IP ACL の設定

この章では、Cisco NX-OS デバイスの IP アクセス コントロール リスト（ACL）を設定する方法について説明します。

特に指定がなければ、IP ACL は IPv4 および IPv6 の ACL を意味します。

この章は、次の項で構成されています。

ACL について

ACL とは、トラフィックのフィルタリングに使用する順序付きのルール セットのことです。 各ルールには、パケットがルールに一致するために満たさなければならない条件のセットが規定されています。 デバイスは、ある ACL がパケットに適用されると判断すると、そのすべてのルールの条件にパケットを照合し、テストします。 最初に一致したルールで、そのパケットが許可されるか拒否されるかが決定されます。 一致するものがなければ、デバイスは適用可能な暗黙の

ルールを適用します。 デバイスは、許可されたパケットの処理を続行し、拒否されたパケットはドロップします。

ACL を使用すると、ネットワークおよび特定のホストを、不要なトラフィックや望ましくないトラフィックから保護できます。 たとえば、ACL を使用して、厳重にセキュリティ保護されたネットワークからインターネットにハイパー テキスト トランスファ プロトコル（HTTP）トラフィックが流入するのを禁止できます。 また、特定のサイトへの HTTP トラフィックだけを許可することもできます。その場合は、サイトの IP アドレスが、IP ACL に指定されているかどうかによって判定します。

ACL のタイプと適用

セキュリティ トラフィック フィルタリングには次のタイプの ACL を使用できます。

IPv4 ACL

IPv4 トラフィックだけに適用されます。

IPv6 ACL

IPv6 トラフィックだけに適用されます。

IP ACL には次のタイプの適用例があります。

ルータ ACL

レイヤ 3 トラフィックのフィルタリング

VTY ACL

仮想テレタイプ（VTY）トラフィックのフィルタリング

次の表に、セキュリティ ACL の適用例の概要を示します。

表 *1*：セキュリティ *ACL* の適用

アプリケーション	サポートされるインターフェイス	サポートする ACL のタイプ
ルータ ACL	• 物理層 3 インターフェイス • レイヤ 3 イーサネット サブインターフェイス • レイヤ 3 イーサネット ポート チャネル インターフェイス • 管理インターフェイス	• IPv4 ACL • IPv6 ACL

アプリケーション	サポートされるインターフェイス	サポートする ACL のタイプ
VTY ACL	• VTY	• IPv4 ACL • IPv6 ACL

ACL の適用順序

デバイスは、パケットを処理する際に、そのパケットの転送パスを決定します。 デバイスがトラフィックに適用する ACL はパスによって決まります。 デバイスは、次の順序で ACL を適用します。

1. 入力ルータ ACL
2. 入力 VTY ACL
3. 出力 VTY ACL
4. 出力ルータ ACL

ルールについて

ACLによるネットワーク トラフィックのフィルタリング方法を設定する際に、何を作成、変更、削除するかを決めるのがルールです。 ルールは実行コンフィギュレーション内に表示されます。 ACL をインターフェイスに適用するか、またはインターフェイスにすでに適用されている ACL 内のルールを変更すると、スーパーバイザ モジュールは実行コンフィギュレーション内のルールから ACL のエントリを作成し、それらの ACL エントリを適用可能な I/O モジュールに送信します。 ACL の設定によっては、ルールよりも ACL エントリの方が数が多くなることがあります。

アクセス リスト コンフィギュレーション モードでルールを作成するには、**permit** または **deny** コマンドを使用します。 これにより、デバイスは許可ルール内の基準と一致するトラフィックを許可し、拒否ルール内の基準と一致するトラフィックをブロックします。 ルールに一致するためにトラフィックが満たさなければならない基準を設定するためのオプションが多数用意されています。

ここでは、ルールを設定する際に使用できるオプションをいくつか紹介します。

IP ACL のプロトコル

IPv4および IPv6 の ACL では、トラフィックをプロトコルで識別できます。 指定の際の手間を省くために、一部のプロトコルは名前で指定できます。 たとえば、IPv4 または IPv6 の ACL では、ICMP を名前で指定できます。

プロトコルはすべて番号で指定できます。

IPv4 および IPv6 ACL では、インターネット プロトコル番号を表す整数でプロトコルを指定できます。

送信元と宛先

各ルールには、ルールに一致するトラフィックの送信元と宛先を指定します。 指定する送信元および宛先には、特定のホスト、ホストのネットワークまたはグループ、あるいは任意のホストを使用できます。 送信元と宛先の指定方法は、IPv4 ACL、IPv6 のうち、どれを設定するかによって異なります。

IP ACL の暗黙ルール

IP ACL には暗黙ルールがあります。暗黙ルールは、実行コンフィギュレーションには設定されていませんが、ACL 内の他のルールと一致しない場合にデバイスがトラフィックに適用するルールです。 ACL のルール単位の統計情報を維持するようにデバイスを設定した場合、暗黙ルールの統計情報はデバイスに維持されません。

すべての IPv4 ACL には、次の暗黙のルールがあります。

```
deny ip any any
```

この暗黙ルールによって、デバイスは不一致 IP トラフィックを確実に拒否します。

すべての IPv6 ACL には、次の暗黙のルールがあります。

```
deny ipv6 any any
```

この暗黙ルールによって、デバイスは不一致 IPv6 トラフィックを確実に拒否します。

（注） IPv6 の nd-na、nd-ns、router-advertisement、router-solicitation パケットは、IPv6 ACL の暗黙の許可ルールとしては使用できません。 明示的に許可するには、次の規則を追加する必要があります。

- **permit icmp any any nd-na**
- **permit icmp any any nd-ns**
- **permit icmp any any router-advertisement**
- **permit icmp any any router-solicitation**

この暗黙ルールによって、デバイスは、不一致トラフィックを確実に拒否します。

その他のフィルタリング オプション

追加のオプションを使用してトラフィックを識別できます。これらのオプションは、ACL のタイプによって異なります。 次のリストには、ほとんどの追加フィルタリング オプションが含まれていますが、すべてを網羅しているわけではありません。

- IPv4 ACL には、次の追加フィルタリング オプションが用意されています。
  - レイヤ 4 プロトコル
  - TCP/UDP ポート
  - ICMP タイプおよびコード
  - IGMP タイプ
  - 優先レベル
  - DiffServ コード ポイント（DSCP）値
  - ACK、FIN、PSH、RST、SYN、または URG ビットがセットされた TCP パケット
  - 確立済み TCP 接続
- IPv6 ACL では、次のフィルタリング オプションが追加されています。
  - レイヤ 4 プロトコル
  - カプセル化セキュリティ ペイロード
  - ペイロード圧縮プロトコル
  - ストリーム制御転送プロトコル（SCTP）
  - SCTP、TCP、および UDP の各ポート
  - ICMP タイプおよびコード
  - DSCP 値
  - ACK、FIN、PSH、RST、SYN、または URG ビットがセットされた TCP パケット
  - 確立済み TCP 接続
  - パケット長

シーケンス番号

デバイスはルールのシーケンス番号をサポートしています。 入力するすべてのルールにシーケンス番号が割り当てられます（ユーザによる割り当てまたはデバイスによる自動割り当て）。 シーケンス番号によって、次の ACL 設定作業が容易になります。

