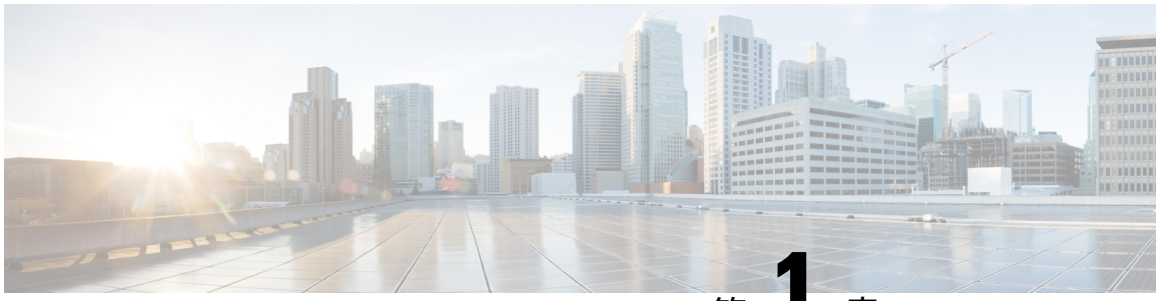
# Cisco IOS XE Gibraltar 16.10.x セキュリティ コンフィギュレーション ガイド：アクセス コントロール リスト


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

# 目 次



第 1 章

# 最初にお読みください

### Cisco IOS XE 16 に関する重要な情報

現行の Cisco IOS XE Release 3.7.0E（Catalyst スイッチング用）および Cisco IOS XE Release 3.17S（アクセスおよびエッジルーティング用）の 2 つのリリースは、単一バージョンのコンバージドリリース Cisco IOS XE 16 に進化（マージ）しました。これにより、スイッチングおよびルーティング ポートフォリオにおける広範なアクセス製品およびエッジ製品を 1 つのリリースでカバーします。

### 機能情報

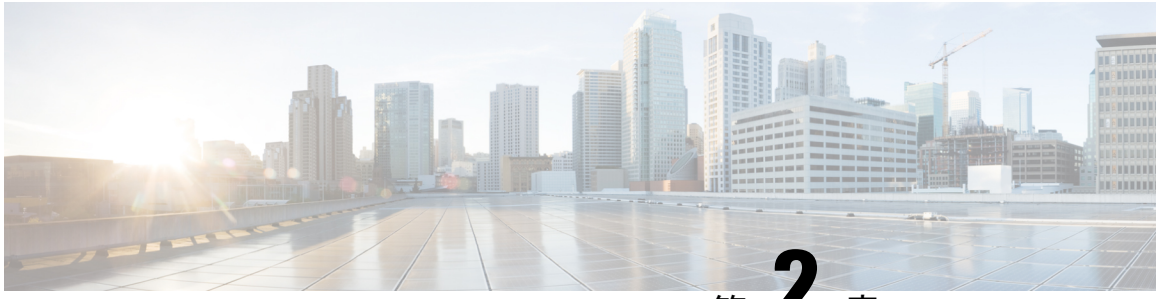
機能のサポート、プラットフォームのサポート、およびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco.com のアカウントは必要ありません。

### 参考資料

- 『Cisco IOS Command References, All Releases』

### マニュアルの入手方法およびテクニカル サポート

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



第 2 章

# IP アクセス リストの概要

アクセス コントロール リスト（ACL）は、パケット フィルタリングを実行して、ネットワークを介して移動するパケットと移動先を制御します。パケット フィルタリングによって、ネットワーク トラフィックを制限し、ユーザおよびデバイスのネットワークに対するアクセスを制限し、トラフィックがネットワークから外部に送信されるのを防ぐことで、セキュリティを実現します。IP アクセス リストによって、スプーフィングやサービス妨害攻撃の可能性を軽減し、ファイアウォールを介した動的で一時的なユーザ アクセスが可能になります。

また、IP アクセス リストは、セキュリティ以外の用途にも使用できます。たとえば、帯域幅制御、ルーティング アップデートのコンテンツの制限、ルートの再配布、ダイヤルオンデマンド（DDR）呼び出しのトリガー、デバッグ出力の制限、Quality of Service（QoS）機能のトラフィックの識別と分類などです。このモジュールでは、IP アクセス リストの概要について説明します。



## 機能情報の確認

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

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

# IP アクセス リストに関する情報

## IP アクセス リストの利点

アクセス コントロール リスト（ACL）は、ネットワークを通過するパケットのフローを制御するためにパケット フィルタリングを実行します。パケット フィルタリングによってユーザおよびデバイスのネットワークに対するアクセスを制限し、セキュリティの手段として利用できます。アクセス リストによってトラフィック数を減らすことで、ネットワーク リソースを節約できます。アクセス リストを使用した場合の利点は次のとおりです。

- 着信 rsh および rcp 要求を認証する：アクセス リストは、デバイスへのアクセスを制御するように構成された認証データベース内のローカル ユーザ、リモート ホスト、およびリモート ユーザの識別を簡素化できます。Cisco ソフトウェアは認証データベースを使用して、リモート シェル（rsh）およびリモート コピー（rcp）プロトコルの着信要求を受け取ることができます。
- 不要なトラフィックまたはユーザをブロックする：アクセス リストを使用すると、インターフェイス上の着信パケットまたは発信パケットをフィルタできるため、送信元アドレス、宛先アドレス、またはユーザ認証に基づいてネットワークへのアクセスを制御できます。また、アクセス リストを使用して、デバイス インターフェイスで転送またはブロックするトラフィックの種類を決定することもできます。たとえば、電子メール トラフィックはネットワークでルーティングすることを許可し、すべての Telnet トラフィックはネットワークに入ることをブロックするようにアクセス リストを使用できます。
- vty へのアクセスを制御する：インバウンド vty（Telnet）でのアクセス リストは、デバイスへの回線にアクセスできるユーザを制御できます。アウトバウンド vty でのアクセス リストは、デバイスからの回線が到達可能な宛先を制御できます。
- QoS 機能のトラフィックを特定または分類する：アクセス リストは、Weighted Random Early Detection（WRED）および専用アクセス レート（CAR）の IP プレシデンスを設定することで、輻輳回避を提供します。また、クラスベース均等化キューイング（CBWFQ）、プライオリティ キューイング、カスタム キューイングのために輻輳管理を提供します。
- debug コマンド出力を制限する：アクセス リストは、IP アドレスやプロトコルに基づいて debug 出力を制限できます。
- 帯域幅制御を提供する：低速リンクでのアクセス リストはネットワークでの過剰なトラフィックを防止できます。
- NAT 制御を提供する：アクセス リストによって、ネットワーク アドレス変換（NAT）が変換するアドレスを制御できます。
- DoS 攻撃の可能性を低減する：アクセス リストは、サービス妨害（DoS）攻撃の可能性を低減させます。ホストからのトラフィック、ネットワーク、またはネットワークにアクセスするユーザを制御するように IP 発信元アドレスを指定します。TCP インターセプト機

能を設定することで、接続に関する要求でサーバにフラッディングが発生しないようにすることができます。

- ルーティング アップデートの内容を制限する：アクセス リストによって、ネットワーク内で送信、受信、または再配布されるルーティング アップデートを制御できます。
- ダイヤルオンデマンド コールをトリガーする：アクセス リストによって、ダイヤルおよび切断条件を適用できます。

# アクセスリストを使用する必要がある境界ルータおよびファイアウォール ルータ

アクセスリストを設定する理由は多数あります。たとえば、アクセスリストを使用して、ルーティング アップデートのコンテンツを制限したり、トラフィック フローを制御したりできます。アクセスリストを設定する最も重要な理由の1つは、ネットワークに対するアクセスを制御することで、ネットワークに基本レベルのセキュリティを提供することです。ルータでアクセスリストを設定しない場合、ルータを通過するすべてのパケットは、ネットワークのすべての部分で許可される可能性があります。

アクセスリストで、ネットワークの一部に対してアクセスを許可するホストと、同じ領域に対してアクセスを禁止するホストを設定できます。以下の図では、適切なアクセスリストをルータのインターフェイスに適用することで、ホスト A は Human Resources ネットワークに対するアクセスが許可され、ホスト B は Human Resources ネットワークに対するアクセスが禁止されます。

ファイアウォール ルータにはアクセス リストを使用する必要があります。多くの場合、ファイアウォールルータは内部ネットワークと外部ネットワーク（インターネット）の間に配置されます。また、ネットワークの2つの部分の間に配置されたルータにアクセスリストを使用して、内部ネットワークの特定の部分に発着信するトラフィックを制御できます。

アクセスリストのセキュリティ上の利点を実現するために、場合によっては、少なくとも境界ルータでアクセスリストを設定する必要があります。境界ルータとは、ネットワークのエッジにあるルータです。このようなアクセスリストは、外部ネットワークから、または内部ネットワークのあまり制御されていない領域から、内部ネットワークの機密性が高い領域に対する基本的なバッファとして機能します。このような境界ルータでは、ルータインターフェイスに設定されている各ネットワーク プロトコルに合わせてアクセス リストを設定する必要があります。インバウンド トラフィック、アウトバウンド トラフィック、またはその両方がインターフェイスでフィルタされるように、アクセス リストを設定できます。

アクセス リストは個々のプロトコル ベースで定義されます。つまり、各プロトコルのトラフィック フローを制御する場合、インターフェイスでイネーブルにするプロトコルごとにアクセス リストを定義する必要があります。

# アクセス リストの定義

アクセス コントロール リスト（ACL）は、ネットワークを通過するパケットの動きを制御するためにパケット フィルタリングを実行します。パケット フィルタリングは、ネットワークへのトラフィックのアクセスを限定し、ユーザおよびデバイスのネットワークに対するアクセスを制限し、トラフィックがネットワークから流出するのを防ぐことで、セキュリティを提供します。IPアクセスリストによって、スプーフィングやサービス妨害攻撃の可能性を軽減し、ファイアウォールを介した動的で一時的なユーザ アクセスが可能になります。

また、IP アクセス リストは、セキュリティ以外の用途にも使用できます。たとえば、帯域幅制御、ルーティングアップデートのコンテンツの制限、ルートの再配布、ダイヤルオンデマンド（DDR）呼び出しのトリガー、デバッグ出力の制限、Quality of Service（QoS）機能のトラフィックの識別と分類などです。

アクセス リストは、少なくとも 1 つの **permit** ステートメント、および任意の 1 つまたは複数の **deny** ステートメントで構成される順次リストです。IP アクセス リストの場合、これらのステートメントはIP アドレス、上位層の IP プロトコルなどの IP パケットのフィールドに適用できます。

アクセスリストは名前または番号で識別および参照されます。アクセスリストはパケットフィルタとして動作し、各アクセスリストに定義されている条件に基づいてパケットがフィルタされます。

アクセスリストを構成した後でアクセスリストを有効にするには、アクセスリストをインターフェイスに適用するか（**ip access-group** コマンドを使用）、vty に適用するか（**access-class** コマンドを使用）、またはアクセス リストを許容するあらゆるコマンドでアクセス リストを参照する必要があります。複数のコマンドから同じアクセス リストを参照できます。

次の構成では、branchoffices という名前の IP アクセス リストがファスト イーサネット インターフェイス 0/1/0 上で構成され、着信パケットに適用されます。発信元アドレスとマスクのペアで指定されているネットワーク以外は、ファスト イーサネット インターフェイス 0/1/0 にアクセスできません。ネットワーク 172.16.7.0 上の送信元から発信されるパケットの宛先に、制限はありません。ネットワーク 172.16.2.0 上の送信元から発信されるパケットの宛先は、172.31.5.4 にする必要があります。

```
ip access-list extended branchoffices
 10 permit 172.16.7.0 0.0.0.3 any
 20 permit 172.16.2.0 0.0.0.255 host 172.31.5.4
!
interface fastethernet 0/1/0
 ip access-group branchoffices in
```

# アクセス リストのルール

アクセス リストには、次のルールが適用されます。

- 1 つのインターフェイス、1 つのプロトコル、1 つの方向につき、許可されるアクセス リストは 1 つだけです。

- アクセス リストには少なくとも 1 つの **permit** ステートメントが含まれる必要があります。そうしないと、ネットワークに入るすべてのパケットが拒否されます。
- アクセス リスト条件または一致基準の構成順序は重要です。パケットを転送するかブロックするかを決定するときに、Cisco ソフトウェアは、それぞれの条件ステートメントに対してステートメントの作成順にパケットをテストします。一致が見つかると、条件ステートメントはそれ以上チェックされません。同じ **permit** ステートメントまたは **deny** ステートメントでも、順序が異なる場合、ある状況では通過し、別の状況では拒否されるパケットが生じる可能性があります。
- アクセス リストを名前によって参照したときに、そのアクセス リストが存在しない場合は、すべてのパケットが通過します。インターフェイスまたはコマンドに空のアクセス リストを適用すると、ネットワークに対するすべてのトラフィックが許可されます。
- 標準のアクセス リストと拡張のアクセス リストの名前は同じにできません。
- パケットが発信インターフェイスにルーティングされる前に、着信アクセス リストがパケットを処理します。ネットワークへのパケット アクセスを拒否するフィルタ条件がある着信アクセス リストは、ルーティング ルックアップ時のオーバーヘッドを削減します。構成されたフィルタ基準に基づいてネットワークへのアクセスを許可されたパケットはルーティング処理されます。インバウンド アクセス リストの場合、**permit** ステートメントを構成するとパケットは受信後に処理され、**deny** ステートメントを構成するとパケットは破棄されます。
- 発信アクセス リストの場合、パケットの処理後にデバイスから送信されます。着信パケットは発信インターフェイスにルーティングされてから、発信アクセス リストで処理されます。アウトバウンド アクセス リストの場合、**permit** ステートメントを構成するとパケットは出力バッファに送信され、**deny** ステートメントを構成するとパケットは破棄されます。
- アクセス リストで、デバイスに到達するトラフィック、またはデバイス経由で送信されるトラフィックは制御できますが、デバイスが送信元のトラフィックは制御できません。

# IP アクセス リストを作成する際に役立つヒント

意図しない結果を回避し、より効率的で有効なアクセス リストを作成するために役立つヒントを紹介します。

- アクセス リストを作成してから、インターフェイス（または別の対象）に適用します。その理由は、存在しないアクセス リストをインターフェイスに適用してから、アクセス リストを設定すると、最初のステートメントが有効になり、それに続く暗黙的な **deny** ステートメントによってアクセスに緊急の問題が発生するおそれがあるためです。
- アクセス リストを設定してから適用するもう 1 つの理由は、空のアクセス リストが適用されたインターフェイスはすべてのトラフィックを許可するためです。
- すべてのアクセス リストには、少なくとも 1 つの **permit** ステートメントが必要です。permit がないと、すべてのパケットは拒否され、トラフィックはまったく通過しません。

- まだ拒否されていないその他のパケットすべてを許可する場合、ステートメント **permit any any** を使用します。ステートメント **permit any any** を使用すると、実質的に、アクセス リストの末尾にある暗黙的な deny ステートメントでその他すべてのパケットが拒否されることを防ぎます。最初のアクセス リスト エントリは **permit any any** にしないでください。すべてのトラフィックが通過し、以降のテストに到達するパケットがなくなります。**permit any any** を指定すると、まだ拒否されていないすべてのトラフィックが通過します。
- すべてのアクセス リストは暗黙的な **deny** ステートメントで終了しますが、明示的な **deny** ステートメント（たとえば **deny ip any any**）の使用を推奨します。ほとんどのプラットフォームでは、**show access-list** コマンドを発行して拒否されるパケット数を表示し、アクセス リストが許可していないパケットに関する詳細情報を調査できます。明示的な **deny** ステートメントで拒否されたパケットのみがカウントされます。これは、明示的な **deny** ステートメントによって、より詳細なデータが生成されるためです。
- アクセス リストの作成中、または作成後に、エントリを削除する場合があります。
  - 番号付きアクセス リストからはエントリを削除できません。削除しようとすると、アクセス リスト全体が削除されます。エントリを削除する必要がある場合、アクセス リスト全体を削除してから最初から作り直す必要があります。
  - 名前付きアクセス リストからはエントリを削除できます。**no permit** または **no deny** コマンドを使用すると、適切なエントリが削除されます。
- 個々のステートメントの用途をひと目で確認および理解しやすくするために、**remark** コマンドを使用して、ステートメントの前後に役立つ注記を書き込むことができます。
- 特定のホストまたはネットワークに対するアクセスを拒否し、そのネットワークまたはホストの誰かがアクセスしようとしたかどうかを検出する場合、対応する **deny** ステートメントを指定した **log** キーワードを含めます。それによって、その送信元からの拒否されたパケットがログに記録されます。
- このヒントは、アクセス リストの配置に適用されます。リソースを保存しようとすると、着信アクセス リストでは常にフィルタ条件を適用した後に、ルーティング テーブルの検索を行います。発信アクセス リストではフィルタ条件を適用する前に、ルーティング テーブルの検索を行います。

# 名前付きまたは番号付きアクセス リスト

すべてのアクセス リストは、名前または番号で識別されます。名前付きアクセス リストは、番号付きアクセス リストよりも便利です。タスクを思いだしやすく関連性がある、わかりやすい名前を指定できるためです。名前付きアクセス リストでは、ステートメントの順序を変更したり、ステートメントを追加したりできます。

名前付きアクセス リストは、番号付きアクセス リストではサポートされない次の機能をサポートします。

- IP オプションのフィルタリング

- 非隣接ポート
- TCP フラグ フィルタリング
- **no permit** または **no deny** コマンドによるエントリの削除

（注） 番号付きアクセス リストを受け入れるコマンドの中には、名前付きアクセス リストを受け入れないコマンドがあります。たとえば、vty には番号付きアクセス リストだけを使用します。

# 標準または拡張アクセス リスト

すべてのアクセス リストは、標準または、拡張アクセス リストのいずれかになります。送信元アドレスでフィルタするだけの場合、より簡易な標準アクセスリストで十分です。送信元アドレス以外のアドレスをフィルタする場合、拡張アクセス リストが必要です。

- 名前付きアクセス リストは、**ip access-list** コマンド構文のキーワード **standard** または **extended** に基づいて標準か拡張かが決まります。
- 番号付きアクセス リストは、**access-list** コマンド構文の番号に基づいて標準か拡張かが決まります。標準 IP アクセス リストには 1 ～ 99 または 1300 ～ 1999 の番号が付けられ、拡張 IP アクセス リストには 100 ～ 199 または 2000 ～ 2699 の番号が付けられます。標準 IP アクセス リストの範囲は、当初は 1 ～ 99 のみでしたが、1300 ～ 1999 の範囲に拡張されました（間の番号は、他のプロトコルに割り当てられました）。拡張アクセスリストの範囲も同様に拡張されました。

### 標準アクセス リスト

標準アクセス リストは、パケットの送信元アドレスのみをテストします（ただし 2 つの例外があります）。標準アクセスリストは送信元アドレスをテストするため、宛先の近くでトラフィックをブロックする際には効率的です。標準アクセスリストのアドレスが送信元アドレスではない例外が 2 つあります。

- アウトバウンド VTY アクセス リストでは、誰かが Telnet を実行しようとすると、アクセス リスト エントリのアドレスは、送信元アドレスではなく宛先アドレスとして使用されます。
- ルートをフィルタする場合、送信元アドレスではなくアドバタイズされたネットワークがフィルタされます。

### 拡張アクセス リスト

拡張アクセス リストは、任意の場所のトラフィックをブロックするために適しています。拡張アクセス リストは、送信元アドレス、宛先アドレス、およびその他の IP パケット データをテストします。たとえば、プロトコル、TCP または UDP ポート番号、タイプ オブ サービス

（ToS）、優先順位、TCP フラグ、IP オプションなどです。また、拡張アクセス リストには、次のように標準アクセス リストにはない機能があります。

- IP オプションのフィルタリング
- TCP フラグのフィルタリング
- パケットの非初期フラグメントのフィルタリング（「Refining an IP Access List」モジュールを参照してください）

（注） 拡張アクセス リストの対象となるパケットは、自律的に切り替えられません。

# アクセスを制御するためにフィルタできる IP パケット フィールド

拡張アクセスリストを使用すると、IP パケットに含まれる次の任意のフィールドについてフィルタできます。送信元アドレスおよび宛先アドレスは、アクセスリストの基礎として最もよく指定される 2 つのフィールドです。

- 送信元アドレス - 特定のネットワーキング デバイスまたはホストから送信されるパケットを制御するために、送信元アドレスを指定します。
- 宛先アドレス - 特定のネットワーキング デバイスまたはホストに対して送信されるパケットを制御するために、宛先アドレスを指定します。
- プロトコル - キーワード **eigrp**、**gre**、**icmp**、**igmp**、**ip**、**ipinip**、**nos**、**ospf**、**tcp**、または **udp** で示される IP プロトコル、あるいは 0 ～ 255 の範囲の整数（インターネット プロトコルを示す）で示される IP プロトコルを指定します。トランスポート層プロトコル（**icmp**、**igmp**、**tcp**、または **udp**）を指定すると、コマンドは固有の構文になります。
  - ポートおよび非隣接ポート - ポート名またはポート番号で TCP または UDP ポートを指定します。ポート番号に非隣接ポート番号は指定できません。ポート番号は、Telnet トラフィックや HTTP トラフィックなどをフィルタする際に有効です。
  - TCP フラグ - TCP パケットに設定された任意のフラグまたはすべてのフラグにパケットが一致することを指定します。特定のフラグについてフィルタすることで、不正な同期パケットを回避できます。
- IP オプション - IP オプションを指定します。IP オプションに基づいてフィルタする理由の 1 つは、IP オプションを含む偽造パケットでルータが飽和状態にならないようにするためです。

# アクセス リストのアドレスに対するワイルドカード マスク

アドレス フィルタリングでは、アクセス リスト エントリ内のアドレス ビットとアクセス リストに送信されるパケットを比較するときに、対応する IP アドレスを確認するか無視するかをソフトウェアに示すために、ワイルドカード マスクを使用します。注意してワイルドカード

マスクを設定することで、許可または拒否テストのために 1 つまたは複数の IP アドレスを指定できます。

IP アドレス ビット用のワイルドカード マスクでは、数値 1 と数値 0 を使用して、対応する IP アドレス ビットをどのように扱うかを指定します。1 と 0 は、サブネット（ネットワーク）マスクで意味する内容が対照的なため、ワイルドカード マスクは逆マスクとも呼ばれます。

- ワイルドカード マスク ビット 0 は、対応するビット値を確認することを示します。ビット値は一致する必要があります。
- ワイルドカード マスク ビット 1 は、対応するビット値を無視することを示します。ビット値が一致する必要はありません。

アクセス リスト ステートメントの送信元アドレスまたは宛先アドレスでワイルドカード マスクを指定しない場合、0.0.0.0（すべての値が一致する必要があることを示します）という暗黙的なワイルドカード マスクが想定されます。

サブネット マスクでは、ネットワークとサブネットを示す隣接ビットをマスクにする必要がありますが、それとは異なり、ワイルドカード マスクではマスクに非隣接ビットを使用できます。

次の表に、アクセス リストの IP アドレスおよびマスクと、それに一致すると見なされる対応するアドレスの例を示します。

**表 *1 : IP* アドレス、ワイルドカード マスク、および一致する結果の例**

| アドレス | ワイルドカード マスク | 一致する結果 |
|---|---|---|
| 0.0.0.0 | 255.255.255.255 | すべてのアドレスはアクセス リスト条件に一致します |
| 172.18.0.0/16 | 0.0.255.255 | ネットワーク 172.18.0.0 |
| 172.18.5.2/16 | 0.0.0.0 | ホスト 172.18.5.2 のみが一致します |
| 172.18.8.0 | 0.0.0.7 | サブネット 172.18.8.0/29 のみが一致します |
| 172.18.8.8 | 0.0.0.7 | サブネット 172.18.8.8/29 のみが一致します |
| 172.18.8.15 | 0.0.0.3 | サブネット 172.18.8.15/30 のみが一致します |
| 10.1.2.0 | 0.0.252.255（マスクの非隣接ビット） | 10.1.2.0 ～ 10.1.254.0 に含まれる偶数のネットワークに一致します |

# アクセス リストのシーケンス番号

IP アクセス リスト エントリにシーケンス番号を適用する機能によって、アクセス リストの変更が簡易になります。IP アクセス リスト エントリ シーケンス番号機能の前には、アクセス リスト内のエントリの位置を指定する方法はありませんでした。以前は、既存のリストの途中にエントリを挿入する場合、目的の位置の後にあるすべてのエントリを削除してから、新しいエ

ントリを追加し、削除したすべてのエントリを再入力する必要がありました。これは手間がかかり、エラーが起こりやすい方法です。

この新しい機能を使用すると、アクセス リスト エントリにシーケンス番号を追加し、順序を変更することができます。新しいエントリを追加する場合、アクセスリストの目的の位置に挿入されるようにシーケンス番号を指定します。必要に応じて、アクセスリストの現在のエントリを並べ替えて、新しいエントリを挿入できる場所を作成できます。

# アクセス リストのロギング

Cisco IOS ソフトウェアには、単一の標準または拡張 IP アクセス リスト エントリで許可または拒否されたパケットに関するロギング メッセージ機能があります。つまり、パケットがエントリに一致する場合は常に、パケットに関する情報を提供するロギング メッセージがコンソールに送信されます。コンソールにロギングするメッセージのレベルは、**logging console** グローバル コンフィギュレーション コマンドで制御します。

アクセス リスト エントリをトリガーする最初のパケットによって、即時にロギング メッセージが作成され、表示またはロギングされるまで、以降のパケットは5分間隔で収集されます。ログ メッセージにはアクセス リスト番号、パケットの許可または拒否に関する状況、パケットの送信元 IP アドレス、および直前の 5 分間に許可または拒否された送信元からのパケット数が示されます。

ただし、**ip access-list log-update** コマンドを使用して、アクセス リストに一致する場合（さらに許可または拒否される場合）に、システムでログ メッセージを生成するパケットの数を設定できます。この手順を実行するのは、5分間隔よりも短い頻度でログ メッセージを受信する場合です。

**注意** *number-of-matches* 引数を 1 に設定すると、ログ メッセージはキャッシングされずにただちに送信されます。この場合、アクセス リストに一致するパケットごとにログ メッセージが発生します。大量のログ メッセージでシステムが過負荷になる可能性があるため、1に設定することは推奨されません。

**ip access-list log-update** コマンドを使用する場合でも、5 分タイマーは有効なままなので、各キャッシュのメッセージ数に関係なく、5 分が経過すると各キャッシュは空になります。ログ メッセージを送信するタイミングに関係なく、しきい値が指定されていない場合と同様に、ログ メッセージのキャッシュは消去され、カウントは 0 にリセットされます。

（注） ロギング メッセージが多すぎて処理できない場合、または1秒以内に処理する必要があるロギング メッセージが複数ある場合、ロギング設備ではロギング メッセージ パケットの一部をドロップすることがあります。この動作によって、ロギング パケットが多すぎてルータがクラッシュすることを回避します。そのため、課金ツールや、アクセス リストと一致する数の正確な情報源としてロギング設備をを使用しないでください。

## アクセス リスト ロギングの代替方法

ログ オプションを使用した ACL 内のエントリのパケット マッチングは代替のプロセスです。ACL でログ オプションを使用することは推奨されません。Null0 の宛先インターフェイスで NetFlow エクスポートおよびマッチングを使用することを推奨します。これは CEF パスで実行されます。Null0 の宛先インターフェイスは、ACL によってドロップされるすべてのパケット用に設定されます。

# その他の IP アクセス リスト機能

標準または拡張アクセス リストを作成する基本手順以外に、次のようにアクセス リストを強化できます。これらの各方法の詳細については、「Refining an Access List」モジュールを参照してください。

- 拡張アクセス リストの **permit** ステートメントまたは **deny** ステートメントを有効にする日時を指定し、アクセス リストを細かくし、絶対的または定期的な期間に限定することができます。
- 名前付きアクセス リストの作成後は、エントリを追加したり、エントリの順序を変更したりできます（これはアクセス リストのシーケンス番号再割り当てとも呼ばれます）。
- パケットの非初期フラグメントについてフィルタすることで、パケットをフィルタするときにより細かい精度を達成できます。

# RSP3 ポートの関連情報

発信アクセス リストは、RSP3 ではサポートされていません。

# アクセス リストを適用する場所

アクセス リストは、デバイスの着信または発信インターフェイスに適用できます。アクセス リストを着信インターフェイスに適用すると、インターフェイスで着信するトラフィックが制御され、アクセス リストを発信インターフェイスに適用すると、インターフェイスから発信されるトラフィックが制御されます。

ソフトウェアは、着信インターフェイスでパケットを受信すると、アクセス リストで設定されているステートメントに対してパケットを検査します。アクセス リストがアドレスを許可している場合は、ソフトウェアはパケットを処理します。着信パケットをフィルタリングするためにアクセス リストを適用すると、フィルタリングされたパケットはデバイスに到達する前に廃棄されるため、デバイスのリソースを節約できます。

発信インターフェイスでは、アクセス リストはインターフェイスから転送（送信）されたパケットをフィルタリングします。発信インターフェイスで Rate-Based Satellite Control Protocol（RBSCP）の TCP アクセス コントロール リスト（ACL）を使用して、発信インターフェイスで TCP 確認応答（ACK）を受けるパケットの種類を制御できます。

**debug** コマンドを使用してアクセス リストを参照し、デバッグ ログの量を制限できます。たとえば、アクセス リストのフィルタリング基準または一致基準に基づいて、デバッグ ログを送信元または宛先のアドレスまたはプロトコルに制限できます。

アクセス リストを使用して、ルーティング アップデート、ダイヤルオンデマンド（DDR）、および Quality of Service（QoS）機能を制御することができます。

# その他の参考資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Commands List, All Releases』 |
| IP アクセス リスト コマンド：コマンド構文、コマンド モード、コマンド履歴、デフォルト設定、使用に関する注意事項および例 | 『Cisco IOS IP Addressing Services Command Reference』 |
| 送信元アドレス、宛先アドレス、またはプロトコルに基づくフィルタリング | 『ICreating an IP Access List and Applying It to an Interface』モジュール |
| IP オプション、TCP フラグ、非隣接ポート、または TTL に基づくフィルタリング | 『Creating an IP Access List to Filter IP Options, TCP Flags, or Noncontiguous Ports』モジュール |

### 標準

| 標準と RFC | タイトル |
|---|---|
| なし | — |

### MIB

| MIB | MIB のリンク |
|---|---|
| なし | 選択したプラットフォーム、Cisco IOS XE ソフトウェア リリース、およびフィーチャ セットの MIB の場所を検索しダウンロードするには、次の URL にある Cisco MIB Locator を使用します。<br>http://www.cisco.com/go/mibs |

### シスコのテクニカル サポート

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

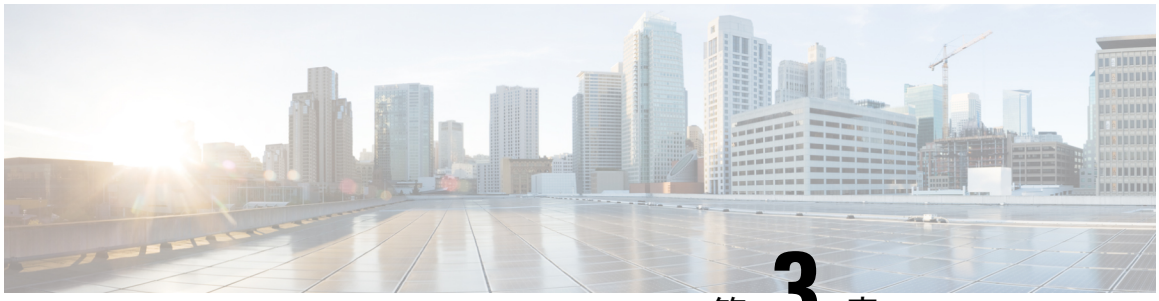
# IP アクセス リストに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *2 : IP* アクセス リストに関する機能情報**

| 機能名 | リリース | 機能の設定情報 |
|---|---|---|
| ACL - IP プロトコル | Cisco IOS XE リリース 3.16 | Cisco IOS XE リリース 3.16 では、Cisco ASR 903 ルータのサポートが追加されました。 |



第 3 章

# IP アクセス リストの作成とインターフェイスへの適用

IP アクセス リストには、ネットワークを保護し、Quality of Service（QoS）係数の設定や **debug** コマンド出力の制限などのセキュリティ以外の目標を達成する際に多数の利点があります。ここでは、標準、拡張、名前付き、および番号付き IP アクセス リストの作成方法について説明します。アクセス リストは、名前または番号で参照できます。標準アクセス リストは、IP パケットの送信元アドレスのみに基づいてフィルタできます。拡張アクセス リストは、IP パケットの送信元アドレス、宛先アドレス、および他のフィールドに基づいてフィルタできます。

アクセス リストの作成後に有効にするには、何かに適用する必要があります。このモジュールでは、アクセス リストをインターフェイスに適用する方法について説明します。ただし、アクセス リストにはその他にも多数の用途があり、このモジュールで言及していますが、他のモジュールでも説明しています。多様なテクノロジーについては、他のコンフィギュレーション ガイドを参照してください。



## 機能情報の確認

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

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

# IPアクセスリストの作成およびインターフェイスへの適用の制限

IPv4 および IPv6 アクセス コントロール リスト（ACL）を設定する場合、次の制限事項が適用されます。

- Application Control Engine（ACE）固有のカウンタは、サポートされていません。
- レイヤ 3 IPv4 および Ipv6 ACL は、同じインターフェイスではサポートされません。
- レイヤ 3 Ipv4 または IPv6 ACL が適用されているイーサネット フローポイント（EFP）またはトランク EFP インターフェイスでは、MAC ACL はサポートされていません。
- ACL あたり最大 500 の ACE がサポートされます。
- IPv4 および IPv6 ACL は、EFP インターフェイスでは現在サポートされていません。IPv4 および IPv6 ACL は、物理インターフェイス、ブリッジ ドメイン インターフェイスおよびポート チャネル インターフェイスでサポートされています。
- レイヤ 4 ポートの範囲と機能は、Ternary Content Addressable Memory（TCAM）に展開されます。IPv4 ACL によって、レイヤ 1K TCAM に制限され、レイヤ 2 ACL スケールは、1K TCAM エントリに制限されます。
- ACL カウンタまたは統計情報は、Cisco ASR 900 RSP3 モジュールではサポートされていません。
- オブジェクト グループは、IP ACL ではサポートされていません。
- アウトバウンド ACL は、Cisco ASR 900 RSP3 モジュールではサポートされていません。

# IPアクセスリストの作成とインターフェイスへの適用に関する情報

## IP アクセス リストを作成する際に役立つヒント

- アクセスリストを作成してから、インターフェイス（または別の対象）に適用します。その理由は、存在しないアクセス リストをインターフェイスに適用してから、アクセス リストを設定すると、最初のステートメントが有効になり、それに続く暗黙的な **deny** ステートメントによってアクセスに緊急の問題が発生するおそれがあるためです。

- アクセス リストを設定してから適用するもう 1 つの理由は、空のアクセス リストが適用されたインターフェイスはすべてのトラフィックを許可するためです。
- すべてのアクセス リストには、少なくとも 1 つの **permit** ステートメントが必要です。permit がないと、すべてのパケットは拒否され、トラフィックはまったく通過しません。
- ネットワークまたはサブネットのより具体的な参照が、より全般的な参照よりも前に出現するように、アクセス リストを構成します。
- パケットは、ACL の最初の ACE に一致します。したがって、**permit ip any any** はすべてのパケットに一致し、以降の ACE はすべて無視されます。
- すべてのアクセス リストは暗黙的な **deny** ステートメントで終了しますが、明示的な **deny** ステートメント（たとえば **deny ip any any**）の使用を推奨します。ほとんどのプラットフォームでは、**show access-list** コマンドを発行して拒否されるパケット数を表示し、アクセス リストが許可していないパケットに関する詳細情報を調査できます。明示的な **deny** ステートメントで拒否されたパケットのみがカウントされます。これは、明示的な **deny** ステートメントによって、より詳細なデータが生成されるためです。
- アクセスリストの作成中、または作成後に、エントリを削除する場合があります。名前付きアクセス リストからはエントリを削除できます。**no permit** または **no deny** コマンドを使用すると、適切なエントリが削除されます。
- 個々のステートメントの用途をひと目で確認および理解しやすくするために、**remark** コマンドを使用して、ステートメントの前後に役立つ注記を書き込むことができます。
- 特定のホストまたはネットワークに対するアクセスを拒否し、そのネットワークまたはホストの誰かがアクセスしようとしたかどうかを検出する場合、対応する **deny** ステートメントを指定した **log** キーワードを含めます。それによって、その送信元からの拒否されたパケットがログに記録されます。
- このヒントは、アクセスリストの配置に適用されます。リソースを保存しようとすると、着信アクセス リストでは常にフィルタ条件を適用した後に、ルーティング テーブルの検索を行います。発信アクセスリストではフィルタ条件を適用する前に、ルーティングテーブルの検索を行います。

# アクセス リストの注釈

任意の IP アクセスリストのエントリについて、コメントまたは注釈を含めることができます。アクセス リストの注釈は、アクセス リスト エントリの前後にあるオプションの注釈です。エントリの内容がわかるので、エントリの目的を解釈する必要はありません。各注釈の長さは 100 文字に制限されます。

コメントは、**permit** ステートメントまたは **deny** ステートメントの前後どちらにでも配置できます。注釈を追加する場所には一貫性があるようにしてください。注釈が関連する **permit** ステートメントや **deny** ステートメントの前にある場合と後にある場合とが混在すると、ユーザが混乱する可能性があります。

後続の **deny** ステートメントの機能を説明する注釈の例を次に示します。

```
ip access-list extended telnetting
 remark Do not allow host1 subnet to telnet out
 deny tcp host 172.16.2.88 any eq telnet
```

## その他の IP アクセス リスト機能

標準または拡張アクセス リストを作成する基本手順以外に、次のようにアクセス リストを強化できます。これらの各方法の詳細については、『*Refining an IP Access List module*』を参照してください。

- 拡張アクセス リストの **permit** ステートメントまたは **deny** ステートメントを有効にする日時を指定し、アクセスリストを細かくし、絶対的または定期的な期間に限定することができます。
- 名前付きまたは番号付きアクセス リストの作成後は、エントリを追加したり、エントリの順序を変更したりできます（これはアクセスリストのシーケンス番号再割り当てとも呼ばれます）。
- パケットの非初期フラグメントについてフィルタすることで、パケットをフィルタするときにより細かい精度を達成できます。

# IPアクセスリストの作成とインターフェイスへの適用方法

ここでは、名前または番号を使用して、標準または拡張アクセスリストを作成する一般的な方法について説明します。アクセスリストには高い柔軟性があります。この作業では、単純に1つの **permit** コマンドと 1 つの **deny** コマンドを使用して、それぞれのコマンド構文を指定します。あとは、必要な **permit** および **deny** コマンドの数とその順序を決めるだけです。

（注） このモジュールの最初の 2 つの作業として、1 つのアクセス リストを作成します。適切に機能するように、アクセス リストを適用する必要があります。インターフェイスにアクセス リストを適用する場合は、「インターフェイスへのアクセスリストの適用」タスクを実行します。

## 送信元アドレスに基づいてフィルタする標準アクセス リストの作成

送信元アドレスのみに基づいてフィルタする場合、簡易な標準アクセスリストで十分です。標準アクセス リストには名前付きと番号付きという 2 種類があります。名前付きアクセス リストを使用すると、番号よりも直感的な名前を使用してアクセスリストを特定できます。また、番号付きアクセス リストよりもサポートする機能が多数です。

# 送信元アドレスに基づいてフィルタする名前付きアクセス リストの作成

送信元アドレスのみに基づいてフィルタする必要がある場合、標準の名前付きアクセスリストを使用します。この作業では、1 つの **permit** ステートメントと 1 つの **deny** ステートメントを使用しますが、使用する実際のステートメントとその順序は、フィルタまたは許可する内容によって変わります。フィルタリングの目標を達成するように、**permit** および **deny** ステートメントを定義します。

**ステップ 1** **enable**

例：

```
Device> enable
```

特権 EXEC モードをイネーブルにします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

例：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **ip access-list standard** *name*

例：

```
Device(config)# ip access-list standard R&D
```

名前を使用して標準IPアクセス リストを定義し、標準名前付きアクセス リストのコンフィギュレーション モードを開始します。

**ステップ 4** **remark** *remark*

例：

```
Device(config-std-nacl)# remark deny Sales network
```

（任意）アクセス リスト エントリに関してユーザにわかりやすいコメントを追加します。

- 注釈はアクセス リスト エントリの前または後に指定できます。
- この例の注釈では、後続のエントリがインターフェイスに対するSalesネットワークのアクセスを拒否することをネットワーク管理者に示しています（このアクセス リストは後でインターフェイスに適用される想定です）。

