Cisco IOS デバイスの強化ガイド [英語]

目次

RSA キーの SSHv2 拡張機能
コンソール ポートと AUX ポート
vty 回線と tty 回線の制御
vty 回線と tty 回線の転送制御
警告バナー
認証、認可、アカウンティング
TACACS+ 認証
認証フォールバック
Type 7 パスワードの使用
TACACS+ コマンド認可
TACACS+ コマンド アカウンティング
冗長 AAA サーバ
Simple Network Management Protocol の強化
SNMP コミュニティ ストリング
SNMP コミュニティ ストリングと ACL
インフラストラクチャ ACL
SNMP ビュー
SNMP バージョン 3
管理プレーン保護
ロギングのベスト プラクティス
ログの一元的な場所への送信
ログ レベル
コンソールまたはモニタ セッションへのログ送信の禁止
バッファ ロギングの使用
ロギングの発信元インターフェイスの設定
ロギングのタイムスタンプの設定
Cisco IOS ソフトウェアのコンフィギュレーション管理
コンフィギュレーションの置換とコンフィギュレーションのロールバック
コンフィギュレーション変更の排他的アクセス
Cisco IOS ソフトウェアのコンフィギュレーション回復
デジタル署名付き Cisco ソフトウェアの識別
コンフィギュレーション変更通知とロギング
コントロール プレーン
コントロール プレーン全般の強化
IP ICMP リダイレクト
ICMP 到達不能
プロキシ ARP
コントロール プレーン トラフィックの CPU への影響の制限
コントロール プレーン トラフィックについて
インフラストラクチャ ACL
受信 ACL
CoPP
コントロール プレーン保護
ハードウェア レート制限機能
BGP の保護

概要

この資料はネットワークの全面的なセキュリティを強化する Cisco IOS ® システム デバイスを保護するのを助けるように情報を記述したものです。 このドキュメントの構成はネットワーク デバイスの機能ごとに 3 つのプレーンに分かれていて、それぞれの機能の概要と関連ドキュメントへの参照を示します。

前提条件

要件

このドキュメントに関する固有の要件はありません。

使用するコンポーネント

このドキュメントは、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。 このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。 ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

背景説明

管理プレーン、コントロール プレーン、およびデータ プレーンというネットワークの 3 つの機能プレーンが持つ機能性はさまざまで、それぞれを保護する必要があります。

- **管理プレーン**：管理プレーンでは、Cisco IOS デバイスに送信されるトラフィックが管理されます。管理プレーンを構成するのは、アプリケーション、およびセキュア シェル（SSH）や Simple Network Management Protocol（SNMP）などのプロトコルです。

- **コントロール プレーン**：ネットワーク デバイスのコントロール プレーンでは、ネットワーク インフラストラクチャの機能性の維持に重要なトラフィックが処理されます。 コントロール プレーンを構成するのは、ネットワーク デバイス間のアプリケーションおよびプロトコルです。Border Gateway Protocol（BGP; ボーダー ゲートウェイ プロトコル）、Enhanced Interior Gateway Routing Protocol（EIGRP）や Open Shortest Path First（OSPF）などの Interior Gateway Protocol（IGP）が、これに含まれます。

- **データ プレーン**：データ プレーンでは、データがネットワーク デバイス経由で転送されます。 ローカルの Cisco IOS デバイスに送信されるトラフィックは、データ プレーンには含まれません。

このドキュメントで扱うセキュリティ機能に関しては、多くの場合、その機能を設定するために十分な情報を提供しています。 しかし、このドキュメントだけでは不十分な場合、それ以上の注意が必要かどうかを判断できるように説明しています。 このドキュメントには、実装すればネットワークの保護に役立つ推奨事項が必要に応じて記載されています。

セキュアな運用

セキュアなネットワーク動作は、重要な課題です。 このドキュメントの大半は、Cisco IOS デバイスの安全なコンフィギュレーションについて説明していますが、ネットワークを完全に保護するためにはコンフィギュレーションのみでは不十分です。 基本となるデバイスのコンフィギュレーション同様に、ネットワークで使用される操作手順も、セキュリティにとって大きな役割を果たします。

下記のトピックに含まれる操作上の推奨事項を実装することを推奨いたします。 下記のトピックでは、ネットワーク動作の重要領域に個別に焦点を当てていますが、すべてを網羅しているわけではありません。

Cisco セキュリティ アドバイザリおよびレスポンスの監視

Cisco Product Security Incident Response Team（PSIRT）は、Cisco 製品のセキュリティ関連問題に関して、PSIRT アドバイザリと呼ばれる通知を作成し、維持しています。 あまり重大ではない問題の通知には、Cisco Security Response が使用されます。 セキュリティ アドバイザリとレスポンスは、http://www.cisco.com/go/psirt から入手できます。

通知方法についての詳細は、『Cisco セキュリティ脆弱性ポリシー』を参照してください。

セキュアなネットワークを維持するために、リリース済みの Cisco セキュリティ アドバイザリおよびレスポンスに注意する必要があります。 ネットワークを危険にさらしかねない脅威を評価できるように、脆弱性に関して知っておく必要があります。 脆弱性の評価プロセスについては、『セキュリティ脆弱性のリスク トリアージに関するアナウンス』を参照してください。

認証、認可、アカウンティングの活用

ネットワーク デバイスをセキュリティで保護するには認証、認可、およびアカウンティング（AAA）フレームワークが重要です。 AAA フレームワークでは、管理セッションの認証が行われると同時に、特定の管理者定義コマンドに対してユーザが制限され、すべてのユーザが入力したすべてのコマンドが記録されます。 AAA の利用については、このドキュメントの「認証、認可、アカウンティングの使用」の項を参照してください。

ログ収集とモニタリングの一元化

セキュリティ事象に関連する既存、新出、過去のイベントを理解するために、イベント ロギングと関連付けを行うための統一的な戦略を持つ必要があります。 この戦略では、すべてのネットワーク デバイスからのロギングを活用し、事前パッケージングされカスタマイズ可能な相関機能を使用する必要があります。

ロギングの一元化を実装した後は、ログの分析と事象のトラッキングを行うための構造的なアプローチを開発する必要があります。 組織のニーズに応じて、ログ データを入念に見直すというシンプルなものから、高度なルールベースの分析までさまざまな方法をとることができます。

Cisco IOS ネットワーク デバイスにロギングを実装する方法についての詳細は、このドキュメントの「ロギングのベスト プラクティス」の項を参照してください。

セキュアなプロトコルの使用（可能な場合）

ネットワーク管理に関する機密データの伝送には、多くのプロトコルが使用されます。 可能な場合は、常にセキュアなプロトコルを使用する必要があります。 セキュアなプロトコルを選択するというのは、Telnet の代わりに SSH を使用して、認証データと管理情報の両方を暗号化することが含まれます。 さらに、コンフィギュレーション データをコピーする場合は、セキュアなファイル転送プロトコルを使用する必要があります。 たとえば、FTP や TFTP の代わりに、Secure Copy Protocol（SCP）を使用します。

Cisco IOS デバイスの安全な管理についての詳細は、このドキュメントの「インタラクティブ管理セッションの保護」の項を参照してください。

NetFlow によるトラフィック情報の取得

NetFlow をイネーブルにすると、ネットワークのトラフィック フローを監視できます。 NetFlow の本来の目的は、ネットワーク管理アプリケーションにトラフィック情報をエクスポートすることですが、ルータ上のフロー情報の表示にも使用できます。 この機能によって、ネットワークをどのようなトラフィックが通過しているかをリアルタイムで表示できます。 フロー情報がリモート コレクタにエクスポートされているかどうかにかかわらず、NetFlow を必要に応じてリアクティブに使用できるようにネットワーク デバイスを設定するように推奨いたします。

この機能についての詳細は、このドキュメントの「トラフィックの識別とトレースバック」の項および http://www.cisco.com/go/netflow（登録ユーザ専用）を参照してください。

コンフィギュレーション管理

コンフィギュレーション管理は、コンフィギュレーションの変更を提案、検討、承認、および展開するプロセスです。 Cisco IOS デバイスのコンフィギュレーション管理に関しては、 コンフィギュレーション アーカイブおよびセキュリティという 2 つの側面が重要です。

コンフィギュレーション アーカイブを使用すると、ネットワーク デバイスの変更を元に戻すことができます。 セキュリティに関しても、コンフィギュレーション アーカイブを使用して、セキュリティの変更点やその時期を特定できます。 この情報を AAA のログ データと組み合わせて使用すると、ネットワーク デバイスのセキュリティ監査に役立ちます。

Cisco IOS デバイスのコンフィギュレーションには、詳細な機密情報が多数含まれます。 たとえば、ユーザ名、パスワード、アクセス コントロール リストの内容が、この種の情報に相当します。 Cisco IOS デバイス コンフィギュレーションのアーカイブに使用するリポジトリをセキュリティで保護する必要があります。 この情報へのアクセスがセキュリティで保護されていない場合、ネットワーク全体のセキュリティが損なわれる可能性があります。

管理プレーン

管理プレーンは、ネットワークの管理目標を実現する機能で構成されます。 SSH を使用するインタラクティブ管理セッションや、SNMP または NetFlow による統計情報収集がこれに含まれます。 ネットワーク デバイスのセキュリティを検討する場合、管理プレーンを保護することが不可欠です。 セキュリティ上の事象によって管理プレーンの機能が弱体化した場合、ネットワークの

回復や安定化ができなくなる可能性があります。

このセクションでは、管理プレーンの強化に役立つ Cisco IOS ソフトウェアのセキュリティ機能とコンフィギュレーションについて、詳しく説明します。

管理プレーン全般の強化

管理プレーンは、デバイスのアクセス、コンフィギュレーション、および管理や、デバイス動作の監視とデバイスが展開されているネットワークの監視に使用されます。 管理プレーンは、このような機能の動作によるトラフィックを送受信するプレーンです。 管理プレーンの動作にはコントロール プレーンの動作が直接影響するので、デバイスの管理プレーンとコントロール プレーンの両方を保護する必要があります。 次に、管理プレーンで使用されるプロトコルを示します。

- Simple Network Management Protocol
- Telnet
- Secure Shell Protocol（SSH）
- File Transfer Protocol（FTP）
- Trivial File Transfer Protocol
- Secure Copy Protocol（SCP）
- TACACS+
- RADIUS
- NetFlow
- ネットワーク タイム プロトコル
- Syslog

セキュリティ障害の発生時に管理プレーンとコントロール プレーンに影響が及ばないように、手段を講じる必要があります。 どちらかのプレーンが悪用されれば、すべてのプレーンのセキュリティが侵害される可能性があります。

パスワード管理

パスワードにより、リソースやデバイスへのアクセスが制御されます。 これは、要求を認証するために使用されるパスワードまたはシークレットを定義することで実現されます。 リソースまたはデバイスへのアクセス要求が受信されると、その要求に対してパスワードと ID の検証が行われ、その結果でアクセスが許可、拒否、または制限されます。 セキュリティのベスト プラクティスとして、パスワードの管理には TACACS+ または RADIUS 認証サーバを使用する必要があります。 しかし、TACACS+ または RADIUS サービスに障害が発生した場合に備えて、特権アクセス用にローカル設定されたパスワードが依然として必要です。 また、デバイスのコンフィギュレーション内には、NTP キー、SNMP コミュニティ ストリング、ルーティング プロトコル キーなど、他のパスワード情報が存在することもあります。

enable secret コマンドを使用すると、Cisco IOS システムへの特権管理アクセスを許可するパスワードを設定できます。 古い **enable password コマンド**ではなく、**enable secret** を使用してください。 **enable password** コマンドには、脆弱な暗号化アルゴリズムが使用されています。

enable secret が設定されていない場合にコンソール tty 回線用のパスワードを設定すると、リモートのバーチャル ターミナル（vty）セッションからでも、コンソール パスワードを使用して特権アクセスを取得できます。 しかしこれは望ましくないことであり、これも enable secret を設定する理由の一つです。

service password-encryption グローバル コンフィギュレーション コマンドは、Cisco IOS ソフトウェアに対して、パスワード、CHAP（Challenge Handshake Authentication Protocol）シークレット、およびコンフィギュレーション ファイルに保存されている同様のデータを暗号化するように指示します。 このような暗号化を使用すれば、たとえばユーザが何気なく管理者の肩越しに画面を見てパスワードを読み取るといった事態を防止できます。 ただし、**service password-encryption コマンド**で使用されるアルゴリズムは、単純な Vigenere 暗号です。 このアルゴリズムは、ある程度高度な知識を持つ攻撃者による本格的な分析からコンフィギュレーション ファイルを保護する設計にはなっていないため、このような目的では使用しないでください。 暗号化されたパスワードを含む Cisco IOS コンフィギュレーション ファイルは、同じパスワードがクリアテキストでのリストになっている場合と同様に注意深く取り扱う必要があります。

この脆弱な暗号化アルゴリズムは、**enable secret** コマンドでは使用されていませんが、**enable password** グローバル コンフィギュレーション コマンドや **password** ライン コンフィギュレーション コマンドでは使用されています。 この種類のパスワードは使用せず、**enable secret** コマンドか、拡張パスワード セキュリティ機能を使用してください。

enable secret コマンドと拡張パスワード セキュリティ機能では、パスワードのハッシングに Message Digest 5（MD5）が使用されています。 このアルゴリズムは十分に公開審査がなされたもので、解読不可能とされています。 ただし、このアルゴリズムも辞書攻撃の対象にはなります。 辞書攻撃とは、攻撃者が辞書やパスワードの候補を記したリストに掲載されているすべての単語を順に試して一致を調べる手法です。 したがって、コンフィギュレーション ファイルは安全な場所に保管し、信頼できる相手とだけ共有するようにしてください。

拡張パスワード セキュリティ

拡張パスワード セキュリティ機能は Cisco IOS ソフトウェア リリース 12.2(8)T で導入されました。この機能を使用すると、**username** コマンドでパスワードに MD5 ハッシングを適用できます。 以前は、Type 0 と Type 7 という 2 種類のパスワードが使用されていました。 Type 0 はクリアテキスト パスワードであり、Type 7 では Vigenere 暗号のアルゴリズムが使用されます。 拡張パスワード セキュリティ機能は、取得にクリアテキスト パスワードが必要なプロトコル（CHAP など）では使用しないでください。

ユーザ パスワードを MD5 ハッシングで暗号化するには、**username secret** グローバル コンフィギュレーション コマンドを発行します。

```
!

username <name> secret <password>

!
```

この機能に関する詳細は、『拡張パスワード セキュリティ』を参照してください。

ログイン パスワード リトライ ロックアウト

ログイン パスワード リトライ ロックアウト機能は Cisco IOS ソフトウェア リリース 12.3(14)T で導入されました。この機能を使用すると、指定した回数だけログインに失敗したローカル ユーザ アカウントをロックアウトできます。 ロックアウトされたユーザのアカウントは、解除されるまでロックアウト状態になります。 特権レベル 15 に設定された認可ユーザを、この機能でロックアウトすることはできません。 特権レベル 15 を持つユーザの数は、最小限にとどめる必要があります。

認可ユーザは、ログインの失敗が既定回数に達した場合に、自身をデバイスからロックアウトできます。 また、悪意のあるユーザが、有効なユーザ名を使用して何度も認証を試行することで、サービス拒絶（DoS）状態を作成する可能性があります。

次の例では、ログイン パスワード リトライ ロックアウト機能をイネーブルにする方法を示しています。

```
!

aaa new-model
aaa local authentication attempts max-fail <max-attempts>
aaa authentication login default local

!

username <name> secret <password>

!
```

この機能は、CHAP やパスワード認証プロトコル（PAP）などの認証方式にも適用できます。

パスワード回復のディセーブル化

Cisco IOS ソフトウェア リリース 12.3(14)T 以降では、パスワード回復のディセーブル化機能を使用すると、コンソールにアクセスした任意のユーザが、安全ではない状態でデバイスのコンフィギュレーションにアクセスしてパスワードを消去することはできなくなります。 また、悪意のあるユーザがコンフィギュレーション レジスタの値を変更したり、NVRAM にアクセスしたりすることもできなくなります。

```
!

no service password-recovery

!
```

Cisco IOS ソフトウェアにはパスワード回復手順が備わっていますが、この手順を実行するには、システム起動時に Break キーを押して ROM モニタ モード（ROMmon）に入る必要があります。 ROMmon モードでは、デバイス ソフトウェアがリロードされ、新しいパスワードを含む新しいシステム コンフィギュレーションにするためのプロンプトを表示できます。

現在のパスワード回復手順では、コンソールにアクセスできる任意のユーザが、デバイスとそのネットワークにアクセスできます。 パスワード回復のディセーブル化機能により、システム起動時に Break キー シーケンスが中断され ROMmon に入ることができなくなります。

デバイスに対して **no service password-recovery** をイネーブルにする場合は、そのデバイス コンフィギュレーションのオフライン コピーを保存すること、およびコンフィギュレーション アーカイブ ソリューションを実装することを推奨いたします。 この機能をイネーブルにした後で Cisco IOS デバイスのパスワードを回復する必要がある場合は、コンフィギュレーション全体が削除されます。

この機能についての詳細は、『ROMmon セキュリティの設定例』を参照してください。

使用していないサービスの無効化

セキュリティ上のベスト プラクティスとして、不要なサービスはすべてディセーブルにする必要があります。 これらの要らないサービスは、User Datagram Protocol （UDP; ユーザ データグラム プロトコル）を使用する特にそれらパケットフィルタリングによって他では防がれるその他の攻撃および DoS を開始するために、まれに合法的な目的で利用されますが、使用することができます。

TCP および UDP のスモール サービスはディセーブルにする必要があります。 提供されるサービスには次のものがあります。

- echo（ポート番号 7）
- discard（ポート番号 9）
- daytime（ポート番号 13）
- chargen（ポート番号 19）

スモール サービスが悪用されるケースの大部分は、アンチスプーフィング アクセス リストによって回避できるか、または危険性を緩和できますが、ネットワークでアクセス可能な任意のデバイスでは、スモール サービスをディセーブルにする必要があります。 スモール サービスは、Cisco IOS ソフトウェア リリース 12.0 以降ではデフォルトで無効になっています。 それより前のソフトウェアでは、**no service tcp-small-servers** と **no service udp-small-servers** のグローバル コンフィギュレーション コマンドを発行してディセーブルにできます。

次のサービスは、使用しない場合はディセーブルにしてください。

- Finger サービス：ディセーブルにするには、**no ip finger** グローバル コンフィギュレーション コマンドを発行します。 この機能は、Cisco IOS ソフトウェア リリース 12.1(5) および 12.1(5)T 以降ではデフォルトでディセーブルになっています。
- ブートストラップ プロトコル（BOOTP）：ディセーブルにするには、**no ip bootp server** グローバル コンフィギュレーション コマンドを発行します。
- Cisco IOS ソフトウェア リリース 12.2(8)T 以降で BOOTP をディセーブルにするには、グローバル コンフィギュレーション モードで **ip dhcp bootp ignore** コマンドを発行します。 これを実行しても、Dynamic Host Configuration Protocol（DHCP）サービスは引き続きイネーブルのままです。
- DHCP サービス（DHCP リレー サービスが不要な場合）： ディセーブルにするには、グローバル コンフィギュレーション モードで **no service dhcp** コマンドを発行します。
- Maintenance Operation Protocol（MOP; メンテナンス オペレーション プロトコル）サービス：ディセーブルにするには、インターフェイス コンフィギュレーション モードで **no mop enabled** コマンドを発行します。
- ドメイン ネーム システム（DNS）サービス：ディセーブルにするには、**no ip domain-lookup**

グローバル コンフィギュレーション コマンドを発行します。

- パケット アセンブラ/ディスアセンブラ（PA）サービス（X.25 ネットワークで使用）：ディセーブルにするには、グローバル コンフィギュレーション モードで **no service pad** コマンドを発行します。

- HTTP サーバおよびセキュア HTTP（HTTPS）サーバ：HTTP サーバをディセーブルにするには、グローバル コンフィギュレーション モードで **no ip http server** コマンドを発行します。HTTPS サーバをディセーブルにするには、**no ip http secure-server** グローバル コンフィギュレーション コマンドを発行します。

- Cisco IOS デバイスが起動時にネットワークからコンフィギュレーションを取得する場合を除いて、**no service config** グローバル コンフィギュレーション コマンドを使用してください。 これにより、Cisco IOS デバイスでは、TFTP を使用してネットワーク上のコンフィギュレーション ファイルの場所が探索されなくなります。

- Cisco Discovery Protocol（CDP）は、他の CDP 対応デバイスのネイバールータとの隣接関係やネットワーク トポロジを検出するためのネットワーク プロトコルです。 CDP は、Network Management System（NMS; ネットワーク管理システム）やトラブルシューティングで使用できます。 非信頼ネットワークに接続しているすべてのインターフェイスで、CDP をディセーブルにする必要があります。 これは、**no cdp enable** インターフェイス コマンドで実行できます。 また、**no cdp run** グローバル コンフィギュレーション コマンドを使用する方法でも CDP をディセーブルにできます。 悪意のあるユーザが偵察やネットワーク マッピングを行うために、CDP が使用される可能性があることに注意してください。

- Link Layer Discovery Protocol（LLDP）は、802.1AB で定義された IEEE プロトコルです。LLDP は CDP と似ています。 ただし、LLDP では、CDP に対応していないデバイス間の相互運用が可能になります。 LLDP は CDP と同じ方法で扱う必要があります。非信頼ネットワークに接続しているすべてのインターフェイスでは、LLDP をディセーブルにしてください。 これを行うには、**no lldp transmit** および **no lldp receive** インターフェイス コンフィギュレーション コマンドを発行します。 LLDP をグローバルでディセーブルにするには、**no lldp run** グローバル コンフィギュレーション コマンドを発行します。 悪意のあるユーザが偵察やネットワーク マッピングを行うために、LLDP が使用される可能性があります。