既存のルールの間に新しいルールを追加

シーケンス番号を指定することによって、ACL内での新規ルールの挿入場所を指定します。 たとえば、ルール番号 100 と 110 の間に新しいルールを挿入する必要がある場合は、シーケンス番号 105 を新しいルールに割り当てます。

ルールの削除

シーケンス番号を使用しない場合は、ルールを削除するために、次のようにルール全体を入力する必要があります。

```
switch(config-acl)# no permit tcp 10.0.0.0/8 any
```

このルールに 101 番のシーケンス番号が付いていれば、次のコマンドだけでルールを削除できます。

```
switch(config-acl)# no 101
```

ルールの移動

シーケンス番号を使用すれば、同じ ACL 内の異なる場所にルールを移動する必要がある場合に、そのルールのコピーをシーケンス番号で正しい位置に挿入してから、元のルールを削除できます。 この方法により、トラフィックを中断せずにルールを移動できます。

シーケンス番号を使用せずにルールを入力すると、デバイスはそのルールを ACL の最後に追加し、そのルールの直前のルールのシーケンス番号よりも 10 大きい番号を割り当てます。 たとえば、ACL 内の最後のルールのシーケンス番号が 225 で、シーケンス番号を指定せずにルールを追加した場合、デバイスはその新しいルールにシーケンス番号 235 を割り当てます。

また、Cisco NX-OS では、ACL 内ルールのシーケンス番号を再割り当てできます。 シーケンス番号の再割り当ては、ACL 内に、100、101 のように連続するシーケンス番号のルールがある場合、それらのルールの間に 1 つ以上のルールを挿入する必要があるときに便利です。

論理演算子と論理演算ユニット

TCP および UDP トラフィックの IP ACL ルールでは、論理演算子を使用して、ポート番号に基づきトラフィックをフィルタリングできます。 Cisco NX-OS では、入力方向でのみ論理演算子をサポートします。

このデバイスは、論理演算ユニット（LOU）というレジスタに、演算子とオペランドの組み合わせを格納します。 各タイプの演算子は、次のように LOU を使用します。

eq

LOU には格納されません。

gt

1 LOU を使用します。

lt

1 LOU を使用します。

neq

1 LOU を使用します。

range

1 LOU を使用します。

ロギング

ルールに一致するパケットに関する情報ログ メッセージの作成をイネーブルにできます。 ログ メッセージには、パケットについての次の情報が含まれます。

- プロトコル
- TCP、UDP、またはICMPのいずれのパケットか、あるいは、番号が付けられただけのパケットか
- 送信元と宛先のアドレス
- 送信元と宛先のポート番号（該当する場合）

時間範囲

時間範囲を使用して、ACLルールが有効になる時期を制御できます。 たとえば、インターフェイスに着信するトラフィックに特定の ACL を適用するとデバイスが判断し、その ACL のあるルールの時間範囲が有効になっていない場合、デバイスは、トラフィックをそのルールと照合しません。 デバイスは、そのデバイスのクロックに基づいて時間範囲を評価します。

時間範囲を使用する ACL を適用すると、デバイスはその ACL で参照される時間範囲の開始時または終了時に影響する I/O モジュールをアップデートします。 時間範囲によって開始されるアップデートはベストエフォート型のプライオリティで実行されます。 時間範囲によってアップデートが生じたときにデバイスの処理負荷が非常に高い場合、デバイスはアップデートを最大数秒間遅らせることがあります。

IPv4、IPv6 の各 ACL は時間範囲をサポートします。 デバイスがトラフィックに ACL を適用する場合、有効なルールは次のとおりです。

- 時間範囲が指定されていないすべてのルール
- デバイスがその ACL をトラフィックに適用した時点（秒）が時間範囲に含まれているルール

名前が付けられた時間範囲は再利用できます。多くのACLルールを設定する場合は、時間範囲を名前で一度設定すれば済みます。 時間範囲の名前は最大 64 の英文字で指定します。

時間範囲には、1 つまたは複数のルールで構成されます。 これらのルールは次の 2 種類に分類できます。

絶対

特定の開始日時、終了日時、その両方を持つルール、またはそのどちらも持たないルール。絶対時間範囲のルールがアクティブかどうかは、開始日時または終了日時の有無によって、次のように決まります。

- 開始日時と終了日時が両方指定されている：この時間範囲ルールは、現在の時刻が開始日時よりも後で終了日時よりも前の場合にアクティブになります。
- 開始日時が指定され、終了日時は指定されていない：この時間範囲ルールは、現在の時刻が開始日時よりも後である場合にアクティブになります。
- 開始日時は指定されず、終了日時が指定されている：この時間範囲ルールは、現在の時刻が終了日時よりも前である場合にアクティブになります。
- 開始日時も終了日時も指定されていない：この時間範囲ルールは常にアクティブです。

たとえば、新しいサブネットへのアクセスを許可するようにネットワークを設定する場合、そのサブネットをオンラインにする予定日の真夜中からアクセスを許可するような時間範囲を指定し、この時間範囲をそのサブネットに適用するACLルールに使用します。デバイスはこのルールを含むACLを適用する場合、開始日時が過ぎると、この時間範囲を使用するルールの適用を自動的に開始します。

定期

毎週1回以上アクティブになるルール。たとえば、定期時間範囲を使用すると、平日の営業時間中だけ、研究室のサブネットにアクセスできるようにすることができます。デバイスは、そのルールを含むACLが適用されていて、時間範囲がアクティブな場合にだけ、この時間範囲を使用するACLルールを自動的に適用します。

（注） デバイスは、時間範囲内のルールの順序に関係なく、時間範囲がアクティブかどうかを判断します。Cisco NX-OSは、時間範囲を編集できるように時間範囲内にシーケンス番号を入れます。

時間範囲には備考を含めることもできます。備考を使用すると、時間範囲にコメントを挿入できます。備考は、最大100文字の英数字で指定します。

デバイスは次の方法で時間範囲がアクティブかどうかを判断します。

- 時間範囲に絶対ルールが1つまたは複数含まれている：現在の時刻が1つまたは複数の絶対ルールの範囲内であれば、その時間範囲はアクティブです。
- 時間範囲に定期ルールが1つまたは複数含まれている：現在の時刻が1つまたは複数の定期ルールの範囲内であれば、その時間範囲はアクティブです。
- 時間範囲に絶対ルールと定期ルールが両方含まれている：現在の時刻が1つまたは複数の絶対ルールと1つ以上の定期ルールの範囲内にある場合に、その時間範囲はアクティブです。

時間範囲に絶対ルールと定期ルールが両方含まれている場合、定期ルールがアクティブになるのは、最低 1 つの絶対ルールがアクティブな場合だけです。

統計情報と ACL

このデバイスは、IPv4 および IPv6 の ACL に設定した各ルールのグローバル統計を保持できます。1 つの ACL が複数のインターフェイスに適用される場合、ルール統計には、その ACL が適用されるすべてのインターフェイスと一致する（ヒットする）パケットの合計数が維持されます。

