CHAPTER 1

セキュリティ アクセス コントロール リストの設定

この章では、Cisco 4700 Series Application Control Engine（ACE）アプライアンスのセキュリティ アクセス コントロール リスト（ACL）を設定する方法を説明します。ACL を利用すると、トラフィックに対するフィルタリングやネットワーク接続の制御を行うことが可能で、ネットワークに基本的セキュリティを与えることができます。主な内容は、次のとおりです。

- ACL の概要
- ACL 設定のクイック スタート
- ACL の設定
- オブジェクト グループによるアクセス制御リストの簡易化
- インターフェイスへの ACL の適用
- コンテキスト内のすべてのインターフェイスへの ACL のグローバルな適用
- ACL によるトラフィックのフィルタリング
- ACL の設定例
- ACL の設定情報および統計情報の表示
- ACL 統計情報のクリア

ACL の概要

ACL は ACL エントリと呼ぶ一連のステートメントで構成されます。このエントリはネットワーク トラフィックのプロファイルを定義するものです。各エントリは、エントリ内で指定したネットワークの一部エリアに対して、ネットワーク トラフィック（インバウンドおよびアウトバウンド）を許可あるいは拒否します。各エントリにはフィルタエレメントとして、送信元アドレス、宛先アドレス、プロトコル、プロトコル特有のパラメータ（例：：ポート）などの判断基準を含めることができます。

各 ACL の末尾には暗黙の全エントリ拒否が存在するため、コネクションを許可するインターフェイスそれぞれに対して ACL を設定する必要があります。これを行わないと、ACE はそのインターフェイス上のすべてのトラフィックを拒否してしまいます。

ACL を利用するとネットワーク コネクションのセットアップ内容を制御できます。パケットをひとつひとつ処理する必要はありません。このような ACL を通常、セキュリティ ACL と呼んでいます。

ACL はその他の機能（たとえばセキュリティ、ネットワーク アドレス トランスレーション [NAT] 、サーバ ロード バランス [SLB] など）の一部として設定できます。ACE はこれら個々の ACL を 1 つの大きな ACL にマージします。この ACL はマージド ACL と呼ばれます。次にマージド ACL は ACL コンパイラにより解析され、ACL 参照メカニズムが生成されます。このマージド ACL への合致が 1 回発生するたびに、複数のアクションを発動させることができます。

たとえば、ACL を利用すると、ある VLAN で E メール トラフィックすべてを許可すると、同時に、Telnet トラフィックを遮断することが可能になります。また、ネットワークの一部エリアに対して、あるクライアントにはアクセスを許可し、別のクライアントには拒否するような場合にも ACL を利用できます。

ACL の設定では、あるインターフェイスを通るトラフィックを制御するためには、ACL をそのインターフェイスに適用しなくてはなりません。ある ACL をインターフェイスに適用することにより、その ACL とそのエントリがそのインターフェイスに対して割り当てられます。

拡張 ACL は、インターフェイスの各送信方向（インバウンドまたはアウトバウンド）に対して 1 つだけ適用できます。また、複数のインターフェイスに同一の ACL を適用することもできます。EtherType ACL はインバウンド方向に 1 つだけ、かつレイヤ 2 インターフェイスにのみ適用できます。

このセクションの内容は、次のとおりです。

- ACL の種類と用途
- ACL の注意事項

ACL の種類と用途

ACE では次の 2 種類の ACL を設定できます。

- 拡張 - IP トラフィック用のコントロール ネットワーク アクセス
- EtherType - 非 IP トラフィック用のコントロール ネットワーク アクセス

（注）　ACE は標準 ACL を明示的にはサポートしていません。標準 ACL を設定するには、拡張 ACL で宛先アドレスを **any** に指定し、かつポートを指定せずにおきます。拡張 ACL の設定の詳細については「拡張 ACL の設定」を参照してください。

ACL の注意事項

ここでは、ご使用のネットワークで ACL を設定、使用する際に考慮すべき注意事項を説明します。このセクションの内容は、次のとおりです。

- ACL エントリの順序
- ACL における暗黙の拒否
- ACL と ACL エントリの最大数

ACL エントリの順序

1 つの ACL は 1 つまたは複数のエントリで構成されます。ACL の種類に応じて、送信元アドレスと宛先アドレスのほか、プロトコル、ポート（TCP または UDP）、ICMP のタイプ、ICMP のコード、またはマッチング基準としての EtherType を指定できます。デフォルトでは、ACE は各 ACL エントリを ACL の末尾に追加します。ACL 内部では各エントリの位置を指定することもできます。

