Tes
tIPSecEncrypDecrypPkt consecutive_failure 6 platform_action 0 queue_priority las
t
#
#  GOLD TestIPSecEncrypDecrypPkt Test TCL script
#
#  March 2005, Hai Qiu
#
#  Copyright (c) 2005-2007 by cisco Systems, Inc.
#  All rights reserved.
#
#
# Register for TestIPSecEncrypDecrypPkt test even
# the elements for register the event
# card [all | card #]
# sub_card [all | sub_card #]
# severity_major | severity_minor | severity_normal  default :  severity_normal
# new_failure [true | false]  default: dont_care
# testing_type [bootup | ondemand | schedule | monitoring]
# test_name [ test name ]
# test_id [ test # ]
# consecutive_failure [ consecutive_failure # ]
# platform_action [action_flag]
# action_flag [ 0 | 1 | 2 ]
# queue_priority [ normal | low | high | last]   default: normal
#
# Note:
#    1: "card" element is required. If other elements are not specified,
#       treat them as dont care, or default.
#
#    2: action_flag is platform specific. It is up to platform to
#       determine what action need to be taken based on the value
#       For Cat6k platform
#       action_flag 0 : TCL script take action to reset card
#       action_flag 1 : TCL script doesn't take action to reset card
#       action_flag 2 : TCL script takes action to reset card for bootup diag
#                       when there is major error
#       action_flag 3 : TCL script doesn't take action to reset card for
#                       bootup diag when there is major error
#
#    3: "queue_priority last" would guarantee this policy will be executed last
#       if there are other EEM events in queue with queue priority other
#       than "last"
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# 1. query the information of latest triggered eem event
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
puts "GOLD EEM TCL policy for TestIPSecEncrypDecrypPkt"
#set msg [format "array=%s", array names arr_einfo]
#puts "msg $msg"
#set msg $arr_einfo(msg)
set card $arr_einfo(card)
set sub_card $arr_einfo(sub_card)
#set overall_result $arr_einfo(overall_result)
#puts "GOLD event msg recieved: $card/$sub_card overall_result= $overall_result"
# 2. execute the user-defined config commands
if [catch {cli_open} result] {
    error $result $errorInfo
```

```
} else {
    array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
    error $result $errorInfo
}
# Use "diagn action mod mod# test testname default" command
# for default platform action
if [catch {cli_exec $cli1(fd) "diagnostic action mod $card test TestIPSecEncrypD
ecrypPkt default"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
    error $result $errorInfo
}
```

ap_perf_test_base_cpu.tcl サンプル ポリシー

次のサンプル ポリシーは、EEM ポリシーの CPU パフォーマンスを測定します。

```
::cisco::eem::event_register_appl sub_system 798 type 9999
#----------------------------------
# EEM policy used for measuring the cpu performance of EEM policies.
#
# July 2005, Cisco EEM team
#
# Copyright (c) 2005, 2006 by cisco Systems, Inc.
# All rights reserved.
#------------------
###
### Input arguments:
###
### arg1 $iter                    - current iteration count
###
### The following EEM environment variables are used:
###
### _perf_iterations (mandatory) - number of iterations over which we
###                                will run our measurement.
### Example:
### event manager environment _perf_iterations 100
###
### _perf_cmd1 (optional)        - optional non interactive cli command
###                                to be executed as part of the
###                                measurement test.
### Example:
### event manager environment _perf_cmd1 enable
###
### _perf_cmd2 (optional)        - optional non interactive cli command
###                to be executed as part of the
###                                measurement test.
###                                To use _perf_cmd2, _perf_cmd1 MUST
###                be defined.
### Example:
### event manager environment _perf_cmd2 show ver
###
### _perf_cmd3 (optional)        - optional non interactive cli command
###                                to be executed as part of the
###                                measurement test.
###                                To use _perf_cmd3, _perf_cmd1 MUST
###                                be defined.
### Example:
```

```
### event manager environment _perf_cmd3 show int counters protocol status
###
### Description:
###   Iterate through _perf_iterations of this policy.
###   It is up to the user to calculate the average
###   execution time based on the system timestamps.
###   Optional commands _perf_cmd1,
###   _perf_cmd2 and _perf_cmd3 are executed if defined.
###
###   A value of 100 is a good starting point.
###
### Outputs:
###   Console output.
###
### Usage example:
###   >conf t
###   >service timestamps debug datetime msec
###   >event manager environment _perf_iterations 100
###   >event manager policy ap_perf_base_cpu.tcl
###   >event manager policy no_perf_test_init.tcl
###   >end
###   2d19h: %SYS-5-CONFIG_I: Configured from console by console
###   >event manager run no_perf_test_init.tcl
###
###   Oct 16 14:57:17.284: %SYS-5-CONFIG_I: Configured from console by console
###   >event manager run no_perf_test_init.tcl
###
###   Oct 16 19:32:02.772: %HA_EM-6-LOG:
###     eem_policy/no_perf_test_init.tcl: EEM performance test start
###   Oct 16 19:32:03.115: %HA_EM-6-LOG:
###     eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 1
###   Oct 16 19:32:03.467: %HA_EM-6-LOG:
###     eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 2
###   ...
###   Oct 16 19:32:36.936: %HA_EM-6-LOG:
###     eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 100
###   Oct 16 19:32:36.936: %HA_EM-6-LOG:
###     eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test end
###
###   The user must calculate execution time and average time of execution.
###   In this example, total time = 19:32:36.936 - 19:32:02.772 = 34.164
###   Average script execution time = 341.64 milliseconds
###
# check if all the env variables we need exist
# If any of them doesn't exist, print out an error msg and quit
if {![info exists _perf_iterations]} {
    set result \
        "Policy cannot be run: variable _perf_iterations has not been set"
    error $result $errorInfo
}
# ensure our target iteration count > 0
if {$_perf_iterations <= 0} {
    set result \
        "Policy cannot be run: variable _perf_iterations <= 0"
    error $result $errorInfo
}
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# query the event info
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
```

```
}
set iter $arr_einfo(data1)
set iter [expr $iter + 1]
# if _perf_cmd1 is defined
if {[info exists _perf_cmd1]} {
    # open the cli library
    if [catch {cli_open} result] {
        error $result $errorInfo
    } else {
        array set cli1 $result
    }
    # execute the comamnd defined in _perf_cmd1
    if [catch {cli_exec $cli1(fd) $_perf_cmd1} result] {
        error $result $errorInfo
    }
    # if _perf_cmd2 is defined
    if {[info exists _perf_cmd2]} {
        # execute the comamnd defined in _perf_cmd2
        if [catch {cli_exec $cli1(fd) $_perf_cmd2} result] {
            error $result $errorInfo
        } else {
            set cmd_output $result
        }
    }
    # if _perf_cmd3 is defined
    if {[info exists _perf_cmd3]} {
        # execute the comamnd defined in _perf_cmd3
        if [catch {cli_exec $cli1(fd) $_perf_cmd3} result] {
            error $result $errorInfo
        } else {
            set cmd_output $result
        }
    }
    # close the cli library
    if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
        error $result $errorInfo
    }
}

# log a message
set msg [format "EEM performance test iteration %s" $iter]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
# use the context info from the previous run to determine when to end
if {$iter >= $_perf_iterations} {
    #log the final messages
    action_syslog priority info msg "EEM performance test end"
    if {$_cerrno != 0} {
        set result [format \
                "component=%s; subsys err=%s; posix err=%s;\n%s" \
                $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
        error $result
    }
    exit 0
}
# cause the next iteration to run
event_publish sub_system 798 type 9999 arg1 $iter
if {$_cerrno != 0} {
    set result [format \
            "component=%s; subsys err=%s; posix err=%s;\n%s" \
```

```
            $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

tm_cli_cmd.tcl サンプル ポリシー

次に、設定可能な CRON エントリが実行されるサンプル ポリシーについて説明します。ポリシーでは、設定可能な Cisco IOS CLI コマンドが実行され、結果が電子メールで送信されます。タイムスタンプとともに出力が末尾に追加される任意のログ ファイルを定義することができます。

```
::cisco::eem::event_register_timer cron name crontimer2 cron_entry $
_cron_entry maxrun 240
#-------------------------------------------------------------------
# EEM policy that will periodically execute a cli command and email the
# results to a user.
#
# July 2005, Cisco EEM team
#
# Copyright (c) 2005 by cisco Systems, Inc.
# All rights reserved.
#-------------------------------------------------------------------
### The following EEM environment variables are used:
###
### _cron_entry (mandatory)            - A CRON specification that determines
###                                      when the policy will run. See the
###                                      IOS Embedded Event Manager
###                                      documentation for more information
###                                      on how to specify a cron entry.
### Example: _cron_entry                 0-59/1 0-23/1 * * 0-7
###
### _log_file (mandatory without _email_....)
###                                    - A filename to append the output to.
###                                      If this variable is defined, the
###                                      output is appended to the specified
###                                      file with a timestamp added.
### Example: _log_file                   bootflash:/my_file.log
###
### _email_server (mandatory without _log_file)
###                                    - A Simple Mail Transfer Protocol (SMTP)
###                                      mail server used to send e-mail.
### Example: _email_server               mailserver.example.com
###
### _email_from (mandatory without _log_file)
###                                    - The address from which e-mail is sent.
### Example: _email_from                 devtest@example.com
###
### _email_to (mandatory without _log_file)
###                                    - The address to which e-mail is sent.
### Example: _email_to                   engineering@example.com
###
### _email_cc (optional)               - The address to which the e-mail must
###                                      be copied.
### Example: _email_cc                   manager@example.com
###
### _show_cmd (mandatory)              - The CLI command to be executed when
###                                      the policy is run.
### Example: _show_cmd                   show version
###
# check if all required environment variables exist
# If any required environment variable does not exist, print out an error msg and quit
```

```
if {![info exists _log_file]} {
    if {![info exists _email_server]} {
 set result \
  "Policy cannot be run: variable _log_file or _email_server has not been set"
 error $result $errorInfo
    }
    if {![info exists _email_from]} {
 set result \
  "Policy cannot be run: variable _log_file or _email_from has not been set"
 error $result $errorInfo
    }
    if {![info exists _email_to]} {
 set result \
  "Policy cannot be run: variable _log_file ore _email_to has not been set"
 error $result $errorInfo
    }
    if {![info exists _email_cc]} {
 #_email_cc is an option, must set to empty string if not set.
 set _email_cc ""
    }
}
if {![info exists _show_cmd]} {
    set result \
        "Policy cannot be run: variable _show_cmd has not been set"
    error $result $errorInfo
}
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# query the event info and log a message
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
global timer_type timer_time_sec
set timer_type $arr_einfo(timer_type)
set timer_time_sec $arr_einfo(timer_time_sec)
# log a message
set msg [format "timer event: timer type %s, time expired %s" \
        $timer_type [clock format $timer_time_sec]]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
# 1. execute the command
if [catch {cli_open} result] {
    error $result $errorInfo
} else {
    array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
    error $result $errorInfo
}
# save exact execution time for command
set time_now [clock seconds]
# execute command
if [catch {cli_exec $cli1(fd) $_show_cmd} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
    # format output: remove trailing router prompt
```

```
    regexp {\n*(.*\n)([^\n]*)$} $result dummy cmd_output
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
    error $result $errorInfo
}

# 2. log the success of the CLI command
set msg [format "Command \"%s\" executed successfully" $_show_cmd]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
# 3. if _log_file is defined, then attach it to the file
if {[info exists _log_file]} {
    # attach output to file
    if [catch {open $_log_file a+} result] {
        error $result
    }
    set fileD $result
    # save timestamp of command execution
    #      (Format = 00:53:44 PDT Mon May 02 2005)
    set time_now [clock format $time_now -format "%T %Z %a %b %d %Y"]
    puts $fileD "%%% Timestamp = $time_now"
    puts $fileD $cmd_output
    close $fileD
}
# 4. if _email_server is defined send the email out
if {[info exists _email_server]} {
    set routername [info hostname]
    if {[string match "" $routername]} {
 error "Host name is not configured"
    }
    if [catch {smtp_subst [file join $tcl_library email_template_cmd.tm]} \
     result] {
 error $result $errorInfo
    }
    if [catch {smtp_send_email $result} result] {
 error $result $errorInfo
    }
}
```

sl_intf_down.tcl サンプル ポリシー

次に、設定可能な Syslog メッセージが記録されるときに実行されるサンプル ポリシーを示します。ポリシーでは、設定可能な CLI コマンドが実行され、結果が電子メールで送信されます。

```
::cisco::eem::event_register_syslog occurs 1 pattern $_syslog_pattern maxrun 90

#------------------------------------------------------------------
# EEM policy to monitor for a specified syslog message.
# Designed to be used for syslog interface-down messages.
# When event is triggered, the given config commands will be run.
#
# July 2005, Cisco EEM team
#
# Copyright (c) 2005 by cisco Systems, Inc.
# All rights reserved.
#------------------------------------------------------------------
```

```
### The following EEM environment variables are used:
###
### _syslog_pattern (mandatory)        - A regular expression pattern match string
###                                      that is used to compare syslog messages
###                                      to determine when policy runs
### Example: _syslog_pattern             .*UPDOWN.*FastEthernet0/0.*
###
### _email_server (mandatory)          - A Simple Mail Transfer Protocol (SMTP)
###                                      mail server used to send e-mail.
### Example: _email_server               mailserver.example.com
###
### _email_from (mandatory)            - The address from which e-mail is sent.
### Example: _email_from                 devtest@example.com
###
### _email_to (mandatory)              - The address to which e-mail is sent.
### Example: _email_to                   engineering@example.com
###
### _email_cc (optional)               - The address to which the e-mail must
###                                      be copied.
### Example: _email_cc                   manager@example.com
###
### _config_cmd1 (optional)            - The first configuration command that
###                                      is executed.
### Example: _config_cmd1                interface Ethernet1/0
###
### _config_cmd2 (optional)            - The second configuration command that
###                                      is executed.
### Example: _config_cmd2                no shutdown
###

# check if all the env variables we need exist
# If any of them doesn't exist, print out an error msg and quit
if {![info exists _email_server]} {
    set result \
        "Policy cannot be run: variable _email_server has not been set"
    error $result $errorInfo
}
if {![info exists _email_from]} {
    set result \
        "Policy cannot be run: variable _email_from has not been set"
    error $result $errorInfo
}
if {![info exists _email_to]} {
    set result \
        "Policy cannot be run: variable _email_to has not been set"
    error $result $errorInfo
}
if {![info exists _email_cc]} {
     #_email_cc is an option, must set to empty string if not set.
     set _email_cc ""
}

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

# 1. query the information of latest triggered eem event
array set arr_einfo [event_reqinfo]

if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
      $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

```
set msg $arr_einfo(msg)
set config_cmds ""

# 2. execute the user-defined config commands
if [catch {cli_open} result] {
    error $result $errorInfo
} else {
    array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
    error $result $errorInfo
}
if [catch {cli_exec $cli1(fd) "config t"} result] {
    error $result $errorInfo
}

if {[info exists _config_cmd1]} {
    if [catch {cli_exec $cli1(fd) $_config_cmd1} result] {
        error $result $errorInfo
    }
    append config_cmds $_config_cmd1
}

if {[info exists _config_cmd2]} {
    if [catch {cli_exec $cli1(fd) $_config_cmd2} result] {
        error $result $errorInfo
    }
    append config_cmds "\n"
    append config_cmds $_config_cmd2
}

if [catch {cli_exec $cli1(fd) "end"} result] {
    error $result $errorInfo
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
    error $result $errorInfo
}

after 60000
# 3. send the notification email
set routername [info hostname]
if {[string match "" $routername]} {
    error "Host name is not configured"
}

if [catch {smtp_subst [file join $tcl_library email_template_cfg.tm]} result] {
    error $result $errorInfo
}
if [catch {smtp_send_email $result} result] {
    error $result $errorInfo
}
```

次に、前述の EEM サンプル ポリシーで使用される電子メール テンプレート ファイルの使用例を示します。

```
email_template_cfg.tm
Mailservername: $_email_server
From: $_email_from
To: $_email_to
Cc: $_email_cc
Subject: From router $routername: Periodic $_show_cmd Output
$cmd_output
```

Embedded Event Manager ポリシーのデバッグの例

次に、CLI ライブラリおよび SMTP ライブラリのデバッグ例を示します。

CLI ライブラリのデバッグ

CLI ライブラリを使用すると、ユーザーは、CLI コマンドを実行し、Tcl のコマンドの出力を取得できます。Embedded Event Manager の **debug** コマンドは、このライブラリのユーザー向けに用意されています。CLI ライブラリのデバッグを有効にするコマンドは、**debug event manager tcl cli_library** です。イネーブルの場合、このコマンドによって、CLI のやり取りを処理する TTY セッションに渡され、TTY セッションから読み戻される、すべてのデータが表示されます。このデータを使用すると、ユーザーが CLI に渡すコマンドが有効になります。

デバッグ イベント マネージャ tcl cli_library コマンドの例

この例では、サンプル ポリシー sl_intf_down.tcl が使用されます。トリガーされると、sl_intf_down.tcl によって、CLI ライブラリを介して CLI にコンフィギュレーション コマンドが渡されます。次で渡されるコマンドは、**show event manager environment** です。このコマンドは、コンフィギュレーション モードでは有効ではありません。**debug** コマンドが有効ではない場合、出力は次のとおりです。

```
00:00:57:sl_intf_down.tcl[0]:config_cmds are show eve man env
00:00:57:%SYS-5-CONFIG_I:Configured from console by vty0
```

前述の出力で、ユーザーは、CLI でコマンドが正常終了したかどうかはわかりません。**debug event manager tcl cli_library** コマンドが有効である場合は、次が表示されます。

```
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : CTL : cli_open called.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson>
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN  : nelson>enable
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN  : nelson#configure terminal
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : Enter configuration commands, one
per line.  End with CNTL/Z.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson(config)#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN  : nelson(config)#show event manager
environment
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT :                 ^
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : % Invalid input detected at '^'
marker.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson(config)#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN  : nelson(config)#end
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : CTL : cli_close called.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN  : nelson#exit
01:17:07: sl_intf_down.tcl[0]: config_cmds are show event manager environment
01:17:07: %SYS-5-CONFIG_I: Configured from console by vty0
```

前述の出力には、**show event manager environment** コマンドがコンフィギュレーション モードでは無効であることが示されています。IN キーワードによって、CLI ライブラリを介して TTY へすべてのデータが渡されることが指定されます。OUT キーワードによって、CLI ライブラリを介して TTY からすべてのデータが読み戻されることが指定されます。CTL キーワードによっ

て、CLI ライブラリで使用されるヘルパー機能が指定されます。これらのヘルパー機能は、CLI への接続の設定や、接続の削除に使用されます。

SMTP ライブラリのデバッグ

SMTP ライブラリを使用すると、ユーザーは、SMTP E メール サーバーへ、E メール メッセージを送信できます。Embedded Event Manager の **debug** コマンドは、このライブラリのユーザー向けに用意されています。SMTP ライブラリのデバッグを有効にするコマンドは、**debug event manager tcl smtp_library** です。イネーブルの場合、このコマンドによって、SMTP ライブラリ ルーチンに渡され、SMTP ライブラリ ルーチンから読み戻される、すべてのデータが表示されます。このデータを使用すると、ユーザーが SMTP ライブラリに渡すコマンドが有効になります。

デバッグ イベント マネージャ tcl smtp_library コマンドの例

この例では、サンプル ポリシー tm_cli_cmd.tcl が使用されます。トリガーされると、tm_cli_cmd.tcl は CLI ライブラリを介して **show event manager policy available system** コマンドを実行します。結果は、SMTP ライブラリを介してメールでユーザーに送信されます。出力を参考に、SMTP ライブラリを使用して、関連する問題をデバッグできます。

debug event manager tcl smtp_library コマンドが有効の場合は、コンソールに次が表示されます。

```
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read  : 220 XXXX.example.com ESMTP
XXXX 1.1.0; Tue,
25 Jun 2002 14:20:39 -0700 (PDT)
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : HELO XXXX.example.com
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read  : 250 XXXX.example.com Hello
XXXX.example.com [XXXX],
pleased to meet you
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : MAIL FROM:<XX@example.com>
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read  : 250 <XX@example.com>...
Sender ok
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : RCPT TO:<XX@example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read  : 250 <XX@example.com>...
Recipient ok
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : RCPT TO:<XX@example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read  : 250 <XX@example.com>...
Recipient ok
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : DATA
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read  : 354 Enter mail, end with "."
 on a line by itself
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Date: 25 Jun 2002 14:35:00
UTC
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Message-ID:
<20020625143500.2387058729877@XXXX.example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : From: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : To: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Cc: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Subject: From router nelson:

Periodic show eve man po ava system Output
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : No. Type    Time Created
          Name
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 1   system  Fri May3
20:42:34 2002  pr_cdp_abort.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 2   system  Fri May3
```

```
20:42:54 2002  pr_iprouting_abort.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 3   system  Wed Apr3
02:16:33 2002  sl_intf_down.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 4   system  Mon Jun24
23:34:16 2002  tm_cli_cmd.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 5   system  Wed Mar27
05:53:15 2002  tm_crash_hist.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : nelson#
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write :
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : .
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read  : 250 ADE90179 Message accepted
 for delivery
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : QUIT
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read  : 221 XXXX.example.com closing
 connection
```

Tcl set コマンド操作のトレースの例

Tcl は、融通性のある言語です。Tcl の融通性の 1 つは、コマンドを上書きできることです。この例では、Tcl **set** コマンドの名前が _set に変更されます。また、テキスト「setting」が含まれるメッセージを表示し、設定しているスカラ変数を末尾に追加する、新バージョンの **set** コマンドが作成されます。この例を使用すると、設定しているスカラ変数のすべてのインスタンスをトレースできます。

```
rename set _set
proc set {var args} {
   puts [list setting $var $args]
   uplevel _set $var $args
};
```

これがポリシーに置かれると、スカラ変数が設定されるたびに、たとえば次のようなメッセージが表示されます。

```
02:17:58: sl_intf_down.tcl[0]: setting test_var 1
```

RPC イベント ディテクタのの例

```
TCL script (rpccli.tcl):
::cisco::eem::event_register_rpc
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
proc run_cli { clist } {
    set rbuf ""
    if {[llength $clist] < 1} {
    return -code ok $rbuf
    }
    if {[catch {cli_open} result]} {
        return -code error $result
    } else {
    array set cliarr $result
    }
    if {[catch {cli_exec $cliarr(fd) "enable"} result]} {
        return -code error $result
    }
    if {[catch {cli_exec $cliarr(fd) "term length 0"} result]} {
        return -code error $result
```

```
    }
    foreach cmd $clist {
    if {[catch {cli_exec $cliarr(fd) $cmd} result]} {
            return -code error $result
    }
    append rbuf $result
    }
    if {[catch {cli_close $cliarr(fd) $cliarr(tty_id)} result]} {
        puts "WARNING: $result"
    }
    return -code ok $rbuf
}
proc run_cli_interactive { clist } {
    set rbuf ""
    if {[llength $clist] < 1} {
    return -code ok $rbuf
    }
    if {[catch {cli_open} result]} {
        return -code error $result
    } else {
    array set cliarr $result
    }
    if {[catch {cli_exec $cliarr(fd) "enable"} result]} {
        return -code error $result
    }
    if {[catch {cli_exec $cliarr(fd) "term length 0"} result]} {
        return -code error $result
    }
    foreach cmd $clist {
        array set sendexp $cmd
    if {[catch {cli_write $cliarr(fd) $sendexp(send)} result]} {
            return -code error $result
    }
    foreach response $sendexp(responses) {
        array set resp $response
        if {[catch {cli_read_pattern $cliarr(fd) $resp(expect)} result]} {
                return -code error $result
        }
        if {[catch {cli_write $cliarr(fd) $resp(reply)} result]} {
                return -code error $result
        }
    }
    if {[catch {cli_read $cliarr(fd)} result]} {
            return -code error $result
    }
    append rbuf $result
    }
    if {[catch {cli_close $cliarr(fd) $cliarr(tty_id)} result]} {
        puts "WARNING: $result"
    }
    return -code ok $rbuf
}
array set arr_einfo [event_reqinfo]
set args $arr_einfo(argc)
set cmds [list]
for { set i 0 } { $i < $args } { incr i } {
    set arg "arg${i}"
    # Split each argument on the '^' character.  The first element is
    # the command, and each subsequent element is a prompt followed by
    # a response to that prompt.
    set cmdlist [split $arr_einfo($arg) "^"]
    set cmdarr(send) [lindex $cmdlist 0]
    set cmdarr(responses) [list]
    if { [expr ([llength $cmdlist] - 1) % 2] != 0 } {
```

```
    return -code 88
    }
    set cmdarr(responses) [list]
    for { set j 1 } { $j < [llength $cmdlist] } { incr j 2 } {
    set resps(expect) [lindex $cmdlist $j]
    set resps(reply) [lindex $cmdlist [expr $j + 1]]
    lappend cmdarr(responses) [array get resps]
    }
    lappend cmds [array get cmdarr]
}
set rc [catch {run_cli_interactive $cmds} output]
if { $rc != 0 } {
    error $output $errorInfo
    return -code 88
}
puts $output
```

その他の参考資料

次の項では、Tcl を使用した Embedded Event Manager ポリシー記述についての関連資料を示します。

関連資料

関連項目	マニュアル タイトル
EEM コマンド：コマンド構文の詳細、デフォルト、コマンド モード、コマンド履歴、使用上の注意事項、および例	Cisco IOS Embedded Event Manager のコマンド リファレンス
Embedded Event Manager 概要	「Embedded Event Manager の概要」の章
CLI を使用して Embedded Event Manager ポリシーを記述する	「Writing Embedded Event Manager Policies Using the Cisco IOS CLI」の章
Embedded Resource Manager	「Embedded Resource Manager」の章

MIB

MIB	MIB のリンク
CISCO-EMBEDDED-EVENT-MGR-MIB	選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	--

シスコのテクニカル サポート

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

Cisco IOS CLI を使用した EEM 4.0 ポリシーの記述の機能情報

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

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

表 *200 : Cisco IOS CLI* を使用した *EEM 4.0* ポリシーの記述の機能情報

機能名	リリース	機能情報
Embedded Event Manager 4.0	15.2(5)E1	この機能は、c2960cx プラットフォームにのみ導入され、サポートされています。

第 90 章

署名済み Tcl スクリプト

署名付き TCL スクリプト機能を使用すると、デジタル署名を生成する証明書を作成し、そのデジタル署名を使用してツールコマンド言語（TCL）スクリプトに署名することが可能になります。この機能は、既存のスクリプトおよび証明書でも動作します。デジタル署名の認証が確認されてから、Tcl インタープリタへの信頼できるアクセスでスクリプトが実行されます。スクリプトにデジタル署名がない場合、そのスクリプトは信頼できないスクリプト用の限定モードで実行されるか、まったく実行されません。

署名済み Tcl スクリプトに関する前提条件

この機能が動作するには、Cisco Public Key Infrastructure（PKI）設定のトラストポイント コマンドを有効にする必要があります。

署名付き TCL スクリプトの制約事項

この機能が動作するには、次を実行している必要があります。

- Cisco IOS 暗号イメージ
- OpenSSL Version 0.9.7a 以降
- Expect

署名済み Tcl スクリプトについて

署名済み Tcl スクリプト機能は Tcl スクリプトにセキュリティを導入します。この機能を使用すると、デジタル署名を生成する証明書を作成し、そのデジタル署名を使用して Tcl スクリプトに署名することが可能になります。この証明書は、Tcl スクリプトを実行する前にそれらを検査します。スクリプトに Cisco 発行のデジタル証明書が含まれているかどうかを確認します。さらに、第三者がデジタル署名でスクリプトに署名することもできます。独自に社内で開発した TCL スクリプトに署名したい場合や、サード パーティ製が開発したスクリプトを使用したい場合もあります。スクリプトに正しいデジタル署名が含まれている場合は本物であると見なされ、Tcl インタープリタにフル アクセスで実行されます。スクリプトにデジタル署名がない場合、そのスクリプトはセーフ Tcl モードという限定されたモードで実行されるか、またはまったく実行されません。

署名付き Tcl スクリプトを作成し、使用するには、次の概念を理解する必要があります。

Cisco PKI

Cisco PKI を使用すると、IP セキュリティ（IPSec）、セキュア シェル（SSH）、セキュア ソケット レイヤ（SSL）などのセキュリティ プロトコルをサポートする証明書管理を実現できます。PKI は以下のエンティティで構成されています。

- セキュアなネットワークで通信する複数のピア
- 証明書を発行および維持する認証局（CA）を最低 1 つ
- デジタル証明書（証明書の有効期間、ピアの ID 情報、セキュアな通信に使用する暗号キー、CA 発行のシグニチャなどで構成）
- 登録要求を処理し CA の負荷を軽減する登録局（RA）（任意）
- 証明書失効リスト（CRL）を配信するメカニズム（Lightweight Directory Access Protocol（LDAP）、HTTP など）

PKI を使用すると、セキュアなデータ ネットワークで暗号化情報と ID 情報を配信、管理、失効するためのスケーラブルでセキュアなメカニズムを実現できます。セキュアな通信に関係するルーティング デバイスはすべて、あるプロセスを経て PKI に登録されます。そのプロセスでは、ルーティング デバイスが Rivest, Shamir, and Adelman（RSA）キー ペア（秘密キーが 1 つ、公開キーが 1 つ）を生成し、信頼されているルーティング デバイス（CA またはトラストポイントともいいます）でキーの ID を確認します。

各ルーティング デバイスが PKI に登録されると、PKI のすべてのピア（エンド ホストともいいます）は、CA が発行したデジタル証明書を付与されます。セキュアな通信セッションをネゴシエーションする必要があるときは、ピアはデジタル証明書を交換します。ピアは証明書内の情報を基に他のピアの ID を確認し、証明書内の公開キーを使って、暗号化されたセッションを確立します。

RSA キーペア

RSA キー ペアは、公開キーと秘密キーで構成されます。PKI を設定する場合、証明書登録要求に公開キーを含める必要があります。証明書が付与された後、ピアが公開キーを使用して、デバイスに送信されるデータを暗号化できるように、公開キーが証明書に組み込まれます。秘密キーはデバイスに保持され、ピアによって送信されたデータの復号化と、ピアとネゴシエーションするときの、トランザクションのデジタル署名に使用されます。

RSA キーペアには、キーのモジュラス値が含まれています。モジュラス値に応じて、RSA キーのサイズが決まります。モジュラス値が大きいほど、RSA キーの安全性が高まります。ただし、モジュラス値が大きくなると、キーの生成にかかる時間が長くなり、キーのサイズが大きくなると暗号化処理および復号化処理にかかる時間が長くなります。

証明書およびトラストポイント

認証局（CA。トラストポイントともいいます）は、証明書要求を管理し、参加ネットワークデバイスに証明書を発行します。証明書要求の管理や証明書発行などのサービスにより、参加デバイスを一元的に管理します。またこれらのサービスによって受信者は、明示的に信頼してアイデンティティを確認し、デジタル証明書を作成できます。PKI の動作を開始する前に、CA は独自の公開キー ペアを生成し、自己署名 CA 証明書を作成します。その後、CA は、証明書要求に署名し、PKI に対してピア登録を開始できます。

CA は、サードパーティの CA ベンダーが提供する CA を使用するか、内部の CA、つまり Cisco 証明書サーバーを使用します。

署名済み Tcl スクリプトの設定方法

キー ペアの生成

キーペアは、秘密キーと公開キーで構成されます。秘密キーは公開されず、作成者のみがアクセス可能にすることを意図しています。公開キーは秘密キーから生成され、公開されることを前提としています。

キーペアを生成するには、**openssl genrsa** コマンドを使用した後、**openssl rsa** コマンドを使用します。

手順の概要

1. **openssl genrsa -out** *private-key-file bit-length*
2. **ls -l**
3. **openssl rsa -in** *private-key-file* **-pubout -out** *public-key-file*
4. **ls -l**

手順の詳細

ステップ 1 **openssl genrsa -out** *private-key-file bit-length*

このコマンドは、*bit-length* ビット長の秘密キーを生成し、そのキーを *private-key-file* ファイルに書き込みます。

```
Host% openssl genrsa -out privkey.pem 2048
```

例：

```
Generating RSA private key, 2048 bit long modulus
.........+++
.....................................................................+++
e is 65537 (0x10001)
```

ステップ 2 **ls -l**

このコマンドは、現在のディレクトリ内の各ファイルに対する詳細な情報（権限、所有者、サイズ、最終変更日時を含む）を表示します。

例：

```
Host% ls -l

total 8
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
```

privkey.pem ファイルには、**openssl genrsa** コマンドを使用して生成した秘密キーが含まれています。

ステップ 3 **openssl rsa -in** *private-key-file* **-pubout -out** *public-key-file*

このコマンドは、*private-key-file* ファイル内の指定された秘密キーに基づいて公開キーを作成し、その公開キーを *public-key-file* ファイルに書き込みます。

例：

```
Host% openssl rsa -in privkey.pem -pubout -out pubkey.pem

writing RSA key
```

ステップ 4 **ls -l**

このコマンドは、現在のディレクトリ内の各ファイルに対する詳細な情報（権限、所有者、サイズ、最終変更日時を含む）を表示します。

例：

```
Host% ls -l

total 16
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
```

pubkey.pem ファイルには、**openssl rsa** コマンドを使用して生成された公開キーが含まれます。

証明書の生成

証明書を生成するには、次のタスクを実行します。X.509 証明書を生成するには、**openssl req** コマンドを使用します。

手順の概要

1. **openssl req -new -x509 -key** *private-key-file* **-out** *certificate-file* **-days** *expiration-days*
2. **ls -l**

手順の詳細

ステップ 1 **openssl req -new -x509 -key** *private-key-file* **-out** *certificate-file* **-days** *expiration-days*

このコマンドは、*private-key-file* ファイルに保存された秘密キーにフルアクセスできる X.509 証明書を作成し、*certificate-file* ファイルに証明書を保存します。証明書は *expiration-days* 日内に期限が切れるように設定されます。

コマンドを実行するには、プロンプトが表示された時点で次の識別名（DN）情報を入力します。

- 国名
- 州、行政区分（都道府県）名
- 組織名
- 組織部署名
- 共通名
- メール アドレス

各プロンプトの角括弧で囲まれたテキストは、Enter を押す前に値を入力しなかった場合に使用されるデフォルト値を示します。

次に、privkey.pem ファイル内の秘密キーに対するフル アクセスを持つ X.509 証明書を生成する方法の例を示します。証明書は cert.pem ファイルに書き込まれ、生成日の 1095 日後に期限切れになります。

例：

```
Host% openssl req -new -x509 -key privkey.pem -out cert.pem -days 1095

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value, If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:US
```

```
State or Province Name (full name) [Berkshire]:California

Locality Name (eg, city) [Newbury]:San Jose

Organization Name (eg, company) [My Company Ltd]:Cisco Systems, Inc.

Organizational Unit Name (eg, section) []:DEPT_ACCT

Common Name (eg, your name or your server's hostname) []:Jane

Email Address []:janedoe@company.com
```

ステップ 2 **ls -l**

このコマンドは、現在のディレクトリ内の各ファイルに対する詳細な情報（権限、所有者、サイズ、最終変更日時を含む）を表示します。

例：

```
Host% ls -l

total 24
-rw-r--r--   1 janedoe eng12       1659 Jun 12 15:01 cert.pem
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
```

cert.pem ファイルには、**openssl req** コマンドを使用して作成された X.509 証明書が含まれています。

Tcl スクリプトの署名

Tcl スクリプトに署名するには、次のタスクを実行します。TCL ファイル、および OpenSSL ドキュメントの出力に、pkcs7（PKCS#7）フォーマットで署名する必要があります。

Tcl ファイルに署名するには、**openssl smime** コマンドと **-sign** キーワードを使用します。

手順の概要

1. **openssl smime -sign -in** *tcl-file* **-out** *signed-tcl-file* **-signer** *certificate-file* **-inkey** *private-key-file* **-outform DER -binary**
2. **ls -l**

手順の詳細

ステップ 1 **openssl smime -sign -in** *tcl-file* **-out** *signed-tcl-file* **-signer** *certificate-file* **-inkey** *private-key-file* **-outform DER -binary**

このコマンドは、*certificate-file* に保存されている証明書と、*private-key-file* に保存されている秘密キーを使用して Tcl ファイル名 *tcl-file* に署名し、署名済みの Tcl ファイルを *signed-tcl-file* ファイルに DER PKCS#7 形式で書き込みます。

例：

```
Host% openssl smime -sign -in hello -out hello.pk7 -signer cert.pem -inkey privkey.pem -outform DER
 -binary
```

ステップ 2 **ls -l**

このコマンドは、現在のディレクトリ内の各ファイルに対する詳細な情報（権限、所有者、サイズ、最終変更日時を含む）を表示します。

例：

```
Host% ls -l

total 40
-rw-r--r--   1 janedoe eng12       1659 Jun 12 15:01 cert.pem
-rw-r--r--   1 janedoe eng12        115 Jun 13 10:16 hello
-rw-r--r--   1 janedoe eng12       1876 Jun 13 10:16 hello.pk7
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
```

hello.pk7 ファイルには、hello という名前の未署名の TCL ファイルから **openssl smime** コマンドと cert.pem ファイル内の X.509 証明書を使用して作成された、署名済み Tcl ファイルが含まれています。

署名の確認

署名がデータと一致していることを確認するには、**openssl smime** コマンドと **-verify** キーワードを使用して次のタスクを実行します。Tcl の元の内容を入力ファイルに提供する必要があります。これは、ファイルに元の内容が含まれていないためです。

手順の概要

1. **openssl smime -verify -in** *signed-tcl-file* **-CAfile** *certificate-file* **-inform DER -content** *tcl-file*
2. **ls -l**

手順の詳細

ステップ 1 **openssl smime -verify -in** *signed-tcl-file* **-CAfile** *certificate-file* **-inform DER -content** *tcl-file*

このコマンドは、*certificate-file* 内の信頼認証局（CA）証明書を使用して DER PKCS#7 形式で *signed-tcl-file* に保存されている署名付き Tcl ファイルを確認した後、デタッチされた内容を *tcl-file* ファイルに書き込みます。

次に、入力ファイルの hello.pk7 を使用して署名を確認する例を示します。

例：

```
Host% openssl smime -verify -in hello.pk7 -CAfile cert.pem -inform DER -content hello

puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
```

```
puts "tcl_interactive = $tcl_interactive"
Verification successful
```

（注） SSL コマンドページでは、**-in** *filename* が暗号化または署名される入力メッセージか、復号または確認される MIME メッセージとして説明されています。詳細については、http://www.openssl.org/ を参照してください。

ステップ 2 **ls -l**

このコマンドは、現在のディレクトリ内の各ファイルに対する詳細な情報（権限、所有者、サイズ、最終変更日時を含む）を表示します。

例：

```
Host% ls -l

total 40
-rw-r--r--   1 janedoe eng12       1659 Jun 13 10:18 cert.pem
-rw-r--r--   1 janedoe eng12        115 Jun 13 10:17 hello
-rw-r--r--   1 janedoe eng12       1876 Jun 13 10:16 hello.pk7
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
```

hello ファイルには、**openssl smime** コマンドを **-verify** キーワードで実行して、署名付き Tcl ファイル hello.pk7 からデタッチされた内容が含まれています。確認に成功した場合、署名者の証明書が cert.pem ファイルの X.509 証明書に書き込まれます。

シグニチャの非バイナリデータへの変換

バイナリから非バイナリ データにシグニチャを変換するには、次のタスクを実行します。

手順の概要

1. **xxd -ps** *signed-tcl-file* > *nonbinary-signature-file*
2. **#Cisco Tcl Signature V1.0** を最初の行に表示するスクリプトを作成し、コメント文字（#）を入力ファイルの各行の先頭に挿入し、入力ファイルの名前にテキスト文字列「_sig」を追加して形成された名前のファイルに各行を書き込みます。
3. 非バイナリ シグニチャ ファイルを含むファイルの名前（*nonbinary-signature-file*）を入力引数として指定して、スクリプトを実行します。
4. **ls -l**
5. **cat** *signed-tcl-file* *commented-nonbinary-signature-file* > *signed-tcl-script*
6. **cat** *signed-tcl-script*

手順の詳細

ステップ 1 **xxd -ps** *signed-tcl-file* > *nonbinary-signature-file*

このコマンドは、*signed-tcl-file* のシグニチャをバイナリから非バイナリのデータに変換して *nonbinary-signature-file* ファイルに 16 進ダンプとして保存します。

例：

```
Host% xxd -ps hello.pk7 > hello.hex
```

ステップ 2 **#Cisco Tcl Signature V1.0** を最初の行に表示するスクリプトを作成し、コメント文字（#）を入力ファイルの各行の先頭に挿入し、入力ファイルの名前にテキスト文字列「_sig」を追加して形成された名前のファイルに各行を書き込みます。

次に、**cat** コマンドを使用して、my_append という名前のスクリプトファイルの内容を表示する例を示します。

例：

```
Host% cat my_append

#!/usr/bin/env expect
set my_first {#Cisco Tcl Signature V1.0}
set newline {}
set my_file [lindex $argv 0]
set my_new_file ${my_file}_sig
set my_new_handle [open $my_new_file w]
set my_handle [open $my_file r]
puts $my_new_handle $newline
puts $my_new_handle $my_first
foreach line [split [read $my_handle] "\n"]  {
   set new_line {#}
   append new_line $line
   puts $my_new_handle $new_line
}

close $my_new_handle
close $my_handle
```

ステップ 3 非バイナリ シグニチャ ファイルを含むファイルの名前（*nonbinary-signature-file*）を入力引数として指定して、スクリプトを実行します。

この例では、my_append スクリプトが、入力として指定された非バイナリ シグニチャ ファイル hello.hex を使用して実行されています。出力ファイルには、hello.hex_sig という名前が付けられます。

例：

```
Host% my_append hello.hex
```

ステップ 4 **ls -l**

このコマンドは、現在のディレクトリ内の各ファイルに対する詳細な情報（権限、所有者、サイズ、最終変更日時を含む）を表示します。

例：

```
Host% ls -l

total 80
-rw-r--r--   1 janedoe eng12       1659 Jun 13 10:18 cert.pem
-rw-r--r--   1 janedoe eng12        115 Jun 13 10:17 hello
```

```
-rw-r--r--   1 janedoe eng12       3815 Jun 13 10:20 hello.hex
-rw-r--r--   1 janedoe eng12       3907 Jun 13 10:22 hello.hex_sig
-rw-r--r--   1 janedoe eng12       1876 Jun 13 10:16 hello.pk7
-rwxr--r--   1 janedoe eng12        444 Jun 13 10:22 my_append
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
```

hello.hex ファイルには、署名済み Tcl ファイル hello.pk7 のバイナリ シグニチャから変換された非バイナリ データ（16 進数のダンプとして格納）が含まれています。my_append ファイルには、入力ファイルの各行の先頭にコメント文字を挿入するスクリプトが含まれています。この hello.hex_sig ファイルは、非バイナリ シグニチャ ファイルで my_append スクリプトを実行して作成されたファイルです。

ステップ 5 **cat** *signed-tcl-file commented-nonbinary-signature-file* > *signed-tcl-script*

このコマンドは非バイナリ シグニチャ ファイル（*commented-nonbinary-signature-file*）の内容を、DER PKCS#7 形式で保存された署名済みの Tcl ファイル（*signed-tcl-file* ファイル）に追加します。連結された出力が *signed-tcl-script* ファイルに書き込まれます。

例：

```
Host% cat hello hello.hex_sig > hello.tcl
```

ステップ 6 **cat** *signed-tcl-script*

このコマンドは、署名済み Tcl ファイルと非バイナリ シグニチャ ファイルから分離された内容を連結した *signed-tcl-script* ファイルの内容を表示します。

例：

```
Host% cat hello.tcl

puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
#Cisco Tcl Signature V1.0
#3082075006092a864886f70d010702a08207413082073d020101310b3009
#06052b0e03021a0500300b06092a864886f70d010701a08204a13082049d
#30820385a003020102020100300d06092a864886f70d0101040500308195
#310b300906035504061302555331133011060355040813 0a43616c69666f
#726e69613111300f060355040713 0853616e204a6f7365311c301a060355
#040a1313436973636f2053797374656d732c20496e632e310e300c060355
#040b13054e53535447310d300b060355040313044a6f686e3121301f0609
#2a864886f70d01090116126a6c6175746d616e40636973636f2e636f6d30
#1e170d3037303631323232303134335a170d3130303631313232303133
#5a308195310b3009060355040613025553311330110603550408130a4361
#6c69666f726e69613111300f060355040713 0853616e204a6f7365311c30
#1a060355040a1313436973636f2053797374656d732c20496e632e310e30
#0c060355040b13054e53535447310d300b060355040313044a6f686e3121
#301f06092a864886f70d01090116126a6c6175746d616e40636973636f2e
#636f6d30820122300d06092a864886f70d01010105000382010f00308201
#0a0282010100a751eb5ec1f3009738c88a55987c07b759c36f3386342283
#67ea20a89d9483ae85e0c63eeded8ab3eb7a08006689f09136f172183665
#c971099ba54e77ab47706069bbefaaab8c50184396350e4cc870c4c3f477
#88c55c52e2cf411f05b59f0eaec0678ff5cc238fdce2263a9fc6b6c244b8
#ffaead865c19c3d3172674a13b24c8f2c01dd8b1bd491c13e84e29171b85
#f28155d81ac8c69bb25ca23c2921d85fbf745c106e7aff93c72316cbc654
#4a34ea88174a8ba7777fa60662974e1fbac85a0f0aeac925dba6e5e850b8
#7caffce2fe8bb04b61b62f532b5893c081522d538005df81670b931b0ad0
```

```
#e1e76ae648f598a9442d5d0976e67c8d55889299147d0203010001a381f5
#3081f2301d0603551d0e04160414bc34132be952ff8b9e1af3b93140a255
#e54a667c3081c20603551d230481ba3081b78014bc34132be952ff8b9e1a
#f3b93140a255e54a667ca1819ba48198308195310b300906035504061302
#5553311330110603550408130a43616c69666f726e69613111300f060355
#0407130853616e204a6f7365311c301a060355040a1313436973636f2053
#797374656d732c20496e632e310e300c060355040b13054e53535447310d
#300b060355040313044a6f686e3121301f06092a864886f70d0109011612
#6a6c6175746d616e40636973636f2e636f6d820100300c0603551d130405
#30030101ff300d06092a864886f70d01010405000382010100c83c1b074
#6720929c9514af6d5df96f0a95639f047c40a607c83d8362507c58fa7f84
#aa699ec5e5bef61b2308297a0662c653ff446acfbb6f5cb2dd162d939338
#a5e4d78a5c45021e5d4dbabb8784efbf50cab0f5125d164487b31f5cf933
#a9f68f82cd111cbab1739d7f372ec460a7946882874b0a0f22dd53acbd62
#a944a15e52e54a24341b3b8a820f23a5bc7ea7b2278bb56838b8a4051926
#af9c167274ff8449003a4e012bcf4f4b3e280f85209249a390d14df47435
#35efabce720ea3d56803a84a2163db4478ae19d7d987ef6971c8312e280a
#aac0217d4fe620c6582a48faa8ea5e3726a99012e1d55f8d61b066381f77
#4158d144a43fb536c77d6a3182027730820273020101301308195310b
#300906035504061302555331133011060355040813a43616c69666f726e
#69613111300f060355040713085361e204a6f7365311c301a060355040a
#1313436973636f2053797374656d732c20496e632e310e300c060355040b
#13054e53535447310d300b060355040313044a6f686e3121301f06092a86
#4886f70d01090116126a6c6175746d616e40636973636f2e636f6d020100
#300906052b0e03021a0500a081b1301806092a864886f70d010903310b06
#092a864886f70d010701301c06092a864886f70d01090531 0f170d303730
#363133331373136343385a302306092a864886f70d0109043116041 4372cb3
#72dc607990577fd0426104a42ee4158d2b305206092a864886f70d01090f
#31453043300a06082a864886f70d0307300e06082a864886f70d03020202
#0080300d06082a864886f70d030202014030070 6052b0e030207300d0608
#2a864886f70d0302020128300d06092a864886f70d010101050004820100
#72db6898742f449b26d3ac18f43a1e7178834fb05ad13951bf042e127eea
#944b72b96f3b8ecf7eb52f3d0e383bf63651750223efe69eae04287c9dae
#b1f31209444108b31d34e46654c6c3cc10b5baba887825c224ec6f376d49
#00ff7ab2d9f88402dab9a2c2ab6aa3ecceeaf5a594bdc7d3a822c55e7daa
#aa0c2b067e06967f22a20e406fe21d9013ecc6bd9cd6d402c2749f8bea61
#9f8f87acfbc9e10d6ce91502e34629adca6ee855419afafe6a8233333e14
#ad4c107901d1f2bca4d7ffaadddbc54192a25da662f8b8509782c76977b8
#94879453fbb00486ccc55f88db50fcc149bae066916b350089cde51a6483
#2ec14019611720fc5bbe2400f24225fc
```

証明書を使用したデバイスの設定

証明書を使用してデバイスを設定するには、次のタスクを実行します。

始める前に

すでに、Cisco IOS 暗号化イメージが用意されている必要があります。用意されていない場合は、証明書を設定できません。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *name*
4. **enrollment terminal**

5. **exit**
6. **crypto pki authenticate** *name*
7. プロンプトで、ベースが暗号化された CA 証明書を入力します。
8. **scripting tcl secure-mode**
9. **scripting tcl trustpoint name** *name*
10. **scripting tcl trustpoint untrusted** {**execute** | **safe-execute** | **terminate**}
11. **exit**
12. **tclsafe**

手順の詳細

ステップ 1 enable

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 configure terminal

グローバル コンフィギュレーション モードを開始します。

例：

```
Device# configure terminal
```

ステップ 3 crypto pki trustpoint *name*

デバイスが認証局（CA）*mytrust* を使用して、CA トラストポイント コンフィギュレーション モードを開始することを宣言します。

例：

```
Device(config)# crypto pki trustpoint mytrust
```

ステップ 4 enrollment terminal

カットアンドペーストによる手動での証明書登録を指定します。このコマンドが有効になると、デバイスはコンソール端末に証明書要求を表示します。これにより、このターミナルに発行済みの証明証が入力できるようになります。

例：

```
Device(ca-trustpoint)# enrollment terminal
```

ステップ 5 exit

CA トラストポイント コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。

例：

```
Device(ca-trustpoint)# exit
```

ステップ 6 **crypto pki authenticate** *name*

CA 証明書を取得して、認証します。証明書フィンガープリントをチェックするよう求められた場合、証明書フィンガープリントをチェックします。

（注） CA では、証明書が自己署名されるため、このコマンドを実行するときは、CA 管理者に問い合わせて、CA の公開キーを手動で認証する必要があります。

例：

```
Device(config)# crypto pki authenticate mytrust
```

ステップ 7 プロンプトで、ベースが暗号化された CA 証明書を入力します。

例：

```
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIEuDCCA6CgAwIBAgIBADANBgkqhkiG9w0BAQQFADCBnjELMAkGA1UEBhMCVVMx
EzARBgNVBAgTCkNhbGlmb3JuaWExETAPBgNVBAcTCFNhbiBKb3NlMRwwGgYDVQQK
ExNDaXNjbyBTeXN0ZW1zLCBJbmMuMQ4wDAYDVQQLEwVOU1NURzEWMBQGA1UEAxMN
Sm9obiBMYXV0bWFubjEhMB8GCSqGSIb3DQEJARYSamxhdXRtYW5AY2lzY28uY29t
MB4XDTA2MTExNzE3NTgwMVoXDTA5MTExNjE3NTgwMVowgZ4xCzAJBgNVBAYTAlVT
MRMwEQYDVQQIEwpDYWxpZm9ybmlhMREwDwYDVQQHEwhTYW4gSm9zZTEcMBoGA1UE
ChMTQ2lzY28gU3lzdGVtcywgSW5jLjEOMAwGA1UECxMFTlNTVEcxFjAUBgNVBAMT
DUpvaG4gTGF1dG1hbm4xITAfBgkqhkiG9w0BCQEWEmpsYXV0bWFuQGNpc2NvLmNv
bTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALxtqTMCirMb+CdyWLuH
oWAM8CEJDwQggL7MWBhoi3TSMd/ww2XBB9biBtdlH6jHsjCiOwAR5OorakwfPyf7
mvRJ2PqJALs+Vn93VBKIG6rZUl4+wdOx686BVddIZvEJQPbROiYTzfazWV70aLMV
bd7/B7vF1SG1YK9y1tX9p9nZyZ0x47OAXetwOaGinvlG7VNuTXaASBLUjCRZsIlz
SBrXXedBzZ6+BuoWm1FK45EYSlag5Rt9RGXXMBqzx91iyhrJ3zDDmkExa45yKJET
mAgDVMcpeteJtif47UDZJK30g4MbMyx/c8WGhmJ54qRL9BZEPmDxMQkNP1018MAl
Q8sCAwEAAaOB/jCB+zAdBgNVHQ4EFgQU9/ToDvbMR3JfJ4xEa4X47oNFq5kwgcsG
A1UdIwSBwzCBwIAU9/ToDvbMR3JfJ4xEa4X47oNFq5mhgaSkgaEwgZ4xCzAJBgNV
BAYTAlVTMRMwEQYDVQQIEwpDYWxpZm9ybmlhMREwDwYDVQQHEwhTYW4gSm9zZTEc
MBoGA1UEChMTQ2lzY28gU3lzdGVtcywgSW5jLjEOMAwGA1UECxMFTlNTVEcxFjAU
BgNVBAMTDUpvaG4gTGF1dG1hbm4xITAfBgkqhkiG9w0BCQEWEmpsYXV0bWFuQGNp
c2NvLmNvbYIBADAMBgNVHRMEBTADAQH/MA0GCSqGSIb3DQEBBAUAA4IBAQBtEs/4
MQeN9pT+XPCPg2ObQU8y2AadI+I34YK+fDHsFOh68hZhpszTN2VpNEvkFXpADhgr
7DkNGtwTCla481v70iNFViQVL+inNrZwWMxoTnUNCK7Hc5kHkXt6cj0mvsefVUzx
Xl70mauhESRVlmYWrJxSsrEILerZYsuv5HbFdand+/rErmP2HVyfdntLnKdSzmXJ
5lwE/Et2QtYNGor0OBlLesowfslR3LhHi4wn+5is7mALgNw/NuTiUr1zH18OeB4m
wcpBIJsLaJu6ZUJQl7IqdswSa3fHd5qq0/k8P9z0YAYrf3+MFQr4ibvsYvHlO087
o2Js1gW4qz34pqNh
Certificate has the following attributes:
       Fingerprint MD5: 1E327DBB 330936EB 2FB8EACB 4FD1133E
      Fingerprint SHA1: EE7FF9F4 05148842 B9D50FAC D76FDC9C E0703246
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
```

ステップ 8 **scripting tcl secure-mode**

インタラクティブ Tcl スクリプトのシグニチャ確認を有効にします。

