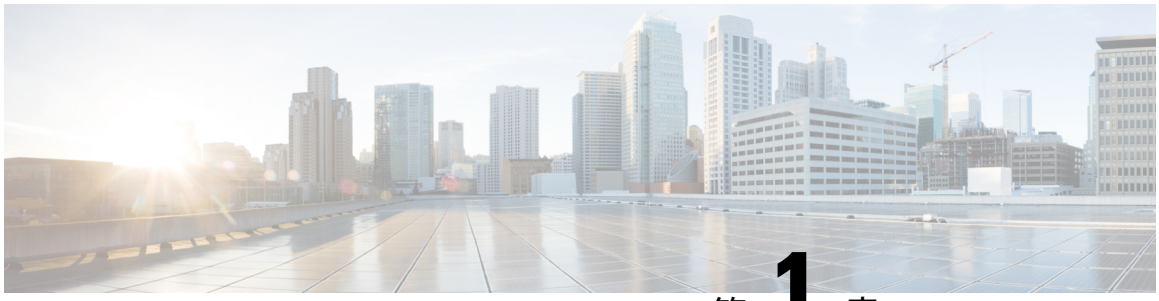
# Cisco IOS XE 17（Cisco NCS 520 シリーズ）セキュリティ コンフィギュレーション ガイド：アクセス制御リスト

**初版**：2019 年 11 月 26 日

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

# 目 次



第 1 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その場所を制御します。このような制御によって、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拒否攻撃の可能性を軽減し、ファイアウォールを介したダイナミックで一時的なユーザ 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 コントロール リストの制約事項

- **deny ip any any** コマンドは、その前に permit tcp any any *port-number* コマンドまたは permit udp any any *port-number* コマンドが使用されている場合、最初にフラグメント化されていないパケットを拒否しません。

  例:

  ```
  permit tcp any any eq 3000
  deny ip any any fragment
  ```

  ポート番号 4000 の TCP ストリームは、最初にフラグメント化されていないパケットに対して拒否されません。ACE には TCP ポート情報があるため、最初のフラグメントに対して正常に機能します。

## セキュリティ ACL の制約事項

- 出力 ACL はサポートされていません。
- 入力 MAC ACL は EFP インターフェイスでのみサポートされています。
- 入力 IP ACL は EFP インターフェイスでのみサポートされています。
- IPv6 ACL はサポートされません。
- MAC ACL は、予約済み MAC アドレスではサポートされていません。
- MAC ACL では、フィルタ条件の 1 つとしてイーサネットタイプはサポートされていません。
- ACL 統計はサポートされていません。
- ACL ロギングはサポートされていません。

# 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 アドレス リストの場合、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、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。

アクセスリストを設定しても、有効になるのは、アクセスリストがインターフェイスに適用されるか（**ip access-group** コマンドを使用）、仮想端末回線（VTY）に適用されるか（**access-class** コマンドを使用）、アクセスリストを受け入れるその他のコマンドで参照されてからです。アクセスリストの用途は多様なので、多くの Cisco IOS ソフトウェアコマンドの構文では、アクセスリストへの参照を受け入れています。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。

次の設定の抜粋で、先頭の 3 行は branchoffices という IP アクセスリストの例です。これは着信パケットのシリアルインターフェイス 0 に適用されます。このインターフェイスにアクセスできるのは、個々の各送信元アドレスとマスク ペアで指定されているネットワーク上の送信元のみです。ネットワーク 172.20.7.0 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 172.29.2.0 上の送信元から発信されるパケットの宛先は、172.25.5.4 にする必要があります。

```
ip access-list extended branchoffices
 10 permit 172.20.7.0 0.0.0.3 any
 20 permit 172.29.2.0 0.0.0.255 host 172.25.5.4
!
interface serial 0
 ip access-group branchoffices in
```

# アクセス リストのソフトウェア処理

アクセスリストがインターフェイス、vty に適用されるとき、または他の Cisco IOS コマンドにより参照されるときの、Cisco IOS ソフトウェアによる処理方法を説明した一般的な手順を次に示します。この手順は、アクセス リスト エントリが 13 以下のアクセス リストに適用されます。

- ソフトウェアが IP パケットや各パケットのテスト部分を受け取ります。これらは、アクセスリストの条件に一度に 1 つずつ（**permit** または **deny** ステートメント）照らし合わせてフィルタリングされます。たとえば、ソフトウェアは、**permit** あるいは **deny** ステートメントの送信元アドレスおよび宛先アドレスに照らし合わせてパケットの送信元アドレスおよび宛先アドレスをテストします。
- 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。
- アクセスリストがパケットを拒否する場合、ソフトウェアはパケットを廃棄し、ICMP ホスト到達不能メッセージを返します。
- いずれの条件とも一致しなかった場合、パケットは廃棄されます。これは、各アクセスリストが暗黙の **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。