EXEC タイムアウト

EXEC コマンド インタープリタがセッションを終了せずにユーザ入力を待機する時間を設定するには、**exec-timeout** ライン コンフィギュレーション コマンドを発行します。 アイドル状態の vty 回線または tty 回線のセッションをログアウトさせるには、**exec-timeout コマンド**を使用します。 デフォルトで、セッションは非アクティブの 10 分後に切断されています。

```
!

line con 0
exec-timeout <minutes> [seconds]
line vty 0 4
exec-timeout <minutes> [seconds]
!
```

TCP セッションのキープアライブ

service tcp-keepalive-in と **service tcp-keepalive-out** グローバル コンフィギュレーション コマンドを使用すると、デバイスから TCP セッションのための TCP キープアライブを送信できます。 デバイスへの着信接続やデバイスからの発信接続で TCP キープアライブをイネーブルにするには、この設定を使用する必要があります。 これにより、接続のリモート エンドにあるデバイスが引き続きアクセス可能なままで、ハーフオープンまたは孤立状態の接続がローカル Cisco IOS デバイスから削除されます。

```
!

service tcp-keepalives-in
service tcp-keepalives-out
!
```

管理インターフェイスの使用

デバイスの管理プレーンは、物理的または論理的管理インターフェイス上のインバンドまたはアウトオブバンドでアクセスできます。 ネットワークの停止中にも管理プレーンにアクセスできるように、インバンドとアウトオブバンド両方の管理アクセスが、ネットワーク デバイスごとに存在するのが理想的です。

デバイスへのインバンド アクセスに使用される最も一般的なインターフェイスの一つが、論理ループバック インターフェイスです。 ループバック インターフェイスは常にアップ状態ですが、物理インターフェイスの状態は変化することがあり、インターフェイスにアクセスできない可能性があります。 ループバック インターフェイスを管理インターフェイスとして各デバイスに追加して、管理プレーン専用にしておくことを推奨いたします。 これにより、管理者は管理プレーンでネットワーク全体にポリシーを適用できます。 デバイスに設定したループバック インターフェイスは、SSH、SNMP、syslog などの管理プレーン プロトコルによってトラフィックの送受信に使用されます。

```
!
interface Loopback0
 ip address 192.168.1.1 255.255.255.0
!
```

メモリしきい値通知

メモリしきい値通知機能は、Cisco IOS ソフトウェア リリース 12.3(4)T で導入されました。この機能を使用すると、デバイスのメモリ不足状況を緩和できます。 この機能では、メモリ不足緩和のために メモリしきい値通知とメモリ予約という 2 つの方式が使用されます。

デバイス上の空きメモリ量が、設定されたしきい値を下回ったことを通知する場合、メモリしきい値通知によって、ログ メッセージが生成されます。 次の設定例では、**memory free low-watermark** グローバル コンフィギュレーション コマンドでこの機能をイネーブルにする方法を示しています。 これにより、空きメモリ量がしきい値を下回ればデバイスで通知が生成され、しきい値を 5 % 上回ると再度通知が生成されます。

```
!

memory free low-watermark processor <threshold>
memory free low-watermark io <threshold>
!
```

メモリ予約を使用すると、重要な通知のために十分なメモリが確保されます。 次の設定例は、この機能をイネーブルにする方法を示しています。 これにより、デバイスのメモリが使い果たされていても、管理プロセスが機能し続けることができます。

```
!
```

```
memory reserve critical <value> !
```
この機能に関する詳細は、『メモリしきい値通知』を参照してください。

CPU しきい値通知

CPU しきい値通知機能は、Cisco IOS ソフトウェア リリース 12.3(4)T で導入されました。この機能を使用すると、デバイスの CPU 負荷が設定されたしきい値を超過すると、これが検出されて通知されるようにできます。 しきい値を超過した場合、デバイスでは SNMP トラップ メッセージが生成されて、送信されます。 Cisco IOS ソフトウェアでは、上昇しきい値および下降しきい値という 2 つの CPU 利用率しきい値方式がサポートされています。

次の設定例は、上昇しきい値および下降しきい値をイネーブルにして CPU しきい値通知メッセージを生成する方法を示しています。

```
!

snmp-server enable traps cpu threshold
!

snmp-server host <host-address> <community-string> cpu
!

process cpu threshold type <type> rising <percentage> interval <seconds>
[falling <percentage> interval <seconds>]
process cpu statistics limit entry-percentage <number> [size <seconds>]
!
```
この機能に関する詳細は、『CPU しきい値通知』を参照してください。

&

コンソール アクセス用のメモリ予約

Cisco IOS ソフトウェア リリース 12.4(15)T 以降では、コンソール アクセス用のメモリ予約機能を使用すると、管理やトラブルシューティングの目的で Cisco IOS デバイスにコンソールからアクセスできる十分な量のメモリを予約できます。 この機能は、デバイスがメモリ不足の状態で動作している場合に特に便利です。 この機能をイネーブルにするには、**memory reserve console** グローバル コンフィギュレーション コマンドを発行します。 次の設定例では、Cisco IOS デバイスでこの用途に 4096 KB を予約しています。

```
!
memory reserve console 4096
!
```
この機能に関する詳細は、『コンソール アクセス用のメモリ予約』を参照してください。

メモリ リーク検出

メモリ リーク検出機能は、Cisco IOS ソフトウェア リリース 12.3(8)T1 で導入されました。この機能を使用すると、デバイスのメモリ リークを検出できます。 メモリ リーク検出は、すべてのメモリ プール、パケット バッファ、およびメモリ チャンクでリークを検出できます。 メモリ リークとは、メモリが静的または動的に割り当てられたまま有効に利用されていないことです。 この機能では、動的なメモリ割り当てに焦点を絞って検出します。 メモリ リークが存在するかどうかを検出するには、**show memory debug leaks** EXEC コマンドを使用できます。

バッファ オーバーフロー： レッドゾーン破損の検出と修正

Cisco IOS ソフトウェア リリース 12.3(7) T 以降では、デバイスでバッファ オーバーフロー： レッドゾーン破損の検出と修正機能をイネーブルにすることによって、メモリ ブロック オーバーフローを検出して修正し、動作を続行できます。

この機能をイネーブルにするには、次のグローバル コンフィギュレーション コマンドを使用します。 いったん設定すると、**show memory overflow** コマンドを使用して、バッファ オーバーフロー検出と修正の統計情報を表示できます。

```
!
exception memory ignore overflow io
exception memory ignore overflow processor
!
```

強化された Crashinfo ファイル回収

強化された Crashinfo ファイル回収機能では、古い crashinfo ファイルが自動的に削除されます。この機能は Cisco IOS ソフトウェア リリース 12.3(11)T で追加されました。この機能を使用すると、領域が解放され、デバイスがクラッシュしたときに crashinfo ファイルを新規作成できるようになります。 また、保存する crashinfo ファイルの数を設定することもできます。

```
!
exception crashinfo maximum files <number-of-files>
!
```

ネットワーク タイム プロトコル

Network Time Protocol（NTP; ネットワーク タイム プロトコル）は特に危険というわけではありませんが、不要なサービスはどれでも、攻撃を媒介する可能性があります。 NTP が使用されている場合は、信頼できるタイミング ソースを明示的に設定して、適切な認証を使用することが重要です。 攻撃の犯罪捜査に syslog を利用したり、VPN 接続のフェーズ 1 認証で証明書に依存する場合は、正確で信頼できる時間が必要です。

- **NTP のタイム ゾーン**：NTP を設定する場合、タイムスタンプが正確に関連付けられるように、タイム ゾーンを設定する必要があります。 国際的に展開されるネットワーク内のデバイスに対してタイム ゾーンを設定するには、通常、2 つの方法があります。 一つは、すべてのネットワーク デバイスを Coordinated Universal Time（UTC; 世界標準時）（以前の Greenwich Mean Time（GMT; グリニッジ標準時））に設定する方法です。 もう一つは、ネットワーク デバイスをローカルのタイム ゾーンに設定する方法です。 この機能についての詳細は、Cisco 製品ドキュメントの『clock timezone』を参照してください。

- **NTP の認証**：NTP の認証を設定すると、信頼できる NTP ピア間で確実に NTP メッセージを交換できます。

NTP 認証を使用した設定の例：

クライアント：

```
(config)#ntp authenticate
(config)#ntp authentication-key 5 md5 ciscotime
(config)#ntp trusted-key 5
(config)#ntp server 172.16.1.5 key 5
```

Server:

```
(config)#ntp authenticate
(config)#ntp authentication-key 5 md5 ciscotime
(config)#ntp trusted-key 5
```

ディセーブル スマートなインストール

Cisco スマートなインストール（SMI）機能のまわりのセキュリティ上の推奨事項は機能が特定の顧客 の 環境で使用されるかによって決まります。 Cisco はこれらのユース ケースを区別します:

- スマートなインストール機能を使用しない顧客。
- ゼロ タッチ配備のためのだけスマートなインストール機能を利用している顧客。
- ゼロ タッチ配備より多くのためのスマートなインストール機能を利用している顧客（設定およびイメージ 管理）。

これらのセクションは各シナリオを詳しく解説しています:

- スマートなインストール機能を使用しない顧客。
- Cisco スマートなインストール機能を使用しない顧客は、コマンドが利用できる Cisco IOS および Cisco IOS XE ソフトウェアのリリースを実行するために、**vstack** コマンドでスマートなインストール機能をディセーブルにし。

  注: vstack コマンドは Cisco IOS Release 12.2(55)SE03 で導入されました。

これはスマートなインストール クライアント機能がディセーブルの状態で Cisco Catalyst スイッチの**提示 vstack** コマンドからの出力例あります:

```
switch# show vstack
config Role: Client (SmartInstall disabled)
Vstack Director IP address: 0.0.0.0
```

ゼロ タッチ配備のためのだけスマートなインストール機能を利用している顧客

ゼロ タッチ インストールが完了するディセーブルにして下さいまたは **vstack** コマンドを使用しないで下さい後スマートなインストール クライアントの機能性を。

vstack コマンドをネットワークに伝搬させないために、これらのメソッドの 1 つを使用して下さい:

- すべてのクライアントの **vstack** コマンドを切り替えます手動でまたはスクリプトと入力しないで下さい。
- ゼロ タッチ インストールの一部として各々のスマートなインストール クライアントに押される Cisco IOSコンフィギュレーションの一部として **vstack** コマンドを追加しないで下さい。
- リリースでは **vstack** コマンド（Cisco IOS Release 12.2(55)SE02 および それ 以前 リリース）をサポートしない、クライアントの Access Control List （ACL）を TCPポート 4786 のトラフィックをブロックするために切り替えます適用して下さい。

スマートなインストール クライアントの機能性 以降を有効に するために、すべてのクライアントの **vstack** コマンドを切り替えます手動でまたはスクリプトと入力して下さい。

ゼロ タッチ配備より多くのためのスマートなインストール機能を利用している顧客

スマートなインストール アーキテクチャの設計では、注意はインフラストラクチャ IPアドレス空間が信頼できないパーティにとってアクセスが不可能であることそのような物奪取 する必要があ

ります。 リリースではスマートなインストール ディレクターだけポート 4786 のすべてのスマートなインストール クライアントへの TCP 接続があることを **vstack** コマンドをサポートしない、確認して下さい。

管理者は Cisco 影響を受けたデバイスのスマートなインストール配備のためにこれらのセキュリティ上の推奨事項を使用できます:

- インターフェイス ACL
- コントロール プレーン ポリシング（CoPP）。 この機能はすべての Cisco IOS ソフトウェア リリースで利用できません。

この例は 10.10.10.1 としてスマートなインストール ディレクター IP アドレスおよび 10.10.10.200 としてスマートなインストール クライアントIPアドレスのインターフェイス ACL を示したものです:

```
ip access-list extended SMI_HARDENING_LIST
Permit tcp host 10.10.10.1 host 10.10.10.200 eq 4786
deny tcp any any eq 4786
permit ip any any
```

この ACL はすべてのクライアントのすべての IP インターフェイスで展開する必要があります。 それはまたディレクターでスイッチが最初に配備されるとき押すことができます。

更にインフラストラクチャ内のすべてのクライアントにアクセスを制限するために、管理者はネットワークでその他のデバイスのこれらのセキュリティ上の推奨事項を使用できます:

- インフラストラクチャ アクセスコントロール アクセス・コントロール・リスト（iACLs）
- VLAN アクセスコントロール アクセス・コントロール・リスト（VACL）

インフラストラクチャ ACL によるネットワーク アクセス制限

ネットワーク デバイスとの不正な直接通信の防止を目的として考案されたインフラストラクチャ アクセス コントロール リスト（iACL）は、ネットワークに実装できる最も重要なセキュリティ制御機能の一つです。 インフラストラクチャ ACL では、ほぼすべてのネットワーク トラフィックはネットワークそのものを宛先とはしないで、単にネットワークを通過するだけであるという考えを有効に活用しています。

iACL を設定して適用するには、ホストまたはネットワークからネットワーク デバイスへのどの接続を許可する必要があるかを指定します。 このような接続の一般的な例として、eBGP、SSH、SNMP などがあります。 必要な接続が許可された後、そのインフラストラクチャへの他のすべてのトラフィックは明示的に拒否されます。 ネットワークを横断するが、そのインフラストラクチャ デバイスを宛先としていないすべての通過トラフィックは、明示的に許可されます。

iACL による保護は、管理プレーンとコントロール プレーンの両方に関係しています。 iACL の実装は、ネットワーク インフラストラクチャ デバイス固有のアドレス指定を使用することで容易になります。 IP アドレッシングによるセキュリティへの影響についての詳細は、『IP アドレッシングに対するセキュリティ志向アプローチ』を参照してください。

次の iACL 設定例では、iACL 実装プロセスを開始する際のスタート地点として使用する必要がある構造を示しています。

```
!

ip access-list extended ACL-INFRASTRUCTURE-IN
!
```

```
!--- Permit required connections for routing protocols and
!--- network management
!

permit tcp host <trusted-ebgp-peer> host <local-ebgp-address> eq 179
permit tcp host <trusted-ebgp-peer> eq 179 host <local-ebgp-address>
permit tcp host <trusted-management-stations> any eq 22
permit udp host <trusted-netmgmt-servers> any eq 161
!
!--- Deny all other IP traffic to any network device
!

deny ip any <infrastructure-address-space> <mask>
!
!--- Permit transit traffic
!

permit ip any any
!
```

作成した iACL は、非インフラストラクチャ デバイスと接続するすべてのインターフェイスに適用する必要があります。 これには、他の組織、リモート アクセス セグメント、ユーザ セグメント、データセンター内のセグメントなどと接続するインターフェイスが含まれます。

インフラストラクチャ ACL についての詳細は、『 コアの保護： インフラストラクチャ保護 ACL』を参照してください。

ICMP パケット フィルタリング

Internet Control Message Protocol (ICMP; インターネット制御メッセージ プロトコル) は、IP コントロール プロトコルとしての設計になっています。 このため、ICPM で伝送されるメッセージは一般に、TCP プロトコルや IP プロトコルに対して広範囲に影響を及ぼす可能性があります。 ネットワーク トラブルシューティング ツールの **ping** や **traceroute** では ICMP を使用しますが、ネットワークが正常に動作している場合、外部 ICMP 接続が必要になることはほとんどありません。

Cisco IOS ソフトウェアには、ICMP メッセージを名前または種類およびコードで詳細にフィルタリングする機能があります。 次の例の ACL は、これまでの例のアクセス コントロール エントリ (ACE) と組み合わせて使用する必要があります。これにより、信頼できる管理ステーションと NMS サーバからの ping が許可され、その他の ICMP パケットはすべてブロックされます。

```
!

ip access-list extended ACL-INFRASTRUCTURE-IN
!
!--- Permit ICMP Echo (ping) from trusted management stations and servers
!

permit icmp host <trusted-management-stations> any echo
permit icmp host <trusted-netmgmt-servers> any echo
!
!--- Deny all other IP traffic to any network device
!

deny ip any <infrastructure-address-space> <mask>
!
!--- Permit transit traffic
!

permit ip any any
```

```
!
```

IP フラグメントのフィルタリング

フラグメント化された IP パケットのフィルタリ プロセスでは、セキュリティ デバイスにとって難しい問題があります。 これは、TCP パケットと UDP パケットのフィルタリングに使用されるレイヤ 4 情報が、先頭フラグメントにしか存在しないからです。 Cisco IOS ソフトウェアでは、特定の方式を使用して、設定されたアクセス リストと先頭以外のフラグメントを照合します。 Cisco IOS ソフトウェアでは、ACL に対してこのような先頭以外のフラグメントを評価し、レイヤ 4 フィルタリング情報を無視します。 これにより、設定された ACE のレイヤ 3 の部分でのみ、先頭以外のフラグメントを評価することになります。

次の設定例では、**192.168.1.1 のポート 22** 宛の TCP パケットが転送中にフラグメント化された場合、先頭フラグメントは、パケット内のレイヤ 4 情報に基づいて 2 番目の ACE によって期待どおりに廃棄されます。 ただし、残り (先頭以外) のフラグメントは、パケットのレイヤ 3 情報と ACE のみに基づいて最初の ACE によって許可されます。 次のシナリオはこの設定を示したものです。

```
!

ip access-list extended ACL-FRAGMENT-EXAMPLE
permit tcp any host 192.168.1.1 eq 80
deny tcp any host 192.168.1.1 eq 22
!>
```

フラグメント処理はわかりにくいため、ACL により IP フラグメントが誤って許可されることがあります。 また、侵入検知システムによる検出を逃れようとして、フラグメンテーションが使用されることもよくあります。 このような理由から、IP フラグメントは攻撃で使用されることが多く、設定された iACL の先頭で明示的にフィルタリングを適用する必要があります。 次の ACL の例には、あらゆる IP フラグメントのフィルタリングが含まれます。 この例の機能は、これまでの例の機能と組み合わせて使用する必要があります。

```
!

ip access-list extended ACL-INFRASTRUCTURE-IN
!
!--- Deny IP fragments using protocol-specific ACEs to aid in
!--- classification of attack traffic
!

deny tcp any any fragments
deny udp any any fragments
deny icmp any any fragments
deny ip any any fragments
!
!--- Deny all other IP traffic to any network device
!

deny ip any <infrastructure-address-space> <mask>
!
!--- Permit transit traffic
!

permit ip any any
!
```

フラグメント化された IP パケットの ACL による処理の詳細は、『アクセス コントロール リストと IP フラグメント』を参照してください。

ACL の IP オプション フィルタリング サポート

Cisco IOS ソフトウェア リリース 12.3(4)T では、ACL を使用して、パケットに含まれる IP オプションに基づいて IP パケットをフィルタリングする機能のサポートが追加されています。 IP オプションは例外パケットとして処理されるので、ネットワーク デバイスのセキュリティにとって難しい問題です。 これには、ネットワークを通過する通常のパケットには必要のないレベルの CPU 作業が必要です。 また、パケット内に IP オプションがあるということは、ネットワーク内のセキュリティ制御を無力化させようとしているか、パケットの転送特性を変えようとしていることを示しています。 このような理由から、IP オプションがついたパケットは、ネットワークのエッジでフィルタリングする必要があります。

IP オプションを含む IP パケットに対して完全なフィルタリングを行うには、次の例を前の例の ACE と組み合わせて使用する必要があります。

```
!

ip access-list extended ACL-INFRASTRUCTURE-IN
!
!--- Deny IP packets containing IP options
!

deny ip any any option any-options
!
!--- Deny all other IP traffic to any network device
!

deny ip any <infrastructure-address-space> <mask>
!
!--- Permit transit traffic
!

permit ip any any
!
```

ACL の TTL 値フィルタリング サポート

Cisco IOS ソフトウェア リリース 12.4(2)T では、ACL を使用して、存続可能時間（TTL）値に基づいて IP パケットをフィルタリングする機能のサポートが追加されました。 IP データグラムの TTL 値は、パケットが発信元から宛先へと移動する中で、ネットワーク デバイスを通過するごとに減少します。 初期値はオペレーティング システムによって異なりますが、パケットの TTL が 0 に達すると、そのパケットは廃棄されます。 TTL を 0 まで減らすことになったデバイスでは、パケットが廃棄され、ICMP Time Exceeded メッセージが生成されてパケットの発信元に送信されます。

このようなメッセージの生成と送信は、例外プロセスです。 期限が切れる IP パケットの数が少ない場合は、ルータでこの機能を実行できますが、期限が切れる IP パケットの数が多い場合、メッセージを生成して送信するために、空いているすべての CPU リソースが使用されます。 これは、DoS 攻撃の兆候を示しています。 このため、期限が切れる IP パケットの大量発生を利用する DoS 攻撃に対抗するため、デバイスのセキュリティを強化する必要があります。

TTL 値が低い IP パケットは、ネットワークのエッジでフィルタリングすることを推奨いたします。 ネットワークの通過するために十分な TTL 値がないパケットを完全にフィルタリングすることで、TTL ベースの攻撃の脅威を緩和できます。

次の ACL の例では、TTL 値が 6 未満のパケットがフィルタリングされます。 これにより、5 ホップまでのネットワークは TTL 期限切れ攻撃から保護されます。

```
!
```

```
ip access-list extended ACL-INFRASTRUCTURE-IN
!
!--- Deny IP packets with TTL values insufficient to traverse the network
!

deny ip any any ttl lt 6
!
!--- Deny all other IP traffic to any network device
!

deny ip any <infrastructure-address-space> <mask>
!
!--- Permit transit traffic
!

permit ip any any
!
```

