Cisco IOS ACL のサポート

（注）
- Cisco IOS ACL の詳細な設定手順については、次のマニュアルを参照してください。

  http://www.cisco.com/en/US/docs/ios-xml/ios/sec_data_acl/configuration/15-sy/sec-data-acl-15-sy-book.html
- この章で使用しているコマンドの構文および使用方法の詳細については、次の資料を参照してください。

  http://www.cisco.com/en/US/products/ps11846/prod_command_reference_list.html
- Cisco IOS Release 15.1SY は、イーサネット インターフェイスだけをサポートしています。Cisco IOS Release 15.1SY は、WAN 機能またはコマンドをサポートしていません。

ヒント Cisco Catalyst 6500 シリーズ スイッチの詳細（設定例およびトラブルシューティング情報を含む）については、次のページに示されるドキュメントを参照してください。

http://www.cisco.com/en/US/products/hw/switches/ps708/tsd_products_support_series_home.html

技術マニュアルのアイデア フォーラムに参加する

Cisco IOS ACL の制約事項

- Cisco IOS ACL をレイヤ 3 ポートおよび VLAN インターフェイスに直接、適用できます。

- VLAN ACL とポート ACL をレイヤ 2 インターフェイスと VLAN に適用できます（第 73 章「ポート ACL（PACL）」および第 74 章「VLAN ACL（VACL）」を参照）。
- 各タイプの ACL（IP、IPX、および MAC）は対応するトラフィック タイプだけをフィルタリングします。**mac packet-classify** コンフィギュレーション コマンドがイネーブルでない限り、Cisco IOS MAC ACL は、IP または IPX トラフィックと一致しません。デフォルトでは、**mac packet-classify** コンフィギュレーション コマンドはディセーブルになります。
- **mac packet-classify** コンフィギュレーション コマンドを入力すると、MAC ACL がすべてのプロトコル トラフィックに適用されます。
- PFC では、ハードウェアで Cisco IOS IPX ACL をサポートしません。Cisco IOS IPX ACL は、ルート プロセッサ（RP）のソフトウェアでサポートされます。
- デフォルトでは、パケットがアクセス グループによって拒否された場合、インターネット制御メッセージ プロトコル（ICMP）到達不能メッセージが RP によって送信されます。

  ip unreachables コマンドがイネーブルの場合（デフォルト）、スイッチは拒否されたパケットの大部分をハードウェアでドロップし、一部のパケットだけが RP に送信されてソフトウェアでドロップされます（これにより ICMP 到達不能メッセージが生成されます）。

  ip unreachables コマンドは、ACL ドロップ パケットのハードウェアの動作に影響を与えず、ACL 拒否パケットのリークはデフォルトでイネーブルです。インターフェイス上のイーサネット ICMP 到達不能メッセージをディセーブルにするには、**no ip unreachables** インターフェイス コンフィギュレーション コマンドを入力します。
- パケットが VACL または PACL によって拒否された場合、ICMP 到達不能メッセージは送信されません。
- 名前付き ACL を使用すると、ACL 設定の作成または変更時およびシステム再起動中の CPU 使用率を低く抑えられるため、番号付き ACL ではなく名前付き ACL を使用してください。ACL エントリを作成する（または既存の ACL エントリを変更する）場合、ソフトウェアでは ACL 設定を PFC ハードウェアにロードするために ACL マージと呼ばれる CPU 中心の動作が行われます。ACL マージはまた、システム再起動中にスタートアップ コンフィギュレーションを適用する際にも発生します。

  名前付き ACL を使用すると、ユーザが **named-acl** コンフィギュレーション モードを終了するときにだけ ACL マージが開始されます。ただし、名前付き ACL では、ACL 定義すべてについて ACL マージが開始されるため、中規模のマージが ACL 設定中に何度も行われることになります。
- グローバル デフォルト結果は、ヒットレス アップデートが成功しない場合、または機能が特定のインターフェイスに設定されていない場合に使用されます。

ACL のレイヤ 4 演算の制約事項

レイヤ 4 演算の使用

次のタイプの演算子を指定できます。

- gt（greater than：より大きい）
- lt（less than：より小さい）
- neq（not equal：等しくない）

- eq（equal：等しい）
- range（inclusive range：包含範囲）

1 つの ACL に、*9* つより多くの*異なる*演算を指定しないよう推奨します。この数を超えると、新しい演算によって影響される ACE が、複数の ACE に分割されることがあります。

レイヤ 4 演算を使用するときは、次の 2 つの注意事項に従ってください。

- レイヤ 4 演算は、演算子またはオペランドが異なっていると、違う演算であると見なされます。たとえば、次の ACL には 3 つの異なるレイヤ 4 演算が定義されています（「gt 10」と「gt 11」は 2 つの異なるレイヤ 4 演算です）。

```
... gt 10 permit
... lt 9 deny
... gt 11 deny
```

（注） 「eq」演算子の使用に制限はありません。「eq」演算子は論理演算ユニット（LOU）またはレイヤ 4 演算ビットを使用しないためです。LOU については、「論理演算ユニット（LOU）の使用」（P.69-3）を参照してください。