**ステップ 5** **deny** {*source* [*source-wildcard*] | **any**} [**log**]

例：

```
Device(config-std-nacl)# deny 172.16.0.0 0.0.255.255 log
```

（任意）送信元アドレスおよびワイルドカード マスクに基づいて、指定した送信元を拒否します。

- *source-wildcard* を省略すると、0.0.0.0 というワイルドカード マスクが想定されます（つまり、すべての送信元アドレスに一致します）。
- 必要に応じて、*source source-wildcard* の代わりに、キーワード **any** を使用して、送信元と 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。
- この例では、ネットワーク 172.16.0.0 のすべてのホストは、アクセス リストへの合格が拒否されます。
- この例では、送信元アドレスを明示的に拒否し、**log** キーワードを指定しているため、その送信元からのパケットが拒否されるとロギングされます。これは、ネットワークまたはホスト上の誰かがアクセスしようとしたことを通知する方法の 1 つです。

**ステップ 6** **remark** *remark*

**例**：

```
Device(config-std-nacl)# remark Give access to Tester's host
```

（任意）アクセス リスト エントリに関してユーザにわかりやすいコメントを追加します。

- 注釈はアクセス リスト エントリの前または後に指定できます。
- この注釈は、後続のエントリがインターフェイスに対する Tester のホスト アクセスを許可することをネットワーク管理者に示します。

**ステップ 7** **permit** {*source* [*source-wildcard*] | **any**} [**log**]

**例**：

```
Device(config-std-nacl)# permit 172.18.5.22 0.0.0.0
```

送信元アドレスおよびワイルドカード マスクに基づいて、指定した送信元を許可します。

- 各アクセス リストには、少なくとも 1 つの **permit** ステートメントが必要です。ただし、最初のエントリにする必要はありません。
- *source-wildcard* を省略すると、0.0.0.0 というワイルドカード マスクが想定されます（つまり、すべての送信元アドレスに一致します）。
- 必要に応じて、*source source-wildcard* の代わりに、キーワード **any** を使用して、送信元と 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。
- この例では、ホスト 172.18.5.22 がアクセス リストに合格できます。

**ステップ 8** アクセス リストの基礎とする送信元の指定が完了するまで、ステップ 4 ～ 7 の手順を繰り返します。

明示的に許可されていないすべての送信元は、アクセス リストの末尾にある暗黙的な **deny** ステートメントで拒否されます。

**ステップ 9** **end**

**例**：

```
Device(config-std-nacl)# end
```

標準の名前付きアクセス リスト コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

**ステップ 10** **show ip access-list**

**例**：

```
Device# show ip access-list
```

（任意）現在の IP アクセス リストすべてのコンテンツが表示されます。

## 送信元アドレスに基づいてフィルタする番号付きアクセス リストの作成

送信元アドレスのみに基づいてフィルタする必要があり、名前付きアクセス リストを使用しない場合、標準の番号付きアクセス リストを設定します。

IP 標準アクセス リストには、1 ～ 99 または 1300 ～ 1999 の番号を付けます。この作業では、1 つの **permit** ステートメントと 1 つの **deny** ステートメントを使用しますが、使用する実際のステートメントとその順序は、フィルタまたは許可する内容によって変わります。フィルタリングの目標を達成するように、**permit** および **deny** ステートメントを定義します。

**ステップ 1** **enable**

**例**：

```
Device> enable
```

特権 EXEC モードをイネーブルにします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

**例**：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **access-list** *access-list-number permit* {*source* [*source-wildcard*] | **any**} [**log**]

**例**：

```
Device(config)# access-list 1 permit 172.16.5.22 0.0.0.0
```

送信元アドレスおよびワイルドカード マスクに基づいて、指定した送信元を許可します。

- 各アクセス リストには、少なくとも 1 つの permit ステートメントが必要です。ただし、最初のエントリにする必要はありません。
- 標準 IP アクセス リストには、1 ～ 99 または 1300 ～ 1999 の番号を付けます。
- source-wildcard を省略すると、0.0.0.0 というワイルドカード マスクが想定されます（つまり、すべての送信元アドレスに一致します）。
- 必要に応じて、source source-wildcard の代わりに、キーワード any を使用して、送信元と 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。
- この例では、ホスト 172.16.5.22 がアクセス リストに合格できます。

**ステップ 4** **access-list** *access-list-number* **deny** {*source* [*source-wildcard*] | **any**} [**log**]

**例**：

```
Device(config)# access-list 1 deny 172.16.7.34 0.0.0.0
```

送信元アドレスおよびワイルドカード マスクに基づいて、指定した送信元を拒否します。

- *source-wildcard* を省略すると、0.0.0.0 というワイルドカード マスクが想定されます（つまり、すべての送信元アドレスに一致します）。
- 必要に応じて、*source source-wildcard* の代わりに、省略形 **any** を使用して、送信元と 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。
- この例では、ホスト 172.16.7.34 はアクセス リストへの合格が拒否されます。

**ステップ 5** アクセス リストの基礎とする送信元の指定が完了するまで、ステップ 3 ～ 6 の手順を繰り返します。

明示的に許可されていないすべての送信元は、アクセス リストの末尾にある暗黙的な **deny** ステートメントで拒否されます。

**ステップ 6** **end**

**例**：

```
Device(config)# end
```

グローバル コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

**ステップ 7** **show ip access-list**

**例**：

```
Device# show ip access-list
```

（任意）現在の IP アクセス リストすべてのコンテンツが表示されます。

# 拡張アクセス リストの作成

送信元アドレス以外の要素に基づいてフィルタする場合、拡張アクセスリストを作成する必要があります。拡張アクセスリストには名前付きと番号付きという2種類があります。名前付きアクセス リストを使用すると、番号よりも直感的な名前を使用してアクセス リストを特定できます。また、サポートする機能が多数です。

送信元アドレスまたは宛先アドレス以外の要素をフィルタする方法の詳細については、コマンド リファレンス マニュアルの構文の説明を参照してください。

## 名前付き拡張アクセス リストの作成

送信元アドレス、宛先アドレス、またはアドレスと他の IP フィールドの組み合わせをフィルタする場合、名前付き拡張アクセス リストを作成します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *name*
4. **deny** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
5. **permit** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
6. アクセス リストの基礎とするフィールドと値の指定が完了するまで、ステップ 4 ～ 7 の手順を繰り返します。
7. **end**
8. **show ip access-list**

### 手順の詳細

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

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 3 | **ip access-list extended** *name*<br>例：<br>`Device(config)# ip access-list extended acl1` | 名前を使用して拡張 IP アクセス リストを定義し、拡張名前付きアクセス リストのコンフィギュレーション モードを開始します。 |
| ステップ 4 | **deny** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** \| **log-input**] [**time-range** *time-range-name*] [**fragments**]<br>例：<br>`Device(config-ext-nacl)# deny ip 172.18.0.0 0.0.255.255 host 172.16.40.10 log` | （任意）ステートメントに指定されたすべての条件に一致するすべてのパケットを拒否します。<br>• *source-wildcard* または *destination-wildcard* を省略すると、0.0.0.0 のワイルドカードマスクが想定されます。つまり、それぞれ送信元アドレスまたは宛先アドレスの全ビットへの一致を意味します。<br>• 必要に応じて、*source source-wildcard* または *destination destination-wildcard* の代わりに、キーワード **any** を使用して、アドレスと 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。<br>• 必要に応じて、キーワード **host** *source* を使用し、*source* 0.0.0.0 の送信元と送信元ワイルドカードを表示して、省略形 **host** *destination* を使用し、*destination* 0.0.0.0 の宛先と宛先ワイルドカードを表示します。<br>• この例では、すべての送信元のパケットは、宛先ネットワーク 172.18.0.0 へのアクセスが拒否されます。アクセス リストによって許可または拒否されるパケットに関するロギング メッセージは、**logging facility** コマンドに設定された設備に送信されます（たとえば、コンソール、端末、syslog）。つまり、パケットがアクセス リストに一致する場合は常に、パケットに関する情報を提供するロギング メッセージが設定された設備に送信されます。コンソールにロギングするメッセージのレベルは、**logging console** コマンドで制御します。 |
| ステップ 5 | **permit** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** \| **log-input**] [**time-range** *time-range-name*] [**fragments**]<br>例： | ステートメントに指定されたすべての条件に一致するすべてのパケットを許可します。<br>• 各アクセス リストには、少なくとも 1 つの permit ステートメントが必要です。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config-ext-nacl)# permit tcp any any` | • *source-wildcard* または *destination-wildcard* を省略すると、0.0.0.0 のワイルドカードマスクが想定されます。つまり、それぞれ送信元アドレスまたは宛先アドレスの全ビットへの一致を意味します。<br>• 必要に応じて、*source source-wildcard* または *destination destination-wildcard* の代わりに、キーワード **any** を使用して、アドレスと 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。<br>• この例では、任意の送信元から任意の宛先への TCP パケットが許可されています。<br>• **log-input** キーワードを使用して、ロギング出力に入力インターフェイス、送信元 MAC アドレス、または仮想回線を含めます。 |
| **ステップ 6** | アクセス リストの基礎とするフィールドと値の指定が完了するまで、ステップ 4 ～ 7 の手順を繰り返します。 | 明示的に許可されていないすべての送信元は、アクセス リストの末尾にある暗黙的な **deny** ステートメントで拒否されます。 |
| **ステップ 7** | **end**<br>例：<br>`Device(config-ext-nacl)# end` | 標準の名前付きアクセス リスト コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。 |
| **ステップ 8** | **show ip access-list**<br>例：<br>`Device# show ip access-list` | （任意）現在の IP アクセス リストすべてのコンテンツが表示されます。 |

### RSP3 ポートの関連情報

ACL は、フラグメント化されたパケットに対してはサポートされていません。

## 番号付き拡張アクセス リストの作成

送信元アドレス、宛先アドレス、またはアドレスと他の IP フィールドの組み合わせに基づいてフィルタし、名前を使用しない場合、番号付き拡張アクセス リストを作成します。拡張 IP アクセス リストには、100 ～ 199 または 2000 ～ 2699 の番号を付けます。

### 手順の概要

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

3. **access-list** *access-list-number* **remark** *remark*
4. **access-list** *access-list-number* **permit** *protocol* {*source* [*source-wildcard*] | **any**} {*destination* [*destination-wildcard*] | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
5. **access-list** *access-list-number* **remark** *remark*
6. **access-list** *access-list-number* **deny** *protocol* {*source* [*source-wildcard*] | **any**} {*destination* [*destination-wildcard*] | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
7. アクセス リストの基礎とするフィールドと値の指定が完了するまで、ステップ 3 ～ 6 の手順を繰り返します。
8. **end**
9. **show ip access-list**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **access-list** *access-list-number* **remark** *remark*<br>例：<br>`Device(config)# access-list 107 remark allow Telnet packets from any source to network 172.69.0.0 (headquarters)` | （任意）アクセス リスト エントリに関してユーザにわかりやすいコメントを追加します。<br>• 最大 100 文字の注釈をアクセス リスト エントリの前または後に指定できます。 |
| ステップ 4 | **access-list** *access-list-number* **permit** *protocol* {*source* [*source-wildcard*] | **any**} {*destination* [*destination-wildcard*] | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]<br>例：<br>`Device(config)# access-list 107 permit tcp any 172.69.0.0 0.0.255.255 eq telnet` | ステートメントに指定されたすべての条件に一致するすべてのパケットを許可します。<br>• 各アクセス リストには、少なくとも 1 つの **permit** ステートメントが必要です。ただし、最初のエントリにする必要はありません。<br>• 拡張 IP アクセス リストには、100 ～ 199 または 2000 ～ 2699 の番号を付けます。<br>• *source-wildcard* または *destination-wildcard* を省略すると、0.0.0.0 のワイルドカードマスクが想定されます。つまり、それぞれ送信元アドレスまたは宛先アドレスの全ビットへの一致を意味します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • 必要に応じて、*source source-wildcard* または *destination destination-wildcard* の代わりに、キーワード **any** を使用して、アドレスと 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。<br>• TCP と他のプロトコルでは、その他の構文も使用できます。複雑な構文の場合、コマンド リファレンスの **access-list** コマンドを参照してください。 |
| **ステップ 5** | **access-list** *access-list-number* **remark** *remark*<br>**例**：<br>`Device(config)# access-list 107 remark deny all other TCP packets` | （任意）アクセス リスト エントリに関してユーザにわかりやすいコメントを追加します。<br>• 最大 100 文字の注釈をアクセス リスト エントリの前または後に指定できます。 |
| **ステップ 6** | **access-list** *access-list-number* **deny** *protocol* {*source* [*source-wildcard*] \| **any**} {*destination* [*destination-wildcard*] \| **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** \| **log-input**] [**time-range** *time-range-name*] [**fragments**]<br>**例**：<br>`Device(config)# access-list 107 deny tcp any any` | ステートメントに指定されたすべての条件に一致するすべてのパケットを拒否します。<br>• *source-wildcard* または *destination-wildcard* を省略すると、0.0.0.0 のワイルドカード マスクが想定されます。つまり、それぞれ送信元アドレスまたは宛先アドレスの全ビットへの一致を意味します。<br>• 必要に応じて、*source source-wildcard* または *destination destination-wildcard* の代わりに、キーワード **any** を使用して、アドレスと 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。 |
| **ステップ 7** | アクセス リストの基礎とするフィールドと値の指定が完了するまで、ステップ 3 ～ 6 の手順を繰り返します。 | 明示的に許可されていないすべての送信元は、アクセス リストの末尾にある暗黙的な **deny** ステートメントで拒否されます。 |
| **ステップ 8** | **end**<br>**例**：<br>`Device(config)# end` | グローバル コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。 |
| **ステップ 9** | **show ip access-list**<br>**例**：<br>`Device# show ip access-list` | （任意）現在の IP アクセス リストすべてのコンテンツが表示されます。 |

# インターフェイスへのアクセス リストの適用

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip access-group** {*access-list-number* | *access-list-name*} {**in** | **out**}
5. **end**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **interface** *type number*<br>例： | インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。 |
| ステップ 4 | **ip access-group** {*access-list-number* \| *access-list-name*} {**in** \| **out**}<br>例：<br>`Device(config-if)# ip access-group acl1 in` | 指定したアクセス リストをインバウンド インターフェイスに適用します。<br>• 送信元アドレスをフィルタリングするには、インバウンド インターフェイスにアクセス リストを適用します。 |
| ステップ 5 | **end**<br>例：<br>`Device(config-if)# end` | インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |

# IPアクセスリストの作成とインターフェイスへの適用に関する設定例

## 例：ホスト送信元アドレスでのフィルタリング

次の例では、user1に属するワークステーションがギガビットイーサネット0/0/0へのアクセスを許可され、user2 に属するワークステーションはアクセスを許可されていません。

```
interface gigabitethernet 0/0/0
 ip access-group workstations in
!
ip access-list standard workstations
 remark Permit only user1 workstation through
 permit 172.16.2.88
 remark Do not allow user2 workstation through
 deny 172.16.3.13
```

## 例：サブネット送信元アドレスでのフィルタリング

次の例では、user1サブネットは、gigabitethernetインターフェイス0/0/0へのアクセスが許可されていませんが、Main サブネットは、アクセスが許可されています。

```
interface gigabitethernet 0/0/0
 ip access-group prevention in
!
ip access-list standard prevention
 remark Do not allow user1 subnet through
 deny 172.22.0.0 0.0.255.255
 remark Allow Main subnet
 permit 172.25.0.0 0.0.255.255
```

## 例：送信元と宛先のアドレスおよびIPプロトコルでのフィルタリング

次の設定例は、2つのアクセスリストを持つインターフェイスを示します。一方のリストは発信パケット、もう一方のリストは着信パケットに適用されます。Internet-filter という標準アクセスリストは、送信元アドレスに基づいて発信パケットをフィルタします。インターフェイスから発信が許可されるパケットは、送信元が 172.16.3.4 である必要があります。

marketing-group という拡張アクセスリストは、着信パケットをフィルタします。このアクセスリストは、任意の送信元からネットワーク 172.26.0.0 への Telnet パケットを許可し、その他すべての TCP パケットを拒否します。また、ICMP パケットはすべて許可します。1024 未満のポート番号を使用する、任意の送信元からネットワーク 172.26.0.0 への UDP パケットは拒否します。最後に、このアクセス リストはその他すべての IP パケットを拒否し、そのエントリによって許可または拒否されるパケットのロギングを実行します。

```
interface gigabitethernet 0/0/0
```

```
 ip address 172.20.5.1 255.255.255.0
 ip access-group Internet-filter out
 ip access-group marketing-group in
!
ip access-list standard Internet-filter
 permit 172.16.3.4
ip access-list extended marketing-group
 permit tcp any 172.26.0.0 0.0.255.255 eq telnet
 deny tcp any any
 permit icmp any any
 deny udp any 172.26.0.0 0.0.255.255 lt 1024
 deny ip any any
```

# 例：番号付きアクセスリストを使用した送信元アドレスでのフィルタリング

次の例では、ネットワーク 10.0.0.0 は、クラス A ネットワークで、2 番目のオクテットでサブネットを指定します。つまり、サブネット マスクは255.255.0.0 です。ネットワーク 10.0.0.0 アドレスの 3 番目および 4 番目のオクテットで特定のホストを指定します。Cisco IOS XE ソフトウェアは、アクセス リスト 2 を使用して、サブネット 48 上の 1 つのアドレスを受け入れ、そのサブネット上のその他のアドレスはすべて拒否します。最後の行は、その他すべてのネットワーク 10.0.0.0 サブネット上のアドレスを受け入れることを示します。

```
interface gigabitethernet 0/0/0
 ip access-group 2 in
!
access-list 2 permit 10.48.0.3
access-list 2 deny 10.48.0.0  0.0.255.255
access-list 2 permit 10.0.0.0  0.255.255.255
```

# 例：サブネットへの Telnet アクセスの防止

次の例では、user1 サブネットは、ギガビット イーサネット インターフェイス 0/0/0 から Telnet にアクセスできません。

```
interface gigabitethernet 0/0/0
 ip access-group telnetting out
!
ip access-list extended telnetting
 remark Do not allow user1 subnet to telnet out
 deny tcp 172.20.0.0 0.0.255.255 any eq telnet
 remark Allow Top subnet to telnet out
 permit tcp 172.33.0.0 0.0.255.255 any eq telnet
```

# 例：ポート番号を使用した TCP および ICMP に基づくフィルタリング

次の例では、acl1 という名前の拡張アクセス リストの最初の行で、1023 よりも大きい宛先ポートを持つ着信 TCP 接続を許可しています。2 行目で、ホスト 172.28.1.2 の Simple Mail Transfer Protocol（SMTP）ポートへの着信 TCP 接続を許可しています。最後の行では、エラー フィードバックのための着信 ICMP メッセージを許可しています。

```
interface gigabitethernet 0/0/0
 ip access-group acl1 in
!
ip access-list extended acl1
 permit tcp any 172.28.0.0 0.0.255.255 gt 1023
 permit tcp any host 172.28.1.2 eq 25
 permit icmp any 172.28.0.0 255.255.255.255
```

# 例：SMTP 電子メールと確立済み TCP 接続の許可

インターネットに接続されているネットワークがあり、イーサネット上のホストでインターネット上の任意のホストに対して TCP 接続を構成するとします。ただし、専用のメール ホストのメール（SMTP）ポートを除き、IP ホストから gigabitethernet 上のホストに対する TCP 接続を構成できないようにします。

SMTP は、接続の一端では TCP ポート 25、もう一端ではランダムなポート番号を使用します。接続の存続中は、この同じ 2 つのポート番号が使用されます。インターネットから着信するメール パケットは、25 という宛先ポートを持ちます。発信パケットは、ポート番号が予約されています。ルータの背後にあるセキュア システムは、ポート 25 でメール接続を常に受け入れるため、着信および発信サービスを個別に制御できます。発信インターフェイスまたは着信インターフェイスで、アクセス リストを設定できます。

次の例で、gigabitethernet ネットワークはアドレスが 172.18.0.0 の クラス B ネットワークで、メール ホストのアドレスは 172.18.1.2 です。**established** キーワードを使用するのは、TCP プロトコルで確立済み接続を指定する場合のみです。TCP データグラムに ACK または RST ビットが設定されている場合に一致が発生します。これは、パケットが既存の接続に属することを示します。

```
interface gigabitethernet 0/0/0
 ip access-group 102 in
!
access-list 102 permit tcp any 172.18.0.0 0.0.255.255 established
access-list 102 permit tcp any host 172.18.1.2 eq 25
```

# 例：ポート名に基づくフィルタによる Web へのアクセス回避

次の例では、w1 および w2 ワークステーションは Web アクセスが許可されていません。ネットワーク 172.20.0.0 上のその他のホストは Web アクセスが許可されています。

```
interface gigabitethernet0/0/0
 ip access-group no-web out
!
ip access-list extended no-web
 remark Do not allow w1 to browse the web
 deny host 172.20.3.85 any eq http
 remark Do not allow w2 to browse the web
 deny host 172.20.3.13 any eq http
 remark Allow others on our network to browse the web
 permit 172.20.0.0 0.0.255.255 any eq http
```

# 例：送信元アドレスでのフィルタリングおよびパケットのロギング

次の例では、アクセス リスト 1 および 2 を定義します。いずれのリストもロギングが有効です。

```
interface gigabitethernet 0/0/0
 ip address 172.16.1.1 255.0.0.0
 ip access-group 1 in

!
access-list 1 permit 172.25.0.0 0.0.255.255 log
access-list 1 deny 172.30.0.0 0.0.255.255 log
!
access-list 2 permit 172.27.3.4 log
access-list 2 deny 172.17.0.0 0.0.255.255 log
```

インターフェイスが 172.25.7.7 から 10 パケットを受信し、172.17.23.21 から 14 パケットを受信する場合、最初のログは次のようになります。

```
list 1 permit 172.25.7.7 1 packet
list 2 deny 172.17.23.21 1 packet
```

5 分後、コンソールは、次のログを受信します。

```
list 1 permit 172.25.7.7 9 packets
list 2 deny 172.17.23.21 13 packets
```

# 例：デバッグ出力の制限

次の設定例では、アクセス リストを使用して、**debug** コマンドの出力を制限します。**debug** の出力を制限すると、データ量が絞られ、目的のデータを探しやすくなるため、時間とリソースを節約できます。

```
Device(config)# ip access-list acl1
Device(config-std-nacl)# remark Displays only advertisements for LDP peer in acl1
Device(config-std-nacl)# permit host 10.0.0.44

Device# debug mpls ldp advertisements peer-acl acl1

tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.17.0.33
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.16.0.31
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.22.0.33
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.0.1
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.0.3
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.1.33
```

# IPアクセスリストの作成とインターフェイスへの適用に関する追加参照資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | Cisco IOS Master Command List, All Releases |
| セキュリティ コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |
| • アクセス リスト エントリの順序<br>• 日または週の時刻に基づくアクセス リスト エントリ<br>• 非初期フラグメントを使用するパケット | Refining an IP Access List |
| IP オプション、TCP フラグ、または非隣接ポートに基づくフィルタリング | 『Creating an IP Access List for Filtering』 |
| ロギング関連のパラメータの制御 | 『Understanding Access Control List Loggingl』 |

### 標準および RFC

| 標準/RFC | タイトル |
|---|---|
| この機能によりサポートされる新規または変更された標準やRFCはありません。またこの機能による既存の標準や RFC のサポートに変更はありません。 | — |

### テクニカル サポート

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

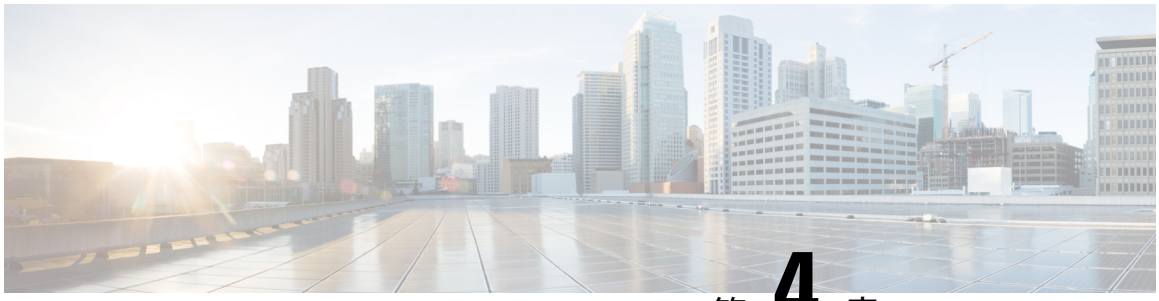
# IPアクセス リストの作成とインターフェイスへの適用に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *3: IP* アクセス リストの作成とインターフェイスへの適用に関する機能情報**

| 機能名 | リリース | 機能の設定情報 |
|---|---|---|
| ACL - アクセス コントロール リスト内の送信元アドレスと宛先アドレスの一致 | Cisco IOS XE リリース 3.5S | Cisco IOS XE リリース 3.5S では、Cisco ASR 903 ルータのサポートが追加されました。 |
| ACL - ICMP コード | Cisco IOS XE リリース 3.5S | Cisco IOS XE リリース 3.5S では、Cisco ASR 903 ルータのサポートが追加されました。 |
| ACL パフォーマンスの強化 | Cisco IOS XE リリース 2.1 | この機能は、Cisco ASR 1000 シリーズのアグリゲーション サービス ルータで導入されました。<br>この機能について導入または変更されたコマンドはありません。 |



第 4 章

# IP オプション、TCP フラグ、非隣接ポート、をフィルタする IP アクセス リストの作成

このモジュールは、特定の IP オプション、TCP フラグ、非隣接ポート、を含む IP パケットをフィルタする IP アクセス リストの使用方法について説明します。



## 機能情報の確認

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

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

# IP オプション、TCP フラグ、非隣接ポートをフィルタする IP アクセス リストの作成に関する前提条件

このモジュールのいずれかのタスクを実行する前に、次のモジュールの情報を把握しておく必要があります。

- 『IP アクセス リストの概要』
- 『IP アクセス リストの作成とインターフェイスへの適用』

# IP オプション、TCP フラグ、非隣接ポートをフィルタする IP アクセス リストの作成に関する情報

## IP オプション

IP は、サービスを提供するときに、タイプ オブ サービス、存続可能時間、オプション、およびヘッダー チェックサムという 4 つの主要メカニズムを使用します。

オプションは一般的に IP オプションと呼ばれ、一部の状況で必要な制御機能のために用意されていますが、ほとんどの一般的な通信では不要です。IP オプションには、タイム スタンプ、セキュリティ、および特殊なルーティングに関する条件が含まれます。

IP オプションはデータグラムに含まれる場合と含まれない場合があります。IP オプションはすべての IP モジュール（ホストとゲートウェイ）で実装する必要があります。オプションというのは、実装ではなく、任意の指定したデータグラムでの送信を指します。環境によっては、セキュリティ オプションがすべてのデータグラムで必要です。

オプション フィールドは長さが可変です。オプションの個数はゼロ個以上です。IP オプションには、次の 2 つの形式のいずれかを使用できます。

- 形式 1：単一オクテットの option-type
- 形式 2：1 つの option-type オクテット、option-length オクテット、および実際の option-data オクテット

option-length オクテットは、option-type オクテット、option-length オクテット、および option-data オクテットの数をカウントします。

option-type オクテットには、1 ビットのコピー済みフラグ、2 ビットのオプション クラス、および 5 ビットのオプション番号という 3 つのフィールドがあります。これらのフィールドは、オプション タイプ フィールドの 8 ビット値を構成します。IP オプションは、一般的にその 8 ビット値で参照されます。

IP オプションの詳細な一覧と説明については、次の URL の RFC 791『*Internet Protocol*』を参照してください。http://www.faqs.org/rfcs/rfc791.html

# IP オプションをフィルタする利点

- ネットワークからの IP オプションを含むパケットをフィルタすることで、ダウンストリームのデバイスとホストにかかるオプション パケットの負荷が軽減されます。
- また、この機能によって、分散型システムでルート プロセッサ（RP）処理が必要な IP オプションを含むパケットについて、RP への負荷が最小限になります。以前は、パケットは常に RP CPU でルーティングまたは処理されていました。パケットをフィルタすることで、パケットの RP への影響を回避できます。

# TCP フラグに基づいてフィルタする利点

ACL TCP フラグ フィルタリング機能には、TCP フラグに基づいてフィルタする柔軟なメカニズムが用意されています。以前は、パケットのいずれかの TCP フラグがアクセス コントロールエントリ（ACE）で指定されたフラグに一致する限り、着信パケットは一致していました。すべてのフラグが設定されたパケットがアクセス コントロール リスト（ACL）を通過する可能性があるため、この動作ではセキュリティの抜け穴を考慮しています。ACL TCP フラグ フィルタリング機能では、フィルタするフラグの任意の組み合わせを選択できます。設定されているフラグ、および設定されていないフラグに基づいてマッチングする機能によって、TCP フラグに基づくフィルタリングの制御性が向上するため、セキュリティが強化されます。

TCP パケットは偽造の同期パケットとして送信され、それがリスニング ポートで受け入れられる可能性があるため、ファイアウォールデバイスの管理者は、偽造の TCP パケットをドロップするフィルタリング ルールを設定することを推奨します。

アクセス リストを構成する ACE を設定し、特定のグループの TCP フラグが設定されているパケットのみ、または設定されていないパケットのみを許可することで、不正な TCP パケットを検出およびドロップできます。ACL TCP フラグ フィルタリング機能によって、次のようにパケット フィルタリングの制御性が向上します。

- フィルタする TCP パケットについて、TCP フラグの任意の組み合わせを選択できます。
- 設定されているフラグと設定されていないフラグに基づいてマッチングできるように、ACE を設定できます。

# TCP Flags

次の表は TCP フラグの一覧です。詳細については、RFC 793『*Transmission Control Protocol*』を参照してください。

表 *4 : TCP Flags*

| TCP フラグ | 目的 |
|---|---|
| ACK | Acknowledge フラグ：セグメントの acknowledgment フィールドが、このセグメントの送信元が受信を予測している番号の次のシーケンス番号を指定することを示します。 |
| FIN | Finish フラグ：接続をクリアするために使用されます。 |
| PSH | Push フラグ：呼び出しのデータを受信ユーザに対してただちにプッシュする必要があることを示します。 |
| RST | Reset フラグ：受信者が以降のやり取りなしで接続を削除する必要があることを示します。 |
| SYN | Synchronize フラグ：接続の確立に使用されます。 |
| URG | Urgent フラグ：urgent フィールドが重要で、セグメント シーケンス番号に追加する必要があることを示します。 |

# アクセス コントロール エントリ機能での非隣接ポートに関する名前付き ACL サポートを使用する利点

この機能によって、同じ送信元アドレス、宛先アドレス、およびプロトコルに関して複数のエントリを処理するために、アクセス コントロール リストで必要なアクセス コントロール エントリ（ACE）の数が大幅に削減されます。大量の ACE を保守している場合、可能な限り、新しいアクセス リスト エントリを作成するときは、この機能を使用して既存のアクセス リスト エントリのグループを統合します。非隣接ポートを使用するアクセス リスト エントリを設定すると、保守するアクセス リスト エントリ数が少なくなります。

# TTL 値のフィルタリング方法

IP は、拡張名前付きおよび番号付きアクセス リストは、インターフェイスを発着信するパケットの TTL 値でフィルタリングできます。有効な TTL 値 0 ～ 255 のパケットを許可または拒否できます（フィルタリング）。その他のフィールド（送信元または宛先アドレスなど）でのフィルタリングと同様に、**ip access-group** コマンドは **in** または **out** を指定します。これにより、アクセス リストの入力または出力が行われ、それぞれ着信または発信パケットに適用されます。TTL 値は、アクセス リスト エントリで指定したプロトコル、アプリケーション、およびその他の設定とともにチェックされ、すべての条件を満たす必要があります。

### 入力インターフェイスに到達した TTL 値 0 または 1 のパケットに対する特別な処理

分散型シスコ エクスプレス フォワーディング（dCEF）、CEF、ファスト スイッチング、プロセス スイッチングなどのソフトウェア スイッチング パスは、通常、アクセス リスト ステートメントに基づいてパケットを許可または廃棄します。ただし、入力インターフェイスに到達したパケットの TTL 値が 0 または 1 であるときには、特別な処理が必要です。TTL 値が 0 または 1 のパケットは、CEF、dCEF、またはファスト スイッチング パスで入力アクセス リストがチェックされる前に、プロセス レベルに送信されます。入力アクセス リストは、TTL 値が 2 ～ 255 であるパケットに適用され、許可または拒否の決定が行われます。

TTL 値が 0 または 1 のパケットは、デバイスから外部に転送されることがないため、プロセス レベルに送信されます。プロセス レベルでは、各パケットがそのデバイス宛であるかどうか、および Internet Control Message Protocol（ICMP）TTL 値期限切れメッセージを返送する必要があるかどうかをチェックする必要があります。つまり、TTL が 0 または 1 のパケットをドロップする意図で TTL 値 0 または 1 のフィルタリングを設定した ACL が入力インターフェイスで設定されている場合でも、高速なパスではパケットのドロップが発生しないということです。代わりに、プロセスが ACL を適用するときに、プロセス レベルで発生します。これはハードウェア スイッチング プラットフォームについてもあてはまります。TTL 値が 0 または 1 のパケットはルート プロセッサ（RP）またはマルチレイヤ スイッチ フィーチャ カード（MSFC）のプロセス レベルに送信されます。

出力インターフェイスでは、TTL 値でのアクセス リスト フィルタリングは、その他のアクセス リスト機能と同じように動作します。チェックはデバイスで有効な最も高速なスイッチング パスで行われます。これは、より高速なスイッチング パスは出力インターフェイスですべての TTL 値（0 ～ 255）を均等に処理するためです。

### TTL 値 0 と 1 でフィルタリングするためのコントロール プレーン ポリシング

TTL 値が 0 または 1 のパケットに対する特別な動作によって、デバイスの CPU 使用率が高くなります。0 または 1 の TTL 値 でフィルタリングする場合は、CPU が過負荷になることを防ぐためにコントロール プレーン ポリシング（CPP）を使用してください。CPP を活用するには、TTL 値 0 および 1 をフィルタリングすることに特化したアクセス リストを設定し、CPP を通じてそのアクセス リストを適用する必要があります。このアクセス リストは、その他のインターフェイス アクセス リストとは別のアクセス リストにします。CPP は個々のインターフェイスにおいてではなくシステム全体に対して機能するため、そのようなアクセスリストはデバイス全体に対して 1 つのみ設定する必要があります。このタスクは、セクション「TTL 値 0 と 1 でフィルタリングするコントロールプレーン ポリシングの有効化」で説明しています。

## TTL 値に基づいてフィルタする利点

- 存続可能時間（TTL）値でのフィルタリングは、デバイスに到達できるパケット、またはデバイスに到達できないパケットを制御する方法を提供します。ネットワーク レイアウトを確認することで、特定のデバイスからのパケットをホップ数に基づいて許可するか拒否するかを選択できます。たとえば、小規模ネットワークでは、ホップ数が 3 より大きい場所からのパケットを拒否する可能性があります。TTL 値でのフィルタリングでは、トラフィックがネイバーデバイスから発信されたかどうかを検証できます。たとえば特定プロ

トコルの初期 TTL 値より 1 小さい TTL 値のパケットのみを受け入れることで、1 ホップで自分に到達するパケットのみを受け入れることができます。

- 多くのコントロール プレーン プロトコルはネイバーのみと通信しますが、パケットを誰からも受信します。TTL でフィルタリングするアクセス リストを受信側ルータに適用すると、不要なパケットをブロックできます。
- Cisco ソフトウェアが送信するすべてのパケットは、プロセス レベルに対して TTL 値が 0 または 1 です。デバイスは、Internet Control Message Protocol（ICMP）TTL 値期限切れメッセージを送信元に送信する必要があります。TTL 値が 0 ～ 2 であるパケットをフィルタリングすることで、プロセス レベルでの負荷を削減できます。

# IP オプション、TCP フラグ、非隣接ポートをフィルタする IP アクセス リストの作成方法

## IP オプションを含むパケットのフィルタリング

アクセス リストを設定して、IP オプションを含むパケットをフィルタし、アクセス リストが適切に設定されていることを確認するには、次の手順を完了します。

（注）
- IP オプションのフィルタリングに関する ACL のサポート機能は、名前付きの拡張 ACL でのみ使用できます。
- この機能を設定する場合、リソース予約プロトコル（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（MPLS TE）、Internet Group Management Protocol バージョン 2（IGMPV2）、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、ドロップまたは無視モードでは機能しない可能性があります。
- ほとんどの Cisco デバイスでは、IP オプションを含むパケットはハードウェアではスイッチされませんが、処理するコントロール プレーン ソフトウェアが必要です（主に、オプションを処理し、IP ヘッダーを書き直す必要があるため）。結果として、IP オプションを含むすべての IP パケットは、ソフトウェアでフィルタとスイッチが行われます。

**ステップ 1　enable**

例：

```
Device> enable
```

特権 EXEC モードをイネーブルにします。

- パスワードを入力します（要求された場合）。

**ステップ 2　configure　terminal**

例：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **ip access-list extended** *access-list-name*

例：

```
Device(config)# ip access-list extended mylist1
```

名前で IP アクセス リストを指定し、名前付きアクセス リストのコンフィギュレーション モードを開始します。

**ステップ 4** [*sequence-number*] **deny** *protocol source source-wildcard destination destination-wildcard* [**option** *option-value*] [**precedence** *precedence*] [ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]

例：

```
Device(config-ext-nacl)# deny ip any any option traceroute
```

（任意）名前付き IP アクセス リスト モードで **deny** ステートメントを指定します。

- このアクセス リストでは **deny** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**permit** ステートメントが最初に使用される可能性もあります。
- **option** キーワードおよび *option-value* 引数を使用して、特定の IP オプションを含むパケットをフィルタします。
- この例では、traceroute IP オプションを含むすべてのパケットが除外されます。
- エントリを削除するには、このコマンドの **no** *sequence-number* 形式を使用します。

**ステップ 5** [*sequence-number*] **permit** *protocol source source-wildcard destination destination-wildcard* [**option** *option-value*] [**precedence** *precedence*] [ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]

例：

```
Device(config-ext-nacl)# permit ip any any option security
```

名前付き IP アクセス リスト モードで **permit** ステートメントを指定します。

- この例では、セキュリティ IP オプションを含むすべてのパケット（まだフィルタされていないパケット）が許可されます。
- エントリを削除するには、このコマンドの **no** *sequence-number* 形式を使用します。

**ステップ 6** 必要に応じて、ステップ 4 またはステップ 5 を繰り返します。

アクセス リストは変更できます。

**ステップ 7** **end**

例：

```
Device(config-ext-nacl)# end
```

（任意）名前付きアクセス リスト コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

**ステップ 8** **show ip access-lists** *access-list-name*

例：

```
Device# show ip access-lists mylist1
```

（任意）IP アクセス リストの内容を表示します。

## 次の作業

アクセス リストをインターフェイスに適用するか、アクセス リストを受け入れるコマンドから参照します。

（注） IP オプションを含むすべてのパケットを効率的に除去するには、**ip options drop** グローバル コマンドを設定することを推奨します。

# TCP フラグを含むパケットのフィルタリング

この作業では、アクセス リストを設定して、TCP フラグを含むパケットをフィルタし、アクセス リストが適切に設定されていることを確認します。

（注）
- TCP フラグのフィルタリングを使用できるのは、名前付きの拡張 ACL のみです。
- ACL TCP フラグ フィルタリング機能は、Cisco ACL の場合にのみサポートされます。
- 事前に、次のコマンドライン インターフェイス（CLI）形式を使用して、TCP フラグチェック メカニズムを設定できます。

**permit tcp any any rst** 同じ ACE を示す次の形式を使用できるようになりました。 **permit tcp any any match-any +rst** いずれの CLI 形式も使用できますが、新しいキーワード **match-all** または **match-any** を選択する場合、プレフィックスに「+」または「-」を付けた新しいフラグを次に指定する必要があります。単一の ACL では、古い形式のみ、または新しい形式のみを使用することを推奨します。CLI の古い形式と新しい形式の混在やマッチングを行うことはできません。

**注意** 新しい構文形式の ACE を持つデバイスを、ACL TCP フラグ フィルタリング機能をサポートしないシスコ ソフトウェアの以前のバージョンでリロードすると、ACE は適用されないため、セキュリティの抜け穴が発生する可能性があります。

**ステップ 1** **enable**

例：

```
Device> enable
```

特権 EXEC モードをイネーブルにします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

例：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **ip access-list extended** *access-list-name*

例：

```
Device(config)# ip access-list extended kmd1
```

名前で IP アクセス リストを指定し、名前付きアクセス リストのコンフィギュレーション モードを開始します。

**ステップ 4** [*sequence-number*] **permit tcp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**established**|{**match-any** | **match-all**} {**+** | **-**} *flag-name*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]

例：

```
Device(config-ext-nacl)# permit tcp any any match-any +rst
```

名前付き IP アクセス リスト モードで **permit** ステートメントを指定します。

- このアクセス リストでは **permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。
- **permit** コマンドの TCP コマンド構文を使用します。
- RST TCP ヘッダー フラグが設定されたすべてのパケットは一致し、ステップ 3 で名前付きアクセス リスト kmd1 に合格できます。

**ステップ 5** [*sequence-number*] **deny tcp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**established**|{**match-any** | **match-all**} {**+** | **-**} *flag-name*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]

例：

```
Device(config-ext-nacl)# deny tcp any any match-all -ack -fin
```

（任意）名前付き IP アクセス リスト モードで **deny** ステートメントを指定します。

- このアクセス リストでは **permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。
- **deny** コマンドの TCP コマンド構文を使用します。