（注） インターフェイスレベルの ACL 統計はサポートされていません。

設定する ACL ごとに、その ACL の統計情報をデバイスが維持するかどうかを指定できます。これにより、ACL によるトラフィック フィルタリングが必要かどうかに応じて ACL 統計のオン、オフを指定できます。また、ACL 設定のトラブルシューティングにも役立ちます。

デバイスには ACL の暗黙ルールの統計情報は維持されません。 たとえば、すべての IPv4 ACL の末尾にある暗黙の **deny ip any any** ルールと一致するパケットのカウントはデバイスに維持されません。 暗黙ルールの統計情報を維持する場合は、暗黙ルールと同じルールを指定した ACL を明示的に設定する必要があります。

関連トピック

IP ACL の統計情報のモニタリングとクリア, （29 ページ）
IP ACL の暗黙ルール, （4 ページ）

Atomic ACL のアップデート

デフォルトでは、Cisco Nexus 9000 シリーズのデバイスのスーパーバイザ モジュールで、ACL の変更を I/O モジュールにアップデートする際には、Atomic ACL のアップデートを実行します。Atomic アップデートでは、アップデートされる ACL が適用されるトラフィックを中断させることがありません。しかし、Atomic アップデートでは、ACL のアップデートを受け取る I/O モジュールに、関係する ACL の既存のすべてのエントリに加えて、アップデートされた ACL エントリを保存するのに十分なリソースがあることが必要です。 アップデートが行われた後、アップデートに使用されたリソースは開放されます。 I/O モジュールに十分なリソースがない場合は、デバイスからエラー メッセージが出力され、この I/O モジュールに対する ACL のアップデートは失敗します。

I/O モジュールに Atomic アップデートに必要なリソースがない場合は、**no hardware access-list update atomic** コマンドを使用して Atomic アップデートをディセーブルにすることができますが、デバイスで既存の ACL を削除して、アップデートされた ACL を適用するには、多少の時間がかかります。ACL が適用されるトラフィックは、デフォルトでドロップされます。

ACL が適用されるすべてのトラフィックを許可し、同時に非 Atomic アップデートを受信するようにするには、**hardware access-list update default-result permit** コマンドを使用してください。

次の例では、ACL に対する Atomic アップデートをディセーブルにする方法を示します。

```
switch# config t
switch(config)# no hardware access-list update atomic
```

次の例では、非 Atomic ACL アップデートの際に、関連するトラフィックを許可する方法を示します。

```
switch# config t
switch(config)# hardware access-list update default-result permit
```

次の例では、Atomic アップデート方式に戻る方法を示します。

```
switch# config t
switch(config)# no hardware access-list update default-result permit
switch(config)# hardware access-list update atomic
```

VTY のサポート

Cisco NX-OS では、ACL を VTY ラインに直接適用できませんが、コントロール プレーン ポリシング（CoPP）を使用して VTY トラフィックをフィルタリングできます。 これを行うには、VTY トラフィックのフィルタリングに使用する 2 つの ACL を定義する必要があります。通過させるトラフィックを許可する ACL と、ドロップするトラフィックを許可する ACL です。 その後、適切なトラフィックと一致する ACL によって許可されたパケットを送信し、不適切なトラフィックと一致する ACL によって許可されたパケットをドロップするように CoPP を設定できます。

次の例の ACL copp-system-acl-allow では、10.30.30.0/24 ネットワークから着信する Telnet、SSH、SNMP、NTP、RADIUS、および TACACS+ トラフィックの通過を明示的に許可し、デバイスから 10.30.30.0/24 ネットワークに発信するトラフィックの通過を許可します。 copp-system-acl-deny では、すべてのトラフィックの通過を明示的に許可します。 ポリシング ポリシーは、copp-system-acl-allow ACL によって許可されたトラフィックを送信し、copp-system-acl-deny ACL によって許可されたトラフィックをドロップするように設定されています。

```
ip access-list copp-system-acl-allow
  10 remark ### ALLOW TELNET from 10.30.30.0/24
  20 permit tcp 10.30.30.0/24 any eq telnet
  30 permit tcp 10.30.30.0/24 any eq 107
  40 remark ### ALLOW SSH from 10.30.30.0/24
  50 permit tcp 10.30.30.0/24 any eq 22
  60 remark ### ALLOW SNMP from 10.30.30.0/24
  70 permit udp 10.30.30.0/24 any eq snmp
  80 remark ### ALLOW TACACS from 10.30.30.0/24
  90 permit tcp 10.30.30.0/24 any eq tacacs
  100 remark ### ALLOW RADIUS from 10.30.30.0/24
  110 permit udp 10.30.30.0/24 any eq 1812
  120 permit udp 10.30.30.0/24 any eq 1813
  130 permit udp 10.30.30.0/24 any eq 1645
  140 permit udp 10.30.30.0/24 any eq 1646
  150 permit udp 10.30.30.0/24 eq 1812 any
  160 permit udp 10.30.30.0/24 eq 1813 any
  170 permit udp 10.30.30.0/24 eq 1645 any
  180 permit udp 10.30.30.0/24 eq 1646 any
  190 remark ### ALLOW NTP from 10.30.30.0/24
  200 permit udp 10.30.30.0/24 any eq ntp
  210 remark ### ALLOW ALL OUTBOUND traffic TO 10.30.30.0/24
  220 permit ip any 10.30.30.0/24
  statistics  # keep statistics on matches
ip access-list copp-system-acl-deny
  10 remark ### this is a catch-all to match any other traffic
  20 permit ip any any
  statistics  # keep statistics on matches
```

```
class-map type control-plane match-any copp-system-class-management-allow
  match access-group name copp-system-acl-allow
class-map type control-plane match-any copp-system-class-management-deny
  match access-group name copp-system-acl-deny
policy-map type control-plane copp-system-policy
    class copp-system-class-management-allow
        police cir 3000 pps bc 32 packets conform transmit violate drop
    class copp-system-class-management-deny
        police cir 3000 pps bc 32 packets conform drop violate drop
control-plane
  service-policy input copp-system-policy
```

IP ACL に対する Session Manager のサポート

Session Manager は、IP ACL の設定をサポートしています。 この機能を使用すると、ACL の設定を調べて、その設定に必要とされるリソースが利用可能であるかどうかを、リソースを実行コンフィギュレーションにコミットする前に確認できます。

IP ACL のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	IP ACL を使用するためにライセンスは必要ありません。 ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。 Cisco NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

IP ACL の前提条件

IP ACL の前提条件は次のとおりです。

- IP ACL を設定するためには、IP 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。
- ACL を設定するインターフェイス タイプについての知識が必要です。

IP ACL の注意事項と制約事項

IP ACL の設定に関する注意事項と制約事項は次のとおりです。

- ACL の設定には Session Manager を使用することを推奨します。 この機能を使用すると、ACL の設定を調べて、その設定に必要とされるリソースが利用可能であるかどうかを、リソース

を実行コンフィギュレーションにコミットする前に確認できます。 この機能は、1,000 以上のルールが含まれている ACL に対して特に有効です。 Session Manager の詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

