IPv4 アクセス コントロール リストの設定

- IPv4 アクセス コントロール リストの設定に関する制約事項 （1 ページ）
- IPv4 アクセスコントロールリストに関する情報 （2 ページ）
- ACL の設定方法 （12 ページ）
- IPv4 ACL のモニタリング （30 ページ）
- ACL の設定例 （31 ページ）
- 例：ACL のトラブルシューティング （37 ページ）
- IPv4 アクセスコントロールリストに関する追加情報 （38 ページ）
- IPv4 アクセス コントロール リストに関する機能情報 （39 ページ）

IPv4アクセスコントロールリストの設定に関する制約事項

一般的なネットワーク セキュリティ

次は、ACL によるネットワーク セキュリティの設定の制約事項です。

- ルータ ACL と VLAN ACL はサポートされていません。
- 番号付き ACL で使用できるすべてのコマンドが名前付き ACL でも使用できるわけではありません。インターフェイスのパケット フィルタおよびルート フィルタ用の ACL では、名前を使用できます。
- 標準 ACL と拡張 ACL に同じ名前は使用できません。
- **appletalk** は、コマンドラインのヘルプ ストリングに表示されますが、**deny** および **permit** MAC アクセスリスト コンフィギュレーション モード コマンドの一致条件としてサポートされていません。
- ACL ワイルドカードは、ダウンストリーム クライアント ポリシーではサポートされていません。

IPv4 ACL ネットワーク インターフェイス

次の制限事項が、ネットワーク インターフェイスへの IPv4 ACL に適用されます。

- インターフェイスへのアクセスを制御する場合、名前付き ACL または番号付き ACL を使用できます。
- レイヤ 2 インターフェイスに ACL を適用する場合、ルーティングをイネーブルにする必要はありません。
- レイヤ 3 ポートおよび SVI では、ACL はサポートされていません。

レイヤ 2 インターフェイスの MAC ACL

MAC ACL を作成し、それをレイヤ 2 インターフェイスに適用すると、そのインターフェイスに着信する非 IP トラフィックをフィルタリングできます。MAC ACL を適用するときには、次の注意事項に留意してください。

- 同じレイヤ 2 インターフェイスには、IP アクセス リストと MAC アクセス リストを 1 つずつしか適用できません。IP アクセス リストは IP パケットだけをフィルタリングし、MAC アクセス リストは非 IP パケットをフィルタリングします。
- 1 つのレイヤ 2 インターフェイスに適用できる MAC アドレス リストは 1 つだけです。すでに MAC ACL が設定されているレイヤ 2 インターフェイスに MAC アクセス リストを適用すると、設定済みの ACL が新しい ACL に置き換えられます。

（注） **mac access-group** インターフェイス コンフィギュレーション コマンドは、物理レイヤ 2 インターフェイスに適用される場合のみ有効です。このコマンドは、EtherChannel ポート チャネルでは使用できません。

IP アクセス リスト エントリ シーケンス番号

- この機能は、ダイナミック アクセス リスト、再帰アクセス リスト、またはファイアウォール アクセス リストをサポートしていません。

IPv4 アクセスコントロールリストに関する情報

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、デバッグ出力の制限、Quality of Service（QoS）機能のためのトラフィックの識別または分類などです。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。

ACL の概要

パケット フィルタリングは、ネットワーク トラフィックを限定し、特定のユーザまたはデバイスによるネットワークの使用を制限するうえで役立ちます。ACL はルータまたはスイッチを通過するトラフィックをフィルタリングし、特定のインターフェイスまたは VLAN（仮想 LAN）でパケットを許可、または拒否します。ACL は、パケットに適用される許可条件および拒否条件の順序付けられた集まりです。パケットがインターフェイスに着信すると、スイッチはパケット内のフィールドを適用される ACL と比較し、アクセス リストに指定された基準に基づいて、パケットが転送に必要な権限を持っているかどうかを確認します。アクセス リスト内の条件を 1 つずつ調べ、パケットをテストします。最初に一致した条件によって、スイッチがパケットを受け入れるか拒否するかが決定されます。スイッチは最初に一致した時点でテストを中止するので、リストに条件を指定する順序が重要です。一致する条件がない場合、スイッチはパケットを拒否します。スイッチは、制限条件がない場合はパケットを転送し、制限条件がある場合はパケットをドロップします。スイッチは、VLAN 内でブリッジングされるパケットを含めて、転送されるすべてのパケットに ACL を使用します。

ネットワークに基本的なセキュリティを導入する場合は、ルータにアクセス リストを設定します。ACL を設定しなければ、スイッチを通過するすべてのパケットがネットワークのあらゆる部分で許可される可能性があります。ACL を使用すると、ネットワークの場所ごとにアクセス可能なホストを制御したり、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。たとえば、電子メール トラフィックの転送を許可し、Telnet トラフィックの転送を拒否することもできます。ACL を着信トラフィック、発信トラフィック、またはその両方をブロックするように設定することもできます。

標準 IPv4 ACL および拡張 IPv4 ACL

ここでは、IP ACL について説明します。

ACL は、許可条件と拒否条件の順序付けられた集まりです。スイッチは、アクセス リスト内の条件を 1 つずつ調べ、パケットをテストします。最初に一致した条件によって、スイッチがパケットを受け入れるか拒否するかが決定されます。スイッチは一致する最初の条件が見つかった時点でパケットのテストを停止するため、条件の順序が重要な意味を持ちます。一致する条件がない場合、スイッチはパケットを拒否します。

このソフトウェアは、IPv4 について次の ACL（アクセス リスト）をサポートします。

- 標準 IP アクセス リストでは、照合操作に送信元アドレスを使用します。
- 拡張 IP アクセス リストでは、照合操作に送信元アドレスと宛先アドレスを使用し、任意でプロトコル タイプ情報を使用して制御のきめ細かさを高めることもできます。

IPv4 ACL スイッチでサポートされていない機能

このスイッチで IPv4 ACL を設定する手順は、他の Cisco スイッチやルータで IPv4 ACL を設定する手順と同じです。

以下の ACL 関連の機能はサポートされていません。

- 非 IP プロトコル ACL またはブリッジ グループ ACL
- IP アカウンティング
- 着信および発信レート制限（QoS ACL によるレート制限を除く）
- 再帰 ACL およびダイナミック ACL はサポートされていません。（スイッチのクラスタリング機能で使用される特別なダイナミック ACL を除く）
- VLAN マップの ACL ロギング