エントリの順序は重要です。あるコネクションを許可するか、拒否するかを ACE が決定する際、ACE は ACL エントリが並ぶ順序に従って、エントリを 1 つ 1 つ参照しながらパケットを検査します。適合するものがあると、ACE はそれ以上のエントリの確認をやめます。たとえば、明示的にすべてのトラフィックを許可するエントリを ACL の先頭に 1 つ作成した場合は、ACE はその ACL 内にある他のステートメントをまったく確認しなくなります。

ACL における暗黙の拒否

すべての ACL の末尾には、暗黙の拒否を行うエントリが存在します。そのため、明示的に許可しないかぎりトラフィックは通過できません。たとえば、特定の IP アドレスのユーザを除いたすべてのユーザに対して、あるネットワークへ ACE を通ってアクセスすることを許可する場合は、その特定の IP アドレスを 1 つのエントリにより拒否するとともに、別のエントリで他のすべての IP アドレスを許可する必要があります。

ACL と ACL エントリの最大数

ACE では、最大 8,192 の一意の ACL と 64,000 の ACL エントリがサポートされます。ポート番号の範囲が大きいものや、複数のネットワークに重複するもの（たとえば、あるエントリが 10.0.0.0/8 を指定し、別のエントリが 10.1.1.0/24 を指定するもの）など、ACL によっては他よりも多くのメモリを消費する場合があります。そのため、ACL の種類にもよりますが、ACE がサポートできるエントリ数の実際の限界は、64,000 より少なくなる場合があります。

ACL エントリ中のオブジェクト グループを使用すると、実際に入力する ACL エントリは少なくなりますが、オブジェクト グループなしでエントリを入力したときと同じ数の*拡張* ACL エントリが使用されます。拡張 ACL エントリの数は、システムの上限に向けてカウントされます。ACL 中の拡張 ACL エントリの数を確認するには、**show access-list** *name* コマンドを使用します。

ACE のメモリ限界を超えると、アプライアンスで Syslog メッセージが生成され、**show interface vlan** *number* コマンドで出力できる Download Failures カウンタがインクリメントされます。その場合でも設定内容は 実行コンフィギュレーション ファイルに残り、当該のインターフェイスは有効に動作します。ACL エントリは、失敗した設定を実行する前と同じ状態を維持します。

たとえば、10 エントリの新しい ACL を追加しようとして、ACE がメモリ不足を生じたために 6 番めのエントリの追加に失敗すると、ACE は入力に成功した 5 つのエントリを削除します。

ACL 設定のクイック スタート

ACL の設定に必要な手順の概要を表 1-1 に示します。各ステップには CLI、または作業を完了するのに必要な手順の参照が含まれます。CLI コマンドに関連した各機能すべてのオプションの詳細については、次の表 1-1 を参照してください。

表 1-1　ACL 設定のクイック スタート

作業内容とコマンドの例

1. 複数のコンテキストで操作している場合、対象のコンテキストで操作しているかどうかを CLI プロンプトを確認します。必要に応じて、正しいコンテキストに変更します。
`host1/Admin# changeto C1`
`host1/C1#`
ここからは特に明記しないかぎり、表中の例では管理コンテキストを使用します。コンテキストの作成方法の詳細については、『*Cisco 4700 Series Application Control Engine Appliance Virtualization Configuration Guide*』を参照してください。

2. コンフィギュレーション モードを開始します。
`host1/Admin# config`
`host1/Admin(config)#`

3. ACL を作成します。 `host1/Admin(config)# access-list INBOUND extended deny ip 192.168.12.0 255.255.255.0 any`
4. 用途に応じて、同一のアクセス リスト名を使用して ACL にエントリを追加します。たとえば、次のように入力します。 `host1/Admin(config)# access-list INBOUND extended permit ip any any`