- ACK フラグが設定されず、FIN フラグも設定されていないパケットは、ステップ 3 で名前付きアクセス リスト kmd1 に合格しません。
- 上位層プロトコル（ICMP、IGMP、TCP、およびUDP）を許可するその他のコマンド構文については、**deny**（IP）コマンドを参照してください。

**ステップ 6** 必要に応じてステップ 4 またはステップ 5 を繰り返し、計画したシーケンス番号でステートメントを追加します。エントリを削除するには、**no** *sequence-number* コマンドを使用します。

アクセス リストは変更できます。

**ステップ 7** **end**

例：

```
Device(config-ext-nacl)# end
```

（任意）コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

**ステップ 8** **show ip access-lists** *access-list-name*

例：

```
Device# show ip access-lists kmd1
```

（任意）IP アクセス リストの内容を表示します。

- 出力を見直して、アクセス リストに新しいエントリが含まれることを確認します。

## 次の作業

アクセス リストをインターフェイスに適用するか、アクセス リストを受け入れるコマンドから参照します。

# 非隣接ポートを使用するアクセス コントロール エントリの設定

非隣接 TCP または UDP ポート番号を使用するアクセス リスト エントリを作成するには、次の作業を実行します。この作業では TCP ポートを使用しますが、**permit** および **deny** コマンドの UDP 構文を使用して、非隣接 UDP ポートをフィルタすることもできます。

この作業では **permit** コマンドを最初に使用していますが、フィルタリングの目標に合わせた順序で、**permit** および **deny** コマンドを使用できます。

（注） ACL：アクセス コントロール エントリでの非隣接ポートに関する名前付き ACL サポート機能を使用できるのは、名前付きの拡張 ACL のみです。

**ステップ 1** **enable**

**例**：

```
Device> enable
```

特権 EXEC モードをイネーブルにします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

**例**：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **ip access-list extended** *access-list-name*

**例**：

```
Device(config)# ip access-list extended acl-extd-1
```

名前で IP アクセス リストを指定し、名前付きアクセス リストのコンフィギュレーション モードを開始します。

**ステップ 4** [*sequence-number*] **permit tcp** *source source-wildcard* [*operator port* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**established** {**match-any** | **match-all**} {**+** | **-**} *flag-name*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]

**例**：

```
Device(config-ext-nacl)# permit tcp any eq telnet ftp any eq 450 679
```

名前付き IP アクセス リスト コンフィギュレーション モードで **permit** ステートメントを指定します。

- 演算子には、**lt**（次の値より小さい）、**gt**（次の値より大きい）、**eq**（次の値に等しい）、**neq**（次の値に等しくない）**range**（次の範囲）があります。
- 演算子が source および source-wildcard 引数の後にある場合、送信元ポートに一致する必要があります。演算子が destination および destination-wildcard 引数の後にある場合、宛先ポートに一致する必要があります。
- **range** 演算子には 2 つのポート番号が必要です。**eq** および **neq** 演算子の後には、最大 10 個のポートを設定できます。他のすべての演算子は 1 つのポート番号が必要です。
- UDP ポートをフィルタするには、このコマンドの UDP 構文を使用します。

**ステップ 5** [*sequence-number*] **deny tcp** *source source-wildcard* [*operator port* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**established** {**match-any** | **match-all**} {**+** | **-**} *flag-name*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]

**例**：

```
Device(config-ext-nacl)# deny tcp any neq 45 565 632 any
```

（任意）名前付きアクセス リスト コンフィギュレーション モードで **deny** ステートメントを指定します。

- 演算子には、**lt**（次の値より小さい）、**gt**（次の値より大きい）、**eq**（次の値に等しい）、**neq**（次の値に等しくない）**range**（次の範囲）があります。
- 演算子が *source* および *source-wildcard* 引数の後にある場合、送信元ポートに一致する必要があります。演算子が *destination* および *destination-wildcard* 引数の後にある場合、宛先ポートに一致する必要があります。
- **range** 演算子には 2 つのポート番号が必要です。**eq** および **neq** 演算子の後には、最大 10 個のポートを設定できます。他のすべての演算子は 1 つのポート番号が必要です。
- UDP ポートをフィルタするには、このコマンドの UDP 構文を使用します。

**ステップ 6** 必要に応じてステップ 4 またはステップ 5 を繰り返し、計画したシーケンス番号でステートメントを追加します。エントリを削除するには、**no** *sequence-number* コマンドを使用します。

アクセス リストは変更できます。

**ステップ 7** **end**

**例**：

```
Device(config-ext-nacl)# end
```

（任意）名前付きアクセス リスト コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

**ステップ 8** **show ip access-lists** *access-list-name*

**例**：

```
Device# show ip access-lists kmd1
```

（任意）アクセス リストの内容を表示します。

# 非隣接ポートを使用する複数アクセス リスト エントリの1つのアクセス リスト エントリへの統合

非隣接ポートを使用するアクセス リスト エントリ グループを 1 つのアクセス リスト エントリに統合するには、次の作業を実行します。

この作業では TCP ポートを使用しますが、**permit** および **deny** コマンドの UDP 構文を使用して、非隣接 UDP ポートをフィルタすることもできます。

この作業では **permit** コマンドを最初に使用していますが、フィルタリングの目標に合わせた順序で、**permit** および **deny** コマンドを使用できます。

**ステップ 1** **enable**

**例**：

```
Device> enable
```

特権 EXEC モードをイネーブルにします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **show ip access-lists** *access-list-name*

**例**：

```
Device# show ip access-lists mylist1
```

（任意）IP アクセス リストの内容を表示します。

- 出力を見直して、アクセス リスト エントリを統合できるかどうかを確認します。

**ステップ 3** **configure terminal**

**例**：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 4** **ip access-list extended** *access-list-name*

**例**：

```
Device(config)# ip access-list extended mylist1
```

名前で IP アクセス リストを指定し、名前付きアクセス リストのコンフィギュレーション モードを開始します。

**ステップ 5** **no** [*sequence-number*] **permit** *protocol source source-wildcard destination destination-wildcard*[**option** *option-name*] [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]

**例**：

```
Device(config-ext-nacl)# no 10
```

統合できる重複するアクセス リスト エントリを削除します。

- このステップを繰り返して、ポート番号のみが異なるために統合できるエントリを削除します。
- このステップを繰り返して、たとえばアクセス リスト エントリ 20、30、および 40 を削除した後は、1 つの **permit** ステートメントに統合されるため、これらのエントリは削除されます。
- *sequence-number* が指定された場合、その他のコマンド構文は任意です。

**ステップ 6** [*sequence-number*] **permit** *protocol source source-wildcard*[*operator port*[*port*]] *destination destination-wildcard*[*operator port*[*port*]] [**option** *option-name*] [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]

**例**：

```
Device(config-ext-nacl)# permit tcp any neq 45 565 632 any eq 23 45 34 43
```

名前付きアクセス リスト コンフィギュレーション モードで **permit** ステートメントを指定します。

- このインスタンスでは、非隣接ポートを使用するアクセス リスト エントリ グループは、1 つの **permit** ステートメントに統合されました。

• **eq** および **neq** 演算子の後には、最大 10 個のポートを設定できます。

**ステップ 7** 必要に応じてステップ 5 と 6 を繰り返し、**permit** または **deny** ステートメントを追加して、可能な場合はアクセス リスト エントリを統合します。エントリを削除するには、**no** *sequence-number* コマンドを使用します。

アクセス リストは変更できます。

**ステップ 8** **end**

**例**：

```
Device(config-std-nacl)# end
```

（任意）名前付きアクセス リスト コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

**ステップ 9** **show ip access-lists** *access-list-name*

**例**：

```
Device# show ip access-lists mylist1
```

（任意）アクセス リストの内容を表示します。

## 次の作業

アクセス リストをインターフェイスに適用するか、アクセス リストを受け入れるコマンドから参照します。

# TTL 値に基づいたパケットのフィルタリング

アクセス リストは柔軟性に優れているため、TTL 値に基づいてパケットをフィルタリングする **permit** と **deny** コマンドの組み合わせ 1 つだけでは定義することができません。次のタスクでは、TTL フィルタリングを実行する例を 1 つだけ示します。独自のフィルタリング プランを満たす **permit** と **deny** ステートメントを適切に設定します。

（注） デバイスで使用する Cisco のソフトウェア リリースに応じて、アクセス リストで演算子 EQ または NEQ を指定する場合、アクセス リストでは最大 10 個の TTL 値を指定できます。TTL 値の数は、シスコのソフトウェア リリースによって異なります。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *access-list-name*

4. [*sequence-number*] **permit** *protocol source source-wildcard destination destination-wildcard*[ **option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**ttl** *operator value*] [**log**] [**time-range** *time-range-name*] [**fragments**]
5. **permit** または **deny** ステートメントを続けて追加し、必要なフィルタリングを実現します。
6. **exit**
7. **interface** *type number*
8. **ip access-group** *access-list-name* {**in** | **out**}

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip access-list extended** *access-list-name*<br>例：<br>`Device(config)# ip access-list extended ttlfilter` | IP アクセス リストを名前で定義します。<br>• TTL 値でフィルタリングするアクセス リストは、拡張アクセス リストである必要があります。 |
| ステップ 4 | [*sequence-number*] **permit** *protocol source source-wildcard destination destination-wildcard*[ **option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**ttl** *operator value*] [**log**] [**time-range** *time-range-name*] [**fragments**]<br>例：<br>`Device(config-ext-nacl)# permit ip host 172.16.1.1 any ttl lt 2` | パケットが名前付き IP アクセス リストを通過できる条件を設定します。<br>• すべてのアクセス リストには、**permit** ステートメントが 1 つ以上必要です。<br>• この例では、送信元 172.16.1.1 から TTL 値が 2 未満の接続先へのパケットが許可されています。 |
| ステップ 5 | **permit** または **deny** ステートメントを続けて追加し、必要なフィルタリングを実現します。 | -- |
| ステップ 6 | **exit**<br>例：<br>`Device(config-ext-nacl)# exit` | コンフィギュレーション モードを終了して、コマンドライン インターフェイス（CLI）モード階層で次に高いレベルのモードを開始します。 |
| ステップ 7 | **interface** *type number*<br>例： | インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config)# interface ethernet 0` | |
| ステップ 8 | **ip access-group** *access-list-name* {**in** \| **out**}<br>例：<br>`Device(config-if)# ip access-group ttlfilter in` | アクセス リストをインターフェイスに適用します。 |

# TTL 値 0 と 1 でフィルタリングするコントロール プレーン ポリシングの有効化

TTL 値 0 または 1 に基づいて IP パケットをフィルタリングしたり、CPU の過負荷を防止したりするには、次のタスクを実行します。このタスクでは、TTL 値 0 と 1 で分類用のアクセス リストを設定し、モジュラ QoS コマンドライン インターフェイス（CLI）（MQC）を設定して、ポリシーマップをコントロールプレーンに適用します。アクセスリストを通過するパケットはドロップされます。この特別なアクセス リストは、他のインターフェイス アクセス リストとは異なります。

アクセス リストは柔軟性に優れているため、TTL 値に基づいてパケットをフィルタリングする **permit** と **deny** コマンドの組み合わせ 1 つだけでは定義することができません。次のタスクでは、TTL フィルタリングを実行する例を 1 つだけ示します。独自のフィルタリング プランを満たす **permit** と **deny** ステートメントを適切に設定します。

## 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *access-list-name*
4. [*sequence-number*] **permit** *protocol source source-wildcard destination destination-wildcard* **ttl** *operator* value
5. **permit** または **deny** ステートメントを続けて追加し、必要なフィルタリングを実現します。
6. **exit**
7. **class-map** *class-map-name* [**match-all** | **match-any**]
8. **match access-group** {*access-group* | **name** *access-group-name*}
9. **exit**
10. **policy-map** *policy-map-name*
11. **class** {*class-name* | **class-default**}
12. **drop**
13. **exit**
14. **exit**
15. **control-plane**
16. **service-policy** {**input** | **output**} *policy-map-name*

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip access-list extended** *access-list-name*<br>例：<br>`Device(config)# ip access-list extended ttlfilter` | IP アクセス リストを名前で定義します。<br>• TTL 値でフィルタリングするアクセス リストは、拡張アクセス リストである必要があります。 |
| ステップ 4 | [*sequence-number*] **permit** *protocol source source-wildcard destination destination-wildcard* **ttl** *operator* value<br>例：<br>`Device(config-ext-nacl)# permit ip host 172.16.1.1 any ttl lt 2` | パケットが名前付き IP アクセス リストを通過できる条件を設定します。<br>• すべてのアクセス リストには、**permit** ステートメントが 1 つ以上必要です。<br>• この例では、送信元 172.16.1.1 から TTL 値が 2 未満の接続先へのパケットが許可されています。 |
| ステップ 5 | **permit** または **deny** ステートメントを続けて追加し、必要なフィルタリングを実現します。 | アクセス リストを通過するパケットはドロップされます。 |
| ステップ 6 | **exit**<br>例：<br>`Device(config-ext-nacl)# exit` | コンフィギュレーション モードを終了して、CLI モード階層で次に高いレベルのモードを開始します。 |
| ステップ 7 | **class-map** *class-map-name* [**match-all** \| **match-any**]<br>例：<br>`Device(config)# class-map acl-filtering` | 指定したクラスへのパケットのマッチングに使用するクラス マップを作成します。 |
| ステップ 8 | **match access-group** {*access-group* \| **name** *access-group-name*}<br>例：<br>`Device(config-cmap)# match access-group name ttlfilter` | 指定したアクセス コントロール リストに基づいて、クラス マップの一致基準を設定します |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 9 | **exit**<br>例：<br>`Device(config-cmap)# exit` | コンフィギュレーション モードを終了して、CLI モード階層で次に高いレベルのモードを開始します。 |
| ステップ 10 | **policy-map** *policy-map-name*<br>例：<br>`Device(config)# policy-map acl-filter` | 1 つ以上のインターフェイスに付加できるポリシーマップを作成または変更し、サービス ポリシーを指定します。 |
| ステップ 11 | **class** {*class-name* \| **class-default**}<br>例：<br>`Device(config-pmap)# class acl-filter-class` | 作成または変更するポリシーのクラス名を指定するか、ポリシーを指定する前にデフォルト クラス（一般に class-default クラスといいます）を指定します。 |
| ステップ 12 | **drop**<br>例：<br>`Device(config-pmap-c)# drop` | 特定のクラスに属するパケットを廃棄するトラフィック クラスを設定します。 |
| ステップ 13 | **exit**<br>例：<br>`Device(config-pmap-c)# exit` | コンフィギュレーション モードを終了して、CLI モード階層で次に高いレベルのモードを開始します。 |
| ステップ 14 | **exit**<br>例：<br>`Device(config-pmap)# exit` | コンフィギュレーション モードを終了して、CLI モード階層で次に高いレベルのモードを開始します。 |
| ステップ 15 | **control-plane**<br>例：<br>`Device(config)# control-plane` | デバイスのコントロール プレーンに関連する属性またはパラメータを関連付けたり、変更したりします。 |
| ステップ 16 | **service-policy** {**input** \| **output**} *policy-map-name*<br>例：<br>`Device(config-cp)# service-policy input acl-filter` | 集約コントロール プレーン サービスのためにポリシーマップをコントロールプレーンに適用します。 |

# IP オプション、TCP フラグ、非隣接ポートのフィルタリングの設定例

## 例：IP オプションを含むパケットのフィルタリング

次の例は、アクセス リスト エントリ（ACE）に指定されている IP オプションが含まれる場合にのみ、TCP パケットを許可するように設定された ACE を含む、mylist2 という拡張アクセス リストを示します。

```
ip access-list extended mylist2
 10 permit ip any any option eool
 20 permit ip any any option record-route
 30 permit ip any any option zsu
 40 permit ip any any option mtup
```

一致し、それによって許可されたパケットの数を示すため、**show access-list** コマンドが入力されました。

```
Device# show ip access-list mylist2
Extended IP access list test
10 permit ip any any option eool (1 match)
20 permit ip any any option record-route (1 match)
30 permit ip any any option zsu (1 match)
40 permit ip any any option mtup (1 match)
```

## 例：TCP フラグを含むパケットのフィルタリング

次のアクセス リストでは、TCP フラグ ACK および SYN が設定され、FIN フラグが設定されていない場合にのみ、TCP パケットを許可します。

```
ip access-list extended aaa
 permit tcp any any match-all +ack +syn -fin
 end
```

**show access-list** コマンドは、ACL を表示するために入力しました。

```
Device# show access-list aaa

Extended IP access list aaa
 10 permit tcp any any match-all +ack +syn -fin
```

## 例：非隣接ポートを使用するアクセス リスト エントリの作成

**eq** および **neq** 演算子の後に最大 10 ポートを入力できるため、次のアクセス リスト エントリを作成できます。

```
ip access-list extended aaa
```

```
 permit tcp any eq telnet ftp any eq 23 45 34
 end
```

**show access-lists** コマンドを入力して、新しく作成されたアクセス リスト エントリを表示します。

```
Device# show access-lists aaa

Extended IP access list aaa
 10 permit tcp any eq telnet ftp any eq 23 45 34
```

# 例：既存の複数のアクセス リスト エントリと非隣接ポートを使用する 1 つのアクセス リスト エントリの統合

**show access-lists** コマンドは、abc というアクセス リストについて、アクセス リスト エントリ グループを表示するために使用されます。

```
Device# show access-lists abc
Extended IP access list abc
 10 permit tcp any eq telnet any eq 450
 20 permit tcp any eq telnet any eq 679
 30 permit tcp any eq ftp any eq 450
 40 permit tcp any eq ftp any eq 679
```

エントリはすべて同じ **permit** ステートメント用であり、ポートのみが異なるため、1 つの新しいアクセス リスト エントリに統合できます。次の例では、重複するアクセス リスト エントリを削除し、以前に表示されていたアクセス リスト エントリ グループを統合する新しいアクセス リスト エントリを作成します。

```
ip access-list extended abc
 no 10
 no 20
 no 30
 no 40
 permit tcp any eq telnet ftp any eq 450 679
 end
```

**show access-lists** コマンドを再入力すると、統合されたアクセス リスト エントリが表示されます。

```
Device# show access-lists abc
Extended IP access list abc
 10 permit tcp any eq telnet ftp any eq 450 679
```

# 例：TTL 値のフィルタリング

次のアクセス リストは、存続可能時間（TTL）の値が 10 と 20 でタイプ オブ サービス（ToS）レベルが 3 の IP パケットをフィルタリングします。また、TTL が 154 を超える IP パケットをフィルタリングし、その規則を先頭以外のフラグメントにも適用します。フラッシュの優先レベルと 1 以外の TTL 値を持つ IP パケットを許可し、そのようなパケットのログ メッセージをコンソールに送信します。他のすべてのパケットは拒否されます。

```
ip access-list extended incomingfilter
 deny ip any any tos 3 ttl eq 10 20
 deny ip any any ttl gt 154 fragments
 permit ip any any precedence flash ttl neq 1 log
!
interface ethernet 0

ip access-group incomingfilter in
```

# 例：TTL 値 0 と 1 でフィルタリングするコントロール プレーン ポリシング

次の例では、acl-filter と呼ばれるポリシー マップで使用するために、acl-filter-class と呼ばれるトラフィック クラスを設定します。アクセス リストは、存続可能時間（TTL）値が 0 または 1 の送信元からの IP パケットを許可します。アクセス リストに一致するパケットがドロップされます。ポリシー マップはコントロール プレーンに結合されます。

```
ip access-list extended ttlfilter

 permit ip any any ttl eq 0 1

class-map acl-filter-class

 match access-group name ttlfilter

policy-map acl-filter

 class acl-filter-class

 drop

control-plane

 service-policy input acl-filter
```

# その他の参考資料

**関連資料**

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | Cisco IOS Master Command List, All Releases |
| セキュリティ コマンド | 『*Cisco IOS Security Command Reference*』 |

| 関連項目 | マニュアル タイトル |
|---|---|
| **no ip options** コマンドを使用した、IP オプションを含むパケットをドロップまたは無視するためのデバイスの設定。 | 『ACL IP Options Selective Drop』 |
| アクセス リストに関する概要情報 | 『IP Access List Overview』 |
| IP アクセス リストの作成とインターフェイスへの適用に関する情報 | 『Creating an IP Access List and Applying It to an Interface』 |
| QoS コマンド | 『*Cisco IOS Quality of Service Solutions Command Reference*』 |

### RFC

| RFC | タイトル |
|---|---|
| RFC 791 | *Internet Protocol*（インターネット プロトコル）<br>http://www.faqs.org/rfcs/rfc791.html |
| RFC 793 | 伝送制御プロトコル（*TCP*） |
| RFC 1393 | 『*Traceroute Using an IP Option*』 |

### テクニカル サポート

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

# フィルタするためのIPアクセスリストの作成に関する機能情報

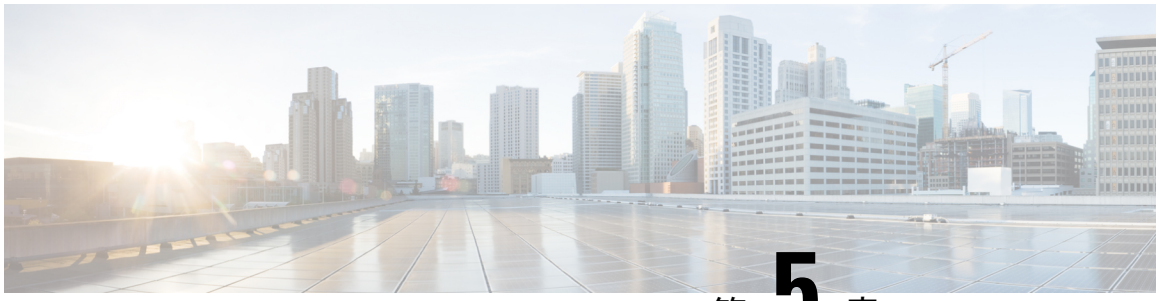
次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだ

けを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *5:* フィルタするための *IP* アクセス リストの作成に関する機能情報**

| 機能名 | リリース | 機能の設定情報 |
|---|---|---|
| ACL -- アクセス コントロールエントリでの非隣接ポートに関する名前付き ACL サポート | 12.3(7)T<br>12.2(25)S | この機能を使用すると、1 つのアクセス コントロール エントリで非隣接ポートを指定できるため、複数のエントリが同じ送信元アドレス、宛先アドレス、およびプロトコルを持ち、ポートのみが異なる場合に、アクセス コントロール リストで必要なエントリ数を大幅に減らすことができます。 |
| IP オプションのフィルタリングに関する ACL のサポート | 12.3(4)T<br>12.2(25)S<br>15.2(2)S<br>15.4(1)S | この機能を使用すると、IP オプションを含むパケットをフィルタできます。その結果、ルータが偽造パケットで飽和状態にならないように防ぎます。<br>Cisco IOS リリース 15.4(1)S では、Cisco ASR 901S ルータのサポートが追加されました。 |
| ACL TCP フラグ フィルタリング | 12.3(4)T<br>12.2(25)S | この機能は、TCP フラグに基づくフィルタリングに柔軟なメカニズムを提供します。Cisco IOS リリース 12.3(4)T 以前は、パケット内のいずれかの TCP フラグがアクセス コントロール エントリ（ACE）で指定されたフラグに一致する限り、着信パケットは一致していました。すべてのフラグが設定されたパケットがアクセス コントロール リスト（ACL）を通過する可能性があるため、この動作ではセキュリティの抜け穴を考慮しています。ACL TCP フラグ フィルタリング機能では、フィルタするフラグの任意の組み合わせを選択できます。設定されているフラグ、および設定されていないフラグに基づいてマッチングする機能によって、TCP フラグに基づくフィルタリングの制御性が向上するため、セキュリティが強化されます。 |



第 5 章

# FQDN ACL の設定

このドキュメントでは、完全修飾ドメイン名（FQDN）を使用したアクセス コントロール リスト（ACL ）を設定する方法について説明します。FQDN ACL 機能を設定することによって、ドメイン名システム（DNS ）に基づいて、ワイヤレス セッションに ACL を設定および適用することができます。ドメイン名を IP アドレスに解決されます。IP アドレスは、DNS 応答の一部としてクライアントに提供され、FQDN は、IP アドレスに基づいて、ACL にマッピングされます。



## 機能情報の確認

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

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

## FQDN ACL の設定に関する制約事項

FQDN ACL 機能の設定は、IPv4 ワイヤレス セッションでのみサポートされます。

# FQDN ACL の設定に関する情報

## FQDN ACL の設定

アクセス コントロール リスト（ACL）が、完全修飾ドメイン名（FQDN）を使用して設定されている場合、宛先ドメイン名に基づいて ACL を適用できます。宛先のドメイン名はその後、DNS 応答の一部としてクライアントに提供される IP アドレスに解決されます。

ゲスト ユーザは、FQDN ACL 名で構成されるパラメータ マップでネットワーク認証を使用してログインできます。

FQDN ACL を設定する前に、次の作業を実行してください。

- IP アクセス リストを設定します。
- IP ドメイン名のリストを設定します。
- ドメイン名と FQDN ACL をマッピングします。

コントローラに **fqdn-acl-name** AAA 属性を送信するように RADIUS サーバを設定して、アクセス リストを特定のドメインに適用できます。オペレーティング システムは、パススルー ドメイン リストとそのマッピングを確認し、FQDN を許可します。FQDN ACL により、クライアントは認証なしで設定されたドメインのみにアクセスできます。

（注） デフォルトでは、IPアクセスリスト名は、パススルードメイン名と同じ名前で設定されます。デフォルト名を上書きするために、グローバル コンフィギュレーション モードで**access-session passthrou-access-group** *access-group-name* **passthrou-domain-list** *domain-list-name* コマンドを使用できます。

# FQDN ACL の設定方法

## IP アクセス リストの設定

手順の概要

1. **configure terminal**
2. **ip access-list extended** *name*
3. **permit ip any any**
4. **end**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **configure terminal**<br>例：<br>`# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 2 | **ip access-list extended** *name*<br>例：<br>`(config)# ip access-list extended ABC` | IP アクセス リストを作成します。 |
| ステップ 3 | **permit ip any any**<br>例：<br>`(config-ext-nacl)# permit ip any any` | ワイヤレス クライアントに許可されるドメインを指定します。ドメインはドメイン名リストで指定されます。 |
| ステップ 4 | **end**<br>例：<br>`(config)# end` | 特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。 |

# ドメイン名リストの設定

アクセス ポイントによる DNS スヌーピングが許可されたドメイン名のリストを含むドメイン名リストを設定できます。DNS ドメイン リスト名の文字列は、拡張アクセス リスト名と一致している必要があります。

手順の概要

1. **configure terminal**
2. **passthrou-domain-list** *name*
3. **match** *word*
4. **end**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **configure terminal**<br>例：<br>`# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 2 | **passthrou-domain-list** *name*<br>例：<br>`(config)# passthrou-domain-list abc`<br>`(config-fqdn-acl-domains)#` | パススルー ドメイン名リストを設定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 3 | **match** *word*<br>例：<br>`(config-fqdn-acl-domains)# match play.google.com`<br>`(config-fqdn-acl-domains)# match www.yahoo.com` | パススルー ドメイン リストを設定します。クライアントがRADIUSサーバを介して認証される必要なくアクセスの照会が許可されるWebサイトのリストを追加します。 |
| ステップ 4 | **end**<br>例：<br>`(config)# end` | 特権 EXEC モードに戻ります。また、Ctrl+Z キーを押しても、グローバル コンフィギュレーション モードを終了できます。 |

# ドメイン名と FQDN ACL のマッピング

## 手順の概要

1. **configure terminal**
2. **access-session passthrou-access-group** *access-group-name* **passthrou-domain-list** *domain-list-name*
3. **parameter-map type webauth** *domain-list-name* and **login-auth-bypass fqdn-acl-name** *acl-name* **domain-name** *domain-name*

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **configure terminal**<br>例：<br>`# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 2 | **access-session passthrou-access-group** *access-group-name* **passthrou-domain-list** *domain-list-name*<br>例：<br>`(config)# access-session passthrou-access-group abc passthrou-domain-list abc` | ドメイン名リストと FQDN ACL AAA 属性名をマッピングします。中央Web認証を設定する場合、このコマンドを使用します。 |
| ステップ 3 | **parameter-map type webauth** *domain-list-name* and **login-auth-bypass fqdn-acl-name** *acl-name* **domain-name** *domain-name*<br>例：<br>`(config)# parameter-map type webauth abc`<br>`(config-params-parameter-map)# login-auth-bypass fqdn-acl-name abc domain-name abc` | ドメイン名リストと FQDN ACL 名をマッピングします。コントローラでローカル認証を設定する場合、このコマンドを使用します。<br>RADIUSサーバは、認証されたユーザプロファイルの一部として FQDN ACL 名を返すように設定できます。FQDN ACL がコントローラで定義される場合、コントローラは FQDN ACL をユーザに動的に適用します。 |

# FQDN ACL のモニタリング

次のコマンドを使用して FQDN ACL をモニタできます。

| コマンド | 目的 |
|---|---|
| **show access-session interface** *interface-name* **details** | インターフェイスに設定された FQDN ACL 情報を表示します。 |
| **show access-session fqdn fqdn-maps** | ドメイン名リストにマッピングされた FQDN ACL を表示します。 |
| **show access-session fqdn list-domain** *domain-name* | ドメイン名を表示します。 |
| **show access-session fqdn passthru-domain-list** | 設定されているドメインを表示します。 |

# FQDN ACL の設定例

## 例：FQDN ACL の設定

次に、IP アクセス リストを作成する例を示します。

```
# config terminal
(config)# ip access-list extended abc
(config-ext-nacl)# permit ip any any
(config-ext-nacl)# end
# show ip access-list abc
```

次に、ドメイン名のリストを設定する例を示します。

```
# config terminal
(config)# passthrou-domain-list abc
(config-fqdn-acl-domains)# match play.google.com
(config-fqdn-acl-domains)# end
# show access-session fqdn fqdn-maps
```

次に、中央集中型 Web 認証を使用してドメイン名と FQDN ACL をマッピングする例を示します。

```
# config terminal
(config)# access-session passthrou-access-group abc passthrou-domain-list abc
(config)# end
# show access-session interface vlan 20
```

次に、ローカル認証を使用してドメイン名と FQDN ACL をマッピングする例を示します。

```
# config terminal
(config)# parameter-map type webauth abc
```

```
(config-params-parameter-map)# login-auth-bypass fqdn-acl-name abc domain-name abc
(config-params-parameter-map)# end
# show access-session fqdn fqdn-maps
```

# FQDN ACL の設定に関するその他の参考資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | Cisco IOS Master Command List, All Releases |
| セキュリティ コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |
| ACL 設定ガイド | 『*Security Configuration Guide: Access Control Lists*』 |

### シスコのテクニカル サポート

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

# FQDN ACL の設定に関する機能情報

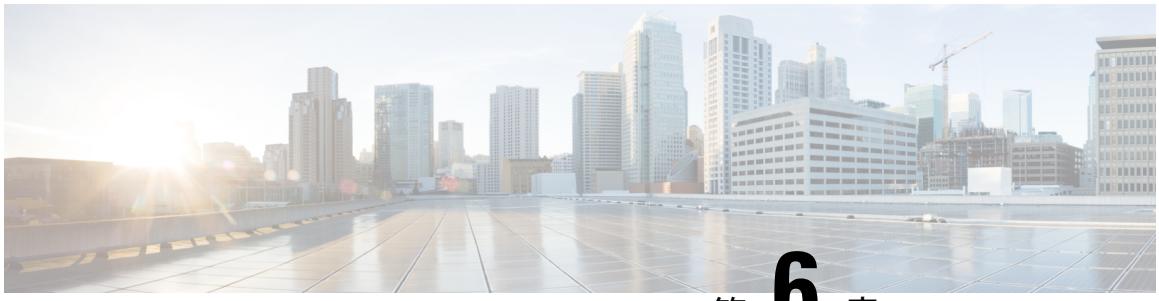
次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだ

けを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

表 *6 : FQDN ACL* の設定に関する機能情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| FQDN ACL の設定 | | FQDN ACL 機能を設定することで、ドメイン名システム（DNS）に基づいてワイヤレス セッションにアクセス コントロール リスト（ACL）を設定、適用することができます。ドメイン名が IP アドレスが DNS 応答の一部として、クライアントに割り当てるられる IP アドレスに解決されます。次に FQDN が IP アドレスに基づいて ACL にマッピングされます。<br><br>次のコマンドが導入または変更されました。**access session passthrou access group**、**login-auth-bypass**、**parameter-map type webauth global**、 **pass throu domain list name**、**show access-session fqdn** |



第 6 章

# IP アクセス リストの精緻化

アクセス リストを作成している間、または作成した後に、アクセス リストを精緻化するにはいくつかの方法があります。アクセス リストのエントリの順序を変更したり、アクセス リストにエントリを追加したりできます。また、アクセス リスト エントリを日または週の特定の時間帯に制限したり、パケットの非初期フラグメントをフィルタリングすることでパケットをフィルタリングするときにより細かく設定することができます。



## 機能情報の確認

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

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

# IP アクセス リストの精緻化に関する情報

## アクセス リストのシーケンス番号

IP アクセス リスト エントリにシーケンス番号を適用する機能によって、アクセス リストの変更が簡易になります。IP アクセス リスト エントリ シーケンス番号機能の前には、アクセス リスト内のエントリの位置を指定する方法はありませんでした。以前は、既存のリストの途中にエントリを挿入する場合、目的の位置の後にあるすべてのエントリを削除してから、新しいエントリを追加し、削除したすべてのエントリを再入力する必要がありました。これは手間がかかり、エラーが起こりやすい方法です。

シーケンス番号を使用して、ユーザはアクセス リスト エントリを追加し、それを並べ替えることができるようになりました。新しいエントリを追加する場合、アクセスリストの目的の位置に挿入されるようにシーケンス番号を指定します。必要に応じて、アクセスリストの現在のエントリを並べ替えて、新しいエントリを挿入できる場所を作成できます。

## アクセス リスト シーケンス番号の利点

アクセス リスト シーケンス番号は、アクセス リストで **permit** または **deny** コマンドを開始する番号です。シーケンス番号により、エントリがアクセスリストに表示される順序が決定されます。IP アクセス リスト エントリにシーケンス番号を適用する機能によって、アクセス リストの変更が簡易になります。

シーケンス番号を設定する前に、アクセス リストの末尾にアクセス リスト エントリを追加できるため、アクセスリスト全体の再設定が必要になるリストの末尾以外の位置では、ステートメントの追加が必要になります。アクセスリスト内でのエントリの位置を指定する方法はありません。以前は、既存のリストの途中にエントリ（ステートメント）を挿入する場合、目的の位置の後にあるすべてのエントリを削除してから、新しいエントリを追加し、削除したすべてのエントリを再入力する必要がありました。これは手間がかかり、エラーが起こりやすい方法です。

この新しい機能を使用すると、アクセス リスト エントリにシーケンス番号を追加し、順序を変更することができます。新しいエントリを追加するとき、アクセスリストの目的の位置に配置されるように、シーケンス番号を選択します。必要に応じて、アクセスリストの現在のエントリを並べ替えて、新しいエントリを挿入できる場所を作成できます。シーケンス番号により、アクセス リストの変更を簡単に実行できるようになりました。

## シーケンス番号の動作

- 以前のリリースとの下位互換性を保つため、シーケンス番号のないエントリが適用された場合には、最初のエントリにはシーケンス番号 10 が割り当てられます。連続してエントリを追加すると、シーケンス番号は10ずつ増分されます。最大シーケンス番号は2147483647です。生成したシーケンス番号がこの最大値を超えると、次のメッセージが表示されます。

```
Exceeded maximum sequence number.
```

- シーケンス番号のないエントリを入力すると、アクセス リストの最後のシーケンス番号に 10 を加えたシーケンス番号が割り当てられ、リストの末尾に配置されます。
- （シーケンス番号以外が）既存のエントリに一致するエントリを入力すると、何も変更されません。
- 既存のシーケンス番号を入力すると、次のエラー メッセージが表示されます。

```
Duplicate sequence number.
```

- グローバル コンフィギュレーション モードで新しいアクセス リストを入力すると、そのアクセス リストのシーケンス番号が自動的に生成されます。
- シーケンス番号が不揮発性生成（NVGEN）されることはありません。つまり、シーケンス番号自体は保存されません。システムのリロード時には、設定されたシーケンス番号はデフォルトのシーケンス開始番号と増分に戻されます。この機能は、シーケンス番号をサポートしないソフトウェア リリースとの下位互換性を保つために提供されています。
- この機能は、名前付きおよび番号付きの標準および拡張IPアクセスリストと連動します。

# 時間範囲の利点

時間範囲の利点および可能な使用法として、次のことが挙げられます。

- ネットワーク管理者は、リソースへのユーザアクセスの許可または拒否の制御をより強化できます。これらのリソースとして、アプリケーション（IP アドレス/マスク ペアとポート番号によって特定されます）、ポリシールーティング、またはオンデマンドリンク（ダイヤラへの関連トラフィックとして認識されます）があります。
- ネットワーク管理者は、次に示すような、時刻ベースのセキュリティ ポリシーを設定できます。
  - アクセス リストを使用した境界セキュリティ
  - IP セキュリティ プロトコル（IPsec）を使用したデータの機密性保持
- プロバイダーのアクセス レートが一日の時間帯によって異なるときは、トラフィックは自動的にコスト効率よく再ルーティングすることが可能です。
- ネットワーク管理者は、ロギング メッセージを制御できます。アクセス リスト エントリは、一日の特定の時間帯にトラフィックをロギングすることはできますが、常にロギングすることはできません。したがって、管理者はピーク時間中に生成された多くのログを分析することなく、単にアクセスを拒否できます。

# パケットの非初期フラグメントをフィルタリングする利点

パケットの初期フラグメントにとどまらず、より多くのトラフィックをブロックするには、拡張アクセスリストを使用してパケットの非初期フラグメントをフィルタリングします。まず、次の概念を理解しておく必要があります。

フラグメントを拒否する追加の IP アクセス リスト エントリで **fragments** キーワードが使用されている場合、フラグメント制御機能を使用すると、次のような利点があります。

### 追加のセキュリティ

パケットの初期フラグメントにとどまらず、より多くのトラフィックをブロックできます。不要なフラグメントは、受信側にリアセンブリ タイムアウトになるまで残りません。これは、このようなフラグメントは受信側に送信される前にブロックされるためです。不要なトラフィックを大量にブロックすることで、セキュリティが高まり、ハッカーから攻撃を受けるリスクが軽減されます。

### コスト削減

パケットの不要な非初期フラグメントをブロックすると、ブロックしたいトラフィックに注意を払う必要がなくなります。

### 使用ストレージの削減

パケットの不要な非初期フラグメントが受信側に届かないようにブロックすることで、宛先はリアセンブリ タイムアウトになるまでフラグメントを保存する必要がなくなります。

### 予期される動作

非初期フラグメントは、初期フラグメントと同様に扱われます。予期されないポリシー ルーティング結果や、ルーティングされるべきでないパケットのフラグメントが生じる可能性も低くなります。

# フラグメントのアクセス リスト処理

**fragments** キーワードを指定するかどうかによるアクセス リスト エントリの動作は、次のようにまとめることができます。

| アクセス リスト エントリの状態... | 結果 |
|---|---|
| ...**fragments** キーワードが指定されず（デフォルト）、すべてのアクセス リスト エントリ情報が一致する | レイヤ 3 情報のみを含むアクセス リスト エントリの場合：<br>• エントリは、非フラグメント パケット、先頭フラグメント、先頭以外のフラグメントに適用されます。<br>レイヤ 3 およびレイヤ 4 情報を含むアクセス リスト エントリの場合：<br>• エントリは、非フラグメント パケットと先頭フラグメントに適用されます。<br>• エントリが **permit** ステートメントであると、パケットまたはフラグメントは許可されます。<br>• エントリが **deny** ステートメントであると、パケットまたはフラグメントは拒否されます。<br>• エントリは、次の方法で先頭以外のフラグメントにも適用されます。非初期フラグメントにはレイヤ 3 情報のみが含まれているため、アクセス リスト エントリのレイヤ 3 の部分のみが適用されます。アクセス リスト エントリのレイヤ 3 の部分が一致し、<br>• エントリが **permit** ステートメントであると、非初期フラグメントは許可されます。<br>• エントリが **deny** ステートメントであると、次のアクセス リスト エントリが処理されます。<br>（注） 非初期フラグメントと、非フラグメントまたは初期フラグメントの場合では、**deny** ステートメントの処理方法は異なります。 |
| ...**fragments** キーワードが指定され、すべてのアクセス リスト エントリ情報が一致する | アクセス リスト エントリは、非初期フラグメントにのみ適用されます。<br>レイヤ 4 情報を含むアクセス リスト エントリに **fragments** キーワードは設定できません。 |

すべてのアクセス リスト エントリに **fragments** キーワードを追加することはできません。IP パケットの最初のフラグメントは非フラグメントとして見なされ、以降のフラグメントとは独立して扱われるためです。初期フラグメントは、アクセスリストの **fragments** キーワードが設定された **permit** または **deny** エントリとは一致しません。パケットは、**fragments** キーワードが設定されていないアクセス リスト エントリによって許可または拒否されるまで、次のアクセス リスト エントリと比較されます。したがって、**deny** エントリごとに、2 つのアクセス リスト エントリが必要になる場合があります。ペアの最初の **deny** エントリには **fragments** キーワードは含まれず、初期フラグメントに適用されます。ペアの 2 番目の **deny** エントリには