アクセス リスト番号

ACL を識別するために使用する番号は、作成するアクセスリストタイプを表します。

次の一覧に、アクセス リスト番号と対応するアクセス リスト タイプを挙げ、このスイッチでサポートされているかどうかを示します。このスイッチは、IPv4 標準アクセス リストおよび拡張アクセス リスト（1 ～ 199 および 1300 ～ 2699）をサポートします。

表 *1:* アクセス リスト番号

アクセス リスト番号	タイプ	サポートあり
1 ～ 99	IP 標準アクセス リスト	あり
100 ～ 199	IP 拡張アクセス リスト	あり
200 ～ 299	プロトコル タイプコード アクセス リスト	なし
300 ～ 399	DECnet アクセス リスト	なし
400 ～ 499	XNS 標準アクセス リスト	なし
500 ～ 599	XNS 拡張アクセス リスト	なし
600 ～ 699	AppleTalk アクセス リスト	なし
700 ～ 799	48 ビット MAC アドレス アクセス リスト	なし
800 ～ 899	IPX 標準アクセス リスト	なし
900 ～ 999	IPX 拡張アクセス リスト	なし
1000 ～ 1099	IPX SAP アクセス リスト	なし

アクセス リスト番号	タイプ	サポートあり
1100 ～ 1199	拡張 48 ビット MAC サマリー アドレス アクセス リスト	なし
1200 ～ 1299	IPX サマリー アドレス アクセス リスト	なし
1300 ～ 1999	IP 標準アクセス リスト（拡張範囲）	あり
2000 ～ 2699	IP 拡張アクセス リスト（拡張範囲）	あり

番号付き標準 ACL および番号付き拡張 ACL に加え、サポートされる番号を使用して名前付き標準 ACL および名前付き拡張 ACL も作成できます。標準 IP ACL の名前は 1 ～ 99 で、拡張 IP ACL の名前は 100 ～ 199 です。番号付きリストの代わりに名前付き ACL を使用することには、エントリを個別に削除できるという利点があります。

番号付き標準 IPv4 ACL

ACL を作成するときには、ACL の末尾にデフォルトで暗黙的な deny ステートメントが追加され、ACL の終わりに到達するまで一致する条件が見つからなかったすべてのパケットに適用されることに注意してください。標準アクセス リストでは、関連付けられた IP ホスト アドレス ACL の指定からマスクを省略すると、0.0.0.0 がマスクと見なされます。

スイッチは、**host** 一致条件があるエントリと *don't care* マスク 0.0.0.0 を含む一致条件があるエントリがリストの先頭に移動し、0 以外の *don't care* マスクを含むエントリよりも前に位置するように、標準アクセス リストの順序を書き換えます。そのため、**show** コマンドの出力やコンフィギュレーション ファイルでは、ACE が必ずしも入力されたとおりの順序で配置されません。

番号付き拡張 IPv4 ACL

標準 ACL では照合に送信元アドレスだけを使用しますが、拡張 ACL では、照合操作に送信元アドレスと宛先アドレスを使用でき、任意でプロトコル タイプ情報を使用して制御のきめ細かさを高めることができます。番号付き拡張アクセス リストの ACE を作成するときには、作成した ACE がリストの末尾に追加されることに注意してください。番号付きリストでは、ACE の順序を変更したり、リスト内の特定の場所に対して ACE を追加または削除したりできません。

このスイッチは、ダイナミックまたは再帰アクセスリストをサポートしていません。また、タイプ オブ サービス（ToS）の minimize-monetary-cost ビットに基づくフィルタリングもサポートしていません。

一部のプロトコルには、特定のパラメータやキーワードも適用されます。

拡張 TCP、UDP、ICMP、IGMP、またはその他の IP ACL を定義できます。また、このスイッチはこれらの IP プロトコルをサポートします。

（注） ICMP エコー応答はフィルタリングできません。他の ICMP コードまたはタイプは、すべてフィルタリングできます。

これらの IP プロトコルがサポートされます。

- 認証ヘッダー プロトコル（**ahp**）
- カプセル化セキュリティペイロード（**esp**）
- Enhanced Interior Gateway Routing Protocol（**eigrp**）
- 総称ルーティング カプセル化（**gre**）
- インターネット制御メッセージ プロトコル（**icmp**）
- インターネット グループ管理プロトコル（**igmp**）
- すべての内部プロトコル（**ip**）
- IP-in-IP トンネリング（**ipinip**）
- KA9Q NOS 互換 IP over IP トンネリング（**nos**）
- Open Shortest Path First ルーティング（**ospf**）
- ペイロード圧縮プロトコル（**pcp**）
- プロトコル独立マルチキャスト（**pim**）
- 伝送制御プロトコル（**tcp**）
- ユーザ データグラム プロトコル（**udp**）

名前付き IPv4 ACL

IPv4 ACL を識別する手段として、番号ではなく英数字のストリング（名前）を使用できます。名前付き ACL を使用すると、ルータ上で番号付きアクセス リストの場合より多くの IPv4 アクセス リストを設定できます。アクセス リストの識別手段として名前を使用する場合のモードとコマンド構文は、番号を使用する場合とは多少異なります。ただし、IP アクセス リストを使用するすべてのコマンドを名前付きアクセス リストで使用できるわけではありません。

（注） 標準 ACL または拡張 ACL に指定する名前は、アクセス リスト番号のサポートされる範囲内の番号にすることもできます。つまり、標準の IP ACL の名前は 1～99 を指定できます。番号付きリストの代わりに名前付き ACL を使用することには、エントリを個別に削除できるという利点があります。

名前付き ACL を設定するときには、次の注意事項に留意してください。

- また、番号付き ACL も使用できます。

- 標準 ACL と拡張 ACL に同じ名前は使用できません。

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

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。

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 アクセス リスト エントリ シーケンス番号の利点

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（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。
- 分散サポートが提供されます。ルート プロセッサ（RP）とライン カードにあるエントリのシーケンス番号は、常に同期されます。
- 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。

ACL へのコメントの挿入

remark キーワードを使用すると、任意の IP 標準または拡張 ACL にエントリに関するコメント（注釈）を追加できます。コメントを使用すると、ACL の理解とスキャンが容易になります。1 つのコメント行の最大長は 100 文字です。