```
Device(config)# scripting tcl secure-mode
```

ステップ 9 **scripting tcl trustpoint name** *name*

設定済みの既存のトラストポイント名と証明書を関連付け、Tcl スクリプトを確認します。

```
Device(config)# scripting tcl trustpoint name mytrust
```

ステップ 10 **scripting tcl trustpoint untrusted** {**execute** | **safe-execute** | **terminate**}

（任意）シグニチャ確認に失敗したか、信頼できないモードであるかにかかわらず、**execute**、**safe-execute** または **terminate** の 3 つのキーワードのいずれかを使用してインタラクティブ Tcl スクリプトを実行できます。

- **execute** ：シグニチャの確認に失敗しても、Tcl スクリプトを実行します。 **execute** キーワードを設定すると、シグニチャの確認は一切実行されません。

（注） シグニチャの確認が実行されないため、通常、このキーワードの使用は推奨されません。

execut キーワードは、内部テスト用に提供されており、これにより柔軟性が向上します。たとえば、証明書の期限が切れていても、他の設定が有効であり、既存の設定で作業したい場合は、execute キーワードを使用して、期限の切れた証明書で対処することができます。

- **safe-execute** ：スクリプトをセーフ モードで実行できます。tclsafe コマンドを使用し、インタラクティブ Tcl シェル セーフ モードを開始すると、使用可能なセーフ モード Tcl コマンドを確認できます。この限定されたセーフ モードで何が使用できるかをより深く理解するには、tclsafe Exec コマンドを使用してオプションを確認します。
- **terminate** ：すべてのスクリプトの実行を停止し、デフォルトの動作に戻します。デフォルト ポリシーは終端します。最後のトラストポイント名が削除されると、信頼できないアクションも削除されます。信用できないアクションは、TCL 用に最低でも 1 つのトラストポイント名が設定されていない場合は開始されません。

次に、シグニチャ確認が失敗した場合に、**safe-execute** キーワードを使用して Tcl スクリプトをセーフモードで実行する例を示します。

```
Device(config)# scripting tcl trustpoint untrusted safe-execute
```

ステップ 11 **exit**

グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

```
Device(config)# exit
```

ステップ 12 **tclsafe**

（任意）インタラクティブ Tcl シェルの信頼できないセーフ モードを有効にします。これにより、Cisco コマンドライン インターフェイスから信頼できないセーフ モードで手動により Tcl コマンドを実行できるようになります。

```
Device# tclsafe
```

例：

トラストポイントの確認

デバイス内に設定されているトラストポイントを表示するには、**show crypto pki trustpoints** コマンドを使用します。

手順の概要

1. **enable**
2. **show crypto pki trustpoints**

手順の詳細

ステップ 1 enable

このコマンドでは、特権 EXEC モードをイネーブルにします。

例：

```
Device> enable
```

ステップ 2 show crypto pki trustpoints

このコマンドは、デバイスに設定されているトラストポイントを表示します。

例：

```
Device# show
 crypto pki trustpoints

Trustpoint mytrust:
    Subject Name:
    ea=janedoe@cisco.com
    cn=Jane
    ou=DEPT_ACCT
    o=Cisco
    l=San Jose
    st=California
    c=US
          Serial Number: 00
    Certificate configured.
```

署名済み Tcl スクリプトの確認

署名済み Tcl スクリプトが正しく実行していることを確認するには、**debug crypto pki transactions** コマンドと **tclsh** コマンドを使用します。

手順の概要

1. **enable**
2. **debug crypto pki transactions**
3. **tclsh** *flash:signed-tcl-file*

手順の詳細

ステップ 1 enable

このコマンドでは、特権 EXEC モードをイネーブルにします。

例：

```
Device> enable
```

ステップ 2 debug crypto pki transactions

このコマンドは、CA とデバイス間のやり取りのトレース（メッセージ タイプ）のデバッグ メッセージを表示します。

例：

```
Device# debug crypto pki transactions

Crypto PKI Trans debugging is on
```

ステップ 3 tclsh *flash:signed-tcl-file*

このコマンドは、Tcl シェルで Tcl スクリプトを実行します。

（注） ファイルは、署名付きの Tcl ファイルである必要があります。

例：

```
Device# tclsh flash:hello.tcl

hello
argc = 0
argv =
argv0 = flash:hello.tcl
tcl_interactive = 0
device#
*Apr 21 04:46:18.563: CRYPTO_PKI: locked trustpoint mytrust, refcount is 1
*Apr 21 04:46:18.563: The PKCS #7 message has 0 verified signers.
*Apr 21 04:46:18.563: CRYPTO_PKI: Success on PKCS7 verify!
*Apr 21 04:46:18.563: CRYPTO_PKI: unlocked trustpoint mytrust, refcount is 0
```

次の作業

- 暗号の概要については、『*Security Configuration Guide*』の「Part 5: Implementing and Managing a PKI」の項を参照してください。

署名済み Tcl スクリプトの設定例

キー ペアの生成の例

次に、キー ペア（秘密キーと公開キー）を生成する方法の例を示します。

秘密キーの生成：例

```
Host% openssl genrsa -out privkey.pem 2048
Generating RSA private key, 2048 bit long modulus
.........+++
.....................................................................................+++
e is 65537 (0x10001)
Host% ls -l
total 8
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
Host%
```

秘密キーからの公開キーの生成

```
Host% openssl rsa -in privkey.pem -pubout -out pubkey.pem
writing RSA key
Host% ls -l
total 16
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
```

証明書の生成の例

次に、証明書を生成する例を示します。

```
Host% openssl req -new -x509 -key privkey.pem -out cert.pem -days 1095
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value, If you enter '.', the field will be left
 blank.
-----
Country Name (2 letter code) [GB]:US
State or Province Name (full name) [Berkshire]:California
Locality Name (eg, city) [Newbury]:San Jose
Organization Name (eg, company) [My Company Ltd]:Cisco Systems, Inc.
Organizational Unit Name (eg, section) []:DEPT_ACCT
Common Name (eg, your name or your server's hostname) []:Jane
```

```
Email Address []:janedoe@company.com
Host% ls -l
total 24
-rw-r--r--   1 janedoe eng12       1659 Jun 12 15:01 cert.pem
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
```

Tcl スクリプトの署名の例

次に、Tcl スクリプトに署名する例を示します。

```
Host% openssl smime -sign -in hello -out hello.pk7 -signer cert.pem -inkey privkey.pem
-outform DER -binary
Host% ls -l
total 40
-rw-r--r--   1 janedoe eng12       1659 Jun 12 15:01 cert.pem
-rw-r--r--   1 janedoe eng12        115 Jun 13 10:16 hello
-rw-r--r--   1 janedoe eng12       1876 Jun 13 10:16 hello.pk7
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
```

署名の確認の例

次に、署名を確認する例を示します。

```
Host% openssl smime -verify -in hello.pk7 -CAfile cert.pem -inform DER -content hello
puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
Verification successful
```

非バイナリデータを使用した署名の変換の例

次に、TCL シグニチャを非バイナリ データに変換する方法の例を示します。

```
#Cisco Tcl Signature V1.0
Then append the signature file to the end of the file.
Host% xxd -ps hello.pk7 > hello.hex
Host% cat my_append
#!/usr/bin/env expect
set my_first {#Cisco Tcl Signature V1.0}
set newline {}
set my_file [lindex $argv 0]
set my_new_file ${my_file}_sig
set my_new_handle [open $my_new_file w]
set my_handle [open $my_file r]

puts $my_new_handle $newline
puts $my_new_handle $my_first
foreach line [split [read $my_handle] "\n"]  {
   set new_line {#}
   append new_line $line
   puts $my_new_handle $new_line
}
```

```
close $my_new_handle
close $my_handle
Host% my_append hello.hex
Host% ls -l
total 80
-rw-r--r--   1 janedoe eng12       1659 Jun 12 15:01 cert.pem
-rw-r--r--   1 janedoe eng12        115 Jun 13 10:16 hello
-rw-r--r--   1 janedoe eng12       3815 Jun 13 10:20 hello.hex
-rw-r--r--   1 janedoe eng12       3907 Jun 13 10:22 hello.hex_sig
-rw-r--r--   1 janedoe eng12       1876 Jun 13 10:16 hello.pk7
-rwxr--r--   1 janedoe eng12        444 Jun 13 10:22 my_append
-rw-r--r--   1 janedoe eng12       1679 Jun 12 14:55 privkey.pem
-rw-r--r--   1 janedoe eng12        451 Jun 12 14:57 pubkey.pem
Host% cat hello hello.hex_sig > hello.tcl
Host% cat hello.tcl
puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
#Cisco Tcl Signature V1.0
#3082075006092a864886f70d010702a08207413082073d020101310b3009
#06052b0e03021a0500300b06092a864886f70d010701a08204a13082049d
#30820385a003020102020100300d06092a864886f70d0101040500308195
#310b300906035504061302555331133011060355040813a43616c69666f
#726e69613111300f060355040713085361e204a6f7365311c301a060355
#040a1313436973636f2053797374656d732c20496e632e310e300c060355
#040b13054e53535447310d300b060355040313044a6f686e3121301f0609
#2a864886f70d01090116126a6c6175746d616e40636973636f2e636f6d30
#1e170d3037303631323232303134335a170d3130303631313232303134335
#5a308195310b300906035504061302555331133011060355040813a4361
#6c69666f726e69613111300f060355040713085361e204a6f7365311c30
#1a060355040a1313436973636f2053797374656d732c20496e632e310e30
#0c060355040b13054e53535447310d300b060355040313044a6f686e3121
#301f06092a864886f70d01090116126a6c6175746d616e40636973636f2e
#636f6d30820122300d06092a864886f70d01010105000382010f00308201
#0a0282010100a751eb5ec1f3009738c88a55987c07b759c36f3386342283
#67ea20a89d9483ae85e0c63eeded8ab3eb7a08006689f09136f172183665
#c971099ba54e77ab47706069bbefaaab8c50184396350e4cc870c4c3f477
#88c55c52e2cf411f05b59f0eaec0678ff5cc238fdce2263a9fc6b6c244b8
#ffaead865c19c3d3172674a13b24c8f2c01dd8b1bd491c13e84e29171b85
#f28155d81ac8c69bb25ca23c2921d85fbf745c106e7aff93c72316cbc654
#4a34ea88174a8ba7777fa60662974e1fbac85a0f0aeac925dba6e5e850b8
#7caffce2fe8bb04b61b62f532b5893c081522d538005df81670b931b0ad0
#e1e76ae648f598a9442d5d0976e67c8d55889299147d0203010001a381f5
#3081f2301d0603551d0e04160414bc34132be952ff8b9e1af3b93140a255
#e54a667c3081c20603551d230481ba3081b78014bc34132be952ff8b9e1a
#f3b93140a255e54a667ca1819ba48198308195310b300906035504061302
#5553311330110603550408130a43616c69666f726e69613111300f060355
#0407130853616e204a6f7365311c301a060355040a1313436973636f2053
#797374656d732c20496e632e310e300c060355040b13054e53535447310d
#300b060355040313044a6f686e3121301f06092a864886f70d0109011612
#6a6c6175746d616e40636973636f2e636f6d820100300c0603551d130405
#30030101ff300d06092a864886f70d0101040500038201010000c83c1b074
#6720929c9514af6d5df96f0a95639f047c40a607c83d8362507c58fa7f84
#aa699ec5e5bef61b2308297a0662c653ff446acfbb6f5cb2dd162d939338
#a5e4d78a5c45021e5d4dbabb8784efbf50cab0f5125d164487b31f5cf933
#a9f68f82cd111cbab1739d7f372ec460a7946882874b0a0f22dd53acbd62
#a944a15e52e54a24341b3b8a820f23a5bc7ea7b2278bb56838b8a4051926
#af9c167274ff8449003a4e012bcf4f4b3e280f85209249a390d14df47435
#35efabce720ea3d56803a84a2163db4478ae19d7d987ef6971c8312e280a
#aac0217d4fe620c6582a48faa8ea5e3726a99012e1d55f8d61b066381f77
#4158d144a43fb536c77d6a3182027730820273020101308195310b
```

```
#3009060355040613025553311330110603550408130a43616c69666f726e
#69613111300f060355040713085361 6e204a6f7365311c301a060355040a
#1313436973636f2053797374656d732c20496e632e310e300c060355040b
#13054e53535447310d300b0603550403130 44a6f686e3121301f06092a86
#4886f70d01090116126a6c6175746d616e40636973636f2e636f6d020100
#300906052b0e03021a0500a081b1301806092a864886f70d010903310b06
#092a864886f70d010701301c06092a864886f70d010905310f170d303730
#363133313731363438 5a302306092a864886f70d01090431160414372cb3
#72dc607990577fd0426104a42ee4158d2b305206092a864886f70d01090f
#31453043300a06082a864886f70d0307300e06082a864886f70d03020202
#0080300d06082a864886f70d030202014030070 6052b0e030207300d0608
#2a864886f70d0302020128300d06092a864886f70d010101050004820100
#72db6898742f449b26d3ac18f43a1e7178834fb05ad13951bf042e127eea
#944b72b96f3b8ecf7eb52f3d0e383bf63651750223efe69eae04287c9dae
#b1f31209444108b31d34e46654c6c3cc10b5baba887825c224ec6f376d49
#00ff7ab2d9f88402dab9a2c2ab6aa3ecceeaf5a594bdc7d3a822c55e7daa
#aa0c2b067e06967f22a20e406fe21d9013ecc6bd9cd6d402c2749f8bea61
#9f8f87acfbc9e10d6ce91502e34629adca6ee855419afafe6a8233333e14
#ad4c107901d1f2bca4d7ffaadddbc54192a25da662f8b8509782c76977b8
#94879453fbb00486ccc55f88db50fcc149bae066916b350089cde51a6483
#2ec14019611720fc5bbe2400f24225fc
```

証明書を使用したデバイスの設定の例

次に、証明書でデバイスを設定する例を示します。

```
crypto pki trustpoint mytrust
 enrollment terminal
!
!
crypto pki authentication mytrust
crypto pki certificate chain mytrust
 certificate ca 00
  308204B8 308203A0 A0030201 02020100 300D0609 2A864886 F70D0101 04050030
  819E310B 30090603 55040613 02555331 13301106 03550408 130A4361 6C69666F
  726E6961 3111300F 06035504 07130853 616E204A 6F736531 1C301A06 0355040A
  13134369 73636F20 53797374 656D732C 20496E63 2E310E30 0C060355 040B1305
  4E535354 47311630 14060355 0403130D 4A6F686E 204C6175 746D616E 6E312130
  1F06092A 864886F7 0D010901 16126A6C 6175746D 616E4063 6973636F 2E636F6D
  301E170D 30363131 31373137 35383031 5A170D30 39313131 36313735 3830315A
  30819E31 0B300906 03550406 13025553 31133011 06035504 08130A43 616C6966
  6F726E69 61311130 0F060355 04071308 53616E20 4A6F7365 311C301A 06035504
  0A131343 6973636F 20537973 74656D73 2C20496E 632E310E 300C0603 55040B13
  054E5353 54473116 30140603 55040313 0D4A6F68 6E204C61 75746D61 6E6E3121
  301F0609 2A864886 F70D0109 0116126A 6C617574 6D616E40 63697363 6F2E636F
  6D308201 22300D06 092A8648 86F70D01 01010500 0382010F 00308201 0A028201
  0100BC6D A933028A B31BF827 7258BB87 A1600CF0 21090F04 2080BECC 5818688B
  74D231DF F0C365C1 07D6E206 D7651FA8 C7B230A2 3B0011E4 EA2B6A4C 1F3F27FB
  9AF449D8 FA8900BB 3E567F77 5412881B AAD9525E 3EC1D3B1 EBCE8155 D74866F1
  0940F6D1 3A2613CD F6B3595E F468B315 6DDEFF07 BBC5D521 B560AF72 D6D5FDA7
  D9D9C99D 31E3B380 5DEB7039 A1A29EF9 46ED536E 4D768048 12D48C24 59B08973
  481AD75D E741CD9E BE06EA16 9B514AE3 91184A56 A0E51B7D 4465D730 1AB3C7DD
  62CA1AC9 DF30C39A 41316B8E 72289113 98080354 C7297AD7 89B627F8 ED40D924
  ADF48383 1B332C7F 73C58686 6279E2A4 4BF41644 3E60F131 090D3F5D 25F0C025
  43CB0203 010001A3 81FE3081 FB301D06 03551D0E 04160414 F7F4E80E F6CC4772
  5F278C44 6B85F8EE 8345AB99 3081CB06 03551D23 0481C330 81C08014 F7F4E80E
  F6CC4772 5F278C44 6B85F8EE 8345AB99 A181A4A4 81A13081 9E310B30 09060355
  04061302 55533113 30110603 55040813 0A43616C 69666F72 6E696131 11300F06
  03550407 13085361 6E204A6F 7365311C 301A0603 55040A13 13436973 636F2053
  79737465 6D732C20 496E632E 310E300C 06035504 0B13054E 53535447 31163014
  06035504 03130D4A 6F686E20 4C617574 6D616E6E 3121301F 06092A86 4886F70D
```

```
  01090116 126A6C61 75746D61 6E406369 73636F2E 636F6D82 0100300C 0603551D
  13040530 030101FF 300D0609 2A864886 F70D0101 04050003 82010100 6D12CFF8
  31078DF6 94FE5CF0 8F83639B 414F32D8 069D23E2 37E182BE 7C31EC14 E87AF216
  61A6CCD3 37656934 4BE4157A 400E182B EC390D1A DC130A56 B8F35BFB D2234556
  24152FE8 A736B670 58CC684E 750D08AE C7739907 917B7A72 3D26BEC7 9F554CF1
  5E5EF499 ABA11124 55966616 AC9C52B2 B1082DEA D962CBAF E476C575 A9DDFBFA
  C4AE63F6 1D5C9F76 7B4B9CA7 52CE65C9 E65C04FC 4B7642D6 0D1A8AF4 38194B7A
  CA307EC9 51DCB847 8B8C27FB 98ACEE60 0B80DC3F 36E4E252 BD731F5F 0E781E26
  C1CA4120 9B0B689B BA654250 97B22A76 CC126B77 C7779AAA D3F93C3F DCF46006
  2B7F7F8C 150AF889 BBEC62F1 E53B4F3B A3626CD6 05B8AB3D F8A6A361
   quit
archive
 log config
scripting tcl trustpoint name mytrust
scripting tcl secure-mode
!
!
end
```

その他の参考資料

ここでは、Cisco IOS CLI を使用した EEM ポリシーの記述に関する関連資料について説明します。

関連資料

関連項目	マニュアル タイトル
EEM コマンド：コマンド構文の詳細、デフォルト、コマンドモード、コマンド履歴、使用上の注意事項、および例	Cisco IOS Embedded Event Manager のコマンド リファレンス
Embedded Event Manager 概要	「Embedded Event Manager の概要」の章
Tcl を使用して Embedded Event Manager ポリシーを記述する	「Tcl を使用した Embedded Event Manager ポリシーの記述」の章
拡張オブジェクト トラッキングの設定	「Configuring Enhanced Object Tracking」の章

標準

標準	タイトル
新しい規格または変更された規格はサポートされていません。また、既存の規格に対するサポートに変更はありません。	--

MIB

MIB	MIB のリンク
CISCO-EMBEDDED-EVENT-MGR-MIB	選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
新しい RFC または変更された RFC はサポートされていません。また、既存の RFC に対するサポートに変更はありません。	--

シスコのテクニカル サポート

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

署名済み Tcl スクリプトの機能情報

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

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

表 *201* : 署名済み *Tcl* スクリプトの機能情報

機能名	リリース	機能情報
署名済み Tcl スクリプト	15.2(5)E1	この機能は、c2960cx にのみ導入され、サポートされています。

用語集

CA：認証局。証明書要求の管理と、関係する IP セキュリティ ネットワーク デバイスへの証明書の発行を担当しているサービス。このサービスは、参加デバイスを一元的に管理します。またこれらのサービスによって受信者は、明示的に信頼してアイデンティティを確認し、デジタル証明書を作成できます。

証明書：ユーザー名またはデバイス名を公開キーにバインドする電子ドキュメント。証明書は、一般的にデジタル署名を確認するために使用されます。

CRL：証明書失効リスト。失効した証明書のリストが含まれる電子ドキュメントです。CRL は、証明書を発行した CA によって作成され、デジタル署名されます。CRL には、証明書の発行日と失効日が含まれています。現行の CRL が失効すると、新しい CRL が発行されます。

IPsec：IP セキュリティ。

ピア証明書：ピアが提示する証明書で、ピアの公開キーが含まれており、トラストポイント CA によって署名されています。

PKI：公開キー インフラストラクチャ。セキュアに設定された通信に使用されているネットワーク コンポーネントの暗号キーと ID 情報を管理するシステムです。

RA：登録局。CA のプロキシとして機能するサーバーで、CA がオフラインのときでも CA の機能を継続できます。RA は CA サーバー上に設定するのが通常ですが、別アプリケーションとして、稼働のための別デバイスを必要とする場合もあります。

RSA キー：公開キー暗号化システムで、Ron Rivest（ロナルド・リベスト）、Adi Shamir（アディ・シャミア）、Leonard Adleman（レオナルド・エーデルマン）の 3 人によって開発されました。デバイスの証明書を取得するには、RSA キーペア（公開キーと秘密キー）が必要です。

SHA1：Secure Hash Algorithm 1。

SSH：セキュア シェル。

SSL：Secure Socket Layer。

注意事項

本ソフトウェア ライセンスに関連する通知内容を以下に示します。

OpenSSL/Open SSL Project

本製品には、OpenSSL Toolkit（http://www.openssl.org/）で使用するために OpenSSL Project によって開発されたソフトウェアが含まれています。

本製品には、Eric Young 氏（eay@cryptsoft.com）によって作成された暗号化ソフトウェアが含まれています。

本製品には、Tim Hudson 氏（tjh@cryptsoft.com）によって作成されたソフトウェアが含まれています。

ライセンスの問題

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License:

Copyright © 1998-2007 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. 本ソフトウェアの機能または使用に言及するすべての広告資料には、以下の謝辞が表示される必要があります。「本製品には、OpenSSL Toolkit で使用するために OpenSSL Project によって開発されたソフトウェアが含まれています（http://www.openssl.org/）」。
4. The names “OpenSSL Toolkit” and “OpenSSL Project” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called “OpenSSL” nor may “OpenSSL” appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment:

「本製品には、OpenSSL Toolkit（http://www.openssl.org/）で使用するために OpenSSL プロジェクトによって開発されたソフトウェアが含まれています」。

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT “AS IS”' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

本製品には、Eric Young 氏（eay@cryptsoft.com）によって作成された暗号化ソフトウェアが含まれています。本製品には、Tim Hudson 氏（tjh@cryptsoft.com）によって作成されたソフトウェアが含まれています。

Original SSLeay License:

Copyright © 1995-1998 Eric Young (eay@cryptsoft.com). All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com).

The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young’s, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

“This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)”.

The word ‘cryptographic’ can be left out if the routines from the library being used are not cryptography-related.

1. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: “This product includes software written by Tim Hudson (tjh@cryptsoft.com)”.

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF

MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The license and distribution terms for any publicly available version or derivative of this code cannot be changed, i.e. this code cannot simply be copied and put under another distribution license [including the GNU Public License.]

第 91 章

EEM CLI ライブラリのコマンド拡張

すべてのコマンドラインインターフェイス（CLI）ライブラリ コマンド拡張は、::cisco::eem 名前空間に属します。

このライブラリによって、ユーザーに対し、CLI コマンドを実行し、Tcl でコマンドの出力を取得する機能が用意されます。コマンドが exec によって実行され、コマンドの出力が読み戻されるようにするため、ユーザーは、このライブラリでコマンドを使用して、exec を生成し、それに対して仮想端末チャネルをオープンし、コマンドを記述してチャネルに対して実行できます。

CLI コマンドには、対話式コマンドと非対話式コマンドの、2 つのタイプがあります。

対話式コマンドでは、コマンドの入力後、デバイスによって異なるユーザー オプションが質問される「Q&A」フェーズがあり、ユーザーは、各質問に対する答えを入力する必要があります。すべての質問が適切に答えられた後、ユーザーのオプションに従って、完了するまでコマンドが実行されます。

非対話式コマンドでは、コマンドが一度入力されると、コマンドが完了まで実行されます。EEM スクリプトを使用してさまざまなタイプのコマンドを実行するには、異なる CLI ライブラリ コマンド シーケンスを使用する必要があります。詳細については、cli_write Tcl コマンドの「CLI ライブラリを使用した非対話式コマンドの実行」の項および「CLI ライブラリを使用した対話式コマンドの実行」の項を参照してください。

vty 行は、**line vty** CLI コンフィギュレーション コマンドを使用して設定された vty 行のプールから割り当てられます。EEM によって vty 行が使用されていない場合で、使用可能な vty 行がある場合、EEM では、vty 行が使用されます。EEM によって vty 行がすでに使用されている場合で、使用可能な 3 行以上の vty 行がある場合も、EEM では、vty 行が使用されます。3 行よりも少ない vty 行が使用可能な場合、残りの vty 行は Telnet で使用するために予約されているので、接続は失敗することに注意してください。

お使いのリリースで XML-PI がサポートされている場合があります。XML-PI サポート、新しい CLI ライブラリ コマンド拡張、および、XML-PI の実装方法の例については、「EEM CLI ライブラリ XML-PI サポート」を参照してください。

cli_close

exec プロセスをクローズし、コマンドライン インターフェイス（CLI）に接続された、vty および指定されたチャネル ハンドラをリリースします。

構文

```
cli_close fd tty_id
```

引数

fd	（必須）CLI チャネル ハンドラ。
tty_id	（必須）**cli_open** コマンド拡張から返された TTY ID。

結果文字列

なし

_cerrno を設定

チャネルをクローズできない。

cli_exec

指定されたチャネル ハンドラにコマンドを記述し、コマンドを実行します。次に、チャネルからコマンドの出力を読み取り、出力を返します。

構文

```
cli_exec fd cmd
```

引数

fd	（必須）コマンドライン インターフェイス（CLI）チャネル ハンドラ。
cmd	（必須）実行する CLI コマンド。

結果文字列

実行された CLI コマンドの出力。

_cerrno を設定

チャネルを読み取れない。

cli_get_ttyname

該当する TTY ID の実際と疑似の TTY の名前を返します。

構文

```
cli_get_ttyname tty_id
```

引数

tty_id	（必須）**cli_open** コマンド拡張から返された TTY ID。

結果文字列

```
pty %s tty %s
```

_cerrno を設定

なし

cli_open

vty を割り当て、EXEC コマンドライン インターフェイス（CLI）セッションを作成し、vty をチャネル ハンドラに接続します。チャネル ハンドラを含む配列を返します。

（注） **cli_open** への各コールによって、Cisco IOS vty 回線を割り当てる Cisco IOS EXEC セッションが開始されます。vty は、**cli_close** ルーチンが呼び出されるまで、使用中のままです。vty 行は、**line vty** CLI コンフィギュレーション コマンドを使用して設定された vty 行のプールから割り当てられます。EEM によって vty 行が使用されていない場合で、使用可能な vty 行がある場合、EEM では、vty 行が使用されます。EEM によって vty 行がすでに使用されている場合で、使用可能な 3 行以上の vty 行がある場合も、EEM では、vty 行が使用されます。3 行よりも少ない vty 行が使用可能な場合、残りの vty 行は Telnet で使用するために予約されているので、接続は失敗することに注意してください。

構文

```
cli_open
```

引数

なし

結果文字列

```
"tty_id {%s} pty {%d} tty {%d} fd {%d}"
```

イベントタイプ	説明
tty_id	TTY ID。
pty	PTY デバイス名。
tty	TTY デバイス名。
fd	CLI チャネル ハンドラ。

_cerrno を設定

- EXEC の pty を取得できない。
- EXEC CLI セッションを作成できない。
- 最初のプロンプトを読み取れない。

cli_read

読み取られている内容でデバイスプロンプトのパターンが発生するまで、指定されたコマンドラインインターフェイス（CLI）のチャネルハンドラからコマンド出力を読み取ります。一致するまで、読み取られたすべての内容を返します。

構文

```
cli_read fd
```

引数

fd	（必須）CLI チャネル ハンドラ。

結果文字列

読み取られたすべての内容。

_cerrno を設定

デバイス名を取得できない。

（注） この Tcl コマンド拡張によって、デバイス プロンプトを待つ状態がブロックされ、読み取られた内容が表示されます。

cli_read_drain

指定されたコマンドライン インターフェイス（CLI）のチャネル ハンドラのコマンド出力を読み取り、排出します。読み取られたすべての内容を返します。

構文

```
cli_read_drain fd
```

引数

fd	（必須）CLI チャネル ハンドラ。

結果文字列

読み取られたすべての内容。

_cerrno を設定

なし

cli_read_line

指定されたコマンドラインインターフェイス（CLI）のチャネルハンドラから、コマンド出力の 1 行を読み取ります。読み取られた回線を返します。

構文

```
cli_read_line fd
```

引数

fd	（必須）CLI チャネル ハンドラ。

結果文字列

読み取られた回線。

_cerrno を設定

なし

（注） この Tcl コマンド拡張によって、行の末尾を待つ状態がブロックされ、読み取られた内容が表示されます。

cli_read_pattern

読み取られている内容でパターンが発生するまで、指定されたコマンドラインインターフェイス（CLI）のチャネルハンドラからコマンド出力を読み取ります。一致するまで、読み取られたすべての内容を返します。

（注） パターンマッチロジックで、Cisco IOS コマンドから配信されるコマンド出力データを探すことによって、照会が試行されます。照会は、出力バッファの最新の 256 文字で常に行われます。ただし、使用可能な文字がより少ない場合は、より少ない文字で照会が行われます。正常な一致に 256 よりも多い文字が必要な場合、パターン マッチは実行されません。

構文

```
cli_read_pattern fd ptn
```

引数

fd	（必須）CLI チャネル ハンドラ。
ptn	（必須）チャネルからコマンド出力を読み取るときに、パターンが照会されます。

結果文字列

読み取られたすべての内容。

_cerrno を設定

なし

（注） このTclコマンド拡張によって、指定されたパターンを待つ状態がブロックされ、読み取られた内容が表示されます。

cli_run

clist にある回数を繰り返し、それぞれが、イネーブル モードで実行されるコマンドライン インターフェイス（CLI）であることを前提とします。正常に実行されると、実行されたすべてのコマンドの出力を返します。失敗すると、失敗からのエラーを返します。

構文

```
cli_run clist
```

引数

clist	（必須）実行されるコマンドのリスト。

結果文字列

出力されるすべてのコマンドの出力、またはエラー メッセージ。

_cerrno を設定

なし。

使用例

次に、**cli_run** コマンド拡張の使用例を示します。

```
set clist [list {sh run} {sh ver} {sh event man pol reg}]
cli_run { clist }
```

cli_run_interactive

3 つの項目がある clist のサブリストを提供します。正常に実行されると、実行されたすべてのコマンドの出力を返します。失敗すると、失敗からのエラーを返します。可能な場合には、配列も使用します。予測と応答を別々に保持することによって、より簡単に後で読み取ることができます。

構文

```
cli_run_interactive clist
```

引数

clist	（必須）3 つの項目のリスト： • command：実行するコマンド • expect：予想される応答プロンプトの正規表現パターンマッチ • responses：2 つの項目のアレイとして構成された応答プロンプトに対して可能性がある応答のリスト • expect：可能性がある応答プロンプトの正規表現パターン マッチ • reply：その予測されるプロンプトの応答

結果文字列

出力されるすべてのコマンドの出力、またはエラーメッセージ。各コマンドが実行されると、その出力が結果の変数に追加されます。入力リストが枯渇すると、CLI チャネルが閉じ、集約結果が返されます。

_cerrno を設定

なし。

使用例

次に、cli_run_ interactive コマンド拡張を使用してインターフェイス fa0/0 のカウンタをクリアする例を示します。

```
set cmdarr(command) "clear counters fa0/0"
set cmdarr(responses) [list]
set resps(expect) {[confirm]}
set resps(reply) "y"
lappend cmdarr(responses) [array get resps]
set rc [catch {cli_run_interactive [list [array get cmdarr]]} result]
```

発生する可能性があるエラーには、次のようなものがあります。

- exec の pty を取得できない。
- exec を生成できない。
- 最初のプロンプトを読み取れない。
- チャネルを読み取れない。
- チャネルをクローズできない。

cli_write

指定されたCLIチャネル ハンドラに対して実行されるコマンドを書き込みます。CLIチャネル ハンドラによって、コマンドが実行されます。

構文

```
cli_write fd cmd
```

引数

fd	（必須）CLI チャネル ハンドラ。
cmd	（必須）実行する CLI コマンド。

結果文字列

なし

_cerrno を設定

なし

使用例

たとえば、次のように、コンフィギュレーション CLI コマンドを使用して、イーサネット インターフェイス 1/0 をアップにします。

```
if [catch {cli_open} result] {
puts stderr $result
exit 1
} else {
array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "config t"} result] {
puts stderr $result
exit 1
```

```
}
if [catch {cli_exec $cli1(fd) "interface Ethernet1/0"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "no shut"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "end"} result] {
puts stderr $result
exit 1
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} } result] {
puts stderr $result
exit 1
```

CLI ライブラリを使用した非対話式コマンドの実行

非対話式コマンドを実行するには、**cli_exec** コマンド拡張を使用して、コマンドを発行し、次に、出力とデバイスプロンプトを待ちます。たとえば、コンフィギュレーション CLI コマンドを使用して、イーサネット インターフェイス 1/0 をアップにする例を示します。

```
if [catch {cli_open} result] {
error $result $errorInfo
} else {
set fd $result
}
if [catch {cli_exec $fd "en"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "config t"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "interface Ethernet1/0"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "no shut"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "end"} result] {
error $result $errorInfo
}
if [catch {cli_close $fd} result] {
error $result $errorInfo
}
```

CLI ライブラリを使用した対話式コマンドの実行

対話式コマンドを実行するには、次の 3 つのフェーズが必要です。

- フェーズ 1：**cli_write** コマンド拡張を使用して、コマンドを発行します。
- フェーズ 2：Q&A フェーズ。**cli_read_pattern** コマンド拡張を使用して質問を読み取り（質問テキストの照合に指定される通常パターン）、**cli_write** コマンド拡張を使用して、代わりに回答を書き戻します。

- フェーズ 3：非対話式フェーズ。すべての質問が回答され、完了までコマンドが実行されます。**cli_read** コマンド拡張を使用して、コマンドの出力とデバイスプロンプトを待ちます。

たとえば、CLI コマンドを使用して、ブートフラッシュをまとめます。Tcl 変数 cmd_output に、このコマンドの出力を保存します。

```
if [catch {cli_open} result] {
error $result $errorInfo
} else {
array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
error $result $errorInfo
}

# Phase 1: issue the command
if [catch {cli_write $cli1(fd) "squeeze bootflash:"} result] {
error $result $errorInfo
}

# Phase 2: Q&A phase
# wait for prompted question:
# All deleted files will be removed. Continue? [confirm]
if [catch {cli_read_pattern $cli1(fd) "All deleted"} result] {
error $result $errorInfo
}
# write a newline character
if [catch {cli_write $cli1(fd) "\n"} result] {
error $result $errorInfo
}
# wait for prompted question:
# Squeeze operation may take a while. Continue? [confirm]
if [catch {cli_read_pattern $cli1(fd) "Squeeze operation"} result] {
error $result $errorInfo
}
# write a newline character
if [catch {cli_write $cli1(fd) "\n"} result] {
error $result $errorInfo
}

# Phase 3: noninteractive phase
# wait for command to complete and the router prompt
if [catch {cli_read $cli1(fd) } result] {
error $result $errorInfo
} else {
set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
error $result $errorInfo
}
```

次に、CLI **reload** コマンドを使用して、デバイスがリロードされる例を示します。EEM **action_reload** コマンドによって、より効率的な方法で同じ結果が達成されますが、この例は、対話式コマンド実行での CLI ライブラリでの柔軟性を示すために示します。

```
# 1. execute the reload command
if [catch {cli_open} result] {
    error $result $errorInfo
} else {
```

```
    array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
    error $result $errorInfo
}
if [catch {cli_write $cli1(fd) "reload"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_read_pattern $cli1(fd) ".*(System configuration has been modified. Save\\\?
 \\\[yes/no\\\]: )"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_write $cli1(fd) "no"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_read_pattern $cli1(fd) ".*(Proceed with reload\\\? \\\[confirm\\\])"}
result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_write $cli1(fd) "y"} result] {
    error $result $errorInfo
} else {
    set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
    error $result $errorInfo
}
```

EEM 4.0 CLI ライブラリ XML-PI サポート

EEM CLI ライブラリ XML-PI サポート

XML プログラマチック インターフェイス（XML-PI）が Cisco IOS Release 12.4(22)T で導入されました。XML-PI は異なるシスコ製品間で矛盾のない方法で、IOS コマンドライン インターフェイス（CLI）show コマンドを XML 形式にカプセル化した、プログラム可能なインターフェイスを提供します。XML-PI を使用する場合は、既知のキーワードを使用して IOS show コマンドの出力を Tcl スクリプトから解析できます。「スクリーン スクレイピング」出力に対する正規表現サポートを使用する必要はありません。

XML-PI コマンド拡張を使用する利点は、CLI **show** コマンドを使用して生成される特定の出力情報の抽出を容易にすることです。ほとんどの show コマンドは出力内の多くのフィールドを返しますが、現在のところ、行の中央に表示される可能性がある特定の情報を抽出するには正規表現を使用する必要があります。XML-PI サポートは一連の Tcl ライブラリ関数を提供し、次の形式の IOS CLI 形式の拡張からの出力の解析を容易にします。

show

```
 <
show-command
> | format
 {
spec-file
}
```

ここで、spec-file は現在サポートされている各 **show** コマンドのすべての SPEC ファイルエントリ（SFE）を連結したものです。XML-PI プロジェクトの一環として、デフォルトの spec-file が IOS リリース 12.4(22)T イメージに組み込まれます。デフォルトの spec-file には、一連の少数のコマンドが組み込まれ、それらのコマンドの SFE には考えられるタグのサブセットが組み込まれます。format コマンドで spec-file が提供されない場合、デフォルトの spec-file が使用されます。

XML-PI に関するより全般的な詳細については、「XML-PI」の章を参照してください。

第 92 章

EEM コンテキスト ライブラリのコマンド拡張

すべての Tcl コンテキスト ライブラリ コマンド拡張は、::cisco::eem 名前空間に属します。

context_retrieve

該当するコンテキスト名、使用されている可能性があるスカラ変数名、配列型変数名、および配列の索引によって指定される Tcl 変数を取得します。取得される情報は、自動的に削除されます。

（注） 保存される情報が一度取得されると、自動的に削除されます。その情報が別のポリシーで必要な場合、（**context_retrieve** コマンド拡張を使用して）それを取得するポリシーも、（**context_save** コマンド拡張を使用して）再度保存する必要があります。

構文

```
context_retrieve ctxt [var] [index_if_array]
```

引数

ctxt	（必須）コンテキスト名。
var	（任意）スカラ変数名または配列型変数名。この引数が指定されない場合、ヌル文字列を定義します。
index_if_array	（任意）配列の索引。

（注） var 引数がスカラ変数の場合、index_if_array 引数は無視されます。

var が未指定の場合、コンテキストに保存されている変数テーブル全体を取得します。

var が指定され、index_if_array が指定されない場合、または、index_if_array が指定されるが var がスカラ変数の場合、var の値を取得します。

var が指定され、index_if_array が指定され、var が配列変数の場合、指定された配列エレメントの値を取得します。

結果文字列

保存が実行されたときの状態に、Tcl グローバル変数をリセットします。

_cerrno を設定

- appl_reqinfo エラーが原因で、_cerrno、_cerr_sub_num、_cerr_sub_err、_cerr_posix_err、_cerr_str を表示する文字列。
- 変数がコンテキストにない。

使用例

次に、**context_save** コマンド拡張機能および **context_retrieve** コマンド拡張機能を使用して、データを保存し、取得する例を示します。例は、保存と取得のペアで示されます。

例 1：保存

var が未指定か、またはパターンが指定される場合、複数の変数をコンテキストに保存します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvara 123
set testvarb 345
set testvarc 789
if {[catch {context_save TESTCTX "testvar*"} errmsg]} {
      action_syslog msg "context_save failed: $errmsg"
} else {
      action_syslog msg "context_save succeeded"
}
```

例 1：取得

var が未指定の場合、複数の変数をコンテキストから取得します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

if {[catch {foreach {var value} [context_retrieve TESTCTX] {set $var $value}} errmsg]}
```

```
{
      action_syslog msg "context_retrieve failed: $errmsg"
} else {
      action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvara]} {
      action_syslog msg "testvara exists and is $testvara"
} else {
      action_syslog msg "testvara does not exist"
}
if {[info exists testvarb]} {
      action_syslog msg "testvarb exists and is $testvarb"
} else {
      action_syslog msg "testvarb does not exist"
}
if {[info exists testvarc]} {
      action_syslog msg "testvarc exists and is $testvarc"
} else {
      action_syslog msg "testvarc does not exist"
}
```

例 2：保存

var が指定される場合、var の値を保存します。

```
::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvar 123
if {[catch {context_save TESTCTX testvar} errmsg]} {
      action_syslog msg "context_save failed: $errmsg"
} else {
      action_syslog msg "context_save succeeded"
}
```

例 2：取得

var が指定され、index_if_array が指定されない場合、または、index_if_array が指定されるが var がスカラ変数の場合、var の値を取得します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
      action_syslog msg "context_retrieve failed: $errmsg"
} else {
      action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
      action_syslog msg "testvar exists and is $testvar"
} else {
      action_syslog msg "testvar does not exist"
}
```

例 3：保存

var が指定される場合、それが配列の場合でも、var の値を保存します。

```
::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
      action_syslog msg "context_save failed: $errmsg"
} else {
      action_syslog msg "context_save succeeded"
}
```

例 3：取得

var が指定され、index_if_array が指定されず、var が配列変数の場合、配列全体を取得します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {array set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
      action_syslog msg "context_retrieve failed: $errmsg"
} else {
      action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
      action_syslog msg "testvar exists and is [array get testvar]"
} else {
      action_syslog msg "testvar does not exist"
}
```

例 4：保存

var が指定される場合、それが配列の場合でも、var の値を保存します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
      action_syslog msg "context_save failed: $errmsg"
} else {
      action_syslog msg "context_save succeeded"
}
```

例 4：取得

var が指定され、index_if_array が指定され、var が配列変数の場合、指定された配列エレメントの値を取得します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar testvar1]} errmsg]} {
      action_syslog msg "context_retrieve failed: $errmsg"
} else {
      action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
      action_syslog msg "testvar exists and is $testvar"
```

```
} else {
      action_syslog msg "testvar doesn't exist"
}
```

context_save

現在およびグローバルな名前空間で、指定されたパターンが、識別情報として指定されたコンテキスト名と一致する、Tcl変数を保存します。この Tcl コマンド拡張を使用すると、ポリシー外の情報が保存されます。保存された情報は、**context_retrieve** コマンド拡張を使用して、異なるポリシーによって取得できます。

（注） 保存される情報が一度取得されると、自動的に削除されます。その情報が別のポリシーで必要な場合、（**context_retrieve** コマンド拡張を使用して）それを取得するポリシーも、（**context_save** コマンド拡張を使用して）再度保存する必要があります。

構文

```
context_save ctxt [pattern]
```

引数

ctxt	（必須）コンテキスト名。
pattern	（任意）**string match** Tcl コマンドによって使用される、glob-style パターン。この引数が指定されない場合、パターンのデフォルトは、ワイルドカード * です。 glob パターンで使用されている、3 つの構成があります。 • * = すべての文字 • ? = 1 文字 • [abc] = 文字のセットの 1 つと照合

結果文字列

なし

_cerrno を設定

appl_setinfo エラーが原因で、_cerrno、_cerr_sub_num、_cerr_sub_err、_cerr_posix_err、_cerr_str を表示する文字列。

使用例

次に、**context_save** コマンド拡張機能および **context_retrieve** コマンド拡張機能を使用して、データを保存し、取得する例を示します。例は、保存と取得のペアで示されます。

例 1：保存

varが未指定か、またはパターンが指定される場合、複数の変数をコンテキストに保存します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvara 123
set testvarb 345
set testvarc 789
if {[catch {context_save TESTCTX "testvar*"} errmsg]} {
      action_syslog msg "context_save failed: $errmsg"
} else {
      action_syslog msg "context_save succeeded"
}
```

例 1：取得

var が未指定の場合、複数の変数をコンテキストから取得します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

if {[catch {foreach {var value} [context_retrieve TESTCTX] {set $var $value}} errmsg]}
{
      action_syslog msg "context_retrieve failed: $errmsg"
} else {
      action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvara]} {
      action_syslog msg "testvara exists and is $testvara"
} else {
      action_syslog msg "testvara does not exist"
}
if {[info exists testvarb]} {
      action_syslog msg "testvarb exists and is $testvarb"
} else {
      action_syslog msg "testvarb does not exist"
}
if {[info exists testvarc]} {
      action_syslog msg "testvarc exists and is $testvarc"
} else {
      action_syslog msg "testvarc does not exist"
}
```

例 2：保存

var が指定される場合、var の値を保存します。

```
::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
```

```
set testvar 123
if {[catch {context_save TESTCTX testvar} errmsg]} {
      action_syslog msg "context_save failed: $errmsg"
} else {
      action_syslog msg "context_save succeeded"
}
```

例 2：取得

var が指定され、index_if_array が指定されない場合、または、index_if_array が指定されるが var がスカラ変数の場合、var の値を取得します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
      action_syslog msg "context_retrieve failed: $errmsg"
} else {
      action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
      action_syslog msg "testvar exists and is $testvar"
} else {
      action_syslog msg "testvar does not exist"
}
```

例 3：保存

var が指定される場合、それが配列の場合でも、var の値を保存します。

```
::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
      action_syslog msg "context_save failed: $errmsg"
} else {
      action_syslog msg "context_save succeeded"
}
```

例 3：取得

var が指定され、index_if_array が指定されず、var が配列変数の場合、配列全体を取得します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {array set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
      action_syslog msg "context_retrieve failed: $errmsg"
} else {
      action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
      action_syslog msg "testvar exists and is [array get testvar]"
} else {
      action_syslog msg "testvar does not exist"
}
```

例 4：保存

var が指定される場合、それが配列の場合でも、var の値を保存します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
      action_syslog msg "context_save failed: $errmsg"
} else {
      action_syslog msg "context_save succeeded"
}
```

例 4：取得

var が指定され、index_if_array が指定され、var が配列変数の場合、指定された配列エレメントの値を取得します。

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar testvar1]} errmsg]} {
      action_syslog msg "context_retrieve failed: $errmsg"
} else {
      action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
      action_syslog msg "testvar exists and is $testvar"
} else {
      action_syslog msg "testvar doesn't exist"
}
```

第 93 章

EEM イベント登録の Tcl コマンド拡張

次の表記法が、Tcl コマンド拡張ページで説明されている構文に使用されます。

- 任意の引数は、たとえば次の例のように、角カッコ内に示されます。

[type ?]

- 疑問符（?）は入力する変数を表します。
- 引数間の選択肢は、たとえば次の例のように、パイプ文字で示されます。

priority low|normal|high

（注） すべての EEM Tcl コマンド拡張について、エラーがあった場合、戻される Tcl 結果文字列には、エラー情報が含まれます。

（注） 数値範囲が指定されていない引数は、–2147483648 から 2147483647 までの整数から取得されます。

event_register_appl

アプリケーション イベントの登録を行います。この Tcl コマンド拡張は、**event_publish** Tcl コマンド拡張の別のポリシーの実行に続いて、アプリケーションイベントがトリガされたときにポリシーを実行するために使用します。**event_publish** コマンド拡張によって、アプリケーションイベントがパブリッシュされます。

アプリケーション イベントを登録するためには、サブシステムを指定する必要があります。Tcl ポリシーまたは内部 Embedded Event Manager（EEM）API のいずれかによって、アプリケーションイベントをパブリッシュできます。イベントがポリシーによってパブリッシュされている場合、ポリシーで予約される sub_system 引数は 798 です。

構文

```
event_register_appl [tag ?] sub_system ? type ? [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
sub_system	（必須）アプリケーション イベントをパブリッシュした EEM ポリシーに割り当てられる番号。他のすべての番号は Cisco での使用のために予約されているため、番号は 798 に設定されます。この引数が指定されない場合、すべてのコンポーネントが照会されます。

type	（必須）指定されたイベント内のイベント サブタイプ。sub_system 引数および type 引数によって、アプリケーション イベントが一意に識別されます。この引数が指定されない場合、すべてのタイプが照会されます。この引数を指定する場合、1 ～ 4294967295 の整数を選択する必要があります。 パブリッシュと登録が機能するためには、**event_publish** コマンド拡張と **event_register_appl** コマンド拡張の間でコンポーネントとタイプが一致する必要があります。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

複数の条件が存在する場合、すべての条件が満たされたときに、アプリケーションイベントが発生します。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"sub_system 0x%x type %u data1 {%s} data2 {%s} data3 {%s} data4 {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	イベントが Embedded Event Manager（EEM）にパブリッシュされたときの、秒単位およびミリ秒単位の時間。
sub_system	アプリケーション イベントをパブリッシュした EEM ポリシーに割り当てられる番号。他のすべての番号は Cisco での使用のために予約されているため、番号は 798 に設定されます。
type	指定されたコンポーネント内のイベント サブタイプ。
data1 data2 data3 data4	イベントがパブリッシュされるときに、アプリケーション固有のイベントに渡される、引数データ。データは、文字テキスト、環境変数、または、この 2 つの組み合わせです。

event_register_cli

CLI イベントの登録を行います。この Tcl コマンドを使用すると、拡張 CLI コマンドに対して実行されるパターン マッチに基づいて、特定パターンの CLI コマンドが入力されるときに、ポリシーが実行されます。

（注） ユーザーは、**sh mem summary** などの省略形の CLI コマンドを入力できます。パーサーによってコマンドが **show memory summary** に拡張され、照会が実行されます。

（注） CLI イベント ディテクタによる機能は、有効な IOS CLI コマンドでの正規表現パターン比較機能だけです。これには、リダイレクションが使用される場合のパイプ記号（|）以降のテキストは含まれません。