リリース 12.4、12.2S、12.0S などの後の Cisco IOS リリースでは、デフォルトでは、13 個を超えるアクセスリストエントリを持つアクセスリストは、13 個以下のエントリを持つアクセスリストとは異なる方法で処理されます。効率を高めるために、13 を超えるエントリが含まれるアクセスリストは、trie ベースのルックアップアルゴリズムを使用して処理されます。このプロセスは自動的に行われます。設定する必要はありません。

# 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。
- 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 リストと番号付きアクセス リストはコマンド構文が異なります。名前付きアクセスリストは、Cisco IOS リリース 11.2 以降と互換性があります。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。用途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 リストよりも新しく、番号付きアクセス リストではサポートされていない次の機能をサポートします。

- TCP フラグ フィルタリング
- IP オプション フィルタリング
- 非隣接ポート
- 再帰アクセスリスト
- **no permit** または **no deny** コマンドでエントリを削除する機能

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、仮想端末回線では番号付きアクセスリストのみが使用されます。

# 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）、TCP フラグ、IP オプション、TTL 値などです。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）
- 時間ベースのエントリ（「時間ベースおよび分散型時間ベースのアクセスリスト」および「Refining an IP Access List」モジュールを参照してください）
- ダイナミックアクセスリスト（「IPアクセスリストのタイプ」の項を参照してください）
- 再帰アクセスリスト（「IP アクセスリストのタイプ」の項および「Configuring IP Session Filtering [Reflexive Access Lists]」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 |

| アドレス | ワイルドカード マスク | 一致する結果 |
|---|---|---|
| 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 | 0.0.254.255（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。

**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。

## 時間ベースおよび分散型時間ベースのアクセスリスト

時刻ベースのアクセス リストでは、その日または週の特定の時刻に基づいて、アクセス リスト エントリを実装します。これは、アクセス リスト エントリを常に有効にしない場合、または適用されるとすぐに有効にする場合に適した方法です。時刻と日付に基づいて許可または拒否条件の実施を細かくするには、時刻ベースのアクセス リストを使用します。

分散型時間ベースのアクセスリストは、Cisco 7500 シリーズ ルータのラインカードでサポートされているものです。時間ベースのアクセスリストを使用して設定されたインターフェイス宛てのパケットは、ラインカードを介して分散スイッチングされます。

# IP アクセスリストのタイプ

複数のタイプのアクセスリストがあり、トリガー方法、一時的な性質、または通常のアクセスリストとの動作の違いによって区別されています。

### 認証プロキシ

認証プロキシでは、動的かつユーザごとの認証と認可、業界標準の TACACS+ および RADIUS 認証プロトコルを使用したユーザの認証が可能です。ユーザによる接続の認証と認可により、ネットワーク攻撃に対するより強力な保護が可能になります。

### コンテキストベース アクセス コントロール

コンテキストベースアクセス制御（CBAC）は、ネットワーク層とトランスポート層の情報だけでなく、アプリケーション層プロトコル情報（FTP 情報など）も参照して、TCP および UDP 接続の状態を学習します。CBAC は、個々の接続の接続状態情報を管理します。この状態情報は、パケットを許可するか拒否するかについてインテリジェントな判断を行うために使用され、ファイアウォールの一時的な開口部を動的に作成および削除します。

### ロックアンドキー機能を使用した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は、Telnet を使用して指定されたユーザに、ファイアウォールを通過して指定されたホストに到達するための一時的なアクセスを提供します。ダイナミックアクセスリストには、ユーザ認証および認可が関係します。

### 再帰アクセスリスト

再帰アクセスリストは、上位層の IP プロトコルセッションのフィルタリングを提供します。これには、新しい IP セッションの開始時に自動的に作成される一時的なエントリが含まれています。これは、インターフェイスに適用される名前付き拡張 IP アクセスリスト内でネストされます。再帰アクセスリストは、通常、内部ネットワークと外部ネットワーク間でトラフィックを渡す境界ルータで設定されます。これは多くの場合、ファイアウォールルータです。再帰アクセスリストは、アクセスリスト内でネストされ、後続のステートメントを検査する必要があるため、暗黙的な deny ステートメントで終了しません。

# アクセス リストを適用する場所

アクセスリストをインターフェイスに適用する場合、**in**（インバウンド）と **out**（アウトバウンド）のいずれを指定するかについては慎重に考慮してください。着信または発信インターフェイスに対してアクセスリストを適用して、ルータのインターフェイスで発着信するトラフィックまたはプロセスレベルを制御します（TTL 値に基づいてフィルタする場合）。

- インバウンド アクセス リストをインターフェイスに適用すると、ソフトウェアはパケットを受信した後に、アクセス リスト ステートメントに対してパケットを確認します。アクセスリストでパケットが許可されている場合、ソフトウェアはパケットの処理を続行します。結果として、着信パケットに関するフィルタリングによって、フィルタされたパケットはルータを通過しないため、ルータ リソースを節約できます。