- レイヤ 4 演算は、同じ演算子 / オペランドの組み合わせでも、送信元ポートに適用するか宛先ポートに適用するかによって異なる演算になります。たとえば次の ACL では、1 つの ACE には送信元ポート、もう 1 つの ACE には宛先ポートが指定されているので、2 つの異なるレイヤ 4 演算が定義されていることになります。

```
... Src gt 10 ...
... Dst gt 10
```

論理演算ユニット（LOU）の使用

Logical Operation Unit（LOU; 論理演算ユニット）は、演算子 / オペランドの組み合わせを保存するレジスタです。ACL はすべて、LOU を使用します。最大 104 の LOU があります。各 LOU には、2 つの異なる演算子 / オペランドの組み合わせを保存でき、LOU レジストの総数は、208 になります。レイヤ 4 演算は、次のように LOU を使用します。

- gt は、1/2 LOU を使用します。
- lt は、1/2 LOU を使用します。
- neq は、1/2 LOU を使用します。
- range は、1 LOU を使用します。
- eq は、LOU を使用しません。

たとえば、次の ACL では、1 つの LOU に 2 つの異なる演算子 / オペランドの組み合わせが保存されます。

```
... Src gt 10 ...
... Dst gt 10
```

以下は、より詳細な例です。

```
ACL1
... (dst port) gt 10 permit
... (dst port) lt 9 deny
... (dst port) gt 11 deny
... (dst port) neq 6 permit
... (src port) neq 6 deny
... (dst port) gt 10 deny
```

```
ACL2
... (dst port) gt 20 deny
... (src port) lt 9 deny
... (src port) range 11 13 deny
... (dst port) neq 6 permit
```

レイヤ 4 演算数と LOU 数は、次のとおりです。

- ACL1 のレイヤ 4 演算：5
- ACL2 のレイヤ 4 演算：4
- LOU：4

LOU は、次のように使用されています。

- LOU1 に、「gt 10」および「lt 9」が保存されます。
- LOU2 に、「gt 11」および「neq 6」が保存されます。
- LOU 3 に、「gt 20」が保存されます（半分は空き）。
- LOU 4 に、「range 11 13」が保存されます（range は 1 LOU を使用）。

ACL サポートについて

ACL は、ハードウェアの場合にはポリシー フィーチャ カード（PFC）、分散型フォワーディング カード（DFC）で、ソフトウェアの場合にはルート プロセッサ（RP）で処理できます。

- 標準 ACL および拡張 ACL（入力および出力）の「deny」ステートメントに一致する ACL フローは、「ip unreachables」がディセーブルに設定されている場合、ハードウェアによってドロップされます。
- 標準 ACL および拡張 ACL（入力および出力）の「permit」ステートメントに一致する ACL フローは、ハードウェアで処理されます。
- VLAN ACL（VACL）フローおよびポート ACL（PACL）フローはハードウェアで処理されます。VACL または PACL で指定されたフィールドのハードウェア処理がサポートされていない場合、このフィールドは無視されるか（ACL の **log** キーワードなど）、または設定全体が拒否されます（IPX ACL パラメータを含む VACL など）。
- IPv6 ACL は 32 ビット符号化を使用します。
- VACL ログ機能はソフトウェアで処理されます。
- VACL は IPX アクセス リストではサポートされません。
- VACL は、拒否パケットのロギングだけをサポートします。
- ダイナミック ACL フローはハードウェアで処理されます。
- アイドル タイムアウトはソフトウェアで処理されます。

（注） アイドル タイムアウトは設定できません。Cisco IOS Release 15.1SY では、**access-enable host timeout** コマンドがサポートされていません。

- MPLS インターフェイス以外では、セッション内の最初のパケットが RP 上のソフトウェアで処理された後、リフレクシブ ACL フローがハードウェアで処理されます。

- リフレクシブ ACL フローは、IP からの各種のタグへのトラフィックの場合、および各種のタグから IP へのトラフィックの場合は、ハードウェアによって加速されません。リフレクシブ ACL フローは、すべてのトンネル インターフェイスへの着信および発信トラフィックの場合、ハードウェアによって加速化されません。
- 特定のポート上の ACL アクセス違反の IP アカウンティングは、拒否されリークされた ACL パケットの場合にのみ、そのポート上で拒否された全パケットを RP に転送し、ソフトウェアで処理させることによってサポートされます。この動作は他のフローには影響しません。
- MAC ACL は、スイッチ ポート（MAC PACL）または VLAN 上で、ハードウェアで VACL の一部としてサポートされます。
- PFC では、ハードウェアで Cisco IOS IPX ACL をサポートしません。Cisco IOS IPX ACL は、RP のソフトウェアでサポートされます。
- 名前ベースの拡張 MAC アドレス ACL は、ハードウェアでサポートされています。
- 次の ACL タイプは、ソフトウェアによって処理されます。
  - Internetwork Packet Exchange（IPX）アクセス リスト
  - 標準 XNS アクセス リスト
  - 拡張 XNS アクセス リスト
  - DECnet アクセス リスト
  - プロトコル タイプコード アクセス リスト