**fragments** キーワードは含まれ、以降のフラグメントに適用されます。同じホストに複数の **deny** エントリがあり、レイヤ 4 ポートが異なる場合は、そのホストで **fragments** キーワードが設定された 1 つの **deny** アクセス リスト エントリを追加する必要があります。このように、パケットのすべてのフラグメントは、アクセス リストによって同様に扱われます。

IP データグラムのパケット フラグメントは個々のパケットと見なされ、それぞれ、アクセス リスト アカウンティングとアクセス リストの違反カウントの 1 つのパケットとして個別にカウントされます。

# IP アクセス リストを精緻化する方法

このモジュールで説明する作業では、アクセス リストを精緻化するためのさまざまな方法を示します（アクセス リストを作成するときに精緻化しなかった場合に利用できます）。アクセス リスト エントリの順序変更、アクセス リストへのエントリの追加、日または週の特定の時間帯でのアクセス リスト エントリの制限などを実行できます。また、パケットの非初期フラグメントをフィルタリングすることでパケットをフィルタリングするときにより細かく設定することができます。

## シーケンス番号を使用したアクセス リストの変更

既存のアクセス リストへのエントリの追加、エントリの順序変更、または（将来の変更に対応するための）アクセス リストのエントリの番号付けを行うには、次の手順を実行します。

（注） アクセス リストからエントリを削除する場合は、コマンドの **no deny** または **no permit** 形式を使用するか、あるいはステートメントにシーケンス番号がすでに指定されている場合は **no** *sequence-number* コマンドを使用するだけです。

（注）
- アクセス リスト シーケンス番号は、ダイナミック、リフレクシブ、またはファイアウォールのアクセス リストをサポートしていません。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list resequence** *access-list-name starting-sequence-number increment*
4. **ip access-list** {**standard**| **extended**} *access-list-name*
5. 次のいずれかを実行します。
   - *sequence-number* **permit** *source source-wildcard*
   - *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]

6. 次のいずれかを実行します。
   - *sequence-number* **deny** *source source-wildcard*
   - *sequence-number* **deny** *protocol source source-wildcard destination destination-wildcard* [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]
7. 必要に応じてステップ 5 とステップ 6 を繰り返し、目的とするシーケンス番号順にステートメントを追加します。エントリを削除するには、**no** *sequence-number* コマンドを使用します。
8. **end**
9. **show ip access-lists** *access-list-name*

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Router> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Router# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip access-list resequence** *access-list-name starting-sequence-number increment*<br>例：<br>`Router(config)# ip access-list resequence kmd1 100 15` | 開始シーケンス番号と、シーケンス番号の増分を使用して、指定した IP アクセス リストを並べ替えます。<br>• この例では、kmd1 という名前のアクセス リストを並べ替えます。開始シーケンス番号は100、増分は 15 です。 |
| ステップ 4 | **ip access-list** {**standard**\| **extended**} *access-list-name*<br>例：<br>`Router(config)# ip access-list standard xyz123` | 名前で IP アクセス リストを指定し、名前付きアクセス リストのコンフィギュレーション モードを開始します。<br>• **standard** を指定する場合は、その後に、標準アクセス リスト構文を使用して **permit** ステートメントまたは **deny** ステートメントを指定します。<br>• **extended** を指定する場合は、その後に、拡張アクセス リスト構文を使用して **permit** ステートメントまたは **deny** ステートメントを指定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| **ステップ 5** | 次のいずれかを実行します。<br>• *sequence-number* **permit** *source source-wildcard*<br>• *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]<br>例：<br>`Router(config-std-nacl)# 105 permit 10.5.5.5 0.0.0.255` | 名前付き IP アクセス リスト モードで permit ステートメントを指定します。<br>• このアクセス リストでは **permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。<br>• 上位層プロトコル（ICMP、IGMP、TCP、およびUDP）を許可するその他のコマンド構文については、**permit**（IP）コマンドを参照してください。<br>• エントリを削除するには、**no** *sequence-number* コマンドを使用します。<br>• プロンプトに示されるとおり、このアクセス リストは標準アクセス リストでした。ステップ 4 で **extended** を指定した場合は、このステップのプロンプトは Router(config-ext-nacl)# となり、拡張 **permit** コマンド構文を使用します。 |
| **ステップ 6** | 次のいずれかを実行します。<br>• *sequence-number* **deny** *source source-wildcard*<br>• *sequence-number* **deny** *protocol source source-wildcard destination destination-wildcard* [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]<br>例：<br>`Router(config-std-nacl)# 110 deny 10.6.6.7 0.0.0.255` | （任意）名前付き IP アクセス リスト モードで deny ステートメントを指定します。<br>• このアクセス リストでは **permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。<br>• 上位層プロトコル（ICMP、IGMP、TCP、およびUDP）を許可するその他のコマンド構文については、**deny**（IP）コマンドを参照してください。<br>• エントリを削除するには、**no** *sequence-number* コマンドを使用します。<br>• プロンプトに示されるとおり、このアクセス リストは標準アクセス リストでした。ステップ 4 で **extended** を指定した場合は、このステップのプロンプトは Router(config-ext-nacl)# となり、拡張 **deny** コマンド構文を使用します。 |
| **ステップ 7** | 必要に応じてステップ 5 とステップ 6 を繰り返し、目的とするシーケンス番号順にステートメントを追 | アクセス リストは変更できます。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | 加します。エントリを削除するには、**no** *sequence-number* コマンドを使用します。 | |
| ステップ 8 | **end**<br>例：<br>`Router(config-std-nacl)# end` | （任意）コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |
| ステップ 9 | **show ip access-lists** *access-list-name*<br>例：<br>`Router# show ip access-lists xyz123` | （任意）IP アクセス リストの内容を表示します。<br>• 出力を見直して、アクセス リストに新しいエントリが含まれることを確認します。 |

**例**

次に、**xyz123** アクセス リストを指定した場合の **show ip access-lists** コマンドの出力例を示します。

```
Router# show ip access-lists xyz123
Standard IP access list xyz123
100 permit 10.4.4.0, wildcard bits 0.0.0.255
105 permit 10.5.5.5, wildcard bits 0.0.0.255
115 permit 10.0.0.0, wildcard bits 0.0.0.255
130 permit 10.5.5.0, wildcard bits 0.0.0.255
145 permit 10.0.0.0, wildcard bits 0.0.0.255
```

# 日または週の特定の時間帯でのアクセス リスト エントリの制限

デフォルトで、アクセス リスト ステートメントは適用されたときに実行されます。ただし、時間範囲を定義し、各アクセス リスト ステートメントにおいて名前ごとに時間範囲を参照することで、**permit** ステートメントまたは **deny** ステートメントが有効になる日または週の時間帯を定義できます。IP および Internetwork Packet exchange（IPX）名前付きまたは番号付きの拡張アクセス リストは、時間範囲に対応します。

## 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *name*
4. [*sequence-number*] **deny** *protocol source*[*source-wildcard*] [*operator port*[*port*]] *destination*[*destination-wildcard*] [*operator port*[*port*]]
5. [*sequence-number*] **deny** *protocol source*[*source-wildcard*][*operator port*[*port*]] *destination*[*destination-wildcard*] [*operator port*[*port*]] **fragments**
6. [*sequence-number*] **permit** *protocol source*[*source-wildcard*] [*operator port*[*port*]] *destination*[*destination-wildcard*] [*operator port*[*port*]]

7. アクセス リストの基本となる値を指定するまで、ステップ 4 ～ 6 を適宜組み合わせて繰り返します。
8. **end**
9. **show ip access-list**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Router> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Router# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip access-list extended** *name*<br>例：<br>`Router(config)# ip access-list extended rstrct4` | 名前を使用して拡張 IP アクセス リストを定義し、拡張名前付きアクセス リストのコンフィギュレーション モードを開始します。 |
| ステップ 4 | [*sequence-number*] **deny** *protocol source*[*source-wildcard*] [*operator port*[*port*]] *destination*[*destination-wildcard*] [*operator port*[*port*]]<br>例：<br>`Router(config-ext-nacl)# deny ip any 172.20.1.1` | （任意）ステートメントに指定されたすべての条件に一致するすべてのパケットを拒否します。<br>• このステートメントは、非フラグメント パケットと初期フラグメントに適用されます。 |
| ステップ 5 | [*sequence-number*] **deny** *protocol source*[*source-wildcard*][*operator port*[*port*]] *destination*[*destination-wildcard*] [*operator port*[*port*]] **fragments**<br>例：<br>`Router(config-ext-nacl)# deny ip any 172.20.1.1 fragments` | （任意）ステートメントに指定されたすべての条件に一致するすべてのパケットを拒否します。<br>• このステートメントは、非初期フラグメントに適用されます。 |
| ステップ 6 | [*sequence-number*] **permit** *protocol source*[*source-wildcard*] [*operator port*[*port*]] *destination*[*destination-wildcard*] [*operator port*[*port*]]<br>例：<br>`Router(config-ext-nacl)# permit tcp any any` | ステートメントに指定されたすべての条件に一致するすべてのパケットを許可します。<br>• 各アクセス リストには、少なくとも 1 つの **permit** ステートメントが必要です。<br>• *source-wildcard* または *destination-wildcard* を省略すると、0.0.0.0 のワイルドカード マスクが想定されます。つまり、それぞれ送信元アドレス |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | または宛先アドレスの全ビットへの一致を意味します。<br>• 必要に応じて、*source source-wildcard* または *destination destination-wildcard* の代わりに、キーワード **any** を使用して、アドレスと 0.0.0.0 255.255.255.255 の送信元ワイルドカードを指定できます。 |
| **ステップ 7** | アクセス リストの基本となる値を指定するまで、ステップ 4 ～ 6 を適宜組み合わせて繰り返します。 | 明示的に許可されていないすべての送信元は、アクセス リストの末尾にある暗黙的な **deny** ステートメントで拒否されます。 |
| **ステップ 8** | **end**<br>例：<br>`Router(config-ext-nacl)# end` | コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |
| **ステップ 9** | **show ip access-list**<br>例：<br>`Router# show ip access-list` | （任意）現在の IP アクセス リストすべてのコンテンツが表示されます。 |

## 次の作業

アクセス リストをインターフェイスに適用するか、アクセス リストを受け入れるコマンドから参照します。

（注） IP オプションを含むすべてのパケットを効率的に除去するには、**ip options drop** グローバル コマンドを設定することを推奨します。

# IP アクセス リストの精緻化の設定例

## 例：アクセス リストのエントリの並べ替え

次に、並べ替える前と後のアクセス リストの例を示します。開始値は 1、増分値は 2 です。後続のエントリはユーザ指定の増分値に基づいて並べられています。範囲は 1 ～ 2147483647 です。

シーケンス番号のないエントリが入力されると、デフォルトで、アクセス リストの最後のエントリのシーケンス番号に 10 を加えたシーケンス番号が割り当てられます。

```
Router# show access-list carls
Extended IP access list carls
    10 permit ip host 10.3.3.3 host 172.16.5.34
    20 permit icmp any any
    30 permit tcp any host 10.3.3.3
    40 permit ip host 10.4.4.4 any
    50 Dynamic test permit ip any any
    60 permit ip host 172.16.2.2 host 10.3.3.12
    70 permit ip host 10.3.3.3 any log
    80 permit tcp host 10.3.3.3 host 10.1.2.2
    90 permit ip host 10.3.3.3 any
    100 permit ip any any
Router(config)# ip access-list extended carls
Router(config)# ip access-list resequence carls 1 2
Router(config)# end
Router# show access-list carls
Extended IP access list carls
    1 permit ip host 10.3.3.3 host 172.16.5.34
    3 permit icmp any any
    5 permit tcp any host 10.3.3.3
    7 permit ip host 10.4.4.4 any
    9 Dynamic test permit ip any any
    11 permit ip host 172.16.2.2 host 10.3.3.12
    13 permit ip host 10.3.3.3 any log
    15 permit tcp host 10.3.3.3 host 10.1.2.2
    17 permit ip host 10.3.3.3 any
    19 permit ip any any
```

# 例：シーケンス番号を指定したエントリの追加

次の例では、新しいエントリ（シーケンス番号 15）がアクセス リストに追加されます。

```
Router# show ip access-list
Standard IP access list tryon
2 permit 10.4.4.2, wildcard bits 0.0.255.255
5 permit 10.0.0.44, wildcard bits 0.0.0.255
10 permit 10.0.0.1, wildcard bits 0.0.0.255
20 permit 10.0.0.2, wildcard bits 0.0.0.255
Router(config)# ip access-list standard tryon
Router(config-std-nacl)# 15 permit 10.5.5.5 0.0.0.255
Router# show ip access-list
Standard IP access list tryon
2 permit 10.4.0.0, wildcard bits 0.0.255.255
5 permit 10.0.0.0, wildcard bits 0.0.0.255
10 permit 10.0.0.0, wildcard bits 0.0.0.255
15 permit 10.5.5.0, wildcard bits 0.0.0.255
20 permit 10.0.0.0, wildcard bits 0.0.0.255
```

# 例：シーケンス番号を指定しないエントリの追加

次に、シーケンス番号が指定されていないエントリをアクセス リストの末尾に追加する方法を示します。シーケンス番号のないエントリを追加すると、自動的にシーケンス番号が割り当てられ、アクセス リストの末尾に配置されます。デフォルトの増分値は 10 であるため、エントリには、既存のアクセス リストの最後のエントリのシーケンス番号に 10 を加えたシーケンス番号が割り当てられます。

```
Router(config)# ip access-list standard resources
Router(config-std-nacl)# permit 10.1.1.1 0.0.0.255
Router(config-std-nacl)# permit 10.2.2.2 0.0.0.255
Router(config-std-nacl)# permit 10.3.3.3 0.0.0.255
Router# show access-list
Standard IP access list resources
10 permit 10.1.1.1, wildcard bits 0.0.0.255
20 permit 10.2.2.2, wildcard bits 0.0.0.255
30 permit 10.3.3.3, wildcard bits 0.0.0.255
Router(config)# ip access-list standard resources
Router(config-std-nacl)# permit 10.4.4.4 0.0.0.255
Router(config-std-nacl)# end
Router# show access-list
Standard IP access list resources
10 permit 10.1.1.1, wildcard bits 0.0.0.255
20 permit 10.2.2.2, wildcard bits 0.0.0.255
30 permit 10.3.3.3, wildcard bits 0.0.0.255
40 permit 10.4.4.4, wildcard bits 0.0.0.255
```

# 例：IP アクセス リスト エントリに適用された時間範囲

次の例では、月曜日 ～ 金曜日の 8:00 am ～ 6:00 p.m に延長した、no-http と呼ばれる時間範囲を作成します。この時間帯は **deny** ステートメントに適用されるため、月曜日 ～ 金曜日の 8:00 am ～ 6:00 p.m の HTTP トラフィックが拒否されます。

udp-yes と呼ばれる時間範囲は、正午から 8:00 p.m までの週末を定義します。この時間範囲は **permit** ステートメントに適用されるため、土曜日 ～ 日曜日の正午から 8:00 p.m の UDP トラフィックのみが許可されます。両方のステートメントを含むアクセス リストは、ファストイーサネット インターフェイス 0/0/0 のインバウンド パケットに適用されます。

```
time-range no-http
 periodic weekdays 8:00 to 18:00
!
time-range udp-yes
 periodic weekend 12:00 to 20:00
!
ip access-list extended strict
 deny tcp any any eq http time-range no-http
 permit udp any any time-range udp-yes
!
interface fastethernet 0/0/0
 ip access-group strict in
```

# 例：IP パケット フラグメントのフィルタリング

次のアクセス リストでは、最初のステートメントはホスト 172.16.1.1 を宛先とする非初期フラグメントのみを拒否します。2 番目のステートメントは、ホスト 172.16.1.1 の TCP ポート 80 を宛先とする残りの非フラグメントと初期フラグメントのみを許可します。3 番目のステートメントは、その他のすべてのトラフィックを拒否します。すべての TCP ポートで非初期フラグメントをブロックするため、ホスト 172.16.1.1 のポート 80 をはじめとするすべての TCP ポートで非初期フラグメントをブロックする必要があります。つまり、非初期フラグメントにはレイヤ 4 ポート情報は含まれないため、指定のポートで該当するトラフィックをブロックするには、すべてのポートのフラグメントをブロックする必要があります。

```
access-list 101 deny ip any host 172.16.1.1 fragments
access-list 101 permit tcp any host 172.16.1.1 eq 80
access-list 101 deny ip any any
```

# その他の参考資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Commands List, All Releases』 |
| **time-range** コマンドを使用した時間範囲の指定 | 『*Cisco IOS XE Network Management Configuration Guide*』の「Performing Basic System Management」章 |
| ネットワーク管理コマンドの説明 | 『*Cisco IOS Network Management Command Reference*』 |

### 標準

| 標準 | タイトル |
|---|---|
| この機能でサポートされる新規の規格または変更された規格はありません。また、既存の規格のサポートは変更されていません。 | -- |

### MIB

| MIB | MIB のリンク |
|---|---|
| この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。 | 選択したプラットフォーム、Cisco IOS XE ソフトウェア リリース、およびフィーチャ セットの MIB の場所を検索しダウンロードするには、次の URL にある Cisco MIB Locator を使用します。<br>http://www.cisco.com/go/mibs |

### RFC

| RFC | タイトル |
|---|---|
| この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。 | -- |

### テクニカル サポート

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

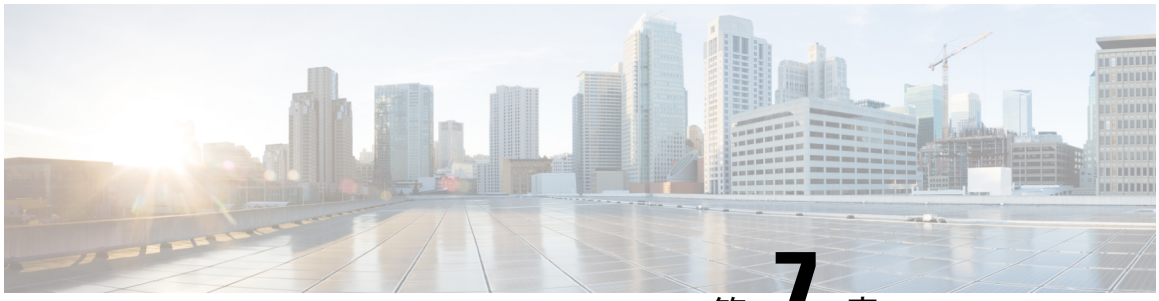
# IP アクセス リストの精緻化に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *7 : IP* アクセス リストの精緻化に関する機能情報**

| 機能名 | リリース | 機能の設定情報 |
|---|---|---|
| 時刻ベースのアクセス リスト | Cisco IOS XE リリース 2.1 | この機能は、Cisco ASR 1000 シリーズのアグリゲーション サービス ルータで導入されました。<br>この機能について導入または変更されたコマンドはありません。 |



第 7 章

# IP 名前付きアクセス コントロール リスト

アクセス コントロール リスト（ACL）は、ネットワークを通過するパケットの動きを制御するためにパケット フィルタリングを実行します。パケット フィルタリングは、ネットワークへのトラフィックのアクセスを限定し、ユーザおよびデバイスのネットワークに対するアクセスを制限し、トラフィックがネットワークから流出するのを防ぐことで、セキュリティを提供します。IPアクセスリストによって、スプーフィングやサービス妨害攻撃の可能性を軽減し、ファイアウォールを介した動的で一時的なユーザ アクセスが可能になります。

IP 名前付きアクセス コントロール リスト機能により、ネットワーク管理者は、管理するアクセス リストを識別するための名前を使用することができます。

このモジュールでは、IP 名前付きアクセス コントロール リスト、およびその設定方法について説明します。



## 機能情報の確認

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

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

# IP 名前付きアクセス コントロール リストに関する情報

## アクセス リストの定義

アクセス コントロール リスト（ACL）は、ネットワークを通過するパケットの動きを制御するためにパケット フィルタリングを実行します。パケット フィルタリングは、ネットワークへのトラフィックのアクセスを限定し、ユーザおよびデバイスのネットワークに対するアクセスを制限し、トラフィックがネットワークから流出するのを防ぐことで、セキュリティを提供します。IPアクセスリストによって、スプーフィングやサービス妨害攻撃の可能性を軽減し、ファイアウォールを介した動的で一時的なユーザ アクセスが可能になります。

また、IP アクセス リストは、セキュリティ以外の用途にも使用できます。たとえば、帯域幅制御、ルーティング アップデートのコンテンツの制限、ルートの再配布、ダイヤルオンデマンド（DDR）呼び出しのトリガー、デバッグ出力の制限、Quality of Service（QoS）機能のトラフィックの識別と分類などです。

アクセス リストは、少なくとも 1 つの **permit** ステートメント、および任意の 1 つまたは複数の **deny** ステートメントで構成される順次リストです。IP アクセス リストの場合、これらのステートメントは IP アドレス、上位層の IP プロトコルなどの IP パケットのフィールドに適用できます。

アクセスリストは名前または番号で識別および参照されます。アクセスリストはパケットフィルタとして動作し、各アクセスリストに定義されている条件に基づいてパケットがフィルタされます。

アクセスリストを構成した後でアクセスリストを有効にするには、アクセスリストをインターフェイスに適用するか（**ip access-group** コマンドを使用）、vty に適用するか（**access-class** コマンドを使用）、またはアクセス リストを許容するあらゆるコマンドでアクセス リストを参照する必要があります。複数のコマンドから同じアクセス リストを参照できます。

次の構成では、branchoffices という名前の IP アクセス リストがファスト イーサネット インターフェイス 0/1/0 上で構成され、着信パケットに適用されます。発信元アドレスとマスクのペアで指定されているネットワーク以外は、ファスト イーサネット インターフェイス 0/1/0 にアクセスできません。ネットワーク 172.16.7.0 上の送信元から発信されるパケットの宛先に、制限はありません。ネットワーク 172.16.2.0 上の送信元から発信されるパケットの宛先は、172.31.5.4 にする必要があります。

```
ip access-list extended branchoffices
 10 permit 172.16.7.0 0.0.0.3 any
 20 permit 172.16.2.0 0.0.0.255 host 172.31.5.4
!
interface fastethernet 0/1/0
 ip access-group branchoffices in
```

# 名前付きまたは番号付きアクセス リスト

すべてのアクセス リストは、名前または番号で識別されます。名前付きアクセス リストは、番号付きアクセスリストよりも便利です。タスクを思いだしやすく関連性がある、わかりやすい名前を指定できるためです。名前付きアクセスリストでは、ステートメントの順序を変更したり、ステートメントを追加したりできます。

名前付きアクセスリストは、番号付きアクセスリストではサポートされない次の機能をサポートします。

- IP オプションのフィルタリング
- 非隣接ポート
- TCP フラグ フィルタリング
- **no permit** または **no deny** コマンドによるエントリの削除

（注） 番号付きアクセス リストを受け入れるコマンドの中には、名前付きアクセス リストを受け入れないコマンドがあります。たとえば、vty には番号付きアクセス リストだけを使用します。

# IP アクセス リストの利点

アクセス コントロール リスト（ACL）は、ネットワークを通過するパケットのフローを制御するためにパケット フィルタリングを実行します。パケット フィルタリングによってユーザおよびデバイスのネットワークに対するアクセスを制限し、セキュリティの手段として利用できます。アクセス リストによってトラフィック数を減らすことで、ネットワーク リソースを節約できます。アクセス リストを使用した場合の利点は次のとおりです。

- 着信 rsh および rcp 要求を認証する：アクセス リストは、デバイスへのアクセスを制御するように構成された認証データベース内のローカル ユーザ、リモート ホスト、およびリモート ユーザの識別を簡素化できます。Cisco ソフトウェアは認証データベースを使用して、リモート シェル（rsh）およびリモート コピー（rcp）プロトコルの着信要求を受け取ることができます。
- 不要なトラフィックまたはユーザをブロックする：アクセス リストを使用すると、インターフェイス上の着信パケットまたは発信パケットをフィルタできるため、送信元アドレス、宛先アドレス、またはユーザ認証に基づいてネットワークへのアクセスを制御できます。また、アクセス リストを使用して、デバイス インターフェイスで転送またはブロックするトラフィックの種類を決定することもできます。たとえば、電子メール トラフィックはネットワークでルーティングすることを許可し、すべての Telnet トラフィックはネットワークに入ることをブロックするようにアクセス リストを使用できます。
- vty へのアクセスを制御する：インバウンド vty（Telnet）でのアクセス リストは、デバイスへの回線にアクセスできるユーザを制御できます。アウトバウンド vty でのアクセス リストは、デバイスからの回線が到達可能な宛先を制御できます。

- QoS 機能のトラフィックを特定または分類する：アクセス リストは、Weighted Random Early Detection（WRED）および専用アクセス レート（CAR）の IP プレシデンスを設定することで、輻輳回避を提供します。また、クラスベース均等化キューイング（CBWFQ）、プライオリティ キューイング、カスタム キューイングのために輻輳管理を提供します。
- debug コマンド出力を制限する：アクセス リストは、IP アドレスやプロトコルに基づいて debug 出力を制限できます。
- 帯域幅制御を提供する：低速リンクでのアクセス リストはネットワークでの過剰なトラフィックを防止できます。
- NAT 制御を提供する：アクセス リストによって、ネットワーク アドレス変換（NAT）が変換するアドレスを制御できます。
- DoS 攻撃の可能性を低減する：アクセス リストは、サービス妨害（DoS）攻撃の可能性を低減させます。ホストからのトラフィック、ネットワーク、またはネットワークにアクセスするユーザを制御するように IP 発信元アドレスを指定します。TCP インターセプト機能を設定することで、接続に関する要求でサーバにフラッディングが発生しないようにすることができます。
- ルーティング アップデートの内容を制限する：アクセス リストによって、ネットワーク内で送信、受信、または再配布されるルーティング アップデートを制御できます。
- ダイヤルオンデマンド コールをトリガーする：アクセス リストによって、ダイヤルおよび切断条件を適用できます。

# アクセス リストのルール

アクセス リストには、次のルールが適用されます。

- 1 つのインターフェイス、1 つのプロトコル、1 つの方向につき、許可されるアクセス リストは 1 つだけです。
- アクセス リストには少なくとも 1 つの **permit** ステートメントが含まれる必要があります。そうしないと、ネットワークに入るすべてのパケットが拒否されます。
- アクセス リスト条件または一致基準の構成順序は重要です。パケットを転送するかブロックするかを決定するときに、Cisco ソフトウェアは、それぞれの条件ステートメントに対してステートメントの作成順にパケットをテストします。一致が見つかると、条件ステートメントはそれ以上チェックされません。同じ **permit** ステートメントまたは **deny** ステートメントでも、順序が異なる場合、ある状況では通過し、別の状況では拒否されるパケットが生じる可能性があります。
- アクセス リストを名前によって参照したときに、そのアクセス リストが存在しない場合は、すべてのパケットが通過します。インターフェイスまたはコマンドに空のアクセス リストを適用すると、ネットワークに対するすべてのトラフィックが許可されます。
- 標準のアクセス リストと拡張のアクセス リストの名前は同じにできません。

- パケットが発信インターフェイスにルーティングされる前に、着信アクセス リストがパケットを処理します。ネットワークへのパケットアクセスを拒否するフィルタ条件がある着信アクセス リストは、ルーティング ルックアップ時のオーバーヘッドを削減します。構成されたフィルタ基準に基づいてネットワークへのアクセスを許可されたパケットはルーティング処理されます。インバウンド アクセス リストの場合、**permit** ステートメントを構成するとパケットは受信後に処理され、**deny** ステートメントを構成するとパケットは破棄されます。
- 発信アクセス リストの場合、パケットの処理後にデバイスから送信されます。着信パケットは発信インターフェイスにルーティングされてから、発信アクセス リストで処理されます。アウトバウンド アクセス リストの場合、**permit** ステートメントを構成するとパケットは出力バッファに送信され、**deny** ステートメントを構成するとパケットは破棄されます。
- アクセス リストで、デバイスに到達するトラフィック、またはデバイス経由で送信されるトラフィックは制御できますが、デバイスが送信元のトラフィックは制御できません。

# IP アクセス リストを作成する際に役立つヒント

意図しない結果を回避し、より効率的で有効なアクセス リストを作成するために役立つヒントを紹介します。

- アクセス リストを作成してから、インターフェイス（または別の対象）に適用します。その理由は、存在しないアクセス リストをインターフェイスに適用してから、アクセス リストを設定すると、最初のステートメントが有効になり、それに続く暗黙的な **deny** ステートメントによってアクセスに緊急の問題が発生するおそれがあるためです。
- アクセス リストを設定してから適用するもう 1 つの理由は、空のアクセス リストが適用されたインターフェイスはすべてのトラフィックを許可するためです。
- すべてのアクセス リストには、少なくとも 1 つの **permit** ステートメントが必要です。permit がないと、すべてのパケットは拒否され、トラフィックはまったく通過しません。
- 最初に（**permit** または **deny** ステートメントに対する）一致が見つかった後は条件のテストが終了するため、パケットが一致する可能性の高いステートメントをアクセス リストの先頭に配置すると処理にかかる時間とリソースが削減されます。最も頻繁に発生する条件を発生頻度の低い条件より前に配置します。
- ネットワークまたはサブネットのより具体的な参照が、より全般的な参照よりも前に出現するように、アクセス リストを構成します。
- まだ拒否されていないその他のパケットすべてを許可する場合、ステートメント **permit any any** を使用します。ステートメント **permit any any** を使用すると、実質的に、アクセス リストの末尾にある暗黙的な deny ステートメントでその他すべてのパケットが拒否されることを防ぎます。最初のアクセス リスト エントリは **permit any any** にしないでください。すべてのトラフィックが通過し、以降のテストに到達するパケットがなくなります。**permit any any** を指定すると、まだ拒否されていないすべてのトラフィックが通過します。

- すべてのアクセス リストは暗黙的な **deny** ステートメントで終了しますが、明示的な **deny** ステートメント（たとえば **deny ip any any**）の使用を推奨します。ほとんどのプラットフォームでは、**show access-list**コマンドを発行して拒否されるパケット数を表示し、アクセス リストが許可していないパケットに関する詳細情報を調査できます。明示的な **deny** ステートメントで拒否されたパケットのみがカウントされます。これは、明示的な **deny** ステートメントによって、より詳細なデータが生成されるためです。
- アクセス リストの作成中、または作成後に、エントリを削除する場合があります。
  - 番号付きアクセスリストからはエントリを削除できません。削除しようとすると、アクセス リスト全体が削除されます。エントリを削除する必要がある場合、アクセス リスト全体を削除してから最初から作り直す必要があります。
  - 名前付きアクセス リストからはエントリを削除できます。**no permit**または **no deny** コマンドを使用すると、適切なエントリが削除されます。
- 個々のステートメントの用途をひと目で確認および理解しやすくするために、**remark** コマンドを使用して、ステートメントの前後に役立つ注記を書き込むことができます。
- 特定のホストまたはネットワークに対するアクセスを拒否し、そのネットワークまたはホストの誰かがアクセスしようとしたかどうかを検出する場合、対応する **deny** ステートメントを指定した **log** キーワードを含めます。それによって、その送信元からの拒否されたパケットがログに記録されます。
- このヒントは、アクセスリストの配置に適用されます。リソースを保存しようとすると、着信アクセス リストでは常にフィルタ条件を適用した後に、ルーティング テーブルの検索を行います。発信アクセスリストではフィルタ条件を適用する前に、ルーティングテーブルの検索を行います。

# アクセス リストを適用する場所

アクセス リストは、デバイスの着信または発信インターフェイスに適用できます。アクセス リストを着信インターフェイスに適用すると、インターフェイスで着信するトラフィックが制御され、アクセスリストを発信インターフェイスに適用すると、インターフェイスから発信されるトラフィックが制御されます。

ソフトウェアは、着信インターフェイスでパケットを受信すると、アクセスリストで設定されているステートメントに対してパケットを検査します。アクセスリストがアドレスを許可している場合は、ソフトウェアはパケットを処理します。着信パケットをフィルタリングするためにアクセスリストを適用すると、フィルタリングされたパケットはデバイスに到達する前に廃棄されるため、デバイスのリソースを節約できます。

発信インターフェイスでは、アクセス リストはインターフェイスから転送（送信）されたパケットをフィルタリングします。発信インターフェイスで Rate-Based Satellite Control Protocol（RBSCP）の TCP アクセス コントロール リスト（ACL）を使用して、発信インターフェイスで TCP 確認応答（ACK）を受けるパケットの種類を制御できます。

**debug** コマンドを使用してアクセス リストを参照し、デバッグ ログの量を制限できます。たとえば、アクセス リストのフィルタリング基準または一致基準に基づいて、デバッグ ログを送信元または宛先のアドレスまたはプロトコルに制限できます。

アクセス リストを使用して、ルーティング アップデート、ダイヤルオンデマンド（DDR）、および Quality of Service（QoS）機能を制御することができます。

# IP 名前付きアクセス コントロール リストの設定方法

## IP 名前付きアクセス リストの作成

IP 名前付きアクセス リストを作成すると、発信元アドレスと宛先アドレス、またはアドレスと他の IP フィールドの組み合わせをフィルタリングすることができます。名前付きアクセス リストにより、分かりやすい名前の付いたアクセス リストを特定できます。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *name*
4. **remark** *remark*
5. **deny** *protocol* [*source source-wildcard*] {**any** | **host** {*address* | *name*} {*destination* [*destination-wildcard*] {**any** | **host** {*address* | *name*} [**log**]
6. **remark** *remark*
7. **permit** *protocol* [*source source-wildcard*] {**any** | **host** {*address* | *name*} {*destination* [*destination-wildcard*] {**any** | **host** {*address* | *name*} [**log**]
8. アクセス リストにステートメントをさらに指定するには、ステップ 4 ～ 7 を繰り返します。
9. **end**
10. **show ip access-lists**

### 手順の詳細

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

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 3 | **ip access-list extended** *name*<br>例：<br>`Device(config)# ip access-list extended acl1` | 名前を使用して拡張 IP アクセス リストを定義し、拡張名前付きアクセス リストのコンフィギュレーション モードを開始します。 |
| ステップ 4 | **remark** *remark*<br>例：<br>`Device(config-ext-nacl)# remark protect server by denying sales access to the acl1 network` | （任意）アクセス リスト ステートメントに説明を追加します。<br>• 注釈は IP アクセス リスト エントリの前または後に指定できます。<br>• この例では、**remark** コマンドによって、ステップ 5 で設定した **deny** コマンドがインターフェイスに対する Sales ネットワーク アクセスを拒否することをネットワーク管理者に示します。 |
| ステップ 5 | **deny** *protocol* [*source source-wildcard*] {**any** \| **host** {*address* \| *name*} {*destination* [*destination-wildcard*] {**any** \| **host** {*address* \| *name*} [**log**]<br>例：<br>`Device(config-ext-nacl)# deny ip 192.0.2.0 0.0.255.255 host 192.0.2.10 log` | （任意）注釈で指定されたすべての条件に一致するパケットをすべて拒否します。 |
| ステップ 6 | **remark** *remark*<br>例：<br>`Device(config-ext-nacl)# remark allow TCP from any source to any destination` | （任意）アクセス リスト ステートメントに説明を追加します。<br>• 注釈は IP アクセス リスト エントリの前または後に指定できます。 |
| ステップ 7 | **permit** *protocol* [*source source-wildcard*] {**any** \| **host** {*address* \| *name*} {*destination* [*destination-wildcard*] {**any** \| **host** {*address* \| *name*} [**log**]<br>例：<br>`Device(config-ext-nacl)# permit tcp any any` | ステートメントで指定されたすべての条件に一致するパケットをすべて許可します。 |
| ステップ 8 | アクセス リストにステートメントをさらに指定するには、ステップ 4 ～ 7 を繰り返します。 | （注） ステートメントによって明示的に許可されていないすべての送信元アドレスは、アクセス リストの末尾にある暗黙的な deny ステートメントで拒否されます。 |
| ステップ 9 | **end**<br>例：<br>`Device(config-ext-nacl)# end` | 拡張名前付きアクセス リストのコンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |
| ステップ 10 | **show ip access-lists**<br>例： | 現在のすべての IP アクセス リストの内容を表示します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device# show ip access-lists` | |

例：

次に、 **show ip access-lists** コマンドの出力例を示します。

```
Device# show ip access-lists acl1

Extended IP access list acl1
   permit tcp any 192.0.2.0 255.255.255.255 eq telnet
   deny tcp any any
   deny udp any 192.0.2.0 255.255.255.255 lt 1024
   deny ip any any log
```

# インターフェイスへのアクセス リストの適用

## 手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip access-group** {*access-list-number* | *access-list-name*} {**in** | **out**}
5. **end**

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **interface** *type number*<br>例： | インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。 |
| ステップ 4 | **ip access-group** {*access-list-number* \| *access-list-name*} {**in** \| **out**}<br>例：<br>`Device(config-if)# ip access-group acl1 in` | 指定したアクセス リストをインバウンド インターフェイスに適用します。<br>• 送信元アドレスをフィルタリングするには、インバウンド インターフェイスにアクセス リストを適用します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 5 | **end**<br>例：<br>`Device(config-if)# end` | インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |

# IP 名前付きアクセス コントロール リストの設定例

## 例：IP 名前付きアクセス コントロール リストの作成

```
Device# configure terminal
Device(config)# ip access-list extended acl1
Device(config-ext-nacl)# remark protect server by denying sales access to the acl1 network
Device(config-ext-nacl)# deny ip 192.0.2.0 0.0.255.255 host 192.0.2.10 log
Device(config-ext-nacl)# remark allow TCP from any source to any destination
Device(config-ext-nacl)# permit tcp any any
```

## 例：インターフェイスへのアクセス リストの適用

```
Device# configure terminal

Device(config-if)# ip access-group acl1 in
```

# IP 名前付きアクセス コントロール リストの追加情報

関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | Cisco IOS Master Command List, All Releases |
| セキュリティ コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |

### シスコのテクニカル サポート

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

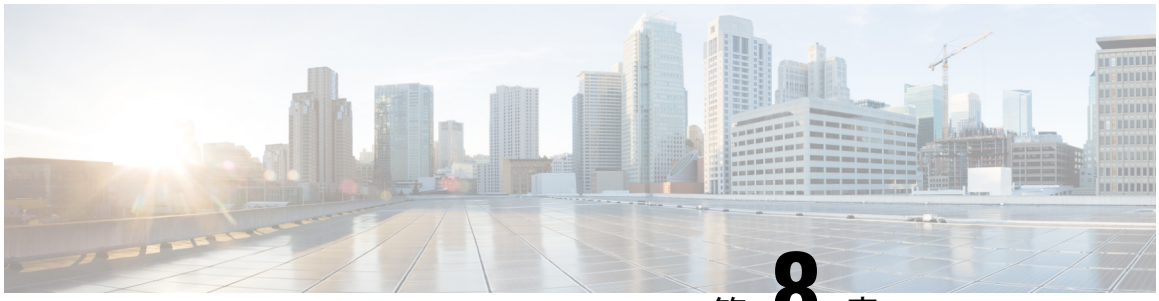
# IP名前付きアクセスコントロールリストに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *8 : IP* 名前付きアクセス コントロール リストに関する機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| IP名前付きアクセスコントロールリスト | | アクセス コントロール リスト（ACL）は、ネットワークを通過するパケットの動きを制御するためにパケット フィルタリングを実行します。パケット フィルタリングは、ネットワークへのトラフィックを限定し、ユーザおよびデバイスのネットワークに対するアクセスを制限し、トラフィックがネットワークから流出するのを防ぐことで、セキュリティを提供します。IP アクセス リストによって、スプーフィングやサービス妨害攻撃の可能性を軽減し、ファイアウォールを介した動的で一時的なユーザ アクセスが可能になります。 |



第 8 章

# 注釈付きの IP アクセス リスト エントリ

注釈付きの IP アクセス リスト エントリ機能により、**deny** または**permit** 条件に関するコメントや注釈を IP アクセス リストに含めることができます。これらの注釈は、ネットワーク管理者がアクセス リストを理解するのを容易にします。各注釈の長さは 100 文字に制限されます。

このモジュールは、注釈付きの IP アクセス リスト エントリ機能に関する情報を提供します。



## 機能情報の確認

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

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

## 注釈付き IP アクセス リスト エントリに関する情報

### IP アクセス リストの利点

アクセス コントロール リスト（ACL）は、ネットワークを通過するパケットのフローを制御するためにパケット フィルタリングを実行します。パケット フィルタリングによってユーザ

およびデバイスのネットワークに対するアクセスを制限し、セキュリティの手段として利用できます。アクセス リストによってトラフィック数を減らすことで、ネットワーク リソースを節約できます。アクセス リストを使用した場合の利点は次のとおりです。