> **注:** TTL 値が低いパケットを正当な目的で使用するプロトコルもあります。 そのようなプロトコルの一つが eBGP です。 TTL 期限切れに基づいた攻撃を緩和する方法の詳細は、『TTL 超過攻撃の識別と緩和』を参照してください。

この機能についての詳細は、『ACL の TTL 値フィルタリング サポート』を参照してください。

インタラクティブ管理セッションの保護

デバイスとの管理セッションでは、デバイスとその動作に関する情報の表示と収集ができます。この情報が悪意のあるユーザに公開されると、そのデバイスが攻撃対象となり、侵入されて、さらなる攻撃に利用される可能性があります。 デバイスへの特権アクセスを持つユーザは、そのデバイスに対して全面的な管理制御を行うことができます。 情報の開示と不正アクセスを防ぐには、管理セッションを保護することが不可欠です。

管理プレーン保護

Cisco IOS ソフトウェア リリース 12.4(6)T 以降では、管理プレーン保護 (MPP) 機能によって、デバイスが管理トラフィックを受信可能なインターフェイスを制限できます。 この機能により管理者は、デバイスとそのアクセス方法に対する制御を強化できます。

次の例は、MPP をイネーブルにして、GigabitEthernet0/1 インターフェイスで SSH と HTTPS のみを許可する方法を示しています。

```
!

control-plane host
management-interface GigabitEthernet 0/1 allow ssh https
!
```

MPP についての詳細は、『管理プレーン保護』を参照してください。

コントロール プレーン保護

コントロール プレーン保護 (CPPr) 機能は、コントロール プレーン ポリシング機能に基づいて構築され、IOS デバイスのルート プロセッサ宛のコントロール プレーン トラフィックの制限と規制が行われます。 CPPr は、Cisco IOS ソフトウェア リリース 12.4(4)T で追加されています。この機能によりコントロール プレーンは、サブインターフェイスと呼ばれる個別のコントロール プレーン カテゴリに分割されます。 コントロール プレーンのサブインターフェイスは、 Host、

Transit、および CEF-Exception の 3 つです。 さらに、CPPr には次のコントロール プレーン保護機能が追加されています。

- **ポートフィルタリング機能**：閉じているか受信状態ではない TCP ポートや UDP ポートに向かうパケットの規制や廃棄を行います。

- **キューしきい値ポリシー機能**：コントロール プレーンの IP 入力キューで許可されている指定されたプロトコルのパケット数を制限します。

CPPr により管理者は、ホスト サブインターフェイスを使用して管理目的でデバイスに送信されるトラフィックを分類、規制、および制限できます。 ホスト サブインターフェイス カテゴリに分類されるパケットの例として、SSH または Telnet などの管理トラフィックや、ルーティング プロトコルがあります。

> **注:** CPPr は IPv6 に対応しておらず、IPv4 入力パスに限定されています。

Cisco CPPr 機能についての詳細は、『コントロール プレーン保護機能ガイド - 12.4T』および『コントロール プレーン保護について』を参照してください。

管理セッションの暗号化

インタラクティブ管理セッションの実行中は情報が開示される可能性があるので、このトラフィックを暗号化して、悪意のあるユーザが送信中のデータにアクセスできないようにする必要があります。 トラフィックを暗号化することで、デバイスとのリモート アクセス接続が保護されます。 管理セッションのトラフィックがネットワーク上にクリアテキストで送信された場合、デバイスとネットワークに関する機密情報が攻撃者に取得される可能性があります。

管理者は、SSH や HTTPS（Secure Hypertext Transfer Protocol）の機能を使用して、デバイスとのリモート アクセス管理接続を暗号化して保護できます。 Cisco IOS ソフトウェアでサポートされるのは、SSH バージョン 1.0（SSHv1）、SSH Version 2.0（SSHv2）、および Secure Sockets Layer（SSL）と Transport Layer Security（TLS）を使用して認証やデータ暗号化を行う HTTPS です。 SSHv1 と SSHv2 には互換性がありません。 SSHv1 は不確か、標準化されない、従って SSHv2 がオプションである場合それは推奨されません。

また、Cisco IOS ソフトウェアでは Secure Copy Protocol（SCP）もサポートされています。これにより、デバイス コンフィギュレーションやソフトウェア イメージをコピーする際の接続が暗号化されて保護されます。 SCP では SSH が使用されています。 次の設定例では、Cisco IOS デバイスに対して SSH をイネーブルにしています。

```
!

ip domain-name example.com
!

crypto key generate rsa modulus 2048
!

ip ssh time-out 60
ip ssh authentication-retries 3
ip ssh source-interface GigabitEthernet 0/1
!

line vty 0 4
transport input ssh
```

```
!
```
次の設定例では、SCP サービスをイネーブルにしています。

```
!

ip scp server enable
!
```
次は、HTTPS サービスの設定例です。

```
!

crypto key generate rsa modulus 2048
!

ip http secure-server
!
```
Cisco IOS ソフトウェアの SSH 機能の詳細は、『Cisco IOS が稼働するルータとスイッチでの Secure Shell の設定』および『Secure Shell (SSH) に関する FAQ』を参照してください。

SSHv2

Cisco IOS ソフトウェア リリース 12.3(4) T で導入された SSHv2 サポート機能により、ユーザは SSHv2 を設定できます。 (これより前の Cisco IOS ソフトウェア リリースでは SSHv1 サポートが導入されていました。) SSH は、信頼性の高いトランスポート層の上部で実行され、強力な認証機能と暗号化機能を提供します。 SSH では、信頼できる転送として定義されているのは TCP のみです。 SSH で、ネットワーク上の他のコンピュータ またはデバイスに安全にアクセスしたり、コマンドを安全に実行できます。 SSH 経由でトンネリングされる Secure Copy Protocol (SCP) 機能により、ファイルを安全に転送できます。

IP ssh verson 2 コマンドが明示的に設定されない場合、Cisco IOS は SSH バージョン 1.99 を有効に します。 SSH バージョン 1.99 は SSHv1 および SSHv2 を両方接続可能にします。 SSHv1 は不確かであると考慮され、システムに対する悪影響をもたらす場合があります。 SSH が有効に なる場合、**IP ssh バージョン 2** コマンドの使用によって SSHv1 をディセーブルにすることを推奨します。

次の設定例では、Cisco IOS デバイスに対して SSHv2 をイネーブルにしています (SSHv1 はディセーブル) 。

```
!

hostname router

!

ip domain-name example.com

!

crypto key generate rsa modulus 2048

!

ip ssh time-out 60
ip ssh authentication-retries 3
ip ssh source-interface GigabitEthernet 0/1

!
```

```
ip ssh version 2

!

line vty 0 4
transport input ssh

!
```

SSHv2 の使用法の詳細については、『Secure Shell バージョン 2 のサポート』を参照してください。

RSA キーの SSHv2 拡張機能

Cisco IOS SSHv2 は、キーボードインタラクティブでパスワード ベースの認証方式をサポートしています。 RSA キーの SSHv2 拡張機能は、クライアントとサーバ向けの RSA ベースの公開キー認証もサポートしています。

ユーザ認証の場合、RSA ベースのユーザ認証では、各ユーザに関連付けられている秘密キー/公開キーのペアを認証に使用します。 ユーザは秘密キー/公開キーのペアをクライアントで生成し、公開キーを Cisco IOS SSH サーバで設定して、認証を完了します。

クレデンシャルの確立を試行する SSH ユーザは、秘密キーを使用して暗号化されたシグニチャを提示します。 シグニチャとユーザの公開キーは、認証のために SSH サーバに送信されます。 SSH サーバでは、ユーザから提示された公開キーに対してハッシュを計算します。 ハッシュは、サーバに一致するエントリがあるかどうかを判断するために使用されます。 一致が見つかった場合、RSA ベースのメッセージ検証が公開キーを使用して実行されます。 その結果、暗号化されたシグニチャに基づいて、ユーザのアクセスは認証されるか拒否されます。

サーバ認証の場合、Cisco IOS SSH クライアントが各サーバにホスト キーを割り当てる必要があります。 クライアントがサーバとの間で SSH セッションを確立しようとすると、キー交換メッセージの一部として、サーバのシグニチャを受信します。 厳密なホスト キーのチェック フラグがクライアント側でイネーブルの場合、そのサーバに対応するホスト キー エントリが事前に設定されているかどうかがクライアントで確認されます。 一致が見つかると、クライアントはサーバ ホスト キーを使用してシグニチャの検証を試行します。 サーバの認証が成功すると、セッションの確立が続行されます。 認証が成功しない場合は、セッションの確立が終了し、「**Server Authentication Failed** 」というメッセージが表示されます。

次の設定例では、Cisco IOS デバイスで、SSHv2 での RSA キーの使用が有効に設定されます。

```
!
! Configure a hostname for the device
!

hostname router
!
! Configure a domain name
!

ip domain-name cisco.com
!
! Specify the name of the RSA key pair (in this case, "sshkeys") to use for SSH
!

ip ssh rsa keypair-name sshkeys
!
! Enable the SSH server for local and remote authentication on the router using
```

```
! the "crypto key generate" command
! For SSH version 2, the modulus size must be at least 768 bits
!

crypto key generate rsa usage-keys label sshkeys modulus 2048
!
! Configure an ssh timeout (in seconds)
!
! The following enables a timeout of 120 seconds for SSH connections
!

ip ssh time-out 120
!
! Configure a limit of five (5) authentication retries
!

ip ssh authentication-retries 5
!
! Configure SSH version 2
!

ip ssh version 2
!
```

SSHv2 での RSA キーの使用法の詳細については、『セキュア シェル バージョン 2 の RSA キーに関する機能拡張』を参照してください。

次の設定例では、Cisco IOS SSH サーバが RSA ベースのユーザ認証を実行できるように設定されます。 サーバに保存されている RSA 公開キーが、クライアントに保存されている公開キーと秘密キーのペアを使用して検証されると、ユーザ認証は成功です。

```
!
! Configure a hostname for the device
!

hostname router
!
! Configure a domain name
!

ip domain-name cisco.com
!
! Generate RSA key pairs using a modulus of 2048 bits
!

crypto key generate rsa modulus 2048
!
! Configure SSH-RSA keys for user and server authentication on the SSH server
!

ip ssh pubkey-chain
!
! Configure the SSH username
!

username ssh-user
!
! Specify the RSA public key of the remote peer
!
! You must then configure either the key-string command
! (followed by the RSA public key of the remote peer) or the
```

```
! key-hash command (followed by the SSH key type and version.)
!
```

SSHv2 での RSA キーの使用法の詳細については、『RSA ベースのユーザ認証を実行するための Cisco IOS SSH サーバの設定』を参照してください。

次の設定例では、Cisco IOS SSH クライアントが RSA ベースのサーバ認証を実行できるように設定されます。

```
!
!

hostname router
!
ip domain-name cisco.c
!
! Generate RSA key pairs
!

crypto key generate rsa
!
! Configure SSH-RSA keys for user and server authentication on the SSH server
!

ip ssh pubkey-chain
!
! Enable the SSH server for public-key authentication on the router
!

server SSH-server-name
!
! Specify the RSA public-key of the remote peer
!
! You must then configure either the key-string command
! (followed by the RSA public key of the remote peer) or the
! key-hash <key-type> <key-name> command (followed by the SSH key
! type and version.)
!
! Ensure that server authentication takes place - The connection will be
! terminated on a failure
!

ip ssh stricthostkeycheck
!
```

SSHv2 での RSA キーの使用法の詳細については、『RSA ベースのサーバ認証を実行するための Cisco IOS SSH クライアントの設定』を参照してください。

コンソール ポートと AUX ポート

Cisco IOS デバイスのコンソール ポートと補助（AUX）ポートは、非同期回線であり、デバイスへのローカル アクセスとリモート アクセスに使用できます。 Cisco IOS デバイスのコンソール ポートには、特権があることに注意する必要があります。 特に、管理者はこのような特権を使用して、パスワード回復手順を実行できることには注意が必要です。 コンソール ポートにアクセスでき、デバイスへの電力供給を遮断するかデバイスをクラッシュさせることができれば、認証されていない攻撃者でもパスワードの回復を実行できます。

そのため、デバイスのコンソール ポートにアクセスするために使用されるあらゆる方法を、デバイスへの特権アクセスに対するセキュリティと同等に保護する必要があります。 アクセス保護に使用される方法としては、AAA、exec-timeout、およびコンソールにモデムが接続されている場

合はモデム パスワードがあります。

パスワード回復が不要な場合は、**no service password-recovery** グローバル コンフィギュレーション コマンドを使用してパスワード回復手順の実行機能を削除するという方法もあります。 ただし、**no service password-recovery コマンド**がイネーブルにされると、管理者はそのデバイスに対するパスワード回復を実行できなくなります。

多くの場合、デバイスの AUX ポートは、不正アクセスを防止するためにディセーブルにする必要があります。 AUX ポートをディセーブルにするには、次のコマンドを使用します。

```
!

line aux 0
transport input none
transport output none
no exec
exec-timeout 0 1
no password
!
```

vty 回線と tty 回線の制御

Cisco IOS ソフトウェアのインタラクティブ管理セッションでは、tty またはバーチャル tty（vty）を使用します。 tty は、デバイスとのローカル アクセス用の端末や、デバイスとのダイヤルアップ アクセス用のモデムと接続できるローカルの非同期回線です。 tty はその他のデバイスのコンソール ポートにも接続できます。 これにより、tty 回線に接続されたデバイスはコンソール サーバとして機能でき、この状態で、tty 回線に接続されたデバイスのコンソール ポートにネットワークを介して接続を確立できます。 ネットワークを経由したこのようなリバース接続の tty 回線も制御する必要があります。

vty 回線は、プロトコルにかかわらず（SSH、SCP、Telnet など）、デバイスによってサポートされるその他のすべてのリモート ネットワーク接続で使用されます。 ローカルまたはリモートの管理セッションを介してデバイスにアクセスできるように、vty 回線および tty 回線の両方を適切に制御する必要があります。 Cisco IOS デバイスの vty 回線の数は限られています; 利用可能な回線数は、show line EXEC コマンドで特定できます。 すべての vty 回線が使用されている場合、新しい管理セッションは確立できません。これにより、デバイスへのアクセスにとっての DoS 状態が発生します。

デバイスの vty または tty に対する最も単純な形式のアクセス コントロールは、ネットワーク内のデバイスの場所にかかわらず、すべての回線で認証を使用することです。 vty 回線にはネットワークを介してアクセスできるので、これは vty 回線にとって不可欠です。 デバイスへのリモート アクセスに使用されているモデムに接続されている tty 回線や、他のデバイスのコンソール ポートに接続されている tty 回線も、ネットワークを介してアクセスできます。 vty および tty のアクセス コントロールを行う他の方法としては、**transport input** または **access-class** コンフィギュレーション コマンドを使用する方法、CoPP 機能と CPPr 機能を使用する方法、またはデバイスのインターフェイスにアクセス リストを適用する方法があります。

AAA を使用することで認証を実行できます。デバイス アクセスの認証には、ローカル ユーザ データベースを使用するか、または vty 回線や tty 回線に直接設定された単純なパスワード認証を使用して AAA を適用することが推奨されています。

アイドル状態の vty 回線または tty 回線のセッションをログアウトさせるには、**exec-timeout コマンド**を使用します。 また、**service tcp-keepalives-in コマンド**を使用して、デバイスへの着信接続で TCP キープアライブをイネーブルにすることも必要です。 これにより、接続のリモート エンドにあるデバイスが引き続きアクセス可能で、ハーフオープンまたは孤立状態の接続がローカル

の IOS デバイスから削除されます。

vty 回線と tty 回線の転送制御

デバイスがコンソール サーバとして使用されている場合は、そのデバイスへの、またはそのデバイスを介した暗号化され安全なリモート アクセス管理接続のみを許可するように vty と tty を設定する必要があります。 このセクションでは tty の場合について説明します。tty 回線は他のデバイス上のコンソール ポートに接続でき、これによりネットワーク経由で tty へのアクセスが可能になるからです。 情報の開示や管理者とデバイスの間で送信されるデータへの不正アクセスを防止するために、Telnet や rlogin などのクリアテキスト プロトコルを使用する代わりに **transport input ssh** を使用します。 tty に対しては、**transport input none** コンフィギュレーションをイネーブルにできます。これにより、事実上リバース コンソール接続で tty 回線を使用できなくなります。

vty 回線と tty 回線はどちらも他のデバイスに接続できます。 発信接続に使用できるトランスポートの種類を制限するには、**transport output** ライン コンフィギュレーション コマンドを使用します。 発信接続が不要の場合は、**transport output none** を使用します。 ただし、発信接続を許可する場合は、**transport output ssh** を使用して、暗号化された安全なリモート アクセス方式で接続するようにします。

> **注:** IPSec がサポートされている場合は、デバイスとの暗号化された安全なリモート アクセス接続に IPSec を使用できます。 IPSec を使用する場合は、デバイスにさらに CPU オーバーヘッドが加わります。 ただし、IPSec を使用する場合でも引き続き SSH をトランスポートとして使用する必要があります。

警告バナー

一部の司法管轄地域では、システムの使用が許可されていないことが不正ユーザに通知されていなければ、それらのユーザを訴追することはできず、悪意のあるユーザの監視も不法行為とみなされる場合があります。 この通知を表示する方法の 1 つとして、Cisco IOS ソフトウェアの banner login コマンドで設定されるバナー メッセージにその通知を含ませる方法があります。

法的通知要件は複雑で、司法管轄地域や状況によっても異なるため、この問題はお客様の担当弁護士と相談する必要があります。 司法管轄地域内でも、複数の法的見解が存在する場合もあります。 弁護士とご相談の上、次の情報の一部または全部をバナーに含めることが考えられます。

- 特別に承認された人のみがシステムへのログインやシステムの使用を許可されていることを伝える通知と、だれが使用を承認できるのかを示す情報。
- システムの不正な使用は違法であり、民事罰および刑事罰が課される場合があることを伝える通知。
- システムのあらゆる使用が、これ以上の警告なしに記録または監視され、その結果得られたログが裁判所での証拠として使用される場合があることを伝える通知。
- 地域法によって規定されている特定の通知

法的観点というよりもセキュリティの観点から、ルータの名前、モデル、ソフトウェア、所有権についての具体的な情報はログイン バナーに含めないでください。 これらの情報は悪意のあるユーザに利用される可能性があります。

認証、認可、アカウンティング（AAA）

ネットワーク デバイスへのインタラクティブ アクセスをセキュリティ保護するには認証、認可、アカウンティング（AAA）フレームワークが重要です。 AAA フレームワークでは、ネットワークのニーズに基づいて詳細に設定できる環境が提供されます。

TACACS+ 認証

TACACS+ は、リモート AAA サーバに対して管理ユーザの認証を行う場合に Cisco IOS デバイスで使用できる認証プロトコルです。 このような管理ユーザは、SSH、HTTPS、Telnet、または HTTP を介して IOS デバイスにアクセスできます。

TACACS+ 認証、またはより一般的に AAA 認証では、各ネットワーク管理者が個々のユーザ アカウントを使用できます。 単一の共有パスワードに依存しない場合、ネットワークのセキュリティが向上すると同時に、アカウンタビリティも強化されます。

RADIUS は、その目的の点で TACACS+ と似たプロトコルです。 ただし、RADIUS ではネットワーク上で送信されるパスワードだけが暗号化されます。 一方、TACACS+ では、ユーザ名とパスワードを含む TCP ペイロード全体が暗号化されます。 このため、AAA サーバで TACACS+ がサポートされている場合は、RADIUS ではなく TACACS+ を使用してください。 これら 2 つのプロトコルの詳細な比較は、『TACACS+ と RADIUS の比較』を参照してください。

Cisco IOS デバイスに対して TACACS+ 認証をイネーブルにするには、次の例のように設定します。

```
!

aaa new-model
aaa authentication login default group tacacs+
!

tacacs-server host <ip-address-of-tacacs-server>
tacacs-server key <key>
!
```

この設定は、組織固有の AAA 認証テンプレートの出発点として使用できます。 AAA の設定についての詳細は、『認証、認可、アカウンティング』を参照してください。

方式リストとは、ユーザ認証のために照会される認証方式を記載したシーケンシャル リストです。 方式リストを使用すると、認証に使用するセキュリティ プロトコルを 1 つまたは複数指定できるため、最初の方式が失敗した場合に備えて認証のバックアップ システムを確保できます。 Cisco IOS ソフトウェアは、リストで、ユーザの認証また拒否が正常に実行する最初の方式を使用します。 それ以降の方式が試行されるのは、サーバが使用できなかったかまたは設定が誤っていたことが原因で、前の方式が失敗した場合だけです。

名前付き方式リストの設定の詳細については、『 認証のための名前付き方式リスト』を参照してください。

認証フォールバック

設定されたすべての TACACS+ サーバが利用不能になった場合、Cisco IOS デバイスはセカンダリ認証プロトコルを使用できます。 一般的には、設定されたすべての TACACS+ サーバが利用不能になった場合は、ローカル認証かイネーブル認証を使用するように設定します。

オンデバイス認証のオプションは、enable、local、および line です。 これらのオプションにはそれぞれの利点があります。 enable secret の使用が推奨されます。秘密鍵のハッシュには、回線認証やローカル認証で使用される Type 7 パスワードで使用される暗号化アルゴリズムよりも、本質的にさらに安全な一方向アルゴリズムが使用されるからです。

ただし、ローカル定義ユーザに対してシークレット パスワードの使用をサポートしている Cisco IOS ソフトウェア リリースでは、ローカル認証にフォールバックすることが推奨されます。 これにより、1 人以上のネットワーク管理者が、ローカル定義ユーザを作成できます。 TACACS+ が完全に利用不能になった場合、各管理者はローカルのユーザ名とパスワードを使用できます。 この措置によって TACACS+ 停止中のネットワーク管理者のアカウンタビリティが強化されますが、すべてのネットワーク デバイス上のローカル ユーザ アカウントを維持する必要があるので、管理上の負担は飛躍的に大きくなります。