（注） ヘッダー長が 5 バイト未満の IP パケットは、アクセス コントロールされません。

- Optimized ACL Logging（OAL; 最適化 ACL ロギング）を設定しない限り、ロギングを必要とするフローはソフトウェアで処理され、ハードウェアでの非ロギング フローの処理には影響しません（「最適化された ACL ロギング」（P.69-13）を参照）。
- ソフトウェアで処理されるフローの転送レートは、ハードウェアで処理されるフローに比べると、大幅に小さくなります。
- ハードウェア統計情報機能がイネーブルである場合に、**show ip access-list** コマンドの出力に表示される一致カウントには、ハードウェアで処理されたパケットが含まれます。
- PFC インターフェイスで **ip unreachables config** コマンドを入力すると、ハードウェアの動作は、変更されないままです。
- IPv4 および IPv6 のヒットレス TCAM アップデートは、TCAM の新機能のアップデート一方で、着信トラフィックに既存の機能を適用する機能があります。特定のインターフェイスの IPv6 ACL の変更によって、すべてのインターフェイス上のすべての IPv6 機能の再プログラミングをトリガーする IPv6 トラフィックには、ヒットレス機能アップデートが必須です。
- ヒットレス アップデートは、デフォルトでイネーブルです。ヒットレス アップデートをディセーブルにするには、**no platform hardware acl update-mode hitless** コマンドを入力します。

（注） 一部のリリース固有の制限事項については、「eFSU の制約事項」（P.5-2）を参照してください。

- ヒットレス アップデートがイネーブルである場合、FM（機能マネージャ）またはスイッチが最近変更された ACL に対してアップデートを実行すると、各 TCAM エントリのコピーがハードウェアにプログラムされます。ACL が変更されていない場合、ヒットレス アップデート用に予約された TCAM スペースは、最も大きい ACL で使用される TCAM エントリの数と等しくなります。

ポリシーベース ACL（PBACL）

PBACL の制約事項

- PBACL はレイヤ 3 インターフェイスでサポートされています（ルーテッド インターフェイスおよび VLAN インターフェイスなど）。
- PBACL 機能によりサポートされるのは IPv4 ACE だけです。
- PBACL 機能では、Cisco IOS ACL だけがサポートされます。それ以外の機能との組み合わせはサポートされません。キーワード **reflexive** および **evaluate** はサポートされていません。
- PBACL 機能では、名前付き Cisco IOS ACL だけがサポートされます。番号付き ACL はサポートされません。
- ポリシーベース ACL の相互作用機能は Cisco IOS ACL と同じです。

PBACL について

PBACL により、オブジェクト グループ全体にアクセス コントロール ポリシーを適用することができます。オブジェクト グループとはユーザまたはサーバの集合です。

オブジェクト グループを IP アドレスの集合として、またはプロトコル ポートの集合として定義します。それからポリシー（許可や拒否など）をオブジェクト グループに適用するアクセス コントロール エントリ（ACE）を作成します。たとえば、ユーザ グループがあるサーバ グループにアクセスすることを許可するポリシーベース ACE を作成することができます。

グループ名を使用して定義された ACE は、ACE が複数あるのと同じです（オブジェクト グループの各エントリに 1 つ適用されます）。PBACL ACE はシステムにより複数の Cisco IOS ACE に拡張され（グループ内の各エントリに対して 1 つの ACE）、ACE は TCAM に読み込まれます。したがって、PBACL 機能により設定が必要なエントリ数が削減されますが、TCAM 使用率は削減されません。

グループ メンバーシップまたはアクセス グループを使用する ACE の内容に変更を行う場合、TCAM 内の ACE がシステムにより更新されます。次に、更新を開始する変更のタイプを示します。

- グループへのメンバの追加
- グループからのメンバの削除
- アクセス グループを使用する ACE のポリシー文の変更

Cisco IOS ACL 拡張コンフィギュレーション コマンドを使用して PBACL を設定します。通常の ACE と同様に、同じアクセス ポリシーを 1 つまたは複数のインターフェイスと関連付けることができます。

ACE の設定時にオブジェクト グループを使用して送信元、宛先、またはその両方を定義できます。

PBACL の設定方法

- 「PBACL オブジェクト グループを使用する ACL の作成」（P.69-8）
- 「インターフェイスでの PBACL の設定」（P.69-8）

PBACL の IP アドレスのオブジェクト グループの設定

PBACL の IP アドレスのオブジェクト グループを作成または変更するには、次の作業を行います。

	コマンド	目的
ステップ 1	`Router(config)# object-group ip address object_group_name`	オブジェクト グループ名を定義します。IP アドレス オブジェクト グループ 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	`Router(config-ipaddr-ogroup)# {ip_address mask} | {host {name | ip_address} }`	グループのメンバを設定します。メンバは、ネットワーク アドレスにマスクを付加したものか（ホスト名または IP アドレスにより識別される）ホストかのどちらかです。
ステップ 3	`Router(config-ipaddr-ogroup)# {end} | {exit}`	コンフィギュレーション モードを終了するには、**end** コマンドを入力します。 IP アドレス オブジェクト グループ コンフィギュレーション モードを終了するには、**exit** コマンドを入力します。

次に、3 つのホストと 1 つのネットワーク アドレスを含むオブジェクト グループを作成する例を示します。

```
Router(config)# object-group ip address myAG
Router(config-ipaddr-pgroup)# host 10.20.20.1
Router(config-ipaddr-pgroup)# host 10.20.20.5
Router(config-ipaddr-pgroup)# 10.30.0.0 255.255.0.0
```