- 着信 rsh および rcp 要求を認証する：アクセス リストは、デバイスへのアクセスを制御するように構成された認証データベース内のローカル ユーザ、リモート ホスト、およびリモート ユーザの識別を簡素化できます。Cisco ソフトウェアは認証データベースを使用して、リモート シェル（rsh）およびリモート コピー（rcp）プロトコルの着信要求を受け取ることができます。
- 不要なトラフィックまたはユーザをブロックする：アクセス リストを使用すると、インターフェイス上の着信パケットまたは発信パケットをフィルタできるため、送信元アドレス、宛先アドレス、またはユーザ認証に基づいてネットワークへのアクセスを制御できます。また、アクセス リストを使用して、デバイス インターフェイスで転送またはブロックするトラフィックの種類を決定することもできます。たとえば、電子メール トラフィックはネットワークでルーティングすることを許可し、すべての Telnet トラフィックはネットワークに入ることをブロックするようにアクセス リストを使用できます。
- vty へのアクセスを制御する：インバウンド vty（Telnet）でのアクセス リストは、デバイスへの回線にアクセスできるユーザを制御できます。アウトバウンド vty でのアクセス リストは、デバイスからの回線が到達可能な宛先を制御できます。
- QoS 機能のトラフィックを特定または分類する：アクセス リストは、Weighted Random Early Detection（WRED）および専用アクセス レート（CAR）の IP プレシデンスを設定することで、輻輳回避を提供します。また、クラスベース均等化キューイング（CBWFQ）、プライオリティ キューイング、カスタム キューイングのために輻輳管理を提供します。
- debug コマンド出力を制限する：アクセス リストは、IP アドレスやプロトコルに基づいて debug 出力を制限できます。
- 帯域幅制御を提供する：低速リンクでのアクセス リストはネットワークでの過剰なトラフィックを防止できます。
- NAT 制御を提供する：アクセス リストによって、ネットワーク アドレス変換（NAT）が変換するアドレスを制御できます。
- DoS 攻撃の可能性を低減する：アクセス リストは、サービス妨害（DoS）攻撃の可能性を低減させます。ホストからのトラフィック、ネットワーク、またはネットワークにアクセスするユーザを制御するように IP 発信元アドレスを指定します。TCP インターセプト機能を設定することで、接続に関する要求でサーバにフラッディングが発生しないようにすることができます。
- ルーティング アップデートの内容を制限する：アクセス リストによって、ネットワーク内で送信、受信、または再配布されるルーティング アップデートを制御できます。
- ダイヤルオンデマンド コールをトリガーする：アクセス リストによって、ダイヤルおよび切断条件を適用できます。

# アクセス リストの注釈

任意のIPアクセス リストのエントリについて、コメントまたは注釈を含めることができます。アクセス リストの注釈は、アクセス リスト エントリの前後にあるオプションの注釈です。エントリの内容がわかるので、エントリの目的を解釈する必要はありません。各注釈の長さは 100 文字に制限されます。

コメントは、**permit** ステートメントまたは **deny** ステートメントの前後どちらにでも配置できます。注釈を追加する場所には一貫性があるようにしてください。注釈が関連する **permit** ステートメントや **deny** ステートメントの前にある場合と後にある場合とが混在すると、ユーザが混乱する可能性があります。

後続の **deny** ステートメントの機能を説明する注釈の例を次に示します。

```
ip access-list extended telnetting
 remark Do not allow host1 subnet to telnet out
 deny tcp host 172.16.2.88 any eq telnet
```

# 注釈付き IP アクセス リスト エントリの設定方法

## 名前付きまたは番号付きアクセス リストへの注釈の書き込み

名前付きまたは番号付きアクセス リスト設定を使用できます。作業する設定用にアクセス リストを作成したら、アクセスリストをインターフェイスまたは端末回線に適用する必要があります。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list** {**standard** | **extended**} {*name* | *number*}
4. **remark** 注記
5. **deny** *protocol* **host** *host-address* **any eq** *port*
6. **end**

### 手順の詳細

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

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 3 | **ip access-list** {**standard** \| **extended**} {*name* \| *number*}<br>例：<br>`Device(config)# ip access-list extended telnetting` | 名前または番号でアクセス リストを特定し、拡張名前付きアクセス リスト コンフィギュレーション モードを開始します。 |
| ステップ 4 | **remark** 注記<br>例：<br>`Device(config-ext-nacl)# remark Do not allow host1 subnet to telnet out` | 名前付き IP アクセス リストのエントリに注釈を追加します。<br>• 注釈は、**permit** または **deny** ステートメントの目的を示します。 |
| ステップ 5 | **deny** *protocol* **host** *host-address* **any eq** *port*<br>例：<br>`Device(config-ext-nacl)# deny tcp host 172.16.2.88 any eq telnet` | パケットを拒否する名前付き IP アクセス リストの条件を設定します。 |
| ステップ 6 | **end**<br>例：<br>`Device(config-ext-nacl)# end` | 拡張名前付きアクセス リスト コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。 |

# 注釈付き IP アクセス リスト エントリの設定例

## 例：IP アクセス リストの備考の書き込み

```
Device# configure terminal
Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow host1 subnet to telnet out
Device(config-ext-nacl)# deny tcp host 172.16.2.88 any eq telnet
Device(config-ext-nacl)# end
```

# 注釈付き IP アクセス リスト エントリの追加情報

**関連資料**

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | Cisco IOS Master Command List, All Releases |

| 関連項目 | マニュアル タイトル |
|---|---|
| セキュリティ コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |

### シスコのテクニカル サポート

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

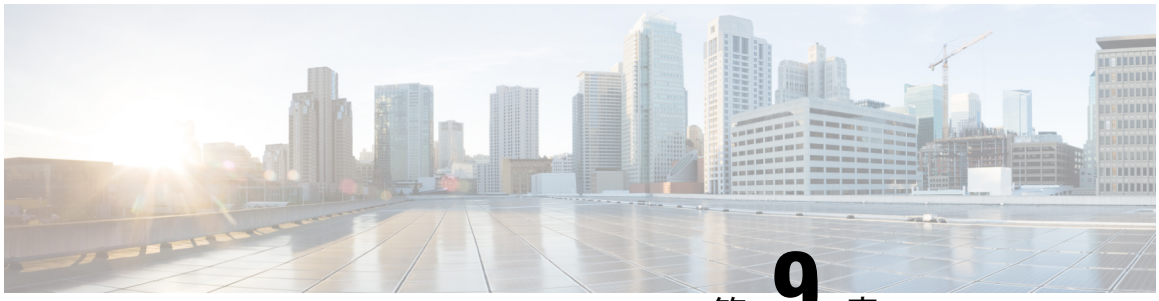
# 注釈付き IP アクセス リスト エントリに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *9*: 注釈付き *IP* アクセス リスト エントリに関する機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| 注釈付きの IP アクセス リスト エントリ | | 注釈付きの IP アクセス リスト エントリ機能により、[deny] または [permit] 条件に関するコメントや備考をどの IP アクセスリストにも含めることができます。これらの注釈は、ネットワーク管理者がアクセス リストを理解するのを容易にします。各注釈の長さは 100 文字に制限されます。<br>次のコマンドが導入または変更されました。<br>**remark** |



第 9 章

# 標準 IP アクセス リストのロギング

標準 IP アクセス リストのロギング機能は、標準 IP アクセス リストによって許可または拒否されるパケットに関するメッセージをロギングする機能を提供します。アクセス リストに一致するパケットによって、デバイス コンソールにあるパケットに関する情報メッセージがロギングされます。

このモジュールは、標準 IP アクセス リスト ロギングに関する情報を提供します。



## 機能情報の確認

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

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

## 標準 IP アクセス リストのロギングに関する制限事項

IP アクセス リスト ロギングは、ルーティング インターフェイスまたはルータ アクセス コントロール リスト（ACL）でのみサポートされます。

# 標準 IP アクセス リストのロギングに関する情報

## 標準 IP アクセス リストのロギング

標準 IP アクセス リストのロギング機能は、標準 IP アクセス リストによって許可または拒否されるパケットに関するメッセージをロギングする機能を提供します。アクセス リストに一致するパケットによって、デバイス コンソールに送信されるパケットに関する情報ロギング メッセージが生成されます。デバイス コンソールに記録されるメッセージのログ レベルは、**logging console** コマンドによって制御されます。

アクセス リストが最初に検査したパケットがアクセスリストをトリガーし、デバイス コンソールにメッセージをロギングします。後続のパケットは、5 分間隔で収集された後、表示またはロギングされます。ログ メッセージには、アクセス リスト番号、パケットの送信元 IP アドレス、その送信元からの、直前の 5 分間隔に許可または拒否されたパケットの数、およびパケットが許可されたか拒否されたかに関する情報が含まれます。特定のアクセス リストによって許可または拒否された複数のパケットについて、各パケットの送信元アドレスなどをモニタすることができます。

# 標準 IP アクセス リストのロギングの設定方法

## 番号を使用した標準 IP アクセス リストの作成

### 手順の概要

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* {**deny** | **permit**} **host** *address* [**log**]
4. **access-list** *access-list-number* {**deny** | **permit**} **any** [**log**]
5. **interface** *type number*
6. **ip access-group** *access-list-number* {**in** | **out**}
7. **end**

### 手順の詳細

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

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device# configure terminal` | |
| ステップ 3 | **access-list** *access-list-number* {**deny** \| **permit**} **host** *address* [**log**]<br>例：<br>`Device(config)# access-list 1 permit host 10.1.1.1 log` | 送信元アドレスとワイルドカードを使用して、標準の名前付き IP アクセス リストを定義し、デバイス コンソールでアクセス リスト エントリと一致したパケットに関する情報メッセージのロギングを設定します。 |
| ステップ 4 | **access-list** *access-list-number* {**deny** \| **permit**} **any** [**log**]<br>例：<br>`Device(config)# access-list 1 permit any log` | 送信元の省略形および送信元マスク 0.0.0.0 255.255.255.255 を使用して、標準の 名前付き IP アクセス リストを定義します。 |
| ステップ 5 | **interface** *type number*<br>例： | インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。 |
| ステップ 6 | **ip access-group** *access-list-number* {**in** \| **out**}<br>例：<br>`Device(config-if)# ip access-group 1 in` | 指定した番号付きアクセス リストを着信または発信インターフェイスに適用します。<br>• 送信元アドレスに基づいてフィルタする場合、一般的に、着信インターフェイスにアクセス リストを適用します。 |
| ステップ 7 | **end**<br>例：<br>`Device(config-if)# end` | インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。 |

# 名前を使用した標準 IP アクセス リストの作成

手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list standard** *name*
4. {**deny** | **permit**} {**host** *address* | **any**} **log**
5. **exit**
6. **interface** *type number*
7. **ip access-group** *access-list-name* {**in** | **out**}
8. **end**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip access-list standard** *name*<br>例：<br>`Device(config)# ip access-list standard acl1` | 標準の IP アクセス リストを定義して、標準の名前付きアクセス リスト コンフィギュレーション モードを開始します。 |
| ステップ 4 | {**deny** \| **permit**} {**host** *address* \| **any**} **log**<br>例：<br>`Device(config-std-nacl)# permit host 10.1.1.1 log` | パケットがネットワークに入らないように拒否したり、パケットがネットワークに入ることを許可したりする名前付き IP アクセス リストで条件を設定し、デバイス コンソールでアクセス リスト エントリと一致するパケットに関する情報メッセージのロギングを設定します。 |
| ステップ 5 | **exit**<br>例：<br>`Device(config-std-nacl)# exit` | 標準の名前付きアクセス リスト コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードを開始します。 |
| ステップ 6 | **interface** *type number*<br>例： | インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。 |
| ステップ 7 | **ip access-group** *access-list-name* {**in** \| **out**}<br>例：<br>`Device(config-if)# ip access-group acl1 in` | 指定したアクセス リストを着信または発信インターフェイスに適用します。<br>• 送信元アドレスに基づいてフィルタする場合、一般的に、着信インターフェイスにアクセス リストを適用します。 |
| ステップ 8 | **end**<br>例：<br>`Device(config-if)# end` | インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。 |

# 標準 IP アクセス リストのロギングの設定例

## 例：数字を使用した標準 IP アクセス リストの作成

```
Device# configure terminal
Device(config)# access-list 1 permit host 10.1.1.1 log
Device(config)# access-list 1 permit any log

Device(config-if)# ip access-group 1 in
```

## 例：名前を使用した標準 IP アクセス リストの作成

```
Device# configure terminal
Device(config)# ip access-list standard acl1
Device(config-std-nacl)# permit host 10.1.1.1 log
Device(config-std-nacl)# exit

Device(config-if)# ip access-group acl1 in
```

## 例：デバッグ出力の制限

次の設定例では、アクセス リストを使用して、**debug** コマンドの出力を制限します。**debug** の出力を制限すると、データ量が絞られ、目的のデータを探しやすくなるため、時間とリソースを節約できます。

```
Device(config)# ip access-list acl1
Device(config-std-nacl)# remark Displays only advertisements for LDP peer in acl1
Device(config-std-nacl)# permit host 10.0.0.44

Device# debug mpls ldp advertisements peer-acl acl1

tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.17.0.33
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.16.0.31
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.22.0.33
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.0.1
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.0.3
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.1.33
```

# 標準 IP アクセス リストのロギングに関する追加情報

**関連資料**

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Commands List, All Releases』 |

| 関連項目 | マニュアル タイトル |
|---|---|
| セキュリティ コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |

### シスコのテクニカル サポート

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

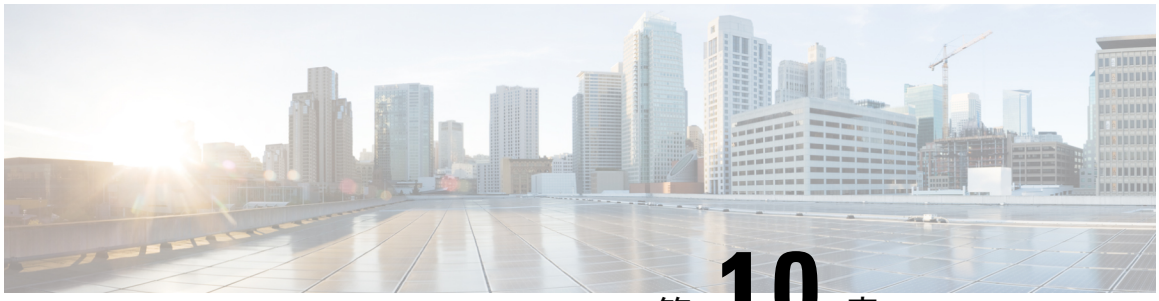
# 標準 IP アクセス リストのロギングに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

表 *10* : 標準 *IP* アクセス リストのロギングに関する機能情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| 標準 IP アクセス リストのロギング | | 標準 IP アクセス リストのロギング機能は、標準 IP アクセス リストによって許可または拒否されるパケットに関するメッセージをロギングする機能を提供します。アクセス リストに一致するパケットによって、デバイス コンソールにあるパケットに関する情報メッセージがロギングされます。 |



第 10 章

# IP アクセス リスト エントリ シーケンス番号

IP アクセス リスト エントリ シーケンス番号機能により、**permit** または **deny** ステートメントにシーケンス番号を適用したり、名前付き IP アクセス リストでそのようなステートメントを順序変更、追加、削除することができます。IP アクセス リスト エントリ シーケンス番号機能を使用すると、IP アクセス リストを非常に簡単に変更することができます。この機能以前は、アクセス リストの末尾にしかアクセス リスト エントリを追加できませんでした。そのため、名前付き IP アクセス リストの末尾以外のどこかにステートメントを追加する必要がある場合、アクセス リスト全体の再設定が必要でした。



## 機能情報の確認

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

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

# IPアクセスリストのエントリシーケンス番号に関する制約事項

- この機能は、ダイナミック アクセス リスト、再帰アクセス リスト、またはファイアウォール アクセス リストをサポートしていません。
- また、名前付きアクセス リストよりも古くから存在する、旧式のスタイルで番号付けされたアクセス リストもサポートしていません。アクセス リストは番号で指定できるため、標準または拡張名前付きアクセス リスト（NACL）コンフィギュレーション モードでは番号を入力することができます。

# IPアクセスリストのエントリシーケンス番号に関する情報

## IP アクセス リストの目的

アクセス リストは、パケット フィルタリングを実行して、ネットワークを介して移動するパケットとその場所を制御します。この処理は、ネットワーク トラフィックを制限し、ユーザやデバイスによるネットワークへのアクセスを制限するのに役立ちます。アクセス リストの用途は多様なので、多くのコマンドの構文でアクセス リストが参照されます。アクセス リストを使用して、次のようなことを実行できます。

- インターフェイスでの着信パケットのフィルタリング
- インターフェイスでの発信パケットのフィルタリング
- ルーティング アップデートの内容の制限
- アドレスまたはプロトコルに基づくデバッグ出力の制限
- 仮想端末回線アクセスの制御
- 輻輳回避、輻輳管理、プライオリティおよびカスタム キューイングなどの高度な機能に使用されるトラフィックの特定または分類
- ダイヤルオンデマンド ルーティング（DDR）呼び出しのトリガー

## IP アクセス リストの機能

アクセス リストは、permit ステートメントと deny ステートメントで構成される順次リストです。これらのステートメントは、IP アドレス、場合によっては上位層 IP プロトコルに適用さ

れます。アクセス リストには、参照に使用される名前があります。多くのソフトウェア コマンドは、構文の一部としてアクセス リストを受け取ります。

アクセス リストを設定して名前を付けることは可能ですが、アクセス リストを受け取るコマンドによってアクセスリストが参照されるまで、有効にはなりません。複数のコマンドから同じアクセス リストを参照できます。アクセス リストで、デバイスに到達するトラフィック、またはデバイス経由で送信されるトラフィックは制御できますが、デバイスが送信元のトラフィックは制御できません。

# IP アクセス リストのプロセスとルール

- アクセス リストの条件に対してフィルタリングされる各パケットの送信元アドレスや宛先アドレス、またはプロトコルがテストされます。一度に 1 つの条件（**permit** ステートメントまたは **deny** ステートメント）がテストされます。
- パケットがアクセス リストのステートメントに一致しないと、そのパケットはリスト内の次のステートメントに対してテストされます。
- パケットとアクセス リスト ステートメントが一致すると、リスト内の残りのステートメントはスキップされ、パケットは一致したステートメントに指定されたとおりに許可または拒否されます。パケットが許可されるか拒否されるかは、パケットが一致する最初のエントリによって決まります。つまり、一致すると、それ以降のエントリは考慮されません。
- アクセス リストでアドレスまたはプロトコルが拒否されると、パケットは廃棄され、インターネット制御メッセージプロトコル（ICMP）ホスト到達不能メッセージが返されます。
- 一致する条件がない場合は、パケットはドロップされます。これは、各アクセス リストは暗黙の **deny** ステートメントで終了するためです。言い換えると、パケットが各ステートメントに対してテストされたときまでに許可されないと、このパケットは拒否されます。
- 最初に一致が見つかった後は条件のテストが終了するため、条件の順序は重要です。同じ **permit** ステートメントまたは **deny** ステートメントでも、順序が異なる場合、ある状況では通過し、別の状況では拒否されるパケットが生じる可能性があります。
- コマンドでアクセス リストを名前によって参照したときに、そのアクセス リストが存在しない場合は、すべてのパケットが通過します。
- 1 つのインターフェイス、1 つのプロトコル、1 つの方向につき、許可されるアクセス リストは 1 つだけです。
- インバウンドアクセス リストは、デバイスに到達するパケットを処理します。着信パケットの処理後に、アウトバウンドインターフェイスへのルーティングが行われます。インバウンドアクセス リストが効率的なのは、フィルタリングテストで拒否されたことでパケットが廃棄される場合、ルーティング検索のオーバーヘッドが抑えられるためです。パケットがテストで許可されると、そのパケットに対してルーティングの処理が実施されます。インバウンド リストの場合、**permit** とは、インバウンド インターフェイスでパケットの受信後に処理が続行されることを示します。**deny** とは、パケットが廃棄されることを示します。

- 発信アクセス リストの場合、パケットの処理後にデバイスから送信されます。着信パケットはアウトバウンド インターフェイスにルーティングされてから、アウトバウンド アクセス リストで処理されます。アウトバウンド リストの場合、**permit** とは、出力バッファに対して送信されることを示し、**deny** とは、パケットが廃棄されることを示します。

# IP アクセス リストを作成する際に役立つヒント

意図しない結果を回避し、より効率的で有効なアクセス リストを作成するために役立つヒントを紹介します。

- アクセス リストを作成してから、インターフェイス（または別の対象）に適用します。その理由は、存在しないアクセス リストをインターフェイスに適用してから、アクセス リストを設定すると、最初のステートメントが有効になり、それに続く暗黙的な **deny** ステートメントによってアクセスに緊急の問題が発生するおそれがあるためです。
- アクセス リストを設定してから適用するもう 1 つの理由は、空のアクセス リストが適用されたインターフェイスはすべてのトラフィックを許可するためです。
- すべてのアクセス リストには、少なくとも 1 つの **permit** ステートメントが必要です。permit がないと、すべてのパケットは拒否され、トラフィックはまったく通過しません。
- 最初に（**permit** または **deny** ステートメントに対する）一致が見つかった後は条件のテストが終了するため、パケットが一致する可能性の高いステートメントをアクセス リストの先頭に配置すると処理にかかる時間とリソースが削減されます。最も頻繁に発生する条件を発生頻度の低い条件より前に配置します。
- ネットワークまたはサブネットのより具体的な参照が、より全般的な参照よりも前に出現するように、アクセス リストを構成します。
- まだ拒否されていないその他のパケットすべてを許可する場合、ステートメント **permit any any** を使用します。ステートメント **permit any any** を使用すると、実質的に、アクセス リストの末尾にある暗黙的な deny ステートメントでその他すべてのパケットが拒否されることを防ぎます。最初のアクセス リスト エントリは **permit any any** にしないでください。すべてのトラフィックが通過し、以降のテストに到達するパケットがなくなります。**permit any any** を指定すると、まだ拒否されていないすべてのトラフィックが通過します。
- すべてのアクセス リストは暗黙的な **deny** ステートメントで終了しますが、明示的な **deny** ステートメント（たとえば **deny ip any any**）の使用を推奨します。ほとんどのプラットフォームでは、**show access-list** コマンドを発行して拒否されるパケット数を表示し、アクセス リストが許可していないパケットに関する詳細情報を調査できます。明示的な **deny** ステートメントで拒否されたパケットのみがカウントされます。これは、明示的な **deny** ステートメントによって、より詳細なデータが生成されるためです。
- アクセス リストの作成中、または作成後に、エントリを削除する場合があります。

- 番号付きアクセスリストからはエントリを削除できません。削除しようとすると、アクセス リスト全体が削除されます。エントリを削除する必要がある場合、アクセスリスト全体を削除してから最初から作り直す必要があります。
- 名前付きアクセス リストからはエントリを削除できます。**no permit**または **no deny** コマンドを使用すると、適切なエントリが削除されます。

- 個々のステートメントの用途をひと目で確認および理解しやすくするために、**remark** コマンドを使用して、ステートメントの前後に役立つ注記を書き込むことができます。
- 特定のホストまたはネットワークに対するアクセスを拒否し、そのネットワークまたはホストの誰かがアクセスしようとしたかどうかを検出する場合、対応する **deny** ステートメントを指定した **log** キーワードを含めます。それによって、その送信元からの拒否されたパケットがログに記録されます。
- このヒントは、アクセスリストの配置に適用されます。リソースを保存しようとすると、着信アクセス リストでは常にフィルタ条件を適用した後に、ルーティング テーブルの検索を行います。発信アクセスリストではフィルタ条件を適用する前に、ルーティングテーブルの検索を行います。

# 送信元アドレスと宛先アドレス

IP パケットの送信元アドレスと宛先アドレスのフィールドは、アクセス リストの基礎となる典型的な2つのフィールドです。送信元アドレスを指定して、特定のネットワーキング デバイスまたはホストから送信されるパケットを制御します。宛先アドレスを指定して、特定のネットワーキング デバイスまたはホストに送信されるパケットを制御します。

# ワイルドカード マスクおよび暗黙のワイルドカード マスク

アドレス フィルタリングでは、アクセス リスト エントリ内のアドレス ビットとアクセス リストに送信されるパケットを比較する際、対応する IP アドレス ビットを確認するか無視するかを決定するために、ワイルドカード マスクが使用されます。管理者は、ワイルドカード マスクを慎重に設定することにより、許可または拒否のテストに 1 つまたは複数の IP アドレスを選択できます。

IP アドレス ビット用のワイルドカード マスクでは、数値 1 と数値 0 を使用して、対応する IP アドレス ビットをどのように扱うかを指定します。1 と 0 は、サブネット（ネットワーク）マスクで意味する内容が対照的なため、ワイルドカード マスクは逆マスクとも呼ばれます。

- ワイルドカード マスク ビット 0 は、対応するビット値を確認することを示します。
- ワイルドカード マスク ビット 1 は、対応するビット値を無視することを示します。

アクセス リスト ステートメントの送信元アドレスまたは宛先アドレスでワイルドカード マスクを指定しない場合、0.0.0.0 というデフォルトのワイルドカード マスクが想定されます。

サブネットマスクでは、ネットワークとサブネットを示す隣接ビットをマスクにする必要がありますが、それとは異なり、ワイルドカード マスクではマスクに非隣接ビットを使用できます。

# トランスポート層の情報

トランスポート層の情報（パケットが TCP、UDP、Internet Control Message Protocol（ICMP）または Internet Group Management Protocol（IGMP）パケットであるか、などの情報）に基づいてパケットをフィルタできます。

# 利点：IP アクセス リスト エントリ シーケンス番号

IP アクセス リスト エントリにシーケンス番号を適用する機能によって、アクセス リストの変更が簡易になります。IP アクセス リスト エントリ シーケンス番号機能の前には、アクセス リスト内のエントリの位置を指定する方法はありませんでした。既存のリストの途中にエントリ（ステートメント）を挿入するには、目的の位置の後ろにあるすべてのエントリを削除する必要がありました。次に、新しいエントリを追加したら、先に削除したすべてのエントリを再入力する必要がありました。これは手間がかかり、エラーが起こりやすい方法です。

IP アクセス リスト エントリ シーケンス番号機能を使用すると、アクセス リスト エントリにシーケンス番号を追加し、リスト内のエントリを並べ替えることができます。新しいエントリを追加する場合、アクセスリストの目的の位置にエントリが挿入されるようにシーケンス番号を選択できます。必要に応じて、アクセスリストの現在のエントリを並べ替えて、新しいエントリを挿入できる場所を作成できます。

# シーケンス番号の動作

- 以前のリリースとの下位互換性を保つため、シーケンス番号のないエントリが適用された場合には、最初のエントリにはシーケンス番号 10 が割り当てられます。連続してエントリを追加すると、シーケンス番号は10ずつ増分されます。最大シーケンス番号は2147483647です。生成したシーケンス番号がこの最大値を超えると、次のメッセージが表示されます。

```
Exceeded maximum sequence number.
```

- シーケンス番号のないエントリを1つ入力すると、アクセス リストの最後のシーケンス番号に 10 を加えたシーケンス番号が割り当てられ、リストの末尾に配置されます。
- （シーケンス番号以外が）既存のエントリに一致するエントリを入力すると、何も変更されません。
- 既存のシーケンス番号を入力すると、次のエラー メッセージが表示されます。

```
Duplicate sequence number.
```

- グローバル コンフィギュレーション モードで新しいアクセス リストを入力すると、そのアクセス リストのシーケンス番号が自動的に生成されます。
- ルート プロセッサ（RP）のエントリとラインカード（LC）のエントリのシーケンス番号を常に同期できるように、分散機能がサポートされています。
- シーケンス番号が不揮発性生成（NVGEN）されることはありません。つまり、シーケンス番号自体は保存されません。システムのリロード時には、設定されたシーケンス番号はデフォルトのシーケンス開始番号とその番号からの増分に戻されます。この機能は、シーケンス番号をサポートしないソフトウェア リリースとの下位互換性を保つために提供されています。
- IP アクセス リスト エントリ シーケンス番号機能では、名前付き標準アクセス リストと拡張 IP アクセス リストが使用されます。アクセス リストの名前を番号として指定できるため、番号も使用できます。

# IP アクセス リストでのシーケンス番号の使用法

## アクセス リスト エントリの順序付けとアクセス リストの変更

ここでは、名前付き IP アクセス リストのエントリにシーケンス番号を割り当てる方法と、アクセスリストに対するエントリの追加または削除を行う方法を説明します。この作業を実行する場合は、次の点に注意してください。

- アクセス リスト エントリの並べ替えは任意です。この作業での並べ替えのステップは、機能の目的の 1 つであり、またその機能の説明が必要と思われることから、必要に応じて説明します。
- 次の手順で、**permit** コマンドはステップ 5 に、**deny** コマンドはステップ 6 に記載されています。ただし、その順番を入れ替えることもできます。設定のニーズに合わせた順番を使用します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list resequence** *access-list-name starting-sequence-number increment*
4. **ip access-list** {**standard**| **extended**} *access-list-name*
5. 次のいずれかを実行します。
   - *sequence-number* **permit** *source source-wildcard*
   - *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]
6. 次のいずれかを実行します。
   - *sequence-number* **deny** *source source-wildcard*

- *sequence-number* **deny** *protocol source source-wildcard destination destination-wildcard* [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]

7. 次のいずれかを実行します。
   - *sequence-number* **permit** *source source-wildcard*
   - *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]
8. 次のいずれかを実行します。
   - *sequence-number* **deny** *source source-wildcard*
   - *sequence-number* **deny** *protocol source source-wildcard destination destination-wildcard* [ **precedence** *precedence*][ **tos** *tos*] [**log**] [ **time-range** *time-range-name*] [**fragments**]
9. 必要に応じてシーケンス番号ステートメントを追加するには、ステップ 5 とステップ 6 を繰り返します。
10. **end**
11. **show ip access-lists** *access-list-name*

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権EXECモードをイネーブルにします。パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip access-list resequence** *access-list-name starting-sequence-number increment*<br>例：<br>`Device(config)# ip access-list resequence kmd1 100 15` | 開始シーケンス番号と、シーケンス番号の増分を使用して、指定した IP アクセス リストを並べ替えます。 |
| ステップ 4 | **ip access-list** {**standard**\| **extended**} *access-list-name*<br>例：<br>`Device(config)# ip access-list standard kmd1` | 名前で IP アクセス リストを指定し、名前付きアクセス リストのコンフィギュレーション モードを開始します。<br>• **standard** を指定する場合は、その後に、標準アクセス リスト構文を使用して **permit** ステートメントまたは **deny** ステートメントを指定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • **extended** を指定する場合は、その後に、拡張アクセス リスト構文を使用して **permit** ステートメントまたは **deny** ステートメントを指定します。 |
| **ステップ 5** | 次のいずれかを実行します。<br>• *sequence-number* **permit** *source source-wildcard*<br>• *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][ **tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]<br>例：<br>`Device(config-std-nacl)# 105 permit 10.5.5.5 0.0.0 255` | 名前付き IP アクセス リスト モードで permit ステートメントを指定します。<br>• このアクセス リストでは **permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。<br>• プロンプトに示されるとおり、このアクセス リストは標準アクセス リストでした。ステップ 4 で **extended** を指定した場合は、このステップのプロンプトは Device(config-ext-nacl) となり、拡張 **permit** コマンド構文を使用します。 |
| **ステップ 6** | 次のいずれかを実行します。<br>• *sequence-number* **deny** *source source-wildcard*<br>• *sequence-number* **deny** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][ **tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]<br>例：<br>`Device(config-std-nacl)# 105 deny 10.6.6.7 0.0.0 255` | （任意）名前付き IP アクセス リスト モードで deny ステートメントを指定します。<br>• このアクセス リストでは **permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。<br>• プロンプトに示されるとおり、このアクセス リストは標準アクセス リストでした。ステップ 4 で **extended** を指定した場合は、このステップのプロンプトは Device(config-ext-nacl) となり、拡張 **deny** コマンド構文を使用します。 |
| **ステップ 7** | 次のいずれかを実行します。<br>• *sequence-number* **permit** *source source-wildcard*<br>• *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][ **tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]<br>例：<br>`Device(config-ext-nacl)# 150 permit tcp any any log` | 名前付き IP アクセス リスト モードで permit ステートメントを指定します。<br>• このアクセス リストでは **permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。<br>• 上位層プロトコル（ICMP、IGMP、TCP、および UDP）を許可するその他のコマンド構文については、permit（IP）コマンドを参照してください。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | | • エントリを削除するには、**no** *sequence-number* コマンドを使用します。 |
| **ステップ 8** | 次のいずれかを実行します。<br>• *sequence-number* **deny** *source source-wildcard*<br>• *sequence-number* **deny** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][ **tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]<br>**例**：<br>`Device(config-ext-nacl)# 150 deny tcp any any log` | （任意）名前付き IP アクセス リスト モードで deny ステートメントを指定します。<br>• このアクセス リストでは **permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。<br>• 上位層プロトコル（ICMP、IGMP、TCP、および UDP）を許可するその他のコマンド構文については、deny（IP）コマンドを参照してください。<br>• エントリを削除するには、**no** *sequence-number* コマンドを使用します。 |
| **ステップ 9** | 必要に応じてシーケンス番号ステートメントを追加するには、ステップ 5 とステップ 6 を繰り返します。 | アクセス リストは変更できます。 |
| **ステップ 10** | **end**<br>**例**：<br>`Device(config-std-nacl)# end` | （任意）コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |
| **ステップ 11** | **show ip access-lists** *access-list-name*<br>**例**：<br>`Device# show ip access-lists kmd1` | （任意）IP アクセス リストの内容を表示します。 |

**例**

アクセス リストに新しいエントリが含まれていることを確認するには、**show ip access-lists** コマンドの出力を確認します。

```
Device# show ip access-lists kmd1

Standard IP access list kmd1
100 permit 10.4.4.0, wildcard bits 0.0.0.255
105 permit 10.5.5.0, wildcard bits 0.0.0.255
115 permit 10.0.0.0, wildcard bits 0.0.0.255
130 permit 10.5.5.0, wildcard bits 0.0.0.255
145 permit 10.0.0.0, wildcard bits 0.0.0.255
```

# IP アクセス リスト エントリ シーケンス番号の設定例

## 例：アクセス リストのエントリの並べ替え

次に、アクセス リストを並べ替える例を示します。開始値は 1、増分値は 2 です。後続のエントリは指定の増分値に基づいて並べられています。範囲は 1 ～ 2147483647 です。

シーケンス番号のないエントリが入力されると、デフォルトで、アクセス リストの最後のエントリのシーケンス番号に 10 を加えたシーケンス番号が割り当てられます。

```
Device# show access-list 150

Extended IP access list 150
    10 permit ip host 10.3.3.3 host 172.16.5.34
    20 permit icmp any any
    30 permit tcp any host 10.3.3.3
    40 permit ip host 10.4.4.4 any
    50 Dynamic test permit ip any any
    60 permit ip host 172.16.2.2 host 10.3.3.12
    70 permit ip host 10.3.3.3 any log
    80 permit tcp host 10.3.3.3 host 10.1.2.2
    90 permit ip host 10.3.3.3 any
    100 permit ip any any

Device(config)# ip access-list extended 150
Device(config)# ip access-list resequence 150 1 2
Device(config)# exit

Device# show access-list 150

Extended IP access list 150
    1 permit ip host 10.3.3.3 host 172.16.5.34
    3 permit icmp any any
    10 permit tcp any any eq 22 log
    5 permit tcp any host 10.3.3.3
    7 permit ip host 10.4.4.4 any
    9 Dynamic test permit ip any any
    11 permit ip host 172.16.2.2 host 10.3.3.12
    13 permit ip host 10.3.3.3 any log
    15 permit tcp host 10.3.3.3 host 10.1.2.2
    17 permit ip host 10.3.3.3 any
    19 permit ip any any
```

## 例：シーケンス番号を持つエントリの追加

次に、指定のアクセス リストに新しいエントリを追加する例を示します。

```
Device# show ip access-list

Standard IP access list tryon
2 permit 10.4.4.2, wildcard bits 0.0.255.255
5 permit 10.0.0.44, wildcard bits 0.0.0.255
10 permit 10.0.0.1, wildcard bits 0.0.0.255
20 permit 10.0.0.2, wildcard bits 0.0.0.255
```

```
Device(config)# ip access-list standard tryon
Device(config-std-nacl)# 15 permit 10.5.5.5 0.0.0.255
Device(config-std-nacl)# exit
Device(config)# exit
Device# show ip access-list

Standard IP access list tryon
2 permit 10.4.0.0, wildcard bits 0.0.255.255
5 permit 10.0.0.0, wildcard bits 0.0.0.255
10 permit 10.0.0.0, wildcard bits 0.0.0.255
15 permit 10.5.5.0, wildcard bits 0.0.0.255
20 permit 10.0.0.0, wildcard bits 0.0.0.255
```

## 例：シーケンス番号のないエントリ

次に、シーケンス番号が指定されていないエントリをアクセス リストの末尾に追加する方法を示します。シーケンス番号のないエントリを追加すると、自動的にシーケンス番号が割り当てられ、アクセス リストの末尾に配置されます。デフォルトの増分値は 10 であるため、エントリには、既存のアクセス リストの最後のエントリのシーケンス番号に 10 を加えたシーケンス番号が割り当てられます。

```
Device(config)# ip access-list standard 1
Device(config-std-nacl)# permit 10.1.1.1 0.0.0.255
Device(config-std-nacl)# permit 10.2.2.2 0.0.0.255
Device(config-std-nacl)# permit 10.3.3.3 0.0.0.255
Device(config-std-nacl)## exit
Device# show access-list

Standard IP access list 1
10 permit 0.0.0.0, wildcard bits 0.0.0.255
20 permit 0.0.0.0, wildcard bits 0.0.0.255
30 permit 0.0.0.0, wildcard bits 0.0.0.255

Device(config)# ip access-list standard 1
Device(config-std-nacl)# permit 10.4.4.4 0.0.0.255
Device(config-std-nacl)# end
Device(config-std-nacl)## exit
Device# show access-list

Standard IP access list 1
10 permit 0.0.0.0, wildcard bits 0.0.0.255
20 permit 0.0.0.0, wildcard bits 0.0.0.255
30 permit 0.0.0.0, wildcard bits 0.0.0.255
40 permit 0.0.0.0, wildcard bits 0.0.0.255
```

# その他の参考資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Command List, All Releases』 |

| 関連項目 | マニュアル タイトル |
|---|---|
| IP アクセス リスト コマンド | 『*Cisco IOS Security Command Reference*』 |
| IP アクセス リストの設定 | 『Creating an IP Access List and Applying It to an Interface』 |

### テクニカル サポート

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

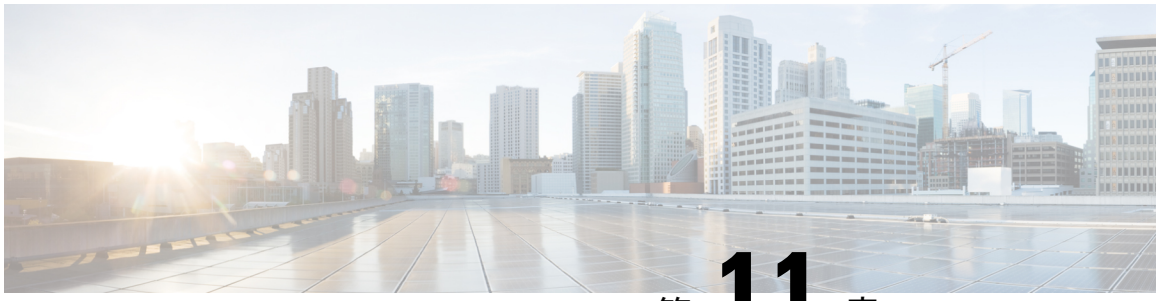
# IP アクセス リスト エントリ シーケンス番号に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

表 *11 : IP* アクセス リスト エントリ シーケンス番号に関する機能情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| IP アクセス リスト エントリ シーケンス番号 | | **permit** または **deny** ステートメントにシーケンス番号を適用し、名前付き IP アクセス リストで、該当するステートメントの再整理、追加、または削除を行うことができます。この機能により、IPアクセスリストを簡単に変更できるようになります。この機能が実装される前は、アクセスリストの最後にエントリを追加することしかできませんでした。そのため、末尾以外の任意の場所にステートメントを追加する必要があるときは、アクセスリスト全体を再設定する必要がありました。<br><br>では、Cisco Catalyst 3850 シリーズ スイッチのサポートが追加されました。<br><br>次のコマンドが導入または変更されました。**deny (IP)**、**ip access-list resequence deny (IP), permit (IP)** |



第 11 章

# ロック アンド キー セキュリティの設定（ダイナミックアクセス リスト）

**機能の履歴**

| リリース | 変更内容 |
|---|---|
| Cisco IOS | Cisco IOS ソフトウェアの機能サポートに関する情報については、Cisco Feature Navigator を使用してください。 |

この章では、ルータでロックアンドキー セキュリティを設定する方法について説明します。ロック アンド キーは、IP プロトコルで使用可能なトラフィック フィルタリング セキュリティ機能です。

ロック アンド キー コマンドの詳細な説明については、『*Cisco IOS* セキュリティ コマンド リファレンス』を参照してください。この章で使用されたその他のコマンドの詳細については、コマンド リファレンス マスタ インデックスを使用するか、オンラインで検索してください。