- ほとんどの場合、IP パケットに対する ACL 処理は I/O モジュール上で実行されます。これには、ACL 処理を加速化するハードウェアを使用します。 場合によっては、スーパーバイザ モジュールで処理が実行されることもあります。この場合、特に非常に多数のルールが設定されている ACL を処理する際には、処理速度が遅くなることがあります。 管理インターフェイス トラフィックは、常にスーパーバイザ モジュールで処理されます。 次のカテゴリのいずれかに属する IP パケットがレイヤ 3 インターフェイスから出る場合、これらのパケットはスーパーバイザ モジュールに送られて処理されます。
  - レイヤ 3 最大伝送ユニット チェックに失敗し、そのためにフラグメント化を要求しているパケット
  - IP オプションがある IPv4 パケット（追加された IP パケット ヘッダーのフィールドは、宛先アドレス フィールドの後）
  - 拡張 IPv6 ヘッダー フィールドがある IPv6 パケット

  レート制限を行うことで、リダイレクト パケットによってスーパーバイザ モジュールに過剰な負荷がかかるのを回避します。
- 時間範囲を使用する ACL を適用すると、デバイスはその ACL エントリで参照される時間範囲の開始時または終了時に ACL エントリをアップデートします。 時間範囲によって開始されるアップデートはベストエフォート型のプライオリティで実行されます。時間範囲によってアップデートが生じたときにデバイスの処理負荷が非常に高い場合、デバイスはアップデートを最大数秒間遅らせることがあります。
- VTY ACL 機能はすべての VTY 回線のすべてのトラフィックを制限します。 異なる VTY 回線に異なるトラフィックの制限を指定できません。

  どのルータの ACL も VTY ACL として設定できます。
- 未定義の ACL をインターフェイスに適用すると、システムは空の ACL と見なし、すべてのトラフィックを許可します。
- Cisco NX-OS は、サブインターフェイスの出力ルータ ACL をサポートしません。

IP ACL のデフォルト設定

次の表に、IP ACL パラメータのデフォルト設定を示します。

表 *2*：*IP ACL* パラメータのデフォルト値

パラメータ	デフォルト
IP ACL	デフォルトでは IP ACL は存在しません。

パラメータ	デフォルト
IP ACL エントリ	1024
ACL ルール	すべての ACL に暗黙のルールが適用されます。
時間範囲	デフォルトでは時間範囲は存在しません。

関連トピック

IP ACL の暗黙ルール, （4 ページ）

IP ACL の設定

IP ACL の作成

デバイスに IPv4 ACL または IPv6 ACL を作成し、これにルールを追加できます。

はじめる前に

ACL の設定には Session Manager を使用することを推奨します。 この機能によって、ACL の設定を確認し、設定を実行コンフィギュレーションにコミットする前に、その設定が必要とするリソースが利用可能かどうかを確認できます。 この機能は、約 1,000 以上のルールが含まれている ACL に対して特に有効です。

手順の概要

1. **configure terminal**
2. 次のいずれかのコマンドを入力します。
   - **ip access-list** *name*
   - **ipv6 access-list** *name*
3. （任意） **fragments** {**permit-all** | **deny-all**}
4. [*sequence-number*] {**permit** | **deny**} *protocol source destination*
5. （任意） **statistics per-entry**
6. （任意） 次のいずれかのコマンドを入力します。
   - **show ip access-lists** *name*
   - **show ipv6 access-lists** *name*
7. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	次のいずれかのコマンドを入力します。
	• **ip access-list** *name*
	• **ipv6 access-list** *name*
	
	例：
	`switch(config)# ip access-list acl-01`
	`switch(config-acl)#`
		IP ACL を作成して、IP ACL コンフィギュレーション モードを開始します。 *name* 引数は 64 文字以内で指定します。

ステップ 3	**fragments** {**permit-all** | **deny-all**} 例： `switch(config-acl)# fragments permit-all`	（任意） 初期状態でないフラグメントのフラグメント処理を最適化します。 デバイスで、**fragments** コマンドが含まれる ACL がトラフィックに適用される場合、**fragments** コマンドは、ACL での明示的な **permit** コマンドまたは **deny** コマンドに一致しない非初期フラグメントだけに一致します。
ステップ 4	[*sequence-number*] {**permit** | **deny**} *protocol source destination* 例： `switch(config-acl)# permit ip 192.168.2.0/24 any`	IP ACL 内にルールを作成します。 多数のルールを作成できます。 *sequence-number* 引数には、1 ～ 4294967295 の整数を指定します。 **permit** コマンドと **deny** コマンドには、トラフィックを識別するための多くの方法が用意されています。
ステップ 5	**statistics per-entry** 例： `switch(config-acl)# statistics per-entry`	（任意） その ACL のルールと一致するパケットのグローバル統計をデバイスが維持するように設定します。
ステップ 6	次のいずれかのコマンドを入力します。 • **show ip access-lists** *name* • **show ipv6 access-lists** *name* 例： `switch(config-acl)# show ip access-lists acl-01`	（任意） IP ACL の設定を表示します。

	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： `switch(config-acl)# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

IP ACL の変更

既存の IPv4 ACL または IPv6 ACL のルールの追加と削除は実行できますが、既存のルールを変更することはできません。 ルールを変更するには、そのルールを削除してから、変更を加えたルールを再作成します。

既存のルールの間に新しいルールを挿入する必要がある場合で、現在のシーケンス番号の空き状況ではすべてを挿入できないときは、**resequence** コマンドを使用してシーケンス番号を再割り当てします。

はじめる前に

ACL の設定には Session Manager を使用することを推奨します。 この機能を使用すると、ACL の設定を調べて、その設定に必要とされるリソースが利用可能であるかどうかを、リソースを実行コンフィギュレーションにコミットする前に確認できます。 この機能は、約 1,000 以上のルールが含まれている ACL に対して特に有効です。

手順の概要

1. **configure terminal**
2. 次のいずれかのコマンドを入力します。
   - **ip access-list** *name*
   - **ipv6 access-list** *name*
3. （任意） [*sequence-number*] {**permit** | **deny**} *protocol source destination*
4. （任意） [**no**] **fragments** {**permit-all** | **deny-all**}
5. （任意） **no** {*sequence-number* | {**permit** | **deny**} *protocol source destination*}
6. （任意） [**no**] **statistics per-entry**
7. （任意） 次のいずれかのコマンドを入力します。
   - **show ip access-lists** *name*
   - **show ipv6 access-lists** *name*
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	次のいずれかのコマンドを入力します。
	• **ip access-list** *name*
	• **ipv6 access-list** *name*
	