コメントは、permit ステートメントまたは deny ステートメントの前後どちらにでも配置できます。コメントがどの permit ステートメントまたは deny ステートメントの説明であるのかが明確になるように、コメントの位置に関して一貫性を保つ必要があります。たとえば、あるコメントは対応する permit または deny ステートメントの前にあり、他のコメントは対応するステートメントの後ろにあると、混乱を招く可能性があります。

番号付き IP 標準または拡張 ACL にコメントを挿入するには、**access-list** *access-list number* **remark** *remark* グローバル コンフィギュレーション コマンドを使用します。コメントを削除するには、このコマンドの **no** 形式を使用します。

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
```

ハードウェアおよびソフトウェアによる IP ACL の処理

ACL の処理は、ハードウェアの側で実行されます。ハードウェアで ACL の設定を保存する容量が不足すると、パケットはCPUに送られ、ACLの処理はソフトウェア側で行われます。ACL をソフトウェアで処理するためにデータ パケットが転送される場合、転送速度はレート制限により、ライン レートよりもかなり低下します。

（注） スイッチのリソース不足が原因でハードウェアに ACL を設定できない場合、影響を受けるのは、スイッチに着信した該当 VLAN 内のトラフィックだけです。パケットのソフトウェア転送が発生すると、消費される CPU サイクル数に応じて、スイッチのパフォーマンスが低下することがあります。

トラフィック フローのロギングと転送の両方を行う場合、転送はハードウェアで処理されますが、ロギングはソフトウェアで処理する必要があります。ハードウェアとソフトウェアではパケット処理能力が異なるため、ロギング中であるすべてのフロー（許可フローと拒否フロー）の合計帯域幅が非常に大きい場合は、転送されたパケットの一部をロギングできません。

show ip access-lists 特権 EXEC コマンドを入力した場合、表示される一致カウントには、ハードウェアでアクセスが制御されるパケットは含まれません。ACL の機能は、次のとおりです。

- 標準 ACL および拡張 ACL（入力および出力）の許可アクションや拒否アクションをハードウェアで制御し、アクセス コントロールのセキュリティを強化します。
- *ip unreachables* が無効の場合、**log**を指定しないと、セキュリティ ACL の *deny* ステートメントと一致するフローがハードウェアによってドロップされます。許可ステートメントと一致するフローは、ハードウェアでスイッチングされます。
- ACL の ACE に **log** キーワードを追加すると、パケットのコピーが CPU に送信され、ロギングだけが行われます。ACE が *permit* ステートメントの場合も、パケットはハードウェアでスイッチングされます。

ACL の時間範囲

time-range グローバル コンフィギュレーション コマンドを使用することによって、時刻および曜日に基づいて拡張 ACL を選択的に適用できます。まず、時間範囲の名前を定義し、その時間範囲内の時刻および日付または曜日を設定します。次に、ACL を適用してアクセス リストに制限を設定するときに時間範囲を入力します。時間範囲を使用すると、ACL の permit ステートメントまたは deny ステートメントの有効期間（指定期間内や指定曜日など）を定義できます。**time-range** キーワードおよび引数については、名前付きおよび番号付き拡張 ACL タスクの表を参照してください。

時間範囲を使用するいくつかの利点を次に示します。

- アプリケーションなどのリソース（IP アドレスとマスクのペア、およびポート番号で識別）へのユーザ アクセスをより厳密に許可または拒否できます。
- ログ メッセージを制御できます。ACL エントリを使用して特定の時刻に関してのみトラフィックをロギングできるため、ピーク時間に生成される多数のログを分析しなくても、簡単にアクセスを拒否できます。

時間ベースのアクセス リストを使用すると、CPU に負荷が生じます。これは、アクセス リストの新規設定を他の機能や、ハードウェア メモリにロードされた結合済みの設定とマージする

必要があるためです。そのため、複数のアクセスリストが短期間に連続して（互いに数分以内に）有効となるような設定とならないように注意する必要があります。

（注） 時間範囲は、スイッチのシステム クロックに基づきます。したがって、信頼できるクロック ソースが必要です。ネットワーク タイム プロトコル（NTP）を使用してスイッチ クロックを同期させることを推奨します。

IPv4 ACL のインターフェイスに関する注意事項

着信 ACL の場合、パケットの受信後スイッチはパケットを ACL と照合します。ACL がパケットを許可する場合、スイッチはパケットの処理を継続します。ACL がパケットを拒否する場合、スイッチはパケットを廃棄します。

発信 ACL の場合、パケットを受信し制御対象インターフェイスにルーティングしたあと、スイッチはパケットを ACL と照合します。ACL がパケットを許可した場合は、スイッチはパケットを送信します。ACL がパケットを拒否する場合、スイッチはパケットを廃棄します。

未定義の ACL をインターフェイスに適用すると、スイッチは ACL がインターフェイスに適用されていないと判断し、すべてのパケットを許可します。ネットワーク セキュリティのために未定義の ACL を使用する場合は、このような結果が生じることに注意してください。

インターフェイスへのアクセス コントロール リストの適用

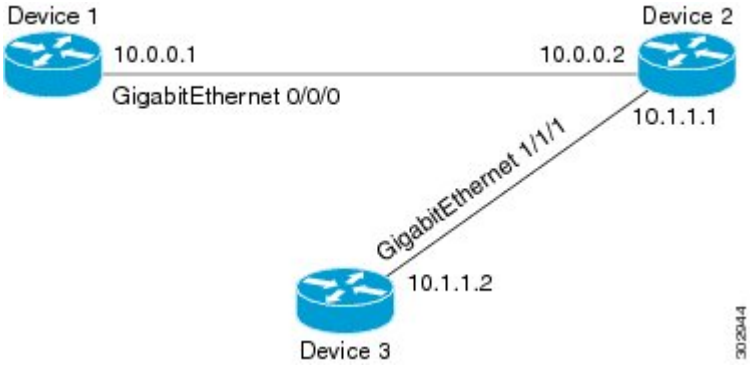
一部のプロトコルでは、インターフェイスに最大 2 つのアクセス リスト（インバウンド アクセス リスト 1 つとアウトバウンド アクセス リスト 1 つ）を適用できます。そのほか、インバウンドとアウトバウンドの両方のパケットをチェックする 1 つのアクセスリストのみを適用するプロトコルもあります。

インバウンド アクセス リストの場合、デバイスがパケットを受信すると、シスコソフトウェアはアクセスリストの条件ステートメントをチェックして一致がないか確認します。パケットが許可されると、ソフトウェアはパケットの処理を継続します。パケットが拒否されると、ソフトウェアはパケットを廃棄します。

アウトバウンド アクセス リストの場合、シスコソフトウェアは、パケットの受信およびアウトバウンド インターフェイスへのルーティング後に、アクセス リストの条件ステートメントをチェックして一致がないか確認します。パケットが許可されると、ソフトウェアはパケットを送信します。パケットが拒否されると、ソフトウェアはパケットを廃棄します。

（注） デバイスのインターフェイスに適用されるアクセスリストは、そのデバイスから送信されたトラフィックにはフィルタ処理を行いません。

図 *1:* アクセス コントロール リスト適用のためのトポロジ



上の図では、デバイス 2 が、デバイス 1 とデバイス 3 に接続されるバイパス デバイスになっています。デバイス 1 のギガビット イーサネット インターフェイス 0/0/0 にアウトバウンド アクセス リストが適用されています。デバイス 1 からデバイス 3 を ping すると、トラフィックがローカルに生成されるため、アクセス リストは発信されるパケットのチェックを行いません。

ローカルに生成されたパケット（常にアウトバウンド）ではアクセス リストによるチェックがバイパスされます。

デフォルトでは、ローカルに生成されたトラフィックのマッチングのためにアウトバウンドインターフェイスに適用されるアクセス リストは、アウトバウンド アクセス リストのチェックをバイパスします。しかし、中継トラフィックはアウトバウンド アクセス リストのチェックを受けます。

（注） 上で説明した動作は、シスコソフトウェアを実行するすべての単一 CPU プラットフォームに適用されます。

ACL ロギング

標準 IP アクセス リストによって許可または拒否されたパケットに関するログ メッセージが、スイッチのソフトウェアによって表示されます。つまり、ACL と一致するパケットがあった場合は、そのパケットに関するログ通知メッセージがコンソールに送信されます。コンソールに表示されるメッセージのレベルは、syslog メッセージを管理する **logging console** コマンドで管理されます。

（注） ルーティングはハードウェアで、ロギングはソフトウェアで実行されます。したがって、**log** キーワードを含む許可（*permit*）または拒否（*deny*）ACE と一致するパケットが多数存在する場合、ソフトウェアはハードウェアの処理速度に追いつくことができないため、一部のパケットはロギングされない場合があります。

ACL を起動した最初のパケットについては、ログ メッセージがすぐに表示されますが、それ以降のパケットについては、5 分間の収集時間が経過してから表示またはロギングされます。ログ メッセージにはアクセス リスト番号、パケットの許可または拒否に関する状況、パケットの送信元 IP アドレス、および直前の 5 分間に許可または拒否された送信元からのパケット数が示されます。

（注） ロギング 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。

ACL の設定方法

この項では、ACL の設定方法について説明します。

IPv4 ACL の設定

スイッチで IP ACL を使用するには、次の手順に従います。

手順

ステップ 1 アクセス リストの番号または名前とアクセス条件を指定して、ACL を作成します。
ステップ 2 ACL をインターフェイスに適用します。

番号付き標準 ACL の作成（CLI）

番号付き標準 ACL を作成するには、次の手順に従います。

手順

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