次の設定例では、前の TACACS+ 認証例を踏まえて、**enable secret コマンド**でローカルに設定されたパスワードへのフォールバック認証が追加されています。

```
!

enable secret <password>
!

aaa new-model
aaa authentication login default group tacacs+ enable
!

tacacs-server host <ip-address-of-tacacs-server>
tacacs-server key <key>
!
```

AAA でフォールバック認証を使用する方法についての詳細は、『認証の設定』を参照してください。

Type 7 パスワードの使用

Type 7 パスワードは本来、保管されたパスワードを迅速に復号化する設計になっており、パスワードを保管するための安全な形式ではありません。 これらのパスワードを簡単に復号化できるツールは多数あります。 Type 7 パスワードを使用するのは、Cisco IOS デバイスで使用中の機能.に必要な場合だけにとどめて、それ以外では使用しないようにしてください。

この種類のパスワードをなくすには、AAA 認証や拡張パスワード セキュリティ機能を使用してください。これにより、**username** グローバル コンフィギュレーション コマンドでローカルに定義されたユーザがシークレット パスワードを使用できます。 Type 7 パスワードの使用を完全にはなくせない場合は、これらのパスワードを暗号化するのではなく難読化することを検討してください。

Type 7 パスワードの除去についての詳細は、このドキュメントの「管理プレーン全般の強化」の項を参照してください。

TACACS+ コマンド認可

TACACS+ と AAA によるコマンド認可は、管理ユーザによって入力された各コマンドを許可または拒否するメカニズムです。 ユーザが EXEC コマンドを入力すると、Cisco IOS によって各コマンドは設定された AAA サーバに送信されます。 次に、その AAA サーバでは、設定されたポリシーを使用して、その特定のユーザに対してコマンドを許可または拒否します。

前の例の AAA 認証に次の設定を追加すると、コマンド認可を実装できます。

```
!

aaa authorization exec default group tacacs none
aaa authorization commands 0 default group tacacs none
aaa authorization commands 1 default group tacacs none
aaa authorization commands 15 default group tacacs none
!
```

コマンド認可についての詳細は、『認可の設定』を参照してください。

TACACS+ コマンド アカウンティング

AAA コマンド アカウンティングを設定すると、入力された各 EXEC コマンドに関する情報が、設定された TACACS+ サーバに送信されます。 TACACS+ サーバに送信される情報には、実行されたコマンド、実行日、およびコマンドを入力したユーザ名が含まれます。 RADIUS ではコマンド アカウンティングはサポートされていません。

次の設定例では、特権レベル 0、1、および 15 で入力された EXEC コマンドに対して AAA コマンド アカウンティングがイネーブルになります。 この設定は、TACACS サーバの設定を含む前の例を基にしています。

```
!

aaa accounting exec default start-stop group tacacs
aaa accounting commands 0 default start-stop group tacacs
aaa accounting commands 1 default start-stop group tacacs
aaa accounting commands 15 default start-stop group tacacs
!
```

AAA アカウンティングの設定についての詳細は、『アカウンティングの設定』を参照してください。

冗長 AAA サーバ

環境内で活用される AAA サーバは、冗長化し、耐障害性の高い方法で展開する必要があります。こうすることで、1 台の AAA サーバが利用できなくなっても、SSH などのインタラクティブ管理アクセスが可能になります。

冗長 AAA サーバ ソリューションを設計または実装する場合は、次の点を考慮に入れてください。

- ネットワーク障害が発生した場合の AAA サーバのアベイラビリティ
- 地理的に分散した場所への AAA サーバの配置
- 定常状態と障害状態での個々の AAA サーバへの負荷
- ネットワーク アクセス サーバと AAA サーバの間のネットワーク遅延
- AAA サーバ データベースの同期

詳細は、『Access Control Server の展開』を参照してください。

Simple Network Management Protocol の強化

このセクションでは、IOS デバイス内の SNMP の展開の保護に使用できる複数の方法を説明しています。 ネットワーク データと、このデータを送信するネットワーク デバイスの両方の機密性、整合性、およびアベイラビリティを保護するには、SNMP を適切に保護することが重要です。 SNMP からは、ネットワーク デバイスの状態に関する豊富な情報が提供されます。 この情報が悪意のあるユーザによるネットワークへの攻撃に利用されないようにするため、この情報を保護する必要があります。

SNMP コミュニティ ストリング

コミュニティ ストリングは、IOS デバイス上の SNMP データへの読み取り専用アクセスと読み書きアクセスの両方を制限するためにデバイスに適用されるパスワードです。 このようなコミュニティ ストリングにはありふれた言葉を使用しないでください。パスワードと同様に、慎重に選択する必要があります。 コミュニティ ストリングは、定期的にネットワーク セキュリティのポリシーに合わせて変更する必要があります。 たとえば、ネットワーク管理者がロールを変更する場合や会社を退社するときにコミュニティ ストリングを変更する必要があります。

次の設定では、読み取り専用のコミュニティ ストリングを READONLY、読み書きのコミュニティ ストリングを READWRITE としています。

```
!

snmp-server community READONLY RO
snmp-server community READWRITE RW
!
```

> **注:** 上記のコミュニティ ストリングの例は、これらの文字列の使用法をわかりやすく説明するために選んだものです。 実稼働環境で使用するコミュニティ ストリングは、慎重に選択し、英数字と記号を取り混ぜたものにする必要があります。 ありふれた文字列ではないパスワードの選択に関する詳細は、『堅牢なパスワードを作成する上での推奨事項』を参照してください。

この機能についての詳細は、『IOS SNMP コマンド リファレンス』を参照してください。

SNMP コミュニティ ストリングと ACL

SNMP アクセスを特定の発信元 IP アドレスのグループに制限するには、コミュニティ ストリングに加えて、ACL を適用します。 次の設定では、SNMP 読み取り専用アクセスを 192.168.100.0/24 のアドレス レンジにあるエンド ホスト デバイスに制限し、SNMP 読み書きアクセスを 192.168.100.1 にあるエンド ホスト デバイスのみに制限しています。

> **注:** これらの ACL で許可されたデバイスが、要求された SNMP 情報にアクセスするには、適切なコミュニティ ストリングが必要です。

```
!

access-list 98 permit 192.168.100.0 0.0.0.255
access-list 99 permit 192.168.100.1
!

snmp-server community READONLY RO 98
snmp-server community READWRITE RW 99
```

```
!
```
この機能の詳細については、『Cisco IOS ネットワーク管理コマンド リファレンス』の「snmp-server community」を参照してください。

インフラストラクチャ ACL

インフラストラクチャ ACL（iACL）を展開すると、信頼できる IP アドレスを持つエンド ホストのみが、IOS デバイスに SNMP トラフィックを送信できるようになります。 iACL には、UDP ポート 161 で不正な SNMP パケットを拒否するポリシーが含まれている必要があります。

iACL の使用方法についての詳細は、このドキュメントの「インフラストラクチャ ACL によるネットワーク アクセス制限」セクションを参照してください。

SNMP ビュー

SNMP ビューは、特定の SNMP MIB へのアクセスを許可または拒否できるセキュリティ機能です。 ビューを作成して **snmp-server community** community-string view グローバル コンフィギュレーション コマンドでコミュニティ ストリングに適用すると、MIB データにアクセスする場合、そのビューに定義されたアクセス権に制限されます。 必要に応じて、SNMP のユーザを必要なデータに制限するためにビューを使用することを推奨いたします。

次の設定例では、コミュニティ ストリング LIMITED が設定された SNMP アクセスを、system グループ内にある MIB データに制限しています。

```
!

snmp-server view VIEW-SYSTEM-ONLY system include
!

snmp-server community LIMITED view VIEW-SYSTEM-ONLY RO
!
```
詳細は、『SNMP サポートの設定』を参照してください。

SNMP バージョン 3

SNMP バージョン 3（SNMPv3）は、RFC3410 、RFC3411、RFC3412、RFC3413、RFC3414、および RFC3415 で定義されており、相互運用可能な標準ベースのネットワーク管理用プロトコルです。 SNMPv3 では、ネットワーク上のパケットが認証され、オプションで暗号化されることから、デバイスへのアクセスが保護されます。 SNMPv3 がサポートされている場合、SNMP を展開する際のセキュリティがより一層強化されます。 SNMPv3 には、次の 3 つの主要設定オプションがあります。

- **no auth**：このモードでは、SNMP パケットの認証や暗号化は不要です。
- **auth**：このモードでは、SNMP パケットの認証は必要ですが、暗号化は不要です。
- **priv**：このモードでは、SNMP パケットの認証と暗号化（プライバシー）が必要です。

正規のエンジン ID は、SNMPv3 セキュリティ メカニズム（認証または認証および暗号化）を使用して SNMP パケットを処理するために存在している必要があります。 デフォルトでは、エンジン ID はローカルに生成されます。 エンジン ID を表示するには、次の例で示すように **show snmp engineID** コマンドを使用します。

```
router#show snmp engineID
Local SNMP engineID: 80000009030000152BD35496
Remote Engine ID IP-addr Port
```

> **注:** engineID が変更されると、すべての SNMP ユーザ アカウントを再設定する必要があります。

次に、SNMPv3 グループの設定を行います。 次のコマンドでは、SNMP サーバ グループ AUTHGROUP の SNMPv3 対応 Cisco IOS デバイスを設定していますが、**auth** キーワードの使用により認証のみがイネーブルにされています。

```
!
snmp-server group AUTHGROUP v3 auth
!
```

次のコマンドでは、SNMP サーバ グループ PRIVGROUP の SNMPv3 対応 Cisco IOS デバイスに **priv キーワード**を使用して、このグループでの認証と暗号化をイネーブルに設定しています。

```
!
snmp-server group PRIVGROUP v3 priv
!
```

次のコマンドでは、SNMPv3 ユーザ snmpv3user に、MD5 認証パスワード **authpassword** と 3DES 暗号化パスワード **privpassword** を設定しています。

```
!

snmp-server user snmpv3user PRIVGROUP v3 auth md5 authpassword priv 3des
privpassword
!
```

snmp-server user コンフィギュレーション コマンドは、RFC 3414 の規定に従って、デバイスのコンフィギュレーション出力には表示されません。 このため、ユーザ パスワードはコンフィギュレーションには表示されません。 設定されたユーザを表示するには、次の例で示すように、**show snmp user** コマンドを入力します。

```
router#show snmp user
User name: snmpv3user
Engine ID: 80000009030000152BD35496
storage-type: nonvolatile active
Authentication Protocol: MD5
Privacy Protocol: 3DES
Group-name: PRIVGROUP
```

この機能についての詳細は、『SNMP サポートの設定』を参照してください。

管理プレーン保護

Cisco IOS ソフトウェアの管理プレーン保護（MPP）機能を使用すると、デバイスで SNMP トラフィックを終端できるインターフェイスが制限されるため、SNMP を保護できます。 管理者は MPP 機能を使用して、1 つ以上のインターフェイスを管理インターフェイスとして指定できます。 管理トラフィックは、これらの管理インターフェイスを経由してのみデバイスに入ることが許可されます。 MPP をイネーブルにすると、指定された管理インターフェイス以外のインターフェイスでは、そのデバイス宛のネットワーク管理トラフィックは許可されません。

MPP は、CPPr 機能のサブセットであり、CPPr をサポートするバージョンの IOS を必要とします。 CPPr についての詳細は、『コントロール プレーン保護について』を参照してください。

次の例では、MPP を使用して SNMP と SSH アクセスを FastEthernet 0/0 インターフェイスのみ

に制限しています。

```
!
control-plane host
management-interface FastEthernet0/0 allow ssh snmp
!
```

詳細は、『管理プレーン保護機能ガイド』を参照してください。

ロギングのベスト プラクティス

イベント ロギングによって、Cisco IOS デバイスとデバイスが展開されているネットワークの動作状況を把握できます。 Cisco IOS ソフトウェアには、組織のネットワーク管理と可視性の目標実現に役立つ複数の柔軟なロギング オプションがあります。

以降のセクションでは、管理者がロギングをうまく活用しながら、Cisco IOS デバイスでのロギングによる影響を最小限に抑えるための基本的なロギングのベスト プラクティスをいくつか紹介しています。

ログの一元的な場所への送信

ロギング情報をリモート syslog サーバに送信することが推奨されます。 こうすることで、複数のネットワーク デバイスが関係するネットワーク イベントとセキュリティ イベントの関連付けや監査をより効果的に実行できるようになります。 syslog メッセージは UDP によってクリアテキストで送信され、信頼性は高くない点に注意してください。 このため、ネットワークで可能な管理トラフィックに対する保護（暗号化やアウトオブバンド アクセスなど）を拡張して syslog トラフィックが保護されるようにしてください。

次の設定例では、ロギング情報をリモート syslog サーバに送信するように Cisco IOS デバイスを設定しています。

```
!
logging host <ip-address>
!
```

ログの関連付けについての詳細は、『ファイアウォールと IOS ルータ syslog イベントを使用したインシデントの識別』を参照してください。

12.4(15) T で統合され、12.0(26)S で初めて導入されたローカル不揮発性ストレージ（ATA ディスク）へのロギング機能では、システム ロギング メッセージを Advanced Technology Attachment（ATA）フラッシュ ディスクに保存できます。 ATA ドライブに保存されたメッセージは、ルータが再起動した後も残ります。

次の設定行では、最大 134,217,728 バイト（128 MB）のロギング メッセージが ATA フラッシュ（disk0）の syslog ディレクトリに書き込まれるように設定され、16,384 バイトのファイル サイズが指定されます。

```
logging buffered
logging persistent url disk0:/syslog size 134217728 filesize 16384
```

ロギング メッセージを ATA ディスク上のファイルに書き込む前に、Cisco IOS ソフトウェアは、十分なディスク領域があるかどうかをチェックします。 十分なディスク スペースがない場合、ロギング メッセージの最も古いファイル（タイムスタンプによる）が削除され、現在のファイルが保存されます。 ファイル名の形式は、log_month: **day: year::** 送信されました。

注: ATA フラッシュ ドライブのディスク領域は限られているため、保存データの上書きを防

ぐためにディスク領域を維持する必要があります。

次に、ルータ ATA フラッシュディスクから、FTP サーバ（192.168.1.129）の外部ディスクにロギング メッセージをコピーする例を示します。

```
copy disk0:/syslog ftp://myuser/mypass@192.168.1.129/syslog
```

この機能の詳細については、『ローカル不揮発性ストレージ（ATA ディスク）へのロギング』を参照してください。

ログ レベル

Cisco IOS デバイスによって生成される各ログ メッセージには、レベル 0（Emergencies）からレベル 7（Debug）までの 8 つの重大度のいずれか 1 つが割り当てられます。 特に必要ない限り、レベル 7 でのロギングは行わないことをお勧めします。レベル 7 でロギングを行うと、デバイスの CPU 負荷が上昇し、その結果、デバイスとネットワークが不安定になることがあります。

どのロギング メッセージをリモート syslog サーバに送信するかを指定するには、グローバル コンフィギュレーション コマンド **logging trap level** を使用します。 level に指定する数値を下限とする重大度のメッセージが送信されます。 バッファにログを記録するには、**logging buffered** level コマンドを使用します。

次の設定例では、リモート syslog サーバとローカル ログ バッファに送信するログ メッセージを重大度 6（informational）から 0（emergencies）に制限しています。

```
!

logging trap 6
logging buffered 6
!
```

詳細は、『トラブルシューティング、障害管理、およびロギング』を参照してください。

コンソールまたはモニタ セッションへのログ送信の禁止

Cisco IOS ソフトウェアでは、ログ メッセージをモニタ セッションやコンソールに送信することが可能です。モニタ セッションは、EXEC コマンド **terminal monitor** が発行されたインタラクティブ管理セッションです。 ただし、これを行うと IOS デバイスの CPU 負荷が上昇することがあるので、推奨されません。 その代わりに、ロギング情報をローカル ログ バッファに送信することが推奨されます。バッファは **show logging** コマンドを使用して表示できます。

コンソールやモニタ セッションへのロギングをディセーブルにするには、グローバル コンフィギュレーション コマンドの **no logging console** と **no logging monitor** を使用します。 次の設定例は、これらのコマンドの使用法を示しています。

```
!

no logging console
no logging monitor
!
```

グローバル コンフィギュレーション コマンドの詳細については、『Cisco IOS ネットワーク管理コマンド リファレンス』を参照してください。

バッファ ロギングの使用

Cisco IOS ソフトウェアでは、生成されたログ メッセージをローカルに表示できるように、ローカル ログ バッファの使用がサポートされています。 コンソールやモニタ セッションにログを送信するのではなく、ログをバッファに記録することを強く推奨いたします。

バッファ ロギングの設定時に関連する 2 つの設定オプションとして、 ロギング バッファ サイズと、バッファに保存されるメッセージの重大度があります。 **ロギング バッファ**のサイズを設定するには、グローバル コンフィギュレーション コマンド **logging buffered** size を使用します。 バッファに記録する最低の重大度を設定するには、logging buffered severity コマンドを使用します。 管理者は **show logging** EXEC コマンドを使用して、ロギング バッファの内容を表示できます。

次の設定例では、ロギング バッファのサイズを 16384 バイト、重大度を 6（informational）に設定しています。これにより、重大度 0（emergencies）から 6（informational）までのメッセージが保管されます。

```
!

logging buffered 16384 6
!
```

バッファ ロギングについての詳細は、『Cisco IOS ネットワーク管理コマンド リファレンス』を参照してください。

ロギングの発信元インターフェイスの設定

ログ メッセージの収集と確認を行う際の一貫性を高めるため、ロギングの発信元インターフェイスを静的に設定することを推奨いたします。 **logging source-interface** インターフェイス コマンドを使用して、ロギングの発信元インターフェイスを静的に設定することで、個々の Cisco IOS デバイスから送信されるすべてのロギング メッセージには、それぞれ同じ IP アドレスが表示されるようになります。 さらに安定性を高めるために、ロギングの発信元としてループバック インターフェイスを使用することをお勧めします。

次の設定例では、**logging source-interface** interface グローバル コンフィギュレーション コマンドを使用して、すべてのログ メッセージでループバック 0 インターフェイスの IP アドレスが使用されるように指定しています。

```
!
logging source-interface Loopback 0
!
```

詳細は、『Cisco IOS コマンド リファレンス』を参照してください。

ロギングのタイムスタンプの設定

ロギングのタイムスタンプを設定すると、複数のネットワーク デバイスが関係するイベントの関連付けに役立ちます。 ロギング データを関連付けられるように、正確で一貫したロギング タイムスタンプが設定されていることが重要です。 ロギング タイムスタンプには、日付と時刻（ミリ秒単位）、およびデバイスが使用されているタイム ゾーンを設定します。

次の例では、ロギング タイムスタンプを Coordinated Universal Time（UTC; 世界標準時）ゾーンのミリ秒単位で設定しています。

```
!
service timestamps log datetime msec show-timezone
!
```

ログの時刻を UTC 以外のタイム ゾーンにする場合は、ローカルのタイム ゾーンを指定して、生成されたログ メッセージにその情報が表示されるように設定できます。 次の例では、Pacific Standard Time (PST; 太平洋標準時) にデバイスを設定しています。

```
!

clock timezone PST -8
service timestamps log datetime msec localtime show-timezone
!
```

Cisco IOS ソフトウェアのコンフィギュレーション管理

Cisco IOS ソフトウェアには、Cisco IOS デバイスでのコンフィギュレーション管理のフォームをイネーブルにできる複数の機能があります。 このような機能には、コンフィギュレーションのアーカイブ、前のコンフィギュレーション バージョンへのロールバック、詳細なコンフィギュレーション変更ログの作成などがあります。

コンフィギュレーションの置換とコンフィギュレーションのロールバック

Cisco IOS ソフトウェア リリース 12.3(7)T 以降では、コンフィギュレーションの置換機能とコンフィギュレーションのロールバック機能により、Cisco IOS デバイス コンフィギュレーションをデバイス上でアーカイブ管理できます。 このアーカイブに手動または自動で保存されたコンフィギュレーションは、**configure replace** filename コマンドを使用して、現在の実行コンフィギュレーションを置き換えることができます。 これは、**copy** *filename* **running-config** コマンドとは異なった働きです。 **copy** コマンドを実行するとマージが行われるのに対して、**configure replace** filename コマンドを使用すると、実行コンフィギュレーションが置き換えられます。

ネットワーク内のすべての Cisco IOS デバイスでこの機能をイネーブルにすることを推奨いたします。 イネーブルにすると、**archive config** 特権 EXEC コマンドを使用して、現在の実行コンフィギュレーションをアーカイブに追加できます。 アーカイブに追加されたコンフィギュレーションは、**show archive** EXEC コマンドを使用して表示できます。

次の例では、コンフィギュレーションを自動的にアーカイブに追加する設定を示しています。 次に、Cisco IOS デバイスに対しアーカイブ済みコンフィギュレーションを archived-config-N という名前のファイルとして disk0: ファイル システムに保存し、最大 14 個のバックアップを維持し、アーカイブを 1 日 1 回 (1440 分) 実行、および管理者が **write memory** EXEC コマンドを発行する場合にアーカイブを実行するように指示する例を示します。

```
!

archive
path disk0:archived-config
maximum 14
time-period 1440
write-memory
!
```

コンフィギュレーション アーカイブ機能では最大 14 個のバックアップ コンフィギュレーションを保存できますが、スペースの要件を考慮した上で **maximum** コマンドを使用することを推奨いたします。