	例：
	`switch(config)# ip access-list acl-01`
	`switch(config-acl)#`
		名前で指定した ACL の IP ACL 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	[*sequence-number*] {**permit** | **deny**} *protocol source destination* 例： `switch(config-acl)# 100 permit ip 192.168.2.0/24 any`	（任意） IP ACL 内にルールを作成します。 シーケンス番号を指定すると、ACL 内のルール挿入位置を指定できます。 シーケンス番号を指定しないと、ルールは ACL の末尾に追加されます。 *sequence-number* 引数には、1 ～ 4294967295 の整数を指定します。 **permit** コマンドと **deny** コマンドには、トラフィックを識別するための多くの方法が用意されています。
ステップ 4	[**no**] **fragments** {**permit-all** | **deny-all**} 例： `switch(config-acl)# fragments permit-all`	（任意） 初期状態でないフラグメントのフラグメント処理を最適化します。 デバイスで、**fragments** コマンドが含まれる ACL がトラフィックに適用される場合、**fragments** コマンドは、ACL での明示的な **permit** コマンドまたは **deny** コマンドに一致しない非初期フラグメントだけに一致します。 **no** オプションを使用すると、フラグメント処理の最適化が削除されます。
ステップ 5	**no** {*sequence-number* | {**permit** | **deny**} *protocol source destination*} 例： `switch(config-acl)# no 80`	（任意） 指定したルールを IP ACL から削除します。 **permit** コマンドと **deny** コマンドには、トラフィックを識別するための多くの方法が用意されています。
ステップ 6	[**no**] **statistics per-entry** 例： `switch(config-acl)# statistics per-entry`	（任意） その ACL のルールと一致するパケットのグローバル統計をデバイスが維持するように設定します。

	コマンドまたはアクション	目的
		no オプションを使用すると、デバイスはその ACL のグローバル統計の維持を停止します。
ステップ 7	次のいずれかのコマンドを入力します。 • **show ip access-lists** *name* • **show ipv6 access-lists** *name* 例： `switch(config-acl)# show ip access-lists acl-01`	（任意） IP ACL の設定を表示します。
ステップ 8	**copy running-config startup-config** 例： `switch(config-acl)# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

関連トピック

IP ACL 内のシーケンス番号の変更，（19 ページ）

VTY ACL の作成

入力方向または出力方向の全 VTY 回線で、すべての IPv4 または IPv6 トラフィックへのアクセスを制御することにより、VTY ACL を設定できます。

はじめる前に

すべての仮想端末回線にユーザが接続できるため、すべての仮想端末回線に同じ制約を設定する必要があります。

ACL の設定には Session Manager を使用することを推奨します。 この機能を使用すると、ACL の設定を調べて、その設定に必要とされるリソースが利用可能であるかどうかを、リソースを実行コンフィギュレーションにコミットする前に確認でき、特に約 1000 以上のルールを含む ACL に役立ちます。

手順の概要

1. **configure terminal**
2. {**ip** | **ipv6**} **access-list** *name*
3. {**permit** | **deny**} *protocol source destination* [**log**] [**time-range** *time*]
4. **exit**
5. **line vty**
6. {**ip** | **ipv6**} **access-class** *name* {**in** | **out**}
7. （任意） **show** {**ip** | **ipv6**} **access-lists**
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	{**ip** | **ipv6**} **access-list** *name* 例： `switch(config)# ip access-list vtyacl`	ACL を作成し、その ACL の IP 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。 *name* 引数の最大長は 64 文字です。
ステップ 3	{**permit** | **deny**} *protocol source destination* [**log**] [**time-range** *time*] 例： `switch(config-ip-acl)# permit tcp any any`	ACL ルールを作成し、指定した送信元とのすべての TCP トラフィックを許可します。

ステップ 4	**exit**
	
	例：
	`switch(config-ip-acl)# exit`
	`switch(config)#`
		IP アクセス リスト コンフィギュレーション モードを終了します。

ステップ 5	**line vty**
	
	例：
	`switch(config)# line vty`
	`switch(config-line)#`
		仮想端末を指定し、ライン コンフィギュレーション モードを開始します。

ステップ 6	{**ip** | **ipv6**} **access-class** *name* {**in** | **out**} 例： `switch(config-line)# ip access-class vtyacl out`	指定された ACL を使用してすべての VTY 回線に対する着信および発信接続を制限します。 *name* 引数の最大長は 64 文字です。

	コマンドまたはアクション	目的
ステップ 7	**show** {**ip** | **ipv6**} **access-lists** 例： `switch# show ip access-lists`	（任意） 任意の VTY ACL を含め、設定された ACL を表示します。
ステップ 8	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

IP ACL 内のシーケンス番号の変更

IP ACL 内のルールに付けられたすべてのシーケンス番号を変更できます。

はじめる前に

ACL の設定には Session Manager を使用することを推奨します。 この機能を使用すると、ACL の設定を調べて、その設定に必要とされるリソースが利用可能であるかどうかを、リソースを実行コンフィギュレーションにコミットする前に確認できます。 この機能は、約 1,000 以上のルールが含まれている ACL に対して特に有効です。

手順の概要

1. **configure terminal**
2. **resequence** {**ip** | **ipv6**} **access-list** *name starting-sequence-number increment*
3. （任意） **show ip access-lists** *name*
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**resequence** {**ip** | **ipv6**} **access-list** *name starting-sequence-number increment* 例： `switch(config)# resequence access-list ip acl-01 100 10`	ACL内に記述されているルールにシーケンス番号を付けます。指定した開始シーケンス番号が最初のルールに付けられます。後続の各ルールには、直前のルールよりも大きい番号が付けられます。 番号の間隔は、指定した増

	コマンドまたはアクション	目的
		分によって決まります。 *starting-sequence-number* 引数と *increment* 引数は、1 ～ 4294967295 の整数で指定します。
ステップ 3	**show ip access-lists** *name* 例： `switch(config)# show ip access-lists acl-01`	（任意） IP ACL の設定を表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

IP ACL の削除

IP ACL をデバイスから削除できます。

はじめる前に

その ACL がインターフェイスに適用されているかどうかを確認します。 削除できるのは、現在適用されている ACL です。 ACL を削除しても、その ACL が適用されていたインターフェイスの設定は影響を受けません。 デバイスは削除された ACL を空であると見なします。 IP ACL が設定されているインターフェイスを探すには、**show ip access-lists** コマンドまたは **show ipv6 access-lists** コマンドと一緒に summary キーワードを使用します。

手順の概要

1. **configure terminal**
2. 次のいずれかのコマンドを入力します。
   - **no ip access-list** *name*
   - **no ipv6 access-list** *name*
3. （任意） 次のいずれかのコマンドを入力します。
   - **show ip access-lists** *name* **summary**
   - **show ipv6 access-lists** *name* **summary**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	次のいずれかのコマンドを入力します。 • **no ip access-list** *name* • **no ipv6 access-list** *name* 例： `switch(config)# no ip access-list acl-01`	名前で指定した IP ACL を実行コンフィギュレーションから削除します。
ステップ 3	次のいずれかのコマンドを入力します。 • **show ip access-lists** *name* **summary** • **show ipv6 access-lists** *name* **summary** 例： `switch(config)# show ip access-lists acl-01 summary`	（任意） IP ACL の設定を表示します。 ACL がインターフェイスに引き続き適用されている場合は、インターフェイスが表示されます。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ルータ ACL としての IP ACL の適用