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard* [**log**] 例： `Device(config)# access-list 2 deny your_host`	送信元アドレスとワイルドカードを使用して標準 IPv4 アクセス リストを定義します。 *access-list-number* には、1 ～ 99 または 1300 ～ 1999 の 10 進数を指定します。 条件が一致した場合にアクセスを拒否する場合は **deny** を指定し、許可する場合は **permit** を指定します。 *source* には、パケットの送信元となるネットワークまたはホストのアドレスを次の形式で指定します。 • ドット付き 10 進表記による 32 ビット長の値。 • キーワード **any** は 0.0.0.0 255.255.255.255 という *source* および *source-wildcard* の省略形です。source-wildcard を入力する必要はありません。 • キーワード **host** は送信元および *source* 0.0.0.0 の *source-wildcard* の省略形です。 （任意）*source-wildcard* は、ワイルドカード ビットを送信元アドレスに適用します。 （任意）**log** を指定すると、エントリと一致するパケットに関するログ通知メッセージがコンソールに送信されます。 （注） ロギングは、レイヤ 3 インターフェイスに割り当てられた ACL でだけサポートされます。

	コマンドまたはアクション	目的
ステップ 4	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例： `Device# show running-config`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

番号付き拡張 ACL の作成（CLI）

番号付き拡張 ACL を作成するには、次の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを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* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] 例：	拡張 IPv4 アクセス リストおよびアクセス条件を定義します。 *access-list-number* には、100 ～ 199 または 2000 ～ 2699 の 10 進数を指定します。

	コマンドまたはアクション	目的
	`Device(config)# access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log`	条件が一致した場合にパケットを拒否する場合は **deny** を指定し、許可する場合は **permit** を指定します。 *protocol* には、インターネット プロトコルの名前または番号を入力します。**ahp**、**eigrp**、**esp**、**gre**、**icmp**、**igmp**、**igrp**、**ip**、**ipinip**、**nos**、**ospf**、**pcp**、**pim**、**tcp**、**udp**、または IP プロトコル番号を表す 0 ～ 255 の整数を使用できます。一致条件としてインターネット プロトコル（ICMP、TCP、UDP など）を指定するには、キーワード **ip** を使用します。 （注） この手順には、ほとんどの IP プロトコルのオプションが含まれています。TCP、UDP、ICMP、および IGMP の追加の特定パラメータについては、次のステップを参照してください。 *source* には、パラメータの送信元であるネットワークまたはホストの番号を指定します。 *source-wildcard* は、ワイルドカード ビットを送信元アドレスに適用します。 *destination* には、パラメータの宛先であるネットワークまたはホストの番号を指定します。 *destination-wildcard* は、ワイルドカード ビットを宛先アドレスに適用します。 source、source-wildcard、destination、および destination-wildcard の値は、次の形式で指定します。 • ドット付き 10 進表記による 32 ビット長の値。 • 0.0.0.0 255.255.255.255（任意のホスト）を表すキーワード **any**。 • 単一のホスト 0.0.0.0 を表すキーワード **host**。

	コマンドまたはアクション	目的
		その他のキーワードはオプションであり、次の意味を持ちます。 • **precedence**：パケットを 0 ～ 7 の番号または名前で指定する優先度と一致させる場合に入力します。指定できる値は、**routine**（0）、**priority**（1）、**immediate**（2）、**flash**（3）、**flash-override**（4）、**critical**（5）、**internet**（6）、**network**（7）です。 • **fragments**：2 つ目以降のフラグメントをチェックする場合に入力します。 • **tos**：パケットを 0 ～ 15 の番号または名前で指定するサービス タイプ レベルと一致させる場合に入力します。指定できる値は、**normal**（0）、**max-reliability**（2）、**max-throughput**（4）、**min-delay**（8）です。 • **time-range**：時間範囲の名前を指定します。 • **dscp**：パケットを 0 ～ 63 の番号で指定する DSCP 値と一致させる場合に入力します。または、指定できる値のリストを表示するには、疑問符（?）を使用します。 （注） **dscp** 値を入力する場合は、**tos** または **precedence** を入力できません。**dscp** を入力せずに **tos** と **precedence** の両方の値を入力できます。
ステップ 4	**access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*]	拡張 TCP アクセス リストおよびアクセス条件を定義します。 次に示す例外を除き、拡張 IPv4 ACL に対して説明するパラメータと同じパラメータを使用します。

	コマンドまたはアクション	目的
	例： `Device(config)# access-list 101 permit tcp any any eq 500`	（任意）*operator* および *port* を入力すると、送信元ポート（*source source-wildcard* の後に入力した場合）または宛先ポート（*destination destination-wildcard* の後に入力した場合）が比較されます。演算子の候補には、**eq**（次の値に等しい）、**gt**（次の値より大きい）、**lt**（次の値より小さい）、**neq**（次の値に等しくない）、および **range**（次の範囲）があります。演算子にはポート番号を指定する必要があります（range の場合は 2 つのポート番号をスペースで区切って指定する必要があります）。 *port* には、10 進数（0 ～ 65535）のポート番号または TCP ポート名を入力します。TCP をフィルタリングするときには、TCP ポートの番号または名前だけを使用します。 他のオプションのキーワードの意味は次のとおりです。 • *flag*：指定された TCP ヘッダー ビットを基準にして照合します。入力できるフラグは、**ack**（確認応答）、**fin**（終了）、**psh**（プッシュ）、**rst**（リセット）、**syn**（同期）、または **urg**（緊急）です。
ステップ 5	**access-list** *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] 例： `Device(config)# access-list 101 permit udp any any eq 100`	（任意）拡張 UDP アクセス リストおよびアクセス条件を定義します。 UDP パラメータは TCP の説明にあるパラメータと同じです。ただし、[operator [port]] ポートの番号または名前は、UDP ポートの番号または名前でなければなりません。また、UDP では、**flag** キーワードは無効です。
ステップ 6	**access-list** *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type*	拡張 ICMP アクセス リストおよびアクセス条件を定義します。

	コマンドまたはアクション	目的
	| [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] 例： `Device(config)# access-list 101 permit icmp any any 200`	ICMP パラメータは拡張 IPv4 ACL の IP プロトコルの説明にあるパラメータとほとんど同じですが、ICMP メッセージ タイプおよびコード パラメータが追加されています。オプションのキーワードの意味は次のとおりです。 • *icmp-type*：ICMP メッセージ タイプでフィルタリングする場合に入力します。指定できる値の範囲は、0 ～ 255 です。 • *icmp-code*：ICMP パケットを ICMP メッセージ コード タイプでフィルタリングする場合に入力します。指定できる値の範囲は、0 ～ 255 です。 • *icmp-message*：ICMP パケットを ICMP メッセージ タイプ名または ICMP メッセージ タイプとコード名でフィルタリングする場合に入力します。