PBACL のプロトコル ポートのオブジェクト グループの設定

PBACL のプロトコル ポートのオブジェクト グループを作成または変更するには、次の作業を行います。

コマンド	目的
`Router(config)# object-group ip port object_group_name`	オブジェクト グループ名を定義します。ポート オブジェクト グループ コンフィギュレーション モードを開始します。
`Router(config-port-ogroup)# {eq number} | {gt number} | {lt number} | {neq number} | {range number number}`	グループのメンバを設定します。メンバは、ポート番号と等しいまたは等しくない、ポート番号より大きいまたは小さい、またはポート番号の範囲のいずれかです。
`Router(config-port-ogroup)# end | exit`	コンフィギュレーション モードを終了するには、**end** コマンドを入力します。 ポート オブジェクト グループ コンフィギュレーション モードを終了するには、**exit** コマンドを入力します。

次に、プロトコル ポート 100 と、300 以外の 200 より大きいポートと一致するポートのオブジェクト グループを作成する例を示します。

```
Router(config)# object-group ip port myPG
Router(config-port-pgroup)# eq 100
Router(config-port-pgroup)# gt 200
Router(config-port-pgroup)# neq 300
```

PBACL オブジェクト グループを使用する ACL の作成

PBACL オブジェクト グループを使用するように ACL を設定するには、次の作業を行います。

	コマンド	目的
ステップ 1	`Router(config)# ip access-list extended acl_name`	名前を指定して拡張 ACL を定義します。拡張 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 2	`Router(config-ext-nacl)# permit tcp addrgroup object_group_name addrgroup object_group_name`	IP アドレスのオブジェクト グループを送信元ポリシーとして、オブジェクト グループを宛先ポリシーとして使用する、TCP トラフィックの ACE を設定します。
ステップ 3	`Router(config-ext-nacl)# exit`	拡張 ACL コンフィギュレーション モードを終了します。

次に、プロトコル ポートが myPG に指定されたポートと一致する場合に、myAG 内のユーザからのパケットを許可するアクセス リストを作成する例を示します。

```
Router(config)# ip access-list extended my-pbacl-policy
Router(config-ext-nacl)# permit tcp addrgroup myAG portgroup myPG any
Router(config-ext-nacl)# deny tcp any any
Router(config-ext-nacl)# exit
Router(config)# exit
Router# show ip access-list my-pbacl-policy
Extended IP access list my-pbacl-policy
10  permit tcp addrgroup AG portgroup PG any
20  permit tcp any any
Router# show ip access-list my-pbacl-policy expand
Extended IP access list my-pbacl-policy expanded
20 permit tcp host 10.20.20.1 eq 100 any
20 permit tcp host 10.20.20.1 gt 200 any
20 permit tcp host 10.20.20.1 neq 300 any
20 permit tcp host 10.20.20.5 eq 100 any
20 permit tcp host 10.20.20.5 gt 200 any
20 permit tcp host 10.20.20.5 neq 300 any
20 permit tcp 10.30.0.0 255.255.0.0 eq 100 any
20 permit tcp 10.30.0.0 255.255.0.0 gt 200 any
20 permit tcp 10.30.0.0 255.255.0.0 neq 300 any
```

インターフェイスでの PBACL の設定

インターフェイスでの PBACL を設定するには、**ip access-group** コマンドを使用します。このコマンド構文および使用方法は Cisco IOS ACL と同じです。詳細は、「Cisco IOS ACL の制約事項」（P.69-1）を参照してください。

次に、アクセス リスト my-pbacl-policy と VLAN 100 を関連付ける例を示します。

```
Router(config)# int vlan 100
Router(config-if)# ip access-group mp-pbacl-policy in
```

MAC ACL

（注）　VLAN ACL（VACL）で MAC ACL を使用できます。詳細については、第 74 章「VLAN ACL（VACL）」を参照してください。

Protocol-Independent MAC ACL フィルタリングの設定方法

プロトコル独立型 MAC ACL フィルタリングでは、すべての入力トラフィック タイプ（MAC レイヤ トラフィック、IPv4 トラフィック、IPv6 トラフィック、MPLS トラフィックなど）に MAC ACL が適用されます。

次のインターフェイス タイプをプロトコル独立型 MAC ACL フィルタリングに設定できます。

- VLAN インターフェイス
- ルーテッド インターフェイス
- 物理 LAN ポート
- 論理 LAN サブインターフェイス

プロトコル独立型 MAC ACL フィルタリング用に設定されたインターフェイスの MAC ACL によって許可または拒否された入力トラフィックは、出力インターフェイスによって MAC レイヤ トラフィックとして処理されます。プロトコル独立型 MAC ACL フィルタリング用に設定されたインターフェイスの MAC ACL によって許可または拒否されたトラフィックには、出力 IP ACL を適用できません。

プロトコル独立型 MAC ACL フィルタリングを設定するには、次の作業を行います。

	コマンド	目的
ステップ 1	`Router(config)# interface {{vlan vlan_ID}	{type slot/port[.subinterface]}	{port-channel number[.subinterface]}}`	設定するインターフェイスを選択します。
ステップ 2	`Router(config-if)# [no] mac packet-classify {input	output	use {ce_cos {input	output}	dscp {input	output}}}`	インターフェイス上でプロトコル独立型 MAC ACL フィルタリングをイネーブルにします。デフォルトでは、**mac packet-classify** コンフィギュレーション コマンドはディセーブルになります。