- アウトバウンド パケットに適用するアクセス リストは、ルータをすでに通過したパケットをフィルタします。アクセスリストに合格したパケットは、インターフェイスから伝送（送信）されます。
- Rate-Based Satellite Control Protocol（RBSCP）の TCP ACL 分割機能は、発信インターフェイスで使用できる機能の一例です。アクセスリストで、TCP ACK 分割の対象となるパケットを制御します。

インターフェイスに適用する以外の方法でもアクセスリストを使用できます。次に、アクセスリストを適用できるその他の場所を示します。

- 着信接続および発信接続を特定の（シスコデバイスへの）vty とアクセスリスト内のアドレスにあるネットワークデバイスとの間に制限するには、アクセスリストを回線に適用します。「Controlling Access to a Virtual Terminal Line」モジュールを参照してください。
- **debug** コマンドからアクセスリストを参照すると、表示される情報量は、アクセスリストで許可されている情報にのみ限定されます。たとえば、送信元、宛先、プロトコルなどです。
- アクセス リストは、ルーティング アップデートの制御、ダイヤルオンデマンド ルーティング（DDR）の制御、および Quality of Service（QoS）機能の制御などに使用できます。これらの機能にアクセスリストを使用する方法については、該当する設定の章を参照してください。

# 次の作業

最初に制限する対象を決定してから、目標を達成するアクセスリストのタイプを選択する必要があります。次に、指定したフィールドの値に基づいてパケットを許可または拒否するアクセスリストを作成し、最後にそのアクセスリストを適用します（その配置を決定します）。

制限する対象と必要なアクセスリストのタイプを決定していると想定すると、次の手順はアクセスリストを作成することです。送信元アドレス、宛先アドレス、またはプロトコルに基づいてアクセスリストを作成する方法については、「IP アクセスリストの作成とインターフェイスへの適用」モジュールで説明されています。『Creating an IP Access List to Filter IP Options, TCP Flags, Noncontiguous Ports, or TTL Values』の説明に従って、他のフィールドでフィルタ処理するアクセスリストを作成できます。仮想回線へのアクセスを制御する場合は、『Controlling Access to a Virtual Terminal Line』を参照してください。アクセスリストの目的がルーティング アップデートまたは QoS 機能を制御することの場合は、たとえば、適切な技術に関する章を参照してください。

# 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 Application 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 | 『Creating an IP Access List and Applying It to an Interface』 |
| 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, Noncontiguous Ports, or TTL Values』 |
| vty 回線へのアクセスの制限 | 『Controlling Access to a Virtual Terminal Line』 |

### 標準

| 標準 | タイトル |
|---|---|
| なし | -- |

### MIB

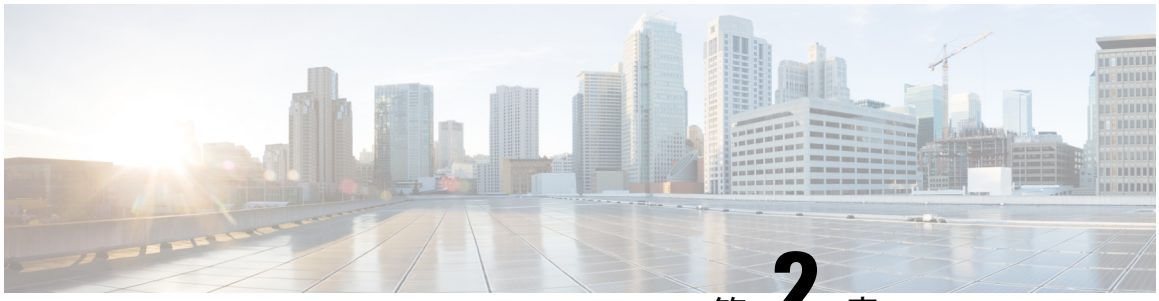
| MIB | MIB のリンク |
|---|---|
| なし | 選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。<br>http://www.cisco.com/go/mibs |

### RFC

| RFC | タイトル |
|---|---|
| なし | -- |

### シスコのテクニカル サポート

| 説明 | リンク |
|---|---|
| ★枠で囲まれた Technical Assistance の場合★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 |



第 2 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、このモジュールで参照していますが、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とインターフェイスへの適用の前提条件

IP アクセスリストを作成または適用する前に、「IP アクセスリストの概要」モジュールの概念を理解しておく必要があります。また、ネットワークで IP が実行されている必要があります。

# 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 アクセス制御リスト（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 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。
- 最大で 256 の TCAM がサポートされます。そのため、ACL ごとに 256 の ACE がサポートされます。
- IPv4 ACL は、EFP インターフェイスでのみサポートされています。
- 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。
- ACL 統計はサポートされていません。
- 最初のステートメントが **deny permit** の場合は、**permit** トラフィックのステートメントを追加する必要があります。