ステップ 7	**access-list** *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] 例： `Device(config)# access-list 101 permit igmp any any 14`	（任意）拡張 IGMP アクセス リストおよびアクセス条件を定義します。 IGMP パラメータは拡張 IPv4 ACL の IP プロトコルの説明にあるパラメータとほとんど同じですが、次に示すオプションのパラメータが追加されています。 *igmp-type*IGMP メッセージ タイプと比較するには、0 ～ 15 の番号またはメッセージ名（**dvmrp**、**host-query**、**host-report**、**pim**、または **trace**）を入力します。
ステップ 8	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

名前付き標準 ACL の作成

名前を使用して標準 ACL を作成するには、次の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを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 20`	名前を使用して標準 IPv4 アクセス リストを定義し、アクセス リスト コンフィギュレーション モードを開始します。 名前には、1 ～ 99 の番号を使用できます。

ステップ 4	次のいずれかを使用します。
	• **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
	• **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
	例：
	`Device(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255`
	または
	`Device(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0`
		アクセス リスト コンフィギュレーション モードで、パケットを転送するのかドロップするのかを決定する 1 つ以上の拒否条件または許可条件を指定します。
		• **host** *source*：送信元および送信元ワイルドカードの値である *source* 0.0.0.0。
		• **any**：送信元および送信元ワイルドカードの値である 0.0.0.0 255.255.255.255。

ステップ 5	**end** 例： `Device(config-std-nacl)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例：	入力を確認します。

	コマンドまたはアクション	目的
	`Device# show running-config`	
ステップ 7	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

名前付き拡張 ACL の作成

名前を使用して拡張 ACL を作成するには、次の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip access-list extended** *name* 例： `Device(config)# ip access-list extended 150`	名前を使用して拡張 IPv4 アクセス リストを定義し、アクセス リスト コンフィギュレーション モードを開始します。 名前には、100 ～ 199 の番号を使用できます。
ステップ 4	{**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] 例： `Device(config-ext-nacl)# permit 0 any`	アクセス リスト コンフィギュレーション モードで、許可条件または拒否条件を指定します。**log** キーワードを使用すると、違反を含むアクセス リストのログ メッセージを取得できます。 • **host** *source*：送信元および送信元ワイルドカードの値である *source* 0.0.0.0。

	コマンドまたはアクション	目的
	any	• **host** *destination*：接続先および接続先ワイルドカードの値である *destination* 0.0.0.0。 • **any**：source および source wildcard の値または destination および destination wildcard の値である 0.0.0.0 255.255.255.255
ステップ 5	**end** 例： `Device(config-ext-nacl)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `Device# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

拡張 ACL を作成するときには、ACL の末尾にデフォルトで暗黙的な deny ステートメントが追加され、ACL の終わりに到達するまで一致する条件が見つからなかったすべてのパケットに適用されることに注意してください。標準 ACL では、関連付けられた IP ホスト アドレス アクセス リストの指定からマスクを省略すると、0.0.0.0 がマスクと見なされます。

ACL の作成後に追加したエントリは、リストの末尾に追加されます。ACL エントリを特定の ACL に選択的に追加できません。ただし、**no permit** および **no deny** アクセスリスト コンフィギュレーション モード コマンドを使用すると、名前付き ACL からエントリを削除できます。

番号付き ACL ではなく名前付き ACL を使用する理由の 1 つとして、名前付き ACL では行を選択して削除できることがあります。

次のタスク

作成した名前付き ACL をインターフェイスに適用できます。

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

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。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 resequence** *access-list-name starting-sequence-number increment* 例： `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* 例： `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。 • **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。 • **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

	コマンドまたはアクション	目的
		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。 • *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**] 例： `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。 • 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。 • 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** コマンドシンタックスを使用します。
ステップ 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**] 例： `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。 • 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。 • 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** コマンドシンタックスを使用します。
ステップ 7	次のいずれかを実行します。 • *sequence-number* **permit** *source source-wildcard*	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。

	コマンドまたはアクション	目的
	• *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**] 例： `Device(config-ext-nacl)# 150 permit tcp any any log`	• 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。 • 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シンタックスについては、permit（IP）コマンドを参照してください。 • 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。 • *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**] 例： `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。 • 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。 • 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シンタックスについては、deny（IP）コマンドを参照してください。 • 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** 例： `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。

	コマンドまたはアクション	目的
ステップ 11	**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。

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
```