- IP アドレスが設定されている VLAN インターフェイス上で、プロトコル独立型 MAC ACL フィルタリングを設定しないでください。
- MAC ACL フィルタリングがイネーブルの場合、RACL、マイクロフロー ポリシングなどの他のプロトコル機能は、すべてハードウェアでは無視されます。

次に、VLAN インターフェイス 4018 をプロトコル独立型 MAC ACL フィルタリングに設定し、設定を確認する例を示します。

```
Router(config)# interface vlan 4018
Router(config-if)# mac packet-classify
Router(config-if)# end
```

```
Router# show running-config interface vlan 4018 | begin 4018
interface Vlan4018
mtu 9216
ipv6 enable
mac packet-classify
end
```

次に、インターフェイス GigabitEthernet 6/1 をプロトコル独立型 MAC ACL フィルタリングに設定し、設定を確認する例を示します。

```
Router(config)# interface gigabitethernet 6/1
Router(config-if)# mac packet-classify
Router(config-if)# end
Router# show running-config interface gigabitethernet 6/1 | begin 6/1
interface GigabitEthernet6/1
mtu 9216
no ip address
mac packet-classify
mpls l2transport route 4.4.4.4 4094
end
```

次に、インターフェイス GigabitEthernet 3/24 およびサブインターフェイス 4000 をプロトコル独立型 MAC ACL フィルタリングに設定し、設定を確認する例を示します。

```
Router(config)# interface gigabitethernet 3/24.4000
Router(config-if)# mac packet-classify
Router(config-if)# end
Router# show running-config interface gigabitethernet 3/24.4000 | begin 3/24.4000
interface GigabitEthernet3/24.4000
encapsulation dot1Q 4000
mac packet-classify
mpls l2transport route 4.4.4.4 4000
end
```

VLAN ベースの MAC QoS フィルタリングをイネーブルにする方法

MAC ACL の VLAN ベースの QoS フィルタリングをグローバルにイネーブルまたはディセーブルにできます。MAC ACL の VLAN ベースの QoS フィルタリングは、デフォルトではディセーブルに設定されています。

MAC ACL の VLAN ベースの QoS フィルタリングをイネーブルにするには、次の作業を行います。

コマンド	目的
Router(config)# mac packet-classify use outer-vlan	MAC ACL の VLAN ベースの QoS フィルタリングをイネーブルにします。MAC ACL の VLAN フィールドは外側 VLAN タグに一致します。 オプションは、**in**（入力 MAC ACL に適用する）および **out**（出力 MAC ACL に適用する）です。

MAC ACL の VLAN ベースの QoS フィルタリングをディセーブルにするには、次の作業を行います。

コマンド	目的
Router(config)# no mac packet-classify use outer-vlan	MAC ACL の VLAN ベースの QoS フィルタリングをディセーブルにします。

MAC ACL の設定

MAC アドレスに基づいて IP、IPX、DECnet、AppleTalk、VINES、または XNS トラフィックをフィルタリングする名前付き ACL を設定できます。

VLAN ベースのフィルタリング、CoS ベースのフィルタリング、またはその両方を行う MAC ACL を設定できます。

MAC ACL の VLAN ベースの QoS フィルタリングを、グローバルにイネーブルまたはディセーブルにできます（デフォルトではディセーブル）。

MAC ACL を設定するには、次の作業を行います。

	コマンド	目的
ステップ 1	`Router(config)# mac host name mac_addr`	（任意）名前 MAC アドレスに割り当てます。
ステップ 2	`Router(config)# mac access-list extended list_name`	MAC ACL を設定します。
ステップ 3	`Router(config-ext-macl)# {permit | deny} {src_mac_mask | {host name src_mac_name} | any} {dest_mac_mask | {host name dst_mac_name} | any} [{protocol_keyword | {ethertype_number ethertype_mask}} [vlan vlan_ID] [cos cos_value]]`	MAC ACL にアクセス コントロール エントリ（ACE）を設定します。送信元および宛先 MAC アドレスは、MAC アドレス マーク、または **mac host** コマンドで作成された名前で指定できます。