5. ACL を個々のインターフェイスに適用します。
`host1/Admin(config)# interface vlan 10`
`host1/Admin(config-if)# access-group input INBOUND`

6. 別の方法としては、1 つのコンテキスト内のすべてのインターフェイスに対して、1 つの ACL をグローバルに適用することもできます。 `host1/Admin(config)# access-group input INBOUND`

7. （任意）設定の変更内容をフラッシュ メモリに保存します。
`host1/Admin(config)# exit`
`host1/Admin# copy running-config startup-config`

8. ACL 設定情報を表示して確認します。 `host1/Admin# show running-config access-list`

ACL の設定

このセクションの内容は、次のとおりです。

- 拡張 ACL の設定
- 拡張 ACL へのコメント設定
- EtherType ACL の設定
- 全エントリの番号付けの変更

拡張 ACL の設定

拡張 ACL を利用すると、トラフィックの送信元と宛先両方の IP アドレスを指定できるほか、次のパラメータも指定できます。

- プロトコル
- TCP ポートまたは UDP ポート
- ICMP のタイプとコード

これらのパラメータは **access-list** コマンドを使用すると直接指定できます。またはパラメータごとにオブジェクト グループを使用することもできます。オブジェクト グループの詳細については、「オブジェクト グループによるアクセス制御リストの簡易化」を参照してください。

ACE は確立したコネクションに対してすべての返信トラフィックを許可するため、TCP、UDP、ICMP コネクションに対しては、宛先側インターフェイスに ACL を適用しなくても返信トラフィックが許可されます。

（注） ACE は標準 ACL を明示的にはサポートしていません。標準 ACL を設定するには、拡張 ACL で宛先アドレスを **any** に指定し、かつポートを指定せずにおきます。

ヒント 設定作業中に見やすくなるように、ACL の名称を大文字で入力します。ACL にはインターフェイスによる名称（INBOUND など）か、目的別の名称（NO_NAT や VPN など）をつけることを推奨します。

拡張 ACL を作成するには、設定モードで **access-list extended** コマンドを使用します。拡張 ACL には大きく分けて次の 2 種類があります。

- 非 ICMP の ACL
- ICMP ACL

ネットワーク接続は、IP プロトコル、送信元および宛先アドレス、TCP ポートまたは UDP ポートに基づいて許可または拒否することができます。非 ICMP 拡張 ACL の構文は次のとおりです。

access-list *name* [**line** *number*] **extended** {**deny** | **permit**}
{*protocol* {**any** | **host** *src_ip_address* | *src_ip_address netmask* | **object-group** *net_obj_grp_name*} [*operator port1* [*port2*]] {**any** | **host** *dest_ip_address* | *dest_ip_address netmask* | **object-group** *net_obj_grp_name*} [*operator port3* [*port4*]]}
| {**object-group** *service_obj_grp_name*} {**any** | **host** *src_ip_address*
| *src_ip_address netmask* | **object-group** *net_obj_grp_name*} {**any** | **host** *dest_ip_address* | *dest_ip_address netmask* | **object-group** *net_obj_grp_name*}

また、ネットワーク接続は ICMP タイプ（echo、echo-reply、unreachable など）に基づいて許可または拒否することもできます。ICMP 拡張 ACL の構文は次のとおりです。

access-list *name* [**line** *number*] **extended** {**deny** | **permit**}
{**icmp** {**any** | **host** *src_ip_address* | *src_ip_address netmask* | **object_group** *net_obj_grp_name*} {**any** | **host** *dest_ip_address* | *dest_ip_address netmask* | **object_group** *network_grp_name*} [*icmp_type* [**code** *operator code1* [*code2*]]]}
| {**object-group** *service_obj_grp_name*} {**any** | **host** *src_ip_address*
| *src_ip_address netmask* | **object-group** *net_obj_grp_name*} {**any** | **host** *dest_ip_address* | *dest_ip_address netmask* | **object-group** *net_obj_grp_name*}