構文

```
event_register_cli [tag ?] sync yes|no skip yes|no
[occurs ?] [period ?] pattern ? [default ?] [enter] [questionmark] [tab] [mode]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
sync	（必須）「yes」は、ポリシー（イベント パブリッシュ）が、CLI コマンドと同期的に実行することを意味します。「no」は、イベント パブリッシュが CLI コマンドと非同期に実行されることを意味します。ポリシーの実行が完了すると、イベントディテクタによって通知されます。ポリシーの終了ステータスは、CLI コマンドを実行する必要があるかどうかを示します。終了ステータスがゼロの場合は、ポリシーが正常に実行されたことを意味し、CLI コマンドは実行されません。それ以外の場合は、CLI コマンドが実行されます。
skip	sync 引数が no の場合は必須で、sync 引数が yes の場合は不要ですskip 引数が yes の場合、CLI コマンドを実行する必要がないことを意味します。skip 引数が no の場合、CLI コマンドを実行する必要があることを意味します。 **注意** skip 引数が yes の場合、パターン マッチがコンフィギュレーション コマンドに対して行われる場合、正規表現に一致する CLI コマンドは実行されないため、想定外の結果が生成される場合があります。
occurs	（任意）イベントが発生する前の発生回数。この引数が指定されない場合、イベントは1回目から発生します。この引数が指定される場合は、1 ～ 4294967295 の範囲の整数である必要があります。
period	（任意）イベントがパブリッシュされるようにするために、すべての CLI イベントが発生する必要がある（occurs 句を満たす必要がある）逆方向検索時間ウィンドウを指定します（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合は、最新のイベントが使用されます。
pattern	（必須）CLI コマンド パターン マッチの実行に使用される正規表現を指定します。

デフォルト	（任意）CLI イベント ディテクタがポリシーの終了を待つ時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。ポリシーが終了する前にデフォルトの時間の期限が切れると、デフォルト アクションが実行されます。デフォルト アクションによって、コマンドが実行されます。この引数が指定されない場合、デフォルトの時間は 30 秒に設定されます。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
enter	（任意）ユーザーが Enter キーを押したときにイベント照会が実行されるよう、指定します。このパラメータが使用されると、照会前には入力文字列は拡張されません。
questionmark	（任意）ユーザーが ? キーを押したときにイベント照会が実行されるよう、指定します。このパラメータが使用されると、照会前には入力文字列は拡張されません。
タブ	（任意）ユーザーが Tab キーを押したときにイベント照会が実行されるよう、指定します。このパラメータが使用されると、照会前には入力文字列は拡張されません。
mode	（任意）パーサーが指定されたパーサー モードの場合のみ、イベントが生成されます。使用可能なモードのリストは、**show parser dump** CLI コマンドを使用して表示できます。オプション パラメータの enter、questionmark、または tab のいずれか 1 つが指定されている場合、mode パラメータが確認されます。

maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

複数の条件が存在する場合、すべての条件が一致したときに、CLI イベントが発生します。

結果文字列

なし

_cerrno を設定

なし

（注） このポリシーは、CLI コマンドの実行前に実行されます。たとえば、 **copy** コマンドが入力されると、policy_CLI が実行のために登録されるとします。**copy** コマンドが入力されると、CLI イベントディテクタがパターンの一致を検出し、このポリシーの実行がトリガーされます。ポリシーの実行が終了すると、CLI イベントディテクタは、「sync」、「skip」（ポリシーで設定）、および、必要に応じてポリシー実行の終了ステータスに従って、**copy** コマンドを実行する必要があるかどうかを判断します。

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u msg {%s} msg_count %d line %u key %u tty %u error_code %u"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	イベントの重大度。
msg	CLI プロンプトで入力されるテキスト。

イベント タイプ	説明
msg_count	イベントがトリガーされる前にパターン マッチされた回数。
line	一致したキーが入力されたポイントまで、パーサーによって拡張できたテキスト。
key	Enter キー、疑問符、または Tab キー。
tty	ユーザーがコマンドを実行する行番号に対応します。
error_code	CLI のエラー コード。 0：パーサーからキーが入力されたポイントまで、エラーはありません。 1：キーが入力されたポイントまで、コマンドはあいまいです。 4：キーが入力されたポイントまで、未知のコマンドです。

event_register_counter

パブリッシャとサブスクライバの両方として、カウンタ イベントの登録を行います。この Tcl コマンド拡張を使用すると、しきい値に近くなった名前付きカウンタに基づいて、ポリシーが実行されます。サブスクライバとして、このイベント カウンタによって、登録に必要なカウンタの名前が指定され、別のポリシーまたは別のプロセスに依存して、カウンタが実際に操作されます。たとえば、policyB をカウンタ ポリシーとして動作させる一方、policyA（カウンタ ポリシーである必要はない）では、**register_counter**、**counter_modify**、または **unregister_counter** の各 Tcl コマンド拡張を使用して、policyB で定義されているカウンタを操作します。

構文

```
event_register_counter [tag ?] name ? entry_op gt|ge|eq|ne|lt|le entry_val ?
exit_op gt|ge|eq|ne|lt|le exit_val ? [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
name	（必須）カウンタの名前。
entry_op	（必須）現在のカウンタの値を開始値と比較するために使用される開始比較演算子。真の場合、イベントが発生し、終了基準を満たすまでイベント モニターリングがディセーブルにされます。

entry_val	（必須）カウンタ イベントを発生させる必要があるかどうかを判断するために、現在のカウンタの値を比較する必要がある値。
exit_op	（必須）現在のカウンタの値を終了値と比較するために使用される終了比較演算子。真の場合、このイベントのイベントモニターリングが再度イネーブルにされます。
exit_val	（必須）終了基準を満たすかどうかを判断するために、現在のカウンタの値を比較する必要がある値。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"name {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
name	カウンタ名。

event_register_gold

Generic Online Diagnostic（GOLD）障害イベントの登録を行います。この Tcl コマンド拡張を使用すると、指定されたカードおよびサブカードの Generic Online Diagnostic（GOLD）障害イベントに基づいて、ポリシーが実行されます。

構文

```
event_register_gold card all|card_number
[subcard all|subcard_number]
[new_failure TRUE|FALSE]
[severity_major TRUE]
[severity_minor TRUE]
[severity_normal TRUE]
[action_notify TRUE|FALSE]
[testing_type [bootup|ondemand|schedule|monitoring]]
[test_name [testname]]
[test_id [testnumber]]
[consecutive_failure consecutive_failure_number]
[platform_action [action_flag]]
[maxrun ?]
[queue_priority low|normal|high|last]
[nice 0|1]
```

引数

card	（必須）すべてのカードまたは 1 つのカードがモニターされるよう指定します。 • card all：すべてのカードを監視対象に指定します。これはデフォルトです。 • card-number：card-number の番号によって指定されたカードを監視対象に指定します。 **event_register_gold** Tcl コマンド拡張を完了させるには、この引数を指定する必要があります。
subcard	（任意）1 つまたは複数のサブカードがモニターされるよう指定します。 • subcard all：すべてのサブカードを監視対象に指定します。 • subcard-number：subcard-number の番号によって指定されたサブカードを監視対象に指定します。 この引数が指定されない場合、すべてのサブカードがデフォルトでモニターされます。
new_failure	（任意）GOLD からの新しいテスト障害情報に基づいて、イベント基準を指定します。 • new_failure TRUE：GOLD からの新しいテスト障害のイベント基準が真であると指定します。 • new_failure FALSE：GOLD からの新しいテスト障害のイベント基準が偽であると指定します。 この引数が指定されない場合、GOLD からの新しいテスト障害情報は、イベント基準で考慮されません。
severity_major	（任意）診断結果のイベント基準が、GOLD からの診断（メジャーエラー）と合致するよう指定します。
severity_minor	（任意）診断結果のイベント基準が、GOLD からの診断（マイナーエラー）と合致するよう指定します。
severity_normal	（任意）診断結果のイベント基準が、GOLD からの診断（通常）と合致するよう指定します。これはデフォルトです。

action_notify	（任意）GOLD からのアクション通知情報に基づいて、イベント基準を指定します。 • action_notify TRUE：GOLD からのアクション通知のイベント基準が真であると指定します。 • action_notify FALSE：GOLD からのアクション通知のイベント基準が偽であると指定します。 この引数が指定されない場合、GOLD からのアクション通知情報は、イベント基準で考慮されません。
testing_type	（任意）GOLD からの診断のテスト タイプに基づいて、イベント基準を指定します。 • testing_type bootup：システム ブート時に実行される診断テストを指定します。 • testing_type ondemand：カードがオンライン後に CLI から実行される診断テストを指定します。 • testing_type schedule：スケジュールされる診断テストを指定します。 • testing_type monitoring：システムの状態を監視するためにバックグラウンドで定期的に実行される診断テストを指定します。 この引数が指定されない場合、GOLD からのテスト タイプ情報は、イベント基準で考慮されず、ポリシーは、すべての診断テスト タイプに適用されます。
test_name	（任意）名前 test-name でのテストに基づいて、イベント基準を指定します。 • test_name test-name：名前 test-name でのテストに基づいて、イベント基準を指定します。 この引数が指定されない場合、GOLD からのテスト名情報は、イベント基準で考慮されません。

test_id	（任意）テスト ID に基づいて、イベント基準を指定します。 • test_id test-id：ID 番号 test-id のテストに基づいてイベント基準を指定します。test-id の最大値は 65535 です。 （注） テスト ID は、異なるラインカード上での同じテストについて、異なる可能性があるため、通常は、代わりに test_name キーワードを使用する必要があります。テスト ID が指定され、指定されたテスト名と矛盾する場合、テスト名によって、テスト ID が上書きされます。 この引数が指定されない場合、GOLD からのテスト ID 情報は、イベント基準で考慮されません。
consecutive_failure	（任意）GOLD からの連続テスト障害情報に基づいて、イベント基準を指定します。 • consecutive_failure consecutive-failure-number：イベント障害が、consecutive-failure-number 連続テスト障害の発生に基づくよう、指定します。 この引数が指定されない場合、GOLD からの連続テスト障害情報は、イベント基準で考慮されません。
platform_action	（任意）すべてのイベント基準が一致した場合に、プラットフォームへのコールバックが必要かどうかを指定します。コールバックが必要な場合、プラットフォームでは、指定されたレジストリを介してコールバック機能を登録する必要があります。 • platform_action action-flag-number：プラットフォームへのコールバックが必要な場合に、特定の情報がプラットフォーム特有の action-flag-number の値によって指定されるよう、指定します。action-flag-number の最大値は 65535 です。 （注） プラットフォームにより、フラグに基づいて行われる必要があるアクションが判断されます。 この引数が指定されない場合、コールバックはありません。
maxrun	（任意）スクリプトの最大実行時間を指定します。 • maxrun max-run-time-number：スクリプトの最大実行時間を、max-run-time-number 秒と指定します。max-run-time-number の最大値は 4294967295 秒です。 この引数が指定されない場合、デフォルトの実行時間は 20 秒です。

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
nice	（任意）次のような、ポリシー実行時間のプライオリティ設定。 • nice 0：ポリシーがデフォルトの実行時間優先度レベルで実行されるよう指定します。 • nice 1：ポリシーがデフォルト優先度レベルよりも低い実行時間優先度で実行されるよう指定します。 この引数が指定されない場合、デフォルトの実行時間プライオリティが使用されます。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u card %u sub_card %u"
"event_severity {%s} event_pub_sec %u event_pub_msec %u overall_result %u"
"new_failure {%s} action_notify {%s} tt %u tc %u bl %u ci %u pc %u cn {%s}"
"sn {%s} tn# {%s} ta# %s ec# {%s} rc# %u lf# {%s} tf# %u cf# %u tr# {%s}"
"tr#p# {%s} tr#d# {%s}"
```

イベント タイプ	説明
action_notify	TRUE または FALSE の、GOLD イベントでのアクション通知情報。
bl	起動診断レベル、次のいずれかの値である。 • 0：完全診断 • 1：最小診断 • 2：バイパス診断
card	GOLD イベントのカード情報。
cf *testnum*	連続的な障害。*testnum* はテスト番号。たとえば、**cf3** は、テスト 3 の連続的な障害の EEM 組み込み環境変数です。
ci	カード インデックス。
cn	カードの名前。
ec *testnum*	テスト エラー コード。*testnum* はテスト番号。たとえば、**ec3** は、テスト 3 のエラー コードの EEM 組み込み環境変数です。
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_pub_msec **event_pub_sec**	イベントが EEM にパブリッシュされたときの、ミリ秒単位および秒単位の時間。
event_severity	GOLD イベントの重大度で、次のいずれかの値を指定できます。 • normal • minor • major
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
lf *testnum*	最終障害時間。*testnum* はテスト番号。たとえば、**lf3** は、テスト 3 の最終障害時間の EEM 組み込み環境変数です。 タイムスタンプの形式は *mmm dd yyyy hh*:*mm*:*ss* です。例：Mar 11 1960 08:47:00。
new_failure	GOLD イベント フラグの新しいテスト障害情報（False または True）。

イベント タイプ	説明
overall_result	総合診断結果、次のいずれかの値である。 • 0：OK • 3：マイナー エラー • 4：メジャー エラー • 14：結果不明
pc	ポート数。
rc *testnum*	テスト総実行回数。*testnum* はテスト番号。たとえば、**rc3** は、テスト 3 の総実行回数の EEM 組み込み変数です。
sn	カード シリアル番号。
sub_card	GOLD 障害イベントが検出されたサブカード。
ta *testnum*	テスト属性名。*testnum* はテスト番号。たとえば、**ta3** は、テスト 3 の属性の EEM 組み込み環境変数です。
tc	テスト数。
tf *testnum*	合計障害回数。*testnum* はテスト番号。たとえば、**tf3** は、テスト 3 の合計障害回数の EEM 組み込み変数です。
tn *testnum*	テストの名前。*testnum* はテスト番号。たとえば、**tn3** は、テスト 3 の名前の EEM 組み込み環境変数です。
tr *testnum*	テストの結果。*testnum* はテスト番号。たとえば、**tr6** はテスト 6 用の EEM 組み込み変数です。テスト 6 はポート単位のテストでも、デバイス単位のテストでもありません。 テスト結果は、次の値のうちのいずれかです。 • P：診断結果 Pass • F：診断結果 Fail • U：診断結果 Unknown

イベント タイプ	説明
tr *testnum* **d** *devnum*	デバイスごとのテスト結果。*testnum* はテスト番号で、*devnum* はデバイス番号です。たとえば、**tr3d20** は、テスト 3、デバイス 20 のテスト結果の EEM 組み込み環境変数です。 テスト結果は、次の値のうちのいずれかです。 • P：診断結果 Pass • F：診断結果 Fail • U：診断結果 Unknown
tr *testnum* **p** *portnum*	ポートごとのテスト結果。*testnum* はテスト番号で、*portnum* はデバイス番号です。たとえば、**tr5p20** は、テスト 5、ポート 20 のテスト結果の EEM 組み込み環境変数です。 テスト結果は、次の値のうちのいずれかです。 • P：診断結果 Pass • F：診断結果 Fail • U：診断結果 Unknown
tt	テストのタイプ。次のうちのいずれかです。 • 1：起動診断 • 2：オンデマンド診断 • 3：スケジュール診断 • 4：モニターリング診断

event_register_identity

ID イベントの登録を行います。この Tcl コマンド拡張を使用して、AAA 認証または許可が成功または失敗したときや、ポート上での通常のユーザー トラフィックのフローが許可された後にイベントを生成します。

構文

```
event_register_identity [tag ?] interface ?
[aaa-attribute ?]
[authc {all | fail | success}]
[authz {all | fail | success}]
[authz-complete]
[mac-address ?]
```

```
[queue_priority {normal | low | high | last}]
[maxrun ?] [nice {0 | 1}]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
interface	インターフェイス名と照合する正規表現パターン。
aaa-attribute	（任意）特定の AAA 属性によってイベントをフィルタリングするために使用可能な正規表現。
authc	（任意）成功した認証、失敗した認証、または成功と失敗の両方の認証で、イベントをトリガーします。
authz	（任意）成功した許可、失敗した許可、または成功と失敗の両方の許可で、イベントをトリガーします。
authz-complete	（任意）インターフェイスに接続されたデバイスが完全に認証、許可され、通常のトラフィックがそのインターフェイスで流れ始めたときにイベントをトリガーします。
mac-address	（任意）リモートデバイスの MAC アドレスによってイベントをフィルタリングするために使用可能な正規表現パターン。
maxrun	（任意）スクリプトの最大実行時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 31536000 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

EEM_EVENT_IDENTITY の Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u identity_stage %u identity_status %u interface %u identity_mac %u
identity_<attribute> {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。

イベント タイプ	説明
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	イベントの重大度。
identity_stage	authentication、authentication、または autherization-complete のステージのうちのいずれか。
identity_status	Success または fail_authc、fail_aaa_server、fail_no_response、fail_timeout、fail_authz のいずれかの障害タイプ。 autherization-complete は常に success になります。
interface	イベントのインターフェイス。
identity_mac	イベントのリモート デバイスの MAC アドレス。
identity_<attribute>	属性リストまたは値リスト内のその AAA 属性に対応する値に対する AAA 属性ごとの一連のダイナミック変数。

event_register_interface

インターフェイス カウンタ イベントの登録を行います。この Tcl コマンド拡張を使用すると、指定されたインターフェイス カウンタが指定されたしきい値を超えたときに、イベントが生成されます。

構文

```
event_register_interface [tag ?] name ?
parameter ? entry_op gt|ge|eq|ne|lt|le
entry_val ? entry_val_is_increment TRUE|FALSE
entry_type value|increment|rate
[exit_comb or|and]
[exit_op gt|ge|eq|ne|lt|le]
[exit_val ?] [exit_val_is_increment TRUE|FALSE]
[exit_type value|increment|rate]
[exit_time ?] [poll_interval ?]
[average_factor ?] [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。

name	（必須）イーサネット 0/0 など、モニターされるインターフェイスの名前。省略形と空白は使用できません。
parameter	（必須）比較されるカウンタの名前は、次のとおりです。 • input_errors：ラント、ジャイアント、バッファなし、CRC、フレーム、オーバーラン、および無視されたカウントが含まれます。他の入力関連のエラーも、入力エラー カウントが大きくなる場合があります。一部のデータグラムには、複数のエラーがあります。したがって、この合計は、列挙型入力エラー カウントとのバランスが取れない場合があります。 • input_errors_crc：発信元 LAN ステーションまたは遠隔エンド デバイスによって生成される巡回冗長検査が、受信したデータから計算されるチェックサムに一致しません。 • input_errors_frame：受信した不正確なパケット数。CRC エラーが発生し、8 ビットの非整数の数です。 • input_errors_overrun：入力レートが、レシーバのデータ処理能力を超えたために、レシーバ ハードウェアによって、受信データをハードウェア バッファに渡せなかった回数。 • input_packets_dropped：入力キューがいっぱいのため、廃棄されたパケット数。 • interface_resets：インターフェイスが完全にリセットされた回数。 • output_buffer_failures：障害が発生したバッファ数およびスワップされたバッファ数。 • output_buffer_swappedout：DRAM にスワップされたパケット数。

parameter（続き）	• output_errors：調べられているインターフェイスからのデータグラムの最終的な送信が妨害されたすべてのエラーの合計。一部のデータグラムには、複数のエラーがある場合があり、また、他のデータグラムには、特に表形式のカテゴリに当てはまらないエラーがある場合があるため、これは、列挙型出力エラーの合計とのバランスが取れないことがあります。 • output_errors_underrun：トランスミッタが、デバイスが処理可能な速度よりも高速だった回数。 • output_packets_dropped：出力キューがいっぱいのため、廃棄されたパケット数。 • receive_broadcasts：インターフェイスによって受信されたブロードキャスト パケットまたはマルチキャスト パケットの数。 • receive_giants：メディアの最大パケット サイズを超過したために廃棄されたパケット数。 • receive_rate_bps：1 秒あたりのバイト単位でのインターフェイス受信レート。 • receive_rate_pps：1 秒あたりのパケット単位でのインターフェイス受信レート。 • receive_runts：メディアの最小パケット サイズよりも小さいために廃棄されたパケット数。 • receive_throttle：バッファまたはプロセッサが過負荷などの理由で、ポート上のレシーバが無効にされた回数。 • reliability：5 分間の幾何平均で計算される、255 の分数でのインターフェイスの信頼性（255/255 が 100% の信頼性）。 • rxload：255 の分数でのインターフェイスの受信レート（255/255 が 100%）。 • transmit_rate_bps：1 秒あたりのバイト単位でのインターフェイス送信レート。 • transmit_rate_pps：1 秒あたりのパケット単位でのインターフェイス送信レート。 • txload：255 の分数でのインターフェイスの送信レート（255/255 が 100%）。
entry_op	（必須）現在のインターフェイスの値を開始値と比較するために使用される比較演算子。真の場合、イベントが発生し、終了基準を満たすまでイベント モニターリングがディセーブルにされます。
entry_val	（必須）イベントがトリガーされる値。

entry_val_is_increment	（必須）TRUE の場合、entry_val フィールドは増分差異として処理され、現在のカウンタの値とイベントが最後に真であったとき（これが新しいイベントの場合は最初にポーリングされたサンプル）の値との差異と、比較されます。負の値によって、減少しているカウンタの増分差異がチェックされます。FALSE の場合、entry_val フィールドが現在のカウンタの値に対して比較されます。 （注） このキーワードは廃止されました。これを指定した場合、その構文は同等な entry-type キーワード構文に変換されます。
entry-type	entry-val 引数によって指定されたオブジェクト ID に適用される操作のタイプを指定します。 値は、entry-val 引数の実際の値として定義されます。 増分では、entry-val フィールドは増分差異として使用され、entry-val は、現在のカウンタの値と、イベントが最後に真であったとき（これが新しいイベントの場合は最初にポーリングされたサンプル）の値との間の差と、比較されます。負の値によって、減少しているカウンタの増分差異がチェックされます。 レートは、ある期間の変更の平均レートとして定義されます。期間は、average-factor の値に、poll-interval の値を乗じたものです。ポーリング間隔ごとに、現在のサンプルと前のサンプルとの間の差が取得され、絶対値として記録されます。前の average-factor 値サンプルの平均は、変更のレートとして取得されます。
exit_comb	（任意）イベント トリガーの再準備に必要な終了条件テストの組み合わせを示すために使用されます。and 演算子が指定される場合、再準備のためには、終了値と終了時間テストの両方が真である必要があります。or 演算子が指定される場合、イベント モニターリングの再準備のためには、終了値または終了時間テストのいずれかが真である可能性があります。
exit_op	（任意）現在のインターフェイスの値を終了値と比較するために使用される比較演算子。真の場合、このイベントのイベント モニターリングが再度イネーブルにされます。
exit_val	（任意）イベントが再度監視されるように再準備される値。
exit_val_is_increment	（任意）TRUE の場合、exit_val フィールドは増分差異として処理され、現在のカウンタの値と、イベントが最後に真であったときの値との差異と、比較されます。負の値によって、減少しているカウンタの増分差異がチェックされます。FALSE の場合、exit_val フィールドが現在のカウンタの値に対して比較されます。 （注） Cisco IOS Release 12.4(20)T では、このキーワードは廃止予定で、指定された場合、構文は同等の exit-type キーワード構文に変換されます。

exit-type	（任意）exit-val 引数によって指定されたオブジェクト ID に適用される操作のタイプを指定します。指定されない場合、値が仮定されます。 値は、exit-val 引数の実際の値として定義されます。 増分では、exit-val フィールドは増分差異として使用され、exit-val は、現在のカウンタの値と、イベントが最後に真であったとき（これが新しいイベントの場合は最初にポーリングされたサンプル）の値との間の差と、比較されます。負の値によって、減少しているカウンタの増分差異がチェックされます。 レートは、ある期間の変更の平均レートとして定義されます。期間は、average-factor の値に、poll-interval の値を乗じたものです。ポーリング間隔ごとに、現在のサンプルと前のサンプルとの間の差が取得され、絶対値として記録されます。前の average-factor 値サンプルの平均は、変更のレートとして取得されます。
exit_time	（任意）イベントが再度監視されるように再準備される時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。
poll_interval	（任意）サンプルが収集される頻度（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、60 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。ポーリング間隔の値には、1 秒よりも小さい値は使用できません。デフォルト値は 1 秒です。
average-factor	（任意）レート ベースの計算に使用される期間の計算に使用される 1 から 64 の範囲の数。average-factor の値は、poll-interval の値を乗じた値で、ミリ秒単位で導き出される期間です。最少平均係数値は 1 です。

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event_severity {%s} name {%s} parameter {%s} value %d"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	インターフェイスイベントの重大度で、次のいずれかの値を指定できます。 • normal • minor • major
name	インターフェイスの名前。
parameter	パラメータの名前。
value	指定された entry_val_is_increment によって、トリガーされた最後のイベントに対する増加または減少の差異、または、監視されているパラメータの絶対値。

event_register_ioswdsysmon

IOSWDSysMon イベントの登録を行います。この Tcl コマンド拡張を使用すると、Cisco IOS タスクが指定された CPU 使用率またはメモリしきい値を超えたときに、イベントが生成されます。Cisco IOS タスクは、ネイティブ Cisco IOS の Cisco IOS プロセスと呼ばれます。

構文

```
event_register_ioswdsysmon [tag ?] [timewin ?] [sub12op and|or] [sub1 ?] [sub2 ?]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。

timewin	（任意）イベントが生成されるようにするために、すべてのサブイベントが発生する必要がある時間ウィンドウを定義します（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。
sub12_op	（任意）サブイベント 1 とサブイベント 2 とを比較する組み合わせ演算子。
sub1	（任意）サブイベント 1 の指定。
sub2	（任意）サブイベント 2 の指定。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

サブイベントの構文

```
cpu_proc path ? taskname ? op gt|ge|eq|ne|lt|le val ? [period ?]
mem_proc path ? taskname ? op gt|ge|eq|ne|lt|le val ? [is_percent TRUE|FALSE] [period
?]
```

サブイベントの引数

cpu_proc	（必須）CPU 統計情報のサンプル収集の使用を指定します。
path	（必須）ソフトウェア モジュール方式イメージのみ。監視される Cisco IOS スケジューラが含まれる POSIX プロセスのパス名。たとえば、/sbin/cdp2.iosproc など。
taskname	（必須）監視される Cisco IOS タスクの名前。
op	（必須）収集される使用サンプルを指定値と比較するために使用される比較演算子。真の場合、このイベントが発生します。
val	（必須）比較される値。
period	（任意）収集されるサンプルの平均が計算される経過時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合は、最新のサンプルが使用されます。
mem_proc	（必須）メモリ統計情報のサンプル収集の使用を指定します。
is_percent	（任意）指定値がパーセンテージかどうか。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"num_subs %u"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
num_subs	サブイベントの番号。

サブイベント情報文字列は、次のような、CPU_UTIL サブイベント用です。

```
"{type %s procname {%s} pid %u taskname {%s} taskid %u value %u sec %ld msec %ld}"
```

サブイベントタイプ	説明
type	サブイベントのタイプ。
procname	このサブイベントの POSIX プロセス名。
pid	このサブイベントの POSIX プロセス ID。
taskname	このサブイベントの Cisco IOS タスク名。
taskid	このサブイベントの Cisco IOS タスク ID。
value	測定された間隔での、実際の平均CPU使用率。
sec , **msec**	この測定間隔の経過時間。

サブイベント情報文字列は、次のような、MEM_UTIL サブイベント用です。

```
"{type %s procname {%s} pid %u taskname {%s} taskid %u is_percent %s value %u diff %d"
"sec %ld msec %ld}"
```

サブイベントタイプ	説明
type	サブイベントのタイプ。
procname	このサブイベントの POSIX プロセス名。
pid	このサブイベントの POSIX プロセス ID。
taskname	このサブイベントの Cisco IOS タスク名。
taskid	このサブイベントの Cisco IOS タスク ID。
is_percent	値がパーセント値かどうかによって、TRUE または FALSE。
value	この測定された間隔の KB 単位でのメモリ使用量の合計、または実際のメモリ使用率の平均。
diff	測定された間隔で最も古いサンプルと、最新のサンプルとの、パーセンテージでの違い。負の値は、減少を表します。
sec , **msec**	この測定間隔の経過時間。

event_register_ipsla

event ipsla コマンドによってトリガーされるイベントの登録を行います。この Tcl コマンド拡張を使用すると、IPSLA の応答がトリガーされるときに、イベントがパブリッシュされます。イベントの登録には、グループ ID または動作 ID が必要です。

構文

```
event_register_ipsla [tag ?] group_name ? operation_id ? [reaction_type ?]
[dest_ip_addr ?][queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
group_name	（必須）IP SLA グループ名を指定します。
operation_id	（必須）IP SLA 動作 ID を指定します。番号は 1 から 2147483647 の範囲の整数です。

reaction_type	（任意）指定した IP SLA 動作に対する応答を指定します。 IP SLA 反応のタイプ：次のキーワードのいずれかを指定できます：**connectionLoss**、 **icpif**、 **jitterAvg**、 **jitterDSAvg**、 **jitterSDAvg**、 **maxOfNegativeDS**、 **maxOfNegativeSD**、 **maxOfPositiveDS**、 **maxOfPositiveSD**、 **mos**、 **packetLateArrival**、 **packetLossDS**、 **packetLossSD**、 **packetMIA**、 **packetOutOfSequence**、 **rtt**、 **timeout** または **verifyError** を指定できます。 IP SLA の応答。次のキーワードの 1 つを指定できます。 • connectionLoss • icpif • jitterAvg • jitterDSAvg • jitterSDAvg • maxOfNegativeDS • maxOfNegativeSD • maxOfPositiveDS • maxOfPositiveSD • mos • packetLateArrival • packetLossDS • packetLossSD • packetMIA • packetOutOfSequence • rtt • timeout • verifyError
dest_ip_address	（任意）IP SLA イベントが監視される宛先ポートの宛先 IP アドレスを指定します。

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大実行時間（SSSSSSSSSS[.MMM]形式で指定します。SSSSSSSSSS は、0 ～ 31536000 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

"event_ID %u event_type %u event_pub_sec %u event_pub_msec %u event_severity %u" "group_name %u operation_id %u condition %u reaction_type %u dest_ip_addr %u" "threshold_rising %u threshold_falling%u measured_threshold_value %u" "threshold_count1 %u threshold count2 %u"

Event Type	Description
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	フローの作成、アップデート、および削除を監視するイベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	イベントの重大度。
group_name	IPSLA グループの名前。
operation_id	IPSLA 動作 ID。
condition	IPSLA の条件で、次の 1 つを使用できます。 • cleared • occurred
reaction_type	IPSLA 応答タイプ。
dest_ip_address	IPSLA 宛先 IP アドレス。
threshold rising	IPSLA で設定されている上昇しきい値。
threshold falling	IPSLA で設定されている下限しきい値。
measured_threshold_value	IPSLA 動作の測定されたしきい値。
threshold_count1	しきい値 type1 の引数に対応します。
threshold_count2	しきい値 type2 の引数に対応します。

event_register_mat

MAT イベントの登録を行います。この Tcl コマンド拡張を使用して、mac-address-table で MAC アドレスが学習されたときにイベントを生成します。

構文

```
event_register_identity [tag ?] interface ?
```

```
[mac-address ?]
[type {add | delete}]
[hold-down ?]
[maxrun ?]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
interface	インターフェイス名と照合する正規表現パターン。
mac-address	インターフェイス パラメータを指定していない場合には必須です。リモート デバイスの MAC アドレスによってイベントをフィルタリングするために使用可能な正規表現パターン。
type	（任意）追加または削除の mac-address-table イベント タイプに基づいてフィルタリングします。指定しなかった場合、イベントをトリガーするかどうかの判断にそのイベント タイプが使用されません。
hold-down	（任意）mac-address-table イベントが着信した場合、ポリシーを処理する前にそのイベントを 1 ～ 4294967295 秒間待機させるようにホールドダウン タイマーを設定できます。このタイマーを設定しなかった場合は、ポリシーの処理は遅延しません。
maxrun	（任意）スクリプトの最大実行時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 31536000 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。

結果文字列

なし

_cerrno を設定

なし

EEM_EVENT_MAT の Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u notification %u intf_name %u mac_address {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。

イベント タイプ	説明
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	イベントの重大度。
notification	通知のタイプ：追加または削除。
intf_name	アドレス テーブル エントリのインターフェイス名。
mac_address	アドレス テーブルのエントリの MAC アドレス。

event_register_neighbor_discovery

ネイバー探索イベントの登録この Tcl コマンド拡張を使用して、Cisco Discovery Protocol（CDP）または Link Layer Discovery Protocol（LLDP）のキャッシュ エントリまたはインターフェイス リンク ステータスが変わった場合にイベントを生成します。

構文

```
event_register_neighbor_discovery [tag ?] interface ?
[cdp {add | update | delete | all}]
[lldp {add | update | delete | all}]
[link-event]
[line-event]
[queue_priority {normal | low | high | last}]
[maxrun ?] [nice {0 | 1}]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
interface	インターフェイス名と照合する正規表現パターン。

cdp	CDP のマッチング イベント発生時にイベントをトリガーします。次のオプションのいずれかを指定する必要があります。 • add：新しい CDP キャッシュ エントリが CDP テーブルに作成された場合にイベントをトリガーします。 • all：CDP キャッシュ エントリが CDP キャッシュ テーブルに追加された場合、または削除された場合、および CDP キャッシュ エントリをアップデートするためにリモート CDP デバイスがキープアライブを送信した場合にイベントをトリガーします。 • delete：CDP キャッシュ エントリが CDP テーブルから削除された場合だけイベントをトリガーします。 • update：CDP キャッシュ エントリが CDP テーブルに追加された場合、または CDP キャッシュ エントリをアップデートするためにリモート CDP デバイスが CDP キープアライブを送信した場合にイベントをトリガーします。
lldp	LLDP のマッチング イベント発生時にイベントをトリガーします。次のオプションのいずれかを指定する必要があります。 • add：新しい CDP キャッシュ エントリが CDP テーブルに作成された場合にイベントをトリガーします。 • all：CDP キャッシュ エントリが CDP キャッシュ テーブルに追加された場合、または削除された場合、および CDP キャッシュ エントリをアップデートするためにリモート CDP デバイスがキープアライブを送信した場合にイベントをトリガーします。 • delete：CDP キャッシュ エントリが CDP テーブルから削除された場合だけイベントをトリガーします。 • update：CDP キャッシュ エントリが CDP テーブルに追加された場合、または CDP キャッシュ エントリをアップデートするためにリモート CDP デバイスが CDP キープアライブを送信した場合にイベントをトリガーします。
line-event	インターフェイス回線プロトコルのステータスが変わった場合にイベントをトリガーします。
link-event	インターフェイス リンクのステータスが変わった場合にイベントをトリガーします。

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大実行時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 31536000 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

EEM_EVENT_NEIGHBOR_DISCOVERY の Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u nd_notification {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントのIDを示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec　event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	イベントの重大度。
共通の Event_Reqinfo	
nd_notification	通知のタイプ：cdp-add、cdp-update、cdp-delete、lldp-add、lldp-update、lldp-delete、link、line。
nd_intf_linkstatus	現在のインターフェイス リンクのステータス。up または down。
nd_intf_linestatus	現在のインターフェイス回線のステータス。down、goingdown、init、testing、up、reset、admindown、deleted。
nd_local_intf_name	イベントのローカル インターフェイスの名前。
nd_short_local_intf_name	イベントのローカル インターフェイスの短い名前。
nd_port_id	CDP プロトコルまたは LLDP プロトコルのいずれかで識別されたポート ID。これは、リンクまたは回線プロトコルのイベントには設定されません。
CDP-specific Event_reqinfo	
nd_protocol	イベントをトリガーしたプロトコルを識別します。CDP の場合は常に cdp に設定されます。
nd_proto_notif	イベント、追加、更新、または削除をトリガーしたプロトコル イベントのタイプを特定します。
nd_proto_new_entry	1 に設定されている場合、キャッシュ エントリは新規であるため、イベントはトリガーされており、それ以外の場合は 0 に設定されます。
nd_cdp_entry_name	CDP テーブル内の CDP キャッシュ エントリの名前。

イベント タイプ	説明
nd_cdp_hold_time	CDP キャッシュ エントリが期限切れになり、CDP テーブルから削除されるまでの残り時間。この時間は、CDP ネイバーからの更新によって最大値にリセットされます。新しいエントリの場合は通常、0 に設定されます。
nd_cdp_mgmt_domain	CDP VTP 管理ドメイン。
nd_cdp_platform	リモート デバイスによって報告されるプラットフォームの名前。
nd_cdp_version	リモートデバイスで実行されているコードのバージョン。
nd_cdp_capabilities_string	文字列形式の CDP capabilities フィールドのコンテンツ：ルータ、トランスブリッジ、ソースルートブリッジ、スイッチ、ホスト、IGMP、リピータ、電話、リモートで管理されているデバイス、CVTA 電話ポート、2 ポート MAC リレー、または、カンマで区切ったこれらの組み合わせ。
nd_cdp_capabilities_bits	先頭に 0x が付加された 16 進数内の CDP 機能ビット。
nd_cdp_capabilities_bit_[0-31]	capabilities フィールドのそのビットが設定されている場合は YES に、設定されていない場合は NOT に設定される一連の値。
LLDP-specific Event_reqinfo	
nd_protocol	イベントをトリガーしたプロトコルを識別します。LLDP の場合は常に lldp に設定されます。
nd_proto_notif	イベント、追加、更新、または削除をトリガーしたプロトコル イベントのタイプを特定します。
nd_proto_new_entry	1 に設定されている場合、キャッシュ エントリは新規であるため、イベントはトリガーされており、それ以外の場合は 0 に設定されます。
nd_lldp_chassis_id	LLDP キャッシュ エントリからの chassis id フィールド。
nd_lldp_system_name	LLDP キャッシュ エントリからのシステム名。
nd_lldp_system_description	LLDP キャッシュ エントリからの system description フィールド。
nd_lldp_ttl	LLDP キャッシュ エントリからの LLDP time to live フィールド。
nd_lldp_port_description	LLDP キャッシュ エントリからの port description フィールド。

イベント タイプ	説明
nd_lldp_system_capabilities_string	LLDP キャッシュ エントリからの LLDP system capabilities フィールド。この文字列には、O、P、B、W、R、T、C、S、またはこれらの任意の組み合わせをカンマで区切って含めることができます。
nd_lldp_enabled_capabilities_string	LLDP キャッシュ エントリからの LLDP enabled system capabilities フィールド。この文字列には、O、P、B、W、R、T、C、S、またはこれらの任意の組み合わせをカンマで区切って含めることができます。
nd_lldp_system_capabilities_bits	LLDP キャッシュ エントリからの LLDP system capabilities bits フィールド。先頭に 0x が付加された 16 進数です。
nd_lldp_enabled_capabilities_bits	LLDP キャッシュ エントリからの LLDP enabled capabilities bits フィールド。先頭に 0x が付加された 16 進数です。
nd_lldp_capabilities_bits	LLDP キャッシュ エントリからの LLDP capabilities bits フィールド。先頭に 0x が付加された 16 進数です。
nd_lldp_capabilities_bit_[0-31]	capabilities フィールドのそのビットが設定されている場合は YES に、設定されていない場合は NOT に設定される一連の値。

event_register_nf

NetFlow イベントが **event nf** コマンドによってトリガーされるときのイベントの登録を行います。この Tcl コマンド拡張を使用すると、NetFlow の応答がトリガーされるときに、イベントがパブリッシュされます。

構文

```
event_register_nf [tag ?] monitor_name ? event_type create|update|delete
exit_event_type create|update|delete event1-event4 ? [maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
monitor_name	（必須）NetFlow モニターの名前。
event_type	（必須）フローの作成、アップデート、および削除を監視するイベントのタイプ。

exit_event_type	（必須）監視のためにイベントが再準備されるイベントタイプ（create、delete、update）。
event1- event4	（必須）監視するイベントとその属性を指定します。有効な値は **event1**、**event2**、**event3**、および **event4** です。 サブイベント キーワードは、単独でも、一緒でも、それぞれの任意の組み合わせでも使用できますが、各キーワードは 1 回のみ使用できます。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice引数が1に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

サブイベントの構文

```
field ? rate_interval ? event1 only entry_value ? entry_op eq|ge|gt|le|lt|wc
[exit_value ?] [exit_op eq|ge|gt|le|lt|wc] [exit_rate_interval ? event1 only]
```

サブイベントの引数

field	（必須）監視されるキャッシュまたはフィールド属性を指定します。次の属性の 1 つを指定できます。 • **counter** {**bytes** | **packets**}：カウンタフィールドを指定します。 • **datalink** {**dot1q** | **mac**}：データリンク（レイヤ 2）フィールドを指定します。 • **flow** {**direction** | **sampler**}：フロー識別フィールドを指定します。 • **interface** {**input** | **output**}：インターフェイスフィールドを指定します。 • **ipv4** *field-type*： IPv4 フィールドを指定します。 • **ipv6** *field-type*： IPv6 フィールド • **routing** *routing-attrribute* **--** ：ルーティング属性を指定します。 • **timestamp sysuptime** {**first** | **last**}**--**：タイムスタンプ フィールドを指定します。 • **transport** *field-type*： トランスポート レイヤ フィールドを指定します。
rate_interval	（必須）レートの計算に使用されるレート間隔値を秒単位で指定します。このフィールドは、event1 でのみ有効です。

entry_value	（必須）フィールドまたはレートの値を指定します。
entry_op	（必須）フィールド演算子を指定します。 次の比較演算子の値が有効です。 • **eq** ：次の値と等しい • **ge** ：次の値以上 • **gt** ：次の値より大きい • **le** ：次の値以下 • **lt** ：次の値より小さい • **wc** ：ワイルドカード
exit_value	（任意）イベントが再度監視されるように再準備される値。
exit_op	（任意）現在のイベントフィールドまたはレートの値を終了値と比較するために使用される比較演算子。真の場合、このイベントのイベント監視が再度イネーブルにされます。 次の比較演算子の値が有効です。 • **eq** ：次の値と等しい • **ge** ：次の値以上 • **gt** ：次の値より大きい • **le** ：次の値以下 • **lt** ：次の値より小さい • **wc** ：ワイルドカード
exit_rate_interval	（任意）終了レート値の計算に使用される終了レート間隔値を秒単位で指定します。このフィールドは、event1 でのみ有効です。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_ID %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u monitor_name %u event1-event4_field %u event1-event4_value
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	フローの作成、アップデート、および削除を監視するイベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	NetFlow イベントの重大度。
montior_name	NetFlow モニターの名前。
event1-event4_field	監視するイベントとその属性を指定します。有効な値は **event1**、**event2**、**event3**、および **event4** です。
event1-event4_value	監視するイベント値とその属性を指定します。有効な値は **event1**、**event2**、**event3**、および **event4** です。

event_register_none

event manager run コマンドによってトリガーされるイベントの登録を行います。これらのイベントは、このイベントをスクリーニングする None イベント ディテクタによって処理されます。

構文

```
event_register_none [tag ?] [sync {yes|no}] [default ?] [queue_priority
low|normal|high|last] [maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
sync	（任意）このキーワードを完了するには、「yes」または「no」が必要です。 • yes キーワードが指定されている場合、ポリシーは、CLI コマンドと同期的に実行されます。 • no キーワードが指定されている場合、ポリシーは、CLI コマンドと非同期的に実行されます。

デフォルト	（任意）CLI イベント ディテクタがポリシーの終了を待つ時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。ポリシーが終了する前にデフォルトの時間の期限が切れると、デフォルト アクションが実行されます。デフォルト アクションによって、コマンドが実行されます。この引数が指定されない場合、デフォルトの時間は 30 秒に設定されます。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u arg %u"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントのIDを示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec event_pub_msec	EEMに対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	イベントの重大度。
argc **arg1** **arg2** **arg3** **arg4** **arg6** **arg7** **arg8** **arg9** **arg10** **arg11** **arg12** **arg13** **arg14** **arg15**	Extensible Markup Language（XML）Simple Object Access Protocol（SOAP）コマンドからスクリプトに渡されるパラメータ。

event_register_oir

活性挿抜（OIR）イベントの登録を行います。このTclコマンド拡張を使用すると、ハードウェア カードOIRイベントの発生時に発生するイベントに基づいて、ポリシーが実行されます。

これらのイベントは、このイベントをスクリーニングする OIR イベント ディテクタによって処理されます。

構文

```
event_register_oir [tag ?] [queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"slot %u event %s"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一のイベント ID を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
slot	影響が及ぼされるカードのスロット番号。
event	OIR の削除イベントまたは OIR の挿入イベントを表す、removed または online の文字列を示します。

event_register_process

プロセス イベントの登録を行います。この Tcl コマンド拡張を使用すると、Cisco IOS ソフトウェア モジュール方式プロセスの開始時と停止時に発生するイベントに基づいて、ポリシーが実行されます。これらのイベントは、このイベントをスクリーニングする System Manager イベント ディテクタによって処理されます。この Tcl コマンド拡張は、ソフトウェア モジュール方式イメージでのみサポートされます。

構文

```
event_register_process [tag ?] abort|term|start|user_restart|user_shutdown
[sub_system ?] [version ?] [instance ?] [path ?] [node ?]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。

abort	（必須）プロセスの異常な終了。ゼロではない終了ステータスでの終了、カーネル生成信号の受信、またはユーザー要求のために送信されない SIGTERM 信号または SIGKILL 信号の受信のため、プロセスが強制終了されることがあります。
term	（必須）プロセスの正常な終了。
start	（必須）プロセスの開始。
user_restart	（必須）CLI コマンドからのプロセスの再起動要求が原因でプロセスを終了。
user_shutdown	（必須）CLI コマンドからのプロセスの終了要求が原因でプロセスを終了。
sub_system	（任意）プロセス イベントをパブリッシュした EEM ポリシーに割り当てられる番号。他のすべての番号は Cisco での使用のために予約されているため、番号は 798 に設定されます。
version	（任意）バージョンマネージャによって割り当てられるプロセスのバージョン番号。major_number.minor_number.level の形式である必要があります。指定される場合、バージョン番号の各コンポーネントは、1 ～ 4294967295 の範囲の整数である必要があります。
instance	（任意）プロセス インスタンス ID。指定される場合、この引数は、1 ～ 4294967295 の範囲の整数である必要があります。
path	（任意）プロセス パス名（正規表現文字列）。process-name 引数の値に空白文字が含まれている場合、二重引用符で囲む必要があります。すべてのプロセスを照合するには、パス「.*」を使用します。
ノード	（任意）ノード名は、「node」という語句と、それに続く、次の形式を使用してスラッシュ文字で区切られた 2 つのフィールドで構成される、文字列です。 node<slot-number>/<cpu-number> slot-number は、ハードウェア スロット番号です。cpu-number は、ハードウェア CPU 番号です。たとえば、スロット 0 にある Cisco Catalyst 6500 シリーズ スイッチのスーパーバイザ カードの SP CPU は、node0/0 と指定されます。たとえば、スロット 0 にある Cisco Catalyst 6500 シリーズ スイッチのスーパーバイザ カードの RP CPU は、node0/1 と指定されます。node 引数が指定されない場合、デフォルトのノード指定は、常に、すべての該当するノードを表す正規表現パターン マッチ * です。

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

任意の引数が指定されない場合、イベントは、引数のすべての可能な値に対して照会されます。複数の引数が存在する場合、すべての条件が一致したときに、プロセスイベントが発生します。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"sub_system 0x%x instance %u process_name {%s} path {%s} exit_status 0x%x"
```

```
"respawn_count %u last_respawn_sec %ld last_respawn_msec %ld fail_count %u"
"dump_count %u node_name {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントのIDを示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEMに対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
sub_system	アプリケーション固有のイベントをパブリッシュしたEEMポリシーに割り当てられる番号。他のすべての番号はCiscoでの使用のために予約されているため、番号は 798 に設定されます。
instance	プロセス インスタンス ID。
process_name	プロセス名。
path	パスを含むプロセスの絶対名。
exit_status	プロセスの最後の終了ステータス。
respawn_count	プロセスが再起動された回数。
last_respawn_sec **last_respawn_msec**	最後の再起動が発生したカレンダー時間。
fail_count	失敗したプロセスの再起動試行の回数。プロセスが正常に再起動されると、0 にリセットされます。
dump_count	プロセスで取られたコア ダンプの数。
node_name	プロセスが存在するノードの名前。ノード名は、「node」という語句と、それに続く、次の形式を使用してスラッシュ文字で区切られた 2 つのフィールドで構成される、文字列です。 **node** *slot-number* / *cpu-number* slot-number は、ハードウェア スロット番号です。cpu-number は、ハードウェア CPU 番号です。

event_register_resource

Embedded Resource Manager（ERM）イベントの登録を行います。この Tcl コマンド拡張を使用すると、指定されたポリシーの ERM イベント レポートに基づいて、ポリシーが実行されます。ERM イベントは、EEM リソース イベントによってスクリーニングされ、これによって、指定された ERM ポリシーへの一致が発生したときに、EEM ポリシーを実行できます。

構文

```
event_register_resource policy policy-name [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

引数

ポリシー	（必須）ポリシーの使用を指定します。
policy-name	（必須）ERM ポリシーの名前。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。

nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"owner_id %lld user_id %lld" time_sent %llu dampen_time %d notify_data_flags %u"
"level {%s} direction {%s} configured_threshold %u current_value %u"
"policy_violation_flag {%s} policy_id %d"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベントタイプのイベントの名前を表すASCII 文字列。
event_pub_sec event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
owner_id	Embedded Resource Manager（ERM）オーナー ID。
user_id	ERM ユーザー ID。
time_sent	ERM イベント時間、ナノ秒単位。
dampen_time	ERM 減衰時間、ナノ秒単位。
notify_data_flags	ERM 通知データ フラグ。
level	ERM イベント レベル。イベント レベルは、Normal、Minor、Major、および Critical の 4 つです。
direction	ERM イベント方向。イベント方向は、アップ、ダウン、または、変更なしのうちのいずれかです。
configured_threshold	設定されている ERM しきい値。
current_value	ERM によって報告された、現在の値。

イベント タイプ	説明
policy_violation_flag	ERM ポリシー違反フラグ（False または True）。
policy_id	ERM ポリシー ID。

event_register_rf

冗長ファシリティ（RF）イベントの登録を行います。この Tcl コマンド拡張を使用すると、RF の進行またはステータス イベントの通知が発生したときに、ポリシーが実行されます。

構文

```
event_register_rf [tag ?] event ?
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。

event	（必須）RF の進行またはステータスイベントの名前。有効な値は次のとおりです。 • RF_PROG_ACTIVE • RF_PROG_ACTIVE_DRAIN • RF_PROG_ACTIVE_FAST = 200 • RF_PROG_ACTIVE_PRECONFIG • RF_PROG_ACTIVE_POSTCONFIG • RF_PROG_EXTRALOAD • RF_PROG_HANDBACK • RF_PROG_INITIALIZATION • RF_PROG_PLATFORM_SYNC • RF_PROG_STANDBY_BULK • RF_PROG_STANDBY_COLD • RF_PROG_STANDBY_CONFIG • RF_PROG_STANDBY_FILESYS • RF_PROG_STANDBY_HOT • RF_PROG_STANDBY_OIR_SYNC_DONE • RF_REGISTRATION_STATUS • RF_STATUS_MAINTENANCE_ENABLE • RF_STATUS_MANUAL_SWACT • RF_STATUS_OPER_REDUNDANCY_MODE_CHANGE • RF_STATUS_PEER_COMM • RF_STATUS_PEER_PRESENCE • RF_STATUS_REDUNDANCY_MODE_CHANGE • RF_STATUS_SWACT_INHIBIT

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event	このイベントが発生する原因となる RF の進行またはステータス イベント通知。

event_register_routing

event routing コマンドによってトリガーされるイベントの登録を行います。これらのイベントは、ルート エントリが Routing Information Base（RIB）インフラストラクチャで変更されるときに、ルーティング イベント ディテクタによって処理され、イベントがパブリッシュされます。この Tcl コマンド拡張を使用すると、このスクリプトのルーティング ポリシーが実行されます。監視されるルートのネットワーク IP アドレスを指定する必要があります。

構文

```
event_register_routing [tag ?] network ? length [ge|le|ne] [type add|remove|modify|all]
 [protocol ?] [queue_priority normal|low|high|last] [maxrun ?] [nice {0 | 1}]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
network	ネットワーク IP アドレスを指定します。ネットワーク番号には、任意の有効な IP アドレスまたはプレフィックスを指定できます。

length	ネットワーク マスクの長さをビット単位で指定します。ビット マスクは 0 から 32 までの番号を使用できます。 • **ge**（任意）照合されるプレフィックスの最小の長さを指定します。**ge** キーワードは、演算子の「以上」を表します。 • **le**（任意）照合されるプレフィックスの最大の長さを指定します。**le** キーワードは、演算子の「以下」を表します。 • **ne**（任意）プレフィックスの長さを照合しない指定をします。**ne** キーワードは、演算子の「等しくない」を表します。 **ge** キーワード、**le** キーワード、および **ne** キーワードが設定されない場合、ネットワーク長の完全一致が処理されます。
type	（任意）必要なポリシーのトリガーを指定します。タイプ オプションは、**add**、**remove**、**modify**、および **all** です。デフォルトは **all** です。
プロトコル	（任意）監視されているネットワークのプロトコルの値を指定します。 次のプロトコルのいずれかを使用できます：**all**、**bgp**、**connected**、**eigrp**、**isis**。**iso-igrp**、**mobile**、**odr**、**ospf**、**rip**、**static** デフォルトは **all** です。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。

nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u" "event_severity {%s} %u network %u mask %u protocol %u lastgateway %u distance %u" "time_sec %u time_msec %u metric %u lastinterface %u"

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベントタイプのイベントの名前を表すASCII文字列。
event_pub_sec event_pub_msec	EEMに対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	イベントの重大度。
network	IP アドレス形式のネットワーク プレフィックス。
mask	IP アドレス形式のネットワーク マスク。
protocol	ネットワーク プロトコルのタイプ。
type	追加、削除、または変更するイベントのタイプ。
lastgateway	最後に認識されたゲートウェイ。
distance	アドミニストレーティブ ディスタンス。
time_sec time_msec	イベントが EEM にパブリッシュされたときの、秒単位およびミリ秒単位でのイベントの時間。
metric	パス メトリック。
lastinterface	最後に認識されたインターフェイス。

event_register_rpc

EEM SSH リモート プロシージャ コール（RPC）コマンドによってトリガーされるイベントの登録を行います。これらのイベントは、このイベントをスクリーニングするRPCイベントディテクタによって処理されます。このTclコマンド拡張を使用すると、このスクリプトのRPCポリシーが実行されます。

構文

```
event_register_rpc [queue_priority {normal | low | high | last}] [maxrun <sec.msec>]
[nice {0 | 1}] [default <sec.msec>]
```