コメント付き IP ACL エントリの設定

名前付きまたは番号付きアクセス リスト設定を使用します。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

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip access-list** {**standard** | **extended**} {*name* | *number*} 例： `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。

	コマンドまたはアクション	目的
ステップ 4	**remark** 注記 例： `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。 • 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* 例： `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** 例： `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。

ACL の時間範囲の設定

ACL の時間範囲パラメータを設定するには、次の手順に従ってください。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device(config)# 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	**time-range** *time-range-name* 例： `Device(config)# time-range workhours`	作成する時間範囲には意味のある名前（*workhours* など）を割り当て、時間範囲コンフィギュレーション モードを開始します。名前にスペースや疑問符を含めることはできません。また、文字から始める必要があります。
ステップ 4	次のいずれかを使用します。 • **absolute** [**start** *time date*] [**end** *time date*]	適用対象の機能がいつ動作可能になるかを指定します。

	コマンドまたはアクション	目的

	• **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm*
	• **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm*
	例：
	`Device(config-time-range)# absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006`
	または
	`Device(config-time-range)# periodic weekdays 8:00 to 12:00`
		• 時間範囲には、**absolute** ステートメントを1つだけ使用できます。複数の absolute ステートメントを設定した場合は、最後に設定したステートメントだけが実行されます。
		• 複数の **periodic** ステートメントを入力できます。たとえば、平日と週末に異なる時間を設定できます。
		設定例を参照してください。

ステップ 5	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `Device# show running-config`	入力を確認します。
ステップ 7	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

次のタスク

複数の項目をそれぞれ異なる時間に有効にする場合は、上記の手順を繰り返してください。

端末回線への IPv4 ACL の適用

番号付き ACL を使用して、1つまたは複数の端末回線へのアクセスを制御できます。端末回線には名前付き ACL を適用できません。すべての仮想端末回線にユーザが接続する可能性があるため、すべてに同じ制限を設定する必要があります。

仮想端末回線と ACL に指定されたアドレス間の着信接続および発信接続を制限するには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device(config)# 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	**line** [**console** | **vty**] *line-number* 例： `Devices(config)# line console 0`	設定する回線を指定し、インライン コンフィギュレーション モードを開始します。 • **console**：コンソール端末回線を指定します。コンソール ポートは DCE です。 • **vty**：リモート コンソール アクセス用の仮想端末を指定します。 *line-number* は、回線タイプを指定する場合に、設定する連続グループ内で最初の回線番号です。指定できる範囲は 0 ～ 16 です。
ステップ 4	**access-class** *access-list-number* **in** 例： `Device(config-line)# access-class 10 in`	（デバイスへの）特定の仮想端末回線とアクセスリストに指定されたアドレス間の着信接続を制限します。
ステップ 5	**end** 例： `Device(config-line)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show running-config** 例： `Device# show running-config`	入力を確認します。

	コマンドまたはアクション	目的
ステップ 7	**copy running-config startup-config** 例： `Device# copy running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

インターフェイスへの IPv4 ACL の適用（CLI）

ここでは、IPv4 ACL をネットワーク インターフェイスへ適用する方法について説明します。

インターフェイスへのアクセスを制御するには、特権 EXEC モードで次の手順に従います。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**interface** *interface-id* 例： `Device(config)# interface gigabitethernet 1/0/1`	設定するインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。 インターフェイスには、レイヤ 2 インターフェイス（ポート ACL）を指定できます。
ステップ 3	**ip access-group** {*access-list-number* | *name*} {**in**} 例： `Device(config-if)# ip access-group 2 in`	指定されたインターフェイスへのアクセスを制御します。
ステップ 4	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show running-config** 例：	アクセス リストの設定を表示します。

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

IPv4 ACL のモニタリング

スイッチに設定されている ACL、およびインターフェイスに適用済みの ACL を表示することで、IPv4 ACL をモニタできます。

ip access-group インターフェイス コンフィギュレーション コマンドを使用して、レイヤ 2 またはレイヤ 3 インターフェイスに ACL を適用した場合は、そのインターフェイスのアクセス グループを表示できます。また、レイヤ 2 インターフェイスに適用された MAC ACL も表示できます。この情報を表示するには、次の表に記載された特権 EXEC コマンドを使用します。