コンフィギュレーション変更の排他的アクセス

Cisco IOS ソフトウェア リリース 12.3(14)T では、コンフィギュレーション変更の排他的アクセス機能が追加され、Cisco IOS デバイスのコンフィギュレーションを変更する管理者は常に一人

だけになります。 この機能により、関連するコンフィギュレーション コンポーネントが同時に変更されることによる、望ましくない影響をなくすことができます。 この機能を設定するには、グローバル コンフィギュレーション コマンド **configuration mode exclusive** mode を使用します。 動作モードには auto と manual の 2 つがあります。 auto モードでは、管理者が **configure terminal** EXEC コマンドを発行すると、コンフィギュレーションが自動的にロックされます。 manual モードでは、コンフィギュレーション モードに入る際に管理者が **configure terminal lock コマンド**を使用して、コンフィギュレーションをロックします。

次の例では、この機能を使用してコンフィギュレーションを自動的にロックする設定を示してます。

```
!
configuration mode exclusive auto
!
```

Cisco IOS ソフトウェアのコンフィギュレーション回復

Cisco IOS ソフトウェア リリース 12.3(8)T では、コンフィギュレーション回復機能が追加され、Cisco IOS デバイスで現在使用されている Cisco IOS ソフトウェア イメージとデバイス コンフィギュレーションのコピーを安全に保存できるようになりました。 この機能をイネーブルにすると、このバックアップ ファイルを変更したり削除したりすることはできません。 不注意や悪意によってこれらのファイルが削除されないように、この機能をイネーブルにすることを推奨いたします。

```
!
secure boot-image
secure boot-config!
```

この機能をイネーブルにすると、削除されたコンフィギュレーションや Cisco IOS ソフトウェア イメージを復元できます。 この機能の現在の実行状態を表示するには、**show secure boot** EXEC コマンドを使用します。

デジタル署名付き Cisco ソフトウェアの識別

Cisco 1900、2900、および 3900 シリーズ ルータの Cisco IOS ソフトウェア リリース 15.0(1) M では、デジタル署名付き Cisco ソフトウェア機能が追加されました。この機能により、セキュアな非対称（公開キー）暗号化を使用してデジタル署名されており信頼できる Cisco IOS ソフトウェアの使用が促進されます。

デジタル署名付きイメージには、イメージ自体の（秘密キーによる）暗号化ハッシュが含まれています。 検査時にデバイスは、キー ストア内の対応する公開キーを使用してハッシュを復号化し、イメージのハッシュを計算します。 復号化されたハッシュが計算されたイメージ ハッシュと一致する場合、そのイメージは改ざんされておらず信頼できます。

デジタル署名付き Cisco ソフトウェア キーは、キーのタイプとバージョンによって識別されます。 使用できるキーのタイプは、特殊、実稼働、およびロールオーバーです。 実稼働キー タイプと特殊キー タイプにはキー バージョンが関連付けられています。キー バージョンは、キーが失効し置換されるたびにアルファベット順で増分します。 デジタル署名付き Cisco ソフトウェア機能を使用するときには、ROMmon および標準 Cisco IOS イメージの両方が特殊キーまたは実稼働キーを使用して署名されます。 ROMmon イメージはアップグレード可能であり、またロードされる特殊イメージまたは実稼働イメージと同じキーを使って署名されている必要があります。

次のコマンドは、デバイス キー ストア内のキーを使用してフラッシュ内の c3900-universalk9-mz.SSA イメージの整合性を検証します。

```
show software authenticity file flash0:c3900-universalk9-mz.SSA
```
デジタル署名付き Cisco ソフトウェア機能は、Cisco Catalyst 4500 E シリーズ スイッチの IOS XE リリース 3.1.0.SG にも統合されました。

この機能についての詳細は、『デジタル署名付き Cisco ソフトウェア』を参照してください。

Cisco IOS ソフトウェア リリース 15.1(1) T 以降では、デジタル署名付き Cisco ソフトウェアのキー置換機能が導入されました。 キーの置換および失効機能により、デジタル署名付き Cisco ソフトウェアの検査に使用されたキーが置換され、プラットフォームのキー ストレージから削除されます。 キーの侵害が発生した場合に失効できるのは、特殊キーと実稼働キーだけです。

（特殊または実稼働）イメージの新しい（特殊または実稼働）キーは、以前の特殊キーまたは実稼働キーを失効させるために使用した（実稼働または失効）イメージに含まれています。 プラットフォームで事前に保存された状態で提供されるロールオーバー キーを使用して、失効イメージの整合性が検証されます。 ロールオーバー キーは変更されません。 ROMmon イメージがアップグレードされ、新しい実稼働イメージが起動されている場合に限り、実稼働キーを失効させる際に、失効イメージのロード後にこのイメージに含まれている新しいキーがキー ストアに追加され、対応する古いキーが失効可能になります。 特殊キーを失効させると、実稼働イメージがロードされます。 このイメージにより新しい特殊キーが追加され、古い特殊キーが失効可能になります。 ROMmon のアップグレード後に、新しい特殊イメージを起動できます。

次に、特殊キーの失効を示す例を示します。 次に示すコマンドは、現在の実稼働イメージからキー ストアに新しい特殊キーを追加し、新しい ROMmon イメージ（C3900_rom-monitor.srec.SSB）をストレージ エリア（usbflash0:）にコピーし、ROMmon ファイルをアップグレードし、古い特殊キーを失効させます。

```
software authenticity key add special
copy tftp://192.168.1.129/C3900_rom-monitor.srec.SSB usbflash0:
upgrade rom-monitor file usbflash0:C3900_PRIV_RM2.srec.SSB
software authenticity key revoke special
```
新しい特殊イメージ（c3900-universalk9-mz.SSB）を、ロードするフラッシュにコピーできます。このイメージの署名は、新しく追加された特殊キー（.SSB）を使用して検証されます。

```
copy /verify tftp://192.168.1.129/c3900-universalk9-mz.SSB flash:
```
Cisco IOS XE ソフトウェアが稼働する Catalyst 4500 E シリーズ スイッチでは、キーの失効と置換はサポートされませんが、デジタル署名付き Cisco ソフトウェア機能がサポートされます。

この機能の詳細については、『デジタル署名付き Cisco ソフトウェア』の「 デジタル署名付き Cisco ソフトウェア鍵の失効と置換」の項を参照してください。

コンフィギュレーション変更通知とロギング

コンフィギュレーション変更通知とロギング機能は、Cisco IOS ソフトウェア リリース 12.3(4)T で導入されました。この機能を使用すると、Cisco IOS デバイスのコンフィギュレーション変更をログに記録できます。 ログはその Cisco IOS デバイスで保存されます。ログに記録されるのは、変更を行った個人のユーザ情報、入力されたコンフィギュレーション コマンド、および変更時刻です。 この機能をイネーブルにするには、**logging enable** 設定変更ロガー コンフィギュレーション モード コマンドを使用します。 デフォルトの設定を改善するには、オプション コマンドの **hidekeys** および **logging size** エントリを使用します。これは、これらのコマンドによってパスワード データをログに記録しないように設定し、変更ログのサイズを増加するためです。

Cisco IOS デバイスのコンフィギュレーション変更履歴がわかりやすくなるように、この機能をイネーブルにすることを推奨いたします。 さらに、コンフィギュレーションの変更時に syslog

メッセージが生成されるように、**notify syslog** コンフィギュレーション コマンドを使用することを推奨します。

```
!

archive
log config
logging enable
logging size 200
hidekeys
notify syslog
!
```

コンフィギュレーション変更通知とロギング機能をイネーブルにすると、特権 EXEC コマンド **show archive log config all** を使用して、コンフィギュレーション ログを表示できます。

コントロール プレーン

コントロール プレーン機能は、発信元から宛先へデータを移動するために、ネットワーク デバイス間でやりとりされるプロトコルとプロセスで構成されます。 これには、ボーダー ゲートウェイ プロトコルなどのルーティング プロトコルや、ICMP および Resource Reservation Protocol（RSVP; リソース予約プロトコル）のようなプロトコルが含まれます。

管理プレーンおよびデータ プレーンでのイベントによって、コントロール プレーンに悪影響が及ばないようにすることが重要です。 DoS 攻撃のようなデータ プレーンのイベントによってコントロール プレーンに影響が及べば、ネットワーク全体が不安定になりかねません。 Cisco IOS ソフトウェア機能とコンフィギュレーションに関するこの情報は、コントロール プレーンの復元力確保に役立ちます。

コントロール プレーン全般の強化

管理プレーンとデータ プレーンの維持と稼働は、コントロール プレーンにかかっているので、ネットワーク デバイスのコントロール プレーンを保護することは重要です。 セキュリティ事象の発生中にコントロール プレーンが不安定になると、ネットワークの安定性を回復できないおそれがあります。

多くの場合、インターフェイスで特定の種類のメッセージの送受信をディセーブルにすることで、不要なパケットを処理するために必要な CPU 負荷を最小にできます。

IP ICMP リダイレクト

同じインターフェイスでパケットを送受信する際に、ルータでは ICMP リダイレクト メッセージが生成される場合があります。 この場合、ルータはパケットを転送し、元のパケットの送信者には ICMP リダイレクト メッセージを送信します。 この動作により、送信者はそのルータをバイパスして、後続パケットを宛先（または宛先により近いルータ）に直接転送できます。 正常に機能している IP ネットワークでは、ルータは自分のローカル サブネット上のホストに対してだけリダイレクトを送信します。 つまり、ICMP リダイレクトがレイヤ 3 バウンダリを超えることはありません。

ICMP リダイレクト メッセージには、 ホスト アドレスのリダイレクトとサブネット全体のリダイレクトという 2 つのタイプがあります。 悪意のあるユーザがルータに連続してパケットを送信し、これにより強制的にルータを ICMP リダイレクト メッセージに対応させて、CPU とルータのパフォーマンスに悪影響を及ぼすことによって、ICMP リダイレクトを送信するルータの機能

を悪用する可能性があります。 ルータが ICMP リダイレクトを送信しないようにするには、**no ip redirects** インターフェイス コンフィギュレーション コマンドを使用します。

ICMP 到達不能

インターフェイス アクセス リストによるフィルタリングを行うと、フィルタリングされたトラフィックの発信元には ICMP 到達不能メッセージが送信されます。 これらのメッセージの生成により、デバイスの CPU 使用率が増加することがあります。 Cisco IOS ソフトウェアでの ICMP 到達不能メッセージの生成は、デフォルトで 500 ミリ秒につき 1 パケットまでに制限されています。 ICMP 到達不能メッセージの生成を無効にするには、インターフェイス コンフィギュレーション コマンド **no ip unreachables** を使用します。 ICMP 到達不能レート制限をデフォルト設定から変更するには、グローバル コンフィギュレーション コマンド **icmp rate-limit unreachable** interval-in-ms を使用します。

プロキシ ARP

プロキシ ARP は、あるデバイス（通常はルータ）が、別のデバイスに宛てられた ARP 要求に応答する技法です。 ルータは ID を「偽装」することによって、実際の宛先にパケットをルーティングする責任を引き受けます。 プロキシ Address Resolution Protocol（ARP）を使用すると、ルーティングやデフォルト ゲートウェイを設定しなくても、サブネット上のマシンがリモートのサブネットに到達できます。 プロキシ ARP は、RFC 1027 で定義されています。

プロキシ ARP を使用するには、いくつかの短所があります。 プロキシ ARP を使用すると、ネットワーク セグメント上の ARP トラフィック、およびリソース枯渇攻撃や中間者（man-in-the-middle）攻撃が増加する可能性があります。 プロキシ ARP では、プロキシ処理されたそれぞれの ARP 要求が少量のメモリを消費するので、リソース枯渇攻撃が誘発されます。 攻撃者は ARP 要求を大量に送信することによって、利用可能なメモリを枯渇させることができます。

中間者攻撃では、ネットワーク上のホストがルータの MAC アドレスをスプーフィングすることによって、無警戒なホストが攻撃者にトラフィックを送信することが可能になります。 プロキシ ARP をディセーブルにするには、インターフェイス コンフィギュレーション コマンド **no ip proxy-arp** を使用します。

この機能についての詳細は、『プロキシ ARP のイネーブル化』を参照してください。

コントロール プレーン トラフィックの CPU への影響の制限

コントロール プレーンの保護は非常に重要です。 データ トラフィックと管理トラフィックが滞ればアプリケーション パフォーマンスとエンドユーザ エクスペリエンスが損なわれる可能性があるので、管理プレーンとデータ プレーンの維持と稼働はコントロール プレーンの持続性にかかっていると言えます。

コントロール プレーン トラフィックについて

きちんと Cisco IOSデバイスのコントロール プレーンを保護するために、CPU によって切り替えられるプロセスのトラフィック の 種類を理解することは必要です。 プロセス スイッチングされるトラフィックは、通常、2 種類のトラフィックで構成されます。 1 つ目の種類のトラフィックは Cisco IOS デバイスに誘導され、Cisco IOS デバイスの CPU で直接処理される必要があります。 このトラフィックは*レシーブ 隣接関係 トラフィックカテゴリ*で構成されています。 このトラフィックは Cisco Express Forwarding（CEF）テーブルで次のルータ ホップが **show ip cef** CLI 出力の条件 レシーブによって示されるデバイス自体であるというエントリが含まれています。 こ

の用語が表示されるのは、Cisco IOS デバイスの CPU で直接処理する必要がある IP アドレスの場合です。これには、インターフェイス IP アドレス、マルチキャスト アドレス レンジ、ブロードキャスト アドレス レンジがあります。

CPU で処理される 2 つ目の種類のトラフィックは、データ プレーン トラフィックです。これは、Cisco IOS デバイス以外を宛先としたトラフィックであり、CPU での特別な処理が必要です。CPU に影響を与えるデータ プレーン トラフィックはこれだけではありませんが、これらの種類のトラフィックはプロセス スイッチングされており、コントロール プレーンの動作に影響する可能性があります。

- **アクセス コントロール リスト ロギング**：ACL ロギング トラフィックは、log キーワードが使用された場合の ACE の一致（許可または拒否）によって生成されるあらゆるパケットで構成されます。

- **ユニキャスト リバース パス フォワーディング（ユニキャスト RPF）**：ユニキャスト RPF は、ACL とともに使用され、特定のパケットのプロセス スイッチングが行われる可能性があります。

- **IP オプション**：オプションが指定された任意の IP パケットは、CPU で処理する必要があります。

- **フラグメンテーション**：フラグメンテーションを必要とする任意の IP パケットは、CPU に渡して処理する必要があります。

- **存続可能時間（TTL）の期限切れ**：TTL 値が 1 以下のパケットでは、インターネット制御メッセージ プロトコルの Time Exceeded（ICMP タイプ 11、コード 0）メッセージが送信される必要があります。これにより CPU 処理が発生します。

- **ICMP 到達不能**：ルーティング、MTU、またはフィルタリングによって ICMP 到達不能メッセージを発生させるパケットは、CPU で処理されます。

- **ARP 要求を必要とするトラフィック**：ARP エントリが存在しない宛先は、CPU での処理が必要です。

- **非 IP トラフィック**：すべての非 IP トラフィックは CPU で処理されます。

次のリストでは、Cisco IOS デバイスの CPU で処理されているトラフィックの種類を判別する方法を詳しく説明しています。

- **show ip cef コマンド**を実行すると、CEF テーブルに含まれる各 IP プレフィクスのネクストホップ情報が表示されます。 前述したように、receive が「Next Hop」と表示されるエントリは受信隣接関係であるとみなされ、そのトラフィックは直接 CPU に送信される必要があることを示しています。

- **show interface switching コマンド**を実行すると、デバイスでプロセス スイッチングされているパケット数の情報が表示されます。

- **show ip traffic コマンド**を実行すると、次の IP パケットの数に関する情報が表示されます。

  宛先がローカルである（受信隣接関係トラフィック）オプションがついているフラグメンテ

ーションを必要としているブロードキャスト アドレス レンジに送られているマルチキャスト アドレス レンジに送られている

- 受信隣接関係トラフィックを識別するには、**show ip cache flow** コマンドを使用します。 Cisco IOS デバイス宛のフローの Destination Interface（DstIf; 宛先インターフェイス）は local と指定されています。

- **コントロール プレーン ポリシング**を使用すると、Cisco IOS デバイスのコントロール プレーンに到達するトラフィックの種類とレートを識別できます。 コントロール プレーン ポリシングを実行するには、詳細な分類の ACL、ロギング、および **show policy-map control-plane** コマンドを使用します。

インフラストラクチャ ACL

Infrastructure ACL（iACL; インフラストラクチャ ACL）によって、外部通信をネットワークのデバイスに制限できます。 インフラストラクチャ ACL については、このドキュメントの「インフラストラクチャ ACL によるネットワーク アクセス制限」の項で詳しく説明しています。

iACL を実装して、すべてのネットワーク デバイスのコントロール プレーンを保護することを推奨します。

受信 ACL

分散プラットフォームでは、Cisco 12000（GSR）用の Cisco IOS ソフトウェア リリース 12.0(21)S2、Cisco 7500 用のリリース 12.0(24)S、および Cisco 10720 用のリリース 12.0(31)S に、Receive ACL（rACL; 受信 ACL）を使用することもできます。 rACL は、ルート プロセッサが有害なトラフィックの影響を受ける前に、そのトラフィックからデバイスを保護します。 受信 ACL は、それが設定されたデバイスを保護するだけの設計になっており、通過トラフィックには影響を与えません。 したがって、次の例の ACL エントリに使用される宛先 IP アドレスは、ルータの物理 IP アドレスまたはバーチャル IP アドレスを参照するだけです。 受信 ACL もネットワーク セキュリティのベスト プラクティスと考えられており、優れたネットワーク セキュリティへの長期的な付加機能として考慮してください。

次に示すのは、192.168.100.0/24 ネットワーク上の信頼できるホストからの SSH（TCP ポート 22）トラフィックを許可するように記述された受信パス ACL です。

```
!
!--- Permit SSH from trusted hosts allowed to the device.
!

access-list 151 permit tcp 192.168.100.0 0.0.0.255 any eq 22
!
!--- Deny SSH from all other sources to the RP.
!

access-list 151 deny tcp any any eq 22
!
!--- Permit all other traffic to the device.
!--- according to security policy and configurations.
!

access-list 151 permit ip any any
!
!--- Apply this access list to the receive path.
```

```
!

ip receive access-list 151
!
```

デバイスへの正当なトラフィックを識別して許可を与え、望ましくないパケットをすべて拒否するには、『GSR： 受信アクセス コントロール リスト』を参照してください。

CoPP

CoPP 機能もインフラストラクチャ デバイスに向かう IP パケットを制限するために使用することができます。 次の例では、信頼できるホストからの SSH トラフィックのみが Cisco IOS デバイスの CPU に到達することを許可されます。

> **注:** 未知の IP アドレスや信頼できない IP アドレスからのトラフィックを廃棄することで、動的に割り当てられた IP アドレスを持つホストが Cisco IOS デバイスに接続するのを防止できます。

```
!

access-list 152 deny tcp <trusted-addresses> <mask> any eq 22
access-list 152 permit tcp any any eq 22
access-list 152 deny ip any any
!

class-map match-all COPP-KNOWN-UNDESIRABLE
match access-group 152
!

policy-map COPP-INPUT-POLICY
class COPP-KNOWN-UNDESIRABLE
drop
!

control-plane
service-policy input COPP-INPUT-POLICY
!
```

前記の CoPP の例では、ACL エントリの permit アクションに一致する不正なパケットがある場合、このようなパケットはポリシーマップの drop 機能によって廃棄されますが、deny アクションに一致するパケットは、ポリシーマップの drop 機能の影響を受けません。

CoPP は、Cisco IOS ソフトウェア リリース トレイン 12.0S、12.2SX、12.2S、12.3T、12.4、および 12.4T で使用できます。

CoPP 機能の設定と使用法についての詳細は、『コントロール プレーン ポリシングの展開』を参照してください。

コントロール プレーン保護

Control Plane Protection（CPPr; コントロール プレーン保護）機能は、Cisco IOS ソフトウェア リリース 12.4(4)T で導入されました。この機能を使用して、Cisco IOS デバイスの CPU 宛のコントロール プレーン トラフィックを制限および規制できます。 CPPr は CoPP と同様に、トラフィックを詳細に制限できます。 CPPr によって集約コントロール プレーンは、サブインターフェイスと呼ばれる 3 つの個別のコントロール プレーン カテゴリに分割されます。 サブインターフェイスが存在するのは、Host、Transit、および CEF-Exception のトラフィック カテゴリです。 さらに、CPPr には次のコントロール プレーン保護機能があります。

- **ポートフィルタリング機能**：閉じているか受信状態ではない TCP ポートや UDP ポートに送信されるパケットの規制や廃棄を行います。

- **キューしきい値機能**：コントロール プレーン IP 入力キューで許可されている指定されたプロトコルのパケット数を制限します。

CPPr 機能の設定と使用法についての詳細は、『コントロール プレーン保護および『コントロール プレーン保護（CPPr）について』を参照してください。

ハードウェア レート制限機能

Cisco Catalyst 6500 シリーズ Supervisor Engine 32 および Supervisor Engine 720 では、特殊なネットワーキング シナリオ用にプラットフォーム固有のハードウェアベースのレート制限機能（HWRL）がサポートされています。 これらのハードウェア レート制限機能は、IPv4、IPv6、ユニキャスト、マルチキャストの DoS 攻撃シナリオに関する詳細な定義済みセットを力バーしており、特殊なケースのレート リミッタと呼ばれています。 HWRL は、CPU で処理する必要があるパケットを必要とするさまざまな攻撃から Cisco IOS デバイスを保護できます。