# 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情報

## 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。
- 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。アクセスリストをインターフェイスに適用しない場合は、アクセスリストのその他の適用方法を説明するモジュールを示す、「次の作業」を参照してください。

# 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 モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

**例**：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **ip 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**}

例：

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

（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。

**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**}

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 モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

例：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

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

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**}

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に基づいてフィルタする場合、名前付き拡張アクセス リストを作成します。

**ステップ 1** **enable**

例：

```
Router> enable
```

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

例：

```
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*

例：

```
Router(config)# ip access-list extended nomarketing
```

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*

**例**：

```
Router(config-ext-nacl)# remark protect server by denying access from the Marketing 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。
- この例では、注釈によって、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 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*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**time-range** *time-range-name*] [**fragments**]

**例**：

```
Router(config-ext-nacl)# deny ip 172.18.0.0 0.0.255.255 host 172.16.40.10
```

（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。

- *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。
- 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。
- 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。

**ステップ 6** **remark** *remark*

**例**：

```
Router(config-ext-nacl)# remark allow TCP from any source to any destination
```

（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。

**ステップ 7** **permit** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**time-range** *time-range-name*] [**fragments**]

**例**：

```
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。

- 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。

- *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。
- 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。
- 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。

**ステップ 8** 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 9** **end**

**例**：

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

コンフィギュレーション モードを終了し、システム特権 EXEC モードに変更します。

**ステップ 10** **show ip access-list**

**例**：

```
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。

## 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。

**ステップ 1** **enable**

**例**：

```
Router> enable
```

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

**例**：

```
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** *access-list-number* **remark** *remark*

例：

```
Router(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。

- 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**] [**time-range** *time-range-name*] [**fragments**]

例：

```
Router(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。

- 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。
- 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。
- *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。
- 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。
- 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*

例：

```
Router(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。

- 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**] [**time-range** *time-range-name*] [**fragments**]

例：

```
Router(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。

- *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。
- 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**

**例**：

```
Router(config)# end
```

コンフィギュレーション モードを終了し、システム特権 EXEC モードに変更します。

**ステップ 9** **show ip access-list**

**例**：

```
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。

# インターフェイスへのアクセス リストの適用

インターフェイスにアクセス リストを適用するには、次の作業を実行します。

**ステップ 1** **enable**

**例**：

```
Router> enable
```

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

**例**：

```
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** **interface** *type slot /subslot /port*

例：

```
Router(config)# interface GigabitEthernet 0/0/1
```

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**}

例：

```
Router(config-if)# ip access-group noncorp 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。

- 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。
- 送信元アドレスに基づくフィルタは、宛先の近くで適用すると、最も効率的です。

# 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：送信元アドレス（ホスト）に基づくフィルタリング

次の例では、Jones に属するワークステーションが Gigabitethernet インターフェイス 0/0/2 へのアクセスを許可され、Smith 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 0/0/2
service instance 10 ethernet
 ip access-group workstations in
!
ip access-list standard workstations
 remark Permit only Jones workstation through
 permit 172.16.2.88
 remark Do not allow Smith workstation through
 deny 172.16.3.13
```

## 例：送信元アドレス（サブネット）に基づくフィルタリング

次の例では、Jones サブネットは Gigabitethernet インターフェイス 0/0/2 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
service instance 10 ethernet
 ip access-group prevention in
```

```
!
ip access-list standard prevention
 remark Do not allow Jones 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：送信元アドレス、宛先アドレス、および IP プロトコルに基づくフィルタリング

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。一方のリストは着信パケットに適用されます。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 0/0/5
service instance 10 ethernet


 ip access-group marketing_group in
!

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（ホストおよびサブネット）に基づくフィルタリング

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 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 0/0/3
service instance 10 ethernet
 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、Jones サブネットは、Gigabitethernet インターフェイス 0/0/2 からの Telnet の発信が許可されていません。

```
interface Gigabitethernet 0/0/2
service instance 10 ethernet
 ip access-group telnetting  in
!
ip access-list extended telnetting
 remark Do not allow Jones 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、goodports 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 0/0/2
service instance 10 ethernet
 ip access-group goodports in
!
ip access-list extended goodports
 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、IP ホストからは、専用メールホストのメール（SMTP）ポートを除き、イーサネット上のホストと TCP 接続を確立できないようにします。

SMTP は、接続の一端では TCP ポート 25、もう一端ではランダムなポート番号を使用します。接続の存続中は、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、イーサネット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 0/0/2
service instance 10 ethernet
 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リングによる Web へのアクセス回避

次の例では、Winter および Smith 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 Gigabitethernet 0/0/2
service instance 10 ethernet
 ip access-group no_web  in
!
ip access-list extended no_web
 remark Do not allow Winter to browse the web
 deny host 172.20.3.85 any eq http
 remark Do not allow Smith 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：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 standard acl1