- Cisco IOS Release 15.1SY は **vlan** および **cos** キーワードをサポートします。
- MAC ACL の VLAN ベースの QoS フィルタリング用の **vlan** キーワードを、グローバルにイネーブルまたはディセーブルにすることができます（デフォルトではディセーブル）。
- MAC アドレスは、ドット付き 16 進表記の 3 つの 2 バイト値で入力できます。たとえば、0030.9629.9f84 を入力できます。
- MAC アドレス マスクは、ドット付き 16 進表記の 3 つの 2 バイト値で入力できます。1 のビットをワイルドカードとして使用します。たとえば、アドレスを完全に一致させるには、0000.0000.0000 を使用します（0.0.0 として入力できます）。
- EtherType および EtherType マスクを 16 進値で入力できます。
- protocol パラメータなしのエントリはどのプロトコルとも一致します。
- ACL エントリは、入力順にスキャンされます。最初に一致したエントリが使用されます。パフォーマンスを向上させるには、最もよく使用されるエントリを ACL の先頭に置きます。
- ACL の末尾に **permit any any** エントリを明示的に指定する場合を除いて、ACL の末尾には暗黙的な **deny any any** エントリが存在します。
- 新しいエントリはすべて既存のリストの最後に置かれます。リストの中間にエントリを追加することができません。
- 次に、EtherType の値と対応するプロトコル キーワードを示します。
  - 0x0600 - xns-idp - Xerox XNS IDP
  - 0x0BAD - vines-ip - Banyan VINES IP
  - 0x0baf - vines-echo - Banyan VINES Echo
  - 0x6000 - etype-6000 - DEC 未割り当て、実験的
  - 0x6001 - mop-dump - DEC Maintenance Operation Protocol（MOP; メンテナンス オペレーション プロトコル）ダンプ / ロード補助
  - 0x6002 - mop-console - DEC MOP リモート コンソール

- 0x6003 - decnet-iv - DEC DECnet Phase IV Route
- 0x6004 - lat - DEC Local Area Transport（LAT; ローカルエリア トランスポート）
- 0x6005 - diagnostic - DEC DECnet Diagnostics
- 0x6007 - lavc-sca - DEC Local-Area VAX Cluster（LAVC）、SCA
- 0x6008 - amber - DEC AMBER
- 0x6009 - mumps - DEC MUMPS
- 0x0800 - ip - 不正な形式、無効、または意図的に壊された IP フレーム
- 0x8038 - dec-spanning - DEC LANBridge Management
- 0x8039 - dsm - DEC DSM/DDP
- 0x8040 - netbios - DEC PATHWORKS DECnet NETBIOS Emulation
- 0x8041 - msdos - DEC Local Area System Transport
- 0x8042 - etype-8042 - DEC 未割り当て
- 0x809B - appletalk - Kinetics EtherTalk（AppleTalk over Ethernet）
- 0x80F3 - aarp - Kinetics AppleTalk Address Resolution Protocol（AARP）

次に、mac_layer という名前の MAC レイヤ ACL を作成する例を示します。この ACL は、送信元アドレスが 0000.4700.0001、宛先アドレスが 0000.4700.0009 である dec-phase-iv トラフィックを拒否しますが、それ以外のトラフィックをすべて許可します。

```
Router(config)# mac access-list extended mac_layer
Router(config-ext-macl)# deny 0000.4700.0001 0.0.0 0000.4700.0009 0.0.0 dec-phase-iv
Router(config-ext-macl)# permit any any
```

ARP ACL

ここでは、ARP ACL の設定方法について説明します。ARP トラフィック（EtherType 0x0806）をフィルタリングする名前付き ACL を設定できます。ARP ACL を設定するには、次の作業を行います。

	コマンド	目的
ステップ 1	`Router(config)# arp access-list list_name`	ARP ACL を設定します。
ステップ 2	`Router(config-arp-nacl)# {permit | deny} {ip {any | host sender_ip | sender_ip sender_ip_wildcardmask} mac any`	ARP ACL にアクセス コントロール エントリ（ACE）を設定します。

- ここでは、PFC によってハードウェアでサポートされる ARP ACL 構文について説明します。疑問符（?）を入力した場合に CLI ヘルプで表示されるその他の ARP ACL 構文はサポートされず、QoS の ARP トラフィックのフィルタリング処理にも使用できません。
- ACL エントリは、入力した順序に従ってスキャンされます。最初に一致したエントリが使用されます。パフォーマンスを向上させるには、最もよく使用されるエントリを ACL の先頭に置きます。
- リストの末尾に **permit ip any mac any** エントリを明示的に指定する場合を除いて、ACL の末尾には暗黙的な **deny ip any mac any** エントリが存在します。
- 新しいエントリはすべて既存のリストの最後に置かれます。リストの中間にエントリを追加することができません。
- PFC は IP ACL を ARP トラフィックに適用しません。

- ARP トラフィックには、マイクロフロー ポリシングを適用できません。

次に、arp_filtering という名前の ARP ACL を作成する例を示します。この ACL は、IP アドレスが 1.1.1.1 から始まるトラフィックだけを許可します。

```
Router(config)# arp access-list arp_filtering
Router(config-arp-nacl)# permit ip host 1.1.1.1 mac any
```

最適化された ACL ロギング

OAL の制約事項

- OAL キャプチャと VACL キャプチャには互換性がありません。スイッチに両方の機能を設定しないでください。OAL が設定された状態で、SPAN を使用してトラフィックをキャプチャします。
- OAL は、VACL キャプチャ、合法的傍受（LI）、および IPv6 学習などのキャプチャを使用して他の機能との競合をチェックします。
- OAL は IPv4 ユニキャスト パケットだけをサポートしています。
- OAL はポート ACL（PACL）ではサポートされません。
- OAL は、次のものに対してはハードウェアでのサポートをしていません。
  - 再帰 ACL
  - 他の機能（QoS など）のトラフィックのフィルタ処理に使用される ACL
  - Unicast Reverse Path Forwarding（uRPF; ユニキャスト RPF）チェック例外のための ACL
  - 例外パケット（Time To Live（TTL）障害や MTU 障害など）
  - IP オプションが指定されたパケット
  - レイヤ 3 でルータへのアドレスが指定されたパケット
  - ICMP 到達不能メッセージを生成するために RP へ送信されるパケット
  - ハードウェアでは加速されず、機能によって処理されるパケット