いくつかの HWRL はデフォルトでイネーブルになっています。 詳細は、『PFC3 ハードウェアベース レート リミッタのデフォルト設定』を参照してください。

HWRL の詳細は、『PFC3 でのハードウェアベース レート リミッタ』を参照してください。

BGP の保護

Border Gateway Protocol（BGP; ボーダー ゲートウェイ プロトコル）は、インターネットのルーティングの基盤です。 したがって、標準より厳しい接続要件を設けている組織では、多くの場合、BGP を利用しています。 BGP は頻繁に偏在およびセットが理由で攻撃者によって目標とされ、より小さい組織の BGP設定の性質を忘れています。 ただし、BGP 設定のセキュリティ向上に利用できる多くの BGP 固有セキュリティ機能があります。

ここでは、最重要の BGP セキュリティ機能の概要を示します。 必要に応じて、コンフィギュレーションの推奨事項も示しています。

TTL ベースのセキュリティ保護

それぞれの IP パケットには、Time To Live（TTL; 存続可能時間）と呼ばれる 1 バイトのフィールドが含まれています。 IP パケットがデバイスを通過するごとに、この値は 1 ずつ減ります。 開始値はオペレーティング システムによって異なりますが、通常は 64 ~ 255 の間です。 TTL 値が 0 に達したパケットは廃棄されます。

Generalized TTL-based Security Mechanism（GTSM）および BGP TTL Security Hack（BTSH）と呼ばれる TTL ベースのセキュリティ保護では、IP パケットの TTL 値を利用することにより、受信 BGP パケットが直接接続されたピアからのものであることが保証されます。 この機能には多くの場合、ピアリング ルータからの同調が必要ですが、 イネーブルにすると、BGP に対する多くの TCP ベースの攻撃を完全に防ぐことができます。

BGP 用の GTSM をイネーブルにするには、**neighbor** BGP ルータ コンフィギュレーション コマンドの **ttl-security** オプションを使用します。 次の例は、この機能の設定を示しています。

!

```
router bgp <asn>
neighbor <ip-address> remote-as <remote-asn>
neighbor <ip-address> ttl-security hops <hop-count>
!
```

BGP パケットが受信されると、TTL 値がチェックされます。TTL 値は、255 から指定の hop-count を差し引いた数値以上である必要があります。

MD5 による BGP ピア認証

MD5 を使用するピア認証により、BGP セッションの一部として送信される各パケットの MD5 ダイジェストが作成されます。 ダイジェストの生成には、具体的には、IP および TCP ヘッダー部分、TCP ペイロード、および秘密鍵が使用されます。

作成されたダイジェストは TCP オプション Kind 19 に保存されます。これは、この目的のために RFC 2385 で定義されたオプションです。 受信 BGP スピーカはこれと同じアルゴリズムと秘密鍵を使用して、メッセージ ダイジェストを再生成します。 受信されたダイジェストと算出されたダイジェストが一致しない場合、パケットは廃棄されます。

MD5 によるピア認証を設定するには、**neighbor** BGP ルータ コンフィギュレーション コマンドの **password** オプションを使用します。 次に、このコマンドの使用法を示します。

```
!

router bgp <asn>
neighbor <ip-address> remote-as <remote-asn>
neighbor <ip-address> password <secret>
!
```

MD5 による BGP ピア認証についての詳細は『ネイバー ルータの認証』を参照してください。

最大プレフィックス数の設定

BGP プレフィクスはルータによりメモリに保持されます。 ルータが保持する必要があるプレフィクスが多くなるほど、BGP が使用する必要があるメモリが増えます。 プロバイダーの顧客ネットワークでデフォルト ルートのみを利用する設定など、設定によっては、すべてのインターネット プレフィクスのサブセットを保持できます。

メモリの枯渇を防ぐために、ピアごとに受け付けるプレフィクスの最大数を設定することが重要です。 各 BGP ピアに上限を設定することを推奨いたします。

neighbor maximum-prefix BGP ルータ コンフィギュレーション コマンドを使用してこの機能を設定するときには、 ピアのシャットダウン前に受け入れられるプレフィックスの最大数を引数として使用する必要があります。 オプションで、1 ~ 100 の数値を入力することもできます。 この数値は最大プレフィクス値に対するパーセンテージを表し、この値に達すると口グ メッセージが送信されます。

```
!

router bgp <asn>
neighbor <ip-address> remote-as <remote-asn>
neighbor <ip-address> maximum-prefix <shutdown-threshold> <log-percent>
!
```

ピアごとの最大プレフィクスについての詳細は、『BGP 最大プレフィクス機能の設定』を参照してください。

プレフィックス リストによる BGP プレフィックスのフィルタリング

プレフィクス リストを使用すると、BGP を介して送受信される特定のプレフィクスを許可または拒否できます。 ネットワーク トラフィックが想定どおりのパスを介して送信されるように、可能な限りプレフィクス リストを使用してください。 プレフィクス リストは、着信と発信の両方向で各 eBGP ピアに対して適用する必要があります。

プレフィクス リストを設定すると、送受信されるプレフィクスは、ネットワークのルーティング ポリシーによって具体的に許可されたプレフィクスに制限されます。 受信されるプレフィクスが多すぎてこれを実行できない場合は、既知の不正なプレフィクスをブロックするようにプレフィクス リストを設定する必要があります。 このような既知の不正プレフィクスには、未割り当ての IP アドレス レンジや、内部用やテスト用に RFC 3330 で予約済みのネットワークが含まれます。 発信プレフィクス リストでは、組織がアドバタイズするプレフィクスのみを具体的に許可するように設定します。

次の設定例では、プレフィクス リストを使用して、学習するルートとアドバタイズするルートを制限しています。 具体的には、プレフィクス リスト BGP-PL-INBOUND によってデフォルト ルートのみが着信を許可され、BGP-PL-OUTBOUND によってプレフィクス 192.168.2.0/24 のみがアドバタイズを許可されます。

```
!

ip prefix-list BGP-PL-INBOUND seq 5 permit 0.0.0.0/0
ip prefix-list BGP-PL-OUTBOUND seq 5 permit 192.168.2.0/24
!

router bgp <asn>
neighbor <ip-address> prefix-list BGP-PL-INBOUND in
neighbor <ip-address> prefix-list BGP-PL-OUTBOUND out
!
```

BGP プレフィクス フィルタリングの詳細は、『外部 BGP を使用したサービス プロバイダーへの接続』を参照してください。

自律システム パス アクセス リストによる BGP プレフィクスのフィルタリング

BGP 自律システム (AS) パス アクセス リストを使用すると、プレフィクスの AS パス アトリビュートに基づいて、受信されるプレフィクスとアドバタイズされるプレフィクスをフィルタリングできます。 これをプレフィクス リストと組み合わせて使用すると、堅牢なフィルタ セットが確立されます。

次の設定例では、AS パス アクセス リストを使用して、着信プレフィクスをリモート AS から発信されたプレフィクスに制限し、発信プレフィクスをローカルの自律システムから発信されたプレフィクスに制限しています。 その他すべての自律システムから発信されたプレフィクスはフィルタリングされ、ルーティング テーブルにはインストールされません。

```
!

ip as-path access-list 1 permit ^65501$
ip as-path access-list 2 permit ^$
!

router bgp <asn>
neighbor <ip-address> remote-as 65501
neighbor <ip-address> filter-list 1 in
neighbor <ip-address> filter-list 2 out
```

```
!
```

内部ゲートウェイ プロトコルの保護

ネットワークが適切にトラフィックを転送し、トポロジの変更や障害から回復する能力は、トポロジの正確なビューに依存しています。 多くの場合、内部ゲートウェイ プロトコル（IGP）を実行することで、このビューが得られます。 デフォルトでは、IGP は動的であり、使用中の特定の IGP と通信するルータの追加を検出します。 また、IGP により、ネットワーク リンク障害の発生中に使用可能なルータを検出することもできます。

以降のサブセクションでは、最重要の IGP セキュリティ機能の概要を示しています。 Routing Information Protocol バージョン 2（RIPv2）、Enhanced IGRP（EIGRP）、および OSPF（Open Shortest Path First）について、推奨事項や使用例を交えて説明しています。

MD5 によるルーティング プロトコル認証と検証

ルーティング情報の交換を保護できなければ、攻撃者が不正なルーティング情報をネットワークに持ち込むことが可能になります。 ルータ間でルーティング プロトコルによるパスワード認証を使用することで、ネットワークのセキュリティを強化できます。 ただし、この認証はクリアテキストで送信されるので、攻撃者がこのセキュリティ制御を弱体化させるのは容易である可能性があります。

認証プロセスに MD5 ハッシュ機能を付加することで、ルーティング アップデートにはクリアテキストのパスワードが含まれなくなり、ルーティング アップデートのコンテンツ全体が改ざんされにくくなります。 ただし、弱いパスワードが選択されている場合は、MD5 認証が力ずくの攻撃や辞書攻撃の影響を受けやすいことに変わりありません。 十分にランダム化されたパスワードを使用してください。 MD5 認証は、パスワード認証と比べると格段にセキュリティが強化されているので、次の例は MD5 認証に特化しています。 IPSec もルーティング プロトコルの検証と保護に使用できますが、次の例ではその使用法については詳しく触れていません。

EIGRP と RIPv2 では、設定の一部としてキー チェーンを使用します。 キー チェーンの設定と使用法の詳細は、『鍵』を参照してください。

MD5 による EIGRP ルータ認証の設定例を次に示します。

```
!

key chain <key-name>
key <key-identifier>
key-string <password>
!

interface <interface>
ip authentication mode eigrp <as-number> md5
ip authentication key-chain eigrp <as-number> <key-name>
!
```

RIPv2 用の MD5 ルータ認証の設定例を次に示します。 RIPv1 では認証はサポートされていません。

```
!

key chain <key-name>
key <key-identifier>
key-string <password>
!
```

```
interface <interface>
ip rip authentication mode md5
ip rip authentication key-chain <key-name>
!
```

MD5 による OSPF ルータ認証の設定例を次に示します。 OSPF ではキー チェーンを使用しません。

```
!

interface <interface>
ip ospf message-digest-key <key-id> md5 <password>
!

router ospf <process-id>
network 10.0.0.0 0.255.255.255 area 0
area 0 authentication message-digest
!
```

詳細は、『OSPF の設定』を参照してください。

passive-interface コマンド

ルーティング情報のアドバタイズメントを制御する **passive-interface** コマンドを使用することで、情報のリーク、つまり不正な情報の IGP への流入を緩和できます。 管理制御の及ばないネットワークへは、情報をいっさいアドバタイズしないでください。

次の例では、この機能の使用方法を示します。

```
!

router eigrp <as-number>
passive-interface default
no passive-interface <interface>
!
```

ルート フィルタリング

不正なルーティング情報がネットワークに持ち込まれる可能性を減らすために、ルート フィルタリングを使用する必要があります。 **passive-interface** ルータ コンフィギュレーション コマンドとは異なり、ルート フィルタリングがイネーブルになると、ルーティングはインターフェイス上で実行されますが、アドバタイズされる情報や処理される情報は制限されます。

EIGRP および RIP の場合、**distribute-list** コマンドに **out** キーワードを指定すると、どの情報をアドバタイズするかが制限され、**in** キーワードを指定すると、どのアップデートが処理されるのかが制限されます。 OSPF では **distribute-list コマンド**を使用できますが、このコマンドによって、フィルタリングされたルートの伝搬がルータで阻止されることはありません。 代わりに、**area filter-list** コマンドが使用できます。

次の EIGRP の例では、**distribute-list** コマンドとプレフィクス リストによって発信アドバタイズメントをフィルタリングしています。

```
!

ip prefix-list <list-name> seq 10 permit <prefix>
!

router eigrp <as-number>
passive-interface default
```

```
no passive-interface <interface>
distribute-list prefix <list-name> out <interface>
!
```

次の EIGRP の例では、プレフィクス リストによって着信アップデートをフィルタリングしています。

```
!

ip prefix-list <list-name> seq 10 permit <prefix>
!

router eigrp <as-number>
passive-interface default
no passive-interface <interface>
distribute-list prefix <list-name> in <interface>
!
```

ルーティング アップデートのアドバタイズと処理を制御する方法の詳細は、『IP ルーティングのプロトコルには依存しない機能の設定』を参照してください。

次の OSPF の例では、OSPF 固有の **area filter-list コマンド**とプレフィクス リストを使用しています。

```
!

ip prefix-list <list-name> seq 10 permit <prefix>
!

router ospf <process-id>
area <area-id> filter-list prefix <list-name> in
!
```

ルーティング プロセスのリソース消費

ルーティング プロトコル プレフィクスはルータでメモリに保持され、ルータが保持するプレフィクスが増えればリソース消費も上昇します。 リソースの枯渇を防ぐために、リソース消費を制限するようにルーティング プロトコルを設定することが重要です。 これを OSPF で実現するには、リンクステート データベース過負荷保護機能を使用します。

次の例では、OSPF のリンクステート データベース過負荷保護機能の設定を示しています。

```
!

router ospf <process-id>
max-lsa <maximum-number>
!
```

OSPF のリンクステート データベース過負荷保護の詳細は、『OSPF プロセスでの自己生成 LSA 数の制限』を参照してください。

ファースト ホップ冗長プロトコルの保護

First Hop Redundancy Protocol（FHRP; ファースト ホップ冗長プロトコル）によって、デフォルト ゲートウェイとして機能するデバイスの復元力と冗長性が強化されます。 この状況やこれらのプロトコルは、2 台のレイヤ 3 デバイスがネットワーク セグメント、またはサーバやワークステーションを含む VLAN においてデフォルト ゲートウェイとして機能している環境では一般的です。

Gateway Load-Balancing Protocol（GLBP; ゲートウェイ ロードバランシング プロトコル）、Hot

Standby Router Protocol (HSRP; ホットスタンバイ ルータ プロトコル) 、および Virtual Router Redundancy Protocol (VRRP; 仮想ルータ冗長プロトコル) は、どれも FHRP です。 デフォルトでは、これらのプロトコルにより認証なしで通信されます。 このような通信では、攻撃者が FHRP に準拠するデバイスになりすましてネットワーク上でデフォルト ゲートウェイの役割を担うことが可能です。 このような乗っ取りが行われた場合、攻撃者は中間者攻撃を実行したり、ネットワーク内のすべてのユーザ トラフィックを傍受したりするおそれがあります。

この種類の攻撃を防止するために、Cisco IOS ソフトウェアでサポートされるすべての FHRP には、MD5 かテキスト文字列のどちらかを使用する認証機能が組み込まれています。 脅威がもたらされるのは認証が行われていない FHRP によるので、これらのプロトコルのインスタンスでは MD5 認証を使用することが推奨されます。 次の設定例では、GLBP、HSRP、および VRRP の MD5 認証の使用法を示しています。

```
!

interface FastEthernet 1
description *** GLBP Authentication ***
glbp 1 authentication md5 key-string <glbp-secret>
glbp 1 ip 10.1.1.1
!

interface FastEthernet 2
description *** HSRP Authentication ***
standby 1 authentication md5 key-string <hsrp-secret>
standby 1 ip 10.2.2.1
!

interface FastEthernet 3
description *** VRRP Authentication ***
vrrp 1 authentication md5 key-string <vrrp-secret>
vrrp 1 ip 10.3.3.1
!
```

データ プレーン

データ プレーンは発信元から宛先へのデータの移動を担当しますが、セキュリティという観点では、データ プレーンは 3 つのプレーンの中で最も重要性が低くなっています。 このため、ネットワーク デバイスを保護する場合は、データ プレーンよりも管理プレーンとコントロール プレーンを保護することの方が重要です。

ただし、データ プレーン自体にも、トラフィックの保護に役立つ機能や設定オプションが多数あります。 以降のセクションでは、ネットワークの保護をより容易にする機能やオプションについて説明しています。

データ プレーン全般の強化

データ プレーン トラフィックの大多数は、ネットワークのルーティング設定に決められたとおりネットワークを通過します。 ただし、ネットワークを通過するパケットのパスを変更する IP ネットワーク機能が存在します。 IP オプション、とりわけソース ルーティング オプションなどの機能は、今日のネットワークにおけるセキュリティ面の課題です。

また、データ プレーンの強化には、トランジット ACL の使用も関係します。

詳細は、このドキュメントの「通過トラフィックのトランジット ACL によるフィルタリング」の項を参照してください。

IP オプションの選択的廃棄

IP オプションに関係するセキュリティの懸念は 2 つあります。 IP オプションを含むトラフィックは、Cisco IOS デバイスによってプロセス スイッチングされる必要があります。このことが、CPU 負荷の上昇を招くことがあります。 また、IP オプションには、ネットワークを通過するトラフィックのパスを変更する機能も含まれています。これによりセキュリティ制御が弱体化する可能性があります。

これらの懸念から、Cisco IOS ソフトウェア リリース 12.3(4)T、12.0(22)S、および 12.2(25)S には、グローバル コンフィギュレーション コマンド ip **options {drop | ignore}** が追加されています。 最初の形式の **ip options drop** を使用すると、Cisco IOS デバイスによって受信される IP オプションを含むすべての IP パケットが廃棄されます。 これにより、CPU 負荷の上昇と、IP オプションによって引き起こされる可能性があるセキュリティ制御の弱体化の両方が防止されます。

2 番目の形式の **ip options ignore** を使用すると、受信パケットに含まれる IP オプションが無視されるように Cisco IOS デバイスが設定されます。 これにより、ローカル デバイスでは IP オプションに関連する脅威が緩和されますが、ダウンストリームのデバイスでは IP オプションの存在による影響を受ける可能性があります。 このため、**drop** 形式でこのコマンドを実行することを強く推奨します。 次に、設定例を示します。

```
!
ip options drop
!
```

RSVP のように、IP オプションを正当な目的で使用するプロトコルもあります。 このようなプロトコルの機能は、このコマンドの影響を受けます。

IP オプションの選択的廃棄をイネーブルにすると、**show ip traffic** EXEC コマンドを使用して、IP オプションの存在によって廃棄されたパケットの数を把握できます。 この情報は、forced drop カウンタに示されます。

この機能についての詳細は、『ACL の IP オプション選択的ドロップ』を参照してください。

IP ソース ルーティングのディセーブル化

IP ソース ルーティングでは、Loose Source Route オプションと Record Route オプションを同時に、または Strict Source Route オプションと Record Route オプションを使用して、パケットが通過するネットワーク パスを IP データグラムのソース側で指定できます。 ネットワークのセキュリティ制御に関するトラフィックをルーティングしようとする場合にもこの機能を使用できます。

IP オプションの選択的廃棄機能によって IP オプションを完全にディセーブルにしていない場合は、IP ソース ルーティングをディセーブルにすることが重要です。 IP ソース ルーティングはすべての Cisco IOS ソフトウェア リリースでデフォルトでイネーブルになっています。これをディセーブルにするには、**no ip source-route** グローバル コンフィギュレーション コマンドを使用します。 次の設定例は、このコマンドの使用方法を示しています。

```
!
no ip source-route
!
```

ICMP リダイレクトのディセーブル化

ICMP リダイレクトは、ネットワーク デバイスに、IP 宛先までのよりよいパスを通知するために

使用されます。 デフォルトでは、受信したインターフェイスを介してルーティングを行う必要があるパケットを受信した場合に、リダイレクトが送信されます。

状況によっては、攻撃者が Cisco IOS デバイスが多数の ICMP リダイレクト メッセージを送信するように仕向け、その結果、CPU 負荷を上昇させることも可能です。 このため、ICMP リダイレクトの伝送をディセーブルにすることを推奨いたします。 ICMP リダイレクトをディセーブルにするには、次の設定例に示すように、インターフェイス コンフィギュレーション コマンド **no ip redirects** を使用します。

```
!

interface FastEthernet 0
no ip redirects
!
```

IP ダイレクト ブロードキャストのディセーブル化または制限

IP ダイレクト ブロードキャストによって、IP ブロードキャスト パケットをリモート IP サブネットに送信できるようになります。 パケットがリモート ネットワークに到達すると、フォワーディング IP デバイスによってパケットはレイヤ 2 ブロードキャストとしてサブネット上の全ステーションに送信されます。 このダイレクト ブロードキャスト機能は、SMURF 攻撃などいくつかの攻撃で増幅やリフレクションの手段として利用されてきました。

現行の Cisco IOS ソフトウェア リリースでは、この機能はデフォルトでディセーブルになっています。 ただし、**ip directed-broadcast** インターフェイス コンフィギュレーション コマンドを使用してイネーブルにできます。 12.0 よりも前の Cisco IOS ソフトウェア リリースでは、この機能はデフォルトでイネーブルになっています。

ネットワークにダイレクト ブロードキャスト機能が不可欠な場合は、使用方法を制御する必要があります。 これを行うには、**ip directed-broadcast コマンド**のオプションとしてアクセス コントロール リストを使用します。 次の設定例では、ダイレクト ブロードキャストを信頼できるネットワーク 192.168.1.0/24 から発信された UDP パケットに制限しています。

```
!

access-list 100 permit udp 192.168.1.0 0.0.0.255 any
!

interface FastEthernet 0
ip directed-broadcast 100
!
```

通過トラフィックのトランジット ACL によるフィルタリング

トランジット ACL (tACL) を使用すると、どのトラフィックがネットワークを通過するかを制御できます。 これは、ネットワーク自体を宛先とするトラフィックのフィルタリングを行うインフラストラクチャ ACL とは対照的です。 特定のデバイス グループへのトラフィックやネットワークを通過しているトラフィックのフィルタリングを行うことが望ましい場合は、tACL によるフィルタリングが便利です。

この種類のフィルタリングは、従来からファイアウォールで実行されています。 ただし、フィルタリングを実行する必要があるにもかかわらずファイアウォールがない場合など、このフィルタリングをネットワーク内の Cisco IOS デバイスで行うことが有益な場合があります。