IPv4 ACL または IPv6 ACL は、次のタイプのインターフェイスに適用できます。

- 物理レイヤ 3 インターフェイスおよびサブインターフェイス
- レイヤ 3 イーサネット ポート チャネル インターフェイス
- 管理インターフェイス

これらのインターフェイス タイプに適用された ACL はルータ ACL と見なされます。

（注） Cisco NX-OS は、サブインターフェイスの出力ルータ ACL をサポートしません。

はじめる前に

適用する ACL が存在し、目的に応じたトラフィック フィルタリングが設定されていることを確認します。

手順の概要

1. switch# **configure terminal**
2. 次のいずれかのコマンドを入力します。
   - switch(config)# **interface ethernet** *slot*/*port*[**.** *number*]
   - switch(config)# **interface port-channel** *channel-number*
   - switch(config)# **interface mgmt** *port*
3. （任意） switch(config-if)# **show running-config aclmgr**
4. （任意） switch(config-if)# **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	switch# **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	次のいずれかのコマンドを入力します。 • switch(config)# **interface ethernet** *slot*/*port*[**.** *number*] • switch(config)# **interface port-channel** *channel-number* • switch(config)# **interface mgmt** *port*	指定したインターフェイス タイプ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 3	switch(config-if)# **show running-config aclmgr**	（任意） ACL の設定を表示します。
ステップ 4	switch(config-if)# **copy running-config startup-config**	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

関連トピック

IP ACL の作成, （13 ページ）

ACL TCAM リージョン サイズの設定

ハードウェアの ACL Ternary Content Addressable Memory（TCAM）リージョンのサイズを変更できます。

手順の概要

1. **configure terminal**
2. **hardware access-list tcam region** {**copp** | **e-ipv6-racl** | **e-racl** | **ipv6-l3qos** | **ipv6-racl** | **l3qos** | **racl** | **redirect**} *tcam-size*
3. **copy running-config startup-config**
4. switch(config)# **show hardware access-list tcam region**
5. switch(config)# **copy running-config startup-config**
6. switch(config)# **reload**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**hardware access-list tcam region** {**copp** | **e-ipv6-racl** | **e-racl** | **ipv6-l3qos** | **ipv6-racl** | **l3qos** | **racl** | **redirect**} *tcam-size*	ACL TCAM リージョン サイズを変更します。 • **copp**：CoPP TCAM リージョンのサイズを設定します。 • **e-ipv6-racl**：IPv6 出力ルータ ACL（ERACL）TCAM リージョン サイズを設定します。 • **e-racl**：IPv4 出力ルータ ACL（ERACL）TCAM リージョン サイズを設定します。 • **ipv6-l3qos**：IPv6 レイヤ 3 Quality of Service (QoS) TCAM リージョンのサイズを設定します。 • **ipv6-racl**：IPv6 RACL TCAM リージョン サイズを設定します。 • **l3qos**：IPv4 レイヤ 3 Quality of Service (QoS) TCAM リージョンのサイズを設定します。 • **racl**：IPv4 ルータの ACL（RACL）TCAM リージョン サイズを設定します。

	コマンドまたはアクション	目的
		• **redirect**：リダイレクト TCAM リージョンのサイズを設定します。 • *tcam-size*：TCAM サイズ。 サイズは 256 の倍数です。 サイズが 256 より大きい場合は、512 の倍数でなければなりません。
ステップ 3	**copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。
ステップ 4	switch(config)# **show hardware access-list tcam region**	デバイスで次のリロード時に適用される TCAM サイズを表示します。
ステップ 5	switch(config)# **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。
ステップ 6	switch(config)# **reload**	デバイスがリロードされます。 （注） 新しいサイズの値は、**copy running-config startup-config** + **reload** を入力するか、すべてのラインカードモジュールをリロードした後にのみ有効になります。

次に、RACL TCAM リージョンのサイズを変更する例を示します。

```
switch(config)# hardware access-list tcam region racl 256
[SUCCESS] New tcam size will be applicable only at boot time.
You need to 'copy run start' and 'reload'
switch(config)# copy running-config startup-config
switch(config)# reload
WARNING: This command will reboot the system
Do you want to continue? (y/n) [n] y
```

次に、変更を確認するために、TCAM リージョンのサイズを表示する例を示します。

```
switch(config)# show hardware access-list tcam region
         TCAM Region Sizes:

               IPV4 RACL size = 2048
        Egress IPV4 RACL size =  768
             IPV4 L3 QoS size =  256
          Ingress System size =  256
           Egress System size =  256
            Ingress COPP size =  256
                Redirect size =  256
```

デフォルトの TCAM リージョン サイズに戻す

手順の概要

1. **configure terminal**
2. **no hardware access-list tcam region** {**copp** | **e-ipv6-racl** | **e-racl** | **ipv6-l3qos** | **ipv6-racl** | **l3qos** | **racl** | **redirect**} *tcam-size*
3. （任意） **copy running-config startup-config**
4. switch(config)# **reload**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**no hardware access-list tcam region** {**copp** | **e-ipv6-racl** | **e-racl** | **ipv6-l3qos** | **ipv6-racl** | **l3qos** | **racl** | **redirect**} *tcam-size*	デフォルト ACL TCAM サイズに設定を戻します。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を永続的に保存します。
ステップ 4	switch(config)# **reload**	デバイスがリロードされます。

次に、デフォルトの RACL TCAM リージョンのサイズに戻す例を示します。

```
switch(config)# no hardware profile tcam region racl 256
[SUCCESS] New tcam size will be applicable only at boot time.
You need to 'copy run start' and 'reload'
switch(config)# copy running-config startup-config
switch(config)# reload
WARNING: This command will reboot the system
Do you want to continue? (y/n) [n] y
```

IPv6 RACL のイネーブル化

デフォルトの TCAM リージョン設定は、IPv6 ルータ ACL（RACL）に対応していません。 IPv6 RACL をイネーブルにするには、もう一方のリージョンの TCAM サイズを減らしてから、IPv6 RACL リージョンの TCAM サイズを増やします。

次の表に、入出力 TCAM リージョンのデフォルト サイズを示します。

表 *3*：デフォルト *TCAM* リージョン設定（入力）

Region Name	Size	幅	合計サイズ
IPv4 RACL	2048	1	2048
レイヤ 3 QoS	256	2	512
CoPP	256	2	512
System	256	2	512
リダイレクト	256	1	256
			3.75K

表 *4*：デフォルト *TCAM* リージョン設定（出力）

Region Name	Size	幅	合計サイズ
IPv4 RACL	768	1	768
System	256	1	256
			1 K

入力 IPv6 RACL TCAM リージョンのサイズを設定するには、2 つのオプションのいずれか 1 つを実行します。

（注） この例では、IPv6 RACL TCAM サイズを 256 に設定します。 サイズが 256 の IPv6 RACL は、IPv6 がダブル幅であるため、512 エントリを使用します。

オプション #1

入力 IPv4 RACL を 512 エントリ減らし（2048 - 512 = 1536）、入力 IPv6 RACL を 512 エントリ増やします。このオプションが優先されます。

```
switch(config)# hardware access-list tcam region racl 1536
Warning: Please reload the linecard for the configuration to take effect
switch(config)# hardware access-list tcam region ipv6-racl 256
Warning: Please reload the linecard for the configuration to take effect
```