キーワード、オプション、および引数は次のとおりです。

- *name* - ACL を識別する重複のない名称。「"」記号で囲まずに、最大 64 文字までの英数字でスペースを入れずにテキスト文字列を入力します。
- **line** *number* - （任意）エントリを入力する ACL 内の位置として行番号を指定します。エントリの位置は ACL 内でのエントリ参照の順番に影響します。エントリの行番号を設定しないと、ACE はデフォルトの値を加えた行番号をそのエントリに適用したあとに ACL の末尾に追加します。
- **extended** - 拡張 ACL を指定します。拡張 ACL を利用すると、宛先 IP アドレスとサブネット マスクの他にも、標準 ACL では利用できない他のパラメータを指定できます。
- **deny** - 割り当てたインターフェイスのコネクションを遮断します。
- **permit** - 割り当てたインターフェイスのコネクションを許可します。
- *protocol* - IP プロトコルの名前または番号。表 1-2 に記載のプロトコル名称、または IP プロトコル番号として、整数 0 ～ 255 を入力します。

表 1-2　サポートされているプロトコルのキーワードと番号

プロトコル名	プロトコル番号	説明
ah	**51**	認証ヘッダ
eigrp	**88**	拡張 IGRP
esp	**50**	カプセル化したセキュリティ ペイロード
gre	**47**	汎用ルーティング カプセル化
icmp	**1**	インターネット制御通知プロトコル
igmp	**2**	インターネット グループ管理プロトコル
ip	**任意**	Internet Protocol
ip-in-ip	**4**	IP-in-IP レイヤ 3 トンネリング プロトコル
ospf	**89**	Open Shortest Path First
pim	**103**	Protocol Independent Multicast
tcp	**6**	Transmission Control Protocol
udp	**17**	User Datagram Protocol

- **any** - 任意の送信元からのネットワーク トラフィックを指定します。
- **host** *src_ip_address* - ネットワーク トラフィックの送信元であるホストの IP アドレスを指定します。このキーワードと引数は、1 つの IP アドレスから送信されるネットワーク トラフィックを指定します。
- *src_ip_address netmask* - 送信元からのトラフィック（IP アドレスとネットワーク マスクで定義）。送信元 IP アドレスの範囲から送信されるネットワーク トラフィックを指定する引数です。
- **object-group** *net_obj_grp_name* - 既存ネットワーク オブジェクト グループの識別名を指定します。詳細については、「オブジェクト グループによるアクセス制御リストの簡易化」を参照してください。
- *operator* - （任意）TCP、TCP-UDP、および UDP プロトコルにおける送信元と宛先のポート番号を比較する際のオペランドです。演算子は次のとおりです。
  - **eq** - 等しい
  - **gt** - より大きい
  - **lt** - より小さい
  - **neq** - 等しくない
  - **range** - ポート番号を含む領域。この演算子では 2 番めのポート番号が領域上限を定義します。
- *port1* [*port2*] - サービスへのアクセスを許可または拒否する TCP または UDP 送信元ポートの名称または番号。0 ～ 65535 の整数を入力します。ポートの包含範囲を入力するには、2 つのポート番号を入力してください。*port2* は *port1* より大きいか、等しくなければなりません。well-known TCP ポートの名称と番号は表 1-3 を、well-known UDP ポートの名称と番号は表 1-4 を参照してください。

表 1-3　Well-known TCP ポート番号およびキーワード

キーワード	ポート番号	説明
aol	**5190**	America-Online
bgp	**179**	Border Gateway Protocol
chargen	**19**	Character Generator プロトコル
citrix-ica	**1494**	Citrix 社 Independent Computing Architecture プロトコル

表 1-3　Well-known TCP ポート番号およびキーワード （続き）

キーワード	ポート番号	説明
cmd	**514**	exec と同じですが、自動ログインを実行
ctiqbe	**2748**	Computer Telephony Interface Quick Buffer Encoding
daytime	**13**	Daytime プロトコル
discard	**9**	Discard プロトコル
domain	**53**	Domain Name System（DNS; ドメイン ネーム システム）
echo	**7**	Echo プロトコル
exec	**512**	Exec プロトコル（RSH）
finger	**79**	Finger プロトコル
ftp	**21**	File Transfer Protocol（FTP; ファイル転送プロトコル）
ftp-data	**20**	FTP データ コネクション
gopher	**70**	Gopher
h323	**1720**	H.323 発呼信号
hostname	**101**	NIC ネームサーバ
http	**80**	Hypertext Transfer Protocol（HTTP; ハイパーテキスト転送プロトコル）
https	**443**	HTTP over TLS/SSL
ident	**113**	Ident プロトコル
imap4	**143**	Internet Message Access Protocol, version 4
irc	**194**	Internet Relay Chat（IRC; インターネット リレー チャット）
kerberos	**88**	Kerberos プロトコル
klogin	**543**	Kerberos ログイン
kshell	**544**	Kerberos シェル
ldap	**389**	Lightweight Directory Access Protocol
ldaps	**636**	LDAP over TLS/SSL
login	**513**	ログイン（rlogin）