トランジット ACL は、静的なアンチスプーフィング保護を実装する場所としても適しています。

詳細は、このドキュメントの「アンチスプーフィング機能」の項を参照してください。

tACL についての詳細は、『トランジット アクセス コントロール リスト： エッジでのフィルタリング』を参照してください。

ICMP パケット フィルタリング

インターネット制御メッセージ プロトコル (ICMP) は、IP 用のコントロール プロトコルとしての設計になっています。 このため、ICMP で伝送されるメッセージは一般に、TCP プロトコルや IP プロトコルに対して広範囲に影響を及ぼす可能性があります。 ネットワーク トラブルシューティング ツールの **ping** や **traceroute**、およびパス MTU ディスカバリでは ICMP が使用されますが、 ネットワークが正常に動作している場合、外部 ICMP 接続が必要になることはほとんどありません。

Cisco IOS ソフトウェアには、ICMP メッセージを名前または種類およびコードで詳細にフィルタリングする機能があります。 次の例の ACL は、信頼できるネットワークからの ICMP を許可し、それ以外の発信元からのすべての ICMP パケットをブロックしています。

```
!

ip access-list extended ACL-TRANSIT-IN
!
!--- Permit ICMP packets from trusted networks only
!

permit icmp host <trusted-networks> any
!
!--- Deny all other IP traffic to any network device
!

deny icmp any any
!
```

IP フラグメントのフィルタリング

このドキュメントの「インフラストラクチャ ACL によるネットワーク アクセス制限」の項で説明したように、フラグメント化された IP パケットのフィルタリングは、セキュリティ デバイスにとっての課題です。

フラグメント処理はわかりにくいため、IP フラグメントが誤って ACL によって許可されることがあります。 また、侵入検知システムによる検出を逃れようとして、フラグメンテーションが使用されることもよくあります。 このような理由から、IP フラグメントは攻撃で使用されることが多く、設定された tACL の先頭で明示的にフィルタリングを適用する必要があります。 次の ACL の例には、あらゆる IP フラグメントのフィルタリングが含まれます。 次の例の機能は、これまでの例の機能と組み合わせて使用する必要があります。

```
!

ip access-list extended ACL-TRANSIT-IN
!
!--- Deny IP fragments using protocol-specific ACEs to aid in
!--- classification of attack traffic
!

deny tcp any any fragments
deny udp any any fragments
deny icmp any any fragments
```

```
deny ip any any fragments
!
```

フラグメント化された IP パケットの ACL による処理の詳細は、『アクセス コントロール リストと IP フラグメント』を参照してください。

ACL の IP オプション フィルタリング サポート

Cisco IOS ソフトウェア リリース 12.3(4)T 以降では、ACL を使用して、パケットに含まれる IP オプションに基づいて IP パケットをフィルタリングする機能がサポートされています。 パケット内に IP オプションがあるということは、ネットワーク内のセキュリティ制御を弱体化させようとしているか、パケットの転送特性を変えようとしていることを示しています。 このような理由から、IP オプションが付いたパケットは、ネットワークのエッジでフィルタリングする必要があります。

IP オプションを含む IP パケットに対して完全なフィルタリングを行うには、次の例をこれまでの例の内容と組み合わせて使用する必要があります。

```
!

ip access-list extended ACL-TRANSIT-IN
!
!--- Deny IP packets containing IP options
!

deny ip any any option any-options
!
```

アンチスプーフィング保護

攻撃の多くは、効果を高めるためや、攻撃の実際の発信元の隠蔽や正確なトレースバックの妨害のために、発信元 IP アドレスのスプーフィングを利用しています。 Cisco IOS ソフトウェアには、発信元 IP アドレスのスプーフィングを利用した攻撃を防止するために、ユニキャスト RPF および IP ソース ガード (IPSG) という機能があります。 さらに、多くの場合、スプーフィングを手動で阻止する手段として ACL とヌル ルーティングが展開されます。

IP ソース ガードは、スイッチ ポート、MAC アドレス、および発信元アドレスの検証を行うことによって、直接の管理制御下にあるネットワークに対するスプーフィングを最小に抑えることができます。 ユニキャスト RPF では、発信元ネットワークの検証が行われ、直接の管理制御下にないネットワークからのスプーフィング攻撃を抑制できます。 ポート セキュリティを使用すると、アクセス レイヤで MAC アドレスの検証が行われます。 Dynamic Address Resolution Protocol (ARP) Inspection (DAI) により、ローカル セグメントの ARP ポイズニングを利用する攻撃が抑制されます。

ユニキャスト RPF

ユニキャスト RPF では、転送されたパケットを受信したインターフェイスを介して、そのパケットの発信元アドレスに到達できるかどうかをデバイスで確認できます。 スプーフィング対策をユニキャスト RPF だけに依存しないでください。 発信元 IP アドレスに戻る適切なルートが存在する場合は、スプーフィングされたパケットが、ユニキャスト RPF に対応したインターフェイスを介してネットワークに侵入する可能性があります。 ユニキャスト RPF を使用するには、各デバイスで Cisco エクスプレス フォワーディングがイネーブルになっている必要があります。ユニキャスト RPF はインターフェイスごとに設定されます。

ユニキャスト RPF には loose と strict という 2 つの動作モードがあり、 どちらかを設定します。

非対称ルーティングが存在する場合は、loose モードを推奨いたします。strict モードではこのような場合、パケットが廃棄されるからです。 **ip verify** インターフェイス コンフィギュレーション コマンドの設定で、キーワード **any** を指定すると loose モード、キーワード **rx** を指定すると strict モードになります。

次の例は、この機能の設定を示しています。

```
!

ip cef
!

interface <interface>
ip verify unicast source reachable-via <mode>
!
```

ユニキャスト RPF の設定と使用法についての詳細は、『ユニキャスト リバース パス転送について』を参照してください。

IP ソース ガード

レイヤ 2 インターフェイスを制御できる場合、IP ソース ガードはスプーフィングを防止する有効な手段です。 IP ソース ガードは、DHCP スヌーピングからの情報を使用して、レイヤ 2 インターフェイス上にポート アクセス コントロール リスト（PACL）を動的に設定し、IP ソース バインディング テーブルに関連付けられていない IP アドレスからのトラフィックを拒否します。

IP ソース ガードは、DHCP スヌーピングがイネーブルの VLAN に属するレイヤ 2 インターフェイスに適用できます。 DHCP スヌーピングは次のコマンドによってイネーブルになります。

```
!

ip dhcp snooping
ip dhcp snooping vlan <vlan-range>
!
```

DHCP スヌーピングをイネーブルにした後、次のコマンドによって IPSG がイネーブルになります。

```
!
interface <interface-id>
ip verify source
!
```

ポート セキュリティをイネーブルにするには、**ip verify source port security** インターフェイス コンフィギュレーション コマンドを使用します。 これには、グローバル コンフィギュレーション コマンド **ip dhcp snooping information option** を実行する必要があります。 さらに、DHCP サーバが DHCP オプション 82 をサポートしている必要があります。

この機能の詳細は、『DHCP 機能および IP ソース ガードの設定』を参照してください。

ポート セキュリティ

ポート セキュリティを使用すると、アクセス インターフェイスでの MAC アドレスのスプーフィングが抑制されます。 ポート セキュリティでは、動的に学習された（スティッキ）MAC アドレスを使用することで、初期設定が容易になります。 ポート セキュリティによって MAC 違反が特定されると、4 つの違反モードのいずれかが適用されます。 これには protect、restrict、shutdown、および shutdown VLAN のモードがあります。 ポートが、標準プロトコルを使用する

1 台のワークステーションのアクセスを提供するだけの場合、最大数は 1 で十分です。 最大数が 1 に設定された場合、バーチャル MAC アドレスを使用する HSRP などのプロトコルは機能しません。

```
!

interface <interface>
switchport
switchport mode access
switchport port-security
switchport port-security mac-address sticky
switchport port-security maximum <number>
switchport port-security violation <violation-mode>
!
```

ポート セキュリティの設定の詳細は、『ポート セキュリティの設定』を参照してください。

ダイナミック ARP インスペクション

ダイナミック ARP インスペクション (DAI) を使用すると、ローカル セグメントの ARP ポイズニング攻撃を抑制できます。 ARP ポイズニング攻撃とは、攻撃者が偽装した ARP 情報をローカル セグメントに送信するという手法です。 この情報は、他のデバイスの ARP キャッシュを破損する設計になっています。 攻撃者は、ARP ポイズニングを使用して中間者攻撃を実行します。

DAI では、信頼できないポートですべての ARP パケットを傍受し、IP アドレスと MAC アドレスの関連付けを検証します。 DHCP 環境では、DAI は DHCP スヌーピング機能によって生成されたデータを使用します。 信頼できるインターフェイスから受信した ARP パケットは検証されず、信頼できないインターフェイス上の無効なパケットは廃棄されます。 非 DHCP 環境では、ARP ACL を使用する必要があります。

DHCP スヌーピングは次のコマンドによってイネーブルになります。

```
!
ip dhcp snooping
ip dhcp snooping vlan <vlan-range>
!
```

DHCP スヌーピングをイネーブルにした後、次のコマンドによって DAI がイネーブルになります。

```
!
ip arp inspection vlan <vlan-range>
!
```

非 DHCP 環境で DAI をイネーブルにするには、ARP ACL を使用する必要があります。 次の設定例は、DAI と ARP ACL の基本設定を示しています。

```
!

arp access-list <acl-name>
permit ip host <sender-ip> mac host <sender-mac>
!

ip arp inspection filter <arp-acl-name> vlan <vlan-range>
!
```

DAI の設定方法の詳細は、『ダイナミック ARP インスペクションの設定』を参照してください。

アンチスプーフィング ACL

手動で設定された ACL は、既知で未使用のアドレス レンジや信頼できないアドレス レンジを使用する攻撃に対して、静的なアンチスプーフィング機能を提供します。 通常、このようなアンチスプーフィング ACL は、より大規模な ACL のコンポーネントとしてネットワーク バウンダリで入トラフィックに適用されます。 アンチスプーフィング ACL は頻繁に変更されることがあるので、定期的に監視する必要があります。 送信 ACL を適用してトラフィックを有効なローカル アドレスに制限すると、ローカル ネットワークから発信するトラフィックでのスプーフィングを最小に抑えることができます。

次の例は、ACL を使用して IP スプーフィングを制限する方法を示しています。 この ACL は、対象のインターフェイスの着信側に適用されます。 この ACL を構成する ACE は、すべてを網羅しているわけではありません。 このような種類の ACL を設定する場合、確実な最新の参照を含めるようにしてください。

```
!

ip access-list extended ACL-ANTISPOOF-IN
deny ip 10.0.0.0 0.255.255.255 any
deny ip 192.168.0.0 0.0.255.255 any
!

interface <interface>
ip access-group ACL-ANTISPOOF-IN in
!
```

アクセス コントロール リストの設定方法の詳細は、『一般的に使用される IP ACL の設定』を参照してください。

未割り当てのインターネット アドレスの公式リストは、Team Cymru によって管理されています。 未使用アドレスのフィルタリングについての詳細は、『Bogon Reference Page』を参照してください。

データ プレーン トラフィックの CPU への影響の制限

ルータとスイッチの主目的は、パケットやフレームをデバイス経由で最終的な宛先まで転送することです。 これらのパケットは、ネットワーク上に展開されたデバイスを通過するので、デバイスの CPU 動作に影響を及ぼす可能性があります。 データ プレーンは、ネットワーク デバイスを通過するトラフィックで構成されているため、管理プレーンとコントロール プレーンが確実に動作するようにするには、データ プレーンを保護する必要があります。 通過トラフィックによってデバイスでのトラフィックのプロセス スイッチングが発生する場合、デバイスのコントロール プレーンに影響が出ることがあり、その結果、動作が中断される可能性もあります。

CPU に影響する機能とトラフィックの種類

特別な CPU 処理を必要とするデータ プレーン トラフィックを以下に示します。これらのトラフィックは CPU でプロセス スイッチングされます。ただし、これはすべてを網羅したリストではありません。

- **ACL ロギング**：ACL ロギング トラフィックは、**log キーワード**が使用された場合の ACE の一致（許可または拒否）によって生成されるパケットで構成されます。
- **ユニキャスト RPF**：ユニキャスト RPF が ACL とともに使用される場合、特定のパケットのプロセス スイッチングが行われる可能性があります。
- **IP オプション**：オプションが指定された任意の IP パケットは、CPU で処理する必要があり

ます。

- **フラグメンテーション**：フラグメンテーションを必要とする任意の IP パケットは、CPU に渡して処理する必要があります。

- **存続可能時間（TTL）の期限切れ**：TTL 値が 1 以下のパケットでは、インターネット制御メッセージ プロトコルの Time Exceeded（ICMP タイプ 11、コード 0）メッセージが送信される必要があります。これにより CPU 処理が発生します。

- **ICMP 到達不能**：ルーティング、MTU、またはフィルタリングによって ICMP 到達不能メッセージを発生させるパケットは、CPU で処理されます。

- **ARP 要求を必要とするトラフィック**：ARP エントリが存在しない宛先は、CPU での処理が必要です。

- **非 IP トラフィック**：すべての非 IP トラフィックは CPU で処理されます。

データ プレーンの強化の詳細については、このドキュメントの「データ プレーン全般の強化」の項を参照してください。

TTL 値に基づくフィルタ

ACL の TTL 値フィルタリング サポート機能を使用すると、拡張 IP アクセス リストで TTL 値に基づいてパケットをフィルタリングできます。この機能は、Cisco IOS ソフトウェア リリース 12.4(2)T で追加されています。 この機能を使用すると、TTL 値が 0 または 1 の通過トラフィックを受信するデバイスを保護できます。 また、TTL 値に基づいてパケットをフィルタリングすることで、TTL 値がネットワークの全長以上であることが保証され、ダウンストリーム インフラストラクチャ デバイスのコントロール プレーンを TTL 期限切れ攻撃から保護できます。

アプリケーションや **traceroute** などのツールでは、テストや診断のために TTL 期限切れパケットが使用されます。 IGMP など一部のプロトコルでは、TTL 値が 1 のパケットが正規の目的で使用されます。

次の ACL の例では、TTL 値が 6 未満の IP パケットをフィルタリングするポリシーが作成されます。

```
!
!--- Create ACL policy that filters IP packets with a TTL value
!--- less than 6
!

ip access-list extended ACL-TRANSIT-IN
deny ip any any ttl lt 6
permit ip any any
!
!--- Apply access-list to interface in the ingress direction
!

interface GigabitEthernet 0/0
ip access-group ACL-TRANSIT-IN in
!
```

TTL 値に基づくパケット フィルタリングについての詳細は、『TTL 超過攻撃の識別と緩和』を参照してください。

この機能についての詳細は、『ACL の TTL 値フィルタリング サポート』を参照してください。

Cisco IOS ソフトウェア リリース 12.4(4)T 以降では、Flexible Packet Matching (FPM) 機能によって、パケット内の任意のビットを照合することができます。 次の FPM ポリシーでは、TTL 値が 6 未満のパケットが廃棄されます。

```
!

load protocol flash:ip.phdf
!

class-map type access-control match-all FPM-TTL-LT-6-CLASS
match field IP ttl lt 6
!

policy-map type access-control FPM-TTL-LT-6-DROP-POLICY
class FPM-TTL-LT-6-CLASS
drop
!

interface FastEthernet0
service-policy type access-control input FPM-TTL-LT-6-DROP-POLICY
!
```

この機能についての詳細は、『Cisco IOS Flexible Packet Matching』ホームページの『Cisco Flexible Packet Matching』を参照してください。

IP オプションの有無によるフィルタ

Cisco IOS ソフトウェア リリース 12.3(4)T 以降では、名前付き拡張 IP アクセス リストで ACL の IP オプション フィルタリング サポート機能を使用して、IP オプションの有無に基づいて IP パケットをフィルタリングできます。 また、IP オプションの有無に基づいて IP パケットをフィルタリングすることで、インフラストラクチャ デバイスのコントロール プレーンで、これらのパケットを CPU レベルで処理する必要を無くすこともできます。

ACL の IP オプション フィルタリング サポート機能は、名前付き拡張 ACL でのみ使用できます。 また、RSVP、マルチプロトコル ラベル スイッチング トラフィック エンジニアリング、IGMP バージョン 2 と 3、および IP オプション パケットを使用するその他のプロトコルは、これらのプロトコルのパケットが廃棄された場合、正常に機能しない可能性があります。 これらのプロトコルをネットワークで使用している場合でも、ACL の IP オプション フィルタリング サポート機能を使用できますが、 ACL の IP オプション選択的廃棄機能によってこれらのトラフィックが廃棄される可能性があり、これらのプロトコルは正常に動作しない可能性があります。 これらのパケットの廃棄に ACL の IP オプション選択的廃棄が推奨されるのは、IP オプションを必要とするプロトコルが使用されていない場合です。

次の ACL の例では、IP オプションを含む IP パケットをフィルタリングするポリシーが作成されます。

```
!

ip access-list extended ACL-TRANSIT-IN
deny ip any any option any-options
permit ip any any
!

interface GigabitEthernet 0/0
ip access-group ACL-TRANSIT-IN in
!
```

次の ACL の例では、特定の 5 つの IP オプションを含む IP パケットをフィルタリングするポリシーを示しています。 次のオプションを含むパケットが拒否されます。

- 0 オプション リストの終端（eool）
- 7 ルートの記録（record-route）
- 68 タイム スタンプ（timestamp）
- 131 ルース ソース ルート（lsr）
- 137 ストリクト ソース ルート（ssr）

```
!

ip access-list extended ACL-TRANSIT-IN
deny ip any any option eool
deny ip any any option record-route
deny ip any any option timestamp
deny ip any any option lsr
deny ip any any option ssr
permit ip any any
!

interface GigabitEthernet 0/0
ip access-group ACL-TRANSIT-IN in
!
```

ACL の IP オプション選択的廃棄についての詳細は、このドキュメントの「データ プレーン全般の強化」の項を参照してください。

tACL についての詳細は、『トランジット アクセス コントロール リスト： エッジでのフィルタリング』を参照してください。

IP オプション付きパケットのフィルタリングに使用できる Cisco IOS ソフトウェアのもう一つの機能に、CoPP があります。 Cisco IOS ソフトウェア リリース 12.3(4)T 以降では、CoPP によってコントロール プレーン パケットのトラフィック フローをフィルタリングできます。 CoPP と、Cisco IOS ソフトウェア リリース 12.3(4)T で追加された ACL の IP オプション フィルタリング サポート機能に対応したデバイスでは、アクセス リスト ポリシーを使用して、IP オプションを含むパケットをフィルタリングできます。

次の CoPP ポリシーでは、デバイスで受信される通過パケットに IP オプションが付いている場合、そのパケットが廃棄されます。

```
!

ip access-list extended ACL-IP-OPTIONS-ANY
permit ip any any option any-options
!

class-map ACL-IP-OPTIONS-CLASS
match access-group name ACL-IP-OPTIONS-ANY
!

policy-map COPP-POLICY
class ACL-IP-OPTIONS-CLASS
drop
!
```

```
control-plane
service-policy input COPP-POLICY
!
```

次の CoPP ポリシーでは、デバイスで受信される通過パケットに次の IP オプションが付いている場合、そのパケットが廃棄されます。

- 0 オプション リストの終端（eool）
- 7 ルートの記録（record-route）
- 68 タイム スタンプ（timestamp）
- 131 ルース ソース ルート（lsr）
- 137 ストリクト ソース ルート（ssr）

```
!

ip access-list extended ACL-IP-OPTIONS
permit ip any any option eool
permit ip any any option record-route
permit ip any any option timestamp
permit ip any any option lsr
permit ip any any option ssr
!

class-map ACL-IP-OPTIONS-CLASS
match access-group name ACL-IP-OPTIONS
!

policy-map COPP-POLICY
class ACL-IP-OPTIONS-CLASS
drop
!

control-plane
service-policy input COPP-POLICY
!
```

上記の CoPP ポリシーでは、アクセス コントロール リスト エントリ（ACE）の permit アクションに一致するパケットがある場合、このようなパケットはポリシーマップの drop 機能によって廃棄されますが、deny アクション（非表示）に一致するパケットは、ポリシーマップの drop 機能の影響を受けません。

CoPP 機能についての詳細は、『コントロール プレーン ポリシングの展開』を参照してください。

コントロール プレーン保護

Cisco IOS ソフトウェア リリース 12.4(4)T では、コントロール プレーン保護（CPPr）機能を使用して、Cisco IOS デバイスの CPU によってコントロール プレーン トラフィックを制限および規制できます。 CPPr は CoPP と似ていますが、CoPP よりも詳細にトラフィックを制限または規制できます。 CPPr によって集約コントロール プレーンは、サブインターフェイスと呼ばれる 3 つの個別のコントロール プレーン カテゴリに分割されます。 Host、Transit、および CEF-Exception サブインターフェイスが存在します。