表 *5*：*IPv4 RACL*（入力）を減らした後の更新された *TCAM* リージョン設定

Region Name	Size	幅	合計サイズ
IPv4 RACL	1536	1	1536
IPv6 RACL	256	2	512
レイヤ 3 QoS	256	2	512
CoPP	256	2	512

Region Name	Size	幅	合計サイズ
System	256	2	512
リダイレクト	256	1	256
			3.75K

オプション #2

IPv4 L3 QoS のサイズを 0 に減らして削除し、入力 IPv6 RACL を追加します。このオプションは、レイヤ 3 QoS を使用していない場合に使用できます。

```
switch(config)# hardware access-list tcam region l3qos 0
Warning: Please reload the linecard for the configuration to take effect
switch(config)# hardware access-list tcam region ipv6-racl 256
Warning: Please reload the linecard for the configuration to take effect
```

表 *6*：レイヤ *3 QoS*（入力）を削除した後の更新された *TCAM* リージョン設定

Region Name	Size	幅	合計サイズ
IPv4 RACL	2048	1	2048
IPv6 RACL	256	2	512
レイヤ 3 QoS	0	2	0
CoPP	256	2	512
System	256	2	512
リダイレクト	256	1	256
			3.75K

サイズ 256 の出力 IPv6 RACL をイネーブルにするには、出力 IPv4 RACL を 256 に減らし、出力 IPv6 RACL を追加します。

```
switch(config)# hardware access-list tcam region e-racl 256
Warning: Please reload the linecard for the configuration to take effect
switch(config)# hardware access-list tcam region e-ipv6-racl 256
Warning: Please reload the linecard for the configuration to take effect
```

表 *7*：*IPv4 RACL*（出力）を減らした後のデフォルト *TCAM* リージョン設定

Region Name	Size	幅	合計サイズ
IPv4 RACL	256	1	256
IPv6 RACL	256	2	512
System	256	1	256
			1 K

TCAM リージョンのサイズを調整した後、**show hardware access-list tcam region** コマンドを入力して、デバイスの次回リロード時に適用される TCAM サイズを表示します。

注目 すべてのモジュールの同期を維持するには、すべてのラインカード モジュールをリロードするか、**copy running-config startup-config + reload** を入力してデバイスをリロードする必要があります。 TCAM リージョン設定が複数であっても、必要なリロードは 1 度だけです。 TCAM リージョン設定がすべて完了するのを待ってから、デバイスをリロードできます。

TCAM リージョンの設定時に、すべての TCAM リージョンの 4K 入力制限を超えると、次のメッセージが表示されます。

```
ERROR: Aggregate TCAM region configuration exceeded the available
Ingress TCAM space. Please re-configure.
```

TCAM リージョンの設定時に、すべての TCAM リージョンの 1K 出力制限を超えると、次のメッセージが表示されます。

```
ERROR: Aggregate TCAM region configuration exceeded the available
Egress TCAM space. Please re-configure.
```

特定の機能の TCAM が設定されていない状態で TCAM カービングを必要とする機能を適用しようとすると、次のメッセージが表示されます。

```
 ERROR: Module x returned status: TCAM region is not
configured. Please configure TCAM region and retry the command.
```

（注） 256 というデフォルトのリダイレクト TCAM リージョン サイズは、多数の BFD または DHCP リレー セッションを実行している場合は十分でない可能性があります。 より多くの BFD または DHCP リレー セッションに対応するために、TCAM サイズを 512 に増やす必要がある場合があります。

関連トピック

ACL TCAM リージョン サイズの設定, （23 ページ）

IP ACL の設定の確認

IP ACL の設定情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show ip access-lists	IPv4 ACL の設定を表示します。
show ipv6 access-lists	IPv6 ACL の設定を表示します。

コマンド	目的
show system internal access-list feature bank map interface {**egress** | **ingress**} [**module** *module*]	機能グループとクラスの組み合わせの表を表示します。
show running-config aclmgr [**all**]	IP ACL の設定および IP ACL が適用されるインターフェイスを含めて、ACL の実行コンフィギュレーションを表示します。
show startup-config aclmgr [**all**]	ACL のスタートアップ コンフィギュレーションを表示します。

IP ACL の統計情報のモニタリングとクリア

IP ACL の統計情報のモニタまたはクリアを行うには、次の表に示すコマンドのいずれかを使用します。

コマンド	目的
show ip access-lists	IPv4 ACL の設定を表示します。 IPv4 ACL に **statistics per-entry** コマンドが含まれている場合は、**show ip access-lists** コマンドの出力に、各ルールと一致したパケットの数が含まれます。
show ipv6 access-lists	IPv6 ACL の設定を表示します。 IPv6 ACL に **statistics per-entry** コマンドが含まれている場合は、**show ipv6 access-lists** コマンドの出力に、各ルールと一致したパケットの数が含められます。
clear ip access-list counters	すべての IPv4 ACL または特定の IPv4 ACL の統計情報をクリアします。
clear ipv6 access-list counters	すべての IPv6 ACL または特定の IPv6 ACL の統計情報をクリアします。

IP ACL の設定例

acl-120 という名前の IPv6 ACL を作成し、これをルータ ACL としてイーサネット インターフェイス 2/3（レイヤ 3 インターフェイス）に適用する例を示します。

```
ipv6 access-list acl-120
  permit tcp 2001:0db8:85a3::/48 2001:0db8:be03:2112::/64
  permit udp 2001:0db8:85a3::/48 2001:0db8:be03:2112::/64
  permit tcp 2001:0db8:69f2::/48 2001:0db8:be03:2112::/64
  permit udp 2001:0db8:69f2::/48 2001:0db8:be03:2112::/64
interface ethernet 2/3
  ipv6 traffic-filter acl-120 in
```

次に、single-source という名前の VTY ACL を作成し、それを VTY 回線上の入力 IP トラフィックに対して適用する例を示します。 この ACL は、通過するすべての TCP トラフィックを許可し、その他のすべての IP トラフィックをドロップします。

```
ip access-list single-source
  permit tcp 192.168.7.5/24 any
  exit
  line vty
  ip access-class single-source in
  show ip access-lists
```

時間範囲の設定

時間範囲に対する Session Manager のサポート

Session Manager は時間範囲の設定をサポートしています。 この機能を使用すると、設定セッションを作成し、時間範囲の設定変更を実行コンフィギュレーションにコミットする前に確認できます。

時間範囲の作成

デバイス上で時間範囲を作成し、これにルールを追加できます。

手順の概要

1. **configure terminal**
2. **time-range** *name*
3. （任意） [*sequence-number*] **periodic** *weekday time* **to** [*weekday*] *time*
4. （任意） [*sequence-number*] **periodic** *list-of-weekdays time* **to** *time*
5. （任意） [*sequence-number*] **absolute start** *time date* [**end** *time date*]
6. （任意） [*sequence-number*] **absolute** [start *time date*] **end** *time date*
7. （任意） **show time-range** *name*
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**time-range** *name*
	