表 1-3　Well-known TCP ポート番号およびキーワード （続き）

キーワード	ポート番号	説明
lotusnotes	**1352**	IBM Lotus Notes
lpd	**515**	印刷サービス
matip-a	**350**	Mapping of Airline Traffic over Internet Protocol（MATIP）Type A
netbios-ssn	**139**	NetBIOS セッション サービス
nntp	**119**	ネットワーク ニュース トランスポート プロトコル
pcanywhere-data	**5631**	PC Anywhere データ
pim-auto-rp	**496**	PIM Auto-RP
pop2	**109**	POP v2
pop3	**110**	POP v3
pptp	**1723**	Point-to-Point Tunneling Protocol、RFC 2637
rtsp	**554**	Real Time Streaming Protocol
sip	**5060**	Session Initiation Protocol
skinny	**2000**	Cisco Skinny Client Control Protocol（SCCP）
smtp	**25**	Simple Mail Transfer Protocol（SMTP; シンプル メール転送プロトコル）
sqlnet	**1521**	Structured Query Language Network
ssh	**22**	Secure Shell（SSH）
sunrpc	**111**	Sun リモート プロシージャ コール
tacacs	**49**	Terminal Access Controller Access Control System
talk	**517**	Talk プロトコル
telnet	**23**	Telnet
time	**37**	Time プロトコル
uucp	**540**	Unix-to-Unix Copy Program
whois	**43**	Nicname
www	**80**	World Wide Web（HTTP）

表 1-4　Well-known UDP キーワードおよびポート番号

キーワード	ポート番号	説明
biff	**512**	メール受信通知
bootpc	**68**	Bootstrap Protocol クライアント
bootps	**67**	Bootstrap Protocol サーバ
discard	**9**	Discard プロトコル
dnsix	**195**	DNSIX セキュリティ プロトコル監査機能（dn6-nlm-aud）
domain	**53**	Domain Name System（DNS; ドメイン ネーム システム）
echo	**7**	Echo プロトコル
isakmp	**500**	Internet Security Association Key Management Protocol
kerberos	**88**	Kerberos プロトコル
mobile-ip	**434**	Mobile IP 登録
nameserver	**42**	Host Name Server
netbios-dgm	**138**	NetBIOS データグラム サービス
netbios-ns	**137**	NetBIOS ネーム サービス
netbios-ssn	**139**	NetBIOS セッション サービス
ntp	**123**	Network Time Protocol
pcanywhere-status	**5632**	PC Anywhere ステータス
radius	**1812**	Remote Authentication Dial-in User Service（RADIUS）
radius-acct	**1813**	RADIUS アカウンティング
rip	**520**	Routing Information Protocol
snmp	**161**	Simple Network Management Protocol
snmptrap	**162**	SNMP トラップ
sunrpc	**111**	Sun リモート プロシージャ コール
syslog	**514**	システム ログ
tacacs	**49**	Terminal Access Controller Access Control System

表 1-4　Well-known UDP キーワードおよびポート番号 （続き）

キーワード	ポート番号	説明
talk	**517**	Talk プロトコル
tftp	**69**	Trivial File Transfer Protocol（TFTP）
time	**37**	Time プロトコル
who	**513**	rwho サービス
wsp	**9200**	コネクションレス型無線セッション プロトコル（WSP）
wsp-wtls	**9202**	セキュアなコネクションレス型 WSP
wsp-wtp	**9201**	接続ベースの WSP
wsp-wtp-wtls	**9203**	セキュアな接続ベースの WSP
xdmcp	**177**	X Display Manager Control Protocol