機能に関連付けられたハードウェア プラットフォームまたはソフトウェア イメージの情報を識別するには、Cisco.com の Feature Navigator を使用して機能についての情報を検索するか、特定のリリースのソフトウェア リリースノートを参照してください。



## ロック アンド キーの設定の必須条件

ロック アンド キーは、IP 拡張アクセス リストを使用します。ロック アンド キーを設定しようとする前に、アクセス リストを使用してトラフィックをフィルタする方法について確実に理

解する必要があります。アクセス リストについては、「アクセス コントロール リスト：概要および指針」を参照してください。

ロックアンドキーは、Ciscoの認証、許可、アカウンティング（AAA）の枠組みで実装されているように、ユーザ認証と認可を使用します。ロック アンド キーを設定する前に、AAA ユーザ認証、許可、アカウンティングの設定方法について理解する必要があります。ユーザ認証および認可は、本書の「認証、認可、アカウンティング（AAA）」のセクションで説明します。

ロック アンド キーは、理解する必要のある **autocommand** コマンドを使用します。このコマンドは、『Cisco IOS Terminal *Services* コマンド リファレンス』を参照してください。

# ロック アンド キー セキュリティ（ダイナミック アクセス リスト）の設定に関する情報

## ロック アンド キーについて

ロック アンド キーは、IP プロトコル トラフィックを動的にフィルタするトラフィック フィルタリング セキュリティ機能です。ロック アンド キーは、IP ダイナミック拡張アクセス リストを使用して設定されます。ロック アンド キーは、その他の標準アクセス リストとスタティック拡張アクセス リストと共に使用できます。

ロック アンド キーが設定されると、IP トラフィックが通常ルータではブロックされる指定されたユーザは、ルータ経由で一時的なアクセスを得ることができます。起動されると、ロック アンド キーは、指定されたユーザに指定されたホストに到達することを許可するよう、インターフェイスの既存の IP アクセス リストを再設定します。その後、ロック アンド キーは、インターフェイスを元の状態に戻すよう、再設定します。

ユーザがロック アンド キーが設定されたルータを介してホストへのアクセスできるようにするため、ユーザは、最初にルータに Telnet セッションを開く必要があります。ユーザがルータに標準 Telnet セッションを開始すると、ロック アンド キーは、自動的にユーザを認証しようとします。ユーザが認証されると、ルータを通じて、一時的なアクセスを取得し、宛先ホストに到達できます。

## ロック アンド キーの利点

ロックアンドキーは、標準およびスタティック拡張アクセス リストと同じ利点があります（これらの利点については、「アクセス コントロール リスト：概要および指針」で説明します）。ただし、ロック アンド キーには、標準およびスタティック拡張アクセス リストに比べ、次の利点もあります。

- ロック アンド キーは、個々のユーザを認証するために実験機能を使用します。
- ロック アンド キーは、より大きなインターネットワークにおけるより簡素な管理を提供します。

- 多くの場合、ロック アンド キーは、アクセス リストに必要なルータ処理の量を減らします。
- ロック アンド キーは、ネットワーク ハッカーが、ネットワークへの侵入する可能性を減らします。

ロック アンド キーを使用すると、送信元および宛先がホストとなるアクセスをどのユーザに許可するかを指定できます。これらのユーザは、指定されたホストへのアクセスが許可される前に、ユーザ認証プロセスをパスする必要があります。ロック アンド キーは、その他の設定されたセキュリティ制約事項を損なうことなく、ファイアウォールを通じてダイナミック ユーザ アクセスを作成します。

# ロック アンド キーを使用するタイミング

ロック アンド キーを使用するタイミングの 2 つの例を以下に示します。

- 特定のリモート ユーザ（またはリモート ユーザのグループに）が、インターネットを介して、そのリモートホストから接続して、ネットワーク内のホストへのアクセスを必要とする場合。ロック アンド キーは、ユーザを認証し、次に、個々のホストまたはサブネットに対して、限られた時間の間、ファイアウォールを介した限られたアクセスを許可します。
- ローカル ネットワーク上のホストのサブセットがファイアウォールによって保護されたリモート ネットワーク上のホストにアクセスする必要がある場合。ロック アンド キーを使用すると、ローカル ユーザが必要とするホストのセットに対してのみリモート ホストへのアクセスを有効にすることができます。ロック アンド キーは、ホストがリモート ホストリモートへアクセスすることを許可する前に、ユーザがTACACS+サーバ、もしくはその他のサーバを通じて、認証を行うことを必要とします。

# ロック アンド キーの機能

次のプロセスは、ロック アンド キー アクセスの動作を説明します。

1. ユーザは、ロック アンド キー用に設定された境界（ファイアウォール）ルータへの Telnet セッションを開きます。ユーザは、ルータ上の仮想端末ポートを介して接続します。
2. Cisco IOS ソフトウェアは、Telnet パケットを受信し、Telnet セッションを開いてパスワードを要求し、ユーザ認証プロセスを実行します。ユーザは、ルータを介したアクセスが許可される前に、認証をパスする必要があります。認証プロセスは、ルータ、またはTACACS+ または RADIUS サーバなどの中央アクセス セキュリティ サーバで実行することもできます。
3. ユーザが認証をパスすると、Telnet セッションからログアウトし、ソフトウェアがダイナミック アクセス リストに一時的なエントリを作成します。（設定ごとに、この一時エントリは、ユーザが一時的なアクセスを与えられるネットワークの範囲を制限できます。）
4. ユーザは、ファイアウォール経由でのデータを交換します。

5. ソフトウェアは、設定されているタイムアウトに到達するか、システム管理者が手動でクリアした場合に、一時的なアクセス リスト エントリを削除します。設定されているタイムアウトは、アイドルタイムアウトまたは絶対タイムアウトのいずれかになることがあります。

（注） ユーザがセッションを終了させた場合、一時アクセス リスト エントリは、自動的に削除されません。一時アクセスリストのエントリは、設定されているタイムアウトに到達するか、システム管理者がクリアされるまで保持されます。

# Cisco IOS リリース 11.1 以前のリリースとの互換性

**access-list** コマンドの拡張機能は、ロック アンド キーに使用されます。これらの機能拡張は、下位互換性があります。Cisco IOS リリース 11.1 以前のリリースから新しいリリースに移行する場合、アクセスリストは、機能拡張を反映するために、自動的に変換されます。ただし、次の注意の項で説明されているように、Cisco IOS リリース 11.1 以前のリリースでロック アンド キーを使用しようとすると、問題が発生する可能性があります。

注意 Cisco IOS リリース 11.1 以前のリリースは、ロック アンド キー アクセス リスト拡張機能と互換性がありません。そのため、リリース 11.1 以前のソフトウェアでアクセスリストを保存し、このソフトウェアを使用する場合、作成されたアクセスリストは、正しく解釈されません。これによって、深刻なセキュリティ上の問題が発生する可能性があります。これらのファイルと共に画像をブートする前に、Cisco IOS リリース 11.1 以降のソフトウェアを使用して、古い設定ファイルを保存する必要があります。

# ロック アンド キーによるスプーフィングのリスク

注意 ロック アンド キー アクセスを使用すると、外部イベント（Telnet セッション）がファイアウォールに穴を開けることができます。この穴がある間、ルータは、送信元アドレスのスプーフィングを受ける可能性があります。

ロック アンド キー が起動されると、ユーザ アクセスを許可するインターフェイスを一時的に再設定することで、ファイアウォール内に動的な穴が作成されます。この穴がある間は、別のホストが認証済みのユーザのアドレスを偽装し、ファイアウォールの裏でのアクセスを獲得する可能性があります。ロック アンド キーは、アドレス スプーフィングの問題を発生させません。この問題は、ユーザの関心事としてここに特定されるだけです。スプーフィングは、すべてのアクセス リストに伴う問題であり、ロック アンド キーは、この問題に具体的に対処していません。

スプーフィングを防ぐには、リモート ホストからのトラフィックがセキュアなリモート ルータで暗号化され、ロック アンド キーを提供するルータ インターフェイス上でローカルで復号化されるように暗号化を設定します。ルータの入力時に、ロック アンド キーを使用して、すべてのトラフィックを暗号化したい場合、ハッカーは、それらが暗号化を複製できないか、暗号化のセットアップ プロセスの必要な部分として認証できないため、送信元アドレスをスプーフィングすることはできません。

# ロック アンド キーによるルータのパフォーマンスへの影響

ロック アンド キーを設定すると、ルータのパフォーマンスは、次のように影響を受ける場合があります。

- ロック アンド キーが起動されると、ダイナミック アクセス リストは、シリコン スイッチング エンジン（SSE）上でのアクセス リストの再構成が強制されます。これによって、SSE スイッチング パスが一瞬低速になります。
- ダイナミック アクセス リストは、アイドル タイムアウト機能（タイムアウトがデフォルトになったとしても）を必要とし、SSE スイッチングにすることはできません。これらのエントリは、プロトコル ファスト スイッチング パスで処理する必要があります。
- リモート ユーザが境界ルータでロック アンド キーを起動すると、追加のアクセス リスト エントリが境界ルータ インターフェイスで作成されます。インターフェイスのアクセス リストが動的に拡大および縮小します。エントリは、アイドル タイムアウトまたは最大タイムアウト期間が経過すると、動的に削除されます。アクセス リストが大きくなると、パケット交換のパフォーマンスが低下し、パフォーマンスの問題の劣化を通知する場合、ロック アンド キーによって生成された一時アクセス リスト エントリを削除するかどうかを確認するために、境界ルータの設定を確認する必要があります。

# ロック アンド キーの保守

ロック アンド キーを使用中の場合、ダイナミック アクセス リストは、認証エントリの追加および削除に伴って動的に増減します。エントリが存在しても、スプーフィング攻撃のリスクがあるため、タイムリーにエントリが削除されていることを確認する必要があります。また、エントリの数が増えれば、ルータのパフォーマンスへの影響も大きくなります。

アイドルまたは絶対タイムアウトを設定していない場合、エントリは、ダイナミック アクセス リスト エントリを手動で削除するまで維持されます。この場合、エントリの削除について配慮してください。

# ダイナミック アクセス リスト

ダイナミック アクセス リストを設定する場合は、次のガイドラインを参照してください。

- いずれか 1 つのアクセス リストに対して複数のダイナミック アクセス リストを作成しないで下さい。ソフトウェアは、定義された最初のダイナミック アクセス リストだけを参照します。

- 別のアクセス リストに同じ名前を割り当てないで下さい。そうすることで、既存のリストを再利用するように、ソフトウェアに指示します。すべての名前付きエントリは、設定内でグローバルに一意である必要があります。
- スタティック アクセス リストに属性を割り当てるのと同じ方法で、ダイナミック アクセス リストに属性を割り当てます。一時アクセス リスト エントリは、このリストに割り当てられているアトリビュートを継承します。
- ルータ経由でのアクセスが許可される前に、ユーザが認証する必要があるルータに対する Telnet セッションを開く必要があるよう、プロトコルとして Telnet を設定します。
- 今度は、**autocommand** 内の **access-enable** コマンド内の **timeout** キーワードで、アイドル タイムアウトを定義するか、後で、**access-list** コマンドで絶対タイムアウト値を定義します。アイドル タイムアウトまたは絶対タイムアウトを定義する必要があります。そうしないと、一時的なアクセス リスト エントリは、管理者が手動でエントリを削除するまで（ユーザがセッションを終了した後でも）、インターフェイスで永久に設定されたままになります。（必要に応じて、アイドル タイムアウトと絶対タイムアウトの両方を設定することもできます）。
- アイドル タイムアウトを設定する場合、アイドル タイムアウト値は、WAN アイドル タイムアウト値と等しくなる必要があります。
- アイドル タイムアウトと絶対タイムアウトの両方を設定する場合、アイドル タイムアウト値は、絶対タイムアウト値未満である必要があります。
- ジョブが ACL の絶対タイマーを超えて動作していることを認識した場合、**access-list dynamic-extend** コマンドを使用して、6 分ほどダイナミック ACL の絶対タイマーを拡張します。このコマンドにより、ロック アンド キーを使用して、自身を再認証するため、ルータに新しい Telnet セッションを開くことができます。
- 一時的なエントリで置換される唯一の値は、入力アクセス リストまたは出力アクセス リスト内にアクセス リストがあったかどうかに応じて、送信元または宛先アドレスになります。ポートなどの他の属性はすべて、メインのダイナミック アクセス リストから引き継がれます。
- ダイナミック リストへの追加はそれぞれ、ダイナミック リストの先頭に常に配置されます。一時アクセス リスト エントリの順序を指定することはできません。
- 一時アクセス リスト エントリが NVRAM には書き込まれません。
- ダイナミック アクセス リストを手動でクリアまたは表示するには、この章で後述される「ロック アンド キーの維持」を参照して下さい。

# ロック アンド キー認証

認証問い合せプロセスを設定するには、3 つの方法があります。この項では、これら 3 つの方法について説明します。

（注） Ciscoは、認証問い合せプロセスには、TACACS+サーバを使用することを推奨します。TACACS+は、認証、許可、アカウンティング サービスを提供します。また、プロトコル サポート、プロトコル仕様、および中央集中型セキュリティ データベースも提供します。TACACS+ サーバの使用については、次項「方法 1 -- セキュリティ サーバの設定」で説明します。

TACACS+ サーバなどのネットワーク アクセス セキュリティ サーバを使用します。この方法には、TACACS+ サーバでの追加設定手順が必要になりますが、より厳しい認証問い合わせとより高度な追跡機能が可能になります。

```
Router(config-line)# login tacacs
```

**username** コマンドを使用します。この方法では、認証はユーザ単位で決定するため、効果的です。

```
Router(config)# username

name
 {nopassword
 |
password
 {
mutual-password
 |
encryption-type

encryption-password
}}
```

**password** および **login** コマンドを使用します。この方法は、パスワードがユーザではなく、このポートに設定されているため、有効でありません。そのため、パスワードを知っているすべてのユーザが正常に認証できます。

```
R
outer(config-line)# password

password
Router(config-line)# login local
```

# autocommand コマンド

**autocommand** コマンドは、ユーザが特定の回線に接続する際に、システムが指定されている特権 EXEC コマンドを自動的に実行するように設定します。**autocommand** コマンドの設定のための次のガイドラインを使用します。

- ユーザを認証するために TACACS+ サーバを使用する場合、TACACS+ サーバ上で、ユーザごとの autocommand として、**autocommand** コマンドを設定する必要があります。ローカル認証を使用する場合、回線上で **autocommand** コマンドを使用します。
- 同じ **autocommand** コマンドで、すべての仮想端末（VTY）ポートを設定します。VYT ポートで **autocommand** コマンドを省略すると、任意のホストがルータの特権 EXEC モー

ドへのアクセスを許可し、ダイナミック アクセス リスト内の一時アクセス リスト エントリを作成しません。

- **autocommand access-enable** コマンドでアイドル タイムアウトを定義しない場合、**access-list** コマンドで絶対タイムアウトを定義する必要があります。アイドル タイムアウトまたは絶対タイムアウトを定義する必要があります。そうしないと、一時的なアクセス リスト エントリは、エントリが管理者によって手動で削除されるまで（ユーザがセッションを終了した後も）インターフェイスで永久に設定されたままになります。（必要に応じて、アイドル タイムアウトと絶対タイムアウトの両方を設定することもできます）。
- アイドル タイムアウトと絶対タイムアウトの両方を設定する場合、絶対タイムアウト値は、アイドル タイムアウト値よりも大きくする必要があります。

# ロック アンド キーセキュリティ（ダイナミック アクセス リスト）の設定方法

## ロック アンド キーの設定

ロック アンド キーを設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。次の手順を実行する際、この章の「ロック アンド キー設定のガイドライン」に記載されているガイドラインに従っていることを確認します。

### 手順の概要

1. Router(config)# **access-list** *access-list-number* [**dynamic** *dynamic-name* [**timeout** *minutes*]] {**deny** | **permit**} **telnet** *source source-wildcard destination destination-wildcard*[ **precedence** *precedence*] [ **tos** *tos*] [**established**] [**log**]
2. Router(config)# **access-list dynamic-extend**
3. Router(config)# **interface** *type number*
4. Router(config-if)# **ip access-group** *access-list-number*
5. Router(config-if)# **exit**
6. Router(config)# **line vty** *line-number* [*ending-line-number*]
7. 次のいずれかを実行します。
   - Router(config-line)# **login tacacs**
   -
   - Router(config-line)# **password** *password*
8. 次のいずれかを実行します。
   - Router(config-line)# **autocommand access-enable** [**host**] [**timeout** *minutes*]
   -
   - Router# **access-enable** [**host**] [**timeout** *minutes*]

### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | Router(config)# **access-list** *access-list-number* [**dynamic** *dynamic-name* [**timeout** *minutes*]] {**deny** \| **permit**} **telnet** *source source-wildcard destination destination-wildcard*[ **precedence** *precedence*] [ **tos** *tos*] [**established**] [**log**] | 一時アクセス リスト エントリのテンプレートとプレースホルダとして動作するダイナミック アクセス リストを設定します。 |
| ステップ 2 | Router(config)# **access-list dynamic-extend** | （任意）ロック アンド キーを使用して、自分の再認証を実行するようにルータに別の Telnet セッションを開く際に、6 分ごとのダイナミック ACL の絶対タイマーを拡張します。ジョブが ACL の絶対タイマー前を実行する場合に、このコマンドを使用します。 |
| ステップ 3 | Router(config)# **interface** *type number* | インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。 |
| ステップ 4 | Router(config-if)# **ip access-group** *access-list-number* | アクセス リストをインターフェイスに適用します。 |
| ステップ 5 | Router(config-if)# **exit** | インターフェイス コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに入ります。 |
| ステップ 6 | Router(config)# **line vty** *line-number* [*ending-line-number*] | 1 つ以上の仮想端末（VTY）ポートを定義し、ライン コンフィギュレーション モードを開始します。複数の VTY ポートを指定する場合、ソフトウェアがラウンドロビン ベースで使用可能な VTY ポートをハントするため、個別に設定する必要がありません。ロック アンド キー アクセスに対して、すべての VTY ポートを設定しない場合、ロック アンド キー サポートに対してのみ、VTY ポートのグループを指定できます。 |
| ステップ 7 | 次のいずれかを実行します。<br>• Router(config-line)# **login tacacs**<br>•<br>• Router(config-line)# **password** *password*<br>例：<br>`Router(config-line)# login local`<br>例：<br>`Router(config-line)# exit`<br>例： | 回線またはグローバル コンフィギュレーション モードでユーザ認証を設定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | then<br>例：<br>Router(config)# **username** *name* **password** *secret* | |
| ステップ 8 | 次のいずれかを実行します。<br>• Router(config-line)# **autocommand access-enable** [**host**] [**timeout** *minutes*]<br>•<br>• Router# **access-enable** [**host**] [**timeout** *minutes*] | 回線設定または特権EXECモードの一時アクセス リスト エントリを作成できます。<br>回線設定モードで **access-enable** コマンドとともに **autocommand** を使用して、回線が接続されたときに、自動的にダイナミック アクセス リスト上の一時アクセス リスト エントリを作成するようシステムを設定します。<br>任意の **host** キーワードを指定しないと、ネットワーク全体のすべてのホストが一時アクセス リスト エントリを設定できます。ダイナミック アクセス リストには、新しいネットワーク接続を許可するためのネットワーク マスクが含まれます。<br>任意の **timeout** キーワードを指定すると、一時アクセス リストに対するアイドル タイムアウトを定義します。<br>有効値の範囲は 1 ～ 9999（分）です。 |

# ロック アンド キーの設定の確認

ユーザに接続をテストするように求めることで、ロック アンド キーがルータで正しく設定されていることを確認できます。ユーザは、ダイナミック アクセス リストで許可されるホストである必要があり、ユーザは、AAA 認証および許可を設定する必要があります。

接続をテストするには、ユーザは、ルータへの Telnet 接続を行い、Telnet セッションを閉じる許可をし、ルータの反対側のホストへのアクセスを試みる必要があります。このホストは、ダイナミック アクセス リストによって許可されているものである必要があります。ユーザは、IP プロトコルを使用するアプリケーションのあるホストにアクセスする必要があります。

次の例は、エンドユーザが正常に認証された場合に、何が見えるかを示しています。パスワードが入力され、認証された後に、Telnet 接続は閉じられます。一時アクセス リスト エントリが作成され、Telnet セッションを開始したホストがファイアウォールの内側のホストにアクセスします。

```
Router% telnet corporate
Trying 172.21.52.1 ...
Connected to corporate.example.com.
Escape character is '^]'.
```

```
User Access Verification
Password:Connection closed by foreign host.
```

ユーザは、ルータで **show access-lists** コマンドを使用して、ルータを介して、ユーザのアクセスを許可する別のエントリを含む、ダイナミック アクセス リストを表示できます。

## ダイナミック アクセス リスト エントリの表示

一時アクセス リスト エントリは、使用中に表示できます。一時アクセス リスト エントリがユーザまたは絶対またはアイドル タイムアウト パラメータによってクリアされた後は表示されなくなります。表示される一致の数は、アクセス リスト エントリがヒットした回数を示します。

現在確立されているダイナミック アクセス リスト エントリ リストおよび一時アクセス リスト エントリ リストを表示するには、特権 EXEC モードで次のコマンドを使用します。

| コマンド | 目的 |
|---|---|
| Router# **show access-lists** [*access-list-number*] | ダイナミック アクセス リストおよび一時アクセス リスト エントリを表示します。 |

## ダイナミック アクセス リスト エントリの手動削除

一時アクセス リスト エントリを手動で削除するには、特権 EXEC モードで次のコマンドを使用します。

| コマンド | 目的 |
|---|---|
| Router# **clear access-template** [*access-list-number* \| *name*] [*dynamic-name*] [*source*] [*destination*] | ダイナミック アクセス リストを削除します。 |

# ロック アンド キーの設定例

## ローカル認証を使用したロック アンド キーの例

この例は、ルータで局所的に生じた認証を使って、ロック アンド キー アクセスを設定する方法を示しています。ロック アンド キーは、Ethernet 0 インターフェイスとして設定されます。

```
interface ethernet0
 ip address 172.18.23.9 255.255.255.0
 ip access-group 101 in
access-list 101 permit tcp any host 172.18.21.2 eq telnet
access-list 101 dynamic mytestlist timeout 120 permit ip any any
line vty 0
```

```
login local
autocommand access-enable timeout 5
```

最初の access-list エントリは、ルータに Telnet だけを許可します。2 番目のアクセス リスト エントリは、ロック アンド キーがトリガーされるまで常に無視されます。

**access-list** コマンドでは、タイムアウトは絶対タイムアウトです。この例では、mytestlist ACL の有効期間は、120 分です。つまり、ユーザがログインし、**access-enable** コマンドを有効にすると、120 分間（最大絶対時間）有効なダイナミック ACL が作成されます。セッションは使用者の有無に関係なく、120 分後に閉じられます。

**access-enable** コマンドでは、タイムアウトは、アイドル タイムアウトです。この例では、ユーザがログインまたは認証するたびに 5 分間セッションがあります。アクティビティがないと、セッションは 5 分後に終了し、ユーザを再認証する必要があります。ユーザが接続を使用すると、絶対時間が作用し、セッションは 120 分後に終了します。

ユーザがルータへの Telnet セッションを開いた後、ルータはユーザを認証しようとします。認証に成功すると、**autocommand** が実行され、Telnet セッションが終了します。**autocommand** は、2 番目のアクセス リスト エントリ（mytestlist）に基づいて、イーサネット 0 インターフェイスで一時的な着信アクセス リスト エントリを作成します。アクティビティがない場合、タイムアウトで規定されているように、この一時エントリは 5 分後に無効となります。

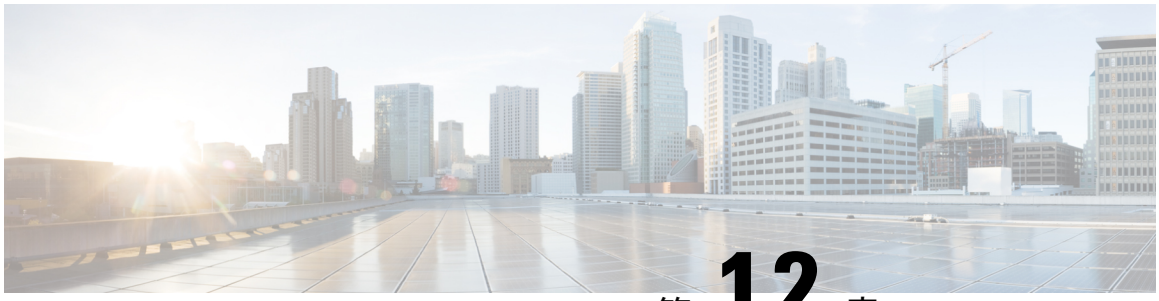
# TACACS+ 認証を使用したロック アンド キーの例

Cisco は、認証に TACACS+ サーバを使用することを推奨します。以下の例を参照して下さい。

以下の例は、TACACS+ サーバでの認証を使用して、ロック アンド キーを設定する方法について説明しまています。ロック アンド キー アクセスは、BRI0 インターフェイスで設定されます。4 つのポートは、VTY パスワード「password1」として定義されています。

```
aaa authentication login default group tacacs+ enable
aaa accounting exec stop-only group tacacs+
aaa accounting network stop-only group tacacs+
enable password ciscotac
!
isdn switch-type basic-dms100
!
interface ethernet0
ip address 172.18.23.9 255.255.255.0
!
interface BRI0
 ip address 172.18.21.1 255.255.255.0
 encapsulation ppp
 dialer idle-timeout 3600
 dialer wait-for-carrier-time 100
 dialer map ip 172.18.21.2 name dialermapname
 dialer-group 1
 isdn spid1 2036333715291
 isdn spid2 2036339371566
 ppp authentication chap
 ip access-group 102 in
!
access-list 102 permit tcp any host 172.18.21.2 eq telnet
access-list 102 dynamic testlist timeout 5 permit ip any any
!
!
```

```
ip route 172.18.250.0 255.255.255.0 172.18.21.2
priority-list 1 interface BRI0 high
tacacs-server host 172.18.23.21
tacacs-server host 172.18.23.14
tacacs-server key test1
tftp-server rom alias all
!
dialer-list 1 protocol ip permit
!
line con 0
 password password1
line aux 0
 line VTY 0 4
 autocommand access-enable timeout 5
 password password1
!
```



第 12 章

# ACL IP オプションの選択的ドロップ

ACL IP オプションの選択的ドロップ機能を使用すると、Cisco ルータが IP オプションが設定されたパケットをフィルタしたり、ルータまたはダウンストリーム ルータ上での IP オプションの影響を軽減したりすることができるようになります。これは、これらのパケットをドロップするか、IP オプションの処理を無視することによって行われます。



## 機能情報の確認

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

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

## ACL IP オプションの選択的ドロップの制約事項

リソース予約プロトコル（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（MPLS TE））、Internet Group Management Protocol バージョン 2（IGMPv2）、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、ドロップまたは無視モードでは機能しない可能性があります。

# ACL IP オプションの選択的ドロップに関する情報

## ACL IP オプションの選択的ドロップの使用

ACL IP オプションの選択的ドロップ機能を使用すると、IP オプションが設定されたパケットをルータでフィルタできるようになります。これにより、これらのパケットのルータまたはダウンストリーム ルータへの影響を軽減し、次の手順を実行できます。

- 受信した IP オプション パケットをすべてドロップし、オプションがネットワークの奥深くまで入り込まないようにします。
- そのルータ宛ての IP オプション パケットを無視し、IP オプションが設定されていないものとして扱います。

多くのユーザにとっては、パケットのドロップが最善策であると言えます。ただし、正規の IP オプションが存在する可能性のある環境では、ルータ上のパケットのロード処理を減らすだけで十分です。したがって、ルータ上のオプション処理をスキップしたうえで、ピュア IP であるかのようにパケットを転送することができます。

## ACL IP オプションの選択的ドロップを使用する利点

- ドロップ モードでは、ネットワークからのパケットをフィルタすることで、オプション パケットからロードするというダウンストリームルータおよびホストの負荷を軽減できます。
- ドロップ モードでは、分散システム上でのルート プロセッサ（RP）処理が必要となるオプションの RP へのロードが最小限に抑えられます。以前は、パケットは常に RP CPU でルーティングまたは処理されていました。現在は、無視またはドロップすることで、パケットが RP パフォーマンスに影響を及ぼすことを回避できます。

# ACL IP オプションの選択的ドロップの設定方法

## ACL IP オプションの選択的ドロップの設定

ここでは、ACL IP オプションの選択的ドロップ機能を設定する方法について説明します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip options** {**drop** | **ignore**}
4. **exit**

5. **show ip traffic**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Router> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Router# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip options** {**drop** \| **ignore**}<br>例：<br>`Router(config)# ip options drop` | ルータに送信された IP オプション パケットをドロップまたは無視します。 |
| ステップ 4 | **exit**<br>例：<br>`Router(config)# exit` | 特権 EXEC モードに戻ります。 |
| ステップ 5 | **show ip traffic**<br>例：<br>`Router# show ip traffic` | （任意）IP トラフィックの統計情報を表示します。 |

# ACL IP オプションの選択的ドロップの設定例

## 例：ACL IP オプションの選択的ドロップの設定

次に、ネットワークに入ったすべてのオプション パケットをドロップするように、ルータ（およびダウンストリーム ルータ）を設定する例を示します。

```
Router(config)# ip options drop
% Warning:RSVP and other protocols that use IP Options packets may not function in drop
 or ignore modes.
end
```

# 例：ACL IP オプションの選択的ドロップの確認

この出力例は、**ip options drop** コマンドを使用した後に表示されます。

```
Router# show ip traffic
IP statistics:
  Rcvd:  428 total, 323 local destination
         0 format errors, 0 checksum errors, 0 bad hop count
         0 unknown protocol, 0 not a gateway
         0 security failures, 0 bad options, 0 with options
  Opts:  0 end, 0 nop, 0 basic security, 0 loose source route
         0 timestamp, 0 extended security, 0 record route
         0 stream ID, 0 strict source route, 0 alert, 0 cipso, 0 ump
         0 other, 30 ignored
  Frags: 0 reassembled, 0 timeouts, 0 couldn't reassemble
         0 fragmented, 0 fragments, 0 couldn't fragment
  Bcast: 0 received, 0 sent
  Mcast: 323 received, 809 sent
  Sent:  809 generated, 591 forwarded
  Drop:  0 encapsulation failed, 0 unresolved, 0 no adjacency
         0 no route, 0 unicast RPF, 0 forced drop, 0 unsupported-addr
         0 options denied, 0 source IP address zero
```

# IP アクセス リスト エントリ シーケンス番号の追加情報

ここでは、IP アクセス リストに関する関連資料について説明します。

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| IP アクセス リストの設定 | 『Creating an IP Access List and Applying It to an Interface』 |
| Cisco IOS コマンド | 『Cisco IOS Master Command List, All Releases』 |
| IP アクセス リスト コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |

### シスコのテクニカル サポート

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

# ACL IP オプションの選択的ドロップに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *12 : ACL IP* オプションの選択的ドロップに関する機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| ACL IP オプションの選択的ドロップ | Cisco IOS XE リリース 2.1 | ACL IP オプションの選択的ドロップ機能を使用すると、Cisco ルータが IP オプションが設定されたパケットをフィルタしたり、ルータまたはダウンストリーム ルータ上での IP オプションの影響を軽減したりすることができるようになります。これは、これらのパケットをドロップするか、IP オプションの処理を無視することによって行われます。<br>この機能は、Cisco ASR 1000 シリーズのアグリゲーション サービス ルータで導入されました。<br>次のコマンドが導入されました。**ip options** |

第 13 章

# ACL 管理性を使用した IP アクセス リスト データの表示及びクリア

このモジュールでは、IP アクセス リスト内のエントリおよび各エントリに一致したパケットの数の表示方法について説明します。ユーザは、ACL 管理性機能を使用して、グローバルに、または、インターフェイスごとのおよび着信または発信トラフィック方向ごとにこれらの統計情報を取得できます。ネットワークデバイスのさまざまなインターフェイス上の着信または発信トラフィックパターンの詳細表示は、特定のインターフェイスへの攻撃に対してデバイスの保護に役立ちます。このモジュールでは、また、アクセス リスト エントリに一致するパケットの数が 0 から再開されるカウンタをクリアする方法について説明します。



## 機能情報の確認

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

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

# ACL 管理性を使用した IP アクセス リスト データの表示及びクリアに関する情報

## ACL 管理性の利点

Cisco IOS リリース 12.4(6)T 以前では、Cisco IOS ソフトウェア内の ACL インフラストラクチャは、ACL 内の各 ACE に対するグローバル統計情報を維持するだけでした。この方法によって、1 つの ACL が複数のインターフェイスに適用される場合、維持された ACE 統計情報は、その ACL が適用されるすべてのインターフェイス上で一致（ヒット）する着信および発信パケットの合計数となります。

ただし、ACE の統計情報がインターフェイスごとおよび着信または発信トラフィック方向ごとに維持される場合、ネットワーク デバイスの様々なインターフェイスにおける着信および発信トラフィック パターンの特定の詳細および ACE の効率性を表示できます。このような情報は、特定のインターフェイス上に着信する攻撃に対するデバイスの保護に役立ちます。

## インターフェイス レベルの ACL 統計情報のサポート

Cisco IOS リリース 12.4(6)T により、Cisco IOS ソフトウェア内の ACL インフラストラクチャは、インターフェイスごとの、および ACL に対する着信または発信トラフィック方向ごとの ACE 統計情報の保守、表示、およびクリアをサポートするよう、拡張されます。このサポートは、『インターフェイス レベルの統計情報のサポート』と呼ばれます。

（注） 同じアクセス グループ ACL が他の機能によっても使用された場合、保持されているインターフェイス統計情報は、パケット一致が他の機能によって検出される際に、更新されません。この例では、ACL のために維持される、すべてのインターフェイス レベル統計情報の合計は、その ACL に対するグローバル統計情報を集約していない場合があります。

# IP アクセス リスト データを表示およびクリアする方法

この項には、IP アクセス リストおよび各リストに一致（ヒット）するパケットの数を表示し、IP アクセス リスト カウンタをクリアするための次の手順が含まれます。

（注） 特定のホストまたはネットワークに対するアクセスを拒否し、そのネットワークまたはホストの誰かがアクセスしようとしたかどうかを検出する場合、対応する **deny** ステートメントを指定した **log** キーワードを含めます。それによって、その送信元からの拒否されたパケットがログに記録されます。詳細については、「IP アクセス リストの概要」の「IP アクセス リスト ロギング」を参照して下さい。

# グローバル IP ACL 統計情報の表示

ルータ上のすべての IP アクセス リストと一致したパケット数を表示するには、次の作業を実行します。

### 手順の概要

1. **enable**
2. **show ip access-list** [*access-list-number* | *access-list-name*]

### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Router> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **show ip access-list** [*access-list-number* | *access-list-name*]<br>例：<br>`Router# show ip access-list limited` | IP アクセス リスト情報を表示します。<br>• この例では、「名前付きアクセス リストを指定します」を使用するすべてのインターフェイスの統計情報を表示します。 |

# インターフェイス レベル IP ACL 統計情報の表示

このセクションでは、インターフェイスに ACL 用の着信または発信トラフィック方向ごとの IP ACE の統計情報を表示する方法について説明します。この機能は、ACL 管理性と呼ばれています。

（注）
- ACL 管理性サポート対象：
  - 非分散型プラットフォーム ソフトウェアでスイッチングされるだけです。
  - 標準と拡張の静的に設定された ACL と脅威緩和サービス（TMS）ダイナミック ACE です。
- ACL 管理性サポート対象外：
  - ファイアウォールおよび認証プロキシなど、再帰かつユーザ設定のダイナミック ACL およびダイナミック ACE ブロック。
  - 仮想テンプレートおよび仮想アクセス インターフェイス。

>

### 手順の概要

1. **enable**
2. **show ip access-list interface** *interface-name* [**in**| **out**]

### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Router> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **show ip access-list interface** *interface-name* [**in**| **out**]<br>例：<br>`Router# show ip access-list interface FastEthernet 0/0 in` | IP アクセス リスト情報を表示します。<br>• この例では、FastEthernet インターフェイスに着信するトラフィックに関する統計情報を表示します。<br>• ACL のインターフェイス レベルの統計情報に関するデバッグ情報を表示するには、**debug ip access-list intstats** コマンドを使用します。 |

# アクセス リスト カウンタのクリア

システムは、アクセス リストの各行に一致（ヒット）するパケットの数を数えます。カウンタは、**show access-lists** EXEC コマンドで表示されます。この作業を行い、アクセス リストのカウンタをクリアします。アクセス リストに一致するゼロから始まるパケットの数を決定しようとする場合に、これを行うことができます。

手順の概要

1. **enable**
2. **clear ip access-list counters** {*access-list-number* | *access-list-name*}

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Router> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **clear ip access-list counters** {*access-list-number* \| *access-list-name*}<br>例：<br>`Router# clear access-list counters corpmark` | IP アクセス リストのカウンタをクリアします。 |

# ACL 管理性を使用した IP アクセス リスト データの表示及びクリアのための設定例

## グローバル IP ACL 統計情報を表示する例

次に、ACL 150 のグローバル統計情報を表示する例を示します。

```
Router# show ip access-list 150

Extended IP access list 150
   10 permit ip host 10.1.1.1 any (3 matches)
   30 permit ip host 10.2.2.2 any (27 matches)
```

## 入力統計情報を表示する例

次の例は、アクセス リスト 150（ACL 番号）に関連付けられているインターフェイス FastEthernet 0/1 から集めた着信パケットの統計情報を示しています。

```
Router#
 show ip access-list interface FastEthernet 0/1 in
Extended IP access list 150 in
   10 permit ip host 10.1.1.1 any (3 matches)
   30 permit ip host 10.2.2.2 any (12 matches)
```

# 出力統計情報を表示する例

次の例は、FastEthernet 0/0 インターフェイスから集めた出力パケットに関する統計情報を示しています。

```
Router#
 show ip access-list interface FastEthernet 0/0 out
Extended IP access list myacl out
    5 deny ip any 10.1.0.0 0.0.255.255
    10 permit udp any any eq snmp (6 matches)
```

# 入出力統計情報を表示する例

（注） 方向を指定しないと、そのインターフェイスに適用された入出力 ACL が表示されます。

次の例の表示から集めた入出力統計情報は、FastEthernet 0/0 を実行します。

```
Router#
 show ip access-list interface FastEthernet 0/0
Extended IP access list 150 in
   10 permit ip host 10.1.1.1 any
   30 permit ip host 10.2.2.2 any (15 matches)
Extended IP access list myacl out
    5 deny ip any 10.1.0.0 0.0.255.255
    10 permit udp any any eq snmp (6 matches)
```

# IPアクセスリスト用のグローバルおよびインターフェイス統計情報のクリアの例

次の例では、IP ACL 150 のグローバルおよびインターフェイスの統計情報をクリアします。

```
Router#
 clear ip access-list counters 150
```

# すべての IP アクセス リスト用のグローバルおよびインターフェイス統計情報のクリアの例

次の例では、すべての IP ACL のグローバルおよびインターフェイスの統計情報をクリアします。

```
Router#
 clear ip access-list counters
```

# その他の参考資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Commands List, All Releases』 |
| セキュリティ コマンド | 『*Cisco IOS Security Command Reference*』 |

### 標準

| 標準 | タイトル |
|---|---|
| この機能がサポートする新しい規格または変更された規格はありません。 | -- |

### MIB

| MIB | MIB のリンク |
|---|---|
| この機能によってサポートされる新しい MIB または変更された MIB はありません。 | 選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。<br>http://www.cisco.com/go/mibs |

### RFC

| RFC | タイトル |
|---|---|
| この機能でサポートされる新規の規格または変更された規格はありません。また、既存の規格のサポートは変更されていません。 | -- |

### テクニカル サポート

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

# IPアクセス リスト情報の表示およびカウンタのクリアに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *13 : ACL* 管理性を使用した *IP* アクセス リスト データの表示及びクリアのための機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| ACL 管理性 | Cisco IOS XE Release 3.9S | ACL 管理性機能により、ユーザは、インターフェイスおよびアクセス コントロール リスト（ACL）に対する入力や出力トラフィック方向ごとのアクセス コントロール エントリ（ACE）の統計情報を表示およびクリアすることができます。 |

第 14 章

# ACL Syslog 相関

アクセス コントロール リスト（ACL）Syslog 相関機能では、アクセス コントロール エントリ（ACE）Syslog エントリにタグ（ユーザ定義の Cookie またはデバイスが生成した MD5 ハッシュ値）を追加します。このタグは Syslog エントリを生成した ACL 内で ACE を一意に特定します。



## 機能情報の確認

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

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

## ACL Syslog 相関の前提条件

ACL Syslog 相関機能を設定する前に、「IP アクセス リストの概要」モジュールでその概念を理解する必要があります。

ACL Syslog 相関機能は、ユーザ定義の cookie またはデバイスで生成されるハッシュ値を syslog 内の ACE メッセージに追加します。ログ オプションが ACE に対してイネーブルになっている場合、これらの値は ACE メッセージにのみ追加されます。

# ACL Syslog 相関に関する情報

## ACL Syslog 相関タグ

ACL Syslog 相関機能では、アクセス コントロール エントリ（ACE）Syslog エントリにタグ（ユーザ定義の Cookie またはデバイスが生成した MD5 ハッシュ値）を追加します。このタグは Syslog エントリを生成した ACE を一意に特定します。