表 *2:* アクセス リストおよびアクセス グループを表示するコマンド

コマンド	目的
show access-lists [*number* | *name*]	最新の IP および MAC アドレス アクセス リストの全体やその一部、または特定のアクセス リスト（番号付きまたは名前付き）の内容を表示します。
show ip access-lists [*number* | *name*]	最新の IP アクセス リスト全体、または特定の IP アクセス リスト（番号付きまたは名前付き）を表示します。
show ip interface *interface-id*	インターフェイスの詳細設定およびステータスを表示します。IP がイネーブルになっているインターフェイスに、**ip access-group** インターフェイス コンフィギュレーション コマンドを使用して ACL を適用した場合は、アクセス グループも表示に含まれます。
show running-config [**interface** *interface-id*]	スイッチまたは指定されたインターフェイスのコンフィギュレーションファイルの内容（設定されたすべての MAC および IP アクセス リストや、どのアクセス グループがインターフェイスに適用されたかなど）を表示します。

コマンド	目的
show mac access-group [**interface** *interface-id*]	すべてのレイヤ 2 インターフェイスまたは指定されたレイヤ 2 インターフェイスに適用されている MAC アクセス リスト を表示します。

ACL の設定例

この項では、IPv4 ACL の設定例を示します。

例：番号付き ACL

次の例のネットワーク 10.0.0.0 は、2 番目のオクテットがサブネットを指定するクラス A ネットワークです。つまり、サブネットマスクは255.255.0.0 です。ネットワーク 10.0.0.0 アドレスの 3 番目および 4 番目のオクテットで特定のホストを指定します。アクセス リスト 2 を使用して、サブネット 48 のアドレスを 1 つ許可し、同じサブネットの他のアドレスはすべて拒否します。このアクセス リストの最終行は、ネットワーク 10.0.0.0 の他のすべてのサブネット上のアドレスが許可されることを示します。この ACL は、ポートに着信するパケットに適用されます。

```
Device(config)# access-list 2 permit 10.48.0.3
Device(config)# access-list 2 deny 10.48.0.0 0.0.255.255
Device(config)# access-list 2 permit 10.0.0.0 0.255.255.255
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip access-group 2 in
```

例：拡張 ACL

次の例の先頭行は、1023 よりも大きい宛先ポートへの着信 TCP 接続を許可します。2 番めの行は、ホスト 128.88.1.2 の SMTP ポートへの着信 TCP 接続を許可します。3 番めの行は、エラー フィードバック用の着信 ICMP メッセージを許可します。

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Device(config)# access-list 102 permit icmp any any
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip access-group 102 in
```

次の例では、インターネットに接続されたネットワークがあり、そのネットワーク上の任意のホストがインターネット上の任意のホストと TCP 接続を確立できるようにする場合を想定しています。ただし、IP ホストからは、専用メール ホストのメール（SMTP）ポートを除き、ネットワーク上のホストと TCP 接続を確立できないようにします。

SMTP は、接続の一端では TCP ポート 25、もう一端ではランダムなポート番号を使用します。接続している間は、同じポート番号が使用されます。インターネットから着信するメール パケットの宛先ポートは 25 です。安全なネットワークシステムは、ポート 25 で常にメール接続を受け入れます。

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip access-group 102 in
```

次の例では、ネットワークはアドレスが 128.88.0.0 のクラス B ネットワークで、メール ホストのアドレスは 128.88.1.2 です。**ACK** または **RST** キーワードを使用して、ACK または RST ビット セットを照合します。これで、パケットが既存の接続に属していることが判明します。

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 RST
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip access-group 102 in
```

例：名前付き ACL

名前付き標準 ACL および名前付き拡張 ACL の作成

次に、*Internet_filter* という名前の標準 ACL および *marketing_group* という名前の拡張 ACL を作成する例を示します。*Internet_filter* ACL は、送信元アドレス 1.2.3.4 から送信されるすべてのトラフィックを許可します。

```
Device(config)# ip access-list standard Internet_filter
Device(config-ext-nacl)# permit 1.2.3.4
Device(config-ext-nacl)# exit
```

marketing_group ACL は、宛先アドレスとワイルドカードの値 171.69.0.0 0.0.255.255 への任意の TCP Telnet トラフィックを許可し、その他の TCP トラフィックを拒否します。ICMP トラフィックを許可し、任意の送信元から、宛先ポートが 1024 より小さい 171.69.0.0 ～ 179.69.255.255 の宛先アドレスへ送信される UDP トラフィックを拒否します。それ以外のすべての IP トラフィックを拒否して、結果を示すログが表示されます。

```
Device(config)# ip access-list extended marketing_group
Device(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Device(config-ext-nacl)# deny tcp any any
Device(config-ext-nacl)# permit icmp any any
Device(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Device(config-ext-nacl)# deny ip any any log
Device(config-ext-nacl)# exit
```

名前付き ACL からの個別 ACE の削除

次に、名前付きアクセス リスト *border-list* から ACE を個別に削除する例を示します。

```
Device(config)# ip access-list extended border-list
Device(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

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：コメント付き IP ACL エントリの設定

次に示す番号付き ACL の例では、Jones が所有するワークステーションにはアクセスを許可し、Smith が所有するワークステーションにはアクセスを許可しません。

```
Device(config)# access-list 1 remark Permit only Jones workstation through
Device(config)# access-list 1 permit 171.69.2.88
Device(config)# access-list 1 remark Do not allow Smith workstation through
Device(config)# access-list 1 deny 171.69.3.13
```

次に示す番号付き ACL の例では、Winter および Smith のワークステーションに Web 閲覧を許可しません。

```
Device(config)# access-list 100 remark Do not allow Winter to browse the web
Device(config)# access-list 100 deny host 171.69.3.85 any eq www
Device(config)# access-list 100 remark Do not allow Smith to browse the web
Device(config)# access-list 100 deny host 171.69.3.13 any eq www
```

次に示す名前付き ACL の例では、Jones のサブネットにアクセスを許可しません。

```
Device(config)# ip access-list standard prevention
Device(config-std-nacl)# remark Do not allow Jones subnet through
Device(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

次に示す名前付き ACL の例では、Jones のサブネットに発信 Telnet の使用を許可しません。

```
Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Device(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

例：ACL での時間範囲を使用

次の例に、*workhours*（営業時間）の時間範囲および会社の休日（2006 年 1 月 1 日）を設定し、設定を確認する例を示します。

```
Device# show time-range
time-range entry: new_year_day_2003 (inactive)
   absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
   periodic weekdays 8:00 to 12:00
   periodic weekdays 13:00 to 17:00
```