OAL について

OAL は、ACL ロギングをハードウェアでサポートしています。OAL を設定しないかぎり、ロギングを必要とするパケットは、RP のソフトウェアで完全に処理されます。OAL では、PFC または DFC のハードウェアでパケットの許可またはドロップを行います。情報は最適化ルーチンを使用して RP に送信され、ロギング メッセージが生成されます。

OAL の設定方法

- 「OAL 情報の表示」（P.69-15）
- 「キャッシュされた OAL エントリのクリア」（P.69-15）

OAL グローバル パラメータの設定

OAL グローバル パラメータを設定するには、次の作業を行います。

コマンド	目的
`Router(config)# logging ip access-list cache {{entries number_of_entries} | {interval seconds} | {rate-limit number_of_packets} | {threshold number_of_packets}}`	OAL グローバル パラメータを設定します。

- **entries** *number_of_entries*
  - キャッシュされるエントリの最大数を設定します。
  - 範囲：0 ～ 1,048,576（カンマを含めないで入力）
  - デフォルト：8192
- **interval***seconds*
  - ログのためにエントリが送信されるまでの最大時間を設定します。この時間中エントリが非アクティブの場合、キャッシュから削除されます。
  - 範囲：5 ～ 86,400（1440 分つまり 24 時間、カンマを含めないで入力）
  - デフォルト：300 秒（5 分）
- **rate-limit** *number_of_packets*
  - ソフトウェアで 1 秒間にログに記録されるパケット数を設定します。
  - 範囲：10 ～ 1,000,000（カンマを含めないで入力）
  - デフォルト：0（レート制限がオフになり、すべてのパケットがログに記録されます）
- **threshold** *number_of_packets*
  - エントリがログに記録されるまでに一致するパケット数を設定します。
  - 範囲：1 ～ 1,000,000（カンマを含めないで入力）
  - デフォルト：0（一致パケット数に達してもログの記録は開始されません）

インターフェイスでの OAL の設定

インターフェイスで OAL を設定するには、次の作業を行います。

	コマンド	目的
ステップ 1	`Router(config)# interface {type slot/port}`	設定するインターフェイスを指定します。
ステップ 2	`Router(config-if)# logging ip access-list cache in`	インターフェイスの入力トラフィックに対して OAL をイネーブルにします。
ステップ 3	`Router(config-if)# logging ip access-list cache out`	インターフェイスの出力トラフィックに対して OAL をイネーブルにします。

OAL 情報の表示

OAL 情報を表示するには、次の作業を行います。

コマンド	目的
`Router# show logging ip access-list cache`	OAL 情報を表示します。

キャッシュされた OAL エントリのクリア

キャッシュされた OAL エントリをクリアするには、次の作業を行います。

コマンド	目的
`Router# clear logging ip access-list cache`	キャッシュされた OAL エントリをクリアします。

ACL のドライ ランのサポート

ドライ ランのサポートの制約事項

- ドライ ランは IPv4 RACL に対してだけサポートされており、インターフェイスにだけ適用できます。
- ドライ ランは名前付き ACL（標準または拡張）でのみサポートされ、番号付き ACL ではサポートされません。
- 1 つのドライ ラン セッションだけが、ドライ ラン セッション上の 1 つまたは複数の ACL と同時に割り当てられます。
- ACL がコンフィギュレーション モードで変更されると、1 つまたは複数のドライ ラン セッション ACL は削除されます。
- ドライ ラン セッションを終了しても、既存の設定はクリアされません。新しい設定を開始する前に、既存のセッションをクリアします。
- 検証プロセス中に設定またはハードウェアの変更がある場合、検証プロセスは中断される可能性があります。
- ドライ ラン モードは、実行コンフィギュレーションに対する変更のコミットをサポートしません。
- ドライ ランは QoS ポリシーで使用される ACL ではサポートされません。
- ドライ ランは、ハードウェア統計情報をイネーブルにした ACL ではサポートされません。
- ドライ ランセッションの進行中に、別の Telnet セッションを使用してスイッチにアクセスできます。

ドライ ランのサポートについて

他のリリースでは、他の機能とともに設定されたインターフェイスに既存の機能の新しい機能を適用する場合、および新しい機能が TCAM に適合しない場合、既存の機能も影響を受け、TCAM から削除されます。機能を段階的にアップデートし、インストールすることなしに機能が TCAM に適合するかどうかを判断するため、スイッチはドライ ランをサポートします。これによって、アプリケーションは通常の要求を送信して、要求を正常にプログラムすることができるかどうかをテストできます。スイッチは、ドライ ラン要求を受信し、その要求に必要な総 TCAM リソースを計算し、使用可能な空きリソースに対してこれらのリソースを比較します。要求が正常に適合した場合は、スイッチが成功を返し、そうでない場合は、失敗を返します。ドライ ラン サポートは、アプリケーションがインテリジェントな判断を行うために役立ちます。