ネットワーク管理ソフトウェアでは、どの ACE が特定の Syslog イベントを生成したかを特定するためにタグを使用できます。たとえば、ネットワーク管理者はネットワーク管理アプリケーションで ACE 規則を選択し、次にその ACE ルールに対応する Syslog イベントを表示できます。

Syslog メッセージにタグを追加するには、Syslog イベントを生成する ACE でログ オプションが有効になっている必要があります。システムは各メッセージに 1 つのタイプのタグ（ユーザ定義の Cookie またはデバイスで生成した MD5 ハッシュ値）のみを追加します。

ユーザ定義の Cookie タグを指定するには、ユーザは ACE ログ オプションを構成する際に Cookie 値を入力する必要があります。Cookie は英数字形式である必要があります。64 文字以上にはできず、16 進数表記（0x など）で始めることはできません。

デバイスで生成した MD5 ハッシュ値タグを指定するには、ハッシュ生成機能をデバイスで有効にする必要があります。また、ACE ログ オプションを構成するときにユーザは Cookie 値を入力してはいけません。

## ACE Syslog メッセージ

パケットが ACL 内のアクセス コントロール エントリ（ACE）と一致すると、そのイベントのログ オプションが有効になっているかどうかシステムでチェックされます。ログ オプションが有効な場合、ACL Syslog 相関機能がデバイスで構成されていると、システムは syslog メッセージにタグを付けます。タグは、標準情報に加えて syslog メッセージの最後に表示されます。

次は、ユーザ定義の Cookie タグを示すサンプル syslog メッセージです。

```
Jun 5 12:55:44.359: %SEC-6-IPACCESSLOGP: list logacl permitted tcp 192.168.16.1(38402)
-> 192.168.16.2(23), 1 packet [User_permiited_ACE]
```

次は、ハッシュ値タグを示すサンプル syslog メッセージです。

```
Jun 5 12:55:44.359: %SEC-6-IPACCESSLOGP: list logacl permitted tcp 192.168.16.1(38402)
-> 192.168.16.2(23), 1 packet [0x723E6E12]
```

# ACL Syslog 相関の設定方法

## デバイスでのハッシュ値生成の有効化

ユーザ定義 Cookie を使用して設定されていないシステム内でログをイネーブルにした各アクセス コントロール エントリ（ACE）の MD5 ハッシュ値を生成するデバイスを設定するには、このタスクを実行します。

ハッシュ値生成設定をイネーブルにすると、システムは既存のすべての ACE をチェックし、ハッシュ値を必要とする各 ACE のハッシュ値を生成します。ハッシュ値生成の設定をディセーブルにすると、これまでに生成されたすべてのハッシュ値がシステムから削除されます。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list logging hash-generation**
4. **end**
5. 次のいずれかを実行します。
   - **show ip access-list** *access-list-number*
   - **show ip access-list** *access-list-name*

### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip access-list logging hash-generation**<br>例：<br>`Device(config)# ip access-list logging hash-generation` | デバイスでハッシュ値生成を有効にします。<br>• ログを有効にした ACE があり、ハッシュ値を必要とする場合、デバイスは自動的に値を生成し、コンソールでその値を表示します。 |
| ステップ 4 | **end**<br>例： | （任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config)# end` | |
| ステップ 5 | 次のいずれかを実行します。<br>• **show ip access-list** *access-list-number*<br>• **show ip access-list** *access-list-name*<br>例：<br>`Device# show ip access-list 101`<br>例：<br>`Device# show ip access-list acl` | （任意）番号付きまたは名前付き IP アクセス リストの内容を表示します。<br>• ログをイネーブルにした ACE のアクセス リストに生成したハッシュ値が含まれることを確認するには、出力を見直します。 |

# デバイスでのハッシュ値生成の無効化

デバイスでのハッシュ値生成をディセーブルにするには、このタスクを実行します。ハッシュ値生成の設定をディセーブルにすると、これまでに生成されたすべてのハッシュ値がシステムから削除されます。

## 手順の概要

1. **enable**
2. **configure terminal**
3. **no ip access-list logging hash-generation**
4. **end**
5. 次のいずれかを実行します。
   • **show ip access-list** *access-list-number*
   • **show ip access-list** *access-list-name*

## 手順の詳細

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

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 3 | **no ip access-list logging hash-generation**<br>例：<br>`Device(config)# no ip access-list logging hash-generation` | デバイスでのハッシュ値生成をディセーブルにします。<br>• これまでに作成されたハッシュ値がシステムから削除されます。 |
| ステップ 4 | **end**<br>例：<br>`Device(config)# end` | （任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |
| ステップ 5 | 次のいずれかを実行します。<br>• **show ip access-list** *access-list-number*<br>• **show ip access-list** *access-list-name*<br>例：<br>`Device# show ip access-list 101`<br>例：<br>`Device# show ip access-list acl` | （任意）IP アクセス リストの内容を表示します。<br>• ログをイネーブルにした ACE のアクセス リストに生成したハッシュ値が含まれないことを確認するには、出力を見直します。 |

# ユーザ定義 Cookie を使用した ACL Syslog 相関の設定

syslog メッセージ タグとしてユーザ定義の Cookie クッキーを使用し、特定のアクセス リストのデバイス上の ACL syslog 相関機能を設定するには、このタスクを実行します。

このセクションでは、番号付きアクセス リストのユーザ定義の Cookie を使用して、ACL Syslog 相関機能を設定する方法について例を示します。ただし、番号付きおよび名前付きアクセス リストの両方、標準および拡張アクセス リストの両方について、ユーザ定義の Cookie を使用し、ACL Syslog 相関機能を設定できます。

（注） 次の制限事項は、ユーザ定義の Cookie 値を選択する場合に適用されます。

- 最大文字数は 64 です。
- Cookie は 16 進表記（0x など）で始めることはできません。
- Cookie は、**reflect**、**fragment**、**time-range** といったキーワードと同じまたはその一部を使用することはできません。たとえば、reflect と ref は無効な値です。ただし、これらのキーワードを先頭に使用することはできます。たとえば、reflectedACE と fragment_33 は有効な値です。
- Cookie に設定できるのは英数字のみです。

>

## 手順の概要

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **permit** *protocol source destination* **log** *word*
4. **end**
5. **show ip access-list** *access-list-number*

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **access-list** *access-list-number* **permit** *protocol source destination* **log** *word*<br>例：<br>`Device(config)# access-list 101 permit tcp host 10.1.1.1 host 10.1.1.2 log UserDefinedValue` | 拡張 IP アクセス リストとユーザ定義の Cookie 値を定義します。<br>• Cookie 値の引数として *word* を入力します。 |
| ステップ 4 | **end**<br>例： | （任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config)# end` | |
| ステップ 5 | **show ip access-list** *access-list-number*<br>例：<br>`Device# show ip access-list 101` | （任意）IP アクセス リストの内容を表示します。<br>• 出力を見直して、アクセス リストにユーザ定義の Cookie 値が含まれることを確認します。 |

**例**

次に、ユーザ定義の Cookie 値を使用したアクセス リストに **show ip access-list** コマンドを使用した際の出力例を示します。

```
Device# show ip access-list
101
Extended IP access list 101
30 permit tcp host 10.1.1.1 host 10.1.1.2 log (tag = UserDefinedValue)
```

# ハッシュ値を使用した ACL Syslog 相関の設定

syslog メッセージ タグとしてデバイスで生成されたハッシュ値を使用し、特定のアクセス リストのデバイス上の ACL Syslog 相関機能を設定するには、このタスクを実行します。

このセクションでは、番号付きアクセス リストのデバイスで生成されたハッシュ値を使用して、ACL Syslog 相関機能を設定する方法についてステップを示します。ただし、番号付きおよび名前付きアクセス リストの両方、標準および拡張アクセス リストの両方について、デバイスで生成されたハッシュ値を使用し、ACL Syslog 相関機能を設定できます。

## 手順の概要

1. **enable**
2. **configure terminal**
3. **ip access-list logging hash-generation**
4. access-list *access-list-number* **permit** *protocol source destination* **log**
5. **end**
6. **show ip access-list** *access-list-number*

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ip access-list logging hash-generation**<br>例：<br>`Device(config)# ip access-list logging hash-generation` | デバイスでハッシュ値生成を有効にします。<br>• ログを有効にした ACE があり、ハッシュ値を必要とする場合、デバイスは自動的に値を生成し、コンソールでその値を表示します。 |
| ステップ 4 | access-list *access-list-number* **permit** *protocol source destination* **log**<br>例：<br>`Device(config)# access-list 102 permit tcp host 10.1.1.1 host 10.1.1.2 log` | 拡張 IP アクセス リストを定義します。<br>• アクセス リストのログ オプションを有効にしますが、Cookie 値は指定しないでください。<br>• デバイスが、新たに定義したアクセス リストのハッシュ値を自動的に生成します。 |
| ステップ 5 | **end**<br>例：<br>`Device(config)# end` | （任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |
| ステップ 6 | **show ip access-list** *access-list-number*<br>例：<br>`Device# show ip access-list 102` | （任意）IP アクセス リストの内容を表示します。<br>• 出力を見直して、アクセス リストにルータが生成したハッシュ値が含まれることを確認します。 |

**例**

次に、デバイスで生成されたハッシュ値を使用したアクセス リストに **show ip access-list** コマンドを使用した際の出力例を示します。

```
Device# show ip access-list
102
Extended IP access list 102
10 permit tcp host 10.1.1.1 host 10.1.1.2 log (hash = 0x7F9CF6B9)
```

# ACL Syslog 相関タグ値の変更

ユーザ定義の Cookie の値を変更したり、ユーザ定義の Cookie とデバイスで生成したハッシュ値を置き換えたりするには、このタスクを実行します。

この手順は、番号付きアクセス リストの ACL Syslog 相関タグ値を変更する方法について示しています。ただし、番号付きおよび名前付きアクセスリストの両方と、標準および拡張アクセス リストの両方について、ACL Syslog 相関タグ値を変更できます。

## 手順の概要

1. **enable**
2. show access-list
3. **configure terminal**
4. access-list *access-list-number* **permit** *protocol source destination* **log** *word*
5. **end**
6. **show ip access-list** *access-list-number*

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | show access-list<br>例：<br>`Device(config)# show access-list` | （任意）アクセス リストの内容を表示します。 |
| ステップ 3 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 4 | access-list *access-list-number* **permit** *protocol source destination* **log** *word*<br>例：<br>`Device(config)# access-list 101 permit tcp host 10.1.1.1 host 10.1.1.2 log NewUDV`<br>例：<br>`OR`<br>例：<br>例：<br>`Device(config)# access-list 101 permit tcp any any log replacehash` | Cookie を修正したり、ハッシュ値を Cookie に変更したりします。<br>• アクセス リスト コンフィギュレーション コマンド全体を入力し、前のタグ値を新しいタグ値で置き換える必要があります。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 5 | **end**<br>例：<br>`Device(config)# end` | （任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |
| ステップ 6 | **show ip access-list** *access-list-number*<br>例：<br>`Device# show ip access-list 101` | （任意）IP アクセス リストの内容を表示します。<br>• 変更を確認するために出力結果を見直します。 |

## トラブルシューティングのヒント

アクセス リストのデバッグ情報を表示するには、**debug ip access-list hash-generation** コマンドを使用します。**debug** コマンドの出力例を次に示します。

```
Device# debug ip access-list hash-generation
 Syslog hash code generation debugging is on
Device# show debug
IP ACL:
 Syslog hash code generation debugging is on
Device# no debug ip access-list hash-generation

 Syslog hash code generation debugging is off
Device# show debug
Device#
```

# ACL Syslog 相関の設定例

## 例：ユーザ定義 Cookie を使用した ACL Syslog 相関の設定

次に、ユーザ定義 Cookie を使用して、デバイス上で ACL Syslog 相関機能を設定する方法について説明します。

```
Device#
Device# debug ip access-list hash-generation
Syslog MD5 hash code generation debugging is on
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# access-list 33 permit 10.10.10.6 log cook_33_std
Device(config)# do show ip access 33
Standard IP access list 33
10 permit 10.10.10.6 log (tag = cook_33_std)
Device(config)# end
```

# 例：ハッシュ値を使用した ACL Syslog 相関の設定

次の例では、デバイスで生成されたハッシュ値を使用して、デバイス上で ACL Syslog 相関機能を設定する方法について説明します。

```
Device# debug ip access-list hash-generation
Syslog MD5 hash code generation debugging is on
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# access-list 33 permit 10.10.10.7 log
Device(config)#
*Nov 7 13:51:23.615: %IPACL-HASHGEN: Hash Input: 33 standard permit 10.10.10.7
Hash Output: 0xCE87F535
Device(config)#
do show ip access 33

Standard IP access list 33
    10 permit 10.10.10.6 log (tag = cook_33_std)
    20 permit 10.10.10.7 log (hash = 0xCE87F535)
```

# 例：ACL Syslog 相関タグ値の変更

次に、既存のアクセス リストのユーザ定義 Cookie と新しい Cookie 値を交換する方法と、デバイス生成ハッシュ値とユーザ定義 Cookie 値を交換する方法について示します。

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# do show ip access-list 101
Extended IP access list 101
    10 permit tcp host 10.1.1.1 host 10.1.1.2 log (tag = MyCookie)
    20 permit tcp any any log (hash = 0x75F078B9)
Device(config)# access-list 101 permit tcp host 10.1.1.1 host 10.1.1.2 log NewUDV
Device(config)# do show access-list
Extended IP access list 101
    10 permit tcp host 10.1.1.1 host 10.1.1.2 log (tag = NewUDV)
    20 permit tcp any any log (hash = 0x75F078B9)
Device(config)# access-list 101 permit tcp any any log replacehash
Device(config)# do show access-list
Extended IP access list 101
    10 permit tcp host 10.1.1.1 host 10.1.1.2 log (tag = NewUDV)
    20 permit tcp any any log (tag = replacehash)
```

# IPv6 IOS ファイアウォールの追加情報

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | *Cisco IOS Master Command List, All Releases* |

| 関連項目 | マニュアル タイトル |
|---|---|
| セキュリティ コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |
| IPv6 コマンド | 『Cisco IOS IPv6 Command Reference』 |
| IPv6 アドレッシングと接続 | 『*IPv6 Configuration Guide*』 |
| Cisco IOS IPv6 機能 | 『*Cisco IOS IPv6 Feature Mapping*』 |

### 標準規格および RFC

| 標準/RFC | タイトル |
|---|---|
| IPv6 に関する RFC | *IPv6 RFCs* |

### テクニカル サポート

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

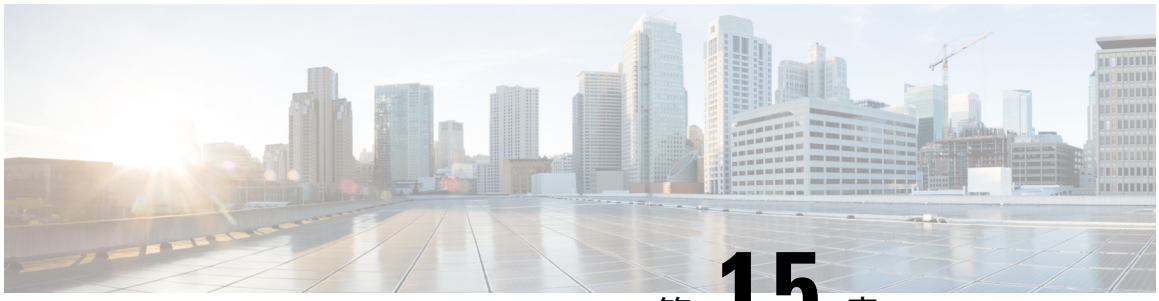
# ACL Syslog 相関に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

表 *14 : ACL Syslog* 相関に関する機能情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| ACL Syslog 相関 | Cisco IOS XE リリース 3.6S | ACL Syslog 相関機能は、ACE Syslog エントリにタグ（ユーザ定義の Cookie またはデバイスが生成した MD5 ハッシュ値）を追加します。このタグは Syslog エントリを生成した ACL 内で ACE を一意に特定します。 |



第 15 章

# IPv6 アクセス コントロール リスト

アクセス リストによって、デバイス インターフェイスでブロックされるトラフィックおよび転送されるトラフィックが決定され、送信元アドレスと宛先アドレスに基づくトラフィックのフィルタリング、および特定のインターフェイスへの着信および発信トラフィックのフィルタリングを行うことができます。標準の IPv6 ACL 機能が拡張されて、IPv6 オプション ヘッダー、および任意でより細かい制御を行うための上位層プロトコル タイプ情報に基づくトラフィック フィルタリングがサポートされています。標準の IPv6 ACL 機能が拡張されて、IPv6 オプション ヘッダー、および任意でより細かい制御を行うための上位層プロトコル タイプ情報に基づくトラフィック フィルタリングがサポートされています。

このモジュールは、仮想端末回線へのアクセスを制御する IPv6 トラフィック フィルタリングの設定方法について説明します。



## RSP3 ポートの関連情報

IPv6 ACL は、RSP3 ではサポートされていません

## 機能情報の確認

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

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

# IPv6 アクセス コントロール リストに関する情報

## IPv6 トラフィック フィルタリングのアクセス コントロール リスト

IPv6 での標準 ACL 機能は、IPv4 での標準 ACL に似ています。アクセス リストによって、デバイスインターフェイスでブロックされるトラフィックおよび転送されるトラフィックが決定され、送信元アドレスと宛先アドレスに基づいて、特定のインターフェイスへの着信と発信をフィルタリングできます。各アクセス リストの末尾には、暗黙的な deny 文があります。IPv6 ACL を定義し、拒否条件と許可条件を設定するには、グローバル コンフィギュレーション モードで **deny** キーワードと **permit** キーワードを指定して **ipv6 access-list** コマンドを使用します。

IPv6 で拡張された ACL では標準 IPv6 ACL 機能を強化して、IPv6 オプション ヘッダー、および任意でより細かい制御を行うための上位層プロトコル タイプ情報に基づくトラフィック フィルタリングがサポートされています（IPv4 における拡張 ACL に類似した機能です）。

## IPv6 パケット インスペクション

ヘッダー フィールド（トラフィック クラス、フローラベル、ペイロード長、次ヘッダー、ホップ リミット、および送信元 IP アドレスや宛先 IP アドレス）は、IPv6 インスペクション用に使用されます。IPv6 ヘッダー フィールドの詳細および説明については、RFC 2474 を参照してください。

## IPv6 でのアクセス クラス フィルタリング

IPv6 ACL に基づく、デバイスとの間の着信接続と発信接続のフィルタリングは、ライン コンフィギュレーション モードで **ipv6 access-class** コマンドを使用して実行します。**ipv6 access-class** コマンドは、IPv6 ACL が名前で定義される点を除き、**access-class** コマンドに似ています。IPv6 ACL が着信トラフィックに適用される場合、ACL 内の送信元アドレスは、着信接続の送信元アドレスと照合され、ACL 内の宛先アドレスは、インターフェイス上のローカル デバイス アドレスと照合されます。IPv6 ACL が発信トラフィックに適用される場合、ACL 内の送信元アドレスは、インターフェイス上のローカル デバイス アドレスと照合され、ACL 内の宛先アドレスは、発信接続の送信元アドレスと照合されます。ユーザが任意の接続を試行できるように、すべての仮想端末回線で同じ制限を設定することを推奨します。

# IPv6 アクセス コントロール リストの設定方法

## IPv6 トラフィック フィルタリングの設定

### トラフィック フィルタリング用の IPv6 ACL の作成および設定

（注） Cisco ASR 1000 プラットフォームの IPv6 ACL には、暗黙の許可ルールは含まれません。IPv6 ネイバー探索プロセスでは、IPv6 ネットワーク層サービスを使用します。したがって、IPv6 ネイバー探索をイネーブルにするには、IPv6 ネイバー探索パケットのインターフェイス上での送受信が許可されるように IPv6 ACL を追加する必要があります。IPv4 では、IPv6 ネイバー探索プロセスと同等のアドレス解決プロトコル（ARP）は、別のデータリンク層プロトコルを使用します。したがってデフォルトでは、IPv4 ACL により、ARP パケットのインターフェイス上での送受信が暗黙的に許可されます。

#### 手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. 次のいずれかを実行します。
   - **permit protocol** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix / prefix-length* | **any** | **host** *destination-ipv6-address*} [**operator** [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]
   - **deny** *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*] ] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport**

#### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ipv6 access-list** *access-list-name*<br>例：<br>`Device(config)# ipv6 access-list inbound` | IPv6 ACL を定義し、IPv6 アクセス リスト コンフィギュレーション モードを開始します。<br>• *access-listname* 引数は、IPv6 ACL の名前を指定します。IPv6 ACL の名前にスペースまたは引用符を含めることはできません。また、先頭を数字にすることはできません。 |
| ステップ 4 | 次のいずれかを実行します。<br>• **permit protocol** {*source-ipv6-prefix/prefix-length* \| **any** \| **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix / prefix-length* \| **any** \| **host** *destination-ipv6-address*} [**operator** [*port-number*]] [**dest-option-type** [*doh-number* \| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* \| *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]<br>• **deny** *protocol* {*source-ipv6-prefix*/*prefix-length* \| **any** \| **host** *source-ipv6-address*} [*operator port-number*]] {*destination-ipv6-prefix/prefix-length* \| **any** \| **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* \| *doh-type*] ] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* \| *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport**<br>例：<br>`Device(config-ipv6-acl)# permit tcp 2001:DB8:0300:0201::/32 eq telnet any`<br>例：<br>`Device(config-ipv6-acl)# deny tcp host 2001:DB8:1::1 any log-input` | IPv6 ACL の許可条件または拒否条件を指定します。 |

## インターフェイスへの IPv6 ACL の適用

### 手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ipv6 traffic-filter** *access-list-name* {**in**| **out**}

### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **interface** *type number*<br>例：<br>`Device(config)# interface gigabitethernet 0/0/0` | インターフェイスのタイプおよび番号を指定し、インターフェイス コンフィギュレーション モードを開始します。 |
| ステップ 4 | **ipv6 traffic-filter** *access-list-name* {**in**| **out**}<br>例：<br>`Device(config-if)# ipv6 traffic-filter inbound in` | 指定した IPv6 アクセス リストを、前のステップで指定したインターフェイスに適用します。 |

# vty へのアクセスの制御

## IPv6 ACL の作成によるアクセス クラス フィルタリングの提供

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. 次のいずれかを実行します。
   - **permit protocol** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix / prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label**

*value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*

- **deny** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport**

## 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ipv6 access-list** *access-list-name*<br>例：<br>`Device(config)# ipv6 access-list cisco` | IPv6 ACL を定義し、IPv6 アクセス リスト コンフィギュレーション モードを開始します。 |
| ステップ 4 | 次のいずれかを実行します。<br>• **permit protocol** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix / prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]<br>• **deny** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport** | IPv6 ACL の許可条件または拒否条件を指定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | 例：<br>`Device(config-ipv6-acl)# permit ipv6 host 2001:DB8:0:4::32 any`<br>例：<br>`Device(config-ipv6-acl)# deny ipv6 host 2001:DB8:0:6::6 any` | |

## 仮想端末回線への IPv6 ACL の適用

### 手順の概要

1. **enable**
2. **configure terminal**
3. **line** [**aux**| **console**| **tty**| **vty**] *line-number*[*ending-line-number*]
4. **ipv6 access-class** *ipv6-access-list-name* {**in**| **out**}

### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **line** [**aux**| **console**| **tty**| **vty**] *line-number*[*ending-line-number*]<br>例：<br>`Device(config)# line vty 0 4` | 設定する特定の回線を識別し、ライン コンフィギュレーション モードを開始します。<br>• この例では、**vty** キーワードを使用して、リモート コンソール アクセス用の仮想端末回線を指定します。 |
| ステップ 4 | **ipv6 access-class** *ipv6-access-list-name* {**in**| **out**}<br>例：<br>`Device(config-line)# ipv6 access-class cisco in` | IPv6 ACL に基づいて、デバイスとの間の着信接続と発信接続をフィルタリングします。 |

# IPv6 アクセス コントロール リストの設定例

## 例：IPv6 ACL 設定の確認

次の例では、**show ipv6 access-list** コマンドを使用して、IPv6 ACL が正しく設定されていることを確認します。

```
Device> show ipv6 access-list

IPv6 access list inbound
    permit tcp any any eq bgp (8 matches) sequence 10
    permit tcp any any eq telnet  (15 matches) sequence 20
    permit udp any any  sequence 30

IPv6 access list Virtual-Access2.1#427819008151 (per-user)
    permit tcp host 2001:DB8:1::32 eq bgp host 2001:DB8:2::32 eq 11000 sequence 1
    permit tcp host 2001:DB8:1::32 eq telnet host 2001:DB8:2::32 eq 11001 sequence 2
```

## 例：IPv6 ACL の作成と適用

次に、HTTP アクセスを日中の特定の時間に制限し、許可されていない時間のアクティビティを記録する方法について例を示します。

```
Device# configure terminal
Device(config)# time-range lunchtime
Device(config-time-range)# periodic weekdays 12:00 to 13:00
Device(config-time-range)# exit
Device(config)# ipv6 access-list INBOUND
Device(config-ipv6-acl)# permit tcp any any eq www time-range lunchtime
Device(config-ipv6-acl)# deny tcp any any eq www log-input
Device(config-ipv6-acl)# permit tcp 2001:DB8::/32 any
Device(config-ipv6-acl)# permit udp 2001:DB8::/32 any
Device(config-ipv6-acl)# end
```

## 例：vty へのアクセスの制御

次の例では、仮想端末回線 0 ～ 4 に着信する接続は、acl1 という名前の IPv6 アクセス リストに基づいてフィルタリングされます。

```
ipv6 access-list acl1
 permit ipv6 host 2001:DB8:0:4::2/32 any
!
line vty 0 4
 ipv6 access-class acl1 in
```

# その他の参考資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Command List, All Releases』 |
| IP アクセス リスト コマンド | 『*Cisco IOS Security Command Reference*』 |
| IP アクセス リストの設定 | 『Creating an IP Access List and Applying It to an Interface』 |

### テクニカル サポート

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

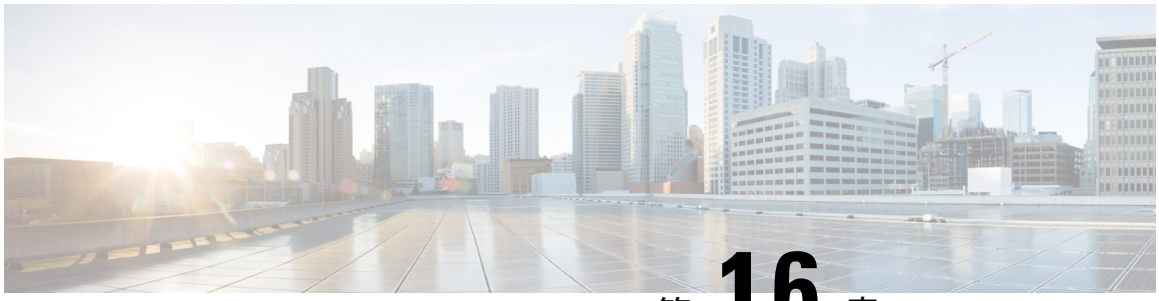
# IPv6 アクセス コントロール リストに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

表 *15 : IPv6* アクセス コントロール リストに関する機能情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| IPv6 サービス：拡張アクセス コントロール リスト | Cisco IOS XE リリース 2.1 | 標準の IPv6 ACL 機能が拡張されて、IPv6 オプション ヘッダー、および任意でより細かい制御を行うための上位層プロトコル タイプ情報に基づくトラフィック フィルタリングがサポートされています。 |



第 16 章

# IPv6 ACL 未決定トランスポートサポート

IPv6 ACL 未決定トランスポートサポート機能は、完全な上位層ヘッダーが存在しない、誤設定されたパケットをドロップするのに役立ちます。



## 機能情報の確認

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

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

## IPv6 ACL 未決定トランスポートサポートの制約事項

- 未決定トランスポート オプションは拒否アクションと IPv6 プロトコルの Cisco Application Control Engine（ACE）でのみサポートされています。
- 未決定トランスポートが nonfirst パケットのフラグメントには適用されません。

# IPv6 ACL 未決定トランスポートサポートに関する情報

## IPv6 ACL 未決定トランスポートサポート

ユーザによる意図しない設定ミスまたはネットワーク上の悪意のある攻撃によって、ネットワーク上のホストに対する運用上の問題が発生する可能性があります。

上位層ヘッダーは、RFC 2460 に説明されているように、IPv6 パケット内の拡張ヘッダー（EH）チェーンの拡張の最後に置かれます。完全な上位層ヘッダーが Ipv6 パケット内にない場合、ルータは、パケットを処理できません。これらのパケットは、誤設定、破損または悪意がある可能性があります。

未決定トランスポートオプションのある IPv6 ACL を使用して、これらのパケットをドロップするよう選択できます。

# IPv6 ACL 未決定トランスポートサポートの設定方法

## IPv6 ACL 未決定トランスポートサポートの設定

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *acl-name*
4. **deny ipv6** {*src-addr* | **any**} {*dest-addr* | **any**} [**undetermined-transport**]
5. **end**

### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ipv6 access-list** *acl-name*<br>例： | IPv6 アクセス リストを設定します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device(config)# ipv6 access-list acl1` | |
| ステップ 4 | **deny ipv6** {*src-addr* \| **any**} {*dest-addr* \| **any**} [**undetermined-transport**]<br>例：<br>`Device(config-ipv6-acl)# deny ipv6 2001:DB8:0300:0201::/32 2001:DB8:1:1::/64 undetermined-transport` | 未決定トランスポートとして、IPv6アクセスリストに対して、拒否状態を設定します。 |
| ステップ 5 | **end**<br>例：<br>`Device(config-ipv6-acl)# end` | 特権 EXEC モードに戻ります。 |

# 例：IPv6 ACL 未決定トランスポートサポートの例

## 例：IPv6 ACL 未決定トランスポートサポートの例

```
Device> enable
Device# configure terminal
Device(config)# ipv6 access-list acl1
Device(config-ipv6-acl)# deny ipv6 2001:DB8:0300:0201::/32 2001:DB8:1:1::/64
undetermined-transport
Device(config-ipv6-acl)# end
```

# IPv6 ACL 未決定トランスポートサポートのその他の参考資料

**関連資料**

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Command List, All Releases』 |
| IPアクセスリストコマンド | 『*Cisco IOS Security Command Reference*』 |
| IP アクセス リストの設定 | 『Creating an IP Access List and Applying It to an Interface』 |

### 標準および RFC

| 標準/RFC | タイトル |
|---|---|
| RFC 2460 | 『Internet Protocol, Version 6 (IPv6) Specification』 |

### テクニカル サポート

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

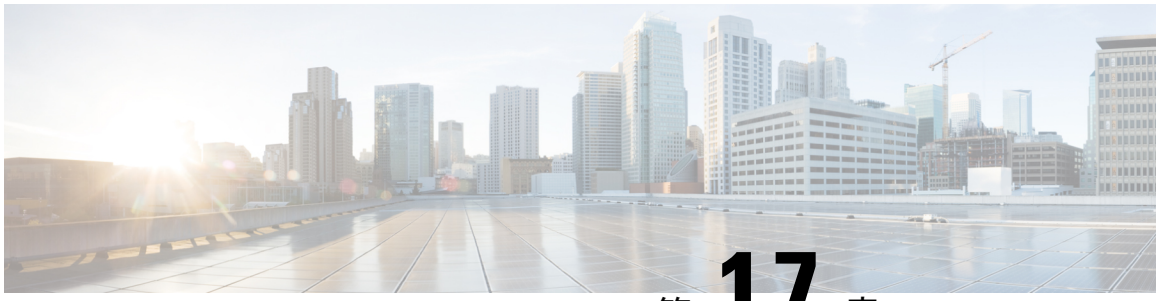
# ACL テンプレートに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *16 : ACL* テンプレートに関する機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| IPv6 ACL 未決定トランスポートサポート | Cisco IOS XE リリース 3.15 | IPv6 ACL 未決定トランスポートサポート機能は、完全な上位層ヘッダーが存在していない誤って設定されたパケットをドロップするのに役立ちます。<br>追加または変更されたコマンドはありません。 |



第 17 章

# テンプレート ACL の設定

ユーザ プロファイルが RADIUS 属性 242 またはベンダー固有属性（VSA ）Cisco AVPairs を使用して設定されると、同様のユーザごとのアクセス コントロール リスト（ACL）は、単一のテンプレート ACL に置き換えられることがあります。つまり、1 つの ACL で多数の類似した ACL を表します。IPv6 テンプレート ACL を使用することで、ACL をサポートするために必要なメモリおよび Ternary Content Addressable Memory（TCAM）リソースを最小限に抑えながら、1 ユーザあたりの ACL の合計数を増やすことができます。

各サブスクライバが独自の ACL を所有するネットワークでは、ユーザの IP アドレスを除いて、ACL をユーザごとに同じとするのが普通です。テンプレート ACL 機能は、システム リソースを節約する 1 つの ACL に多くの一般的なアクセス コントロール要素（ACE ）で ACL をグループ化します。



## 機能情報の確認

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

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

# テンプレート ACL の前提条件

- Cisco ASR 1000 シリーズ ルータ
- Cisco IOS XE リリース 2.4 以降のリリース

# テンプレート ACL の制約事項

テンプレート ACL は、RADIUS 属性 242 または VSA Cisco-AVPairs（ip:inacl/outacl）を通じて設定されたユーザごとの ACL に対してのみ有効になります。その他のタイプの ACL は、テンプレート ACL 機能によって処理されません。

テンプレート ACL 機能は、IPv4 ACL でのみ使用できます。

テンプレート ACL 機能は、ユーザごとの ACL の次のタイプには利用はできません。

- 時間ベース ACL
- ダイナミック ACL
- 評価 ACL
- 再帰 ACL
- ISG IP セッションで設定された ACL
- IPv6 ACL

### テンプレート ACL 機能の無効化

テンプレート ACL 機能を無効にすると、システムは、すべての既存のテンプレート ACL インスタンスを ACL と置き換えます。システムに必要な数の ACL を設定するための十分なリソース（具体的には、TCAM リソース）がない場合、システムは、エラー メッセージを生成し、テンプレート ACL 機能を無効にする要求は失敗します。

# テンプレート ACL の設定に関する情報

## テンプレート ACL 機能設計

サービス プロバイダーが、AAA サーバを使用して、RADIUS 属性 242 または Cisco VSA AVPairs を使用する、権限のあるセッションに対する ACL を設定する場合、セッション数は、システムで許容される最大の ACL 数を簡単に上回ります。

各サブスクライバが ACL を有するネットワークでは、ユーザの IP アドレスを除いて、ACL が各ユーザに対して同じになることは普通です。テンプレート ACL は、システム リソースを高

速で編集し、多くの共通 ACE を持つ ACL を節約する 1 つの ACL にグループ化することで、この問題を軽減します。

テンプレート ACL 機能は、デフォルトで有効になっており、RADIUS 属性 242 または Cisco VSA AVPairs VSA を使用した ACL 設定は、テンプレート ステータスの対象となります。

テンプレート ACL 機能を有効にすると、システムは、すべての設定済みセッション単位の ACL をスキャンおよび評価して、必要なテンプレート ACL を作成します。

### テンプレート ACL の無効化

テンプレート ACL 機能を無効にすると、システムは、すべての既存のテンプレート ACL インスタンスを ACL と置き換えます。システムに必要な数の ACL を設定するための十分なリソース（特に TCAM リソース）がない場合、システムは、エラーメッセージを生成し、テンプレート ACL 機能を無効にする要求が失敗します。

そのため、テンプレート ACL 機能を無効にする前に、**show access-list template summary** コマンドを使用して、システム内のテンプレート ACL の数を表示し、この数がシステムの制限を超えているかを確認します。

テンプレート ACL 機能を無効にすると、新しい ACL は、テンプレーティングの対象にはなりません。

## 複数の ACL

テンプレート ACL 機能を有効にすると、システムは、2 ユーザごとの ACL が類似している場合を特定し、2 つのユーザごとの ACL を 1 つのテンプレート ACL に統合します。

たとえば、次の例は、2 人の個別のユーザに対する 2 つの ACL を示します。

```
ip access-list extended Virtual-Access1.1#1 (PeerIP: 10.1.1.1)
permit igmp any host 10.1.1.1
permit icmp host 10.1.1.1 any
deny ip host 10.31.66.36 host 10.1.1.1
deny tcp host 10.1.1.1 host 10.31.66.36
permit udp any host 10.1.1.1
permit udp host 10.1.1.1 any
permit udp any host 192.168.2.1
permit udp any host 192.168.222.1
permit icmp host 10.55.15.4 host 192.168.2.1
permit udp 10.22.11.0 0.0.0.255 host 192.168.211.2
permit tcp any host 192.168.222.1
permit ip host 10.55.15.4 host 192.168.2.1
permit tcp 10.22.11.0 0.0.0.255 host 192.168.211.2
ip access-list extended Virtual-Access1.1#2 (PeerIP: 10.13.11.2)
permit igmp any host 10.13.11.2
permit icmp host 10.13.11.2 any
deny ip host 10.31.66.36 host 10.13.11.2
deny tcp host 10.13.11.2 host 10.31.66.36
permit udp any host 10.13.11.2
permit udp host 10.13.11.2 any
permit udp any host 192.168.2.1
permit udp any host 192.168.222.1
permit icmp host 10.55.15.4 host 192.168.2.1
permit udp 10.22.11.0 0.0.0.255 host 192.168.211.2
permit tcp any host 192.168.222.1
```

```
permit ip host 10.55.15.4 host 192.168.2.1
permit tcp 10.22.11.0 0.0.0.255 host 192.168.211.2
```

テンプレート ACL 機能を有効にすると、システムは、これら 2 つの ACL が類似していることを認識し、次のように、テンプレート ACL を作成します。

```
ip access-list extended Template_1
permit igmp any host <PeerIP>
permit icmp host <PeerIP> any
deny ip host 10.31.66.36 host <PeerIP>
deny tcp host <PeerIP> 10.31.66.36
permit udp any host <PeerIP>
permit udp host <PeerIP> any
permit udp any host 192.168.2.1
permit udp any host 192.168.222.1
permit icmp host 10.55.15.4 host 192.168.2.1
permit udp 10.22.11.0 0.0.0.255 host 192.168.211.2
permit tcp any host 192.168.222.1
permit ip host 10.55.15.4 host 192.168.2.1
permit tcp 10.22.11.0 0.0.0.255 host 192.168.211.2
```

この例では、ピアの IP アドレスは次のように関連付けられています。

- Virtual-Access1.1#1 10.1.1.1
- Virtual-Access1.1#2 10.13.11.2

2 つの ACL は、1 つのテンプレート ACL に統合され、次のように参照されます。

Template_1(10.1.1.1) への Virtual-Access1.1#1 マップ

Template_1(10.13.11.2) への Virtual-Access1.1#2 マップ

# VSA Cisco-AVPairs

テンプレート ACL 処理は、Cisco-AVPairs を使用して設定される ACL に対して発生します。ACL 番号を使用して定義される AVPairs のみが、テンプレーティングプロセスの対象になります。

テンプレーティングの対象となるために、入力 ACL のための AVPairs は、次の形式に従う必要があります。

ip:inacl#number={standard-access-control-list | extended-access-control-list}

例：ip:inacl#10=deny ip any 10.13.16.0 0.0.0.255

テンプレーティングの対象になるためには、出力 ACL のための AVPairs は、次の形式に従う必要があります:

ip:outacl#number={standard-access-control-list | extended-access-control-list}

例：ip:outacl#200=permit ip any any

Cisco-AVPairs の詳細については、『Cisco IOS ISG RADIUS CoA インターフェイス ガイド』の「Cisco ベンダー固有 AVPair Attributes」のセクションを参照してください。

# RADIUS 属性 242

RADIUS 属性 242 を使用して設定される ACL に対して、テンプレート ACL 処理が発生します。属性 242 は、IP データ フィルタに対して、次の形式があります。

Ascend-Data-Filter = “ip <dir> <action> [dstip <dest_ipaddr\subnet_mask>] [srcp <src_ipaddr\subnet_mask>] [<proto> [dstport <cmp> <value>] [srcport <cmp> <value>] [<est>]]”

次の表で、IP データ フィルタの属性 242 エントリ内の要素について説明します。

表 *17 : IP* データ フィルタ構文要素

| 要素 | 説明 |
|---|---|
| **ip** | IP アドレスを指定します。 |
| **<dir>** | フィルタの方向を指定します。有効値は、**in**（ルータに着信するパケットのフィルタリング）または、**out**（ルータから発信するパケットのフィルタリング）です。 |
| **<action>** | ルータがフィルタに一致したパケットに取るべきアクションを指定します。有効な値は **forward** または **drop** です。 |
| **dstip <dest_ipaddr\subnet_mask>** | 宛先 IP アドレス フィルタリングを有効にします。宛先アドレスが **<dest_ipaddr>** の値に一致するパケットに適用されます。アドレスのサブネット マスクの部分が存在する場合、ルータはマスクされたビットのみを比較します。0.0.0.0 に **<dest_ipaddr>** を設定するか、またはこのキーワードがなければ、フィルタは、すべての IP パケットに一致します。 |
| **srcp<src_ipaddr\subnet_mask>** | 送信元 IP アドレス フィルタリングを有効にします。送信元アドレスが **<src_ipaddr>** の値に一致するパケットに適用されます。アドレスのサブネット マスクの部分が存在する場合、ルータはマスクされたビットのみを比較します。0.0.0.0 に **<src_ipaddr>** を設定するか、またはこのキーワードがなければ、フィルタは、すべての IP パケットに一致します。 |
| **<proto>** | 名前または番号として指定するプロトコルを指定します。プロトコル フィールドがこの値に一致するパケットに適用されます。使用できる名前と番号は **icmp**（**1**）、**tcp**（**6**）、**udp**（**17**）、および **ospf**（**89**）です。この値をゼロ（0）に設定すると、フィルタは、一切のプロトコルに一致します。 |
| **dstport <cmp> <value>** | 宛先ポートフィルタリングを有効にします。このキーワードは、**<proto>** が **tcp**（**6**）または **udp**（**17**）に設定されている場合に限り有効です。宛先ポートを指定しないと、フィルタは、一切のポートと一致します。<br><br>**<cmp>** は、指定された **<value>** と実際の宛先ポートとを比較する方法を定義します。この値として **<**、**=**、**>**、または **!** を使用できます。<br><br>**<value>** 名前も番号も使用可能です。使用できる名前と番号は **ftp-data (20)**、**ftp (21)**、**telnet (23)**、**nameserver (42)**、**domain (53)**、**tftp (69)**、**gopher (70)**、**finger (79)**、**www (80)**、**kerberos (88)**、**hostname (101)**、**nntp (119)**、**ntp (123)**、**exec (512)**、**login (513)**、**cmd (514)**、および **talk (517)** です。 |

| 要素 | 説明 |
|---|---|
| **srcport <cmp> <value>** | 送信元ポートフィルタリングを有効にします。このキーワードは、**<proto>** が **tcp**（**6**）または **udp**（**17**）に設定されている場合に限り有効です。送信元ポートを指定しないと、フィルタは、一切のポートと一致します。<br>**<cmp>** は、指定された **<value>** と実際の宛先ポートとを比較する方法を定義します。この値として **<**、**=**、**>**、または **!** を使用できます。<br>**<value>** 名前も番号も使用可能です。使用できる名前と番号は **ftp-data**（**20**）、 **ftp**（**21**）、**telnet**（**23**）、**nameserver**（**42**）、**domain**（**53**）、**tftp**（**69**）、**gopher**（**70**）、**finger**（**79**）、**www**（**80**）、**kerberos**（**88**）、**hostname**（**101**）、**nntp**（**119**）、**ntp**（**123**）、**exec**（**512**）、**login**（**513**）、**cmd**（**514**）、および **talk**（**517**）です。 |
| **<est>** | 1 に設定すると、TCP セッションがすでに確立されている場合にのみ、パケット フィルタと一致していると指定します。この引数は、**<proto>** が **tcp (6)** に設定されている場合に限り有効です。 |

「RADIUS 属性 242 IP データ フィルタ エントリ」は、4 つの属性 242 IP データフィルタエントリを示します。

**RADIUS 属性 242 IP データフィルタエントリ**