引数

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

デフォルト	（任意）CLI イベント ディテクタがポリシーの終了を待つ時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。ポリシーが終了する前にデフォルトの時間の期限が切れると、デフォルト アクションが実行されます。デフォルト アクションによって、コマンドが実行されます。この引数が指定されない場合、デフォルトの時間は30秒に設定されます。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
arg %u"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントのIDを示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec　event_pub_msec	EEMに対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。

argc **arg0** **arg1** **arg2** **arg3** **arg4** **arg6** **arg7** **arg8** **arg9** **arg10** **arg11** **arg12** **arg13** **arg14**	Extensible Markup Language（XML）Simple Object Access Protocol（SOAP）コマンドからスクリプトに渡されるパラメータ。

event_register_snmp

簡易ネットワーク管理プロトコル（SNMP）統計イベントの登録を行います。この Tcl コマンド拡張を使用すると、SNMP オブジェクト ID（OID）によって指定されたカウンタが、定義されたしきい値に近くなったときに、ポリシーが実行されます。

構文

```
event_register_snmp [tag ?] oid ? get_type exact|next
entry_op gt|ge|eq|ne|lt|le entry_val ?
entry_type value|increment|rate
[exit_comb or|and]
[exit_op gt|ge|eq|ne|lt|le] [exit_val ?]
[exit_type value|increment|rate]
[exit_time ?] poll_interval ? [average_factor ?]
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。

oid	（必須）SNMP ドット付き表記でのデータ エレメントの OID 番号（たとえば、1.3.6.1.2.1.2.1.0）。次のタイプの OID を使用できます。 • COUNTER_TYPE • COUNTER_64_TYPE • GAUGE_TYPE • INTEGER_TYPE • OCTET_PRIM_TYPE • OPAQUE_PRIM_TYPE • TIME_TICKS_TYPE
entry_op	（必須）現在の OID データの値を開始値と比較するために使用される開始比較演算子。真の場合、イベントが発生し、終了基準を満たすまでイベント モニターリングがディセーブルにされます。
get_type	（必須）指定された OID に適用する必要がある SNMP 取得操作のタイプ。get_type 引数が「exact」の場合、指定された OID の値が取得されます。get_type 引数が「next」の場合、指定された OID の辞書順での後続値が取得されます。
entry_val	（必須）SNMP イベントを発生させる必要があるかどうかを判断するために、現在の OID データの値を比較する必要がある値。
entry-type	entry-val 引数によって指定されたオブジェクト ID に適用される操作のタイプを指定します。 値は、entry-val 引数の実際の値として定義されます。 増分では、entry-val フィールドは増分差異として使用され、entry-val は、現在のカウンタの値と、イベントが最後に真であったとき（これが新しいイベントの場合は最初にポーリングされたサンプル）の値との間の差と、比較されます。負の値によって、減少しているカウンタの増分差異がチェックされます。 レートは、ある期間の変更の平均レートとして定義されます。期間は、average-factor の値に、poll-interval の値を乗じたものです。ポーリング間隔ごとに、現在のサンプルと前のサンプルとの間の差が取得され、絶対値として記録されます。前の average-factor 値サンプルの平均は、変更のレートとして取得されます。
exit_comb	（任意）イベント モニターリングが再度イネーブルにされるよう、終了基準が満たされているかどうかを判断するために必要な、終了条件テストの組み合わせを示す、終了組み合わせ演算子を使用します。「and」の場合は、終了基準を満たすために、終了値と終了時間テストの両方を渡す必要があります。「or」の場合は、終了基準を満たすために、終了値または終了時間テストのいずれかを渡します。exit_comb が「and」の場合、exit_op と exit_val（exit_time）が存在する必要があります。exit_comb が「or」の場合、（exit_op と exit_val）または（exit_time）が存在する必要があります。

exit_op	（任意）現在の OID データの値を終了値と比較するために使用される終了比較演算子。真の場合、このイベントのイベント モニターリングが再度イネーブルにされます。
exit_val	（任意）終了基準を満たすかどうかを判断するために、現在の OID データの値を比較する必要がある値。
exit-type	（任意）exit-val 引数によって指定されたオブジェクト ID に適用される操作のタイプを指定します。指定されない場合、値が仮定されます。 値は、exit-val 引数の実際の値として定義されます。 増分では、exit-val フィールドは増分差異として使用され、exit-val は、現在のカウンタの値と、イベントが最後に真であったとき（これが新しいイベントの場合は最初にポーリングされたサンプル）の値との間の差と、比較されます。負の値によって、減少しているカウンタの増分差異がチェックされます。 レートは、ある期間の変更の平均レートとして定義されます。期間は、average-factor の値に、poll-interval の値を乗じたものです。ポーリング間隔ごとに、現在のサンプルと前のサンプルとの間の差が取得され、絶対値として記録されます。前の average-factor 値サンプルの平均は、変更のレートとして取得されます。
exit_time	（任意）イベント モニターリングが再度イネーブルにされるときに発生するイベントの後の、POSIX タイマー ユニットの数。SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数である必要があります。MMM はミリ秒を表し、0 ～ 999 の整数である必要があります。
poll_interval	（必須）POSIX タイマー ユニットの連続的なポーリング間の間隔。間隔は、現在、最小で 1 秒に設定されます（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。
average-factor	（任意）レート ベースの計算に使用される期間の計算に使用される 1 から 64 の範囲の数。average-factor の値は、poll-interval の値を乗じた値で、ミリ秒単位で導き出される期間です。最少平均係数値は 1 です。

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event_severity {%s} oid {%s} val {%s} delta_val {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	SNMP イベントの重大度で、次のいずれかの値を指定できます。 • normal • minor • major
oid	SNMP ドット付き表記での、データ エレメントのオブジェクト ID。
val	データ エレメントの値。
delta_val	ポリシーの値間のデルタ値。

event_register_snmp_notification

簡易ネットワーク管理プロトコル（SNMP）通知トラップ イベントの登録を行います。この Tcl コマンド拡張を使用すると、特定のインターフェイスまたはアドレスで、指定された SNMP オブジェクト ID（OID）で SNMP トラップが検出されるときに、ポリシーが実行されます。SNMP 通知が Tcl ポリシーを使用して動作するようにするには、**snmp-server manager** CLI コマンドを有効にする必要があります。

構文

```
event_register_snmp_notification [tag ?] oid ? oid_val ?
op {gt|ge|eq|ne|lt|le}
[maxrun ?]
[src_ip_address ?]
[dest_ip_address ?]
[queue_priority {normal|low|high|last}]
[maxrun ?]
[nice {0|1}]
[default ?]
[direction {incoming|outgoing}]
[msg_op {drop|send}]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
oid	（必須）SNMP ドット付き表記でのデータ エレメントの OID 番号（たとえば、1.3.6.1.2.1.2.1.0）。指定された OID がドット（.）で終わっている場合、ドットの前の OID 番号で始まっているすべての OID が、照会されます。次のタイプの OID を使用できます。 • COUNTER_TYPE • COUNTER_64_TYPE • GAUGE_TYPE • INTEGER_TYPE • OCTET_PRIM_TYPE • OPAQUE_PRIM_TYPE • TIME_TICKS_TYPE
oid_val	（必須）SNMP イベントを発生させる必要があるかどうかを判断するために、現在の OID データの値を比較する必要がある OID 値。
op	（必須）現在の OID データの値を、SNMP プロトコル データ ユニット（PDU）の OID データ値と比較するために使用される、比較演算子。真の場合、イベントが発生します。
maxrun	（任意）スクリプトの最大実行時間（ssssssss[.mmm] 形式で指定します。ssssssss は、0 ～ 31536000 の秒数を表す整数で、mmm は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
src_ip_address	（任意）SNMP 通知トラップが発信される発信元 IP アドレス。デフォルトは all です。すべての IP アドレスから SNMP 通知トラップを受信するよう、設定されます。
dest_ip_address	（任意）SNMP 通知トラップが送信される宛先 IP アドレス。デフォルトは all です。すべての宛先 IP アドレスから SNMP トラップを受信するよう、設定されます。

queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 queue_priority_last 引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
デフォルト	（任意）SNMP 通知イベント ディテクタがポリシーの終了を待つ、秒単位での時間を指定します。time 時間は、sssssssss[.mmm] 形式で指定します。sssssssss は、0 ～ 4294967295 の秒数を表す整数で、mmm は 0 ～ 999 のミリ秒数を表す整数である必要があります。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です
direction	（任意）発着信 SNMP トラップまたは通知 PDU がフィルタリングする方向。デフォルトは incoming です。
msg_op	（任意）イベントが一度トリガーされると、SNMP PDU（廃棄または送信）で行われるアクション。デフォルトは send です。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u
event_severity {%s}" "oid {%s} oid_val {%s} src_ip_addr {%s} dest_ip_addr {%s} x_x_x_x_x
 (varbinds) {%s} trunc_vb_buf {%s} trap_oid {%s} enterprise_oid {%s} generic_trap %u
specific_trap %u"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
oid	ユーザー指定オブジェクト ID。
oid_val	ユーザー指定オブジェクト ID 値。
src_ip_addr	SNMP プロトコル データ ユニット（PDU）の発信元 IP アドレス。
dest_ip_addr	SNMP PDU の宛先の IP アドレス。
x_x_x_x_x (varbinds)	SNMP PDU varbind 情報。
trap_oid	トラップ OID 値を示します。
enterprise_oid	エンタープライズ OID 値を示します。
generic_trap	汎用トラップ タイプの番号の 1 つを示します。0 から 6 の、7 つの汎用トラップ タイプがあります。
specific_trap	指定されたトラップ コードの番号の 1 つを示します。

event_register_snmp_object

簡易ネットワーク管理プロトコル（SNMP）オブジェクト イベントの登録を行います。この Tcl コマンド拡張を使用すると、特定のインターフェイスまたはアドレスで、指定された SNMP オブジェクト ID（OID）で SNMP が検出されるときに、値が置き換えられます。

構文

```
event_register_snmp_object oid ?
```

```
type {int|uint|counter|counter64|gauge|ipv4||oid|string}
sync {yes|no}
skip {yes|no}
[istable {yes|no}]
[default ?]
[queue_priority {normal|low|high|last}]
[maxrun ?]
[nice {0|1}]
```

引数

oid	（必須）SNMP ドット付き表記でのデータ エレメントの OID 番号（たとえば、1.3.6.1.2.1.2.1.0）。指定された OID がドット（.）で終わっている場合、ドットの前の OID 番号で始まっているすべての OID が、照会されます。次のタイプの OID を使用できます。 • COUNTER_TYPE • COUNTER_64_TYPE • GAUGE_TYPE • INTEGER_TYPE • OCTET_PRIM_TYPE • OPAQUE_PRIM_TYPE • TIME_TICKS_TYPE
type	（必須）OID 値のタイプ。
sync	（必須）「yes」は、EEM ポリシーが通知されることを意味します。アプレット set_exit_status または Tcl 戻り値が 0 の場合、SNMP によって、要求が処理されます。戻り値が 1 の場合、SNMP によって、get 要求のポリシーで指定された値が使用され、set 要求は処理されません。「no」は、EEM は通知されず、SNMP によって要求が処理されることを意味します。 1 つの OID のみが、同期ポリシーに関連付けられます。ただし、複数の同期ポリシーが、同じ OID に登録できます。
skip	sync 引数が no の場合は必須で、sync 引数が yes の場合は不要ですskip 引数が「yes」の場合、SNMP によって要求が処理されることを意味します。skip 引数が「no」の場合、SNMP は、オブジェクトが存在しないかのように動作することを意味します。
istable	（任意）値「no」は、OIDがスカラーオブジェクトであることを意味し、「yes」は、OID がテーブル オブジェクトであることを意味します。

デフォルト	（任意）SNMP オブジェクト イベント ディテクタがポリシーの終了を待つ時間（sssssssss[.mmm] 形式で指定します。sssssssss は、0 ～ 4294967295 の秒数を表す整数で、mmm は 0 ～ 999 のミリ秒数を表す整数である必要があります）。ポリシーが終了する前にデフォルトの時間の期限が切れると、デフォルト アクションが実行されます。デフォルト アクションは、通常、SNMP サブシステムによって set 要求または get 要求を処理することです。この引数が指定されない場合、デフォルトの時間は 30 秒に設定されます。
maxrun	（任意）スクリプトの最大実行時間（ssssssss[.mmm] 形式で指定します。ssssssss は、0 ～ 31536000 の秒数を表す整数で、mmm は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 queue_priority_last 引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u
event_severity {%s}" "oid {%s} request {%s} request_type {%s} value %u"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
event_severity	イベントの重大度。
oid	受信した get 要求または set 要求の SNMP オブジェクトの ID。
request	get または set の要求タイプ。
request_type	要求のタイプ（現在または次の）。
value	set 要求のみ。オブジェクトに設定される値。

event_register_syslog

Syslog イベントの登録を行います。この Tcl コマンド拡張を使用すると、一定の時間内に一定回数の発生後、特定パターンの Syslog メッセージが記録されるときに、ポリシーがトリガーされます。

構文

```
event_register_syslog [tag ?] [occurs ?] [period ?] pattern ?
[priority all|emergencies|alerts|critical|errors|warnings|notifications|
informational|debugging|0|1|2|3|4|5|6|7]
[queue_priority low|normal|high|last]
[severity_fatal] [severity_critical] [severity_major]
[severity_minor] [severity_warning] [severity_notification]
[severity_normal] [severity_debugging]
[maxrun ?] [nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。

occurs	（任意）イベントが発生する前の発生回数。この引数が指定されない場合、イベントは1回目から発生します。指定される場合、0より大きい値を指定する必要があります。
period	（任意）イベントを発生させるために取る必要がある 1 つまたは複数のイベントの間の、秒単位およびミリ秒単位の時間の間隔（SSSSSSSSSS[.MMM]形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0～999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、期間チェックは適用されません。
pattern	（必須）Syslog メッセージ パターン マッチの実行に使用される正規表現。この引数は、記録された Syslog メッセージを指定するためにポリシーによって使用されます。
priority	（任意）スクリーニングされるメッセージのプライオリティ。この引数が指定される場合、指定されたロギング プライオリティ レベルまたはそれ以下メッセージのみがスクリーニングされます。この引数が指定されない場合、デフォルトのプライオリティは 0 です。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。

nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です
severity_xxx	（任意）スクリーニングされるイベントの重大度。この引数が指定される場合、指定された重大度のメッセージのみがスクリーニングされます。syslog イベントの重大度レベルのマッピングについては、「Syslog イベントの重大度のマッピング」というタイトルの表を参照してください。

複数の条件が存在する場合、すべての条件が一致したときに、Syslogイベントが発生します。

表 *202 : Syslog* イベントの重大度のマッピング

重大度のキーワード	Syslogのプライオリティ	説明
severity_fatal	LOG_EMERG (0)	システムが使用不可能な状態。
severity_critical	LOG_ALERT (1)	クリティカル条件で、即時対応が必要であることを示す
severity_major	LOG_CRIT (2)	重大な状態。
severity_minor	LOG_ERR (3)	軽微な状態。
severity_warning	LOG_WARNING (4)	警告状態。
severity_notification	LOG_NOTICE (5)	基本的な通知、情報メッセージ
severity_normal	LOG_INFO (6)	正常なイベント、正常な状態に戻ったことを伝える
severity_debugging	LOG_DEBUG (7)	デバッグ メッセージ。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"msg {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントのIDを示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
msg	パターンと一致する最後の Syslog メッセージ。

event_register_timer

パブリッシャとサブスクライバの両方として、タイマーを作成し、タイマーイベントの登録を行います。時間特有または時間に基づいたポリシーをトリガーする必要があるときに、この Tcl コマンド拡張を使用します。このイベント タイマーは、イベントのパブリッシャとサブスクライバの両方です。パブリッシャの部分は、名前付きタイマーがオフになるという条件を示します。サブスクライバの部分は、イベントが登録されているタイマーの名前を示します。

（注） CRON および絶対時間の指定は、現地時間で動作します。

構文

```
event_register_timer [tag ?] watchdog|countdown|absolute|cron
[name ?] [cron_entry ?]
[time ?]
[queue_priority low|normal|high|last] [maxrun ?]
[nice 0|1]
```

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
watchdog	（必須）ウォッチドッグ タイマー。
countdown	（必須）カウントダウン タイマー。
絶対	（必須）絶対タイマー。
cron	（必須）CRON タイマー。

name	（任意）タイマーの名前。

cron_entry	

	（任意）CRON タイマー タイプが指定される場合に、指定する必要があります。他のいずれかのタイマー タイプが指定される場合には、指定しないでください。cron_entry は、UNIX CRON デーモンで使用される部分的な UNIX Crontab エントリ（最初の 5 つのフィールド）です。 cron_entry の指定は、5 つのフィールドが使用されるテキスト文字列で構成されます。フィールドは、空白文字で区切られます。フィールドは、CRON タイマー イベントがトリガーされるときの時刻と日付を表します。フィールドの説明については、「CRON イベントがトリガーされるときの時刻と日付」というタイトルの表を参照してください。 番号の範囲を使用できます。範囲は、ハイフンで区切られる 2 つの数字で表示されます。範囲には、2つの数字自身も含まれます。たとえば、時刻に入力される 8-11 は、8 時、9 時、10 時、および 11 時での実行を示します。 フィールドはアスタリスク記号（*）も使用でき、これは常に「first-last」を表します。 リストを使用できます。リストは、カンマで区切られた番号のセット（または範囲）です。例："1,2,5,9" および "0-4,8-12"。 手順の値は、範囲の組み合わせで使用できます。範囲に続く「/<number>」によって、範囲内での省略値を指定します。たとえば、2時間ごとにイベントのトリガーを指定する場合、「0-23/2」を hour フィールドに使用できます。アスタリスク記号後にも手順を使用でき、「2 時間ごと」と指定する場合は、「*/2」を使用します。 month フィールドと day of week フィールドには、名前も使用できます。特定の日または月の最初の 3 文字を使用します（ケースは問題ではありません）。名前の範囲またはリストは使用できません。 タイマー イベントがトリガーされる日は、day of month と day of week の 2 つのフィールドで指定できます。両方のフィールドが制限される（つまり * ではない）場合、いずれかのフィールドが現在の時刻と一致すると、イベントがトリガーされます。たとえば、「30 4 1,15 * 5」の場合、各月の 1 日と 15 日に加え、金曜日の午前 4:30 にイベントがトリガーされます。 最初の 5 つのフィールドの代わりに、7 つの特殊文字列の 1 つが表示されることがあります。これらの 7 つの特殊文字の説明については、「cron_entry の特殊文字列」というタイトルの表を参照してください。 例 1：「0 0 1,15 * 1」では、各月の 1 日と 15 日、および月曜日ごとに、真夜中の 0 時に、イベントがトリガーされます。1 つのフィールドによってのみ日を指定する場合、他のフィールドは * に設定する必要があります。「0 0 * * 1」では、月曜日にのみ、真夜中の 0 時に、イベントがトリガーされます。 例 2：「15 16 1 * *」では、各月の 1 日の午後 4:15 にイベントがトリガーされます。 例 3：「0 12 * * 1-5」では、各週の月曜日から金曜日まで、正午に、イベントがトリガーされます。

	例 4：「@weekly」では、1 週間に一度、日曜日の真夜中の 0 時に、イベントがトリガーされます。
time	（任意）CRON 以外のタイマー タイプが指定される場合に、指定する必要があります。CRON タイマー タイプが指定される場合には、指定しないでください。ウォッチドッグ タイマーとカウントダウン タイマーでは、タイマーの期限が切れるまでの秒およびミリ秒の単位での数です。絶対タイマーでは、期限切れ時刻のカレンダー時間です。時間は、SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります。期限の絶対日付は、1970 年 1 月 1 日以降の秒およびミリ秒の単位での数です。指定された日付がすでに過ぎた場合、タイマーの期限はただちに切れます。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

表 *203 : CRON* イベントがトリガーされるときの時刻と日付

フィールド	使用可能な値
minute	0 ～ 59
hour	0 ～ 23
day of month	1 ～ 31
month	1 ～ 12（または名前、下記を参照）
day of week	0 ～ 7（0 または 7 が日曜日または名前。「cron_entry の特殊文字列」というタイトルの表を参照）

表 *204 : cron_entry* の特殊文字列

文字列	意味
@yearly	1 年に 1 回トリガーする、「0 0 1 1 *」。
@annually	@yearly と同じ。
@monthly	1 か月に 1 回トリガーする、「0 0 1 * *」。
@weekly	1 週間に 1 回トリガーする、「0 0 * * 0」。
@daily	1 日に 1 回トリガーする、「0 0 * * *」。
@midnight	@daily と同じ。
@hourly	1 時間に 1 回トリガーする、「0 * * * *」。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"timer_type %s timer_time_sec %ld timer_time_msec %ld"
"timer_remain_sec %ld timer_remain_msec %ld"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec　event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
timer_type	タイマーのタイプ。次のいずれかです。 • watchdog • countdown • 絶対
timer_time_sec　timer_time_msec	タイマーの期限が切れる時間。
timer_remain_sec **timer_remain_msec**	次の期限切れ前の残りの時間。

関連項目

event_register_timer_subscriber

event_register_timer_subscriber

サブスクライバとしてタイマー イベントの登録を行います。この Tcl コマンド拡張を使用すると、サブスクライバとして、登録するイベント タイマーの名前が指定されます。イベント タイマーは、別のポリシーまたは別のプロセスに依存して、カウンタが実際に操作されます。たとえば、policyB はタイマー加入者ポリシーとして動作しますが、policyA（タイマー ポリシーは不要ですが）では、register_counter、timer_arm、または timer_cancel の各 Tcl コマンド拡張を使用して、policyB で参照されているカウンタが操作されます。

構文

```
event_register_timer_subscriber watchdog|countdown|absolute|cron
name ? [queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

引数

watchdog	（必須）ウォッチドッグ タイマー。

countdown	（必須）カウントダウン タイマー。
絶対	（必須）絶対タイマー。
cron	（必須）CRON タイマー。
name	（必須）タイマーの名前。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

（注） タイマーイベントまたはカウンタイベントの登録を行う EEM ポリシーは、パブリッシャとサブスクライバの両方として動作できます。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"timer_type %s timer_time_sec %ld timer_time_msec %ld"
"timer_remain_sec %ld timer_remain_msec %ld"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec　event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
timer_type	タイマーのタイプ。次のいずれかです。 • watchdog • countdown • 絶対
timer_time_sec　timer_time_msec	タイマーの期限が切れる時間。
timer_remain_sec **timer_remain_msec**	次の期限切れ前の残りの時間。

関連項目

event_register_timer

event_register_track

Cisco IOS Object Tracking サブシステムからのレポート イベントの登録を行います。この Tcl コマンド拡張を使用すると、指定されたオブジェクト番号の Cisco IOS Object Tracking サブシステム レポートに基づいて、ポリシーがトリガーされます。

構文

```
event_register_track ? [tag ?] [state up|down|any] [queue_priority low|normal|high|last]
```

```
 [maxrun ?]
[nice 0|1]
```

引数

?（番号を表す）	（必須）1 から 500 の範囲でトラックされるオブジェクト番号。
tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
state	（任意）トラックされるオブジェクトの状態遷移によってイベントが発生するよう、指定します。**up** が指定されている場合、トラックされるオブジェクトが down 状態から up 状態に遷移するときにイベントが発生します。**down** が指定されている場合、トラックされるオブジェクトが up 状態から down 状態に遷移するときにイベントが発生します。**any** が指定されている場合、トラックされるオブジェクトがある状態から別の状態に遷移するときにイベントが発生します。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

任意の引数が指定されない場合、イベントは、引数のすべての可能な値に対して照会されます。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"track_number {%u} track_state {%s}"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一のイベント ID を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec **event_pub_msec**	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
track_number	イベントがトリガーされる原因となるトラックされるオブジェクトの番号。
track_state	イベントがトリガーされたときのトラックされるオブジェクトの状態。有効な値は up または down です。

event_register_wdsysmon

Watchdog System Monitor イベントの登録を行います。この Tcl コマンド拡張を使用すると、いくつかのサブイベントまたは条件の組み合わせである複合イベントの登録が行われます。たとえば、特定処理の CPU の使用率が 80% を超える場合で、かつ処理に使用されるメモリが初期割り当て容量の 50% よりも大きい場合といった条件を組み合わせて登録できます。この Tcl コマンド拡張は、ソフトウェア モジュール方式イメージでのみサポートされます。

構文

```
event_register_wdsysmon [tag ?] [timewin ?]
[sub12_op and|or|andnot]
[sub23_op and|or|andnot]
```

```
[sub34_op and|or|andnot]
[sub1 subevent-description]
[sub2 subevent-description]
[sub3 subevent-description]
[sub4 subevent-description] [node ?]
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

各引数は、位置に依存しません。

（注） 演算子の定義は、and（論理 And 操作）or（論理 Or 操作）、andnot（論理 And Not 操作）です。たとえば、「sub12_op and」では、サブイベント 1 およびサブイベント 2 が真であるときにイベントが発生するよう定義されます。「sub23_op or」では、sub12_op で定義された条件が真で、サブイベント 3 が真であるときに、イベントが発生するよう定義されます。ロジックは、次のようにダイアグラム化できます。(((sub1 sub12_op sub2) sub23_op sub3) sub34_op sub4) が真の場合、イベントが発生

引数

tag	（任意）Tcl スクリプト内で複数のイベント文をサポートするため、Tcl コマンド拡張のトリガーとともに使用できるタグを指定する文字列。
timewin	（任意）イベントが生成されるようにするために、すべてのサブイベントが発生する必要がある時間ウィンドウ（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。
sub12_op	（任意）サブイベント 1 とサブイベント 2 とを比較する組み合わせ演算子。
sub23_op	（任意）サブイベント 1、2 とサブイベント 3 とを比較する組み合わせ演算子。
sub34_op	（任意）サブイベント 1、2、サブイベント 3、サブイベント 4 とを比較する組み合わせ演算子。
sub1	（任意）サブイベント 1 の指定を意味します。
subevent-description	（任意）サブイベントの構文。
sub2	（任意）サブイベント 2 の指定を意味します。
sub3	（任意）サブイベント 3 の指定を意味します。
sub4	（任意）サブイベント 4 の指定を意味します。

ノード	（任意）デッドロック条件が監視されるノード名は、「node」という語句と、それに続く、次の形式を使用してスラッシュ文字で区切られた 2 つのフィールドで構成される文字列です。 node<slot-number>/<cpu-number> slot-number は、ハードウェア スロット番号です。cpu-number は、ハードウェア CPU 番号です。たとえば、スロット 0 にある Cisco Catalyst 6500 シリーズ スイッチのスーパーバイザ カードの SP CPU は、node0/0 と指定されます。たとえば、スロット 0 にある Cisco Catalyst 6500 シリーズ スイッチのスーパーバイザ カードの RP CPU は、node0/1 と指定されます。node 引数が指定されない場合、デフォルトのノード指定は、登録が行われているローカル ノードです。
queue_priority	（任意）次のような、スクリプトがキューに入れられる優先度レベル。 • queue_priority low：3 つの優先度レベルの最も低いレベルでスクリプトがキューに入れられるよう指定します。 • queue_priority normal：low プライオリティよりも高く、high プライオリティよりも低い優先度レベルでスクリプトがキューに入れられるよう指定します。 • queue_priority high：3 つの優先度レベルの最も高いレベルで、スクリプトがキューに入れられるよう指定します。 • queue_priority last：最も低い優先度レベルでスクリプトがキューに入れられるよう指定します。 「queue_priority_last」引数が設定された状態で複数のスクリプトが登録されている場合、これらのスクリプトは、イベントのパブリッシュ順に実行されます。 （注） queue_priority 引数によって、登録されているスクリプトの実行優先度ではなく、キューイングの優先度が指定されます。 この引数が指定されない場合、デフォルトのキューイング プライオリティは normal です。
maxrun	（任意）スクリプトの最大ランタイム（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合、デフォルトの 20 秒ランタイム制限が使用されます。
nice	（任意）ポリシーの実行時間優先度の設定。nice 引数が 1 に設定されている場合、ポリシーは、デフォルトの優先度よりも低い実行時間優先度で実行されます。デフォルト値は 0 です

サブイベント

subevent description の構文は、7 つのケースのうちの 1 つを使用できます。

subevent descriptions の引数では、number 引数の値に次の制約事項が適用されます。

- dispatch_mgr では、val は、0 ～ 4294967295 の範囲の整数である必要があります。
- cpu_proc および cpu_tot では、val は、0 ～ 100 の整数である必要があります。
- mem_proc、mem_tot_avail、および mem_tot_used では、is_percent が偽の場合、val は、0 ～ 4294967295 の範囲の整数である必要があります。

```
1. deadlock procname ?
```

引数

procname	（必須）デッドロック条件をモニターするプロセス名を指定する正規表現。指定された場合、サブイベントによって、時間ウィンドウは無視されます。

```
2. dispatch_mgr [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [period ?]
```

引数

procname	（任意）dispatch_manager ステータスをモニターするプロセス名を指定する正規表現。
op	（任意）収集されたイベント数を指定された値と比較するために使用される比較演算子。真の場合、このイベントが発生します。
val	（任意）発生したイベント数の値を比較する必要があります。
period	（任意）発生したイベント数の時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合は、最新のサンプルが使用されます。

```
3. cpu_proc [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [period ?]
```

引数

procname	（任意）CPU の使用条件をモニターするプロセス名を指定する正規表現。
op	（任意）収集された CPU 使用率サンプル パーセンテージを、指定されたパーセント値と比較するために使用される、比較演算子。真の場合、このイベントが発生します。
val	（任意）サンプル期間の平均 CPU 使用率のパーセント値を比較する必要があります。

period	（任意）サンプルの収集の平均の時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合は、最新のサンプルが使用されます。

```
4. cpu_tot [op gt|ge|eq|ne|lt|le] [val ?] [period ?]
```

引数

op	（任意）収集された合計システム CPU 使用率サンプル パーセンテージを、指定されたパーセント値と比較するために使用される、比較演算子。真の場合、このイベントが発生します。
val	（任意）サンプル期間の平均 CPU 使用率のパーセント値を比較する必要があります。
period	（任意）サンプルの収集の平均の時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合は、最新のサンプルが使用されます。

```
5. mem_proc [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [is_percent TRUE|FALSE] [period
 ?]
```

引数

procname	（任意）メモリ使用状況をモニターするプロセス名を指定する正規表現。
op	（任意）収集された使用メモリを、指定された値と比較するために使用される比較演算子。真の場合、このイベントが発生します。
val	（任意）キロバイト単位で指定される、パーセンテージまたは絶対値。パーセンテージは、指定された時間内で最も古いサンプルと、最新のサンプルとの違いを表します。メモリ使用量が時間内で 150 KB から 300 KB に増えた場合、増加パーセンテージは 100 です。これは、測定された値を比較する必要がある値です。
is_percent	（任意）真の場合、パーセンテージの値が収集され、比較されます。これ以外の場合、絶対値が収集され、比較されます。
period	（任意）is_percent が真に設定される場合、時間のパーセンテージが計算されます。これ以外の場合、収集されるサンプルの平均が計算される時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合は、最新のサンプルが使用されます。

```
6. mem_tot_avail [op gt|ge|eq|ne|lt|le] [val ?] [is_percent TRUE|FALSE] [period ?]
```

引数

op	（任意）使用可能な収集されたメモリを指定された値と比較するために使用される比較演算子。真の場合、このイベントが発生します。
val	（任意）キロバイト単位で指定される、パーセンテージまたは絶対値。パーセンテージは、指定された時間内で最も古いサンプルと、最新のサンプルとの違いを表します。使用可能なメモリ使用量が時間内で 300 KB から 150 KB に減った場合、減少パーセンテージは 50 です。これは、測定された値を比較する必要がある値です。
is_percent	（任意）真の場合、パーセンテージの値が収集され、比較されます。これ以外の場合、絶対値が収集され、比較されます。
period	（任意）is_percent が真に設定される場合、時間のパーセンテージが計算されます。これ以外の場合、収集されるサンプルの平均が計算される時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合は、最新のサンプルが使用されます。

```
7. mem_tot_used [op gt|ge|eq|ne|lt|le] [val ?] [is_percent TRUE|FALSE] [period ?]
```

引数

op	（任意）収集された使用メモリを、指定された値と比較するために使用される比較演算子。真の場合、このイベントが発生します。
val	（任意）キロバイト単位で指定される、パーセンテージまたは絶対値。パーセンテージは、指定された時間内で最も古いサンプルと、最新のサンプルとの違いを表します。メモリ使用量が時間内で 150 KB から 300 KB に増えた場合、増加パーセンテージは 100 です。これは、測定された値を比較する必要がある値です。
is_percent	（任意）真の場合、パーセンテージの値が収集され、比較されます。これ以外の場合、絶対値が収集され、比較されます。
period	（任意）is_percent が真に設定される場合、時間のパーセンテージが計算されます。これ以外の場合、収集されるサンプルの平均が計算される時間（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。この引数が指定されない場合は、最新のサンプルが使用されます。 （注） is_percent が真に設定されている場合、この引数は必須です。これ以外の場合、この引数は任意です。

結果文字列

なし

_cerrno を設定

なし

Event_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"num_subs %u"
```

イベント タイプ	説明
event_id	パブリッシュされた該当イベントの ID を示す一意の番号。同一のイベントで複数のポリシーを実行可能であり、その場合、各ポリシーは同一の event_id を保持します。
event_type	イベントのタイプ。
event_type_string	このイベント タイプのイベントの名前を表す ASCII 文字列。
event_pub_sec event_pub_msec	EEM に対してイベントがパブリッシュされた、秒単位およびミリ秒単位の時間。
num_subs	サブイベント番号。

サブイベント情報文字列は、次のような、デッドロック サブイベント用です。

```
"{type %s num_entries %u entries {entry 1, entry 2, ...}}"
```

サブイベントタイプ	説明
type	Wdsysmon サブイベントのタイプ。
num_entries	デッドロックのプロセスおよびスレッドの番号。
entries	デッドロックのプロセスおよびスレッドの情報。

各エントリは次のとおりです。

```
"{node {%s} procname {%s} pid %u tid %u state %s b_node %s b_procname %s b_pid %u
b_tid %u}"
```

このエントリでは、プロセス A のスレッド m によって、プロセス B のスレッド n でブロックされるシナリオが記述されているとすると、次のようになります。

サブイベントタイプ	説明
node	プロセス A のスレッド m があるノードの名前。
procname	プロセス A の名前。
pid	プロセス A のプロセス ID。

サブイベントタイプ	説明
tid	プロセス A のスレッド m のスレッド ID。
state	プロセス A のスレッド m のスレッド状態。次のいずれかになります。 • STATE_CONDVAR • STATE_DEAD • STATE_INTR • STATE_JOIN • STATE_MUTEX • STATE_NANOSLEEP • STATE_READY • STATE_RECEIVE • STATE_REPLY • STATE_RUNNING • STATE_SEM • STATE_SEND • STATE_SIGSUSPEND • STATE_SIGWAITINFO • STATE_STACK • STATE_STOPPED • STATE_WAITPAGE • STATE_WAITTHREAD
b_node	プロセス B のスレッドがあるノードの名前。
b_procname	プロセス B の名前。
b_pid	プロセス B のプロセス ID。
b_tid	プロセス B のスレッド n のスレッド ID。0 は、プロセス A のスレッド m は、プロセス B のすべてのスレッド上でブロックされることを意味します。

dispatch_mgr サブイベントについて

```
"{type %s node {%s} procname {%s} pid %u value %u sec %ld msec %ld}"
```

サブイベント タイプ	説明
type	Wdsysmon サブイベントのタイプ。
node	POSIX プロセスが存在するノードの名前。
procname	このサブイベントの POSIX プロセス名。
pid	このサブイベントの POSIX プロセス ID。 （注） 前述の 3 つのフィールドは、このディスパッチ マネージャのオーナー プロセスについて説明します。
value	**sec** 変数と **msec** 変数が、0 に指定されている、またはイベント登録 Tcl コマンド拡張で指定されていない場合、ディスパッチ マネージャによって処理されたイベント数は、最新サンプル内にあります。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、このディスパッチ マネージャによって処理されるイベントの合計数は、該当する時間ウィンドウにあります。
sec msec	イベント登録 Tcl コマンド拡張で、**sec** 変数と **msec** 変数が 0 に指定されているか、または指定されていない場合、両方とも 0 です。登録 Tcl コマンド拡張で時間ウィンドウが指定され、かつその時間ウィンドウがゼロよりも大きい場合、**sec** 変数および **msec** 変数は、この時間ウィンドウの最も古いサンプルと最新のサンプルとの実際の時間の差分です。

cpu_proc サブイベントについて

```
"{type %s node {%s} procname {%s} pid %u value %u sec %ld msec %ld}"
```

サブイベント タイプ	説明
type	Wdsysmon サブイベントのタイプ。
node	POSIX プロセスが存在するノードの名前。
procname	このサブイベントの POSIX プロセス名。
pid	このサブイベントの POSIX プロセス ID。 （注） 前述の 3 つのフィールドは、その CPU 使用率がモニターされているプロセスについて説明します。

サブイベント タイプ	説明
value	**sec** 変数と **msec** 変数が 0 に指定されている、またはイベント登録 Tcl コマンド拡張で指定されていない場合、プロセスの CPU 使用率は最新サンプル内にあります。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、プロセス CPU 使用率の平均は、該当する時間ウィンドウにあります。
sec　msec	イベント登録 Tcl コマンド拡張で、**sec** 変数と **msec** 変数が 0 に指定されているか、または指定されていない場合、両方とも 0 です。登録 Tcl コマンド拡張で時間ウィンドウが指定され、かつその時間ウィンドウがゼロよりも大きい場合、**sec** 変数および **msec** 変数は、この時間ウィンドウの最も古いサンプルと最新のサンプルとの実際の時間の差分です。

cpu_tot サブイベントについて

```
"{type %s node {%s} value %u sec %ld msec %ld}"
```

サブイベント タイプ	説明
type	Wdsysmon サブイベントのタイプ。
node	CPU 使用率の合計がモニターされているノードの名前。
value	**sec** 変数と **msec** 変数が 0 に指定されている、またはイベント登録 Tcl コマンド拡張で指定されていない場合、合計 CPU 使用率は最新サンプル内にあります。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、合計 CPU 使用率の平均は、該当する時間ウィンドウにあります。
sec　msec	イベント登録 Tcl コマンド拡張で、**sec** 変数と **msec** 変数が 0 に指定されているか、または指定されていない場合、両方とも 0 です。登録 Tcl コマンド拡張で時間ウィンドウが指定され、かつその時間ウィンドウがゼロよりも大きい場合、**sec** 変数および **msec** 変数は、この時間ウィンドウの最も古いサンプルと最新のサンプルとの実際の時間の差分です。

mem_proc サブイベントについて

```
"{type %s node {%s} procname {%s} pid %u is_percent %s value %u diff %d sec %ld msec
%ld}"
```

サブイベント タイプ	説明
type	Wdsysmon サブイベントのタイプ。

サブイベントタイプ	説明
node	POSIX プロセスが存在するノードの名前。
procname	このサブイベントの POSIX プロセス名。
pid	このサブイベントの POSIX プロセス ID。 （注） 前述の 3 つのフィールドは、そのメモリ使用率がモニターされているプロセスについて説明します。
is_percent	TRUE または FALSE のいずれかです。TRUE は、値がパーセント値であることを示します。FALSE は、値が絶対値であることを示します（平均値の場合もあります）。
value	**sec** 変数と **msec** 変数が 0 に指定されている、またはイベント登録 Tcl コマンド拡張で指定されていない場合、プロセスで使用されたメモリは最新サンプル内にあります。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、プロセスで使用されたメモリ使用率の平均は、該当する時間ウィンドウにあります。
サブイベントタイプ	説明
diff	**sec** 変数と **msec** 変数が 0 に指定されている、またはイベント登録 Tcl コマンド拡張で指定されていない場合、**diff** は、今まで収集されたプロセスで使用されたメモリの最初のサンプルと、プロセスで使用されたメモリの最新サンプルのパーセンテージの差分です。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、**diff** は、プロセスで使用されたメモリの使用状況のうち、指定された時間ウィンドウで最も古い値と最新の値のパーセンテージの差分です。
sec　msec	イベント登録 Tcl コマンド拡張で、**sec** 変数と **msec** 変数が 0 に指定されているか、または指定されていない場合、両方とも 0 です。登録 Tcl コマンド拡張で時間ウィンドウが指定され、かつその時間ウィンドウがゼロよりも大きい場合、**sec** 変数および **msec** 変数は、この時間ウィンドウの最も古いサンプルと最新のサンプルとの実際の時間の差分です。

is_percent 引数が FALSE であり、イベント登録 Tcl コマンド拡張で **sec** 引数と **msec** 引数が 0 に指定されているか、または指定されていない場合は、次のようになります。

- **value** は最新のサンプルでプロセスによって使用されたメモリです。
- **diff** は 0 です。
- **sec** と **msec** は両方とも 0 です。

is_percent 引数が FALSE であり、イベント登録 Tcl コマンド拡張で時間ウィンドウがゼロよりも大きい値に指定されている場合は、次のようになります。

- **value** は指定された時間ウィンドウでプロセスによって使用されたメモリ サンプル値の平均です。
- **diff** は 0 です。
- **sec** および **msec** は、両方とも、この時間ウィンドウ内の最も古いサンプルのタイムスタンプと最新のサンプルのタイムスタンプの実際の時間の差分です。

is_percent 引数が TRUE であり、イベント登録 Tcl コマンド拡張で時間ウィンドウがゼロよりも大きい値に指定されている場合は、次のようになります。

- **value** は 0 です。
- **diff** は指定された時間ウィンドウの、最も古いプロセスで使用されたメモリ サンプルと最新のプロセスで使用されたメモリ サンプルとのパーセンテージによる差分です。
- **sec** および **msec** は、プロセスで使用されたメモリ サンプルの、この時間ウィンドウ内の最も古いタイムスタンプと最新のプタイムスタンプの実際の時間の差分です。

is_percent 引数が TRUE であり、イベント登録 Tcl コマンド拡張で **sec** 引数と **msec** 引数が 0 に指定されているか、または指定されていない場合は、次のようになります。

- **value** は 0 です。
- **diff** は今まで収集された、最初のプロセスで使用されたメモリ サンプルと、最新のプロセスで使用されたメモリ サンプルとのパーセンテージによる差分です。
- **sec** および **msec** は、今まで収集されたプロセスで使用されたメモリの最初のサンプルのタイムスタンプと、プロセスで使用されたメモリの最新のサンプルのタイムスタンプの実際の時間の差分です。

mem_tot_avail サブイベントについて

```
"{type %s node {%s} is_percent %s used %u avail %u diff %d sec %ld msec %ld}"
```

サブイベント タイプ	説明
type	Wdsysmon サブイベントのタイプ。
node	使用可能なメモリの合計がモニターされているノードの名前。
is_percent	TRUE または FALSE のいずれかです。TRUE は、値がパーセント値であることを示します。FALSE は、値が絶対値であることを示します（平均値の場合もあります）。

サブイベントタイプ	説明
used	**sec** 変数と **msec** 変数が、0 に指定されるか、または、イベント登録 Tcl コマンド拡張で指定されない場合、使用されたメモリの合計は、最新のサンプルにあります。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、使用された合計メモリ使用率の平均は、該当する時間ウィンドウにあります。
avail	**sec** 変数と **msec** 変数が 0 に指定されている、または、イベント登録 Tcl コマンド拡張で指定されていない場合、**avail** は使用可能な総メモリの最新サンプル内にあります。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、**avail** は、指定された時間ウィンドウ内での使用可能な総メモリの使用率です。
diff	**sec** 変数と **msec** 変数が 0 に指定されている、または、イベント登録 Tcl コマンド拡張で指定されていない場合、**diff** は、今まで収集された使用可能な総メモリの最初のサンプルと、使用可能な総メモリの最新サンプルのパーセンテージの差分です。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、**diff** は、使用可能な総メモリの使用率のうち、指定された時間ウィンドウで最も古い値と最新の値のパーセンテージの差分です。
sec msec	イベント登録 Tcl コマンド拡張で、**sec** 変数と **msec** 変数が 0 に指定されているか、または指定されていない場合、両方とも 0 です。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、これらの変数は、この時間ウィンドウの、最も古いサンプルと最新のサンプルとの実際の時間の差分です。

is_percent 引数が FALSE であり、イベント登録 Tcl コマンド拡張で **sec** 引数と **msec** 引数が 0 に指定されているか、または指定されていない場合は、次のようになります。

- **used** は最新のサンプルで使用されたメモリの合計です。
- **avail** は最新のサンプルで使用可能なメモリの合計です。
- **diff** は 0 です。
- **sec** と **msec** は両方とも 0 です。

is_percent 引数が FALSE であり、イベント登録 Tcl コマンド拡張で時間ウィンドウがゼロよりも大きい値に指定されている場合は、次のようになります。

- **used** は 0 です。
- **avail** は指定された時間ウィンドウで使用可能な合計メモリ サンプル値の平均です。
- **diff** は 0 です。
- **sec** および **msec** は、両方とも、この時間ウィンドウ内の使用可能な総メモリの最も古いサンプルのタイムスタンプと最新サンプルのタイムスタンプの実際の時間の差分です。

is_percent 引数が TRUE であり、イベント登録 Tcl コマンド拡張で時間ウィンドウがゼロよりも大きい値に指定されている場合は、次のようになります。

- **used** は 0 です。
- **avail** は 0 です。
- **diff** 指定された時間ウィンドウの、最も古い使用可能なメモリ サンプルの合計と最新の可能なメモリ サンプルの合計とのパーセンテージによる差分です。
- **sec** および **msec** は、両方とも、この時間ウィンドウ内の使用可能な総メモリの最も古いサンプルのタイムスタンプと最新サンプルのタイムスタンプの実際の時間の差分です。

is_percent 引数が TRUE であり、イベント登録 Tcl コマンド拡張で **sec** 引数と **msec** 引数が 0 に指定されているか、または指定されていない場合は、次のようになります。

- **used** は 0 です。
- **avail** は 0 です。
- **diff** 今まで収集された、最初の使用可能なメモリ サンプルの合計と、最新の使用可能なメモリ サンプルの合計との間の、パーセンテージによる差です。
- **sec** および **msec** は、今まで収集された使用可能な総メモリの最初のサンプルのタイムスタンプと、使用可能な総メモリの最新サンプルのタイムスタンプ間の実際の時間の差です。

mem_tot_used サブイベントについて

```
"{type %s node {%s} is_percent %s used %u avail %u diff %d sec %ld msec %ld}"
```

サブイベントタイプ	説明
type	Wdsysmon サブイベントのタイプ。
node	使用されているメモリの合計がモニターされているノードの名前。
is_percent	TRUE または FALSE のいずれかです。TRUE は、値がパーセント値であることを示します。FALSE は、値が絶対値であることを示します（平均値の場合もあります）。
used	**sec** 変数と **msec** 変数が、0 に指定されるか、または、イベント登録 Tcl コマンド拡張で指定されない場合、使用されたメモリの合計は、最新のサンプルにあります。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、使用された合計メモリ使用率の平均は、該当する時間ウィンドウにあります。

サブイベントタイプ	説明
avail	**sec** 変数と **msec** 変数が、0 に指定されている、または、イベント登録 Tcl コマンド拡張で指定されていない場合、**avail** は使用されたメモリ合計の最新サンプル内にあります。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、**avail** は、指定された時間ウィンドウ内での使用されたメモリ合計の使用状況です。
diff	**sec** 変数と **msec** 変数が 0 に指定されている、または、イベント登録 Tcl コマンド拡張で指定されていない場合、**diff** は、今まで収集された使用されたメモリ合計の最初のサンプルと、使用されたメモリ合計の最新サンプルのパーセンテージの差分です。時間ウィンドウが指定され、登録 Tcl コマンド拡張でゼロよりも大きい場合、**diff** は、使用されたメモリ合計の使用状況のうち、指定された時間ウィンドウで最も古い値と最新の値のパーセンテージの差分です。
sec msec	イベント登録 Tcl コマンド拡張で、**sec** 変数と **msec** 変数が 0 に指定されているか、または指定されていない場合、両方とも 0 です。登録 Tcl コマンド拡張で時間ウィンドウが指定され、かつその時間ウィンドウがゼロよりも大きい場合、**sec** 変数および **msec** 変数は、この時間ウィンドウの最も古いサンプルと最新のサンプルとの実際の時間の差分です。

is_percent 引数が FALSE であり、イベント登録 Tcl コマンド拡張で sec 引数と msec 引数が 0 に指定されているか、または指定されていない場合は、次のようになります。

- **used** は最新のサンプルで使用されたメモリの合計です。
- **avail** は最新のサンプルで使用可能なメモリの合計です。
- **diff** は 0 です。
- **sec** と **msec** は両方とも 0 です。

is_percent 引数が FALSE であり、イベント登録 Tcl コマンド拡張で時間ウィンドウがゼロよりも大きい値に指定されている場合は、次のようになります。

- **used** は指定された時間ウィンドウで使用された合計メモリ サンプル値の平均です。
- **avail** は 0 です。
- **diff** は 0 です。
- **sec** および **msec** は、両方とも、この時間ウィンドウ内の使用されたメモリ合計の最も古いサンプルのタイムスタンプと最新サンプルのタイムスタンプの実際の時間の差分です。

is_percent 引数が TRUE であり、イベント登録 Tcl コマンド拡張で時間ウィンドウがゼロよりも大きい値に指定されている場合は、次のようになります。

- **used** は 0 です。

- **avail** は 0 です。
- **diff** 指定された時間ウィンドウの、使用された最も古いメモリ サンプルの合計と使用された最新のメモリ サンプルの合計とのパーセンテージによる差分です。
- **sec** および **msec** は、両方とも、この時間ウィンドウ内の使用されたメモリ合計の最も古いサンプルのタイムスタンプと最新サンプルのタイムスタンプの実際の時間の差分です。

is_percent 引数が TRUE であり、イベント登録 Tcl コマンド拡張で sec 引数と msec 引数が 0 に指定されているか、または指定されていない場合は、次のようになります。

- **used** は 0 です。
- **avail** は 0 です。
- **diff** は今まで収集された、使用された最初のメモリ サンプルの合計と、使用された最新のメモリ サンプルの合計との間のパーセンテージによる差です。
- **sec** および **msec** は、今まで収集された使用されたメモリ合計の最初のサンプルのタイムスタンプと、使用されたメモリ合計の最新サンプルのタイムスタンプ間の実際の時間の差です。

（注） サブイベントの説明内部では、各引数は、位置に依存しません。

第 94 章

EEM イベントの Tcl コマンド拡張

次の表記法が、Tcl コマンド拡張ページで説明されている構文に使用されます。

- 任意の引数は、たとえば次の例のように、角カッコ内に示されます。

[type ?]