!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
```

# 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 | https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mcl/allreleasemcl/all-book.html |

### 標準および RFC

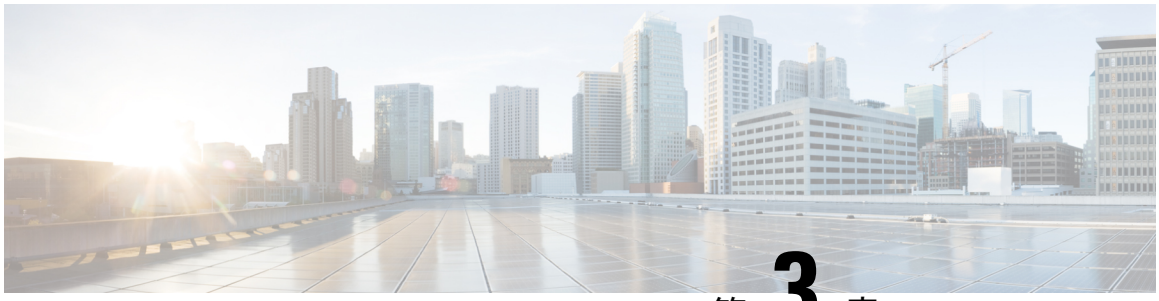
| 標準/RFC | タイトル |
|---|---|
| このマニュアルに記載された機能によってサポートされている特定の標準規格および RFC はありません。 | — |

### MIB

| MB | 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 |
|---|---|
| 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/cisco/web/support/index.html |



第 3 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 *2 : 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。

# 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 モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

例：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **ip 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*] [ **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*] [ **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 モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

**例**：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **ip 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*] [**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*] [**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。

# 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 モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

**例**：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

**ステップ 3** **ip 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*] [**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*] [**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 モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **show ip 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*] [ **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*] [ **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。

# 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
```

# その他の参考資料

### 関連資料

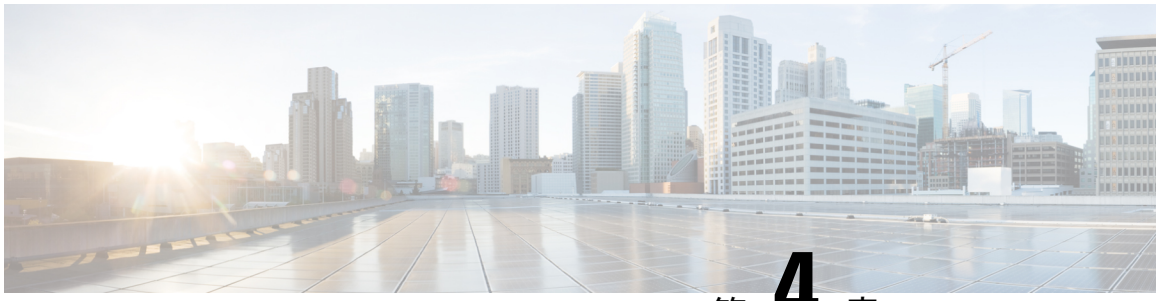
| 関連項目 | マニュアル タイトル |
|---|---|
| Cisco IOS コマンド | 『Cisco IOS Master Command List, All Releases』 |
| セキュリティ コマンド | 『*Cisco IOS Security Command Reference*』 |
| **no ip options**コマンドを使用した、IP オプションを含むパケットをドロップまたは無視するためのデバイスの設定。 | 『ACL IP Options Selective Drop』 |
| アクセス リストに関する概要情報 | 『IP Access List Overview』 |
| 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*』 |

### シスコのテクニカル サポート

| 説明 | リンク |
|---|---|
| ★枠で囲まれた Technical Assistance の場合★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 |



第 4 章

# MAC アクセス制御リスト

この章では、シスコルータでの MAC アクセス制御リスト（ACL）の設定方法について説明します。ここで説明する内容は、次のとおりです。



## 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。

## MAC アクセス制御リストの前提条件

- MAC ACL を設定するためには、MAC アドレッシングおよびプロトコルに関する知識が必要です。

# MAC アクセス制御リストの制約事項

- MAC ACL は、EFP または TEFP でのみサポートされています。
- MAC ACL は、IP パケットではサポートされていません。
- MAC ACL カウンタはサポートされていません。
- MAC ACL は、ポート、ルーテッドインターフェイス、および BDI ではサポートされていません。
- ACL と QoS は、同じ EFP に適用できます。
- アウトバウンド MAC ACL はサポートされていません。

# MAC アクセス制御リストに関する情報

## MAC アクセス制御リスト

MAC ACL は、各パケットのレイヤ 2 ヘッダー内の情報を使用してトラフィックをフィルタリングする ACL です。ACL を使用すると、ネットワークの場所ごとにアクセス可能なホストを制御したり、ルータインターフェイスで転送またはブロックされるトラフィックの種類を決定したりできます。MAC ACL は、EFP とクロスコネクトでサポートされています。