```
Ascend-Data-Filter=”ip in drop”
Ascend-Data-Filter=”ip out forward tcp”
Ascend-Data-Filter=”ip out forward tcp dstip 10.0.200.3/16 srcip 10.0.200.25/16
dstport!=telnet”
Ascend-Data-Filter=”ip out forward tcp dstip 10.0.200.3/16 srcip 10.0.200.25/16 icmp”
```

# テンプレート ACL の設定方法

ACL が RADIUS 属性 242 または VSA Cisco-AVPairs を使用して設定されると、ACL は、デフォルトでは有効になりません。

# テンプレート ACL の最大サイズの設定

デフォルトでは、テンプレートの ACL ステータスは 100 台以下のルールの ACL に限定されます。ただし、この制限を低い値に設定できます。テンプレート ACL とみなされるため、既存の ACL は、以下のようなルールの最大数を設定するには、このセクションの手順を実行してください:

## 手順の概要

1. **enable**
2. **configure terminal**
3. **access-list template** *number*
4. **exit**

5. **show access-list template summary**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Router> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Router# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **access-list template** *number*<br>例：<br>`Router(config)# access-list template 50` | テンプレート ACL の処理をイネーブルにします。<br>指定された数のルール（またはより少ないルール）の ACL だけがテンプレートのステータスの対象となります。 |
| ステップ 4 | **exit**<br>例：<br>`Router(config)# exit` | グローバル コンフィギュレーション モードを終了します。 |
| ステップ 5 | **show access-list template summary**<br>例：<br>`Router# show access-list template summary` | （任意）ACLテンプレートに関する要約情報が表示されます。 |

## トラブルシューティングのヒント

次のコマンドを使用すると、テンプレート ACL をトラブルシューティングできます。

- **show access-list template**
- show platform hardware qfp active classification class-group-manager class-group client acl all
- **show platform hardware qfp active feature acl** {**control** | **node** *acl-node-id*}
- show platform software access-list

# テンプレート ACL の設定例

## テンプレート ACL の最大サイズの例

次の例では、テンプレートのステータスを 50 と対象するために ACL が含むことができるルールの最大数の設定方法を示しています。ルールの数は同じか、または 50 よりも少ない ACL のみがテンプレート ステータスの対象となります。

```
Router> enable

Router# configure terminal

Router(config)# access-list template 50
Router(config)# exit
```

## ACL のテンプレートの概要情報を示す例

以下の例は、システム内の全 ACL 用の要約情報を表示する方法を示しています。このコマンドからの出力には、次の情報が含まれています。

- テンプレート ACL ごとのルールの最大数
- 発見されたアクティブなテンプレート数
- これらのテンプレートによって置き換えられた ACL 数
- レッドブラックツリー内の要素数

```
Router# show access-list template summary
Maximum rules per template ACL = 100
Templates active = 9
Number of ACLs those templates represent = 14769
Number of tree elements = 13
```

### レッドブラックツリー要素

ツリー要素の数は、レッドブラックツリー内の要素の数です。各テンプレートは、レッドブラックツリー内の一意のエントリを 1 つ含みます。システムは、ピア IP アドレスをマスクする各 ACL 上の巡回冗長検査（CRC）を計算し、レッドブラックツリーに CRC を送信します。次に例を示します。

システムに 9 つのテンプレート（14769 個の ACL を表す）、および 13 のツリーの要素があります。レッドブラックツリー内で各テンプレートに一意のエントリが 1 つしかない場合、その他 4 つのツリー要素は、システムには、テンプレート化されていない 4 個のユーザあたりの ACL が含まれているということです。

# ACL のテンプレート ツリー情報を示す例

以下の例は、システム内の全 ACL 用のレッドブラックツリー情報を表示する方法を示しています。

このコマンドからの出力には、次の情報が含まれています。

- レッドブラックツリー上の ACL 名
- 元の CRC32 値
- ACL のユーザ数
- 計算された CRC32 値

```
Router# show access-list template tree
ACL name       OrigCRC    Count  CalcCRC
4Temp_1073741891108      59DAB725   98  59DAB725
```

# その他の参考資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Command List, All Releases』 |
| IP アクセス リスト コマンド | 『*Cisco IOS Security Command Reference*』 |
| IP アクセス リストの設定 | 『Creating an IP Access List and Applying It to an Interface』 |

### テクニカル サポート

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

# ACL テンプレートに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *18 : ACL* テンプレートに関する機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| ACL テンプレート | 12.2(28) SB 12.2(31) SB2 Cisco IOS XE リリース 2.4 | 12.2(28)SB では、この機能が Cisco 10000 シリーズ ルータで追加されました。<br>12.2(31)SB2 では、PRE3 のサポートが追加されました。<br>この機能は、Cisco IOS XE Release 2.4 で、Cisco ASR 1000 シリーズ ルータに実装されました。<br>次のコマンドが導入または変更されました。**access-list template, show access-list template** |

第 18 章

# IPv6 テンプレート ACL

ベンダー固有属性（VSA）の Cisco AV ペアを使用してユーザ プロファイルが設定されている場合は、類似した 1 ユーザ単位の IPv6 ACL を 1 つのテンプレート ACL で置き換えることができます。つまり、1 つの ACL で多数の類似した ACL を表します。IPv6 テンプレート ACL を使用することで、ACL をサポートするために必要なメモリおよび Ternary Content Addressable Memory（TCAM）リソースを最小限に抑えながら、1 ユーザあたりの ACL の合計数を増やすことができます。

IPv6 テンプレート ACL 機能では、次の ACL フィールドを使用してテンプレートを作成します。

- IPv6 の送信元アドレスおよび宛先アドレス
- すべての関連ポート（0 ～ 65535）を含む TCP および UDP
- ICMP ネイバー探索アドバタイズメントおよび要請
- 指定した DSCP 値による IPv6 DSCP

この機能により、ACL の名前はたとえば次のように動的に生成されます。

- 6Temp_#152875854573 - 親 ACL のテンプレートとして動的に生成されたテンプレート名の例
- Virtual-Access2.32135#152875854573 - 子 ACL またはテンプレートの一部とされていない ACL の例。

# 機能情報の確認

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

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

# IPv6 ACL に関する情報：テンプレート ACL

## IPv6 テンプレート ACL

ベンダー固有属性（VSA）の Cisco AV ペアを使用してユーザ プロファイルが設定されている場合は、類似した 1 ユーザ単位の IPv6 ACL を 1 つのテンプレート ACL で置き換えることができます。つまり、1 つの ACL で多数の類似した ACL を表します。IPv6 テンプレート ACL を使用することで、ACL をサポートするために必要なメモリおよび Ternary Content Addressable Memory（TCAM）リソースを最小限に抑えながら、1 ユーザあたりの ACL の合計数を増やすことができます。

IPv6 テンプレート ACL 機能では、次の ACL フィールドを使用してテンプレートを作成します。

- IPv6 の送信元アドレスおよび宛先アドレス
- すべての関連ポート（0 ～ 65535）を含む TCP および UDP
- ICMP ネイバー探索アドバタイズメントおよび要請
- 指定した DSCP 値による IPv6 DSCP

この機能により、ACL の名前はたとえば次のように動的に生成されます。

- 6Temp_#152875854573 - 親 ACL のテンプレートとして動的に生成されたテンプレート名の例
- Virtual-Access2.32135#152875854573 - 子 ACL またはテンプレートの一部とされていない ACL の例。

# IPv6 ACL を有効にする方法：テンプレート ACL

## IPv6 テンプレートの処理の有効化

手順の概要

1. **enable**
2. **configure terminal**
3. **access-list template** [*number-of-rules*]
4. **exit**
5. **show access-list template** {**summary** | *aclname* | **exceed** *number* | **tree**}

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Router> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Router# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **access-list template** [*number-of-rules*]<br>例：<br>`Router(config)# access-list template 50` | テンプレート ACL の処理をイネーブルにします。<br>• このタスクの例では、50以下のルールを設定した ACL がテンプレート ACL ステータスとして見なされるように指定しています。<br>• *number-of-rules* 引数のデフォルトは 100 です。 |
| ステップ 4 | **exit**<br>例：<br>`Router(config)# exit` | グローバル コンフィギュレーション モードを終了して、ルータを特権 EXEC モードにします。 |
| ステップ 5 | **show access-list template** {**summary** \| *aclname* \| **exceed** *number* \| **tree**}<br>例：<br>`Router# show access-list template summary` | ACL テンプレートの情報を表示します。 |

# IPv6 ACL の設定例：テンプレート ACL

## 例：IPv6 テンプレート ACL の処理

この例では、内容は同じでも、名前が ACL1 と ACL2 で異なります。

```
ipv6 access-list extended ACL1 (PeerIP: 2001:1::1/64)
permit igmp any                  2003:1::1/64
permit icmp 2002:5::B/64         any
permit udp  any                  host 2004:1::5
permit udp  any                  host 2002:2BC::a
permit icmp host 2001:BC::7      host 2003:3::7
ipv6 access-list extended ACL2 (PeerIP: 2007:2::7/64)
permit igmp any                  2003:1::1/64
permit icmp 2002:5::B/64         any
permit udp  any                  host 2004:1::5
permit udp  any                  host 2002:2BC::a
permit icmp host 2001:BC::7      host 2003:3::7
```

これらの ACL のテンプレートは次のとおりです。

```
ipv6 access-list extended Template_1
permit igmp any                  2003:1::1/64
permit icmp 2002:5::B/64         any
permit udp  any                  host 2004:1::5
permit udp  any                  host 2002:2BC::a
permit icmp host 2001:BC::7      host 2003:3::7
```

# その他の参考資料

**関連資料**

| 関連項目 | マニュアル タイトル |
|---|---|
| IPv6 アドレッシングと接続 | 『*IPv6 Configuration Guide*』 |
| Cisco IOS コマンド | 『*Cisco IOS Master Commands List, All Releases*』 |
| IPv6 コマンド | 『*Cisco IOS IPv6 Command Reference*』 |
| Cisco IOS IPv6 機能 | 『*Cisco IOS IPv6 Feature Mapping*』 |

### 標準規格および RFC

| 標準/RFC | タイトル |
|---|---|
| IPv6 に関する RFC | *IPv6 RFCs* |

### MIB

| MIB | MIB のリンク |
|---|---|
| | 選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。<br>http://www.cisco.com/go/mibs |

### シスコのテクニカル サポート

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

# IPv6 ACL - テンプレート ACL に関する機能情報

表 *19 : IPv6 ACL* - テンプレート *ACL* に関する機能情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| IPv6 ACL - テンプレート ACL | Cisco IOS XE リリース 3.2S | この機能により、類似のユーザごとの IPv6 ACL を単一のテンプレート ACL に置き換えることができます。<br>次のコマンドが導入または変更されました。**access-list template**、**show access-list template** |

第 19 章

# IPv4 ACL チェーニング サポート

マルチアクセス コントロール リストとも呼ばれる ACL チェーニングにより、アクセス コントロール リスト（ACL）を分割することができます。このモジュールでは、IPv4 ACL チェーニング サポートによって ACL を共通 ACL とユーザ専用 ACL に明示的に分割する方法、および両 ACL をデバイスでのトラフィック フィルタリングのためにバインドする方法について説明します。この方法では、Ternary Content Addressable Memory（TCAM）内の共通 ACL は複数のターゲットにより共有され、これによりリソース使用量が削減されます。



## 機能情報の確認

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

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

## IPv4 ACL チェーニング サポートの制限事項

- 単一のアクセス コントロール リスト（ACL）を、同じ方向の同じターゲットに対する共通、標準の両 ACL に使用することはできません。

- ACL チェーニングはセキュリティ ACL にのみ適用されます。サービス品質（QoS）、ファイアウォール サービス モジュール（FW）、ポリシーベース ルーティング（PBR）などのフィーチャ ポリシーではサポートされません。
- 共通 ACL ではターゲットごとの統計情報はサポートされません。

# IPv4 ACL チェーニング サポートに関する情報

## ACL チェーニングの概要

パケット フィルタリング プロセスは、1 つのインターフェイスの 1 つの方向および 1 つのプロトコルごとに適用される単一のアクセス コントロール リスト（ACL）のみをサポートします。そのため、多数のインターフェイスに共通 ACL エントリが必要な場合、管理性と拡張性の問題が生じます。そのようなインターフェイスにはすべて重複アクセス コントロール エントリ（ACE）が設定されており、共通 ACE の変更はすべての ACL で行われる必要があります。

インターネット サービス プロバイダー（ISP）のエッジ ボックスの典型的な ACL には次の 2 組の ACE が含まれます。

- 共通 ISP 専用 ACE
- 顧客/インターフェイス専用 ACE

これらのアドレスブロックは、ISP の保護されたインフラストラクチャ ネットワークへのアクセスを拒否するため、および顧客の送信元アドレスブロックのみを許可することでスプーフィングを防ぐために行われます。この結果、インターフェイスごとに一意の ACL が設定され、ほとんどの ACE がデバイス上のすべての ACL で共通になります。ACL をプロビジョニングし、変更するのは非常に面倒ですが、ACE を変更すれば全ターゲットに影響を及ぼすことができます。

## IPv4 ACL チェーニング サポート

IPv4 ACL チェーニング サポートを使用して、アクセス コントロール リスト（ACL）を共通 ACL と顧客専用 ACL に分割したり、両 ACL を共通セッションにアタッチすることができます。この方法では、共通 ACL を 1 コピーのみ Ternary Content Addressable Memory（TCAM）にアタッチしこれを全ユーザで共有することで、共通 ACE の維持が簡略化されます。

IPv4 ACL チェーニング機能により、次の 2 つの IPv4 ACL を 1 方向ごとに 1 つのインターフェイスでアクティブにできます。

- 共通
- 標準
- 共通と標準

（注） 1 つのインターフェイスで共通と標準の両 ACL を設定している場合、共通 ACL が標準 ACL に優先されます。

# IPv4 ACL チェーニング サポートの設定方法

ACL チェーニングは、**ip traffic filter** コマンドの拡張によりサポートされます。

**ip traffic filter** コマンドは追加式ではありません。このコマンドを使用すると、このコマンドの以前のインスタンスが置き換えられます。

詳細については、『Security Configuration Guide: Access Control Lists Configuration Guide』の「*IPv6 ACL Chaining with a Common ACL*」セクションを参照してください。

## 共通 ACL を受け入れるインターフェイスの設定

このタスクを実行すると、インターフェイス固有の ACL とともに、共通のアクセス コントロール リスト（ACL）を受け入れるようにインターフェイスを設定できます。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*}
4. **ip access-group** {**common** {*common-access-list-name* {*regular-access-list* | **acl**}}{**in** | **out**}}
5. **end**

### 手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **interface** *type number*}<br>例：<br>`Device(config)# interface gigabitethernet 0/0/0` | インターフェイス（この場合、gigabitethernet interface）を設定し、インターフェイス コンフィギュレーション モードを開始します。 |

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 4 | **ip access-group** {**common** {*common-access-list-name* {*regular-access-list* \| **acl**}}{**in** \| **out**}}<br>例：<br>`Device(config)# ipv4 access-group common acl-p acl1 in` | インターフェイス固有の ACL とともに、共通 ACL を受け入れるようにインターフェイスを設定します。 |
| ステップ 5 | **end**<br>例：<br>`Device(config-if)# end` | （任意）コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。 |

# IPv4 ACL チェーニング サポートの設定例

ここでは、共通アクセス コントロール リスト（ACL）の設定例を示します。

## 例：共通 ACL を受け入れるインターフェイスの設定

次に、ACL を明示的に削除しないでインターフェイスで設定したアクセス コントロール リスト（ACL）を交換する方法例を示します。

```
interface gigabitethernet 0/0/0
ipv4 access-group common  C_acl ACL1 in
end
replace interface acl ACL1 by ACL2
interface gigabitethernet 0/0/0
ipv4 access-group common  C_acl ACL2 in
end
```

次に、インターフェイスから共通 ACL を明示的に削除しないと、共通 ACL をインターフェイスで交換できない方法例を示します。

```
interface gigabitethernet 0/0/0
ipv4 access-group common  C_acl1 ACL1 in
end
change the common acl to C_acl2
interface gigabitethernet 0/0/0
no ipv4 access-group common  C_acl1  ACL1 in
end
interface gigabitethernet 0/0/0
ipv4 access-group common  C_acl2  ACL1 in
end
```

（注） 共通 ACL を再設定する際、ライン カードの他のインターフェイスが共通 ACL に取り付けられないことを確認する必要があります。

（注） 共通 ACL とインターフェイス ACL の両方をインターフェイスに取り付け、その一方をインターフェイスで再構成すると、他は自動的に削除されます。

次に、インターフェイス ACL の削除方法を示します。

```
interface gigabitethernet 0/0/0
ipv4 access-group common C_acl1 ACL1 in
end
```

# IPv4 ACL チェーニング サポートの追加参考資料

**関連資料**

| 関連項目 | マニュアル タイトル |
|---|---|
| IPv6 ACL チェーニング サポート | |
| Cisco IOS コマンド | Cisco IOS Master Command List, All Releases |
| セキュリティ コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |

### シスコのテクニカル サポート

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

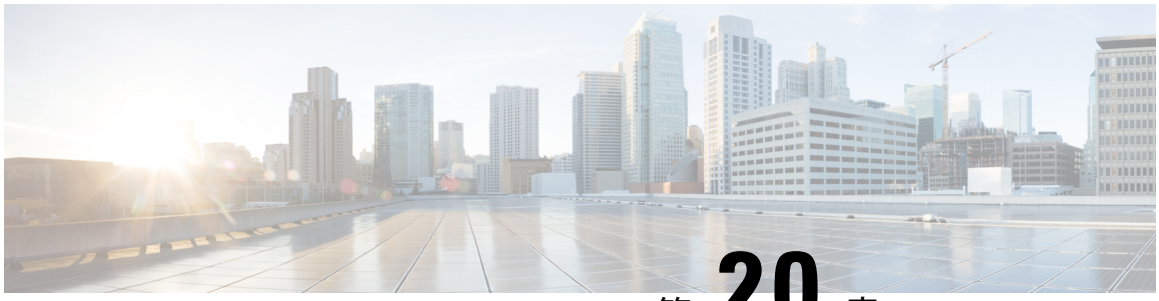
# IPv4 ACL チェーニング サポートに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

表 *20 : IPv4 ACL* チェーニング サポートに関する機能情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| IPv4 ACL チェーニング サポート | Cisco IOS XE リリース 3.11S<br>Cisco IOS XE リリース 3.6E | IPv4 ACL チェーニング サポートは、アクセス コントロール リスト（ACL ）を明示的に共通およびユーザ固有の ACL に分割して、両方の ACL をデバイス上でのトラフィック フィルタリングのためのセッションにバインドする方法について説明します。この方法では、Ternary Content Addressable Memory（TCAM）内の共通 ACL は複数のターゲットにより共有され、これによりリソース使用量が削減されます。<br>次のコマンドが導入または変更されました。**ip access-group command** |



第 20 章

# 共通 ACL による IPv6 ACL チェーニング

マルチアクセス コントロール リストとも呼ばれる ACL チェーニングにより、ACL を分割することができます。このマニュアルでは、IPv6 ACL チェーニング サポートによって ACL を共通 ACL とユーザ専用 ACL に明示的に分割する方法、および両 ACL をデバイスでのトラフィック フィルタリングのためにバインドする方法について説明します。この方法では、Ternary Content Addressable Memory（TCAM）内の共通 ACL は複数のターゲットにより共有され、これによりリソース使用量が削減されます。



## 機能情報の確認

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

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

# 共通 ACL による IPv6 ACL チェーニングに関する情報

## ACL チェーニングの概要

パケット フィルタリング プロセスは、1 つのインターフェイスの 1 つの方向および 1 つのプロトコルごとに適用される単一のアクセス コントロール リスト（ACL）のみをサポートします。そのため、多数のインターフェイスに共通 ACL エントリが必要な場合、管理性と拡張性の問題が生じます。そのようなインターフェイスにはすべて重複アクセス コントロール エントリ（ACE）が設定されており、共通 ACE の変更はすべての ACL で行われる必要があります。

インターネット サービス プロバイダー（ISP）のエッジ ボックスの典型的な ACL には次の 2 組の ACE が含まれます。

- 共通 ISP 専用 ACE
- 顧客/インターフェイス専用 ACE

これらのアドレスブロックは、ISPの保護されたインフラストラクチャ ネットワークへのアクセスを拒否するため、および顧客の送信元アドレスブロックのみを許可することでスプーフィングを防ぐために行われます。この結果、インターフェイスごとに一意の ACL が設定され、ほとんどの ACE がデバイス上のすべての ACL で共通になります。ACL をプロビジョニングし、変更するのは非常に面倒ですが、ACEを変更すれば全ターゲットに影響を及ぼすことができます。

## 共通 ACL による IPv6 ACL チェーニング

IPv6 ACL チェーニングを使用して、トラフィック フィルタを次の ACL とチェーニングできます。

- 共通 ACL
- 専用 ACL
- 共通 ACL と専用 ACL

各アクセス コントロール リスト（ACL）は順に照合されます。たとえば、共通 ACL と専用 ACL の両方を指定している場合、パケットはまず共通 ACL に対して照合され、一致が見つからなければ専用 ACL に対して照合されます。

（注） 任意の IPv6 ACL を共通または専用 ACL としてトラフィック フィルタで設定できます。ただし、同じ ACL を同じトラフィック フィルタで共通と専用の両方として指定することはできません。

# 共通 ACL による IPv6 ACL チェーニングの設定方法

### 始める前に

IPv6 ACL チェーニングは、既存の IPv6 トラフィック フィルタ コマンド **ipv6 traffic-filter** [ **common** *common-acl*] [*specific-acl*] [ **in** | **out**] の拡張機能を使用して、インターフェイス上で設定します。

（注） 次のいずれかを設定できます。

- 共通 ACL のみ。例：**ipv6 traffic-filter common** *common-acl*
- 特定の ACL のみ。例：**ipv6 traffic-filter** *common-acl*
- 両方の ACL。例：**ipv6 traffic-filter common** *common-acl specific-acl*

ipv6 traffic-filter コマンドは追加式ではありません。このコマンドを使用すると、このコマンドの以前のインスタンスが置き換えられます。たとえば、コマンド シーケンス **ipv6 traffic-filter** [ **common** *common-acl*] [*specific-acl*] **in ipv6 traffic-filter** [*specific-acl*] **in** は、共通 ACL とトラフィック フィルタをバインディングし、共通 ACL を削除してから、特定の ACL をバインディングします。

## インターフェイスへの IPv6 ACL の設定

このタスクを実行すると、インターフェイス固有の ACL とともに、共通のアクセス コントロール リスト（ACL）を受け入れるようにインターフェイスを設定できます。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*}
4. **ipv6 traffic filter** {*common-access-list-name* {**in** | **out**}}
5. **end**

### 手順の詳細

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

| | コマンドまたはアクション | 目的 |
|---|---|---|
| | `Device# configure terminal` | |
| ステップ 3 | **interface** *type number*}<br>例：<br>`Device(config)# interface gigabitethernet 0/0/0` | インターフェイスのタイプおよび番号を指定し、インターフェイス コンフィギュレーション モードを開始します。 |
| ステップ 4 | **ipv6 traffic filter** {*common-access-list-name* {**in** \| **out**}}<br>例：<br>`Device(config)# ipv6 traffic-filter outbound out` | 指定した IPv6 アクセス リストを、前のステップで指定したインターフェイスに適用します。 |
| ステップ 5 | **end**<br>例：<br>`Device(config-if)# end` | （任意）コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。 |

# 共通 ACL による IPv6 ACL チェーニングの設定例

特定の順序でなくても、次の組み合わせを設定できます。

- 共通 ACL。例： **ipv6 traffic-filter common** *common-acl* **in**
- 特定の ACL。例： **ipv6 traffic-filter** *specific-acl* **in**
- 両方の ACL。例： **ipv6 traffic-filter common** *common-acl* *specific-acl* **in**

## 例：共通 ACL を受け入れるインターフェイスの設定

次に、ACL を明示的に削除しないでインターフェイスで設定したアクセス コントロール リスト（ACL）を交換する方法例を示します。

```
interface gigabitethernet 0/0/0
ipv6 access-group common  C_acl ACL1 in
end
replace interface acl ACL1 by ACL2
interface gigabitethernet 0/0/0
ipv6 access-group common  C_acl ACL2 in
end
```

次の例では、共通 ACL をインターフェイスから削除する方法を示します。インターフェイスから共通 ACL を明示的に削除しないと、共通 ACL をインターフェイスで交換できません。

```
interface gigabitethernet 0/0/0
ipv6 access-group common  C_acl1 ACL1 in
end
change the common acl to C_acl2
interface gigabitethernet 0/0/0
```

```
no ipv6 access-group common  C_acl1  ACL1 in
end
interface gigabitethernet 0/0/0
ipv6 access-group common  C_acl2  ACL1 in
end
```

（注） 共通 ACL を再設定する際、ライン カードの他のインターフェイスが共通 ACL に取り付けられないことを確認する必要があります。

（注） 共通 ACL とインターフェイス ACL の両方をインターフェイスに取り付け、その一方をインターフェイスで再構成すると、他は自動的に削除されます。

次に、インターフェイス ACL を削除する方法を示します。

```
interface gigabitethernet 0/0/0
ipv6 access-group common C_acl1 ACL1 in
end
```

# 共通 ACL による IPv6 ACL チェーニングの追加情報

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| IPv4 ACL チェーニング サポート | 『Security Configuration Guide: Access Control Lists, Cisco IOS XE Release 3S』 |
| Cisco IOS コマンド | 『Cisco IOS Master Commands List, All Releases』 |
| セキュリティ コマンド | • 『Cisco IOS Security Command Reference: Commands A to C』<br>• 『Cisco IOS Security Command Reference: Commands D to L』<br>• 『Cisco IOS Security Command Reference: Commands M to R』<br>• 『Cisco IOS Security Command Reference: Commands S to Z』 |

### シスコのテクニカル サポート

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

# 共通 ACL による IPv6 ACL チェーニングに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

表 *21* : 共通 *ACL* による *IPv6 ACL* チェーニングに関する機能情報

| 機能名 | リリース | 機能情報 |
|---|---|---|
| 共通 ACL による IPv6 ACL チェーニング | Cisco IOS XE リリース 3.11S<br>Cisco IOS XE リリース 3.6E | ACL チェーニング機能（別名、マルチ ACL）により、IPv6 トラフィック フィルタのアクセス コントロール リスト（ACL ）を明示的にコモンおよびセッション単位の ACL に分割できます。このように、使用される共通のアクセス コントロール エントリ（ACE）は、Ternary Content Addressable Memory（TCAM）内のセッションごとに各 ACL エントリのリソース使用量を減らします。<br>次のコマンドが導入または変更されました。**ip access-group common** |

第 21 章

# ホップ バイ ホップ フィルタリングに対応するための IPv6 ACL の拡張

ホップ バイ ホップ フィルタリングに対応するための IPv6 ACL の拡張機能により、ホップ バイ ホップ拡張ヘッダーを含む可能性がある IPv6 トラフィックを制御することができます。アクセス コントロール リスト（ACL）を設定して、すべてのホップ バイ ホップ トラフィックを拒否するか、またはプロトコルに基づいて選択的にトラフィックを許可することができます。



## 機能情報の確認

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

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

# ホップバイホップフィルタリングに対応するためのIPv6 ACLの拡張に関する情報

## ACLおよびトラフィック転送

IPv6アクセス コントロール リスト（ACL）は、デバイス インターフェイスでブロックされるトラフィックと転送されるトラフィックを決定します。ACL を使用すると、特定のインターフェイスへの着信および発信を、送信元アドレスと宛先アドレスに基づいてフィルタリングできます。**ipv6 access-list** コマンドを使用して IPv6 ACL を定義し、**deny** および **permit** コマンドを使用してその条件を構成します。

ホップ バイ ホップ フィルタリングに対応するための IPv6 ACL の拡張機能は、上位層プロトコル タイプでのトラフィック フィルタリングをサポートするために RFC 2460 を実装します。

# ホップバイホップフィルタリングに対応するためのIPv6 ACLの拡張の設定方法

## ホップ バイ ホップ フィルタリングに対応するための IPv6 ACL の拡張の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. **permit** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address* | **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix*/*prefix-length* | **any** | **host** *destination-ipv6-address* | **auth**} [*operator* [*port-number*]] [**dest-option-type** [*header-number* | *header-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**hbh**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**reflect** *name* [**timeout** *value*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]
5. **deny** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address* | **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix*/*prefix-length* | **any** | **host** *destination-ipv6-address* | **auth**} [*operator* [*port-number*]] [**dest-option-type** [*header-number* | *header-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**hbh**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport**]
6. **end**

手順の詳細

| | コマンドまたはアクション | 目的 |
|---|---|---|
| ステップ 1 | **enable**<br>例：<br>`Device> enable` | 特権 EXEC モードをイネーブルにします。<br>• パスワードを入力します（要求された場合）。 |
| ステップ 2 | **configure terminal**<br>例：<br>`Device# configure terminal` | グローバル コンフィギュレーション モードを開始します。 |
| ステップ 3 | **ipv6 access-list** *access-list-name*<br>例：<br>`Device(config)# ipv6 access-list hbh-acl` | IPv6 ACL を定義し、IPv6 アクセス リスト コンフィギュレーション モードを開始します。 |
| ステップ 4 | **permit** *protocol* {*source-ipv6-prefix*/*prefix-length* \| **any** \| **host** *source-ipv6-address* \| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix*/*prefix-length* \| **any** \| **host** *destination-ipv6-address* \| **auth**} [*operator* [*port-number*]] [**dest-option-type** [*header-number* \| *header-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**hbh**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* \| *mh-type*]] [**reflect** *name* [**timeout** *value*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]<br>例：<br>`Device(config-ipv6-acl)# permit icmp any any dest-option-type` | IPv6 ACL の許可条件を設定します。 |
| ステップ 5 | **deny** *protocol* {*source-ipv6-prefix*/*prefix-length* \| **any** \| **host** *source-ipv6-address* \| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix*/*prefix-length* \| **any** \| **host** *destination-ipv6-address* \| **auth**} [*operator* [*port-number*]] [**dest-option-type** [*header-number* \| *header-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**hbh**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* \| *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport**]<br>例：<br>`Device(config-ipv6-acl)# deny icmp any any dest-option-type` | IPv6 ACL の拒否条件を設定します。 |
| ステップ 6 | **end**<br>例：<br>`Device (config-ipv6-acl)# end` | 特権EXECコンフィギュレーションモードに戻ります。 |

# ホップバイ ホップ フィルタリングに対応するための IPv6 ACL の拡張の設定例

## 例：ホップ バイ ホップ フィルタリングに対応するための IPv6 ACL の拡張

```
Device(config)# ipv6 access-list hbh_acl
Device(config-ipv6-acl)# permit tcp any any hbh
Device(config-ipv6-acl)# permit tcp any any
Device(config-ipv6-acl)# permit udp any any
Device(config-ipv6-acl)# permit udp any any hbh
Device(config-ipv6-acl)# permit hbh any any
Device(config-ipv6-acl)# permit any any
Device(config-ipv6-acl)# hardware statistics
Device(config-ipv6-acl)# exit

! Assign an IP address and add the ACL on the interface.

Device(config)# interface FastEthernet3/1
Device(config-if)# ipv6 address 1001::1/64
Device(config-if)# ipv6 traffic-filter hbh_acl in
Device(config-if)# exit
Device(config)# exit
Device# clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#

! Verify the configurations.

Device# show running-config interface FastEthernet3/1

Building configuration...

Current configuration : 114 bytes
!
interface FastEthernet3/1
no switchport
ipv6 address 1001::1/64
ipv6 traffic-filter hbh_acl
end
```

# その他の参考資料

### 関連資料

| 関連項目 | マニュアル タイトル |
|---|---|
| IPv6 アドレッシングと接続 | 『*IPv6 Configuration Guide*』 |
| Cisco IOS コマンド | 『*Cisco IOS Master Commands List, All Releases*』 |
| IPv6 コマンド | 『*Cisco IOS IPv6 Command Reference*』 |
| Cisco IOS IPv6 機能 | 『*Cisco IOS IPv6 Feature Mapping*』 |

### 標準規格および RFC

| 標準/RFC | タイトル |
|---|---|
| IPv6 に関する RFC | *IPv6 RFCs* |

### MIB

| MIB | MIB のリンク |
|---|---|
| | 選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。<br>http://www.cisco.com/go/mibs |

### シスコのテクニカル サポート

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

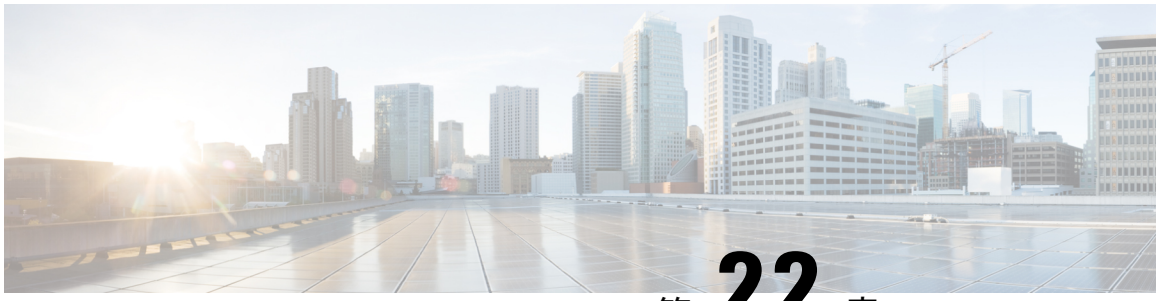
# ホップバイホップフィルタリングに対応するためのIPv6 ACLの拡張に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *22*：ホップ バイ ホップ フィルタリングに対応するための *IPv6 ACL* の拡張に関する機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| ホップ バイ ホップ フィルタリングに対応するための IPv6 ACL の拡張 | Cisco IOS リリース XE 3.4S<br>Cisco IOS リリース XE 3.5S<br>Cisco IOS リリース XE 3.6S<br>Cisco IOS リリース XE 3.3SG | これによって、ホップ バイ ホップ拡張ヘッダーを含む IPv6 トラフィックを制御できます。<br>次のコマンドが導入または変更されました。**deny**（IPv6）、**permit**（IPv6）。 |



第 22 章

# セキュリティ（ACL）の拡張機能

セキュリティ（ACL）の拡張機能では、1つのボックスで設定できるACL、ACE、またはこれらの両方の数を制限するオプションが用意されています。ボックスでACLまたはACEの数を制限することにより、ボックスのパフォーマンスに悪影響を与える可能性のあるTCAMスペースの枯渇または過使用を防ぐことができます。



## 機能制限

- acl-ace-limit の設定は、ACL ごとであり、ボックスのすべての ACL に適用されます。
- acl-limit および acl-ace-limit は、global-ace-limit と同時に使用できません。acl-limit と acl-ace-limt が設定されている場合、global-ace-limit は設定できず、global-ace-limit が設定されている場合、acl-limit と acl-ace-limt は設定できません。

  設定する制限は、ボックスの既存の ACL/ACE の数未満にはできません。
- acl-limit、acl-ace-limit、または global-ace-limit 設定は、デバイスの起動中に内部で作成された ACL/ACE に適用されます。
- オブジェクト グループ ACE（ogace）拡張を備えた ACL は、このリリースではサポートされていません。お客様の要件に基づいて、これは詳しく調査できます。各 ogace は 1 つの ace としてカウントされます。
- acl-limit、acl-ace-limit、または global-ace-limit 設定は、すべての静的 ACL および動的に作成されたすべての ACL に適用されます（ただし、テンプレート ACL は除きます）。
- 設定可能な acl-limit、acl-ace-limit、または global-ace-limit によって、TCAM スペースの過使用や枯渇が発生しなくなるという訳ではありません。ラボでの事前テストから、ボックスでサポートできる正確な設定可能制限を認知しておく必要があります。
- ボックスで設定されているすべての ACL がインターフェイスに適用されるということが前提であり、これは TCAM スペースに影響します。

- ボックスが設定可能な acl-limit、acl-ace-limit、または global-ace-limit に到達し、かつクライアントが動的 ACL/ACE を作成しようとすると、その要求は拒否され、syslog エラーメッセージが出力されます。これに応じて障害を処理するのはユーザの責任です。

# セキュリティ（ACL）の拡張機能の設定

V4 および V6 に対して ACL および ACE 制限を設定するには：

```
enable
configure terminal
 access-list acl-limit 10
 access-list acl-ace-limit 12
 access-list global-ace-limit 14
 end
```

（注） acl-limit および acl-ace-limit は、global-ace-limit と同時に使用できません。

### 特記事項

- 設定可能な最大 ACL 制限の範囲は 1 ～ $2^{16}$ です。
- 設定可能な ACL あたりの最大 ACE 制限の範囲は 1 ～ $2^{32}$ です。
- 設定可能な最大グローバル ACE 制限の範囲は 1 ～ $2^{32}$ です。
- acl-ace-limit 設定は、すでに設定されているすべての ACL、およびこれから設定されるすべての ACL に適用されます。

### セキュリティ（ACL）の拡張機能の設定の確認

**show access-list acl-limit** コマンドを使用すると、設定されている ACL と ACE の数を表示できます。

```
Device# show access-list acl-limit
Max ACLs configurable:       50
Number of ACLs configured:   10

Max aces/ACL configurable:   10

Max aces configurable:       100
Number of aces configured:   67
```

# セキュリティ（ACL）の拡張機能の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだ

けを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

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

**表 *23* : セキュリティ（*ACL*）の拡張機能の機能情報**

| 機能名 | リリース | 機能情報 |
|---|---|---|
| セキュリティ（ACL）の拡張機能 | Cisco IOS XE Everest 16.4.1 | セキュリティ（ACL）の拡張機能では、1つのボックスで設定できる ACL、ACE、またはこれらの両方の数を制限するオプションが用意されています。ボックスで ACL または ACE の数を制限することにより、ボックスのパフォーマンスに悪影響を与える可能性のある TCAM スペースの枯渇または過使用を防ぐことができます。<br>次のコマンドが導入または変更されました。<br>**acl-ace-limit**、**acl-limit**、**global-ace-limit**。 |