- 疑問符（?）は入力する変数を表します。
- 引数間の選択肢は、たとえば次の例のように、パイプ文字で示されます。

priority low|normal|high

（注） すべての EEM Tcl コマンド拡張について、エラーがあった場合、戻される Tcl 結果文字列には、エラー情報が含まれます。

（注） 数値範囲が指定されていない引数は、–2147483648 から 2147483647 までの整数から取得されます。

event_completion

トリガーしたイベントのサービスが行われている EEM サーバーに、通知を送信します。イベントでは、このイベント インスタンスの **return_code** である 1 つの引数のみが使用されます。

構文

```
event_completion status ?
```

引数

status	（必須）このイベントインスタンスの終了ステータス（return_code）。ゼロの値によって、エラーがないことが示され、他のすべての整数によって、エラーが示されます。

結果文字列

なし

_cerrno を設定

なし

event_completion_with_wait

event_completion_with_wait コマンドは、2 つのコマンド、**event_completion** と **event_wait** を使いやすいように 1 つのコマンドに組み合わせたものです。

event_completion コマンドによって、ポリシーをトリガーしたイベントに対してポリシーがサービスを実行したことが EEM サーバーに通知されます。イベントでは、このイベント インスタンスの **return_code** である 1 つの引数のみが使用されます。

event_wait ポリシーがスリープ状態になります。Tcl ポリシーで、新しいイベントを通知する新しい信号を受信すると、ポリシーは使用状態になり、再度スリープ状態に戻ります。このループが継続されます。**event_wait** ポリシーは、**event_completed** ポリシーの前に起動され、エラーが発生して、ポリシーが終了します。

構文

```
event_completion_with_wait status ? [refresh_vars]
```

引数

status	（必須）このイベント インスタンスの exit_status（return_code）。ゼロの値は、エラーがないことを示します。他のすべての整数は、エラーを示します。
refresh_vars	（任意）組み込み変数と環境変数は、このイベントインスタンス中に EEM Policy Director からアップデート（リフレッシュ）する必要があるかどうかを示します。

結果文字列

なし

_cerrno を設定

可

使用例

この 1 つのコマンドを使用した前述の例の類似例を示します。

```
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set i 1
while {1 == 1} {  # Start high performance policy loop
   array set arr_einfo [event_reqinfo]
   if {$_cerrno != 0} {
       set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
       $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
       error $result
   }
   action_syslog msg "event $i serviced" priority info
   if {$i == 5} {
       action_syslog msg "Exiting after servicing 5 events" priority info
       exit 0
   }
   incr i
   array set _event_state_arr [event_completion_with_wait status 0 refresh_vars 1]
   if {$_event_state_arr(event_state) != 0} {
       action_syslog msg "Exiting: failed event_state " \
                         " $event_state_arr(event_state)" priority info
       exit 0
   }
}
```

（注） 実行される設定の出力は、event_publish Tcl コマンドと同じです。

event_publish

アプリケーション固有のイベントをパブリッシュします。

構文

```
event_publish sub_system ? type ? [arg1 ?] [arg2 ?] [arg3 ?] [arg4 ?]
```

引数

sub_system	（必須）アプリケーション固有のイベントをパブリッシュしたEEMポリシーに割り当てられる番号。他のすべての番号はCiscoでの使用のために予約されているため、番号は 798 に設定されます。

type	（必須）指定されたコンポーネント内のイベント サブタイプ。sub_system 引数および type 引数によって、アプリケーション イベントが一意に識別されます。1 ～ 4294967295 の範囲の整数である必要があります。
[arg1 ?]-[arg4 ?]	（任意）4 つのアプリケーション イベントのパブリッシャの文字列データ。

結果文字列

なし

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

使用例

次に、ある機能（Tcl ステートメントの所定のグループによって CPU 時間の長さを測定するなど）を実行するため、**event_publish** Tcl コマンド拡張を使用してスクリプトを *n* 回、反復して実行する例を示します。この例では、2 つの Tcl スクリプトが使用されます。

Script1 によって、タイプ 9999 EEM イベントがパブリッシュされ、Script2 の 1 回目の実行が行われます。Script1 は、none イベントとして登録され、Cisco IOS CLI **event manager run** コマンドを使用して実行されます。Script2 は、タイプ 9999 の EEM アプリケーション イベントとして登録され、このスクリプトによって、アプリケーションによってパブリッシュされた arg1 データ（繰り返し回数）が、EEM 環境変数 test_iterations の値を超過したかどうかがチェックされます。test_iterations の値が超えた場合、スクリプトによってメッセージが書き込まれ、終了します。これ以外の場合、スクリプトによって残りの文が実行され、別の実行が再スケジュールされます。Script2 の CPU 使用率を測定するには、10 の倍数である test_iterations の値を使用して、Script2 によって使用される CPU 時間の平均の長さを計算します。

Tcl スクリプトを実行するには、次の Cisco IOS コマンドを使用します。

```
configure terminal
 event manager environment test_iterations 100
 event manager policy script1.tcl
 event manager policy script2.tcl
 end
event manager run script1.tcl
```

Tcl スクリプト Script2 によって、100 回実行されます。余分な処理なしてスクリプトを実行し、CPU 使用率の平均を導き出し、次に余分な処理を追加して、テストを繰り返す場合、以降の CPU 使用率から前の CPU 使用率を差し引き、余分な処理の平均を調べることができます。

Script1（script1.tcl）

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# Query the event info.
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}

action_syslog priority info msg "EEM application_publish test start"
if {$_cerrno != 0} {
    set result [format \
          "component=%s; subsys err=%s; posix err=%s;\n%s" \
          $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}

# Cause the first iteration to run.
event_publish sub_system 798 type 9999 arg1 0
if {$_cerrno != 0} {
    set result [format \
          "component=%s; subsys err=%s; posix err=%s;\n%s" \
          $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

Script2（script2.tcl）

```
::cisco::eem::event_register_appl sub_system 798 type 9999

# Check if all the required environment variables exist.
# If any required environment variable does not exist, print out an error msg and quit.
if {![info exists test_iterations]} {
    set result \
      "Policy cannot be run: variable test_iterations has not been set"
    error $result $errorInfo
}

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

# Query the event info.
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
# Data1 contains the arg1 value used to publish this event.
set iter $arr_einfo(data1)

# Use the arg1 info from the previous run to determine when to end.
if {$iter >= $test_iterations} {
    # Log a message.
    action_syslog priority info msg "EEM application_publish test end"
    if {$_cerrno != 0} {
      set result [format \
            "component=%s; subsys err=%s; posix err=%s;\n%s" \
            $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
      error $result
    }
```

```
    exit 0
}
set iter [expr $iter + 1]

# Log a message.
set msg [format "EEM application_publish test iteration %s" $iter]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
      $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}

# Do whatever processing that you want to measure here.

# Cause the next iteration to run. Note that the iteration is passed to the
# next operation as arg1.
event_publish sub_system 798 type 9999 arg1 $iter
if {$_cerrno != 0} {
    set result [format \
          "component=%s; subsys err=%s; posix err=%s;\n%s" \
          $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

event_wait

Tclポリシーがスリープ状態になります。Tclポリシーで、新しいイベントを通知する新しい信号を受信すると、ポリシーは使用状態になり、再度スリープ状態に戻ります。このループが継続されます。**event_wait** ポリシーは、**event_completed** ポリシーの前に起動され、エラーが発生して、ポリシーが終了します。

構文

```
event_wait [refresh_vars]
```

引数

refresh_vars	（任意）組み込み変数と環境変数は、このイベントインスタンス中にEEM Policy Directorからアップデート（リフレッシュ）する必要があるかどうかを示します。

結果文字列

なし

_cerrno を設定

なし

使用例

event_wait イベント ディテクタは、**event_state** という名前の単一要素でアレイ タイプ値を返します。Event_state は、イベントの処理中にエラーが発生したかどうかを示す EEM サーバー

から戻される値です。この場合のエラーの例は、ユーザーがイベントインスタンスを処理するときに、**event_completion** を設定する前に **event_wait** を設定した場合のエラーを示しています。

次に、**event_completion** Tcl コマンドと**event_wait** コマンドの両方を使用した出力例を示します。

```
::cisco::eem::event_register_syslog tag e1 occurs 1 pattern CLEAR maxrun 0
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set i 1
while {1 == 1} {  # Start high performance policy loop
   array set arr_einfo [event_reqinfo]
   if {$_cerrno != 0} {
       set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
       $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
       error $result
   }
   action_syslog msg "event $i serviced" priority info
   if {$i == 5} {
       action_syslog msg "Exiting after servicing 5 events" priority info
       exit 0
   }
   incr i
   event_completion status 0
   array set _event_state_arr [event_wait refresh_vars 0]
   if {$_event_state_arr(event_state) != 0} {
       action_syslog msg "Exiting: failed event_state " \
                         " $event_state_arr(event_state)" priority info
       exit 0
   }
}
```

次に、実行コンフィギュレーションの例を示します。

```
Device#
01:00:44: %SYS-5-CONFIG_I: Configured from console by consoleclear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:00:49: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:49: %HA_EM-6-LOG: high_perf_example.tcl: event 1 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:00:53: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:53: %HA_EM-6-LOG: high_perf_example.tcl: event 2 serviced
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:00:56: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:56: %HA_EM-6-LOG: high_perf_example.tcl: event 3 serviced
Device#
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:00:59: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
```

```
01:00:59: %HA_EM-6-LOG: high_perf_example.tcl: event 4 serviced
01:00:59: %HA_EM-6-LOG: high_perf_example.tcl: Exiting after servicing 5 events
Device#
Device#
Device#copy tftp disk1:
Address or name of remote host [dirt]?
Source filename [user/eem_scripts/high_perf_example.tcl]?
Destination filename [high_perf_example.tcl]?
%Warning:There is a file already existing with this name
Do you want to over write? [confirm]
Accessing tftp://dirt/user/eem_scripts/high_perf_example.tcl...
Loading user/eem_scripts/high_perf_example.tcl from 192.0.2.19 (via FastEthernet0/0): !
[OK - 909 bytes]
909 bytes copied in 0.360 secs (2525 bytes/sec)
Device#
Device#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)#no event manager policy high_perf_example.tcl
Device(config)#event manager po high_perf_example.tcl
Device(config)#end
Device#
Device#
Device#
Device#
01:02:19: %SYS-5-CONFIG_I: Configured from console by consoleclear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:02:23: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
Device#
01:02:23: %HA_EM-6-LOG: high_perf_example.tcl: event 1 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:26: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:26: %HA_EM-6-LOG: high_perf_example.tcl: event 2 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:29: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:29: %HA_EM-6-LOG: high_perf_example.tcl: event 3 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:33: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
01:02:33: %HA_EM-6-LOG: high_perf_example.tcl: event 4 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
Device#
01:02:36: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:36: %HA_EM-6-LOG: high_perf_example.tcl: event 5 serviced
01:02:36: %HA_EM-6-LOG: high_perf_example.tcl: Exiting after servicing 5 events
Device#
```

また、イベントがサービスされ、次のイベントの到着を待っている間に、**show event manager policy active** コマンドによって、次の出力が表示されます。

```
Device#show event manager policy active
Key: p - Priority        :L - Low, H - High, N - Normal, Z - Last
     s - Scheduling node :A - Active, S - Standby
default class - 1 script event
 no.  job id      p s status  time of event             event type          name
 1    11          N A wait    Mon Oct20 14:15:24 2008   syslog
high_perf_example.tcl
```

前述の例では、ステータスは待ち状態です。これは、ポリシーが次のイベントの到着を待っていることを示します。

第 95 章

EEM ライブラリのデバッグ コマンド拡張

cli_debug

コマンドライン インターフェイス（CLI）のデバッグ文を、Syslog に出力します。**debug event manager tcl cli_library** Cisco IOS コマンドが有効な場合に、この Tcl コマンド拡張を使用すると、CLI デバッグステートメントが syslog に出力されます。

構文

```
cli_debug spec_string debug_string
```

引数

spec_string	（必須）spec_string 引数が使用され、デバッグ文のタイプを示します。
debug_string	（必須）debug_string 引数が使用され、デバッグ テキストを示します。

結果文字列

なし

_cerrno を設定

なし

smtp_debug

シンプル メール転送プロトコル（SMTP）のデバッグ文を、Syslog に出力します。**debug event manager tcl smtp_library** のコマンドライン インターフェイス（CLI）コマンドが有効な場合に、この Tcl コマンド拡張によって、SMTP デバッグ文が Syslog に出力されます。

構文

```
smtp_debug spec_string debug_string
```

引数

spec_string	（必須）spec_string 引数が使用され、デバッグ文のタイプを示します。
debug_string	（必須）debug_string 引数が使用され、デバッグ テキストを示します。

結果文字列

なし

_cerrno を設定

なし

第 96 章

EEM 複数イベントサポートの Tcl コマンド拡張

次の表記法が、Tcl コマンド拡張ページで説明されている構文に使用されます。

- 任意の引数は、たとえば次の例のように、角カッコ内に示されます。

[type ?]

- 疑問符（?）は入力する変数を表します。
- 引数間の選択肢は、たとえば次の例のように、パイプ文字で示されます。

priority low|normal|high

（注） すべての EEM Tcl コマンド拡張について、エラーがあった場合、戻される Tcl 結果文字列には、エラー情報が含まれます。

（注） 数値範囲が指定されていない引数は、–2147483648 から 2147483647 までの整数から取得されます。

attribute

複雑なイベントを指定します。

構文

```
attribute tag ? [occurs ?]
```

引数

tag	イベントを関連付けるために **attribute** コマンドで使用できる *event-tag* 引数を使用して、タグを指定します。
occurs	（任意）EEM イベントがトリガーされる前の発生数を指定します。指定されない場合、EEM イベントは 1 回目から発生します。範囲は 1 ～ 4294967295 です。

結果文字列

なし

_cerrno を設定

なし

correlate

イベントおよびトラックされるオブジェクトに関連する、1つの複雑なイベントを構築し、ブール値のロジックを使用します。

構文

```
correlate event ? track ? [andnot | and | or] event ? track ?
```

引数

event	スクリプト内で複数のイベント文をサポートするために、**trigger** コマンドで使用できるイベントを指定します。 *event-tag* 引数に関連付けられているイベントが、**trigger** コマンドによって指定されて何度も発生する場合、結果は真です。これ以外の場合、結果は偽です。
track	トラックするイベント オブジェクト番号を指定します。指定できる範囲は 1 ～ 500 です。 トラックされるオブジェクトが設定されている場合、評価の結果は真です。トラックされるオブジェクトが未設定または未定義の場合、評価の結果は偽です。この結果は、オブジェクトの状態には関係ありません。
andnot	（任意）イベント 1 が発生した場合にアクションが実行され、さらに、イベント 2 およびイベント 3 が一緒に発生した場合にはアクションが実行されないよう、指定します。

および	（任意）イベント 1 が発生した場合にアクションが実行され、さらに、イベント 2 およびイベント 3 が一緒に発生した場合にアクションが実行されるよう、指定します。 （注）　「and」を使用して、トラップや syslog メッセージなどのイベントをグループ化した場合、デフォルトのトリガー発生時間枠は 3 分です。
または	（任意）イベント 1 が発生した場合にアクションが実行されるか、または、イベント 2 およびイベント 3 が一緒に発生した場合にアクションが実行されるよう、指定します。

結果文字列

なし

_cerrno を設定

なし

trigger

Embedded Event Manager（EEM）イベントの複数イベントの設定機能を指定します。複数イベントは、1つまたは複数のイベント発生、1つまたは複数のトラックされるオブジェクト状態、および発生するイベントの時間を起動できるイベントです。イベントは指定されたパラメータに基づいて発生します。

構文

```
trigger [occurs ?] [period ?] [period-start ?] [delay ?]
```

引数

occurs	（任意）EEM イベントが発生する前に発生した合計相関回数を指定します。数が指定されない場合、EEM イベントは 1 回目から発生します。範囲は 1 ～ 4294967295 です。
period	（任意）1 つまたは複数が発生する必要がある間の、秒単位、および、任意でミリ秒単位での、時間の間隔。これは、ssssssssss[.mmm] 形式で指定します。ssssssssss は、0 ～ 4294967295 の秒数を表す整数で、mmm は 0 ～ 999 のミリ秒数を表す整数である必要があります。
period-start	（任意）イベント相関ウィンドウの開始を指定します。指定されない場合、最初の CRON 期間の発生後、イベント監視はイネーブルにされます。

delay	（任意）すべての条件が真の場合にイベントの発生後の秒数とミリ秒数（任意）を指定します（sssssssss[.mmm]形式で指定します。sssssssss は、0 ～ 4294967295 の秒数を表す整数で、mmm は 0 ～ 999 のミリ秒数を表す整数である必要があります）。

結果文字列

なし

_cerrno を設定

なし

第 97 章

EEM SMTP ライブラリのコマンド拡張

すべてのシンプル メール転送プロトコル（SMTP）ライブラリ コマンドは、::cisco::lib 名前空間に属します。

このライブラリを使用するには、ユーザーは、電子メール テンプレート ファイルを用意する必要があります。テンプレートファイルに Tcl グローバル変数を含めると、**event manager environment Cisco IOS** コマンドライン インターフェイス（CLI）コンフィギュレーション コマンドを使用して電子メールサービスと電子メールテキストを設定できるようになります。電子メールテンプレートファイルでグローバル変数を置き換え、設定された電子メールサーバーを使用して、設定された To アドレス、CC アドレス、From アドレス、および Subject 行プロパティで必要な電子メールコンテキストを送信するには、このライブラリにあるコマンドを使用します。

電子メール テンプレート

電子メール テンプレート ファイルの形式は、次のとおりです。

（注） RFC 2554 に基づき、SMTP 電子メール サーバー名 Mailservername には、username:password@host、username@host、または host のテンプレート形式のいずれか 1 つを使用できます。

```
Mailservername:<space><the list of candidate SMTP server addresses>
From:<space><the e-mail address of sender>
To:<space><the list of e-mail addresses of recipients>
Cc:<space><the list of e-mail addresses that the e-mail will be copied to>
Sourceaddr:<space><the IP addresses of the recipients>
Subject:<subject line>
<a blank line>
<body>
```

（注） テンプレートには、通常、設定のための Tcl グローバル変数が含まれていることに注意してください。

Tcl ポリシーでは、電子メール テンプレートの「Port」行でポート番号を指定できます。ポートを指定しなかった場合、デフォルトのポート 25 が使用されます。

次に、サンプル E メール テンプレート ファイルを挙げます。

```
Mailservername: $_email_server
From: $_email_from
To: $_email_to
Cc: $_email_cc
Sourceaddr: $_email_ipaddr
Port: <port number>
Subject: From router $routername: Process terminated
process name: $process_name
subsystem: $sub_system
exit status: $exit_status
respawn count: $respawn_count
```


- smtp_send_email （2350 ページ）
- smtp_subst （2351 ページ）

smtp_send_email

電子メール テンプレート ファイルのテキストが、すべてのグローバル変数ですでに置き換えられている場合、シンプル メール転送プロトコル（SMTP）を使用して電子メールを送信します。電子メール テンプレートによって、候補メール サーバーのアドレス、To アドレス、CC アドレス、From アドレス、件名の行、および電子メールの本文が指定されます。

（注） ライブラリが、リストにあるサーバーの 1 つに接続できるまで、サーバーへの接続が、1 つ 1 つ試行されるよう、候補電子メール サーバーのリストを用意できます。

構文

```
smtp_send_email text
```

引数

text	（必須）すべてのグローバル変数ですでに置き換えられた、E メール テンプレート ファイルのテキスト。

結果文字列

なし

_cerrno を設定

- 1 行目の形式が間違っている：Mailservername：サーバー名のリスト。
- 2 行目の形式が間違っている：From：送信元アドレス。

- 3 行目の形式が間違っている：To：送信先アドレスのリスト。
- 4 行目の形式が間違っている：CC：コピー送信先アドレスのリスト。
- メール サーバーへの接続エラー：リモート サーバーによって $sock が閉じられている（$sock はメール サーバーに開かれているソケットの名前）。
- メール サーバーへの接続エラー：$sock 応答コードが service ready greeting ではなく $k である（$sock はメール サーバーに開かれているソケットの名前、$k は $sock の応答コード）。
- メール サーバーへの接続エラー：すべてのメール サーバー候補に接続できない。
- メール サーバーからの接続解除エラー：リモート サーバーによって $sock が閉じられている（$sock はメール サーバーに開かれているソケットの名前）。

サンプル スクリプト

電子メールテンプレートですべての必要なグローバル変数が定義された後には、次のようになります。

```
if [catch {smtp_subst [file join $tcl_library email_template_sm]} result] {
    puts stderr $result
    exit 1
}
if [catch {smtp_send_email $result} result] {
    puts stderr $result
    exit 1
}
```

smtp_subst

電子メール テンプレート ファイル e-mail_template の場合、ファイルにある各グローバル変数を、そのユーザー定義値によって置き換えます。置換後に、ファイルのテキストを返します。

構文

```
smtp_subst e-mail_template
```

引数

e-mail_template	（必須）グローバル変数が、ユーザー定義値によって置き換えられる必要がある、電子メール テンプレート ファイルの名前。ファイル名の例は /disk0://example.template で、スロット 0 の ATA フラッシュ ディスクの上位レベルディレクトリにある example.template という名前のファイルを表します。

結果文字列

すべてのグローバル変数で置き換えられた、電子メール テンプレート ファイルのテキスト。

_cerrno を設定

- 電子メール テンプレート ファイルを開けられない。
- 電子メール テンプレート ファイルを閉じられない。

第 98 章

EEM システム情報の Tcl コマンド拡張

次の表記法が、Tcl コマンド拡張ページで説明されている構文に使用されます。

- 任意の引数は、たとえば次の例のように、角カッコ内に示されます。

[type ?]

- 疑問符（?）は入力する変数を表します。
- 引数間の選択肢は、たとえば次の例のように、パイプ文字で示されます。

priority low|normal|high

（注） すべての EEM システム情報コマンド（**sys_reqinfo_xxx**）では、Set _cerrno セクションが yes に設定されています。

（注） すべての EEM Tcl コマンド拡張について、エラーがあった場合、戻される Tcl 結果文字列には、エラー情報が含まれます。

（注） 数値範囲が指定されていない引数は、–2147483648 から 2147483647 までの整数から取得されます。

sys_reqinfo_cli_freq

すべてのコマンドライン インターフェイス（CLI）イベントの頻度情報を問い合わせます。

構文

```
sys_reqinfo_cli_freq
```

引数

なし

結果文字列

```
rec_list {{CLI frequency string 0},{CLI frequency str 1}, ...}
```

各 CLI の頻度の文字列は、次のとおりです。

time_sec %ld time_msec %ld match_count %u raise_count %u occurs %u period_sec %ld period_msec %ld pattern {%s}

rec_list	CLI イベント頻度リストの開始をマークします。
time_sec time_msec	この CLI イベントが発生した最後の時刻。
match count	CLI イベントによって指定されたパターンが、CLI コマンドによって照会される回数。
raise_count	この CLI イベントが発生した回数。次のフィールドは、CLI イベント指定に関する情報です。 • sync：「yes」は、イベント パブリッシュが同期的に実行される必要があることを意味します。Event Manager Server がイベントのパブリッシュを完了したときに、イベントディテクタが通知されます。Event Manager Server は、CLI コマンドが実行される必要があるかどうかを示すコードを返します。 • skip：「yes」は、sync フラグが設定されているときに、CLI コマンドは実行してはいけないことを意味します。
occurs	イベントが発生する前の発生回数。この引数が指定されない場合、イベントは 1 回目から発生します。

period_sec period_msec	イベントを発生させるには、発生回数が POSIX タイマー ユニットのこの数以内である必要があります。この引数が指定されない場合は、適用されません。
pattern	CLI コマンドのパターン マッチの実行に使用される正規表現。

_cerrno を設定

対応

sys_reqinfo_cli_history

コマンドライン インターフェイス（CLI）コマンドの履歴を問い合わせます。

構文

```
sys_reqinfo_cli_history
```

引数

なし

結果文字列

```
rec_list {{CLI history string 0}, {CLI history str 1},...}
```

各 CLI の履歴の文字列は、次のとおりです。

```
time_sec %ld time_msec %ld cmd {%s}
```

rec_list	CLI コマンド履歴リストの開始をマークします。
time_sec time_msec	CLI コマンドが実行された時刻。
cmd	CLI コマンドのテキスト。

_cerrno を設定

対応

sys_reqinfo_cpu_all

指定された期間で、指定された順序で、上位プロセスの CPU 使用率（POSIX プロセスと IOS プロセスの両方）を問い合わせます。この Tcl コマンド拡張は、ソフトウェア モジュール方式イメージでのみサポートされます。

構文

```
sys_reqinfo_cpu_all order cpu_used [sec ?] [msec ?] [num ?]
```

引数

order	（必須）プロセスの CPU 使用率のソートに使用される順序。
cpu_used	（必須）指定されたウィンドウの、CPU 使用率の平均が、降順でソートされるよう、指定します。
sec msec	（任意）CPU 使用率の平均が計算される、秒単位およびミリ秒単位での時間。0 から 4294967295 の範囲の整数である必要があります。指定されない場合か、または、sec と msec の両方が 0 と指定される場合、最新の CPU サンプルが使用されます。
num	（任意）表示される、ソートされたプロセスのリストの上位からのエントリの数。1 ～ 4294967295 の範囲の整数である必要があります。デフォルト値は 5 です。

結果文字列

```
rec_list {{process CPU info string 0},{process CPU info string 1}, ...}
```

各プロセスの CPU 情報文字列は、次のとおりです。

```
pid %u name {%s} cpu_used %u
```

rec_list	プロセス CPU 情報リストの開始をマークします。
pid	プロセス ID。
name	プロセス名。
cpu_used	sec と msec が、ゼロよりも大きい数で指定される場合、平均パーセンテージは、指定された時間のプロセス CPU 使用率から計算されるよう、指定します。sec と msec が、両方ともゼロか、または指定されない場合。平均パーセンテージは、最新のサンプルのプロセス CPU 使用率から計算されます。

_cerrno を設定

対応

sys_reqinfo_crash_history

クラッシュしたすべてのプロセスのプロセス情報を問い合わせます。この Tcl コマンド拡張は、ソフトウェア モジュール方式イメージでのみサポートされます。

構文

```
sys_reqinfo_crash_history
```

引数

なし

結果文字列

```
rec_list {{crash info string 0},{crash info string 1}, ...}
Where each crash info string is:
job_id %u name {%s} respawn_count %u fail_count %u dump_count %u
inst_id %d exit_status 0x%x exit_type %d proc_state {%s} component_id 0x%x
crash_time_sec %ld crash_time_msec %ld
```

job_id	システム マネージャによってプロセスに割り当てられるジョブ ID。1 ～ 4294967295 の整数。
name	プロセス名。
respawn_count	プロセスの再起動の合計回数。
fail_count	プロセスの再起動試行の回数。プロセスが正常に再起動されると、このカウントはゼロにリセットされます。
dump_count	実行されたコア ダンプの回数。
inst_id	プロセス インスタンス ID。
exit_status	プロセスの最後の終了ステータス。
exit_type	最後の終了タイプ。
proc_state	Sysmgr プロセスの状態。error、forced_stop、hold、init、ready_to_run、run、run_rnode、stop、waitEOltimer、wait_rnode、wait_spawntimer、wait_tpl の 1 つです。
component_id	プロセスが属するコンポーネントのコンポーネント ID に割り当てられているバージョン マネージャ。
crash_time_sec crash_time_msec	1970 年 1 月 1 日以降の秒およびミリ秒の単位で、プロセスがクラッシュした最後の時刻を表します。

_cerrno を設定

対応

sys_reqinfo_mem_all

指定された期間で、指定された順序で、上位プロセスのメモリの使用状況（POSIX と IOS の両方）を問い合わせます。この Tcl コマンド拡張は、ソフトウェア モジュール方式イメージでのみサポートされます。

構文

```
sys_reqinfo_mem_all order allocates|increase|used [sec ?] [msec ?] [num ?]
```

引数

order	（必須）プロセスのメモリの使用状況のソートに使用される順序。
allocates	（必須）指定された時間ウィンドウの期間に、メモリの使用状況が、プロセス割り当ての数によって降順でソートされるよう、指定します。
increase	（必須）指定された時間ウィンドウの期間に、メモリの使用状況が、プロセスで増えたメモリのパーセンテージによって降順でソートされるよう、指定します。
used	（必須）メモリが、プロセスによって使用される現在のメモリによってソートされるよう、指定します。
sec msec	（任意）プロセスでのメモリの使用状況が計算される、秒単位およびミリ秒単位での時間。0 から 4294967295 の範囲の整数である必要があります。sec と msec の両方が指定され、ゼロではない場合、割り当て数は、該当する時間で収集された最も古いサンプルと最新のサンプルでの、割り当て数の差です。パーセンテージは、該当する時間で収集された最も古いサンプルと最新のサンプルとの、パーセンテージの差分として計算されます。指定されない場合か、または、sec と msec の両方が 0 と指定される場合、収集された最初のサンプルが、最も古いサンプルとして使用されます。つまり、時間は、起動から現在時までの時間で設定されます。
num	（任意）表示される、ソートされたプロセスのリストの上位からのエントリの数。1 ～ 4294967295 の範囲の整数である必要があります。デフォルト値は 5 です。

結果文字列

```
rec_list {{process mem info string 0},{process mem info string 1}, ...}
```

各プロセスのメモリ情報文字列は、次のとおりです。

```
pid %u name {%s} delta_allocs %d initial_alloc %u current_alloc %u percent_increase %d
```

rec_list	プロセスのメモリの使用状況情報リストの開始をマークします。
pid	プロセス ID。

name	プロセス名。
delta_allocs	該当する期間で収集された、最も古いサンプルと最新のサンプルでの、割り当て数の差として、差を指定します。
initial_alloc	時間の開始時にプロセスによって使用される、キロバイト単位での、メモリの容量を指定します。
current_alloc	プロセスによって使用される、キロバイト単位での、メモリの容量を指定します。
percent_increase	該当する時間で収集された最も古いサンプルと最新のサンプルとの、使用メモリのパーセンテージの差分を指定します。パーセンテージの差は、current_alloc から initial_alloc の 100 を差し引いた数として、および、initial_alloc で割った数として、表すことができます。

_cerrno を設定

対応

sys_reqinfo_proc

1 つの POSIX プロセスに関する情報を問い合わせます。この Tcl コマンド拡張は、ソフトウェア モジュール方式イメージでのみサポートされます。

構文

```
sys_reqinfo_proc job_id ?
```

引数

job_id	（必須）システム マネージャによってプロセスに割り当てられるジョブ ID。1 ～ 4294967295 の範囲の整数である必要があります。

結果文字列

```
job_id %u component_id 0x%x name {%s} helper_name {%s} helper_path {%s} path {%s}
node_name {%s} is_respawn %u is_mandatory %u is_hold %u dump_option %d
max_dump_count %u respawn_count %u fail_count %u dump_count %u
last_respawn_sec %ld last_respawn_msec %ld inst_id %u proc_state %s
level %d exit_status 0x%x exit_type %d
```

job_id	システム マネージャによってプロセスに割り当てられるジョブ ID。1 ～ 4294967295 の整数。
component_id	プロセスが属するコンポーネントのコンポーネント ID に割り当てられているバージョン マネージャ。

name	プロセス名。
helper_name	ヘルパー プロセスの名前。
helper_path	ヘルパー プロセスの実行可能パス。
path	プロセスの実行可能パス。
node_name	プロセスが属するノードのノード名に割り当てられているシステム マネージャ。
is_respawn	プロセスが再生成できることを指定するフラグ。
is_mandatory	プロセスが実行され続ける必要があることを指定するフラグ。
is_hold	APIによって呼び出されるまでプロセスが再生成されることを指定するフラグ。
dump_option	コア ダンプのオプション。
max_dump_count	許可されるコア ダンプの最大数。
respawn_count	プロセスの再起動の合計回数。
fail_count	プロセスの再起動試行の回数。プロセスが正常に再起動されると、このカウントはゼロにリセットされます。
dump_count	実行されたコア ダンプの回数。
last_respawn_sec last_respawn_msec	1970 年 1 月 1 日以降の POSIX タイマー ユニットでの秒およびミリ秒の単位で、プロセスが開始された最後の時刻を表します。
inst_id	プロセス インスタンス ID。
proc_state	Sysmgr プロセスの状態。error、forced_stop、hold、init、ready_to_run、run、run_rnode、stop、waitEOltimer、wait_rnode、wait_spawntimer、wait_tpl の 1 つです。
level	プロセス実行レベル。
exit_status	プロセスの最後の終了ステータス。
exit_type	最後の終了タイプ。

_cerrno を設定

対応

sys_reqinfo_proc_all

すべての POSIX プロセスの情報を問い合わせます。この Tcl コマンド拡張は、ソフトウェアモジュール方式イメージでのみサポートされます。

構文

```
sys_reqinfo_proc_all
```

引数

なし

結果文字列

```
rec_list {{process info string 0}, {process info string 1},...}
```

各プロセスの情報文字列は、**sysreq_info_proc** Tcl コマンド拡張の結果文字列と同じです。

_cerrno を設定

対応

sys_reqinfo_routername

デバイス名を問い合わせます。

構文

```
sys_reqinfo_routername
```

引数

なし

結果文字列

```
routername %s
```

この場合、routername がデバイスの名前です。

_cerrno を設定

対応

sys_reqinfo_snmp

簡易ネットワーク管理プロトコル（SNMP）オブジェクトIDによって指定されたエンティティの値を問い合わせます。

構文

```
sys_reqinfo_snmp oid ? get_type exact|next
```

引数

oid	（必須）ドット付き表記での SNMP OID（たとえば、1.3.6.1.2.1.2.1.0）。
get_type	（必須）指定された OID に適用する必要がある SNMP 取得操作のタイプ。get_type が「exact」の場合、指定された OID の値が取得されます。get_type が「next」の場合、指定された OID の辞書順での後続値が取得されます。

結果文字列

```
oid {%s} value {%s}
```

oid	SNMP OID。
value	割り当てられたSNMPデータエレメントの値文字列。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされるPOSIX errno値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部EEMイベントディテクタポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 37)    FH_ENOSNMPDATA  (can't retrieve data from SNMP)
```

このエラーは、SNMP オブジェクト タイプのデータがなかったことを意味します。

```
(_cerr_sub_err = 51)    FH_ESTATSTYP  (invalid statistics data type)
```

このエラーは、SNMP 統計データ タイプが無効であったことを意味します。

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

このエラーは、イベント ディテクタが使用できなかったことを意味します。

sys_reqinfo_syslog_freq

すべての Syslog イベントの頻度情報を問い合わせます。

構文

```
sys_reqinfo_syslog_freq
```

引数

なし

結果文字列

```
rec_list {{event frequency string 0}, {log freq str 1}, ...}
```

各イベントの頻度の文字列は、次のとおりです。

```
time_sec %ld time_msec %ld match_count %u raise_count %u occurs %u
period_sec %ld period_msec %ld pattern {%s}
```

time_sec time_msec	1970 年 1 月 1 日以降の POSIX タイマー ユニットでの秒およびミリ秒の単位で、最後のイベントが発生した時刻を表します。
match_count	イベントの登録以降、この Syslog イベント指定によって指定されたパターンが、Syslog メッセージによって照会される回数。
raise_count	この Syslog イベントが発生した回数。
occurs	イベントを発生させるために必要な発生回数。指定されない場合、イベントは 1 回目から発生します。
period_sec period_msec	イベントを発生させるには、発生回数が POSIX タイマーユニットのこの数以内である必要があります。この引数が指定されない場合、時間のチェックは適用されません。
pattern	Syslog メッセージのパターン マッチの実行に使用される正規表現。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  9)    FH_EMEMORY  (insufficient memory for request)
```

このエラーは、メモリの内部 EEM 要求に障害が発生したことを意味します。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部 EEM イベント ディテクタ ポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 45)    FH_ESEQNUM  (sequence or workset number out of sync)
```

このエラーは、イベント ディテクタ シーケンスまたは作業セット番号が無効であったことを意味します。

```
(_cerr_sub_err = 46)    FH_EREGEMPTY  (registration list is empty)
```

このエラーは、イベント ディテクタ登録リストが空であったことを意味します。

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

このエラーは、イベント ディテクタが使用できなかったことを意味します。

sys_reqinfo_syslog_history

指定された Syslog メッセージの履歴を問い合わせます。

構文

```
sys_reqinfo_syslog_history
```

引数

なし

結果文字列

```
rec_list {{log hist string 0}, {log hist str 1}, ...}
```

各記録の履歴の文字列は、次のとおりです。

```
time_sec %ld time_msec %ld msg {%s}
```

time_sec time_msec	1970 年 1 月 1 日以降の秒およびミリ秒の単位で、メッセージが記録された時刻を表します。
msg	Syslog メッセージ。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部 EEM イベント ディテクタ ポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 44)    FH_EHISTEMPTY  (history list is empty)
```

このエラーは、履歴のリストが空であったことを意味します。

```
(_cerr_sub_err = 45)    FH_ESEQNUM  (sequence or workset number out of sync)
```

このエラーは、イベント ディテクタ シーケンスまたは作業セット番号が無効であったことを意味します。

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

このエラーは、イベント ディテクタが使用できなかったことを意味します。

第 99 章

EEM ユーティリティの Tcl コマンド拡張

次の表記法が、Tcl コマンド拡張ページで説明されている構文に使用されます。

- 任意の引数は、たとえば次の例のように、角カッコ内に示されます。

[type ?]

- 疑問符（?）は入力する変数を表します。
- 引数間の選択肢は、たとえば次の例のように、パイプ文字で示されます。

priority low|normal|high

（注） すべての EEM Tcl コマンド拡張について、エラーがあった場合、戻される Tcl 結果文字列には、エラー情報が含まれます。

（注） 数値範囲が指定されていない引数は、–2147483648 から 2147483647 までの整数から取得されます。

appl_read

Embedded Event Manager（EEM）アプリケーションの揮発性データを読み取ります。この Tcl コマンド拡張では、EEM アプリケーションの揮発性データの読み取りがサポートされます。EEM アプリケーションの揮発性データは、API をパブリッシュする EEM アプリケーションが使用される Cisco ソフトウェア プロセスによってパブリッシュすることができます。EEM アプリケーションの揮発性データは、EEM ポリシーによってパブリッシュできません。

（注） 現在、アプリケーション揮発性データをパブリッシュする Cisco ソフトウェアはありません。

構文

```
appl_read name ? length ?
```

引数

name	（必須）アプリケーションによってパブリッシュされる文字列データの名前。
length	（必須）読み取る文字列データの長さ。1 ～ 4294967295 の範囲の整数である必要があります。

結果文字列

```
data %s
```

data は、読み取られる、アプリケーションによってパブリッシュされた文字列データです。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  7)    FH_ENOSUCHKEY  (could not find key)
```

このエラーは、アプリケーション イベント ディテクタ情報キーまたはその他の ID が見つからなかったことを意味します。

```
(_cerr_sub_err =  9)    FH_EMEMORY  (insufficient memory for request)
```

このエラーは、メモリの内部 EEM 要求に障害が発生したことを意味します。

appl_reqinfo

Embedded Event Manager（EEM）から、前に保存された情報が取得されます。この Tcl コマンド拡張によって、一意のキーで前に保存された EEM からの情報の取得がサポートされます。これは、情報を取得するために指定する必要があります。情報の取得によって、その情報が EEM から削除されることに、注意してください。再度取得できるようにするには、再保存する必要があります。

構文

```
appl_reqinfo key ?
```

引数

キー	（必須）データの文字列キー。

結果文字列

```
data %s
```

data は、取得されるアプリケーション文字列データです。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  7)    FH_ENOSUCHKEY  (could not find key)
```

このエラーは、アプリケーション イベント ディテクタ情報キーまたはその他の ID が見つからなかったことを意味します。

appl_setinfo

Embedded Event Manager（EEM）に情報を保存します。この Tcl コマンド拡張によって、同じポリシーまたは別のポリシーによって、後で取得できる Embedded Event Manager への情報の保存がサポートされます。一意のキーを指定する必要があります。このキーを使用すると、情報を後で取得することができます。

構文

```
appl_setinfo key ? data ?
```

引数

キー	（必須）データの文字列キー。
data	（必須）保存するアプリケーション文字列データ。

結果文字列

なし

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  8)    FH_EDUPLICATEKEY  (duplicate appl info key)
```

このエラーは、アプリケーション イベント ディテクタ情報キーまたはその他の ID が重複していたことを意味します。

```
(_cerr_sub_err =  9)    FH_EMEMORY  (insufficient memory for request)
```

このエラーは、メモリの内部 EEM 要求に障害が発生したことを意味します。

```
(_cerr_sub_err = 34)    FH_EMAXLEN  (maximum length exceeded)
```

このエラーは、オブジェクト長またはオブジェクト数が、最大値を超えたことを意味します。

```
(_cerr_sub_err = 43)    FH_EBADLENGTH  (bad API length)
```

このエラーは、API メッセージ長が無効であったことを意味します。

counter_modify

カウンタの値を変更します。

構文

```
counter_modify event_id ? val ? op nop|set|inc|dec
```

引数

event_id	（必須）**register_counter** Tcl コマンド拡張によって返されるカウンタイベント ID。0 ～ 4294967295 の範囲の整数である必要があります。
val	（必須） （注）　op nop 引数値の組み合わせが指定されている以外は必須です。 • op が設定されている場合、この引数は、設定されるカウンタ値を表します。 • op が inc の場合、この引数は、カウンタを増やすために使用される値です。 • op が dec の場合、この引数は、カウンタを減らすために使用される値です。
op	（必須） • nop：現在のカウンタの値を取得します。 • set：カウンタの値を指定値に設定します。 • inc：カウンタの値を指定値分増やします。 • dec：カウンタの値を指定値分減らします。

結果文字列

```
val_remain %d
```

val_remain は、カウンタの現在の値です。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err = 11)    FH_ENOSUCHESID  (unknown event specification ID)
```

このエラーは、イベントが登録されたときか、またはイベントディテクタの内部イベント構造が破損したときに、イベント指定 ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部 EEM イベント ディテクタ ポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 30)    FH_ECTBADOPER  (bad counter threshold operator)
```

このエラーは、カウンタ イベント ディテクタの設定演算子または変更演算子が、無効であったことを意味します。

description

記録されたポリシーの簡単な説明を記述します。

構文

```
description ?
```

引数

line	（任意）1 文字から 240 文字で構成されるポリシーの簡単な説明。

結果文字列

なし

_cerrno を設定

可

使用例

説明文は、ポリシーの作成者によって入力されます。Tcl のイベント登録文の前または後に表示できます。ポリシーには、1 つの説明のみ使用できます。

（注） 1 つのポリシーに複数の説明文を登録した場合、障害が発生します。

次に、**event_register_syslog** ポリシーに簡単な説明が指定される例を示します。

```
::cisco::eem::description "This Tcl command looks for the word count in syslog messages."
::cisco::eem::event_register_syslog tag 1 ...
::cisco::eem::event_register_snmp_object tag 2 ...
::cisco::eem::trigger {
       ::cisco::eem::correlate event 1 and event 2
       ::cisco::eem::attribute tag 1 occurs 1
       ::cisco::eem::attribute tag 2 occurs 1
}
```

fts_get_stamp

最後にソフトウェアがブートされて以来の経過時間を返します。この Tcl コマンド拡張を使用すると、配列「nsec nnnn」に、ブート以降のナノ秒数が返されます。ここで、nnnn はナノ秒数です。

構文

```
fts_get_stamp
```

引数

なし

結果文字列

```
nsec %d
```

nsec は、ブート以降のナノ秒数です。

_cerrno を設定

なし

register_counter

カウンタを登録し、カウンタ イベント ID を返します。この Tcl コマンド拡張は、カウンタのパブリッシャによって使用され、イベント ID を使用してカウンタを操作する前に、この登録が実行されます。

構文

```
register_counter name ?
```

引数

name	（必須）操作されるカウンタの名前。

結果文字列

```
event_id %d
event_spec_id %d
```

event_id は、指定されたカウンタのカウンタイベント ID です。**unregister_counter** または **counter_modify** Tcl コマンド拡張によって、カウンタの操作に使用されます。event_spec_id 引数は、指定されたカウンタのイベント指定 ID です。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  4)    FH_EINITONCE  (Init() is not yet done, or done twice.)
```

このエラーは、EEM イベント ディテクタがその初期化を完了する前に、特定のイベントを登録する要求が行われたことを意味します。

```
(_cerr_sub_err =  6)    FH_EBADEVENTTYPE  (unknown EEM event type)
```

このエラーは、内部イベント指定で指定されたイベント タイプが無効であったことを意味します。

```
(_cerr_sub_err =  9)    FH_EMEMORY  (insufficient memory for request)
```

このエラーは、メモリの内部 EEM 要求に障害が発生したことを意味します。

```
(_cerr_sub_err = 10)    FH_ECORRUPT  (internal EEM API context is corrupt)
```

このエラーは、内部 EEM API コンテキスト構造が破損したことを意味します。

```
(_cerr_sub_err = 11)    FH_ENOSUCHESID  (unknown event specification ID)
```

このエラーは、イベントが登録されたときか、またはイベントディテクタの内部イベント構造が破損したときに、イベント指定 ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 12)    FH_ENOSUCHEID  (unknown event ID)
```

このエラーは、イベントが登録されたときか、またはイベントディテクタの内部イベント構造が破損したときに、イベント ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 16)    FH_EBADFMPPTR  (bad ptr to fh_p data structure)
```

このエラーは、各 EEM API コールで使用されるコンテキスト ポインタが不正確であったことを意味します。

```
(_cerr_sub_err = 17)    FH_EBADADDRESS  (bad API control block address)
```

このエラーは、EEM API に渡された制御ブロック アドレスが不正確であったことを意味します。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部 EEM イベント ディテクタ ポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 25)    FH_ESUBSEXCEED  (number of subscribers exceeded)
```

このエラーは、タイマーまたはカウンタのサブスクライバの数が、最大値を超えたことを意味します。

```
(_cerr_sub_err = 26)    FH_ESUBSIDXINV  (invalid subscriber index)
```

これは、サブスクライバの索引が無効であったことを意味します。

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

このエラーは、イベント ディテクタが使用できなかったことを意味します。

```
(_cerr_sub_err = 56)    FH_EFDCONNERR  (event detector connection error)
```

このエラーは、この要求を処理する EEM イベント ディテクタは使用できないことを意味します。

register_timer

タイマーを登録し、タイマー イベント ID を返します。この Tcl コマンド拡張は、カウンタのパブリッシャによって使用され、パブリッシャまたはサブスクライバとしての登録に、**event_register_timer** コマンド拡張が使用されなかった場合に、イベント ID を使用してタイマーを操作する前に、この登録が実行されます。

構文

```
register_timer watchdog|countdown|absolute|cron name ?
```

引数

name	（必須）操作されるタイマーの名前。

結果文字列

```
event_id %u
```

event_id は指定したタイマーのタイマーイベント ID です（これを使用して、**timer_arm** または **timer_cancel** コマンド拡張によってタイマーを操作するために使用されます）。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  4)    FH_EINITONCE  (Init() is not yet done, or done twice.)
```

このエラーは、EEM イベント ディテクタがその初期化を完了する前に、特定のイベントを登録する要求が行われたことを意味します。

```
(_cerr_sub_err =  6)    FH_EBADEVENTTYPE  (unknown EEM event type)
```

このエラーは、内部イベント指定で指定されたイベント タイプが無効であったことを意味します。

```
(_cerr_sub_err =  9)    FH_EMEMORY  (insufficient memory for request)
```

このエラーは、メモリの内部 EEM 要求に障害が発生したことを意味します。

```
(_cerr_sub_err = 10)    FH_ECORRUPT  (internal EEM API context is corrupt)
```

このエラーは、内部 EEM API コンテキスト構造が破損したことを意味します。

```
(_cerr_sub_err = 11)    FH_ENOSUCHESID  (unknown event specification ID)
```

このエラーは、イベントが登録されたときか、またはイベント ディテクタの内部イベント構造が破損したときに、イベント指定 ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 16)    FH_EBADFMPPTR  (bad ptr to fh_p data structure)
```

このエラーは、各 EEM API コールで使用されるコンテキスト ポインタが不正確であったことを意味します。

```
(_cerr_sub_err = 17)    FH_EBADADDRESS  (bad API control block address)
```

このエラーは、EEM API に渡された制御ブロック アドレスが不正確であったことを意味します。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部 EEM イベント ディテクタ ポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 25)    FH_ESUBSEXCEED  (number of subscribers exceeded)
```

このエラーは、タイマーまたはカウンタのサブスクライバの数が、最大値を超えたことを意味します。

```
(_cerr_sub_err = 26)    FH_ESUBSIDXINV  (invalid subscriber index)
```

これは、サブスクライバの索引が無効であったことを意味します。

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

このエラーは、イベント ディテクタが使用できなかったことを意味します。

```
(_cerr_sub_err = 56)    FH_EFDCONNERR  (event detector connection error)
```

このエラーは、この要求を処理する EEM イベント ディテクタは使用できないことを意味します。

timer_arm

タイマーを搭載します。タイプは、CRON、ウォッチドッグ、カウントダウン、または絶対の場合があります。

構文

```
timer_arm event_id ? cron_entry ?|time ?
```

引数

event_id	（必須）**register_timer** Tcl コマンド拡張によって返されるタイマー イベント ID。0 ～ 4294967295 の範囲の整数である必要があります。
cron_entry	（必須）タイマー タイプが CRON の場合に存在する必要があります。他のタイプのタイマーの場合には、存在させることはできません。CRON タイマー指定によって、CRON テーブル エントリの形式が使用されます。
time	（必須）タイマー タイプが CRON ではない場合に存在する必要があります。タイマー タイプが CRON の場合には、存在できません。ウォッチドッグ タイマーおよびカウントダウン タイマーでは、タイマーの期限が切れるまでの秒数およびミリ秒数です。絶対タイマーでは、期限切れ時刻のカレンダー時間です（SSSSSSSSSS[.MMM] 形式で指定します。SSSSSSSSSS は、0 ～ 4294967295 の秒数を表す整数で、MMM は 0 ～ 999 のミリ秒数を表す整数である必要があります）。期限の絶対日付は、1970 年 1 月 1 日以降の秒およびミリ秒の単位での数です。指定された日付がすでに過ぎた場合、タイマーの期限はただちに切れます。

結果文字列

```
sec_remain %ld msec_remain %ld
```

sec_remain および msec_remain は、タイマーの次の期限切れまでの残り時間です。

（注） タイマー タイプが CRON の場合、sec_remain 引数および msec_remain 引数には 0 が返されます。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  6)    FH_EBADEVENTTYPE  (unknown EEM event type)
```

このエラーは、内部イベント指定で指定されたイベント タイプが無効であったことを意味します。

```
(_cerr_sub_err =  9)    FH_EMEMORY  (insufficient memory for request)
```

このエラーは、メモリの内部 EEM 要求に障害が発生したことを意味します。

```
(_cerr_sub_err = 11)    FH_ENOSUCHESID  (unknown event specification ID)
```

このエラーは、イベントが登録されたときか、またはイベント ディテクタの内部イベント構造が破損したときに、イベント指定 ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 12)    FH_ENOSUCHEID  (unknown event ID)
```

このエラーは、イベントが登録されたときか、またはイベント ディテクタの内部イベント構造が破損したときに、イベント ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部 EEM イベント ディテクタ ポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 27)    FH_ETMDELAYZR  (zero delay time)
```

このエラーは、タイマーの搭載に指定された時間がゼロであったことを意味します。

```
(_cerr_sub_err = 42)    FH_ENOTREGISTERED  (request for event spec that is unregistered)
```

このエラーは、イベント検出が登録できなかったことを意味します。

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

このエラーは、イベント ディテクタが使用できなかったことを意味します。

```
(_cerr_sub_err = 56)    FH_EFDCONNERR  (event detector connection error)
```

このエラーは、この要求を処理する EEM イベント ディテクタは使用できないことを意味します。

timer_cancel

タイマーを取り消します。

構文

```
timer_cancel event_id ?
```

引数

event_id	（必須）**register_timer** Tcl コマンド拡張によって返されるタイマー イベント ID。0 ～ 4294967295 の範囲の整数である必要があります。

結果文字列

```
sec_remain %ld msec_remain %ld
```

sec_remain および msec_remain は、タイマーの次の期限切れまでの残り時間です。

（注） タイマー タイプが CRON の場合、sec_remain および msec_remain には 0 が返されます。

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  6)    FH_EBADEVENTTYPE  (unknown EEM event type)
```

このエラーは、内部イベント指定で指定されたイベント タイプが無効であったことを意味します。

```
(_cerr_sub_err =  7)    FH_ENOSUCHKEY  (could not find key)
```

このエラーは、アプリケーション イベント ディテクタ情報キーまたはその他の ID が見つからなかったことを意味します。

```
(_cerr_sub_err = 11)    FH_ENOSUCHESID  (unknown event specification ID)
```

このエラーは、イベントが登録されたときか、またはイベント ディテクタの内部イベント構造が破損したときに、イベント指定 ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 12)    FH_ENOSUCHEID  (unknown event ID)
```

このエラーは、イベントが登録されたときか、またはイベントディテクタの内部イベント構造が破損したときに、イベント ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部 EEM イベントディテクタ ポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

このエラーは、イベント ディテクタが使用できなかったことを意味します。

```
(_cerr_sub_err = 56)    FH_EFDCONNERR  (event detector connection error)
```

このエラーは、この要求を処理する EEM イベント ディテクタは使用できないことを意味します。

unregister_counter

カウンタの登録を解除します。この Tcl コマンド拡張は、以前に **register_counter** Tcl コマンド拡張に登録されていたカウンタの登録を解除するために、カウンタパブリッシャによって使用されます。

構文

```
unregister_counter event_id ? event_spec_id ?
```

引数

event_id	（必須）**register_counter** コマンド拡張によって返されるカウンタイベント ID。0 ～ 4294967295 の範囲の整数である必要があります。
event_spec_id	（必須）**register_counter** コマンド拡張によって返された、指定されたカウンタのカウンタイベント指定 ID。0 ～ 4294967295 の範囲の整数である必要があります。

結果文字列

なし

_cerrno を設定

可

```
(_cerr_sub_err =  2)    FH_ESYSERR  (generic/unknown error from OS/system)
```

このエラーは、オペレーティング システムによってレポートされたエラーを意味します。エラーとともにレポートされる POSIX errno 値を使用して、オペレーティング システム エラーの原因を調べます。

```
(_cerr_sub_err =  9)    FH_EMEMORY  (insufficient memory for request)
```

このエラーは、メモリの内部 EEM 要求に障害が発生したことを意味します。

```
(_cerr_sub_err = 11)    FH_ENOSUCHESID  (unknown event specification ID)
```

このエラーは、イベントが登録されたときか、またはイベントディテクタの内部イベント構造が破損したときに、イベント指定 ID を照会できなかったことを意味します。

```
(_cerr_sub_err = 22)    FH_ENULLPTR  (event detector internal error - ptr is null)
```

このエラーは、内部 EEM イベント ディテクタ ポインタに値が含まれている必要があったときに、ヌルであったことを意味します。

```
(_cerr_sub_err = 26)    FH_ESUBSIDXINV  (invalid subscriber index)
```

これは、サブスクライバの索引が無効であったことを意味します。

```
(_cerr_sub_err = 54)    FH_EFDUNAVAIL  (connection to event detector unavailable)
```

このエラーは、イベント ディテクタが使用できなかったことを意味します。

```
(_cerr_sub_err = 56)    FH_EFDCONNERR  (event detector connection error)
```

このエラーは、この要求を処理する EEM イベント ディテクタは使用できないことを意味します。

第 XII 部

VLAN

第 100 章

VTP の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

VTP の前提条件

VLAN を作成する前に、ネットワークで VLAN Trunking Protocol（VTP）を使用するかどうかを決定する必要があります。VTP を使用すると、1 つまたは複数のdevices上で中央集約的に設定変更を行い、その変更を自動的にネットワーク上の他のdevicesに伝達できます。VTP を使用しない場合、VLAN 情報を他のdevicesに送信することはできません。