# MAC アクセス制御リストの設定方法

## ACL の設定

ACL を設定するには、次の手順を実行します。

**ステップ 1** **enable**

**例：**

```
Router> enable
```

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

**ステップ 2** **configure terminal**

**例：**

```
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** **mac access-list extended** *name*

**例**：

```
Router(config)# mac access-list ext macext2
```

拡張 MAC アクセス制御リスト（ACL）を作成し、そのアクセス制御エントリ（ACE）を定義します。

- *name*：エントリが属する ACL 名。

**ステップ 4** {**permit** | **deny**} {**any** | **host** *src-MAC-addr*} {**any** | **host** *dst-MAC-addr*}

**例**：

```
Router(config-ext-macl)# deny any any
```

条件が一致した場合にレイヤ 2 トラフィックが転送されるのを許可または拒否します。

- **permit**：条件が一致した場合にレイヤ 2 トラフィックが転送されるのを許可します。
- **deny**：条件が一致した場合にレイヤ 2 トラフィックが転送されるのを拒否します。
- **any**：送信元または宛先 MAC アドレスを拒否するキーワードです。
- **host** *src-MAC-addr*：ホスト MAC アドレスを定義します。MAC アドレス ベースのサブネットは許可されません。
- **host** dst-MAC-addr：宛先 MAC アドレスを定義します。MAC アドレス ベースのサブネットは許可されません。

**ステップ 5** **end**

**例**：

```
Router(config-ext-macl)# end
```

特権 EXEC モードに戻ります。

# MAC アクセス制御リストの確認

MAC ACL 設定を確認するには、次の **show** コマンドを使用します。

- **show access-lists** *name*：名前付きアクセスリストに関する情報を表示します。

```
Router# show access-list macext4
```

```
Extended MAC access list macext4 permit any host 0000.0000.0009 permit any
 host 0000.0000.0010 permit any host 0000.0000.0011 permit any host
0000.0000.0012
```

# MAC アクセス制御リストの設定例

## MAC ACL の設定

### 例：指定された送信元または宛先 MAC アドレスの許可

```
(config)#mac access-list extended macext5
(config-ext-macl)#permit any host 0000.0000.0009
(config-ext-macl)#permit any host 0000.0000.0010
(config-ext-macl)#permit any host 0000.0000.0011
(config-ext-macl)#permit any host 0000.0000.0012
```

### 例：すべての送信元または宛先 MAC アドレスの許可

```
(config)#mac access-list extended macext9
(config-ext-macl)#permit any any
```

# MAC アクセス制御リストに関する追加情報

### 関連資料

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

### 標準および RFC

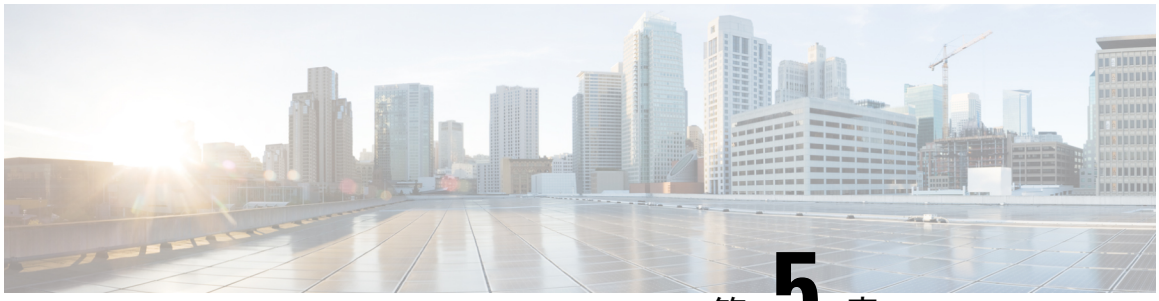
| 標準/RFC | タイトル |
|---|---|
| 標準 | — |

### MIB

| MIB | MIB のリンク |
|---|---|
| • CISCO-MB | 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 |
| --- | --- |
| 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><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><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/cisco/web/support/index.html |



第 5 章

# ストーム制御の設定

ここでは、ルータ上でストーム制御を設定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。

## ストーム制御の前提条件

- ポートレベルのストーム制御は、EVC インターフェイスで設定する必要があります。
- ストーム制御のしきい値は、CIR（bps、kbps、%）として設定する必要があります。
- すべてのタイプのストーム（ユニキャスト、ブロードキャスト、およびマルチキャスト）に適用されます。

# ストーム制御の制約事項

- ストーム制御は、EVC 設定を使用するポートでのみ有効になります。
- ストーム制御は、レイヤ2物理インターフェイスとポートチャネルに固有のものです。レイヤ 3 インターフェイスまたは BDI ではサポートされていません。
- ストーム制御は、不明なユニキャスト、ブロードキャスト、および不明なマルチキャストの入力トラフィックでのみサポートされています。出力トラフィックではサポートされていません。
- ルータでは、ポートレベルのストーム制御がサポートされています。EFP レベルのストーム制御はサポートされていません。
- ローカル接続およびクロスコネクトでのストーム制御はサポートされていません。

# ストーム制御に関する情報

ストームは、大量のブロードキャスト、マルチキャスト、またはユニキャストのパケットが LAN にフラッディングすると発生し、過剰なトラフィックが生み出され、ネットワークパフォーマンスを低下させます。プロトコルスタック実装内またはネットワーク設定内のエラーも、ストームの原因となる場合があります。このようなイベントを防止して制御するメカニズムを、ストーム制御と呼びます。

ストーム制御は着信トラフィックレベルを、1 秒ごとのトラフィックストーム制御でモニタします。そのインターバルの中で、トラフィックレベルを、設定したトラフィックストーム制御レベルと比較します。トラフィックストーム制御しきい値レベルは、ポートの利用可能な帯域幅全体に対する割合です。各ポートには、ブロードキャスト、マルチキャスト、およびユニキャストタイプのトラフィック用の、さまざまなストーム制御レベルがあります。

ストーム制御は上限と下限のしきい値を使用して、ブロードキャスト、ユニキャスト、またはマルチキャストのパケットの転送をブロックしてから復元します。

- 上限しきい値は、超えるとその特定のトラフィックがブロックされるトラフィック制限です。
- 下限しきい値は、下回ると特定のトラフィックがブロックされている場合にそのトラフィックの転送が再び開始されるトラフィック制限です。

（注） 特定のタイプの入力トラフィック（ユニキャスト、ブロードキャスト、マルチキャスト）が、設定されている上限しきい値を超えた場合、インターフェイスはその特定のトラフィックに対してブロック状態になります。

ストーム制御は、ポート上で発生したブロードキャスト、マルチキャスト、またはユニキャスト ストームによって LAN 上のトラフィックが混乱することを防ぎます。ストーム制御は物理インターフェイスに適用され、レイヤ2インターフェイス上のユニキャスト、ブロードキャスト、およびマルチキャストの入力トラフィックを制限するために使用されます。この機能は、ルータではデフォルトで無効になっています。

# ストーム制御の設定

始める前に

- EVC 設定を使用してポートを設定します。

（注） ストーム制御機能を無効にするには、**no storm-control** コマンドを使用します。

ストーム制御を設定するには、次の手順を実行します。

- 特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