時間範囲を適用するには、時間範囲を実装できる拡張 ACL 内に時間範囲名を入力します。次に、拡張アクセス リスト 188 を作成して確認する例を示します。このアクセス リストでは、定義された休業時間中はすべての送信元からすべての宛先への TCP トラフィックを拒否し、営業時間中はすべての TCP トラフィックを許可します。

```
Device(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Device(config)# access-list 188 permit tcp any any time-range workhours
Device(config)# end
Device# show access-lists
Extended IP access list 188
   10 deny tcp any any time-range new_year_day_2006 (inactive)
   20 permit tcp any any time-range workhours (inactive)
```

次に、名前付き ACL を使用して同じトラフィックを許可および拒否する例を示します。

```
Device(config)# ip access-list extended deny_access
Device(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Device(config-ext-nacl)# exit
Device(config)# ip access-list extended may_access
Device(config-ext-nacl)# permit tcp any any time-range workhours
Device(config-ext-nacl)# end
Device# show ip access-lists
Extended IP access list lpip_default
    10 permit ip any any
Extended IP access list deny_access
    10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
    10 permit tcp any any time-range workhours (inactive)
```

例：IP ACL に適用される時間範囲

次に、月曜日から金曜日の午前 8 時 ～午後 6 時（18 時）の間、IP の HTTP トラフィックを拒否する例を示します。UDP トラフィックは、土曜日および日曜日の正午～午後 8 時（20 時）の間だけ許可されます。

```
Device(config)# time-range no-http
Device(config)# periodic weekdays 8:00 to 18:00
!
Device(config)# time-range udp-yes
Device(config)# periodic weekend 12:00 to 20:00
!
Device(config)# ip access-list extended strict
Device(config-ext-nacl)# deny tcp any any eq www time-range no-http
Device(config-ext-nacl)# permit udp any any time-range udp-yes
!
Device(config-ext-nacl)# exit
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip access-group strict in
```

例：ACL ロギング

ACL では 2 種類のロギングがサポートされています。**log** キーワードを指定すると、エントリと一致するパケットに関するログ通知メッセージがコンソールに送信されます。**log-input** キーワードを指定すると、ログ エントリに入力インターフェイスが追加されます。

次の例では、名前付き標準アクセス リスト *stan1* は 10.1.1.0 0.0.0.255 からのトラフィックを拒否し、その他のすべての送信元からのトラフィックを許可します。**log** キーワードも指定されています。

```
Device(config)# ip access-list standard stan1
Device(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Device(config-std-nacl)# permit any log
Device(config-std-nacl)# exit
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip access-group stan1 in
Device(config-if)# end
Device# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 37 messages logged
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 37 messages logged
    File logging: disabled
    Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

次に、名前付き拡張アクセス リスト *ext1* によって、任意の送信元から 10.1.1.0 0.0.0.255 への ICMP パケットを許可し、すべての UDP パケットを拒否する例を示します。

```
Device(config)# ip access-list extended ext1
Device(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Device(config-ext-nacl)# deny udp any any log
Device(config-std-nacl)# exit
Device(config)# interface gigabitethernet 1/0/2
Device(config-if)# ip access-group ext1 in
```

次に、拡張 ACL のログの例を示します。

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1
packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8
packets
```

IP ACL のすべてのロギング エントリは %SEC-6-IPACCESSLOG で開始します。エントリの形式は、一致した ACL やアクセス エントリの種類に応じて若干異なります。

次に、**log-input** キーワードを指定した場合の出力メッセージの例を示します。

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1 0001.42ef.a400)
 ->
10.1.1.61 (0/0), 1 packet
```

log キーワードを指定した場合、同様のパケットに関するログ メッセージには入力インターフェイス情報が含まれません。

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0),
 1 packet
```

例：ACL のトラブルシューティング

次の ACL マネージャ メッセージが表示されて [chars] がアクセス リスト名である場合は次のようにします。

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

スイッチには、ACL のハードウェア表現を作成するのに使用可能なリソースが不足しています。このリソースには、ハードウェア メモリおよびラベル スペースが含まれますが、CPU メモリは含まれません。この問題の原因は、使用可能な論理演算ユニットまたは専用のハードウェア リソースの不足です。論理演算ユニットは、TCP フラグの一致、または TCP、UDP、SCTP ポート番号での **eq**（**ne**、**gt**、**lt**、または **range**）のテストで必要です。

次のいずれかの回避策を使用します。

- ACL の設定を変更して使用するリソースを減らします。
- 他の ACL 名または番号よりも英数字順で先に表示される名前または番号に ACL の名前を変更します。

十分なリソースがない ACL の設定の詳細については、Bug Toolkit の CSCsq63926 を参照してください。

たとえば、次の ACL をインターフェイスに適用します。

```
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
permit tcp source source-wildcard destination destination-wildcard
```

なおかつ次のメッセージが表示される場合は次のようにします。

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

フラグ関連の演算子は使用できません。この問題を回避するには、

- **ip access-list resequence** グローバル コンフィギュレーション コマンドを使用することによって、4 つ目の ACE を 1 つ目の ACE の前に移動させます。

  ```
  permit tcp source source-wildcard destination destination-wildcard
  permit tcp source source-wildcard destination destination-wildcard range 5 60
  permit tcp source source-wildcard destination destination-wildcard range 15 160
  permit tcp source source-wildcard destination destination-wildcard range 115 1660
  ```

  または

- 他の ACL 名または番号よりも英数字順で先に表示される名前または番号に ACL の名前を変更します（たとえば、ACL *79* を ACL *1* に変更します）。

これで、ACL 内の 1 つ目の ACE をインターフェイスに適用できます。スイッチによって、ACE が、Opselect インデックス内の利用可能なマッピング ビットに割り当てられ、次に、ハードウェア メモリ内の同じビットを使用するフラグ関連の演算子が割り当てられます。

IPv4 アクセスコントロールリストに関する追加情報

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	Consolidated Platform Command Reference, Cisco IOS Release 15.2(7)Ex (Catalyst 1000 Switches)

関連項目	マニュアル タイトル
ACL	詳細については、以下を参照してください。 • 『Security Configuration Guide』の「Access Control Lists Overview」 • 『Security Configuration Guide』の「Configuring IPv6 Access Control Lists」

IPv4 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	変更内容
IPv4 アクセスコントロールリスト 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 IP アクセス リスト エントリ シーケンス番号	Cisco IOS Release 15.2(7)E1	この機能が導入されました。