ACL のドライ ラン サポートの設定方法

ドライ ラン サポートを設定するには、次の作業を行います。

	コマンド	目的
ステップ 1	`Router(config)# configure session session_name`	コンフィギュレーション セッションを作成し、ドライ ラン モードを開始します
ステップ 2	`Router(dry-run-config)# {default | exit | ip | no | validate}`	ドライ ラン セッションを設定するオプションを選択します。
ステップ 3	`Router(dry-run-config)# ip access-list {extended | standard} acl_name`	ACL タイプを選択します。

次に、既存の ACL RACL10K でセッションにドライ ラン サポートを設定する例を示します。

```
Router(config)# configure session test
Router(dry-run-config)# ip access-list extended RACL10K
Router(dr-config-ext-nacl)# permit tcp host 10.20.0.1 host 11.20.0.1
Router(dr-config-ext-nacl)# permit tcp host 10.20.0.2 host 11.20.0.2
Router(dr-config-ext-nacl)# permit tcp host 10.20.0.3 host 11.20.0.3
Router(dr-config-ext-nacl)# permit tcp host 10.20.0.4 host 11.20.0.4
Router(dr-config-ext-nacl)# permit tcp host 10.20.0.5 host 11.20.0.5

Router(dr-config-ext-nacl)# exit

Router(dry-run-config)# validate

Router(dry-run-config)# exit
Router#
.Feb 23 2010 13:46:52.528: Validation is in progress !!
.Feb 23 2010 13:46:52.528: Please try again later.
.Feb 23 2010 13:46:53.136: %FM-6-SESSION_VALIDATION_RESULT_INFO: Session Validation Result
:  "Validation Completed Sucessfully."
. Please use 'show config session test status' to get more details of the config
validation status

Router# show configuration session test status
====================================
Status of last config validation:
Timestamp: 2010-02-23@13:46:51
====================================
SLOT = [1]    Result = Configuration will fit in TCAM
SLOT = [2]    Result = Configuration will fit in TCAM
SLOT = [5]    Result = Configuration will fit in TCAM
```

```
Router# clear configuration session test

Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.

Router(config)# ip access-list extended RACL10K
Router(config-ext-nacl)# permit tcp host 10.20.0.1 host 11.20.0.1
Router(config-ext-nacl)# permit tcp host 10.20.0.2 host 11.20.0.2
Router(config-ext-nacl)# permit tcp host 10.20.0.3 host 11.20.0.3
Router(config-ext-nacl)# permit tcp host 10.20.0.4 host 11.20.0.4
Router(config-ext-nacl)# permit tcp host 10.20.0.5 host 11.20.0.5
Router(config-ext-nacl)# end

Router#
```

ハードウェア ACL 統計情報

ハードウェア ACL 統計情報の制約事項

- ハードウェア ACL 統計情報は、入力と出力の両方向の IPv4 および IPv6 RACL でサポートされます。
- IPv4 では、ハードウェア ACL 統計情報は、番号付きと名前付きの両方の ACL でサポートされます。
- ハードウェア統計情報は、60 秒ごとにハードウェアをポーリングすることによって取得されます。
- ハードウェア統計情報はステートフル スイッチオーバー（SSO）後に失われます。
- ハードウェア統計情報は ACL ごとに保持されます。複数のインターフェイスが同じ ACL を使用している場合、統計情報は集約されます。
- ODM（Order-Dependent Merge）の最適化をイネーブルにすると、ハードウェア統計情報はディセーブルになります。

ハードウェア ACL 統計情報について

ハードウェア ACL 統計情報を使用して、特定の ACL のハードウェア カウンタは収集され、集約され、IOS のアクセス リストの出力に表示されます。

ACE ヒット カウントはハードウェアから取得され、次のコマンドを使用して表示できます。

show ip access-list および **show ipv6 access-list**

ハードウェア統計は、デフォルトではディセーブルです。ハードウェア統計をイネーブルまたはディセーブルにするには、ハードウェア統計情報のコマンドを入力します。

ハードウェア ACL 統計情報の設定方法

次に、ACL racl1 のハードウェア統計情報をイネーブルにする例を示します。

```
Router(config)# ip access-list extended racl1
Router(config-ext-nacl)# [no] hardware statistics
Router(config-ext-nacl)# permit ip host 1.1.1.1 host 2.2.2.2
Router(config-ext-nacl)# permit ip host 3.3.3.3 host 4.4.4.4
Router(config-ext-nacl)# deny ip any any
Router(config-ext-nacl)# end
```

次に、ACL racl1 のハードウェア統計情報を表示する例を示します。

```
Router# show ip access-lists racl1
Extended IP access list racl1
    hardware statistics
    10 permit ip host 1.1.1.1 host 2.2.2.2
acl hw hit count 5
    20 permit ip host 3.3.3.3 host 4.4.4.4
acl hw hit count 20
    30 deny ip any any
```

各 ACE のハードウェア統計情報は、**acl hw hit count** の文字列の後に表示され、ハードウェアでスイッチングされたパケットの数を示します。

ヒント Cisco Catalyst 6500 シリーズ スイッチの詳細（設定例およびトラブルシューティング情報を含む）については、次のページに示されるドキュメントを参照してください。

http://www.cisco.com/en/US/products/hw/switches/ps708/tsd_products_support_series_home.html

技術マニュアルのアイデア フォーラムに参加する