	例：
	`switch(config)# time-range workday-daytime`
	`switch(config-time-range)#`
		時間範囲を作成し、時間範囲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	[*sequence-number*] **periodic** *weekday time* **to** [*weekday*] *time* 例： `switch(config-time-range)# periodic monday 00:00:00 to friday 23:59:59`	（任意） 指定開始日時と終了日時の間（両端を含める）の 1 日以上の連続した曜日だけ有効になるような定期ルールを作成します。
ステップ 4	[*sequence-number*] **periodic** *list-of-weekdays time* **to** *time* 例： `switch(config-time-range)# periodic weekdays 06:00:00 to 20:00:00`	（任意） *list-of-weekdays* 引数で指定された曜日の指定開始時刻と終了時刻の間（両端を含む）だけ有効になるような定期ルールを作成します。 *list-of-weekdays* 引数の値には次のキーワードも使用できます。 • **daily**：すべての曜日 • **weekdays**：月曜から金曜まで（Monday ～ Friday） • **weekend**：土曜と日曜（Saturday ～ Sunday）

	コマンドまたはアクション	目的
ステップ 5	[*sequence-number*] **absolute start** *time date* [**end** *time date*] 例： `switch(config-time-range)# absolute start 1:00 15 march 2013`	（任意） **start** キーワードの後ろに指定した日時から有効になる絶対ルールを作成します。 **end** キーワードを省略すると、そのルールは開始日時を過ぎると常に有効になります。
ステップ 6	[*sequence-number*] **absolute** [start *time date*] **end** *time date* 例： `switch(config-time-range)# absolute end 23:59:59 31 may 2013`	（任意） **end** キーワードの後ろに指定した日時まで有効になる絶対ルールを作成します。 **start** キーワードを省略すると、そのルールは終了日時を過ぎるまで常に有効です。
ステップ 7	**show time-range** *name* 例： `switch(config-time-range)# show time-range workday-daytime`	（任意） 時間範囲の設定を表示します。
ステップ 8	**copy running-config startup-config** 例： `switch(config-time-range)# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

時間範囲の変更

既存の時間範囲のルールの追加および削除を実行できます。 既存のルールは変更できません。ルールを変更するには、そのルールを削除してから、変更を加えたルールを再作成します。

既存のルールの間に新しいルールを挿入する必要がある場合で、現在のシーケンス番号の空き状況ではすべてを挿入できないときは、**resequence** コマンドを使用してシーケンス番号を再割り当てします。

手順の概要

1. **configure terminal**
2. **time-range** *name*
3. （任意） [*sequence-number*] **periodic** *weekday time* **to** [*weekday*] *time*
4. （任意） [*sequence-number*] **periodic** *list-of-weekdays time* **to** *time*
5. （任意） [*sequence-number*] **absolute start** *time date* [**end** *time date*]
6. （任意） [*sequence-number*] **absolute** [**start** *time date*] **end** *time date*
7. （任意） **no** {*sequence-number* | **periodic** *arguments* . . . | **absolute** *arguments*. . .}
8. （任意） **show time-range** *name*
9. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**time-range** *name*
	
	例：
	`switch(config)# time-range workday-daytime`
	`switch(config-time-range)#`
		特定の時間範囲の時間範囲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	[*sequence-number*] **periodic** *weekday time* **to** [*weekday*] *time* 例： `switch(config-time-range)# periodic monday 00:00:00 to friday 23:59:59`	（任意） 指定開始日時と終了日時の間（両端を含める）の 1 日以上の連続した曜日だけ有効になるような定期ルールを作成します。
ステップ 4	[*sequence-number*] **periodic** *list-of-weekdays time* **to** *time* 例： `switch(config-time-range)# 100 periodic weekdays 05:00:00 to 22:00:00`	（任意） *list-of-weekdays* 引数で指定された曜日の指定開始時刻と終了時刻の間（両端を含む）だけ有効になるような定期ルールを作成します。 *list-of-weekdays* 引数の値には次のキーワードも使用できます。 • **daily**：すべての曜日 • **weekdays**：月曜から金曜まで（Monday ～ Friday） • **weekend**：土曜と日曜（Saturday ～ Sunday）

	コマンドまたはアクション	目的
ステップ 5	[*sequence-number*] **absolute start** *time date* [**end** *time date*] 例： `switch(config-time-range)# absolute start 1:00 15 march 2013`	（任意） **start** キーワードの後ろに指定した日時から有効になる絶対ルールを作成します。 **end** キーワードを省略すると、そのルールは開始日時を過ぎると常に有効になります。
ステップ 6	[*sequence-number*] **absolute** [**start** *time date*] **end** *time date* 例： `switch(config-time-range)# absolute end 23:59:59 31 may 2013`	（任意） **end** キーワードの後ろに指定した日時まで有効になる絶対ルールを作成します。 **start** キーワードを省略すると、そのルールは終了日時を過ぎるまで常に有効です。
ステップ 7	**no** {*sequence-number* | **periodic** *arguments* . . . | **absolute** *arguments*. . .} 例： `switch(config-time-range)# no 80`	（任意） 時間範囲から特定のルールを削除します。
ステップ 8	**show time-range** *name* 例： `switch(config-time-range)# show time-range workday-daytime`	（任意） 時間範囲の設定を表示します。
ステップ 9	**copy running-config startup-config** 例： `switch(config-time-range)# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

関連トピック

時間範囲のシーケンス番号の変更, （35 ページ）

時間範囲の削除

デバイスから時間範囲を削除できます。

はじめる前に

その時間範囲が ACL ルールのいずれかに使用されているかどうかを確認します。 削除できるのは、ACL ルールに使用されている時間範囲です。 ACL ルールに使用されている時間範囲を削除しても、その ACL が適用されているインターフェイスの設定には影響しません。 デバイスは削除された時間範囲を使用する ACL ルールを空であると見なします。

手順の概要

1. **configure terminal**
2. **no time-range** *name*
3. （任意） **show time-range**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**no time-range** *name* 例： `switch(config)# no time-range daily-workhours`	名前を指定した時間範囲を削除します。
ステップ 3	**show time-range** 例： `switch(config-time-range)# show time-range`	（任意） すべての時間範囲の設定を表示します。 削除された時間範囲は表示されません。
ステップ 4	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

時間範囲のシーケンス番号の変更

時間範囲のルールに割り当てられているすべてのシーケンス番号を変更できます。

手順の概要

1. **configure terminal**
2. **resequence time-range** *name starting-sequence-number increment*
3. （任意） **show time-range** *name*
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**resequence time-range** *name starting-sequence-number increment*
	
	例：
	`switch(config)# resequence time-range daily-workhours 100 10`
	`switch(config)#`
		時間範囲のルールにシーケンス番号を割り当てます。指定した開始シーケンス番号は最初のルールに割り当てられます。後続の各ルールには、直前のルールよりも大きい番号が付けられます。番号の間隔は、指定した増分によって決まります。

ステップ 3	**show time-range** *name* 例： `switch(config)# show time-range daily-workhours`	（任意） 時間範囲の設定を表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

時間範囲設定の確認

時間範囲の設定情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show time-range	時間範囲の設定を表示します。
show running-config aclmgr	すべての時間範囲を含めて、ACL の設定を表示します。