```
Router> enable
```

- グローバル コンフィギュレーション モードを開始します。

```
Router# configure terminal
```

- インターフェイス タイプを設指定して、インターフェイス コンフィギュレーション モードを開始します。

```
Router(config)# interface gigabitethernet 0/0/0
```

- グローバルブロードキャスト、マルチキャスト、またはユニキャストストーム制御の抑制レベルを指定します。
  - **broadcast**：ブロードキャストストーム制御を設定します。
  - **multicast**：マルチキャストストーム制御を設定します。
  - **unicast**：不明なユニキャストストーム制御を設定します。
  - **level**：ブロードキャスト、マルチキャスト、またはユニキャストトラフィックのしきい値レベルを指定します。
  - *rising_threshold*：上限しきい値レベル。
  - *falling_threshold*：下限しきい値レベル。
  - **bps**：抑制レベルをビット/秒単位で指定します。
  - **pps**：抑制レベルをパケット/秒単位で指定します。

```
Router(config)# storm-control broadcast level 1 .50
```

- ポート上でストームが発生した場合に実行するアクションを指定します。
  - **shutdown**：ストームの間、ポートを無効にします。**shutdown** アクションは、ストームの間、ポートをシャットダウン状態に設定します。ストームが設定された下限しきい値を下回ったときに **no shutdown** コマンドが出されて回復されるまで、ポートはシャットダウン状態のままになります。
  - **trap**：SNMP トラップを送信します。**trap** アクションは、ストームが検出されると SNMP トラップを生成します。デフォルトでは、特定の入力トラフィックを制限し、トラップは送信しません。

    ```
    Router(config)# storm-control action trap
    ```

- 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、ルータをグローバル コンフィギュレーション モードに戻します。

  ```
  Router(config)# exit
  ```

**設定例**:

```
interface GigabitEthernet0/0/1
 no ip address
 negotiation auto
 storm-control broadcast level bps 50k 40k
 storm-control multicast level pps 100 90
 storm-control unicast level 1.00 0.50
service instance 1 ethernet
  encapsulation dot1q 2
  rewrite ingress tag pop 1 symmetric
  bridge-domain 1
!
```

# ストーム制御の確認

- ストーム制御機能の設定を確認するには、**show storm-control** コマンドを使用します。