VTP は、1 つのdeviceで行われた更新が VTP を介してドメイン内の他のdevicesに送信される環境で動作するように設計されています。VLANデータベースに対する複数の更新が同一ドメイ

ン内のdevices上で同時に発生する環境の場合、VTP は適切に機能せず、VLANデータベースの不整合が生じます。

deviceは合計 1000 の VLAN をサポートします。ただし、ルーテッドポート、SVI、およびその他の設定済み機能の個数によって、deviceハードウェアの使用状況は左右されます。VTP が新しい VLAN をdeviceに通知し、deviceが使用可能な最大限のハードウェア リソースをすでに使用している場合、コントローラはハードウェア リソース不足を伝えるメッセージを送信して、VLAN をシャットダウンします。**show vlan** EXEC コマンドの出力に、中断状態の VLAN が示されます。

トランク ポートはVTPアドバタイズを送受信するので、device上で少なくとも1つのトランク ポートが設定されており、そのトランク ポートが別のdeviceのトランク ポートに接続されていることを確認する必要があります。そうでない場合、deviceはVTPアドバタイズを受信できません。

VTP の制約事項

（注） VTP クライアント device を VTP ドメインに追加する前に、必ず VTP コンフィギュレーション リビジョン番号が VTP ドメイン内の他の devices のコンフィギュレーション リビジョン番号より小さいことを確認してください。VTP ドメインの Devices は、VTP 設定リビジョン番号が最も高い device の VLAN 設定をいつも使用します。VTP ドメイン内のリビジョン番号よりも大きなリビジョン番号を持つdeviceを追加すると、VTPサーバーおよび VTP ドメインからすべての VLAN 情報が消去される場合があります。

次に、VTP を設定する際の制約事項を示します。

- 1K VLAN は Lan Base のデフォルト テンプレートが設定された LAN Base イメージを実行しているスイッチ上でのみサポートされます。
- 標準範囲の VLAN 設定の CPU 使用率が高いことを示す警告メッセージを回避するには、使用する VLAN を 256 までにすることを推奨します。

  この場合、約10 のアクセス インターフェイス、または 5つのトランク インターフェイスが同時にフラップできます。これによる CPU 使用率への影響はごくわずかです（同時にフラップするインターフェイスが多い場合は、CPU使用率が非常に高くなる場合があります）。

VTP の概要

VTP

VTP は、レイヤ 2 のメッセージ プロトコルであり、ネットワーク全体にわたって VLAN の追加、削除、名前の変更を管理することにより、VLAN 設定の整合性を維持します。VTP によ

り、VLAN 名の重複、誤った VLAN タイプの指定、セキュリティ違反など、さまざまな問題を引き起こしかねない設定の誤りや矛盾が最小限に抑えられます。

VTP バージョン 1 およびバージョン 2 は、標準範囲の VLAN（VLAN ID 1 ～ 1005）だけをサポートします。VTP バージョン 3 は、VLAN 範囲全体（VLAN 1 ～ 4094）をサポートします。拡張範囲 VLAN（VLAN 1006 ～ 4094）は、VTP バージョン 3 でだけサポートされます。

拡張 VLAN がドメインに設定されている場合は、VTP バージョン 3 から VTP バージョン 2 に変換できません。

VTP ドメイン

VTP ドメイン（別名 VLAN 管理ドメイン）は、1 つの device、または複数の相互接続された devices で構成されます。deviceは、1 つの VTP ドメインにだけ所属できます。そのドメインに対してグローバル VLAN の設定を変更します。

デフォルトの設定では、トランク リンク（複数 VLAN のトラフィックを伝送するリンク）を介してドメインについてのアドバタイズを受信しない限り、またはユーザーがドメイン名を設定しない限り、deviceはVTP非管理ドメイン ステートです。管理ドメイン名を指定するか学習するまでは、VTP サーバー上で VLAN を作成または変更できません。また、VLAN 情報はネットワークを介して伝播されません。

deviceが、トランク リンクを介してVTPアドバタイズを受信した場合、管理ドメイン名およびVTP設定のリビジョン番号を継承します。その後deviceは、別のドメイン名または古いコンフィギュレーション リビジョン番号が指定されたアドバタイズについては、すべて無視します。

VTP サーバー上の VLAN 設定を変更すると、その変更は VTP ドメイン内のすべてのdevicesに伝播されます。VTP アドバタイズは、IEEE 802.1Q を含め、すべての IEEE トランク接続に送信されます。VTP は、複数の LAN タイプにわたり、固有の名前と内部インデックスの対応によって VLAN を動的にマッピングします。このマッピングにより、ネットワーク管理者がデバイスを管理するための作業負担が大幅に軽減されます。

VTP トランスペアレント モードでdeviceを設定した場合、VLAN の作成および変更は可能ですが、その変更はドメイン内の他のdevicesには送信されません。また、変更が作用するのは、個々のdeviceに限られます。ただし、deviceがこのモードのときに設定を変更すると、変更内容がdeviceの実行コンフィギュレーションに保存されます。この変更はdeviceのスタートアップ コンフィギュレーション ファイルに保存することもできます。

VTP モード

表 *205 : VTP* モード

VTP モード	説明
VTP サーバ	VTP サーバ モードでは、VLAN の作成、変更、削除ができます。また、VTP ドメイン全体に対して他のコンフィギュレーション パラメータ（VTP バージョンなど）を指定できます。VTP サーバーは、同一 VTP ドメイン内の他のdevicesに自身の VLAN 設定をアドバタイズし、トランク リンクを介して受信したアドバタイズに基づいて、自身の VLAN 設定を他のdevicesと同期させます。 VTP サーバがデフォルトのモードです。 VTP サーバ モードでは、VLAN 設定は NVRAM に保存されます。deviceがコンフィギュレーションを NVRAM に書き込んでいる間に障害を検出すると、VTPモードはサーバーモードからクライアント モードに自動的に移行します。この場合、NVRAM が正常に動作するまで、deviceを VTP サーバー モードに戻すことはできません。
VTP クライアント	VTP クライアントは VTP サーバと同様に機能し、そのトランクで VTP アップデートを送受信しますが、VTP クライアント上で VLAN の作成、変更、削除を行うことはできません。VLANは、ドメインに含まれる、他のサーバーモードのdeviceで設定します。 VTP バージョン 1 および 2 の VTP クライアント モードでは、VLAN 設定は NVRAM に保存されません。VTP バージョン 3 では、VLAN 設定はクライアント モードで NVRAM に保存されます。

VTP モード	説明
VTP トランスペアレント	VTP トランスペアレント devicesは、VTP に参加しません。VTP トランスペアレント device は自身の VLAN 設定をアドバタイズせず、受信したアドバタイズに基づいて自身の VLAN 設定を同期させることもありません。ただし、VTP バージョン 2 またはバージョン 3 では、トランスペアレント devicesは、トランク インターフェイスを介して他のdevicesから受信した VTP アドバタイズを転送します。VTP トランスペアレント モードでは、device上の VLAN を作成、変更、削除できます。 VTP バージョン 1 および 2 では、プライベート VLAN を作成するときに、deviceは VTP トランスペアレント モードにする必要があります。また、このプライベート VLAN の設定後はVTPモードをトランスペアレント モードからクライアント モードやサーバー モードに変更しないでください。VTPバージョン3では、クライアント モードとサーバー モードでもプライベート VLAN をサポートします。プライベート VLAN が設定されている場合、VTP モードをトランスペアレントからクライアント モードやサーバー モードに変更しないでください。 deviceが VTP トランスペアレント モードの場合、VTP および VLAN の設定は NVRAM に保存されますが、他のdevicesにはアドバタイズされません。このモードでは、VTP モードおよびドメイン名は device の実行コンフィギュレーションに保存されます。この情報を device の実行コンフィギュレーションに保存するには、**copy running-config startup-config** 特権 EXEC コマンドを使用します。
VTP オフ	VTP オフ モードでのdeviceの機能は、トランクを介して VTP アドバタイズを転送しないことを除くと VTP トランスペアレント deviceとしての機能と同じです。

VTP アドバタイズ

VTP ドメイン内の各deviceは、専用のマルチキャストアドレスに対して、それぞれのトランクポートからグローバル コンフィギュレーション アドバタイズを定期的に送信します。ネイバーdevicesは、このようなアドバタイズを受信し、必要に応じて各自の VTP および VLAN 設定をアップデートします。

トランク ポートは VTP アドバタイズを送受信するので、スイッチ スタック上で少なくとも 1 つのトランクポートが設定されており、そのトランク ポートが別のスイッチのトランク ポートに接続されていることを確認する必要があります。そうでない場合、スイッチは VTP アドバタイズを受信できません。

VTP アドバタイズにより、次のグローバル ドメイン情報が配信されます。

- VTP ドメイン名
- VTP 設定のリビジョン番号
- アップデート ID およびアップデート タイムスタンプ
- 各 VLAN の最大伝送単位（MTU）サイズを含む MD5 ダイジェスト VLAN コンフィギュレーション
- フレーム形式

VTP アドバタイズではさらに、設定されている各 VLAN について、次の VLAN 情報が配信されます。

- VLAN ID（IEEE 802.1Q を含む）
- VLAN 名
- VLAN タイプ
- VLAN ステート
- VLAN タイプ固有のその他の VLAN 設定情報

VTP バージョン 3 では、VTP アドバタイズにはプライマリ サーバ ID、インスタンス番号、および開始インデックスも含まれます。

VTP バージョン 2

ネットワークで VTP を使用する場合、VTP のどのバージョンを使用するかを決定する必要があります。デフォルトでは、バージョン 1 の VTP が動作します。

VTP バージョン 1 でサポートされず、バージョン 2 でサポートされる機能は、次のとおりです。

- トークンリング サポート：VTP バージョン 2 は、トークンリング ブリッジ リレー機能（TrBRF）およびトークンリング コンセントレータ リレー機能（TrCRF）VLANをサポートします。

- 認識不能な Type-Length-Value（TLV）のサポート：VTP サーバまたは VTP クライアントは、TLV が解析不能であっても、設定の変更を他のトランクに伝播します。認識されなかった TLV は、deviceが VTP サーバー モードで動作している場合、NVRAM に保存されます。
- バージョン依存型トランスペアレント モード：VTP バージョン 1 の場合、VTP トランスペアレント deviceがVTP メッセージ中のドメイン名およびバージョンを調べ、バージョンおよびドメイン名が一致する場合に限りメッセージを転送します。VTP バージョン 2 がサポートするドメインは 1 つだけですが、VTP バージョン 2 トランスペアレント deviceは、ドメイン名が一致した場合のみメッセージを転送します。
- 整合性検査：VTP バージョン 2 では、CLI または SNMP を介して新しい情報が入力された場合に限り、VLAN整合性検査（VLAN名、値など）を行います。VTP メッセージから新しい情報を取得した場合、またはNVRAMから情報を読み込んだ場合には、整合性検査を行いません。受信した VTP メッセージの MD5 ダイジェストが有効であれば、情報を受け入れます。

VTP バージョン 3

VTP バージョン 1 または 2 でサポートされず、バージョン 3 でサポートされる機能は、次のとおりです。

- 拡張認証：認証を **hidden** または **secret** として設定できます。設定を **hidden** にした場合、パスワード文字列からの秘密鍵は VLAN のデータベースファイルに保存されますが、設定においてプレーンテキストで表示されることはありません。代わりに、パスワードに関連付けられているキーが 16 進表記で実行コンフィギュレーションに保存されます。ドメインにテイクオーバー コマンドを入力する際は、パスワードを再入力する必要があります。**secret** キーワードを入力する場合、パスワードに秘密鍵を直接設定できます。
- 拡張範囲 VLAN （VLAN 1006 ～ 4094）データベース伝播のサポート：VTP バージョン 1 および 2 では VLAN 1 ～ 1005 だけが伝播されます。

（注） VTP プルーニングは引き続き VLAN 1 ～ 1005 にだけ適用され、VLAN 1002 ～ 1005 は予約されたままで変更できません。

- プライベート VLAN のサポート。
- ドメイン内のデータベースのサポート：VTP 情報の伝播に加え、バージョン 3 では、Multiple Spanning Tree（MST）プロトコル データベース情報も伝播できます。VTP プロトコルの個別インスタンスが VTP を使用する各アプリケーションで実行されます。
- VTP プライマリ サーバと VTP セカンダリ サーバ：VTP プライマリ サーバは、データベース情報を更新し、システム内のすべてのデバイスに適用されるアップデートを送信します。VTP セカンダリ サーバで実行できるのは、プライマリ サーバから NVRAM に受け取ったアップデート済み VTP コンフィギュレーションのバックアップだけです。

デフォルトでは、すべてのデバイスはセカンダリ サーバとして起動します。**vtp primary** 特権EXECコマンドを入力して、プライマリサーバを指定することができます。プライマリ サーバのステータスは、管理者がドメインでテイクオーバー メッセージを発行する場合、データベースのアップデート用に必要となるだけです。プライマリ サーバなしで実用VTP ドメインを持つことができます。プライマリ サーバーのステータスは、deviceにパスワードが設定されている場合でも、装置がリロードしたり、ドメインのパラメータが変更したりすると失われます。

VTP プルーニング

VTPプルーニングを使用すると、トラフィックが宛先デバイスに到達するために使用しなければならないトランク リンクへのフラッディング トラフィックが制限されるので、使用可能なネットワーク帯域幅が増えます。VTPプルーニングを使用しない場合、deviceは受信側のdevicesで廃棄される可能性があっても、VTP ドメイン内のすべてのトランク リンクに、ブロードキャスト、マルチキャスト、および不明のユニキャスト トラフィックをフラッディングします。VTP プルーニングはデフォルトでディセーブルです。

VTP プルーニングは、プルーニング適格リストに指定された VLAN トランク ポートへの不要なフラッディング トラフィックを阻止します。プルーニング適格リストに指定された VLAN だけが、プルーニングの対象になります。デフォルトでは、deviceのトランク ポート上でVLAN 2 ～ 1001 がプルーニング適格です。プルーニング不適格として設定した VLAN については、引き続きフラッディングが行われます。VTP プルーニングはすべてのバージョンの VTP でサポートされます。

VTP バージョン 1 および 2 では、VTP サーバーでプルーニングをイネーブルにすると、その VTP ドメイン全体でプルーニングがイネーブルになります。VTP バージョン 3 では、ドメイン内の各device上で手動によってプルーニングを有効にする必要があります。VLAN をプルーニング適格または不適格として設定する場合、影響を受けるのは、そのトランク上の VLAN のプルーニングだけです（VTP ドメイン内のすべてのdevicesに影響するわけではありません）。

VTP プルーニングは、イネーブルにしてから数秒後に有効になります。VTP プルーニング不適格のVLANからのトラフィックは、プルーニングの対象になりません。VLAN 1 およびVLAN 1002 ～ 1005 は常にプルーニング不適格です。これらの VLAN からのトラフィックはプルーニングできません。拡張範囲 VLAN（1005 を超える VLAN ID）もプルーニング不適格です。

VTP 設定時の注意事項

VTP の設定要件

VTP を設定する場合は、deviceがドメイン内の他のdevicesとVTPアドバタイズを送受信できるように、トランク ポートを設定する必要があります。

VTP バージョン 1 および 2 ではプライベート VLAN をサポートしません。VTP バージョン 3 ではプライベート VLAN をサポートします。プライベート VLAN を設定した場合、deviceは VTP トランスペアレント モードでなければなりません。プライベート VLAN がdeviceに設定

されている場合、VTPモードをトランスペアレントモードからクライアントモードやサーバーモードに変更しないでください。

VTP の設定

VTP 情報は VTP VLAN データベースに保存されます。VTP モードが透過的である場合、VTP ドメイン名およびモードは device 実行コンフィギュレーション ファイルに保存されます。この情報を device スタートアップ コンフィギュレーション ファイルに保存するには、**copy running-config startup-config** 特権 EXEC コマンドを入力します。deviceをリセットした場合にも、VTPモードをトランスペアレントとして保存するには、このコマンドを使用する必要があります。

deviceのスタートアップ コンフィギュレーション ファイルにVTP情報を保存して、deviceを再起動すると、deviceの設定は次のように選択されます。

- スタートアップ コンフィギュレーションおよびVLANデータベース内のVTPモードがトランスペアレントであり、VLANデータベースとスタートアップ コンフィギュレーション ファイルのVTPドメイン名が一致する場合は、VLANデータベースが無視され（クリアされ）、スタートアップ コンフィギュレーション ファイル内のVTPおよびVLAN設定が使用されます。VLANデータベース内のVLANデータベース リビジョン番号は変更されません。
- スタートアップ コンフィギュレーション内のVTPモードまたはドメイン名がVLANデータベースと一致しない場合、VLAN ID 1 ～ 1005 のドメイン名、VTPモード、およびVTP設定にはVLANデータベース情報が使用されます。

VTP 設定のためのドメイン名

VTP を初めて設定するときは、必ずドメイン名を割り当てる必要があります。また、VTP ドメイン内のすべてのdevicesを、同じドメイン名で設定する必要があります。VTPトランスペアレントモードのDevicesは、他のdevicesとVTPメッセージを交換しません。これらのコントローラについてはVTP ドメイン名を設定する必要はありません。

（注） NVRAM および DRAM の記憶域が十分にある場合は、VTP ドメイン内のすべてのdevicesを VTP サーバー モードにする必要があります。

注意 すべてのdevicesがVTPクライアントモードで動作している場合は、VTPドメインを設定しないでください。ドメインを設定すると、そのドメインのVLAN設定を変更できなくなります。VTP ドメイン内の少なくとも 1 台のdeviceを VTP サーバー モードに設定してください。

VTP ドメインのパスワード

VTP ドメインのパスワードは設定できますが、必須ではありません。ドメイン パスワードを設定する場合は、すべてのドメインdevicesで同じパスワードを共有し、管理ドメイン内のdeviceごとにパスワードを設定する必要があります。パスワードのないDevices、またはパスワードが不正なコントローラは、VTP アドバタイズを拒否します。

ドメインに VTP パスワードを設定する場合、VTP 設定なしで起動したdeviceは、正しいパスワードを使用して設定しない限り、VTP アドバタイズを受信しません。設定後、deviceは同じパスワードおよびドメイン名を使用した次の VTP アドバタイズを受信します。

VTP 機能を持つ既存のネットワークに新しいdeviceを追加した場合、その新しいdeviceに適切なパスワードを設定して初めて、そのコントローラはドメイン名を学習します。

⚠

注意 VTP ドメイン パスワードを設定したにもかかわらず、ドメイン内の各deviceに管理ドメイン パスワードを割り当てなかった場合には、管理ドメインが正常に動作しません。

VTP バージョン

実装する VTP バージョンを決定する場合は、次の注意事項に従ってください。

- VTP ドメイン内のすべてのdevicesは同じドメイン名を使用する必要がありますが、すべてが同じ VTP バージョンを実行する必要はありません。
- VTP バージョン 2 対応のdevice上で VTP バージョン 2 がディセーブルに設定されている場合、VTP バージョン 2 対応deviceは、VTP バージョン 1 を実行しているdeviceと同じ VTP ドメインで動作できます（デフォルトでは VTP バージョン 2 はディセーブルになっています）。
- VTP バージョン 1 を実行しているものの、VTP バージョン 2 に対応可能なdeviceが VTP バージョン 3 アドバタイズを受信すると、このコントローラは VTP バージョン 2 に自動的に移行します。
- VTP バージョン 3 を実行しているdeviceが VTP バージョン 1 を実行しているdeviceに接続すると、VTP バージョン 1 のdeviceは VTP バージョン 2 に移行し、VTP バージョン 3 のdeviceは、スケールダウンしたバージョンの VTP パケットを送信するため、VTP バージョン 2 deviceは自身のデータベースをアップデートできます。
- VTP バージョン 3 を実行するdeviceは、拡張 VLAN を持つ場合はバージョン 1 または 2 に移行できません。
- 同一 VTP ドメイン内のすべてのdeviceがバージョン 2 に対応可能な場合を除いて、devices 上で VTP バージョン 2 をイネーブルにしないでください。1 つのdeviceでバージョン 2 をイネーブルにすると、ドメイン内のすべてのバージョン 2 対応devicesでバージョン 2 がイネーブルになります。バージョン 1 専用のdeviceがドメインに含まれている場合、そのコントローラはバージョン 2 対応devicesとの間で VTP 情報を交換できません。

- VTP バージョン 1 および 2 devicesは、VTP バージョン 3 アドバタイズメントを転送できないため、ネットワークのエッジに配置することをお勧めします。
- 使用環境に TrBRF および TrCRF トークンリング ネットワークが含まれている場合に、トークンリング VLAN スイッチング機能を正しく動作させるには、VTP バージョン 2 またはバージョン 3 をイネーブルにする必要があります。トークンリングおよびトークンリング Net を実行する場合は、VTP バージョン 2 をディセーブルにします。
- VTP バージョン 1 およびバージョン 2 は、拡張範囲 VLAN（VLAN 1006 ～ 4094）の設定情報を伝播しません。これらの VLAN は各装置で手動によって設定する必要があります。VTP バージョン 3 は拡張範囲 VLAN と、拡張範囲 VLAN データベースの伝播をサポートします。
- VTP バージョン 3 装置のトランク ポートが VTP バージョン 2 装置からのメッセージを受信した場合、この装置は、VLAN データベースをスケールダウンし、その特定のトランク上で VTP バージョン 2 フォーマットを使用して送信します。VTP バージョン 3 装置は、最初にそのトランク ポートで VTP バージョン 2 パケットを受信しない限り、VTP バージョン 2 フォーマットのパケットを送信しません。
- VTP バージョン 3 装置が、あるトランク ポートで VTP バージョン 2 装置を検出した場合、両方のネイバーが同一トランク上で共存できるように、VTP バージョン 2 パケットだけでなく VTP バージョン 3 パケットの送信も継続します。
- VTP バージョン 3 装置は、VTP バージョン 2 またはバージョン 1 の装置からの設定情報は受け入れません。
- 2 つの VTP バージョン 3 リージョンは、VTP バージョン 1 リージョンまたはバージョン 2 リージョンでは、トランスペアレント モードでだけ通信できます。
- VTP バージョン 1 にだけ対応する装置は、VTP バージョン 3 装置との相互運用はできません。
- VTP バージョン 1 およびバージョン 2 は、拡張範囲 VLAN（VLAN 1006 ～ 4094）の設定情報を伝播しません。これらの VLAN を各装置上に手動で設定する必要があります。

VTP のデフォルト設定

次の表に、VTP のデフォルト設定を記載します。

表 *206 : VTP* のデフォルト設定

機能	デフォルト設定
VTP ドメイン名	ヌル
VTP モード（VTP バージョン 1 およびバージョン 2）	サーバ

機能	デフォルト設定
VTP モード（VTP バージョン 3）	このモードは、VTP バージョン 3 に変換する前のバージョン 1 または 2 のモードと同じです。
VTP バージョン	バージョン 1
MST データベース モード	トランスペアレント
VTP バージョン 3 のサーバ タイプ	セカンダリ
VTP パスワード	なし
VTP プルーニング	ディセーブル

VTP の設定方法

VTP モードの設定

次のいずれかに VTP モードを設定できます。

- VTP サーバー モード：VTP サーバー モードでは、VLAN の設定を変更し、ネットワーク全体に伝播させることができます。
- VTP クライアント モード：VTP クライアント モードでは、VLAN の設定を変更できません。クライアント deviceは、VTP ドメイン内の VTP サーバーから VTP アップデート情報を受信し、それに基づいて設定を変更します。
- VTP トランスペアレント モード：VTP トランスペアレント モードでは、deviceで VTP がディセーブルになります。deviceは VTP アップデートを送信せず、他のdeviceから受信した VTP アップデートにも反応しません。ただし、VTP バージョン 2 を実行する VTP トランスペアレント モードのdeviceは、対応するトランク リンクで、受信した VTP アドバタイズを転送します。
- VTP オフ モード：VTP オフ モードは、VTP アドバタイズが転送されない以外は、VTP トランスペアレント モードと同じです。

設定したドメイン名は、削除できません。別のドメインにdeviceを再び割り当てるしかありません。

手順の概要

1. **enable**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**}

5. **vtp password** *password*
6. **end**
7. **show vtp status**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vtp domain** *domain-name* 例： `スイッチ(config)# vtp domain eng_group`	VTP管理ドメイン名を設定します。1 ～ 32 文字の名前を使用できます。同一管理下にあるVTPサーバーモードまたはクライアントモードのdevicesは、すべて同じドメイン名に設定する必要があります。 サーバ モード以外にはこのコマンドは任意です。VTPサーバモードではドメイン名が必要です。deviceが VTP ドメインにトランク接続されている場合、deviceはドメイン内の VTP サーバーからドメイン名を取得します。 他の VTP パラメータを設定する前に、VTP ドメインを設定する必要があります。
ステップ 4	**vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**} 例： `スイッチ(config)# vtp mode server`	VTPモード（クライアント、サーバー、トランスペアレント、またはオフ）のdeviceの設定。 • **vlan**：何も設定されていない場合はVLANデータベースがデフォルトです。 • **mst**：マルチ スパニング ツリー（MST）データベース。 • **unknown**：データベース タイプは不明です。
ステップ 5	**vtp password** *password* 例：	（任意）VTP ドメイン用のパスワードを設定します。パスワードに使用できる文字数は 8 ～ 64 文字です。VTPパスワードを設定したにもかかわらず、

	コマンドまたはアクション	目的
	スイッチ(config)# **vtp password mypassword**	ドメイン内の各deviceに同じパスワードを割り当てなかった場合には、VTP ドメインが正常に動作しません。
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show vtp status** 例： スイッチ# **show vtp status**	表示された *[VTP Operating Mode]* および *[VTP Domain Name]* フィールドの設定を確認します。
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）スタートアップ コンフィギュレーション ファイルに設定を保存します。 deviceの実行コンフィギュレーションに保存され、スタートアップ コンフィギュレーション ファイルにコピーできるのは、VTP モードおよびドメイン名だけです。

VTP バージョン 3 のパスワードの設定

deviceで VTP バージョン 3 のパスワードを設定できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **vtp version 3**
4. **vtp password** *password* [**hidden** | **secret**]
5. **end**
6. **show vtp password**
7. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ> **enable**	
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vtp version 3** 例： スイッチ(config)# **vtp version 3**	デバイスで VTP バージョン 3 を有効にします。デフォルトは VTP バージョン 1 です。
ステップ 4	**vtp password** *password* [**hidden** | **secret**] 例： スイッチ(config)# **vtp password mypassword hidden**	（任意）VTP ドメイン用のパスワードを設定します。パスワードに使用できる文字数は 8 ～ 64 文字です。 • （任意）**hidden**：パスワード文字列から生成される秘密キーが、nvram:vlan.dat ファイルに保存されます。VTP プライマリ サーバを設定してテイクオーバーを設定しようとすると、パスワードの再入力を要求されます。 • （任意）**secret**：パスワードを直接設定します。シークレット パスワードには 16 進数文字を 32 個含める必要があります。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show vtp password** 例： スイッチ# **show vtp password**	入力を確認します。次のような出力が表示されます。 VTP password: 89914640C8D90868B6A0D8103847A733
ステップ 7	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

VTP バージョン 3 のプライマリ サーバーの設定

VTP サーバを VTP プライマリ サーバとして設定すると、テイクオーバー操作が開始されます。

手順の概要

1. **vtp version 3**
2. **vtp primary** [**vlan** | **mst**] [**force**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**vtp version 3** 例： `スイッチ(config)# vtp version 3`	デバイスで VTP バージョン 3 を有効にします。デフォルトは VTP バージョン 1 です。
ステップ 2	**vtp primary** [**vlan**	**mst**] [**force**] 例： `スイッチ# vtp primary vlan force`	deviceの動作ステートをセカンダリ サーバー（デフォルト）からプライマリ サーバーに変更し、その設定をドメインにアドバタイズします。deviceのパスワードが **hidden** に設定されている場合は、パスワードの再入力を要求されます。 • （任意）**vlan**：テイクオーバー機能としてVLANデータベースを選択します。これはデフォルトです。 • （任意）**mst**：テイクオーバー機能としてマルチスパニングツリー（MST）データベースを選択します • （任意）**force**：競合するサーバの設定が上書きされます。**force** を入力しない場合、テイクオーバーの実行前に確認を求められます。

VTP バージョンのイネーブル化

デフォルトで VTP バージョン 2 およびバージョン 3 はディセーブルになっています。

- 1 つのdevice上で VTP バージョン 2 をイネーブルにすると、VTP ドメイン内の VTP バージョン 2 に対応可能なすべてのdeviceでバージョン 2 がイネーブルになります。VTP バージョン 3 をイネーブルにするには、各device上で手動によって設定する必要があります。
- VTP バージョン 1 および 2 では、このバージョンを設定できるのは、VTP サーバー モードまたはトランスペアレント モードのdevicesだけです。deviceが VTP バージョン 3 を実行

し、かつdeviceがクライアント モードの場合、既存の拡張 VLAN や既存のプライベート VLAN がなく、パスワードが非表示に設定されていないときであれば、バージョン 2 に変更できます。

⚠

注意 同一 VTP ドメイン内のdevices上で、VTP バージョン 1 と VTP バージョン 2 は相互運用できません。VTP ドメイン内のすべてのdeviceが VTP バージョン 2 をサポートしている場合を除き、VTP バージョン 2 をイネーブルにはしないでください。

- TrCRF および TrBRF トークンリング環境では、トークンリング VLAN スイッチング機能を正しく動作させるために、VTP バージョン 2 または VTP バージョン 3 をイネーブルにする必要があります。トークンリングおよびトークンリング Net メディアの場合は、VTP バージョン 2 をディセーブルにします。

⚠

注意 VTP バージョン 3 では、プライマリ サーバとセカンダリ サーバの両方がドメイン内の 1 つのインスタンスに存在できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **vtp version** {**1** | **2** | **3**}
4. **end**
5. **show vtp status**
6. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**vtp version** {**1** | **2** | **3**} 例： `スイッチ(config)# vtp version 2`	deviceで VTP バージョンをイネーブルにします。デフォルトは VTP バージョン 1 です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show vtp status** 例： `スイッチ# show vtp status`	設定された VTP バージョンがイネーブルであることを確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VTP プルーニングのイネーブル化

始める前に

VTP プルーニングは VTP トランスペアレント モードでは機能しないように設計されています。ネットワーク内に VTP トランスペアレント モードのdevicesが 1 台または複数存在する場合は、次のいずれかの操作を実行する必要があります。

- ネットワーク全体の VTP プルーニングをオフにします。
- VTP トランスペアレント deviceのアップストリーム側にあるdeviceのトランク上で、すべての VLAN をプルーニング不適格にすることによって、VTP プルーニングをオフにします。

インターフェイスに VTP プルーニングを設定するには、**switchport trunk pruning vlan** インターフェイス コンフィギュレーション コマンドを使用します。VTP プルーニングは、インターフェイスがトランキングを実行している場合に作用します。VLAN プルーニングの適格性は、VTP ドメインで VTP プルーニングがイネーブルであるかどうか、特定の VLAN が存在するかどうか、およびインターフェイスが現在トランキングを実行しているかどうかにかかわらず、設定できます。

手順の概要

1. **enable**
2. **configure terminal**
3. **vtp pruning**
4. **end**
5. **show vtp status**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vtp pruning** 例： `スイッチ(config)# vtp pruning`	VTP管理ドメインでプルーニングをイネーブルにします。 プルーニングは、デフォルトではディセーブルに設定されています。VTPサーバーモードの1台のdevice上に限ってプルーニングをイネーブルにする必要があります。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show vtp status** 例： `スイッチ# show vtp status`	表示された [VTP Pruning Mode] フィールドの設定を確認します。

ポート単位の VTP の設定

VTP バージョン 3 では、ポート単位で VTP をイネーブルまたはディセーブルにできます。VTP は、トランク モードのポート上でだけイネーブルにできます。VTP トラフィックの着信または発信はブロックされ、転送されません。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **vtp**
5. **end**
6. **show running-config interface** *interface-id*
7. **show vtp status**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet0/1`	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**vtp** 例： `スイッチ(config-if)# vtp`	指定したポートの VTP をイネーブルにします。
ステップ 5	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	
ステップ 6	**show running-config interface** *interface-id* 例： スイッチ# **show running-config interface gigabitethernet 1/0/1**	ポートの変更を確認します。
ステップ 7	**show vtp status** 例： スイッチ# **show vtp status**	設定を確認します。

VTP ドメインへの VTP クライアント の追加

VTP ドメインに追加する前にdevice上で VTP コンフィギュレーション リビジョン番号を確認およびリセットするには、次の手順に従います。

始める前に

VTP クライアントを VTP ドメインに追加する前に、必ず VTP コンフィギュレーション リビジョン番号が VTP ドメイン内の他のdevicesのコンフィギュレーション リビジョン番号より小さいことを確認してください。VTP ドメイン内のDevicesは常に、VTP コンフィギュレーション リビジョン番号が最大のdeviceの VLAN コンフィギュレーションを使用します。VTP バージョン 1 および 2 では、VTP ドメイン内のリビジョン番号よりも大きなリビジョン番号を持つdeviceを追加すると、VTP サーバーおよび VTP ドメインからすべての VLAN 情報が消去される場合があります。VTP バージョン 3 では、VLAN 情報が消去されることはありません。

device 上で VTP をディセーブルにし、VTP ドメイン内の他の devices に影響を与えることなく VLAN 情報を変更するには、**vtp mode transparent** グローバル コンフィギュレーション コマンドを使用します。

手順の概要

1. **enable**
2. **show vtp status**
3. **configure terminal**
4. **vtp domain** *domain-name*
5. **end**
6. **show vtp status**
7. **configure terminal**
8. **vtp domain** *domain-name*

9. **end**
10. **show vtp status**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**show vtp status** 例： スイッチ# **show vtp status**	VTP コンフィギュレーション リビジョン番号をチェックします。 番号が 0 の場合は、deviceを VTP ドメインに追加します。 番号が 0 より大きい場合は、次の手順に従います。 • ドメイン名を書き留めます。 • コンフィギュレーション リビジョン番号を書き留めます。 • 次のステップに進んで、deviceのコンフィギュレーションリビジョン番号をリセットします。
ステップ 3	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 4	**vtp domain** *domain-name* 例： スイッチ(config)# **vtp domain domain123**	ドメイン名を、ステップ 1 で表示された元の名前から新しい名前に変更します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。deviceの VLAN 情報が更新され、コンフィギュレーション リビジョン番号が 0 にリセットされます。
ステップ 6	**show vtp status** 例：	コンフィギュレーション リビジョン番号が 0 にリセットされていることを確認します。

	コマンドまたはアクション	目的
	スイッチ# **show vtp status**	
ステップ 7	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 8	**vtp domain** *domain-name* 例： スイッチ(config)# **vtp domain domain012**	deviceの元のドメイン名を開始します。
ステップ 9	**end** 例： スイッチ(config)# **end**	特権EXECモードに戻ります。deviceのVLAN情報が更新されます。
ステップ 10	**show vtp status** 例： スイッチ# **show vtp status**	（任意）ドメイン名がステップ1のものと同じであり、コンフィギュレーション リビジョン番号が 0 であることを確認します。

VTP のモニタ

ここでは、VTPの設定を表示およびモニタリングするために使用するコマンドについて説明します。

VTPの設定情報（ドメイン名、現在のVTPバージョン、VLAN数）を表示することによって、VTP をモニタします。deviceで送受信されたアドバタイズに関する統計情報を表示することもできます。

表 *207 : VTP* モニタ コマンド

コマンド	目的
show vtp counters	送受信された VTP メッセージに関するカウンタを表示します。

コマンド	目的
show vtp devices [**conflict**]	ドメイン内のすべての VTP バージョン 3 デバイスに関する情報を表示します。プライマリ サーバと競合する VTP バージョン 3 の装置が表示されます。**show vtp devices** コマンドは、deviceがトランスペアレントモードまたはオフモードのときは情報を表示しません。
show vtp interface [*interface-id*]	すべてのインターフェイスまたは指定されたインターフェイスに対する VTP のステータスおよび設定を表示します。
show vtp password	VTP パスワードを表示します。表示されるパスワードの形式は、**hidden** キーワードが入力されているか、または、暗号化がdeviceでイネーブル化されているかどうかによって異なります。
show vtp status	VTP device設定情報を表示します。

VTP の設定例

例：スイッチをプライマリ サーバとして設定する

次に、パスワードが非表示またはシークレットに設定されている場合に、VLANデータベースのプライマリ サーバー（デフォルト）としてdeviceを設定する方法の例を示します。

```
スイッチ# vtp primary vlan
Enter VTP password: mypassword
This switch is becoming Primary server for vlan feature in the VTP  domain

VTP Database Conf Switch ID      Primary Server Revision System Name
------------ ---- --------------- -------------- -------- --------------------
VLANDB       Yes  00d0.00b8.1400=00d0.00b8.1400 1        stp7

Do you want to continue (y/n) [n]? y
```

例：VTP サーバとしてのスイッチの設定

次に、ドメイン名が *eng_group*、パスワードが *mypassword* という VTP サーバとしてスイッチを設定する例を示します。

```
Switch(config)# vtp domain eng_group
Setting VTP domain name to eng_group.
```

```
Switch(config)# vtp mode server
Setting device to VTP Server mode for VLANS.

Switch(config)# vtp password mypassword
Setting device VLAN database password to mypassword.
Switch(config)# end
```

例：インターフェイスでの VTP のイネーブル化

インターフェイス上でVTPをイネーブルにするには、**vtp**インターフェイス コンフィギュレーション コマンドを使用します。インターフェイス上で VTP をディセーブルにするには、 **no vtp** インターフェイス コンフィギュレーション コマンドを使用します。

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# vtp
Switch(config-if)# end
```

例：VTP パスワードの作成

次に、VTP パスワードを作成する例を示します。

```
Switch(config)# vtp password mypassword hidden
Generating the secret associated to the password.
Switch(config)# end
Switch# show vtp password
VTP password: 89914640C8D90868B6A0D8103847A733
```

次の作業

VTP を設定したら、次の項目を設定できます。

- VLANS
- VLAN トランキング
- VLAN メンバーシップ ポリシー サーバー（VMPS）
- 音声 VLAN

次の作業

第 101 章

VLAN の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

VLAN の前提条件

VLAN 設定時の前提条件と考慮事項を次に示します。

- VLANを作成する前に、VLANトランキングプロトコル（VTP）を使用してネットワークのグローバルな VLAN 設定を維持するかどうかを決定する必要があります。
- スイッチは、VTPクライアント、サーバー、およびトランスペアレントの各モードで1000の VLAN をサポートしています。

VLAN の制約事項

次に、VLAN を設定する際の制約事項を示します。

- 標準範囲の VLAN 設定の CPU 使用率が高いことを示す警告メッセージを回避するには、使用する VLAN を 256 までにすることを推奨します。この場合、約10 のアクセス インターフェイス、または5つのトランク インターフェイスが同時にフラップできます。これによる CPU 使用率への影響はごくわずかです（同時にフラップするインターフェイスが多い場合は、CPU 使用率が非常に高くなる場合があります）。

VLAN について

論理ネットワーク

VLANは、ユーザの物理的な位置に関係なく、機能、プロジェクトチーム、またはアプリケーションなどで論理的に分割されたスイッチドネットワークです。VLAN は、物理 LAN と同じ属性をすべて備えていますが、同じ LAN セグメントに物理的に配置されていないエンド ステーションもグループ化できます。どのようなdeviceポートでもVLANに属することができ、ユニキャスト、ブロードキャスト、マルチキャストのパケットは、その VLAN 内のエンド ステーションだけに転送またはフラッディングされます。各 VLAN は 1 つの論理ネットワークと見なされ、VLAN に属さないステーション宛のパケットは、ルータまたはフォールバック ブリッジングをサポートするdeviceを経由して伝送しなければなりません。VLAN はそれぞれが独立した論理ネットワークと見なされるので、VLANごとに独自のブリッジ管理情報ベース（MIB）情報があり、スパニングツリーの独自の実装をサポートできます。

VLAN は通常、IP サブネットワークに対応付けられます。たとえば、特定の IP サブネットに含まれるエンドステーションはすべて同じVLANに属します。device上のインターフェイスのVLAN メンバーシップは、インターフェイスごとに手動で割り当てます。この方法でdeviceインターフェイスを VLAN に割り当てた場合、これをインターフェイス ベース（またはスタティック）VLAN メンバーシップと呼びます。

VLAN 間のトラフィックは、ルーティングする必要があります。

deviceは、device仮想インターフェイス（SVI）を使用して、VLAN 間でトラフィックをルーティングできます。VLAN 間でトラフィックをルーティングするには、SVI を明示的に設定して IP アドレスを割り当てる必要があります。

サポートされる VLAN

スイッチは、VTP クライアント、サーバ、およびトランスペアレントの各モードで VLAN をサポートしています。VLAN は、1 ～ 4094 の番号で識別します。VLAN ID 1002 ～ 1005 は、トークンリングおよびファイバ分散データ インターフェイス（FDDI）VLAN 専用です。

VTP バージョン 1 およびバージョン 2 は、標準範囲の VLAN（VLAN ID 1 ～ 1005）だけをサポートします。これらのバージョンで 1006 ～ 4094 の VLAN ID を作成する場合は、スイッチを VTP トランスペアレント モードにする必要があります。Cisco IOS Release 12.2(52)SE 以降では VTP バージョン 3 をサポートします。VTP バージョン 3 は、VLAN 範囲全体（VLAN 1 ～ 4094）をサポートします。拡張範囲 VLAN（VLAN 1006 ～ 4094）は、VTP バージョン 3 でのみサポートされます。拡張 VLAN がドメインに設定されている場合は、VTP バージョン 3 から VTP バージョン 2 に変換できません。

スイッチは、最大 128 のスパニングツリー インスタンスを持つ Per-VLAN Spanning-Tree Plus（PVST+）または Rapid PVST+ をサポートします。VLAN ごとに 1 つずつスパニングツリー インスタンスを使用できます。スイッチは、イーサネット ポート経由の VLAN トラフィックの送信方式として IEEE 802.1Q トランキングのみをサポートします。

VLAN ポート メンバーシップ モード

VLANに所属するポートは、メンバーシップ モードを割り当てることで設定します。メンバーシップ モードは、各ポートが伝送できるトラフィックの種類、および所属できる VLAN の数を指定します。

ポートが VLANに所属すると、deviceはVLAN単位で、ポートに対応するアドレスを学習して管理します。

表 *208 :* ポートのメンバーシップ モードとその特性

メンバーシップ モード	VLAN メンバーシップの特性	VTP の特性
スタティック アクセス	スタティック アクセス ポートは、手動で割り当てられ、1 つの VLAN だけに所属します。	VTP は必須ではありません。VTP にグローバルに情報を伝播させないようにする場合は、VTP モードをトランスペアレント モードに設定します。VTP に加入するには、別のdeviceのトランク ポートに接続されているdevice少なくとも 1 つのトランク ポートが必要です。
トランク（IEEE 802.1Q） • IEEE 802.1Q：業界標準のトランキング カプセル化方式です。	デフォルトで、トランク ポートは拡張範囲 VLAN を含むすべての VLAN のメンバです。ただし、メンバーシップは許可 VLAN リストを設定して制限できます。また、プルーニング適格リストを変更して、リストに指定したトランク ポート上の VLAN へのフラッディング トラフィックを阻止することもできます。	VTP を推奨しますが、必須ではありません。VTP は、ネットワーク全体にわたって VLAN の追加、削除、名前変更を管理することにより、VLAN 設定の整合性を維持します。VTP はトランク リンクを通じて他のdevicesと VLAN コンフィギュレーション メッセージを交換します。

メンバーシップ モード	VLAN メンバーシップの特性	VTP の特性
ダイナミック アクセス	ダイナミックアクセス ポートは 1 つの VLAN（VLAN ID が 1 ～ 4094）にのみ所属し、VLAN Member Policy Server（VMPS）によって動的に割り当てられます。 VMPS には Catalyst 6500 シリーズのスイッチを使用できますが、Catalyst スイッチなどは使用できません。Catalyst スイッチは VMPS クライアントです。 同一のdevice上でダイナミックアクセス ポートとトランク ポートを使用できますが、ダイナミックアクセス ポートは別のdeviceではなく、エンド ステーションまたはハブに接続する必要があります。	VTP は必須です。 VMPS およびクライアントを同じ VTP ドメイン名で設定してください。 VTP に加入するには、別のdeviceのトランク ポートに、device上の少なくとも 1 つのトランク ポートが接続されている必要があります。
音声 VLAN	音声 VLAN ポートは、Cisco IP Phone に接続し、電話に接続されたデバイスからの音声トラフィックに 1 つの VLAN を、データ トラフィックに別の VLAN を使用するように設定されたアクセス ポートです。	VTP は不要です。VTP は音声 VLAN に対して無効です。

VLAN コンフィギュレーション ファイル

VLAN ID 1 ～ 1005 の設定は vlan.dat ファイル（VLAN データベース）に書き込まれます。この設定を表示するには、**show vlan** 特権 EXEC コマンドを入力します。*vlan.dat* ファイルはフラッシュメモリに格納されます。VTP モードがトランスペアレント モードの場合、それらの設定もdeviceの実行コンフィギュレーション ファイルに保存されます。

さらに、インターフェイス コンフィギュレーション モードを使用して、ポートのメンバーシップ モードの定義、VLAN に対するポートの追加および削除を行います。これらのコマンドの実行結果は、実行コンフィギュレーション ファイルに書き込まれます。このファイルを表示するには、**show running-config** 特権 EXEC コマンドを入力します。

VLAN および VTP 情報（拡張範囲 VLAN 設定情報を含む）をスタートアップ コンフィギュレーション ファイルに保存して、deviceを再起動すると、deviceの設定は次のように選択されます。

- スタートアップ コンフィギュレーションおよび VLAN データベース内の VTP モードがトランスペアレントで、VLANデータベースとスタートアップ コンフィギュレーション ファイルの VTP ドメイン名が一致する場合は、VLAN データベースが無視され（クリアされ）、スタートアップ コンフィギュレーション ファイル内の VTP および VLAN 設定が使

用されます。VLAN データベース内の VLAN データベース リビジョン番号は変更されません。

- スタートアップ コンフィギュレーション内の VTP モードまたはドメイン名が VLAN データベースと一致しない場合、VLAN ID 1 ～ 1005 のドメイン名、VTP モード、および VTP 設定には VLAN データベース情報が使用されます。
- VTP バージョン 1 および 2 では、VTP モードがサーバである場合、VLAN ID 1 ～ 1005 のドメイン名と VLAN 設定で VLAN データベース情報が使用されます。VTP バージョン 3 は、VLAN 1006 ～ 4094 もサポートします。
- イメージ 15.0(02)SE6 から、vtp トランスペアレントおよびオフ モードでは、VLAN はインターフェイスに適用されない場合でも、startup-config から作成されます。

（注） スイッチの設定をリセットする前に、**write erase** コマンドを使用して、必ずコンフィギュレーション ファイルと一緒に vlan.dat ファイルを削除してください。これにより、リセット時にスイッチが正しく再起動します。

標準範囲 VLAN 設定時の注意事項

標準範囲 VLAN は、ID が 1 ～ 1005 の VLAN です。

VTP 1 および 2 は、標準範囲 VLAN だけをサポートします。

ネットワーク内で標準範囲 VLAN を作成または変更する場合には、次の注意事項に従ってください。

- 標準範囲 VLAN は、1 ～ 1001 の番号で識別します。VLAN 番号 1002 ～ 1005 は、トークンリングおよび FDDI VLAN 専用です。
- VLAN 1 ～ 1005 の VLAN 設定は、常に VLAN データベースに格納されます。VTP モードがトランスペアレント モードの場合、VTP と VLAN の設定もdeviceの実行コンフィギュレーション ファイルに保存されます。
- deviceが VTP サーバー モードまたは VTP トランスペアレント モードの場合は、VLAN データベース内の VLAN 2 ～ 1001 の設定を追加、変更、または削除できます（VLAN ID 1 および 1002 ～ 1005 は自動作成され、削除できません）。
- VTP バージョン 1 および 2 では、deviceは VTP トランスペアレント モード（VTP は無効）でのみ、VLAN ID 1006 ～ 4094 をサポートします。これらは拡張範囲 VLAN であり、設定オプションには制限があります。VTP トランスペアレント モードで作成された拡張範囲 VLAN は、VLAN データベースに保存されず、伝播されません。VTP バージョン 3 では、VTP サーバー モードでの拡張範囲 VLAN（VLAN 1006～4094）データベース伝播をサポートします。拡張 VLAN を設定している場合は、VTP バージョン 3 からバージョン 1 または 2 に変換できません。

- VLAN を作成する前に、deviceを VTP サーバー モードまたは VTP トランスペアレント モードにする必要があります。deviceが VTP サーバーである場合には、VTP ドメインを定義する必要があります。VTP ドメインを定義しないと、VTP は機能しません。
- deviceは、トークンリングまたは FDDI メディアをサポートしません。deviceは FDDI、FDDI-Net、TrCRF、または TrBRF トラフィックを転送しませんが、VTP を介して VLAN 設定を伝播します。
- deviceでは、一定数のスパニングツリーインスタンスがサポートされています（最新情報についてはデータシートを参照してください）。deviceのアクティブな VLAN 数が、サポートされているスパニングツリーインスタンス数より多い場合でも、スパニングツリーはサポートされている数の VLAN でのみ有効になり、残りの VLAN ではスパニングツリーは無効になります。

  device上の使用可能なスパニングツリーインスタンスをすべて使い切ってしまった後に、VTP ドメインの中にさらに別の VLAN を追加すると、そのdevice上にスパニングツリーが稼働しない VLAN が生成されます。そのdeviceのトランク ポート上でデフォルトの許可リスト（すべての VLAN を許可するリスト）が設定されていると、すべてのトランク ポート上に新しい VLAN が割り当てられます。ネットワーク トポロジによっては、新しい VLAN 上で、切断されないループが生成されることがあります。特に、複数の隣接devicesでスパニングツリー インスタンスをすべて使用してしまっている場合には注意が必要です。スパニングツリー インスタンスの割り当てを使い果たしたdevicesのトランク ポートに許可リストを設定することにより、このような可能性を防ぐことができます。

  device上の VLAN の数がサポートされているスパニングツリーインスタンスの最大数を超える場合、device上に IEEE 802.1s Multiple STP（MSTP）を設定して、複数の VLAN を単一のスパニングツリー インスタンスにマッピングすることを推奨します。

拡張範囲 VLAN 設定時の注意事項

拡張範囲 VLAN は、ID が 1006 ～ 4094 の VLAN です。

VTP 3 は拡張範囲 VLAN のみをサポートしています。

拡張範囲 VLAN を作成するときは次の注意事項に従ってください。

- 拡張範囲の VLAN ID は、deviceが VTP バージョン 3 を実行していない場合は VLAN データベースに保存されず、VTP で認識されません。
- プルーニング適格範囲に拡張範囲 VLAN を含めることはできません。
- VTP バージョン 1 または 2 では、グローバル コンフィギュレーション モードで、VTP モードをトランスペアレントに設定できます。VTP トランスペアレント モードでdeviceが始動するように、この設定をスタートアップ コンフィギュレーションに保存する必要があります。このようにしないと、deviceをリセットした場合に、拡張範囲 VLAN 設定が失われます。VTP バージョン 3 で拡張範囲 VLAN を作成する場合は、VTP バージョン 1 または 2 に変更できません。

VLAN のデフォルト設定

イーサネット VLAN のデフォルト設定

次の表に、イーサネット VLAN のデフォルト設定を記載します。

（注） スイッチがサポートするのは、イーサネット インターフェイスだけです。FDDI およびトークンリング VLAN は、ローカルではサポートされないので、FDDI およびトークンリング メディア固有の特性は、他のスイッチに対する VTP グローバル アドバタイズにのみ設定します。

表 *209* : イーサネット *VLAN* のデフォルトおよび範囲

パラメータ	デフォルト	範囲
VLAN ID	1	1 ～ 4094 （注） 拡張範囲 VLAN（VLAN ID 1006 ～ 4094）は、VTP バージョン 3 の VLAN データベースにのみ保存されます。
VLAN 名	VLANxxxx。xxxx は VLAN ID 番号に等しい 4 桁の数字（先行ゼロを含む）です。	範囲なし
IEEE 802.10 SAID	100001（100000 と VLAN ID の和）	1 ～ 4294967294
IEEE 802.10 SAID	1500	576 ～ 18190
プライベート VLAN	設定なし	2 ～ 1001、1006 ～ 4094

VLAN のデフォルト設定

拡張範囲 VLAN については MTU サイズ、プライベート VLAN、およびリモート SPAN 設定ステートしか変更できません。残りのすべての特性はデフォルト状態のままでなければなりません。

（注） リモート SPAN をサポートするには、スイッチが LAN Base イメージを実行している必要があります。

VLAN の設定方法

標準範囲 VLAN の設定方法

VLAN データベースに新しい標準範囲 VLAN を作成したり、VLAN データベース内の既存の VLAN を変更したりする場合、次のパラメータを設定できます。

- VLAN ID
- VLAN 名
- VLAN タイプ
  - イーサネット
  - Fiber Distributed Data Interface [FDDI]
  - FDDI ネットワーク エンティティ タイトル [NET]
  - TrBRF または TrCRF
  - トークンリング
  - トークンリング Net
- VLAN ステート（アクティブまたは中断）
- Security Association Identifier（SAID）
- TrBRF VLAN のブリッジ識別番号
- FDDI および TrCRF VLAN のリング番号
- TrCRF VLAN の親 VLAN 番号
- TrCRF VLAN のスパニングツリー プロトコル（STP）タイプ
- ある VLAN タイプから別の VLAN タイプに変換するときに使用する VLAN 番号

vlan.dat ファイルを手動で削除しようとすると、VLAN データベースの不整合が生じる可能性があります。VLAN 設定を変更する場合は、この項の手順に従ってください。

イーサネット VLAN の作成または変更

VLAN データベース内の各イーサネット VLAN の ID は 4 桁の一意の数字で、1 ～ 1001 を指定できます。VLAN ID 1002 ～ 1005 は、トークンリングおよび FDDI VLAN 用に予約されています。標準範囲 VLAN を作成して VLAN データベースに追加するには、VLAN に番号および名前を割り当てます。

（注） VTP バージョン 1 および 2 でdeviceが VTP トランスペアレント モードの場合は、1006 を超える VLAN ID を割り当てることができますが、それらを VLAN データベースに追加できません。

手順の概要

1. **enable**
2. **configure terminal**
3. **vlan** *vlan-id*
4. **name** *vlan-name*
5. **mtu** *mtu-size*
6. **remote-span**
7. **end**
8. **show vlan** { **name** *vlan-name* | **id** *vlan-id*}
9. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vlan** *vlan-id* 例： `スイッチ(config)# vlan 20`	VLAN ID を入力して、VLAN コンフィギュレーション モードを開始します。新規の VLAN ID を入力して VLAN を作成するか、または既存の VLAN ID を入力してその VLAN を変更します。 （注） このコマンドで指定できる VLAN ID 範囲は 1 ～ 4094 です。
ステップ 4	**name** *vlan-name* 例： `スイッチ(config-vlan)# name test20`	（任意）VLAN の名前を入力します。VLAN 名を指定しなかった場合には、デフォルトとして、VLAN という語の後ろに先行ゼロを含めた *vlan-id* 値が付加されます。たとえば、VLAN 4 のデフォルトの VLAN 名は VLAN0004 になります。

	コマンドまたはアクション	目的
ステップ 5	**mtu** *mtu-size* 例： `スイッチ(config-vlan)# mtu 256`	（任意）MTU サイズ（または他の VLAN 特性）を変更します。
ステップ 6	**remote-span** 例： `スイッチ(config-vlan)# remote-span`	（任意）リモート スイッチド ポート アナライザ（SPAN）セッションに対する RSPAN VLAN として、VLAN を設定します。
ステップ 7	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show vlan** { **name** *vlan-name* | **id** *vlan-id*} 例： `スイッチ# show vlan name test20 id 20`	入力を確認します。
ステップ 9	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VLAN の削除

VTP サーバー モードのdeviceから VLAN を削除すると、VTP ドメイン内のすべてのdevicesの VLAN データベースから、その VLAN が削除されます。VTP トランスペアレント モードのdeviceから VLAN を削除した場合、その特定のdeviceスイッチ上に限り VLAN が削除されます。

イーサネット VLAN 1 および FDDI、またはトークンリング VLAN 1002 ～ 1005 の、メディアタイプ別のデフォルト VLAN は削除できません。

⚠

注意 VLAN を削除すると、その VLAN に割り当てられていたすべてのポートが非アクティブになります。これらのポートは、新しい VLAN に割り当てられるまで、元の VLAN に（非アクティブで）対応付けられたままです。

手順の概要

1. **enable**
2. **configure terminal**
3. **no vlan** *vlan-id*
4. **end**
5. **show vlan brief**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no vlan** *vlan-id* 例： `スイッチ(config)# no vlan 4`	VLAN ID を入力して、VLAN を削除します。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show vlan brief** 例： `スイッチ# show vlan brief`	VLAN が削除されたことを確認します。
ステップ 6	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

VLAN へのスタティック アクセス ポートの割り当て

VTP をディセーブルにすることによって（VTP トランスペアレント モード）、VTP に VLAN 設定情報をグローバルに伝播させずに、スタティック アクセス ポートを VLAN に割り当てることができます。

存在しない VLAN にインターフェイスを割り当てると、新しい VLAN が作成されます

手順の概要

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **switchport access vlan** *vlan-id*
5. **end**
6. **show running-config interface** *interface-id*
7. **show interfaces** *interface-id* **switchport**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	VLAN に追加するインターフェイスを入力します。
ステップ 3	**switchport mode access** 例： `スイッチ(config-if)# switchport mode access`	ポート（レイヤ 2 アクセス ポート）の VLAN メンバーシップ モードを定義します。
ステップ 4	**switchport access vlan** *vlan-id* 例： `スイッチ(config-if)# switchport access vlan 2`	VLAN にポートを割り当てます。指定できる VLAN ID の範囲は 1 ～ 4094 です。
ステップ 5	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	
ステップ 6	**show running-config interface** *interface-id* 例： スイッチ# **show running-config interface gigabitethernet 1/0/1**	インターフェイスの VLAN メンバーシップ モードを確認します。
ステップ 7	**show interfaces** *interface-id* **switchport** 例： スイッチ# **show interfaces gigabitethernet 1/0/1 switchport**	表示された [Administrative Mode] フィールドおよび [Access Mode VLAN] フィールドの設定を確認します。

拡張範囲 VLAN の設定方法

VTP バージョン 1 およびバージョン 2 でスイッチが VTP トランスペアレント モード（VTP がディセーブル）の場合、拡張範囲 VLAN（1006 ～ 4094）を作成できます。VTP バージョンは、拡張範囲 VLAN をサーバ モードおよびトランスペアレント モードでサポートします。サービス プロバイダーは拡張範囲 VLAN を使用することにより、インフラストラクチャを拡張して、多数の顧客に対応できます。拡張範囲 VLAN ID は、VLAN ID が許可されている **switchport** コマンドで使用できます。

VTP バージョン 1 または 2 での拡張範囲 VLAN の設定は VLAN データベースに格納されません。ただし、VTP モードがトランスペアレントであるため、スイッチの実行コンフィギュレーション ファイルにストアされます。設定をスタートアップ コンフィギュレーション ファイルに保存するには、**copy running-config startup-config** 特権 EXEC コマンドを使用します。VTP バージョン 3 で作成された拡張範囲 VLAN は、VLAN データベースに保存されます。

拡張範囲 VLAN の作成

グローバル コンフィギュレーション モードで拡張範囲 VLAN を作成するには、**vlan** グローバル コンフィギュレーション コマンドを入力し、1006 ～ 4094 の VLAN ID を指定します。拡張範囲 VLAN にはデフォルトのイーサネット VLAN 特性が適用されます。変更できるパラメータは MTU サイズおよび RSPAN 設定のみです。すべてのパラメータのデフォルト設定については、コマンドリファレンスに記載された **vlan** グローバル コンフィギュレーション コマンドの説明を参照してください。VTP バージョン 1 または 2 で、スイッチが VTP トランスペアレント モードでない場合に拡張範囲 VLAN ID を入力すると、VLAN コンフィギュレーション モードの終了時にエラー メッセージが生成され、拡張範囲 VLAN が作成されません。

VTP バージョン 1 および 2 では、拡張範囲 VLAN は VLAN データベースに保存されず、スイッチの実行コンフィギュレーション ファイルに保存されます。拡張範囲 VLAN 設定をスイッチ

のスタートアップ コンフィギュレーション ファイルに保存するには、 **copy running-config startup-config** 特権 EXEC コマンドを使用します。VTP バージョン 3 は、拡張範囲 VLAN を VLAN データベースに保存します。

手順の概要

1. **configure terminal**
2. **vtp mode transparent**
3. **vlan** *vlan-id*
4. **mtu** *mtu size*
5. **remote-span**
6. **end**
7. **show vlan id** *vlan-id*
8. **copy running-config startup config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**vtp mode transparent** 例： `スイッチ(config)# vtp mode transparent`	deviceを VTP トランスペアレント モードで設定し、VTP をディセーブルにします。 （注） この手順は、VTPバージョン3では不要です。
ステップ 3	**vlan** *vlan-id* 例： `スイッチ(config)# vlan 2000` `スイッチ(config-vlan)#`	拡張範囲VLAN IDを入力して、VLANコンフィギュレーションモードを開始します。指定できる範囲は 1006 ～ 4094 です。
ステップ 4	**mtu** *mtu size* 例： `スイッチ(config-vlan)# mtu 1024`	MTU サイズを変更して、VLAN を変更します。
ステップ 5	**remote-span** 例： `スイッチ(config-vlan)# remote-span`	（任意）RSPAN VLANとしてVLANを設定します。

	コマンドまたはアクション	目的
ステップ 6	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show vlan id** *vlan-id* 例： スイッチ# **show vlan id 2000**	VLAN が作成されたことを確認します。
ステップ 8	**copy running-config startup config** 例： スイッチ# **copy running-config startup-config**	device スタートアップ コンフィギュレーション ファイルに設定項目を保存します。 拡張範囲 VLAN 設定を保存するには、device のスタートアップ コンフィギュレーション ファイルに VTP トランスペアレント モード設定と拡張範囲 VLAN 設定を保存する必要があります。これらを保存しないと、device をリセットした場合に、スイッチがデフォルトで VTP サーバー モードになり、拡張範囲 VLAN ID は保存されません。 （注） VTP バージョン 3 では、VLAN が VLAN データベースに保存されるため、この手順は必要ありません。

VLAN のモニタリング

表 *210 :* 特権 *EXEC* 表示コマンド

コマンド	目的
show interfaces [**vlan** *vlan-id*]	device 上に設定されたすべてのインターフェイスまたは特定の VLAN の特性を表示します。

設定例

例：VLAN 名の作成

次に、イーサネット VLAN 20 を作成し、test20 という名前を付け、VLAN データベースに追加する例を示します。

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# name test20
Switch(config-vlan)# end
```