次の CPPr ポリシーでは、デバイスで受信される通過パケットの TTL 値が 6 未満の場合、またはデバイスで受信される通過パケットや非通過パケットの TTL 値が 0 か 1 の場合、そのパケットは廃棄されます。 さらに、デバイスで受信されるパケットに指定の IP オプションが付いている場合、そのパケットもドロップされます。

```
!

ip access-list extended ACL-IP-TTL-0/1
permit ip any any ttl eq 0 1
!

class-map ACL-IP-TTL-0/1-CLASS
match access-group name ACL-IP-TTL-0/1
!

ip access-list extended ACL-IP-TTL-LOW
permit ip any any ttl lt 6
!

class-map ACL-IP-TTL-LOW-CLASS
match access-group name ACL-IP-TTL-LOW
!

ip access-list extended ACL-IP-OPTIONS
permit ip any any option eool
permit ip any any option record-route
permit ip any any option timestamp
permit ip any any option lsr
permit ip any any option ssr
!

class-map ACL-IP-OPTIONS-CLASS
match access-group name ACL-IP-OPTIONS
!

policy-map CPPR-CEF-EXCEPTION-POLICY
class ACL-IP-TTL-0/1-CLASS
drop
class ACL-IP-OPTIONS-CLASS
drop
!

!-- Apply CPPr CEF-Exception policy CPPR-CEF-EXCEPTION-POLICY to
!-- the CEF-Exception CPPr sub-interface of the device

!

control-plane cef-exception
service-policy input CPPR-CEF-EXCEPTION-POLICY
!

policy-map CPPR-TRANSIT-POLICY
class ACL-IP-TTL-LOW-CLASS
drop
!

control-plane transit
service-policy input CPPR-TRANSIT-POLICY
!
```

前述の CPPr ポリシーでは、アクセス コントロール リスト エントリの permit アクションに一致するパケットがある場合、このようなパケットはポリシーマップの drop 機能によって廃棄されますが、deny アクション（非表示）に一致するパケットは、ポリシーマップの drop 機能の影響を

受けません。

CPPr 機能の詳細については、『コントロール プレーン保護について』および『コントロール プレーン保護』を参照してください。

トラフィックの識別とトレースバック

時には、ネットワーク トラフィックを迅速に識別してトレースバックする必要があることもあります。とりわけ、問題への対応時やネットワーク パフォーマンスが悪いときです。 Cisco IOS ソフトウェアでこれを実現する主な方法には、NetFlow と分類 ACL の 2 つがあります。 NetFlow を使用すると、ネットワーク上のすべてのトラフィックを把握できます。 さらに、NetFlow には長期的なトレンディングと自動分析を実行するコレクタを実装できます。 分類 ACL は、ACL のコンポーネントであり、トラフィックを識別するための事前計画と分析中の手動介入が必要です。 以下のセクションでは、これらの機能の簡単な概要を示します。

NetFlow

NetFlow は、ネットワーク フローをトラッキングすることで、異常なネットワーク アクティビティやセキュリティに関係するネットワーク アクティビティを特定します。 NetFlow のデータは、CLI から表示と分析ができます。また、市販またはフリーウェアの NetFlow コレクタにデータをエクスポートして、集計や分析を行うこともできます。 NetFlow コレクタは、長期的なトレンディングから、ネットワーク動作や使用状況を分析できます。 NetFlow は、IP パケット内の特定のアトリビュートを分析し、フローを作成することによって機能します。 最もよく使用されている NetFlow のバージョンは 5 ですが、バージョン 9 の方が拡張性に富んでいます。 NetFlow のフローは、高ボリュームの環境でサンプリングされたトラフィック データを使用して作成できます。

NetFlow をイネーブルにするには、前提条件として CEF または分散型 CEF が必要です。 NetFlow はルータやスイッチ上で設定できます。

次の例では、この機能の基本設定を示しています。 Cisco IOS ソフトウェアのこれまでのリリースでは、インターフェイスに対して NetFlow をイネーブルにするコマンドは、**ip route-cache flow** です（**ip flow {ingress | egress}** ではありません）。

```
!

ip flow-export destination <ip-address> <udp-port>
ip flow-export version <version>
!

interface <interface>
ip flow <ingess|egress>
!
```

CLI からの NetFlow の出力例を次に示します。 SrcIf アトリビュートはトレースバックに使用できます。

```
router#show ip cache flow
IP packet size distribution (26662860 total packets):
1-32 64 96 128 160 192 224 256 288 320 352 384 416 448 480
.741 .124 .047 .006 .005 .005 .002 .008 .000 .000 .003 .000 .001 .000 .000

512 544 576 1024 1536 2048 2560 3072 3584 4096 4608
.000 .000 .001 .007 .039 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 4456704 bytes
55 active, 65481 inactive, 1014683 added
```

```
41000680 ager polls, 0 flow alloc failures
Active flows timeout in 2 minutes
Inactive flows timeout in 60 seconds
IP Sub Flow Cache, 336520 bytes
110 active, 16274 inactive, 2029366 added, 1014683 added to flow
0 alloc failures, 0 force free
1 chunk, 15 chunks added
last clearing of statistics never
Protocol Total Flows Packets Bytes Packets Active(Sec) Idle(Sec)
-------- Flows /Sec /Flow /Pkt /Sec /Flow /Flow
TCP-Telnet 11512 0.0 15 42 0.2 33.8 44.8
TCP-FTP 5606 0.0 3 45 0.0 59.5 47.1
TCP-FTPD 1075 0.0 13 52 0.0 1.2 61.1
TCP-WWW 77155 0.0 11 530 1.0 13.9 31.5
TCP-SMTP 8913 0.0 2 43 0.0 74.2 44.4
TCP-X 351 0.0 2 40 0.0 0.0 60.8
TCP-BGP 114 0.0 1 40 0.0 0.0 62.4
TCP-NNTP 120 0.0 1 42 0.0 0.7 61.4
TCP-other 556070 0.6 8 318 6.0 8.2 38.3
UDP-DNS 130909 0.1 2 55 0.3 24.0 53.1
UDP-NTP 116213 0.1 1 75 0.1 5.0 58.6
UDP-TFTP 169 0.0 3 51 0.0 15.3 64.2
UDP-Frag 1 0.0 1 1405 0.0 0.0 86.8
UDP-other 86247 0.1 226 29 24.0 31.4 54.3
ICMP 19989 0.0 37 33 0.9 26.0 53.9
IP-other 193 0.0 1 22 0.0 3.0 78.2
Total: 1014637 1.2 26 99 32.8 13.8 43.9

SrcIf SrcIPaddress DstIf DstIPaddress Pr SrcP DstP Pkts
Gi0/1 192.168.128.21 Local 192.168.128.20 11 CB2B 07AF 3
Gi0/1 192.168.150.60 Gi0/0 10.89.17.146 06 0016 101F 55
Gi0/0 10.89.17.146 Gi0/1 192.168.150.60 06 101F 0016 9
Gi0/1 192.168.150.60 Local 192.168.206.20 01 0000 0303 11
Gi0/0 10.89.17.146 Gi0/1 192.168.150.60 06 07F1 0016 1
```

NetFlow の機能の詳細は、『Cisco IOS NetFlow』を参照してください。

NetFlow の技術概要については、『Cisco IOS NetFlow の概要 - 技術的概要』を参照してください。

分類 ACL

分類 ACL を使用すると、インターフェイスを通過するトラフィックを把握できます。 分類 ACL によって、ネットワークのセキュリティ ポリシーが変更されることはありません。通常、分類 ACL は、個別のプロトコル、発信元アドレス、または宛先を分類するように作成されます。 たとえば、すべてのトラフィックを許可する ACE は、プロトコル単位、またはポート単位に分類できます。 各トラフィック カテゴリにヒット カウンタが付いているので、トラフィックを特定の ACE へと詳細に分類することで、ネットワーク トラフィックを把握しやすくなります。 また、ACL の最後にある暗黙的な deny を詳細な ACE に分類することで、拒否したトラフィックの種類を識別することもできます。

show access-list EXEC コマンドと **clear ip access-list counters** EXEC コマンドで分類 ACL を使用することで、問題に迅速に対応できます。

次の例では、デフォルトの deny の前に SMB トラフィックを識別する分類 ACL の設定を示しています。

```
!
```

```
ip access-list extended ACL-SMB-CLASSIFY
remark Existing contents of ACL
remark Classification of SMB specific TCP traffic
deny tcp any any eq 139
deny tcp any any eq 445
deny ip any any
!
```

分類 ACL を使用するトラフィックを識別するには、**show access-list** acl-name EXEC コマンドを使用します。 ACL カウンタをクリアするには、**clear ip access-list counters** acl-name EXEC コマンドを使用します。

```
router#show access-list ACL-SMB-CLASSIFY
Extended IP access list ACL-SMB-CLASSIFY
10 deny tcp any any eq 139 (10 matches)
20 deny tcp any any eq 445 (9 matches)
30 deny ip any any (184 matches)
```

ACL の ロギング機能をイネーブルにする方法についての詳細は、『アクセス コントロール リストのロギングについて』を参照してください。

VLAN マップとポート アクセス コントロール リストによるアクセス コントロール

VLAN アクセス コントロール リスト（VACL）、つまり VLAN マップとポート ACL（PACL）を使用すると、ルーティングされたインターフェイスに適用されるアクセス コントロール リストよりもエンドポイント デバイスに近い場所にある、ルーティングされていないトラフィックに対してアクセス コントロールを適用できます。

このセクションでは、VACL と PACL の機能、利点、考えられる使用状況シナリオの概要を示しています。

VLAN マップによるアクセス コントロール

VACL（VLAN に流入するすべてのパケットに適用される VLAN マップ）を使用すると、VLAN 内のトラフィックに対してアクセス コントロールを適用できます。 これは、ルーティングされたインターフェイスに対して ACL を使用するのでは不可能なことです。 たとえば、VLAN マップを使用して、同じ VLAN 内のホストが相互に通信することを防止できます。これにより、ローカルの攻撃者やワームによる、同じネットワーク セグメント内のホストの悪用を最小に抑えることができます。 VLAN マップを使用してパケットを拒否するには、そのトラフィックに照合する ACL を作成し、VLAN マップ内で廃棄するアクションを設定します。 VLAN マップを設定すると、LAN に流入するすべてのパケットは順に、設定された VLAN マップに照らして評価されます。 VLAN アクセス マップでは、IPv4 と MAC のアクセス リストがサポートされています。 ただし、ACL のロギングや IPv6 ACL はサポートされません。

次の例では、名前付き拡張アクセス リストを使用し、この機能の設定を示しています。

```
!

ip access-list extended <acl-name>
permit <protocol> <source-address> <source-port> <destination-address>
<destination-port>
!

vlan access-map <name> <number>
match ip address <acl-name>
action <drop|forward>
!
```

次の例では、VLAN マップを使用して TCP ポート 139 とポート 445、および vines-ip プロトコルを拒否する方法を示しています。

```
!

ip access-list extended VACL-MATCH-ANY
permit ip any any
!

ip access-list extended VACL-MATCH-PORTS
permit tcp 192.168.1.0 0.0.0.255 192.168.1.0 0.0.0.255 eq 445
permit tcp 192.168.1.0 0.0.0.255 192.168.1.0 0.0.0.255 eq 139
!

mac access-list extended VACL-MATCH-VINES
permit any any vines-ip
!

vlan access-map VACL 10
match ip address VACL-MATCH-VINES
action drop
!

vlan access-map VACL 20
match ip address VACL-MATCH-PORTS
action drop
!

vlan access-map VACL 30
match ip address VACL-MATCH-ANY
action forward
!

vlan filter VACL vlan 100
!
```

VLAN マップの設定についての詳細は、『ACL によるネットワーク セキュリティの設定』を参照してください。

PACL によるアクセス コントロール

PACL を適用できるのは、スイッチのレイヤ 2 物理インターフェイスの着信側のみです。 PACL では VLAN マップと同様に、ルーティングされていないトラフィックやレイヤ 2 トラフィックに対してアクセス コントロールが適用されます。 PACL を作成するための構文はルータ ACL と同じであり、PACL は VLAN マップおよびルータ ACL より優先されます。 レイヤ 2 インターフェイスに適用される ACL は、PACL と呼ばれます。 設定では、IPv4、IPv6、または MAC の ACL の作成と、レイヤ 2 インターフェイスへの適用を行います。

次の例では、名前付き拡張アクセス リストを使用し、この機能の設定を示しています。

```
!

ip access-list extended <acl-name>
permit <protocol> <source-address> <source-port> <destination-address>
<destination-port>
!

interface <type> <slot/port>
switchport mode access
switchport access vlan <vlan_number>
```

```
ip access-group <acl-name> in
!
```

PACL の設定についての詳細は、『ACL によるネットワーク セキュリティの設定』のポート ACL に関するセクションを参照してください。

MAC によるアクセス コントロール

MAC アクセス コントロール リストまたは拡張リストは、インターフェイス コンフィギュレーション モードで次のコマンドを使用して IP ネットワークに適用できます。

```
Cat6K-IOS(config-if)#mac packet-classify
```

> **注:** レイヤ 2 パケットとしてレイヤ 3 パケットを分類します。 このコマンドは Cisco IOS ソフトウェア リリース 12.2(18)SXD（Sup 720 の場合）および Cisco IOS ソフトウェア リリース 12.2(33)SRA 以降でサポートされています。

このインターフェイス コマンドを入力インターフェイスに適用する必要があります。このコマンドは、フォワーディング エンジンに対して IP ヘッダーを検査しないように指示します。 これにより、IP 環境で MAC アクセス リストを使用できます。

プライベート VLAN ドメイン

プライベート VLAN（PVLAN）は、VLAN 内のワークステーションやサーバ間の接続を制限するレイヤ 2 セキュリティ機能です。 PVLAN を使用しない場合、レイヤ 2 VLAN 上のすべてのデバイスは自由に通信可能です。 単一の VLAN 上でデバイス間の通信を制限することで、セキュリティを保護できるネットワーキング環境があります。 たとえば、一般アクセスが可能なサブネット内でサーバ間の通信を禁止するために、PVLAN がよく使用されます。 1 台のサーバに侵入されても、PVLAN の適用により他のサーバへの接続が阻止されれば、被害はその 1 台のみに限定できます。

プライベート VLAN には、 隔離 VLAN、コミュニティ VLAN、およびプライマリ VLAN という 3 つの種類があります。 PVLAN の設定では、プライマリ VLAN とセカンダリ VLAN を使用します。 プライマリ VLAN には、すべての混合モード ポート（後述）と、1 つ以上のセカンダリ VLAN（隔離 VLAN またはコミュニティ VLAN）が含まれます。

隔離 VLAN

セカンダリ VLAN を隔離 VLAN として設定することで、セカンダリ VLAN 内のデバイス間の通信を完全に禁止できます。 1 つのプライマリ VLAN に含めることができる隔離 VLAN は 1 つだけです。また、隔離 VLAN 内のポートと通信できるのは、混合モード ポートのみです。 ゲスト ユーザをサポートするネットワークなど非信頼ネットワークでは、隔離 VLAN を使用する必要があります。

次の設定例では、VLAN 11 を隔離 VLAN として設定し、プライマリ VLAN である VLAN 20 に関連付けています。 また、インターフェイス FastEthernet 1/1 を VLAN 11 の隔離ポートとして設定しています。

```
!

vlan 11
private-vlan isolated
!
```

```
vlan 20
private-vlan primary
private-vlan association 11
!

interface FastEthernet 1/1
description *** Port in Isolated VLAN ***
switchport mode private-vlan host
switchport private-vlan host-association 20 11
!
```

コミュニティ VLAN

セカンダリ VLAN がコミュニティ VLAN として設定されている場合、VLAN のメンバ間の通信は許可され、プライマリ VLAN の任意の混合モード ポートを使用できます。 ただし、2 つのコミュニティ VLAN 間の通信や、コミュニティ VLAN から隔離 VLAN への通信は許可されません。 サーバを相互に接続する必要があるが、VLAN 内のその他のデバイスと接続する必要はない場合、サーバをグループ化するには、コミュニティ VLAN を使用する必要があります。 これは、一般アクセスが可能なネットワークや、信頼できないクライアントにサーバがコンテンツを提供する場合に一般的なシナリオです。

次の例では、1 つのコミュニティ VLAN を設定し、スイッチ ポート FastEthernet 1/2 をその VLAN のメンバとして設定しています。 コミュニティ VLAN である VLAN 12 は、プライマリ VLAN である VLAN 20 に対してセカンダリ VLAN です。

```
!

vlan 12
private-vlan community
!

vlan 20
private-vlan primary
private-vlan association 12
!

interface FastEthernet 1/2
description *** Port in Community VLAN ***
switchport mode private-vlan host
switchport private-vlan host-association 20 12
!
```

混合モード ポート

プライマリ VLAN に配置されるスイッチ ポートは、混合モード ポートと呼ばれます。 混合モード ポートは、プライマリ VLAN とセカンダリ VLAN の他のすべてのポートと通信できます。 これらの VLAN で見られる最も一般的なデバイスは、ルータやファイアウォールのインターフェイスです。

次の設定例では、これまでの隔離 VLAN とコミュニティ VLAN の例を組み合わせて、インターフェイス FastEthernet 1/12 を混合モード ポートとして追加しています。

```
!

vlan 11
private-vlan isolated
!

vlan 12
```

```
private-vlan community
!

vlan 20
private-vlan primary
private-vlan association 11-12
!

interface FastEthernet 1/1
description *** Port in Isolated VLAN ***
switchport mode private-vlan host
switchport private-vlan host-association 20 11
!

interface FastEthernet 1/2
description *** Port in Community VLAN ***
switchport mode private-vlan host
switchport private-vlan host-association 20 12
!

interface FastEthernet 1/12
description *** Promiscuous Port ***
switchport mode private-vlan promiscuous
switchport private-vlan mapping 20 add 11-12
!
```

PVLAN を実装する場合に重要なのは、展開されているレイヤ 3 設定が PVLAN による制限に対応し、PVLAN の設定が妨害されないことを確認することです。 ルータ ACL やファイアウォールでレイヤ 3 フィルタリングを行うことによって、PVLAN の設定が妨害されないようにできます。

プライベート VLAN の使用法と設定についての詳細は、『LAN セキュリティ』ホームページの『プライベート VLAN（PVLAN）：混合モード、隔離、コミュニティ』を参照してください。

結論

このドキュメントでは、Cisco IOS システム デバイスの保護に使用できる方法の概要を幅広く説明しています。 デバイスを保護することで、管理対象のネットワークの全体的なセキュリティが高まります。 この概要では、管理プレーン、コントロール プレーン、およびデータ プレーンの保護について説明し、設定に関する推奨事項を示しています。 関連のある各機能の設定については、可能な限り、十分詳しく説明しています。 ただし、より詳しい評価に必要な情報を提供するためには、すべての場合で包括的な参照先を示しています。

謝辞

このドキュメントにある機能の説明の一部は、Cisco 情報開発チームが作成したものです。

付録： Cisco IOS デバイス強化のチェックリスト

このチェックリストは、このガイドで説明した強化手順をすべてまとめています。 管理者は、該当しないために実装されなかった強化機能がある場合でも、Cisco IOS デバイスで考慮および使用されるすべての強化機能をまとめた参照情報として使用できます。 管理者に対し、オプションの実装前に各オプションに潜在的なリスクがあるかどうかを評価することが推奨されます。

管理プレーン

- パスワード

  イネーブル パスワードとローカル ユーザ パスワードの MD5 ハッシュ（秘密オプション）をイネーブルにするパスワード再試行ロックアウトを設定するパスワード回復をディセーブルにする（リスクを検討）

- 使用していないサービスを無効にする

- 管理セッションの TCP キープアライブを設定する

- メモリおよび CPU しきい値通知を設定する

- 設定

  メモリおよび CPU しきい値通知コンソール アクセス用のメモリ予約メモリ リーク検出バッファ オーバーフロー検出拡張 crashinfo 収集

- iACL を使用して管理アクセスを制限する

- フィルタリング（リスクを検討）

  ICMP パケットIP フラグメントIP オプションパケットの TTL 値

- コントロール プレーン保護

  ポート フィルタリングを設定するキューのしきい値を設定する

- 管理アクセス

  管理プレーン保護を使用して管理インターフェイスを制限するexec タイムアウトを設定するCLI アクセスに暗号化トランスポート プロトコル（SSH など）を使用するvty 回線と tty 回線の転送を制御するバナーを使用した警告

- [AAA]

  認証とフォールバックに AAA を使用するコマンド認証に AAA（TACACS+）を使用するアカウンティングに AAA を使用する冗長な AAA サーバを使用する

- SNMP

  SNMPv2 コミュニティを設定して ACL を適用するSNMPv3 を設定する

- ロギング

  集中型ロギングを設定する関連するすべてのコンポーネントのログ レベルを設定するlogging source-interface を設定するロギング タイムスタンプの詳細を設定する

- コンフィギュレーション管理

置換とロールバックコンフィギュレーション変更の排他的アクセスソフトウェア復元の設定設定変更の通知

コントロール プレーン

- ディセーブル（リスクを検討）

  ICMP リダイレクトICMP unreachablesプロキシ ARP

- NTP を使用する場合は NTP 認証を設定する
- コントロール プレーン ポリシング/保護（ポート フィルタリング、キューしきい値）を設定する
- ルーティング プロトコルを保護する

  BGP（TTL、MD5、最大プレフィックス、プレフィックス リスト、システム パス ACL）IGP（MD5、パッシブ インターフェイス、ルート フィルタリング、リソース消費）

- ハードウェア レート制限機能を設定する
- ファーストホップ冗長プロトコル（GLBP、HSRP、VRRP）を保護する

データ プレーン

- IP オプションの選択的ドロップを設定する
- ディセーブル（リスクを検討）

  IP 送信元ルーティングIP 誘導ブロードキャストICMP リダイレクト

- IP ダイレクト ブロードキャストを制限する
- tACL を設定する（リスクを検討）

  ICMP をフィルタリングするIP フラグメントをフィルタリングするIP オプションをフィルタリングするTTL 値をフィルタリングする

- 必要なアンチ スプーフィング保護を設定する

  ACLIP ソース ガードダイナミック ARP インスペクションユニキャスト RPFポート セキュリティ

- コントロール プレーン保護（control-plane cef-exception）
- トラフィックを識別するため NetFlow および分類 ACL を設定する

- 必要なアクセス コントロール ACL（VLAN マップ、PACL、MAC）を設定する
- プライベート VLAN を設定する