  ```
  Router# show storm-control

  Interface   Type   Filter State   Upper         Lower         Current
  ---------  ------   -------------  -----------  -----------  ----------
  Gi0/0/0    Bcast    Forwarding          0 pps        0 pps        0 pps
  Gi0/0/0    Ucast    Forwarding       80.00%       20.00%       39.99%
  Gi0/0/1    Bcast    Blocking           50k bps      40k bps  362.25k bps
  Gi0/0/1    Mcast    Blocking           100 pps       90 pps      265 pps
  Gi0/0/1    Ucast    Blocking          1.00%        0.50%        1.28%
  ```

（注） 現在のトラフィックレートが表示方法を下回るたびに、ストーム制御がインターフェイス上のトラフィックをブロックし、現在のトラフィックレートは常に 0% と表示されます。

- 設定されたストームタイプのストーム制御設定を表示するには、**show storm-control** コマンドを使用します。

```
Router# show storm-control broadcast
Interface   Type   Filter State    Upper        Lower        Current
---------  ------  -------------  -----------  -----------  ----------
Gi0/0/2    Bcast    Blocking         280k pps     260k pps  284.64k pps
Te0/0/12   Bcast    Blocking        2.29g bps       2g bps    2.49g bps
Te0/0/13   Bcast    Blocking         4.6m pps     4.3m pps    4.69m pps
Po4        Bcast    Link Down         45k pps      43k pps        0 pps
Po5        Bcast    Blocking         240k pps     235k pps  241.85k pps
Router#
Router#show storm-control unicast
Interface   Type   Filter State   Upper        Lower        Current
---------  ------  -------------  -----------  -----------  ----------
Gi0/0/2    Ucast    Blocking         290k pps     280k pps  298.19k pps
Te0/0/12   Ucast    Blocking           4m pps     3.5m pps    4.74m pps
Te0/0/13   Ucast    Blocking         4.5m pps     4.3m pps    4.91m pps
Po4        Ucast    Link Down         45k pps      43k pps        0 pps
Po5        Ucast    Blocking         250k pps     240k pps  253.36k pps
Router#
Router#show storm-control multicast
Interface   Type   Filter State   Upper        Lower        Current
---------  ------  -------------  -----------  -----------  ----------
Gi0/0/2    Mcast    Blocking         240m bps     220m bps   260.5m bps
Te0/0/12   Mcast    Blocking        2.25g bps       2g bps    2.38g bps
Te0/0/13   Mcast    Blocking           3g bps     2.5g bps     3.3g bps
Po4        Mcast    Link Down         45k pps      43k pps        0 pps
Po5        Mcast    Blocking         200k pps     190k pps  206.28k pps
Router#
```

- インターフェイスでのストーム制御機能の設定を確認するには、**show storm-control GigabitEthernet** コマンドを使用します。

```
Router # show storm control GigabitEthernet 0/0/1

Interface   Type   Filter State   Upper        Lower        Current
---------  ------  -------------  -----------  -----------  ----------
Gi0/0/1    Bcast    Blocking          50k bps      40k bps  362.25k bps
Gi0/0/1    Mcast    Blocking          100 pps       90 pps      265 pps
Gi0/0/1    Ucast    Blocking         1.00%        0.50%        1.28%
```

- ポートに設定されているアクショントラップを確認するには、**show run interface** コマンドを使用します。

```
Router# show run interface GigabitEthernet 0/0/1

Building configuration...
Current configuration : 300 bytes
!
interface GigabitEthernet0/0/1
 no ip address
 negotiation auto
storm-control broadcast level 9.00 7.00
 storm-control action trap
 service instance trunk 1 ethernet
  encapsulation dot1q 1-200
  rewrite ingress tag pop 1 symmetric
  bridge-domain from-encapsulation
 !
end
```

- 次に、ストームがヒットしたときに **action trap** が送信される例を示します。

```
Router# show storm-control G 0/0/1
Interface   Type    Filter State   Upper         Lower         Current
---------   ------   -------------  -----------  -----------  -----------
Gi0/4/2     Bcast     Blocking          9.00%         7.00%        11.00%
May 29 14:46:28.008 IST: %STORM_CONTROL-3-TRAP: A packet storm was detected on
Gi0/4/2.
Sending SNMP trap
```

- 次に、**action shutdown** の設定例を示します。

```
Router# show run interface Gi0/0/1

Building configuration...
Current configuration : 300 bytes
!
interface GigabitEthernet0/0/1
 no ip address
 negotiation auto
storm-control broadcast level 9.00 7.00
 storm-control action shutdown
 service instance trunk 1 ethernet
  encapsulation dot1q 1-200
  rewrite ingress tag pop 1 symmetric
  bridge-domain from-encapsulation
 !
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 |
|---|---|
| Cisco IOS コマンド | https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mcl/allreleasemcl/all-book.html |

### 標準および RFC

| 標準/RFC | タイトル |
|---|---|
| このマニュアルに記載された機能によってサポートされている特定の標準規格および RFC はありません。 | — |

### 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 |
| --- | --- |
| 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/cisco/web/support/index.html |