例：アクセス ポートとしてのポートの設定

次に、VLAN 2 のアクセス ポートとしてポートを設定する例を示します。

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
Switch(config-if)# end
```

例：拡張範囲 VLAN の作成

次に、すべてデフォルトの特性で拡張範囲 VLAN を新規作成し、VLAN コンフィギュレーション モードを開始して、新規 VLAN をスイッチのスタートアップ コンフィギュレーション ファイルに保存する例を示します。

```
Switch(config)# vtp mode transparent
Switch(config)# vlan 2000
Switch(config-vlan)# end
Switch# copy running-config startup config
```

次の作業

VLAN を設定したら、次の項目を設定できます。

- VLAN トランキング プロトコル（VTP）
- VLAN トランク
- プライベート VLAN

第 102 章

VLAN トランクの設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

VLAN トランクの前提条件

IEEE 802.1Q トランクは、ネットワークのトランキング方式について次の制約があります。

- IEEE 802.1Q トランクを使用して接続している Cisco devicesのネットワークでは、devices はトランク上で許容される VLAN ごとに 1 つのスパニングツリー インスタンスを維持します。他社製のデバイスは、すべての VLAN でスパニングツリー インスタンスを 1 つサポートする場合があります。

  IEEE 802.1Q トランクを使用して Cisco deviceを他社製のデバイスに接続する場合、Cisco deviceは、トランクの VLAN のスパニングツリー インスタンスを、他社製の IEEE 802.1Q deviceのスパニングツリーインスタンスと結合します。ただし、各 VLANのスパニングツリー情報は、他社製の IEEE 802.1Q devicesからなるクラウドにより分離された Cisco devices

によって維持されます。Cisco devicesを分離する他社製のIEEE 802.1Qクラウドは、devices間の単一トランク リンクとして扱われます。

- IEEE 802.1Qトランクに対応するネイティブVLANが、トランク リンクの両側で一致していなければなりません。トランクの片側のネイティブVLANと反対側のネイティブVLANが異なっていると、スパニングツリー ループが発生する可能性があります。
- ネットワーク上のすべてのネイティブ VLAN についてスパニングツリーをディセーブルにせずに、IEEE 802.1Q トランクのネイティブ VLAN 上のスパニングツリーをディセーブルにすると、スパニングツリー ループが発生することがあります。IEEE 802.1Q トランクのネイティブVLAN上でスパニングツリーをイネーブルのままにしておくか、またはネットワーク上のすべての VLAN でスパニングツリーをディセーブルにすることを推奨します。また、ネットワークにループがないことを確認してから、スパニングツリーをディセーブルにしてください。

VLAN トランクについて

トランキングの概要

トランクとは、1 つまたは複数のイーサネット device インターフェイスと他のネットワーキング デバイス（ルータ、deviceなど）の間のポイントツーポイント リンクです。イーサネット トランクは 1 つのリンクを介して複数の VLAN トラフィックを伝送するので、VLAN をネットワーク全体に拡張できます。

（注） トランクを設定できるのは、1つのイーサネットインターフェイスまたはEtherChannelバンドルに対してです。

トランキング モード

イーサネット トランク インターフェイスは、さまざまなトランキング モードをサポートします。インターフェイスをトランキングまたは非トランキングとして設定したり、ネイバーインターフェイスとトランキングのネゴシエーションを行ったりするように設定できます。トランキングを自動ネゴシエーションするには、インターフェイスが同じ VTP ドメインに存在する必要があります。

トランク ネゴシエーションは、ポイントツーポイントプロトコル（PPP）であるダイナミック トランキング プロトコル（DTP）によって管理されます。ただし、一部のインターネットワーキング デバイスによって DTP フレームが不正に転送されて、矛盾した設定となる場合があります。

レイヤ 2 インターフェイス モード

表 *211 :* レイヤ *2* インターフェイス モード

モード	機能
switchport mode access	インターフェイス（アクセス ポート）を永続的な非トランキング モードにして、リンクの非トランク リンクへの変換をネゴシエートします。インターフェイスは、ネイバーインターフェイスがトランク インターフェイスかどうかに関係なく、非トランク インターフェイスになります。
switchport mode dynamic auto	インターフェイスがリンクをトランク リンクに変換できるようにします。インターフェイスは、ネイバーインターフェイスが **trunk**または **desirable** モードに設定されている場合、トランクインターフェイスになります。すべてのイーサネットインターフェイスのデフォルトのスイッチポート モードは **dynamic auto**です。
switchport mode dynamic desirable	インターフェイスがリンクのトランク リンクへの変換をアクティブに実行するようにします。インターフェイスは、ネイバーインターフェイスが **trunk**、 **desirable** または **auto** モードに設定されている場合、トランクインターフェイスになります。
switchport mode trunk	インターフェイスを永続的なトランキング モードにして、ネイバー リンクのトランク リンクへの変換をネゴシエートします。インターフェイスは、ネイバーインターフェイスがトランク インターフェイスでない場合でも、トランク インターフェイスになります。
switchport nonegotiate	インターフェイスが DTP フレームを生成しないようにします。このコマンドは、インターフェイス スイッチポート モードが **access** または **trunk** の場合だけ使用できます。トランク リンクを確立するには、手動でネイバーインターフェイスをトランク インターフェイスとして設定する必要があります。
switchport mode private-vlan	プライベート VLAN モードを設定します。 （注） **switchport mode private-vlan** コマンドオプションはサポートされていません。

トランクでの許可 VLAN

デフォルトでは、トランク ポートはすべての VLAN に対してトラフィックを送受信します。各トランクですべての VLAN ID（1 ～ 4094）が許可されます。ただし、許可リストから VLAN を削除することにより、それらの VLAN からのトラフィックがトランク上を流れないようにすることができます。

スパニングツリー ループまたはストームのリスクを減らすには、許可リストから VLAN 1 を削除して個々の VLAN トランク ポートの VLAN 1 をディセーブルにできます。トランク ポートから VLAN 1 を削除した場合、インターフェイスは引き続き VLAN 1 内で Cisco Discovery Protocol（CDP）、ポート集約プロトコル（PAgP）、Link Aggregation Control Protocol（LACP）、DTP、および VTP などの管理トラフィックを送受信します。

VLAN 1 をディセーブルにしたトランク ポートが非トランク ポートになると、そのポートはアクセス VLAN に追加されます。アクセス VLAN が 1 に設定されると、**switchport trunk allowed** の設定には関係なく、ポートは VLAN 1 に追加されます。ポート上でディセーブルになっている任意の VLAN について同様のことが当てはまります。

トランク ポートは、VLAN がイネーブルになっており、VTP が VLAN を認識し、なおかつポートの許可リストにその VLAN が登録されている場合に、VLAN のメンバになることができます。VTP が新しくイネーブルにされた VLAN を認識し、その VLAN がトランク ポートの許可リストに登録されている場合、トランク ポートは自動的にその VLAN のメンバになります。VTP が新しい VLAN を認識し、その VLAN がトランク ポートの許可リストに登録されていない場合には、トランク ポートはその VLAN のメンバにはなりません。

トランク ポートでの負荷分散

負荷分散により、devicesに接続しているパラレル トランクの提供する帯域幅が分割されます。STP は通常、ループを防止するために、devices間で 1 つのパラレル リンク以外のすべてのリンクをブロックします。負荷分散を行うと、トラフィックの所属する VLAN に基づいて、リンク間でトラフィックが分散されます。

トランク ポートで負荷分散を設定するには、STP ポート プライオリティまたは STP パス コストを使用します。STP ポート プライオリティを使用して負荷分散を設定する場合には、両方の負荷分散リンクを同じdeviceに接続する必要があります。STP パス コストを使用して負荷分散を設定する場合には、それぞれの負荷分散リンクを同一のdeviceに接続することも、2 台の異なるdevicesに接続することもできます。

STP プライオリティによるネットワーク負荷分散

同一のdevice上の 2 つのポートがループを形成すると、deviceは STP ポート プライオリティを使用して、どのポートをイネーブルとし、どのポートをブロッキング ステートとするかを判断します。パラレル トランク ポートにプライオリティを設定することにより、そのポートに、特定の VLAN のすべてのトラフィックを伝送させることができます。VLAN に対するプライオリティの高い（値の小さい）トランク ポートがその VLAN のトラフィックを転送します。同じ VLAN に対してプライオリティの低い（値の大きい）トランク ポートは、その VLAN に対してブロッキング ステートのままです。1 つのトランク ポートが特定の VLAN に関するすべてのトラフィックを送受信することになります。

STP パス コストによるネットワーク負荷分散

トランクにそれぞれ異なるパス コストを設定し、各パス コストをそれぞれ異なる VLAN 群に対応付け、各 VLAN でポートをブロックすることによって、VLAN トラフィックを分散する

パラレル トランクを設定できます。VLAN はトラフィックを分離し、リンクが失われた場合に備えて冗長性を維持します。

機能の相互作用

トランキングは他の機能と次のように相互作用します。

- トランク ポートをセキュア ポートにすることはできません。
- トランク ポートをまとめて EtherChannel ポート グループにすることはできますが、グループ内のすべてのトランクに同じ設定をする必要があります。グループを初めて作成したときには、そのグループに最初に追加されたポートのパラメータ設定値をすべてのポートが引き継ぎます。次に示すパラメータのいずれかの設定を変更すると、deviceは、入力された設定をグループ内のすべてのポートに伝播します。
  - 許可 VLAN リスト。
  - 各 VLAN の STP ポート プライオリティ。
  - STP PortFast の設定値。
  - トランク ステータス：

    ポート グループ内の 1 つのポートがトランクでなくなると、すべてのポートがトランクでなくなります。
- Per VLAN Spanning Tree（PVST）モードでは最大 24 までのトランク ポート、マルチ スパニング ツリー（MST）モードでは最大 40 までのトランク ポートを設定することを推奨します。
- トランク ポートで IEEE 802.1X をイネーブルにしようとすると、エラー メッセージが表示され、IEEE 802.1X はイネーブルになりません。IEEE 802.1X 対応ポートのモードをトランクに変更しようとしても、ポート モードは変更されません。
- ダイナミック モードのポートは、ネイバーとトランク ポートへの変更をネゴシエートする場合があります。ダイナミック ポートで IEEE 802.1x をイネーブルにしようとすると、エラー メッセージが表示され、IEEE 802.1x はイネーブルになりません。IEEE 802.1x 対応ポートをダイナミックに変更しようとしても、ポート モードは変更されません。

レイヤ 2 イーサネット インターフェイス VLAN のデフォルト設定

次の表に、レイヤ 2 イーサネット インターフェイス VLAN のデフォルト設定を記載します。

表 *212 :* レイヤ *2* イーサネット インターフェイス *VLAN* のデフォルト設定

機能	デフォルト設定
インターフェイス モード	**switchport mode dynamic auto**

機能	デフォルト設定
VLAN 許容範囲	VLAN 1 ～ 4094
プルーニングに適格な VLAN 範囲	VLAN 2 ～ 1001
デフォルト VLAN（アクセス ポート用）	VLAN 1
ネイティブ VLAN（IEEE 802.1Q トランク用）	VLAN 1

VLAN トランクの設定方法

トランクの誤設定を避けるために、DTP をサポートしないデバイスに接続されたインターフェイスが DTP フレームを転送しないように（つまり DTP をオフにするように）設定してください。

- これらのリンク上でトランキングを行わない場合は、**switchport mode access** インターフェイス コンフィギュレーション コマンドを使用して、トランキングをディセーブルにします。
- DTP をサポートしていないデバイスへのトランキングをイネーブルにするには、**switchport mode trunk** および **switchport nonegotiate** インターフェイス コンフィギュレーション コマンドを使用して、インターフェイスがトランクになっても DTP フレームを生成しないように設定します。

トランク ポートとしてのイーサネット インターフェイスの設定

トランク ポートの設定

トランク ポートは VTP アドバタイズを送受信するので、VTP を使用する場合は、device上で少なくとも 1 つのトランク ポートが設定されており、そのトランク ポートが別のdeviceのトランク ポートに接続されていることを確認する必要があります。そうでない場合、deviceは VTP アドバタイズを受信できません。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**}
5. **switchport access vlan** *vlan-id*
6. **switchport trunk native vlan** *vlan-id*
7. **end**
8. **show interfaces** *interface-id* **switchport**
9. **show interfaces** *interface-id* **trunk**

10. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/2**	トランクに設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**} 例： スイッチ(config-if)# **switchport mode dynamic desirable**	インターフェイスをレイヤ2トランクとして設定します（インターフェイスがレイヤ 2 アクセス ポートまたはトンネル ポートであり、トランキング モードを設定する場合に限り必要となります）。 • **dynamic auto**：ネイバー インターフェイスが trunk または desirable モードに設定されている場合に、インターフェイスをトランク リンクとして設定します。これはデフォルトです。 • **dynamic desirable**：ネイバー インターフェイスが trunk、desirable、または auto モードに設定されている場合に、インターフェイスをトランク リンクとして設定します。 • **trunk**：ネイバーインターフェイスがトランク インターフェイスでない場合でも、インターフェイスを永続的なトランキング モードに設定して、リンクをトランク リンクに変換するようにネゴシエートします。
ステップ 5	**switchport access vlan** *vlan-id* 例：	（任意）インターフェイスがトランキングを停止した場合に使用するデフォルトVLANを指定します。

	コマンドまたはアクション	目的
	スイッチ(config-if)# **switchport access vlan 200**	
ステップ 6	**switchport trunk native vlan** *vlan-id* 例： スイッチ(config-if)# **switchport trunk native vlan 200**	IEEE 802.1Q トランク用のネイティブ VLAN を指定します。
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show interfaces** *interface-id* **switchport** 例： スイッチ# **show interfaces gigabitethernet 1/0/2 switchport**	インターフェイスのスイッチ ポート設定を表示します。[Administrative Mode] および [Administrative Trunking Encapsulation] フィールドに表示されます。
ステップ 9	**show interfaces** *interface-id* **trunk** 例： スイッチ# **show interfaces gigabitethernet 1/0/2 trunk**	インターフェイスのトランクの設定を表示します。
ステップ 10	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

トランクでの許可 VLAN の定義

VLAN 1 は、すべての Cisco devicesのすべてのトランク ポートのデフォルト VLAN です。以前は、すべてのトランク リンクで VLAN 1 を必ずイネーブルにする必要がありました。VLAN 1 の最小化機能を使用して、個々の VLAN トランク リンクで VLAN 1 をディセーブルに設定できます。これにより、ユーザー トラフィック（スパニングツリー アドバタイズなど）はVLAN 1 で送受信されなくなります。

手順の概要

1. **enable**

2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode trunk**
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	設定するポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport mode trunk** 例： `スイッチ(config-if)# switchport mode trunk`	インターフェイスを VLAN トランク ポートとして設定します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show interfaces** *interface-id* **switchport** 例： `スイッチ# show interfaces gigabitethernet 1/0/1 switchport`	表示された [Trunking VLANs Enabled] フィールドの設定を確認します。

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

プルーニング適格リストの変更

プルーニング適格リストは、トランク ポートだけに適用されます。トランク ポートごとに独自の適格リストがあります。この手順を有効にするには、VTP プルーニングがイネーブルに設定されている必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet0/1**	VLAN プルーニングを適用するトランク ポートを選択し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport trunk pruning vlan** {**add**	**except**	**none**	**remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]	トランクからのプルーニングを許可する VLAN のリストを設定します。

	コマンドまたはアクション	目的
		add、**except**、**none** および **remove** キーワードの使用方法については、このリリースに対応するコマンドリファレンスを参照してください。 連続していない VLAN ID は、カンマ（スペースなし）で区切ります。ID の範囲はハイフンで指定します。有効な ID 範囲は 2 ～ 1001 です。拡張範囲 VLAN（VLAN ID 1006 ～ 4094）はプルーニングできません。 プルーニング不適格の VLAN は、フラッディングトラフィックを受信します。 デフォルトでは、プルーニングが許可される VLAN のリストには、VLAN 2 ～ 1001 が含まれます。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show interfaces** *interface-id* **switchport** 例： `スイッチ# show interfaces gigabitethernet 1/0/1 switchport`	表示された [Pruning VLANs Enabled] フィールドの設定を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーションファイルに設定を保存します。

タグなしトラフィック用ネイティブ VLAN の設定

IEEE 802.1Q タギングが設定されたトランク ポートは、タグ付きトラフィックおよびタグなしトラフィックの両方を受信できます。デフォルトでは、device はタグなしトラフィックを、ポートに設定されたネイティブ VLAN に転送します。ネイティブ VLAN は、デフォルトでは VLAN 1 です。

ネイティブ VLAN には任意の VLAN ID を割り当てることができます。

パケットの VLAN ID が出力ポートのネイティブ VLAN ID と同じであれば、そのパケットはタグなしで送信されます。ネイティブ VLAN ID と異なる場合は、deviceはそのパケットをタグ付きで送信します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport trunk native vlan** *vlan-id*
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	IEEE 802.1Q トランクとして設定するインターフェイスを定義して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport trunk native vlan** *vlan-id* 例： `スイッチ(config-if)# switchport trunk native vlan 12`	トランク ポート上でタグなしトラフィックを送受信する VLAN を設定します。 *vlan-id* に指定できる範囲は 1 ～ 4094 です。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show interfaces** *interface-id* **switchport** 例： `スイッチ# show interfaces gigabitethernet 1/0/2`	[Trunking Native Mode VLAN] フィールドの設定を確認します。

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

トランク ポートの負荷分散の設定

STP ポート プライオリティによる負荷分散の設定

次の手順では、STP ポート プライオリティを使用した負荷分散を指定してネットワークを設定する方法について説明します。

手順の概要

1. **enable**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **vtp mode server**
5. **end**
6. **show vtp status**
7. **show vlan**
8. **configure terminal**
9. **interface** *interface-id*
10. **switchport mode trunk**
11. **end**
12. **show interfaces** *interface-id* **switchport**
13. デバイス A で、device の 2 番目のポートに対して前述の手順を繰り返します。
14. デバイス B で前述の手順を繰り返し、デバイス A で設定したトランク ポートに接続するトランク ポートを設定します。
15. **show vlan**
16. **configure terminal**
17. **interface** *interface-id*
18. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
19. **exit**
20. **interface** *interface-id*
21. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
22. **end**
23. **show running-config**
24. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	デバイス A で、グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vtp domain** *domain-name* 例： スイッチ(config)# **vtp domain workdomain**	VTP 管理ドメインを設定します。 1 ～ 32 文字のドメイン名を使用できます。
ステップ 4	**vtp mode server** 例： スイッチ(config)# **vtp mode server**	デバイス A を VTP サーバーとして設定します。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show vtp status** 例： スイッチ# **show vtp status**	デバイス A およびデバイス B の両方で、VTP 設定を確認します。 表示された *VTP Operating Mode* および *VTP Domain Name* フィールドをチェックします。
ステップ 7	**show vlan** 例： スイッチ# **show vlan**	デバイス A のデータベースに VLAN が存在していることを確認します。
ステップ 8	**configure terminal** 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 9	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet1/0/1**	トランクとして設定するインターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 10	**switchport mode trunk** 例： スイッチ(config-if)# **switchport mode trunk**	ポートをトランク ポートとして設定します。
ステップ 11	**end** 例： スイッチ(config-if)# **end**	特権 EXEC モードに戻ります。
ステップ 12	**show interfaces** *interface-id* **switchport** 例： スイッチ# **show interfaces gigabitethernet 1/0/1 switchport**	VLAN の設定を確認します。
ステップ 13	デバイス A で、device の 2 番目のポートに対して前述の手順を繰り返します。	
ステップ 14	デバイス B で前述の手順を繰り返し、デバイス A で設定したトランク ポートに接続するトランク ポートを設定します。	
ステップ 15	**show vlan** 例： スイッチ# **show vlan**	トランク リンクがアクティブになると、VTP がデバイス B に VTP および VLAN 情報を渡します。このコマンドは、デバイス B が VLAN 設定を学習したことを確認します。
ステップ 16	**configure terminal** 例： スイッチ# **configure terminal**	デバイス A で、グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 17	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/1`	STP のポート プライオリティを設定するインターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 18	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* 例： `スイッチ(config-if)# spanning-tree vlan 8-10 port-priority 16`	指定された VLAN 範囲にポート プライオリティを割り当てます。0～240 のポート プライオリティ値を入力します。ポート プライオリティ値は 16 ずつ増分します。
ステップ 19	**exit** 例： `スイッチ(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 20	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 1/0/2`	STP のポート プライオリティを設定するインターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 21	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* 例： `スイッチ(config-if)# spanning-tree vlan 3-6 port-priority 16`	指定された VLAN 範囲にポート プライオリティを割り当てます。0～240 のポート プライオリティ値を入力します。ポート プライオリティ値は 16 ずつ増分します。
ステップ 22	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 23	**show running-config** 例： `スイッチ# show running-config`	入力を確認します。

	コマンドまたはアクション	目的
ステップ 24	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

STP パス コストによる負荷分散の設定

次の手順では、STP パス コストを使用した負荷分散を指定してネットワークを設定する方法について説明します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode trunk**
5. **exit**
6. デバイス A 内の別のインターフェイスでステップ 2 ～ 4 を繰り返します。
7. **end**
8. **show running-config**
9. **show vlan**
10. **configure terminal**
11. **interface** *interface-id*
12. **spanning-tree vlan** *vlan-range* **cost** *cost-value*
13. **end**
14. デバイス A に設定したもう一方のトランク インターフェイスでステップ 9 ～ 13 を繰り返し、VLAN 8、9、および 10 のスパニングツリー パス コストを 30 に設定します。
15. **exit**
16. **show running-config**
17. **copy running-config startup-config**

手順の詳細

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

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 3	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	トランクとして設定するインターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport mode trunk** 例： スイッチ(config-if)# **switchport mode trunk**	ポートをトランク ポートとして設定します。
ステップ 5	**exit** 例： スイッチ(config-if)# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 6	デバイス A 内の別のインターフェイスでステップ 2 ～ 4 を繰り返します。	
ステップ 7	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。画面で、インターフェイスがトランク ポートとして設定されていることを確認してください。
ステップ 9	**show vlan** 例： スイッチ# **show vlan**	トランク リンクがアクティブになると、デバイス A がもう一方の devices から VTP 情報を受信します。このコマンドは、デバイス A が VLAN コンフィギュレーションを学習したことを確認します。
ステップ 10	**configure terminal** 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	スイッチ# **configure terminal**	
ステップ 11	**interface** *interface-id* 例： スイッチ(config)# **interface gigabitethernet 1/0/1**	STP コストを設定するインターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 12	**spanning-tree vlan** *vlan-range* **cost** *cost-value* 例： スイッチ(config-if)# **spanning-tree vlan 2-4 cost 30**	VLAN 2 ～ 4 のスパニングツリー パス コストを 30 に設定します。
ステップ 13	**end** 例： スイッチ(config-if)# **end**	グローバル コンフィギュレーション モードに戻ります。
ステップ 14	デバイス A に設定したもう一方のトランク インターフェイスでステップ 9 ～ 13 を繰り返し、VLAN 8、9、および 10 のスパニングツリー パス コストを 30 に設定します。	
ステップ 15	**exit** 例： スイッチ(config)# **exit**	特権 EXEC モードに戻ります。
ステップ 16	**show running-config** 例： スイッチ# **show running-config**	入力を確認します。両方のトランク インターフェイスに対してパス コストが正しく設定されていることを表示で確認します。
ステップ 17	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

VLAN トランキングの設定例

例：トランク ポートの設定

次に、IEEE 802.1Q トランクとしてポートを設定する例を示します。この例では、ネイバー インターフェイスが IEEE 802.1Q トランキングをサポートするように設定されていることを前提としています。

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport mode dynamic desirable
Switch(config-if)# end
```

例：ポートからの VLAN の削除

次に、ポートの許可 VLAN リストから VLAN 2 を削除する例を示します。

```
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# switchport trunk allowed vlan remove 2
Switch(config-if)# end
```

次の作業

VLAN トランクを設定したら、次の項目を設定できます。

- VLAN
- プライベート VLAN

第 103 章

VMPS の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

VMPS の前提条件

ダイナミックアクセス ポートとしてポートを設定する前に、VLAN メンバーシップ ポリシー サーバー（VMPS）を設定する必要があります。

ポートをダイナミックアクセス ポートとして設定すると、そのポートに対してスパニングツリーの PortFast 機能が自動的にイネーブルになります。PortFast モードにより、ポートをフォワーディング ステートに移行させるプロセスが短縮されます。

VMPS クライアントと VMPS サーバーの VTP 管理ドメインは、同じでなければなりません。

VMPS の制約事項

次に、VMPS を設定する際の制約事項を示します。

- IEEE 802.1x ポートをダイナミックアクセス ポートとして設定することはできません。ダイナミックアクセス（VQP）ポートでIEEE 802.1xをイネーブルにしようとすると、エラーメッセージが表示され、IEEE 802.1x はイネーブルになりません。IEEE 802.1x 対応ポートを変更してダイナミック VLAN を割り当てようとしても、エラーメッセージが表示され、VLAN 設定は変更されません。
- トランクポートをダイナミックアクセスポートにすることはできませんが、トランクポートに対して **switchport access vlan dynamic** インターフェイス コンフィギュレーション コマンドを入力することは可能です。その場合、device の設定は維持され、後にアクセスポートとして設定された場合には、その設定が適用されます。ダイナミックアクセス設定を有効にするには、ポート上でトランキングをオフにしておく必要があります。
- ダイナミックアクセス ポートをモニター ポートにすることはできません。
- セキュア ポートをダイナミックアクセス ポートにすることはできません。ポートをダイナミックにするには、ポート上でポートセキュリティをディセーブルにしておく必要があります。
- プライベート VLAN ポートは、ダイナミックアクセス ポートにできません。
- ダイナミックアクセス ポートを EtherChannel グループのメンバにすることはできません。
- ポート チャネルをダイナミックアクセス ポートとして設定することはできません。
- VMPS サーバー上に設定された VLAN を音声 VLAN にしないでください。

VMPS について

ダイナミック VLAN 割り当て

VLAN Query Protocol（VQP）は、ダイナミックアクセス ポートをサポートする場合に使用します。ダイナミックアクセス ポートは VLAN に永続的に割り当てられるのではなく、ポートで認識された MAC（メディア アクセス コントロール）送信元アドレスに基づいて VLAN を割り当てます。未知の MAC アドレスが検出されるたびに、deviceはリモート VLAN メンバーシップ ポリシー サーバー（VMPS）に VQP クエリーを送信します。そのクエリーには、新たに検出された MAC アドレスおよび検出場所のポートが含まれます。VMPS はそのポートの VLAN 割り当てで応答します。このdeviceをVMPSサーバーにすることはできませんが、VMPS のクライアントとして機能させ、VQP を介して通信することができます。

クライアント deviceは新しいホストの MAC アドレスを受信するたびに、VMPS に VQP クエリーを送信します。このクエリーを受信したVMPSは、データベースでMACアドレスとVLANのマッピングを検索します。サーバーの応答は、このマッピングと、サーバーがオープン モー

ドかセキュア モードかに基づいて行われます。セキュア モードの場合、サーバーは不正なホストが検出されると、ポートをシャットダウンします。オープン モードでは、サーバーはホストに対してポート アクセスを拒否します。

ポートが未割り当ての場合（つまり、VLAN 割り当てがまだ設定されていない場合）、VMPS は次のいずれかの応答を行います。

- そのポートでホストが許可されている場合、VMPS は割り当てられた VLAN 名を指定し、ホストへのアクセスを許可する VLAN 割り当て応答をクライアントに送信します。
- そのポートでホストが許可されておらず、なおかつ VMPS がオープン モードの場合、VMPS はアクセス拒否応答を送信します。
- そのポートで VLAN が許可されておらず、なおかつ VMPS がセキュア モードの場合、VMPS はポートシャットダウン応答を送信します。

ポートに VLAN 割り当てがすでに設定されている場合、VMPS は次のいずれかの応答を行います。

- データベース内の VLAN がポート上の現在の VLAN と一致した場合、VMPS は成功応答を送信し、ホストへのアクセスを許可します。
- データベース内の VLAN がポート上の現在の VLAN と一致せず、なおかつポート上にアクティブ ホストが存在する場合、VMPS は VMPS のセキュア モードに応じて、アクセス拒否またはポートシャットダウン応答を送信します。

VMPS からアクセス拒否応答を受け取った場合、deviceはそのホスト MAC アドレスとの間のトラフィックを引き続きブロックします。deviceはポート宛てのパケットを引き続きモニターし、新しいホスト アドレスを検出すると VMPS にクエリーを送信します。VMPS からポートシャットダウン応答を受信した場合、deviceはそのポートをディセーブルにします。Network Assistant、CLI（コマンドライン インターフェイス）、または SNMP（簡易ネットワーク管理プロトコル）を使用して、ポートを手動で再びイネーブルにする必要があります。

ダイナミックアクセス ポート VLAN メンバーシップ

ダイナミックアクセス ポートが所属できるのは、VLAN ID が 1 ～ 4094 の 1 つの VLAN だけです。リンクがアクティブになっても、VMPS によって VLAN が割り当てられるまで、device はこのポートとの間のトラフィック転送を行いません。VMPS は、ダイナミックアクセス ポートに接続した新しいホストの最初のパケットから送信元 MAC アドレスを受信し、VMPS データベースの VLAN とその MAC アドレスを照合します。

一致した場合、VMPS はそのポートの VLAN 番号を送信します。クライアント deviceがまだ設定されていない場合、VMPS からトランク ポートで受信した最初の VTP パケットからのドメイン名を使用します。クライアント deviceがすでに設定されている場合は、クエリ パケットにスイッチのドメイン名を含めて VMPS に送信し、VLAN 番号を取得します。VMPS はパケット内のドメイン名が自身のドメイン名と一致することを確認した後、要求を受け入れ、クライアントに割り当てられた VLAN 番号を応答します。一致しない場合、（VMPS セキュア モードの設定に応じて）VMPS は要求を拒否するか、ポートをシャットダウンします。

ダイナミックアクセス ポート上で複数のホスト（MAC アドレス）をアクティブにできますが、それらのホストはすべて同じ VLAN に存在する必要があります。ただし、ポート上でアクティブなホスト数が 20 を超えると、VMPS はダイナミックアクセス ポートをシャットダウンします。

ダイナミックアクセス ポート上でリンクがダウンになると、ポートは切り離された状態に戻り、VLANの所属から外れます。ポート経由でオンラインになるホストはVMPSによってVQP経由で再チェックされてから、ポートが VLAN に割り当てられます。

ダイナミックアクセスポートは、直接ホスト接続に使用したり、ネットワークに接続したりできます。device上のポートごとに、最大 20 個の MAC アドレスを使用できます。ダイナミックアクセス ポートが一度に所属できる VLAN は 1 つだけですが、VLAN は検出された MAC アドレスに基づいて後で変更されることがあります。

デフォルトの VMPS クライアント設定

次の表に、クライアント スイッチ上のデフォルトの VMPS およびダイナミックアクセス ポートの設定を記載します。

表 *213 : VMPS* クライアントおよびダイナミックアクセス ポートのデフォルト設定

機能	デフォルト設定
VMPS ドメイン サーバー	なし
VMPS 再確認インターバル	60 分
VMPS サーバー再試行回数	3
ダイナミックアクセス ポート	未設定

VMPS の設定方法

VMPS の IP アドレスの入力

（注） スイッチ クラスタに対して VMPS を定義する場合は、コマンド スイッチにこのアドレスを入力する必要があります。

始める前に

スイッチをクライアントとして設定するには、サーバーの IP アドレスを最初に入力する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **vmps server** *ipaddress* **primary**
4. **vmps server** *ipaddress*
5. **end**
6. **show vmps**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vmps server** *ipaddress* **primary** 例： スイッチ(config)# **vmps server 10.1.2.3 primary**	プライマリ VMPS サーバーとして機能するdeviceの IP アドレスを入力します。
ステップ 4	**vmps server** *ipaddress* 例： スイッチ(config)# **vmps server 10.3.4.5**	（任意）セカンダリ VMPS サーバーとして機能するdeviceの IP アドレスを入力します。 セカンダリ サーバーのアドレスは、3 つまで入力できます。
ステップ 5	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 6	**show vmps** 例： スイッチ# **show vmps**	表示された [VMPS Domain Server] フィールドの設定を確認します。

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

VMPS クライアント上のダイナミックアクセス ポートの設定

注意 ダイナミックアクセス ポート VLAN メンバーシップはエンド ステーション用、またはエンド ステーションに接続されたハブ用です。他のスイッチにダイナミックアクセス ポートを接続すると、接続が切断されることがあります。

クラスタメンバーdeviceのポートをダイナミックアクセスポートとして設定する場合には、最初に **rcommand** 特権 EXEC コマンドを使用して、そのクラスタメンバー deviceにログインします。

始める前に

ダイナミックアクセス ポートを動作させるには、VMPS に IP 接続できなければなりません。IP 接続が可能かどうかをテストするには、VMPS の IP アドレスに ping を実行し、応答が得られるかどうかを確認します。

（注） インターフェイスをデフォルト設定に戻すには、**default interface** *interface-id* インターフェイス コンフィギュレーション コマンドを使用します。インターフェイスをデフォルトのスイッチポートモード（dynamic auto）に戻すには、**no switchport mode** インターフェイス コンフィギュレーション コマンドを使用します。アクセスモードを device のデフォルト VLAN にリセットするには、**no switchport access vlan** インターフェイス コンフィギュレーション コマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode access**
5. **switchport access vlan dynamic**
6. **end**
7. **show interfaces** *interface-id* **switchport**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet 0/1`	エンド ステーションに接続するdevice ポートを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport mode access** 例： `スイッチ(config-if)# switchport mode access`	ポートをアクセス モードに設定します。
ステップ 5	**switchport access vlan dynamic** 例： `スイッチ(config-if)# switchport access vlan dynamic`	ポートをダイナミック VLAN メンバーシップ適格として設定します。 ダイナミックアクセス ポートは、エンド ステーションに接続されている必要があります。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show interfaces** *interface-id* **switchport** 例： `スイッチ# show interfaces gigabitethernet 0/1 switchport`	表示された [Operational Mode] フィールドの設定を確認します。

	コマンドまたはアクション	目的
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

VLAN メンバーシップの再確認

このタスクでは、deviceが VMPS から受信したダイナミックアクセス ポート VLAN メンバーシップの割り当てを確認します。

手順の概要

1. **enable**
2. **vmps reconfirm**
3. **show vmps**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**vmps reconfirm** 例： スイッチ# **vmps reconfirm**	ダイナミックアクセス ポート VLAN メンバーシップを再確認します。
ステップ 3	**show vmps** 例： スイッチ# **show vmps**	ダイナミック VLAN 再確認ステータスを確認します。

再確認インターバルの変更

VMPS クライアントは、VMPS から受信した VLAN メンバーシップ情報を定期的に再確認します。この再確認を行う間隔を分単位で設定できます。

（注） クラスタのメンバdeviceを設定する場合、このパラメータはコマンドdeviceの再確認インターバルの設定値以上でなければなりません。また、メンバーdeviceにログインするには、最初に **rcommand** 特権 EXEC コマンドを使用する必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **vmps reconfirm** *minutes*
4. **end**
5. **show vmps**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vmps reconfirm** *minutes* 例： `スイッチ(config)# vmps reconfirm 90`	ダイナミック VLAN メンバーシップの再確認を行う間隔（分）を設定します。指定できる範囲は 1 ～ 120 です。デフォルトは 60 分です。
ステップ 4	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show vmps** 例： `スイッチ# show vmps`	表示された [Reconfirm Interval] フィールドのダイナミック VLAN の再確認ステータスを確認します。

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

再試行回数の変更

deviceが次のサーバーにクエリーを送信する前に VMPS への接続を試行する回数を変更するには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **vmps retry** *count*
4. **end**
5. **show vmps**
6. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： スイッチ> **enable**	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： スイッチ# **configure terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vmps retry** *count* 例： スイッチ(config)# **vmps retry 5**	再試行の回数を変更します。指定できる再試行回数の範囲は 1 ～ 10 です。デフォルトは 3 です。
ステップ 4	**end** 例：	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config)# **end**	
ステップ 5	**show vmps** 例： スイッチ# **show vmps**	表示された [Server Retry Count] フィールドの設定を確認します。
ステップ 6	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

ダイナミックアクセス ポート VLAN メンバーシップのトラブルシューティング

問題 VMPS は次の状況でダイナミックアクセス ポートをシャットダウンします。

- **問題** VMPS がセキュア モードであり、なおかつホストのポートへの接続を許可しない場合。VMPS はポートをシャットダウンして、ホストがネットワークに接続できないようにします。
- **問題** ダイナミックアクセス ポート上のアクティブ ホストが 20 を超えた場合。

解決法 ディセーブルになっているダイナミックアクセス ポートを再びイネーブルにするには、**no shutdown** インターフェイス コンフィギュレーション コマンドに続けて、**shutdown** インターフェイス コンフィギュレーション コマンドを入力します。

VMPS のモニターリング

show vmps 特権 EXEC コマンドを使用して、VMPS に関する情報を表示できます。device は VMPS に関する次の情報を表示します。

- VMPS VQP バージョン：VMPS との通信に使用する VQP のバージョン。device は VQP バージョン 1 を使用する VMPS にクエリーを送信します。
- 再確認インターバル：device が VLAN と MAC アドレスの割り当てを再確認する間隔（分）。
- サーバー再試行回数：VQP が VMPS にクエリーを再送信する回数。この回数試行しても応答が得られない場合、device はセカンダリ VMPS へのクエリーを開始します。

- VMPS ドメイン サーバー：設定されている VLAN メンバーシップ ポリシー サーバーの IP アドレス。deviceスイッチは*current*と表示されているサーバーにクエリーを送信します。*primary* と表示されているサーバーは、プライマリ サーバーです。
- VMPS 動作：最近の再確認の結果。再確認は、再確認インターバルが経過したときに自動的に行われますが、**vmps reconfirm** 特権 EXEC コマンドを入力するか、Network Assistant あるいはSNMP で同等の操作を行うことによって、強制的に再確認することもできます。

次に、**show vmps** 特権 EXEC コマンドの出力例を示します。

```
スイッチ# show vmps
VQP Client Status:
--------------------
VMPS VQP Version:   1
Reconfirm Interval: 60 min
Server Retry Count: 3
VMPS domain server: 172.20.128.86 (primary, current)
                    172.20.128.87

Reconfirmation status
---------------------
VMPS Action:         other
```

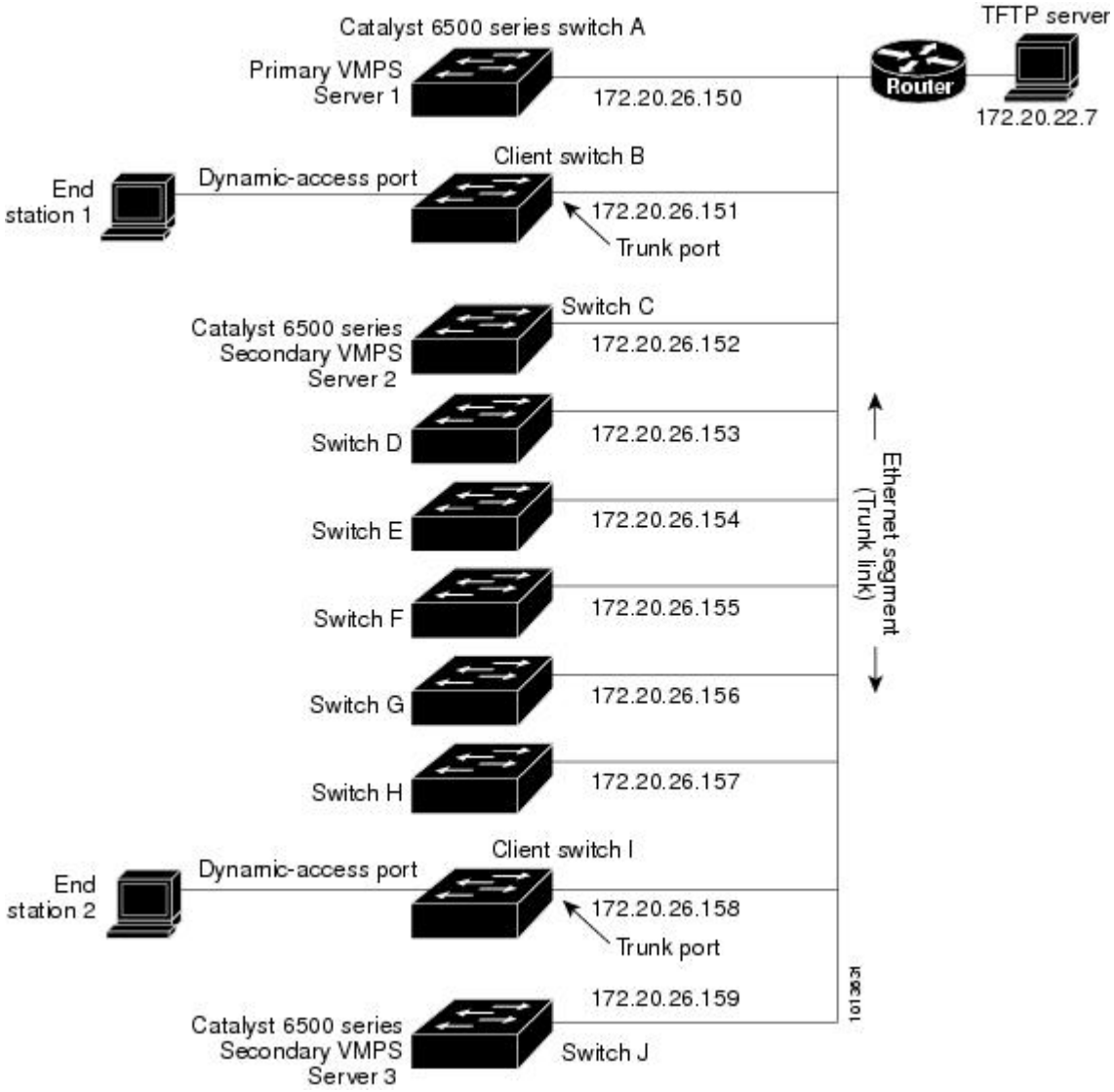
VMPS の設定例

例：VMPS の設定

図 *135* : ダイナミック ポート *VLAN* メンバーシップの構成例

VMPS サーバー スイッチと VMPS クライアント スイッチでダイナミックアクセス ポートを使用するこのネットワークは、次のように設定されます。

- VMPS サーバーと VMPS クライアントは、それぞれ別のスイッチです。
- Catalyst 6500 シリーズのスイッチ A が、プライマリ VMPS サーバーです。
- Catalyst 6500 シリーズのスイッチ C およびスイッチ J が、セカンダリ VMPS サーバーです。
- エンド ステーションはクライアント（スイッチ B、スイッチ I）に接続されています。
- データベース コンフィギュレーション ファイルは、IP アドレス 172.20.22.7 の TFTP サーバーに保存されています。



次の作業

次の設定を行えます。

- VTP
- VLAN
- VLAN トランキング
- プライベート VLAN
- 音声 VLAN

次の作業

第 104 章

音声 VLAN の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

音声 VLAN の前提条件

音声 VLAN の前提条件は、次のとおりです。

- 音声 VLAN 設定はdeviceのアクセスポートだけでサポートされており、トランクポートではサポートされていません。

（注） トランク ポートは、標準 VLAN と同様に、任意の数の音声 VLAN を伝送できます。トランク ポートでは、音声 VLAN の設定がサポートされません。

- 音声 VLAN をイネーブルにする前に、**mls qos** グローバル コンフィギュレーション コマンドを入力してdevice上でQoSをイネーブルに設定し、さらに**mls qos trust cos** インターフェイス コンフィギュレーション コマンドを入力してポートの信頼状態を trust に設定しておくことを推奨します。Auto-QoS 機能を使用すると、これらは自動的に設定されます。
- Cisco IP Phone にコンフィギュレーションを送信するために、Cisco IP Phone に接続するdeviceポート上でCDPをイネーブルにする必要があります（デフォルト設定では、CDPがすべてのdeviceインターフェイスでグローバルにイネーブルです）。

音声 VLAN の制約事項

音声 VLAN には、スタティック セキュア MAC アドレスを設定できません。

音声 VLAN に関する情報

音声 VLAN

音声 VLAN 機能を使用すると、アクセス ポートで IP Phone からの IP 音声トラフィックを伝送できます。deviceを Cisco 7960 IP Phone に接続すると、IP Phone はレイヤ 3 IP 値およびレイヤ 2サービスクラス（CoS）値を使用して、音声トラフィックを送信します。どちらの値もデフォルトでは 5 に設定されます。データ送信が均質性に欠ける場合、IP Phone の音質が低下することがあります。そのため、このdeviceは IEEE 802.1p CoS に基づく Quality of Service（QoS）をサポートしています。QoS は、分類およびスケジューリングを使用して、deviceからのネットワーク トラフィックを予測可能な方法で送信します。

Cisco 7960 IP Phone は設定可能なデバイスであり、IEEE 802.1p の優先度に基づいてトラフィックを転送するように設定できます。Cisco IP Phone によって割り当てられたトラフィックの優先度を信頼したり、オーバーライドしたりするようにdeviceを設定できます。

Cisco IP Phone の音声トラフィック

Cisco IP Phone と接続するアクセス ポートを、1 つの VLAN は音声トラフィック用に、もう 1 つの VLAN は Cisco IP Phone に接続しているデバイスからのデータ トラフィック用に使用するように設定できます。Cisco Discovery Protocol（CDP）パケットを送信するよう、device上のアクセス ポートを設定できます。CDP パケットは、接続する IP Phone に対して、次のいずれかの方法で音声トラフィックをdeviceに送信するよう指示します。

- レイヤ 2 CoS プライオリティ値のタグ付き音声 VLAN による送信
- レイヤ 2 CoS プライオリティ値のタグ付きアクセス VLAN による送信
- タグなし（レイヤ 2 CoS プライオリティ値なし）のアクセス VLAN による送信

（注） いずれの設定でも、音声トラフィックはレイヤ 3 IP precedence 値（音声トラフィックはデフォルトで 5、音声制御トラフィックは 3）を伝送します。

Cisco IP Phone のデータ トラフィック

deviceは、Cisco IP Phoneのアクセス ポートに接続されたデバイスから送られる、タグ付きデータ トラフィック（IEEE 802.1Q または IEEE 802.1p フレーム タイプのトラフィック）を処理することもできます。CDP パケットを送信するよう、device上のレイヤ 2 アクセス ポートを設定できます。CDP パケットは、接続する IP Phone に対して、次のいずれかのモードで IP Phone アクセス ポートを設定するよう指示します。

- trusted（信頼性がある）モードでは、Cisco IP Phone のアクセス ポート経由で受信したすべてのトラフィックがそのまま IP Phone を通過します。
- untrusted（信頼性がない）モードでは、Cisco IP Phone のアクセス ポート経由で受信した IEEE 802.1Q および IEEE 802.1p フレームのすべてのトラフィックに、設定されたレイヤ 2 CoS 値を与えます。デフォルトのレイヤ 2 CoS 値は 0 です。信頼できないモードがデフォルト設定です。

（注） Cisco IP Phone に接続されたデバイスからのタグなしトラフィックは、IP Phone のアクセス ポートの信頼状態に関係なく、そのまま IP Phone を通過します。

音声 VLAN 設定時の注意事項

- Cisco 7960 IP Phone は、PC やその他のデバイスとの接続もサポートしているので、device を Cisco IP Phone に接続するポートは、さまざまな種類のトラフィックを伝送できます。ポートを設定することによって、Cisco IP Phone による音声トラフィックおよびデータ トラフィックの伝送方法を決定できます。
- IP Phone で音声 VLAN 通信が適切に行われるには、device上に音声 VLAN が存在し、アクティブになっている必要があります。VLAN が存在しているかどうかを確認するには、**show vlan** 特権 EXEC コマンドを使用します（リストで表示されます）。VLAN がリストされていない場合は、音声 VLAN を作成します。

- Power Over Ethernet（PoE）devicesは、シスコ先行標準の受電デバイスまたは IEEE 802.3af 準拠の受電デバイスが AC 電源から電力を供給されてない場合に、それらの受電デバイスに自動的に電力を供給できます。
- 音声 VLAN を設定すると、PortFast 機能が自動的にイネーブルになります。音声 VLAN をディセーブルにしても、PortFast 機能は自動的にディセーブルになりません。
- Cisco IP Phone とその IP Phone に接続されたデバイスが同じ VLAN 上にある場合、両方とも同じ IP サブネットに属していなければなりません。次の条件が満たされている場合は、同じ VLAN 上にあります。
  - 両方とも IEEE 802.1p またはタグなしフレームを使用する。
  - Cisco IP Phone が IEEE 802.1p フレームを使用し、デバイスがタグなしフレームを使用する。
  - Cisco IP Phone がタグなしフレームを使用し、デバイスが IEEE 802.1p フレームを使用する。
  - Cisco IP Phone が IEEE 802.1Q フレームを使用し、音声 VLAN がアクセス VLAN と同じである。
- Cisco IP Phone と IP Phone に接続されたデバイスは、同一 VLAN、同一サブネット上にあっても、使用するフレーム タイプが異なる場合は通信できません。トラフィックは同一サブネット上でルーティングされないからです（ルーティングによってフレーム タイプの相違が排除されます）。
- 音声 VLAN ポートには次のポート タイプがあります。
  - ダイナミック アクセス ポート。
  - IEEE 802.1x 認証ポート。

（注） 音声 VLAN が設定され Cisco IP Phone が接続されているアクセス ポートで IEEE 802.1x を有効にした場合、その IP Phone からdeviceへの接続が最大 30 秒間失われます。

  - 保護ポート。
  - SPAN または RSPAN セッションの送信元ポートまたは宛先ポート。
  - セキュア ポート。

（注） 音声 VLAN も設定しているインターフェイス上でポート セキュリティをイネーブルにする場合、ポートで許容されるセキュアアドレスの最大数を、アクセスVLANにおけるセキュア アドレスの最大数に 2 を足した数に設定する必要があります。ポートを Cisco IP Phone に接続している場合、IP Phone に最大で 2 つの MAC アドレスが必要になります。IP Phone のアドレスは、音声 VLAN で学習され、アクセス VLAN でも学習される場合があります。PC を IP Phone に接続する場合、追加の MAC アドレスが必要になります。

音声 VLAN のデフォルト設定

音声 VLAN 機能は、デフォルトではディセーブルに設定されています。

音声 VLAN 機能がイネーブルの場合、すべてのタグなしトラフィックはポートのデフォルトの CoS プライオリティに従って送信されます。

IEEE 802.1p または IEEE 802.1Q のタグ付きトラフィックでは、CoS 値が信頼されません。

音声 VLAN の設定方法

Cisco IP Phone の音声トラフィックの設定

Cisco IP Phone に CDP パケットを送信して IP Phone による音声トラフィックの送信方法を設定するように、IP Phone に接続するポートを設定できます。IP Phone は指定された音声 VLAN に、レイヤ 2 CoS 値を使用して、IEEE 802.1Q フレームの音声トラフィックを伝送できます。IEEE 802.1p のプライオリティ タグを使用すると、音声トラフィックにさらに高いプライオリティを与え、すべての音声トラフィックをネイティブ（アクセス）VLAN 経由で転送できます。Cisco IP Phone はタグなしの音声トラフィックを送信する、または独自の設定を使用してアクセス VLAN で音声トラフィックを送信することもできます。いずれの設定でも、音声トラフィックはレイヤ 3 IP precedence 値（デフォルトは 5）を伝送します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **mls qos trust cos**
5. **switchport voice** {**vlan**{*vlan-id* | **dot1p** | **none** | **untagged**}}
6. **end**
7. 次のいずれかを使用します。

- **show interfaces** *interface-id* **switchport**
- **show running-config interface** *interface-id*

8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `Device(config)# interface gigabitethernet 1/0/1`	IP Phone に接続するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**mls qos trust cos** 例： `Device(config-if)# mls qos trust cos`	パケットの CoS 値を使用して着信トラフィック パケットを分類するよう、インターフェイスを設定します。タグなしパケットの場合、ポートのデフォルト CoS 値が使用されます。 （注） ポートの信頼状態を設定する前に、最初に **mls qos** グローバル コンフィギュレーション コマンドを使用して、QoS をグローバルでイネーブルに設定しておく必要があります。
ステップ 5	**switchport voice** {**vlan**{*vlan-id* | **dot1p** | **none** | **untagged**}} 例： `Device(config-if)# switchport voice vlan dot1p`	音声 VLAN を設定します。 • *vlan-id*：すべての音声トラフィックが特定の VLAN を経由して転送されるように IP Phone を設定します。デフォルトでは、Cisco IP Phone は IEEE 802.1p プライオリティ 5 を使用して音声トラフィックを転送します。指定できる VLAN ID の範囲は 1 ～ 4094 です。 • **dot1p**：VLAN ID 0（ネイティブ VLAN）のタグが付けられた音声およびデータ IEEE 802.1p プライオリティフレームを受け入れるよう、device

	コマンドまたはアクション	目的
		を設定します。デフォルトでは、deviceはVLAN 0 のタグが付いたすべての音声およびデータトラフィックをドロップします。802.1p に対応するよう設定されると、Cisco IP Phone は IEEE 802.1p プライオリティ 5 を使用してトラフィックを転送します。 • **none**：IP Phone が独自の設定を使用してタグなしの音声トラフィックを送信するようにします。 • **untagged**：タグなしの音声トラフィックを送信するように電話を設定します。
ステップ 6	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 7	次のいずれかを使用します。 • **show interfaces** *interface-id* **switchport** • **show running-config interface** *interface-id* 例： `Device# show interfaces gigabitethernet 1/0/1 switchport` または `Device# show running-config interface gigabitethernet 1/0/1`	音声 VLAN の設定、または QoS および音声 VLAN の設定を確認します。
ステップ 8	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

着信データ フレームのプライオリティ設定

PC またはその他のデータ デバイスを Cisco IP Phone ポートに接続できます。タグ付きデータトラフィック（IEEE 802.1Q または IEEE 802.1p フレーム）を処理するために、CDP パケットを送信するようdeviceを設定できます。CDP パケットは Cisco IP Phone に対して、IP Phone 上

のアクセス ポートに接続されたデバイスからのデータ パケット送信方法を指示します。PC は、CoS 値が割り当てられたパケットを生成できます。接続デバイスから IP Phone のポートに届いたフレームのプライオリティを変更しない（信頼する）または変更する（信頼しない）ように、IP Phone を設定できます。

Cisco IP Phone で非音声ポートから受信するデータ トラフィックのプライオリティを設定するには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport priority extend** { **cos** *value* | **trust**}
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/1`	Cisco IP Phone に接続するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport priority extend** { **cos** *value*	**trust**} 例： `スイッチ(config-if)# switchport priority extend trust`	Cisco IP Phone のアクセス ポートから受信したデータ トラフィックのプライオリティを次のように設定します。 • **cos** *value*：PC または接続しているデバイスから受信したプライオリティを、指定の CoS 値にオーバーライドするよう、IP Phone を設定します。値は 0 ～ 7 です。7 が最高のプライオリティ

	コマンドまたはアクション	目的
		です。デフォルトのプライオリティは、**cos** 0 です。 • **trust**：PC または接続しているデバイスから受信したプライオリティを信頼するよう IP Phone アクセス ポートを設定します。
ステップ 5	**end** 例： `スイッチ(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show interfaces** *interface-id* **switchport** 例： `スイッチ# show interfaces gigabitethernet1/0/1 switchport`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `スイッチ# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

音声 VLAN のモニタリング

インターフェイスの音声 VLAN 設定を表示するには、**show interfaces** *interface-id* **switchport** 特権 EXEC コマンドを使用します。

設定例

例：Cisco IP Phone の音声トラフィックの設定

次の例では、CoS 値を使用して着信トラフィックを分類し、VLAN ID 0 のタグが付いた音声およびデータ プライオリティ トラフィックを受け付けるよう、Cisco IP Phone に接続しているポートを設定する方法について示します。

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

```
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# switchport voice vlan dot1p
Switch(config-if)# end
```

ポートをデフォルトの設定に戻す場合は、 **no switchport voice vlan** インターフェイス コンフィギュレーション コマンドを使用します。

例：着信データ フレームのプライオリティの設定

次に、Cisco IP Phone に接続しているポートを設定して、PC または接続しているデバイスから受信するフレームのプライオリティを変更しないようにする例を示します。

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport priority extend trust
Switch(config-if)# end
```

ポートをデフォルトの設定に戻す場合は、 **no switchport priority extend** インターフェイス コンフィギュレーション コマンドを使用します。

次の作業

音声 VLAN を設定した後は、次の設定を行うことができます。

- VLAN
- VLAN トランキング
- VTP
- プライベート VLAN

第 105 章

プライベート VLAN の設定

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

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

プライベート VLAN の前提条件

プライベート VLAN は、VTP 1、2、および 3 のトランスペアレント モードでサポートされます。プライベート VLAN は、VTP 3 のサーバー モードでもサポートされます。

プライベート VLAN をdeviceに設定するときに、ユニキャストルートとレイヤ 2 エントリとの間のシステムリソースのバランスを取るために、常にデフォルトの Switch Database Management（SDM）テンプレートを使用してください。別の SDM テンプレートが設定されている場合

は、**sdm prefer default** グローバル コンフィギュレーション コマンドを使用してデフォルトのテンプレートを設定します。

プライベート VLAN の制約事項

プライベート VLAN は、LAN Base イメージを実行しているスイッチではサポートされません。

（注） 一部の状況では、エラー メッセージが表示されずに設定が受け入れられますが、コマンドには効果がありません。

- プライベート VLAN が設定されているdevicesでは、フォールバック ブリッジングを設定しないでください。
- リモート SPAN（RSPAN）をプライベート VLAN のプライマリまたはセカンダリ VLAN として設定しないでください。
- 次のような機能が設定されているインターフェイスにプライベート VLAN ポートを設定しないでください。
  - ダイナミック アクセス ポート VLAN メンバーシップ
  - ダイナミック トランキング プロトコル（DTP）
  - IPv6 Security Group（SG）
  - ポート集約プロトコル（PAgP）
  - リンク集約制御プロトコル（LACP）
  - マルチキャスト VLAN レジストレーション（MVR）
  - 音声 VLAN
  - Web Cache Communication Protocol（WCCP）
- IEEE 802.1x ポートベース認証をプライベート VLAN ポートに設定できますが、802.1x とポート セキュリティ、音声 VLAN、またはポート単位のユーザー ACL は、プライベート VLAN ポートに設定できません。
- プライベートVLANホストまたは無差別ポートはSPAN宛先ポートにはできません。SPAN宛先ポートをプライベート VLAN ポートに設定した場合、ポートは非アクティブになります。
- プライマリ VLAN の無差別ポートでスタティック MAC アドレスを設定する場合は、すべての関連セカンダリ VLAN に同じスタティック アドレスを追加する必要はありません。同様に、セカンダリ VLAN のホスト ポートでスタティック MAC アドレスを設定する場合は、関連プライマリ VLAN に同じスタティック MAC アドレスを追加する必要はありま

せん。さらに、スタティック MAC アドレスをプライベート VLAN ポートから削除する際に、設定されている MAC アドレスのすべてのインスタンスをプライベート VLAN から削除する必要はありません。

（注） プライベート VLAN のセカンダリ VLAN で学習したダイナミック MAC アドレスは、関連プライマリ VLAN で複製されます。プライマリ VLAN からトラフィックが入力される場合でも、すべての MAC エントリはセカンダリ VLANで学習されます。MAC アドレスがプライマリ VLAN で動的に学習される場合は、関連セカンダリ VLAN では複製されません。

- レイヤ 3 VLAN インターフェイス（SVI）はプライマリ VLAN にだけ設定してください。

プライベート VLAN について

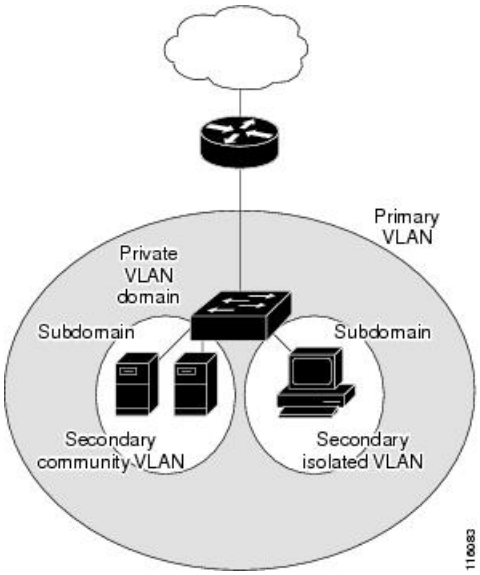
プライベート VLAN ドメイン

PVLAN 機能を使用すると、サービス プロバイダが VLAN を使用したときに直面する 2 つの問題に対処できます。

- IP ルーティングをイネーブルにするには、各 VLAN にサブネット アドレス空間またはアドレス ブロックを割り当てますが、これにより、未使用の IP アドレスが無駄になり、IP アドレスの管理に問題が起きます。

図 *136 :* プライベート *VLAN* ドメイン

プライベート VLAN の使用でスケーラビリティの問題に対処でき、サービス プロバイダにとっては IP アドレス管理上の利得がもたらされ、カスタマーに対してはレイヤ 2 セキュリティを提供できます。プライベート VLAN では、通常の VLAN ドメインをサブドメインに分割します。サブドメインは、プライマリ VLAN とセカンダリ VLAN のペアで表されます。プライベート VLAN には複数の VLAN ペアを設定可能で、各サブドメインにつき 1 ペアになります。プライベート VLAN 内のすべての VLAN ペアは同じプライマリ VLAN を共有します。セカンダリ VLAN ID は、各サブドメインの区別に使用されます。



セカンダリ VLAN

セカンダリ VLAN には、次の 2 種類があります。

- 独立 VLAN：独立 VLAN 内のポートは、レイヤ 2 レベルでは相互に通信できません。
- コミュニティ VLAN：コミュニティ VLAN 内のポートは互いに通信できますが、レイヤ 2 レベルにある他のコミュニティ内のポートとは通信できません。

プライベート VLAN ポート

プライベート VLAN では、同じプライベート VLAN 内のポート間をレイヤ 2 で分離します。プライベート VLAN ポートは、次のいずれかの種類に属するアクセス ポートです。

- 無差別：無差別ポートは、プライベート VLAN に属し、プライマリ VLAN と関連しているセカンダリ VLAN に属するコミュニティ ポートや独立ホスト ポートなどの、すべてのインターフェイスと通信できます。
- 独立：独立ポートは、独立セカンダリ VLAN に属しているホスト ポートです。これは、無差別ポートを除く、同じプライベート VLAN 内の他のポートからレイヤ 2 で完全に分離されています。プライベート VLAN は、無差別ポートからのトラフィックを除き、独立ポート宛のトラフィックをすべてブロックします。独立ポートから受信されたトラフィックは、無差別ポートにだけ転送されます。
- コミュニティ：コミュニティ ポートは、1 つのコミュニティ セカンダリ VLAN に属しているホスト ポートです。コミュニティ ポートは、同一コミュニティ VLAN のその他のポート、および無差別ポートと通信します。これらのインターフェイスは、他のコミュニティの他のすべてのインターフェイスおよびプライベート VLAN 内の独立ポートとレイヤ 2 で分離されます。

（注） トランク ポートは、通常の VLAN からのトラフィックを伝送し、またプライマリ、独立、およびコミュニティ VLAN からのトラフィックも伝送します。

プライマリおよびセカンダリ VLAN には次のような特性があります。

- プライマリ VLAN：プライベート VLAN には、プライマリ VLAN を 1 つだけ設定できます。プライベート VLAN 内のすべてのポートは、プライマリ VLAN のメンバーです。プライマリ VLAN は、無差別ポートからの単一方向トラフィックのダウンストリームを、（独立およびコミュニティ）ホスト ポートおよび他の無差別ポートへ伝送します。
- 独立 VLAN：プライベート VLAN の独立 VLAN は 1 つだけです。独立 VLAN はセカンダリ VLAN であり、ホストから無差別ポートおよびゲートウェイに向かう単一方向トラフィック アップストリームを搬送します。
- コミュニティ VLAN：コミュニティ VLAN は、アップストリーム トラフィックをコミュニティ ポートから無差別ポート ゲートウェイおよび同じコミュニティ内の他のホスト ポートに伝送するセカンダリ VLAN です。複数のコミュニティ VLAN を 1 つのプライベート VLAN に設定できます。

無差別ポートは、1 つのプライマリ VLAN、1 つの独立 VLAN、複数のコミュニティ VLAN だけで動作できます。レイヤ 3 ゲートウェイは通常、無差別ポートを介してdeviceに接続されます。無差別ポートでは、広範囲なデバイスをプライベート VLAN のアクセス ポイントとして接続できます。たとえば、すべてのプライベート VLAN サーバーを管理ワークステーションから監視したりバックアップしたりするのに、無差別ポートを使用できます。

ネットワーク内のプライベート VLAN

スイッチング環境では、個々のエンド ステーションに、または共通グループのエンド ステーションに、個別のプライベート VLAN や、関連する IP サブネットを割り当てることができます。エンド ステーションはデフォルト ゲートウェイとの通信を行うだけで、プライベート VLAN の外部と通信することができます。

プライベートVLANを使用し、次の方法でエンドステーションへのアクセスを制御できます。

- エンドステーションに接続されているインターフェイスを選択して独立ポートとして設定し、レイヤ2の通信をしないようにします。たとえば、エンドステーションがサーバーの場合、この設定によりサーバー間のレイヤ 2 通信ができなくなります。
- デフォルト ゲートウェイおよび選択したエンド ステーション（バックアップ サーバーなど）に接続されているインターフェイスを無差別ポートとして設定し、すべてのエンドステーションがデフォルト ゲートウェイにアクセスできるようにします。

複数のデバイスにわたるようにプライベート VLAN を拡張するには、プライマリ VLAN、独立 VLAN、およびコミュニティ VLAN を、プライベート VLAN をサポートする他のデバイスにトランキングします。使用するプライベート VLAN 設定のセキュリティを確保して、プライベート VLAN として設定された VLAN が他の目的に使用されないようにするには、プライ

ベート VLAN ポートがないデバイスを含めて、すべての中間デバイスでプライベート VLAN を設定します。

プライベート VLAN での IP アドレッシング方式

各カスタマーに個別の VLAN を割り当てると、次のように IP アドレッシング方式が非効率的になります。

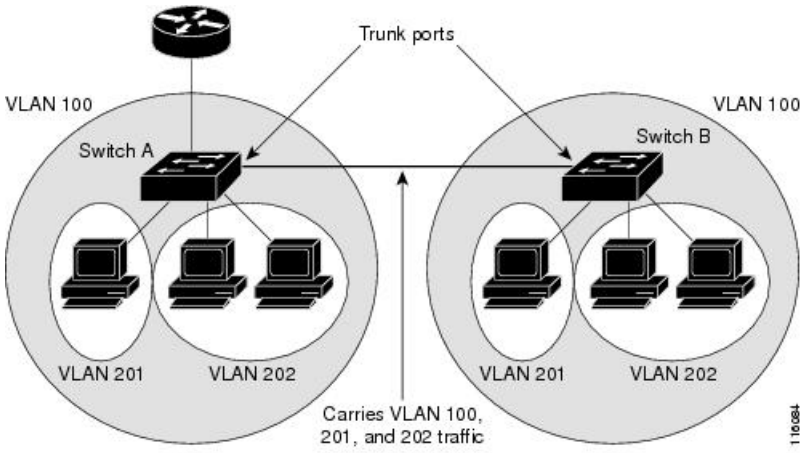
- カスタマー VLAN にアドレスのブロックを割り当てると、未使用 IP アドレスが発生することがあります。
- VLAN 内のデバイス数が増加した場合、それに対応するだけのアドレスを割り当てられない場合があります。

この問題は、プライベート VLAN を使用すると軽減します。プライベート VLAN では、プライベート VLAN のすべてのメンバーが、プライマリ VLAN に割り当てられている共通アドレス空間を共有するためです。ホストはセカンダリ VLAN に接続され、プライマリ VLAN に割り当てられているアドレスのブロックから IP アドレスが DHCP サーバーによってホストに割り当てられますが、同一プライマリ VLAN 内のセカンダリ VLAN には割り当てられません。さまざまなセカンダリ VLAN のカスタマー デバイスには後続 IP アドレスが割り当てられます。新しいデバイスを追加すると、サブネットアドレスの巨大プールから次に使用できるアドレスが、DHCP サーバーによって割り当てられます。

複数にまたがるプライベート VLAN Devices

図 *137 :* 複数のスイッチにまたがるプライベート *VLAN*

通常の VLAN と同じように、プライベート VLAN は複数の devices に広げることができます。トランク ポートはプライマリ VLAN およびセカンダリ VLAN をネイバー device に伝送します。トランク ポートはプライベート VLAN を他の VLAN として扱います。複数の devices に及ぶプライベート VLAN には、デバイス A の独立ポートからのトラフィックが、デバイス B の独立ポートに達しないという特徴があります



プライベート VLAN は、VTP 1、2、および 3 のトランスペアレント モードでサポートされます。プライベート VLAN は VTP 3 のサーバー モードでもサポートされます。VTP 3 を使用して設定したサーバー クライアントがある場合、サーバーに設定されているプライベート VLAN をクライアント上に反映させる必要があります。

プライベート VLAN の他機能との相互作用

プライベート VLAN とユニキャスト、ブロードキャスト、およびマルチキャスト トラフィック

通常の VLAN では、同じ VLAN にあるデバイスはレイヤ 2 レベルで互いに通信しますが、別の VLAN にあるインターフェイスに接続されたデバイスとはレイヤ 3 レベルで通信する必要があります。プライベート VLAN の場合、無差別ポートはプライマリ VLAN のメンバーであり、ホスト ポートはセカンダリ VLAN に属します。セカンダリ VLAN はプライマリ VLAN に対応付けられているため、これらの VLAN のメンバーはレイヤ 2 レベルで互いに通信できます。

通常の VLAN の場合、ブロードキャストはその VLAN のすべてのポートに転送されます。プライベート VLAN のブロードキャストの転送は、次のようにブロードキャストを送信するポートによって決まります。

- 独立ポートは、無差別ポートまたはトランク ポートだけにブロードキャストを送信します。
- コミュニティ ポートは、すべての無差別ポート、トランク ポート、同一コミュニティ VLAN のポートにブロードキャストを送信します。
- 無差別ポートは、プライベート VLAN のすべてのポート（その他の無差別ポート、トランク ポート、独立ポート、コミュニティ ポート）にブロードキャストを送信します。

マルチキャスト トラフィックのルーティングとブリッジングは、プライベート VLAN 境界を横断して行われ、単一コミュニティ VLAN 内でも行われます。マルチキャスト トラフィックは、同一独立 VLAN のポート間、または別々のセカンダリ VLAN のポート間で転送されません。

プライベート VLAN のマルチキャスト転送は次の状況をサポートします。

- 送信側が VLAN 外に存在する可能性があり、受信側が VLAN ドメイン内に存在している可能性がある。
- 送信側が VLAN 内に存在する可能性があり、受信側が VLAN ドメイン外に存在している可能性がある。
- 送信側と受信側が同一のコミュニティ VLAN に存在している可能性がある。

プライベート VLAN と SVI

レイヤ 3 device では、device 仮想インターフェイス（SVI）が VLAN のレイヤ 3 インターフェイスを表します。レイヤ 3 デバイスは、セカンダリ VLAN ではなく、プライマリ VLAN だけを介してプライベート VLAN と通信します。レイヤ 3 VLAN インターフェイス（SVI）はプライマリ VLAN にだけ設定してください。レイヤ 3 VLAN インターフェイスをセカンダリ VLAN 用に設定できません。VLAN がセカンダリ VLAN として設定されている間、セカンダリ VLAN の SVI はアクティブになりません。

- SVI がアクティブである VLAN をセカンダリ VLAN として設定する場合、SVI をディセーブルにしないと、この設定は許可されません。
- セカンダリ VLAN として設定されている VLAN に SVI を作成しようとしてセカンダリ VLAN がすでにレイヤ 3 にマッピングされている場合、SVI は作成されず、エラーが返されます。SVI がレイヤ 3 にマッピングされていない場合、SVI は作成されますが、自動的にシャットダウンされます。

プライマリ VLAN をセカンダリ VLAN と関連付けてマッピングすると、プライマリ VLAN の設定がセカンダリ VLAN の SVI に伝播されます。たとえば、プライマリ VLAN の SVI に IP サブネットを割り当てると、このサブネットは、プライベート VLAN 全体の IP サブネット アドレスになります。

プライベート VLAN 設定時の注意事項

セカンダリ VLAN およびプライマリ VLAN の設定

プライベート VLAN の設定時は、次の注意事項に従ってください。

- プライベート VLAN は、VTP 1、2、および 3 のトランスペアレント モードでサポートされます。device で VTP バージョン 1 または 2 が稼働している場合は、VTP をトランスペアレントモードに設定する必要があります。プライベート VLAN を設定した後で、VTP モードをクライアントまたはサーバーに変更できません。VTP バージョン 3 は、すべてのモードでプライベート VLAN をサポートします。

- VTP バージョン 1 または 2 でプライベート VLAN を設定した後、**copy running-config startup config** 特権 EXEC コマンドを使用して、VTP トランスペアレントモード設定とプライベート VLAN 設定をdeviceスタートアップ コンフィギュレーション ファイルに保存します。保存しないと、deviceをリセットした場合、デフォルトの VTP サーバーモードになり、プライベート VLAN をサポートしなくなります。VTP バージョン 3 ではプライベート VLAN をサポートします。
- VTP バージョン 1 および 2 では、プライベート VLAN 設定の伝播は行われません。プライベート VLAN ポートが必要なデバイスで VTP バージョン 3 が実行されていない場合は、VTP3 はプライベート VLAN を伝播するため、そのデバイス上でプライベート VLAN を設定する必要があります。
- VLAN 1 または VLAN 1002 ～ 1005 をプライマリ VLAN またはセカンダリ VLAN として設定できません。拡張 VLAN（VLAN ID 1006 ～ 4094）はプライベート VLAN に属することができます。
- プライマリ VLAN には、1 つの独立 VLAN および複数のコミュニティ VLAN を関連付けることができます。独立 VLAN またはコミュニティ VLAN には、1 つのプライマリ VLAN だけを関連付けることができます。
- プライベート VLAN には複数の VLAN が含まれますが、プライベート VLAN 全体で実行可能なスパニングツリー プロトコル（STP）インスタンスは 1 つだけです。セカンダリ VLAN がプライマリ VLAN に関連付けられている場合、プライマリ VLAN の STP パラメータがセカンダリ VLAN に伝播されます。
- TFTP サーバーから PVLAN 設定をコピーし、それを実行中の設定に適用しても、PVLAN の関連付けは形成されません。プライマリ VLAN がすべてのセカンダリ VLAN に確実に関連付けられていることを確認する必要があります。

  copy flash:config_file running-configの代わりに**configure replace flash:config_file force**を使用することもできます。
- DHCP スヌーピングはプライベート VLAN 上でイネーブルにできます。プライマリ VLAN で DHCP スヌーピングをイネーブルにすると、DHCP スヌーピングはセカンダリ VLAN に伝播されます。セカンダリ VLAN で DHCP を設定しても、プライマリ VLAN をすでに設定している場合、DHCP 設定は有効になりません。
- プライベート VLAN ポートで IP ソース ガードをイネーブルにする場合は、プライマリ VLAN で DHCP スヌーピングをイネーブルにする必要があります。
- プライベート VLAN でトラフィックを伝送しないデバイスのトランクから、プライベート VLAN をプルーニングすることを推奨します。
- プライマリ VLAN、独立 VLAN、およびコミュニティ VLAN には、別々の Quality of Service（QoS）設定を適用できます
- sticky ARP には、次の考慮事項があります。
  - sticky ARP エントリとは、SVI およびレイヤ 3 インターフェイス上で学習されるエントリです。これらのエントリは、期限切れになることはありません。

- **ip sticky-arp** グローバル コンフィギュレーション コマンドは、プライベート VLAN に属する SVI でだけサポートされます。
- **ip sticky-arp** インターフェイス コンフィギュレーション コマンドは、以下でのみサポートされます。
  - レイヤ 3 インターフェイス
  - 標準 VLAN に属する SVI
  - プライベート VLAN に属する SVI

  ip sticky-arp グローバルコンフィギュレーションおよび **ip sticky-arp interface** コンフィギュレーションコマンドの使用の詳細については、このリリースのコマンドリファレンスを参照してください。

- プライマリ VLAN およびセカンダリ VLAN で VLAN マップを設定できますただし、プライベート VLAN のプライマリおよびセカンダリ VLAN に同じ VLAN マップを設定することを推奨します。
- PVLAN は双方向です。これらは、入力側と出力側の両方に適用されます。

  レイヤ 2 のフレームがプライベート VLAN 内で転送されると、入力側と出力側で VLAN マップが適用されます。フレームがプライベート VLAN の内側から外部ポートにルーティングされる場合、プライベート VLAN マップが入力側に適用されます。同様に、フレームが外部ポートからプライベート VLAN にルーティングされると、プライベート VLAN は出力側に適用されます。

  ブリッジング

  - セカンダリ VLAN からプライマリ VLAN へのアップストリーム トラフィックの場合、セカンダリ VLAN の MAP は入力側に適用され、プライマリ VLAN の MAP は出力側に適用されます。
  - プライマリ VLAN からセカンダリ VLAN へのダウンストリーム トラフィックの場合は、プライマリ VLAN の MAP は入力方向で適用され、セカンダリ VLAN のMAP は出力方向で適用されます。

  ルーティング

  プライベート VLAN ドメインが 2 つ（PV1（sec1、prim1）および PV2（sec2、prim2））ある場合を想定します。PV1 から PV2 にルーティングされるフレームについては次のようになります。

  - sec1 の MAP および prim1 の L3 ACL は入力ポートに適用されます。
  - sec1 の MAP および prim2 の L3 ACL は出力ポートに適用されます。
  - 分離されたホスト ポートから無差別ポートへのアップストリームまたはダウンストリームに従うパケットの場合、分離された VLAN の VACL は入力方向に適用され、プライマリ VLAN の VACL は出力方向に適用されます。これにより、ユーザーは同

じプライマリ VLAN ドメインの別のセカンダリ VLAN に異なる VACL を設定することができます。

プライベート VLAN の特定 IP トラフィックをフィルタリングするには、プライマリ VLAN およびセカンダリ VLAN の両方に VLAN マップを適用する必要があります。

- プライマリ VLAN SVI にだけルータ ACL を適用できます。ACL はプライマリおよびセカンダリ VLAN のレイヤ 3 トラフィックに適用されます。
- プライベート VLAN がレイヤ 2 でホストを分離していても、ホストはレイヤ 3 で互いに通信できます。
- プライベート VLAN では、次のスイッチド ポート アナライザ（SPAN）機能がサポートされます。
  - プライベート VLAN を SPAN 送信元ポートとして設定できます。
  - プライマリ VLAN、独立 VLAN、およびコミュニティ VLAN 上で VLAN ベースの SPAN（VSPAN）を使用したり、単一の VLAN 上で SPAN を使用したりして、出力トラフィックまたは入力トラフィックを個別に監視することができます。

プライベート VLAN ポートの設定

プライベート VLAN ポートの設定時は、次の注意事項に従ってください。

- プライマリ VLAN、独立 VLAN、またはコミュニティ VLAN にポートを割り当てるには、プライベート VLAN コンフィギュレーション コマンドだけを使用します。プライマリ VLAN、独立 VLAN、またはコミュニティ VLAN として設定する VLAN に割り当てられているレイヤ 2 アクセス ポートは、この VLAN がプライベート VLAN の設定に含まれている場合、非アクティブです。レイヤ 2 トランク インターフェイスは STP フォワーディング ステートのままです。
- PAgP または LACP EtherChannel に属するポートを、プライベート VLAN ポートとして設定しないでください。ポートがプライベート VLAN の設定に含まれている間は、そのポートの EtherChannel 設定はいずれも非アクティブです。
- 設定ミスによる STP ループの発生を防ぎ、STP コンバージェンスを高速化するには、独立ホスト ポートおよびコミュニティ ホスト ポート上で PortFast および BPDU ガードをイネーブルにします。イネーブルの場合、STP はすべての PortFast が設定されたレイヤ 2 LAN ポートに BPDU ガード機能を適用します。PortFast および BPDU ガードを無差別ポートでイネーブルにしないでください。
- プライベート VLAN の設定で使用される VLAN を削除すると、この VLAN に関連付けられたプライベート VLAN ポートが非アクティブになります。
- ネットワーク デバイスをトランク接続し、プライマリ VLAN およびセカンダリ VLAN がトランクから削除されていない場合、プライベート VLAN ポートはさまざまなネットワーク デバイス上で使用できます。

プライベート VLAN の設定タスク

プライベート VLAN を設定するには、次の手順を実行します。

1. VTP モードをトランスペアレントに設定します。
2. プライマリおよびセカンダリ VLAN を作成してこれらを対応付けします。

（注） VLAN がまだ作成されていない場合、プライベート VLAN 設定プロセスでこれを作成します。

3. インターフェイスを独立ポートまたはコミュニティ ホスト ポートに設定して、ホスト ポートに VLAN メンバーシップを割り当てます。
4. インターフェイスを無差別ポートとして設定し、無差別ポートをプライマリおよびセカンダリ VLAN のペアにマッピングします。
5. VLAN 間ルーティングを使用する場合は、プライマリ SVI を設定し、セカンダリ VLAN をプライマリにマッピングします。
6. プライベート VLAN の設定を確認します。

プライベート VLAN の設定方法

プライベート VLAN 内の VLAN の設定および対応付け

VLAN コンフィギュレーション モードを終了するまで、**private-vlan** コマンドは有効ではありません。

プライベート VLAN 内で VLAN を設定し、関連付けるには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **vtp mode transparent**
4. **vlan** *vlan-id*
5. **private-vlan primary**
6. **exit**
7. **vlan** *vlan-id*
8. **private-vlan isolated**
9. **exit**
10. **vlan** *vlan-id*
11. **private-vlan community**

12. **exit**
13. **vlan** *vlan-id*
14. **private-vlan community**
15. **exit**
16. **vlan** *vlan-id*
17. **private-vlan association** [**add** | **remove**] *secondary_vlan_list*
18. **end**
19. **show vlan private-vlan** [**type**] or **show interfaces status**
20. **copy running-config startup config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**vtp mode transparent** 例： `スイッチ(config)# vtp mode transparent`	VTP モードをトランスペアレントに設定します（VTP をディセーブルにします）。 （注） VTP3 の場合、サーバーまたはトランスペアレント モードのいずれにもモードを設定できます。
ステップ 4	**vlan** *vlan-id* 例： `スイッチ(config)# vlan 20`	VLAN コンフィギュレーション モードを開始して、プライマリ VLAN となる VLAN を指定または作成します。VLAN ID の範囲は 2 ～ 1001 および 1006 ～ 4094 です。
ステップ 5	**private-vlan primary** 例： `スイッチ(config-vlan)# private-vlan primary`	VLAN をプライマリ VLAN として指定します。
ステップ 6	**exit** 例：	グローバル コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
	スイッチ(config-vlan)# **exit**	
ステップ 7	**vlan** *vlan-id* 例： スイッチ(config)# **vlan 501**	（任意）VLAN コンフィギュレーション モードを開始して、独立 VLAN となる VLAN を指定または作成します。VLAN ID の範囲は 2 ～ 1001 および 1006 ～ 4094 です。
ステップ 8	**private-vlan isolated** 例： スイッチ(config-vlan)# **private-vlan isolated**	VLAN を独立 VLAN として指定します。
ステップ 9	**exit** 例： スイッチ(config-vlan)# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 10	**vlan** *vlan-id* 例： スイッチ(config)# **vlan 502**	（任意）VLAN コンフィギュレーション モードを開始して、コミュニティ VLAN となる VLAN を指定または作成します。VLAN ID の範囲は 2 ～ 1001 および 1006 ～ 4094 です。
ステップ 11	**private-vlan community** 例： スイッチ(config-vlan)# **private-vlan community**	VLAN をコミュニティ VLAN として指定します。
ステップ 12	**exit** 例： スイッチ(config-vlan)# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 13	**vlan** *vlan-id* 例： スイッチ(config)# **vlan 503**	（任意）VLAN コンフィギュレーション モードを開始して、コミュニティ VLAN となる VLAN を指定または作成します。VLAN ID の範囲は 2 ～ 1001 および 1006 ～ 4094 です。

	コマンドまたはアクション	目的
ステップ 14	**private-vlan community** 例： `スイッチ(config-vlan)# private-vlan community`	VLAN をコミュニティ VLAN として指定します。
ステップ 15	**exit** 例： `スイッチ(config-vlan)# exit`	グローバル コンフィギュレーション モードに戻ります。
ステップ 16	**vlan** *vlan-id* 例： `スイッチ(config)# vlan 20`	ステップ 4 で指定したプライマリ VLAN に関して VLAN コンフィギュレーション モードを開始します。
ステップ 17	**private-vlan association** [**add** | **remove**] *secondary_vlan_list* 例： `スイッチ(config-vlan)# private-vlan association 501-503`	セカンダリ VLAN をプライマリ VLAN に関連付けます。単一のプライベート VLAN ID でも、またはハイフンで連結したプライベート VLAN ID でもかまいません。 • *secondary_vlan_list* パラメータには、スペースを含めないでください。カンマで区切った複数の項目を含めることができます。各項目として入力できるのは、単一のプライベート VLAN ID またはハイフンで連結したプライベート VLAN ID です。 • *secondary_vlan_list* パラメータには複数のコミュニティ VLAN ID を含められますが、独立 VLAN ID は 1 つだけです。 • *secondary_vlan_list* を入力するか、または *secondary_vlan_list* で **add** キーワードを指定し、セカンダリ VLAN とプライマリ VLAN を関連付けます。 • セカンダリ VLAN とプライマリ VLAN 間の関連付けをクリアするには、*secondary_vlan_list* に **remove** キーワードを使用します。 • このコマンドは、VLAN コンフィギュレーション モードを終了するまで機能しません。

	コマンドまたはアクション	目的
ステップ 18	**end** 例： スイッチ(config)# **end**	特権 EXEC モードに戻ります。
ステップ 19	**show vlan private-vlan** [**type**] or **show interfaces status** 例： スイッチ# **show vlan private-vlan**	設定を確認します。
ステップ 20	**copy running-config startup config** 例： スイッチ# **copy running-config startup-config**	device スタートアップ コンフィギュレーション ファイルに設定項目を保存します。

プライベート VLAN ホスト ポートとしてのレイヤ 2 インターフェイスの設定

レイヤ 2 インターフェイスをプライベート VLAN ホスト ポートとして設定し、これをプライマリおよびセカンダリ VLAN に関連付けるには、次の手順を実行します。

（注） 独立およびコミュニティ VLAN はいずれもセカンダリ VLAN です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode private-vlan host**
5. **switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id*
6. **end**
7. **show interfaces** [*interface-id*] **switchport**
8. **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `スイッチ> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `スイッチ# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/22`	設定するレイヤ 2 インターフェイスに対して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport mode private-vlan host** 例： `スイッチ(config-if)# switchport mode private-vlan host`	レイヤ 2 ポートをプライベート VLAN ホスト ポートとして設定します。
ステップ 5	**switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id* 例： `スイッチ(config-if)# switchport private-vlan host-association 20 501`	レイヤ 2 ポートをプライベート VLAN と関連付けます。 （注） これは、レイヤ 2 インターフェイスに PVLAN を関連付けるために必要な手順です。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show interfaces** [*interface-id*] **switchport** 例： `スイッチ# show interfaces gigabitethernet1/0/22 switchport`	設定を確認します。

	コマンドまたはアクション	目的
ステップ 8	**copy running-config startup-config** 例： スイッチ# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

プライベート VLAN 無差別ポートとしてのレイヤ 2 インターフェイスの設定

レイヤ 2 インターフェイスをプライベート VLAN 無差別ポートとして設定し、これをプライマリおよびセカンダリ VLAN にマッピングするには、次の手順を実行します。

（注） 独立およびコミュニティ VLAN はいずれもセカンダリ VLAN です。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode private-vlan promiscuous**
5. **switchport private-vlan mapping** *primary_vlan_id* {**add** | **remove**} *secondary_vlan_list*
6. **end**
7. **show interfaces** [*interface-id*] **switchport**
8. **copy running-config startup config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface** *interface-id* 例： `スイッチ(config)# interface gigabitethernet1/0/2`	設定するレイヤ 2 インターフェイスに対して、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**switchport mode private-vlan promiscuous** 例： `スイッチ(config-if)# switchport mode private-vlan promiscuous`	レイヤ 2 ポートをプライベート VLAN 無差別ポートとして設定します。
ステップ 5	**switchport private-vlan mapping** *primary_vlan_id* {**add** | **remove**} *secondary_vlan_list* 例： `スイッチ(config-if)# switchport private-vlan mapping 20 add 501-503`	プライベート VLAN 無差別ポートをプライマリ VLAN、および選択したセカンダリ VLAN にマッピングします。 • *secondary_vlan_list* パラメータには、スペースを含めないでください。カンマで区切った複数の項目を含めることができます。各項目として入力できるのは、単一のプライベート VLAN ID、またはハイフンで連結したプライベート VLAN ID の範囲です。 • セカンダリ VLAN とプライマリ VLAN をプライベート VLAN 無差別ポートにマッピングするには、*secondary_vlan_list*、または **add** キーワードを指定した *secondary_vlan_list* を使用します。 • セカンダリ VLAN とプライベート VLAN 無差別ポートのマッピングを解除するには、**remove** キーワードを指定した *secondary_vlan_list* を使用します。
ステップ 6	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 7	**show interfaces** [*interface-id*] **switchport** 例： `スイッチ# show interfaces gigabitethernet1/0/2 switchport`	設定を確認します。

	コマンドまたはアクション	目的
ステップ 8	**copy running-config startup config** 例： スイッチ# **copy running-config startup-config**	device スタートアップ コンフィギュレーション ファイルに設定項目を保存します。

セカンダリ VLAN のプライマリ VLAN レイヤ 3 VLAN インターフェイスへのマッピング

プライベート VLAN が VLAN 間ルーティングに使用される場合、SVI をプライマリ VLAN に設定してセカンダリ VLAN を SVI にマッピングできます。

（注） 独立およびコミュニティ VLAN はいずれもセカンダリ VLAN です。

セカンダリ VLAN をプライマリ VLAN の SVI にマッピングしてプライベート VLAN トラフィックのレイヤ 3 スイッチングを可能にするには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface vlan** *primary_vlan_id*
4. **private-vlan mapping** [**add** | **remove**] *secondary_vlan_list*
5. **end**
6. **show interface private-vlan mapping**
7. **copy running-config startup config**

手順の詳細

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

	コマンドまたはアクション	目的
ステップ 3	**interface vlan** *primary_vlan_id* 例： `スイッチ(config)# interface vlan 20`	プライマリ VLAN でインターフェイス コンフィギュレーション モードを開始して、VLAN を SVI として設定します。VLAN ID の範囲は 2 ～ 1001 および 1006 ～ 4094 です。
ステップ 4	**private-vlan mapping** [**add** | **remove**] *secondary_vlan_list* 例： `スイッチ(config-if)# private-vlan mapping 501-503`	セカンダリ VLAN をプライマリ VLAN のレイヤ 3 VLAN インターフェイスにマッピングして、プライベート VLAN 入力トラフィックのレイヤ 3 スイッチングを可能にします。 （注） **private-vlan mapping** インターフェイス コンフィギュレーション コマンドは、レイヤ 3 スイッチングされているプライベート VLAN トラフィックにだけ影響を与えます。 • *secondary_vlan_list* パラメータには、スペースを含めないでください。カンマで区切った複数の項目を含めることができます。各項目として入力できるのは、単一のプライベート VLAN ID またはハイフンで連結したプライベート VLAN ID です。 • *secondary_vlan_list* を入力するか、または **add** キーワードを指定した *secondary_vlan_list* を使用して、セカンダリ VLAN をプライマリ VLAN にマッピングします。 • **remove** キーワードを指定した *secondary_vlan_list* を使用して、セカンダリ VLAN とプライマリ VLAN のマッピングを解除します。
ステップ 5	**end** 例： `スイッチ(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show interface private-vlan mapping** 例： `スイッチ# show interfaces private-vlan mapping`	設定を確認します。

	コマンドまたはアクション	目的
ステップ 7	**copy running-config startup config** 例： スイッチ# **copy running-config startup-config**	device スタートアップ コンフィギュレーション ファイルに設定項目を保存します。

プライベート VLAN のモニター

次の表に、プライベート VLAN をモニターするために使用するコマンドを記載します。

表 *214* : プライベート *VLAN* モニタリング コマンド

コマンド	目的
show interfaces status	所属する VLAN を含む、インターフェイスのステータスを表示します。
show vlan private-vlan [**type**]	スイッチのプライベート VLAN 情報を表示します。
show interface switchport	インターフェイス上のプライベート VLAN 設定を表示します。
show interface private-vlan mapping	VLAN SVI のプライベート VLAN マッピングに関する情報を表示します。

プライベート VLAN の設定例

例：ホスト ポートとしてのインターフェイスの設定

次に、インターフェイスをプライベートVLANホスト ポートとして設定し、それをプライベート VLAN ペアに関連付けて、その設定を確認する例を示します。

```
スイッチ# configure terminal
スイッチ(config)# interface gigabitethernet1/0/22
スイッチ(config-if)# switchport mode private-vlan host
スイッチ(config-if)# switchport private-vlan host-association 20 501
スイッチ(config-if)# end
スイッチ# show interfaces gigabitethernet1/0/22 switchport
Name: Gi1/0/22
Switchport: Enabled
Administrative Mode: private-vlan host
Operational Mode: private-vlan host
```

```
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: 20 501
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan:
20 501

<output truncated>
```

例：プライベート VLAN 無差別ポートとしてのインターフェイスの設定

次の例では、インターフェイスをプライベート VLAN 無差別ポートとして設定し、それをプライベート VLAN にマッピングする方法を示します。インターフェイスは、プライマリ VLAN 20 のメンバで、セカンダリ VLAN 501 ～ 503 がマッピングされます。

```
スイッチ# configure terminal
スイッチ(config)# interface gigabitethernet1/0/2
スイッチ(config-if)# switchport mode private-vlan promiscous
スイッチ(config-if)# switchport private-vlan mapping 20 add 501-503
スイッチ(config-if)# end
```

show vlan private-vlan または **show interface status** 特権 EXEC コマンドを使用してプライマリおよびセカンダリ VLAN と スイッチ 上のプライベート VLAN ポートを表示します。

例：セカンダリ VLAN をプライマリ VLAN インターフェイスにマッピングする

次に、VLAN 501 および 502 のインターフェイスをプライマリ VLAN 10 にマッピングする例を示します。これにより、プライベート VLAN 501 および 502 からのセカンダリ VLAN 入力トラフィックのルーティングが可能になります。

```
スイッチ# configure terminal
スイッチ(config)# interface vlan 20
スイッチ(config-if)# private-vlan mapping 501-503
スイッチ(config-if)# end
スイッチ# show interfaces private-vlan mapping
Interface Secondary VLAN Type
--------- -------------- -----------------
vlan20    501            isolated
```

```
vlan20    502             community
vlan20    503             community
```

例：プライベート VLAN のモニタリング

次に、**show vlan private-vlan** コマンドの出力例を示します。

```
スイッチ# show vlan private-vlan
Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
20      501       isolated          Gi1/0/22, Gi1/0/2
20      502       community         Gi1/0/2
20      503       community         Gi1/0/2
```

次の作業

次の設定を行えます。

- VTP
- VLAN
- VLAN トランキング
- VLAN メンバーシップ ポリシー サーバー（VMPS）
- 音声 VLAN

その他の参考資料

関連資料

関連項目	マニュアル タイトル
CLI コマンド	LAN Switching コマンド リファレンス, Cisco IOS リリース

標準および RFC

標準/RFC	タイトル
RFC 1573	
RFC 1757	
RFC 2021	

MIB

MIB	MIB のリンク
本リリースでサポートするすべての MIB • BRIDGE-MIB（RFC1493） • CISCO-BRIDGE-EXT-MIB • CISCO-CDP-MIB • CISCO-PAGP-MIB • CISCO-PRIVATE-VLAN-MIB • CISCO-LAG-MIB • CISCO-L2L3-INTERFACE-CONFIG-MIB • CISCO-MAC-NOTIFICATION-MIB • CISCO-STP-EXTENSIONS-MIB • CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB • CISCO-VLAN-MEMBERSHIP-MIB • CISCO-VTP-MIB • IEEE8023-LAG-MIB • IF-MIB（RFC 1573） • RMON-MIB（RFC 1757） • RMON2-MIB（RFC 2021）	選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

シスコのテクニカル サポート

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

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。