- *dest_ip_address netmask* - パケットの送信先となるネットワークまたはホストの IP アドレスおよび宛先 IP アドレスに適用するネットワーク マスク ビット。宛先 IP アドレスの幅を指定する引数です。
- **any** - すべての送信先のネットワーク トラフィックを指定します。
- **host** *dest_address* - フロー内のパケットの宛先の IP アドレスとサブネットマスク。このキーワードと引数は、単一 IP アドレスへ送信されるネットワーク トラフィックを指定します。
- *operator* - （任意）TCP および UDP プロトコルにおける送信元と宛先のポート番号を比較する際のオペランドです。演算子は次のとおりです。
  - **lt** - より小さい
  - **gt** - より大きい
  - **eq** - 等しい
  - **neq** - 等しくない
  - **range** - ポート番号を含む領域。この演算子では 2 番めのポート番号が領域上限を定義します。
- *port3* [*port4*] - サービスへのアクセスを許可または拒否する TCP または UDP の宛先ポートの名称または番号。オプションのポート範囲を入力するには、ポート番号を 2 つ入力してください。*port4* は *port3* より大きいか、等しくなければなりません。well-known ポートは表 1-3 で一覧できます。

- **object-group** *service_obj_grp_name* -（任意）既存のサービス オブジェクト グループの識別名を指定します。詳細については、「オブジェクト グループによるアクセス制御リストの簡易化」を参照してください。
- *icmp_type* -（任意）ICMP 通知のタイプ。表 1-5 に記載の ICMP コード番号を整数で入力するか、ICMP タイプを入力します。

表 1-5　ICMP タイプ

ICMP コード番号	ICMP タイプ
0	エコー応答
3	到達不能
4	発信抑制
5	リダイレクト
6	代替アドレス
8	エコー
9	ルータアドバタイズメント
10	ルータ選択要求
11	時間超過
12	パラメータ異常
13	タイムスタンプ要求
14	タイムスタンプ応答
15	情報要求
16	情報応答
17	マスク要求
18	マスク応答
30	トレースルート
31	変換エラー
32	移動体リダイレクト

- **code** -（任意）数値演算子と ICMP コードを指定します。
- *operator* - 後に続く ICMP コードに ACE が適用する演算子。次のいずれかの演算子を入力します。
  - **lt** - より小さい

  - **gt** - より大きい
  - **eq** - 等しい
  - **neq** - 等しくない
  - **range** - ICMP コード値を含む領域。この演算子を使用するときは、2 つのコード番号を指定して領域を定義します。
- *code1, code2* - ICMP タイプに対応する ICMP コード番号。表 1-5 を参照してください。**range** 演算子を入力する場合は、2 番めの ICMP コード値を入力して領域上限を定義してください。

（注） セキュリティ上の理由から、ACE では、アプライアンス の片側にある VLAN 上のインターフェイスへのアプライアンスを通して、ACE の反対側にあるもう一つの VLAN 上のインターフェイスを ping できません。たとえば、ホストでは、自身と同じ VLAN を使用して IP サブネット上にある ACE アドレスで ping を実行できますが、ACE の別の VLAN に設定された IP アドレスで ping は実行できません。

たとえば、TCP 拡張 ACL を設定するには次のように入力します。

```
host1/Admin(config)# access-list INBOUND line 10 extended permit tcp
192.168.12.0 255.255.255.0 gt 1024 172.27.16.0 255.255.255.0 lt 4000
```

たとえば、拡張 ACL からエントリを 1 つ削除するには次のように入力します。

```
host1/Admin(config)# no access-list INBOUND line 10
```

ping を制御するには **echo** (**8**) (host to ACE) と指定します。

たとえば、IP アドレス 192.168.12.5 の外部ホストに対して、IP アドレス 10.0.0.5 の ACE の背後にあるホストへの ping を許可するには、次のように入力します。

```
host1/Admin(config)# access-list INBOUND extended permit icmp host
192.168.12.5 host 10.0.0.5 echo code eq 0
```

たとえば、ICMP ACL からエントリを 1 つ削除するには次のように入力します。

```
host1/Admin(config)# no access-list INBOUND extended permit icmp host
192.168.12.5 echo
```

拡張 ACL へのコメント設定

拡張 ACL はその役割を明示するために説明のコメントを加えることができます。ACL にコメントを加えるにはコンフィギュレーション モードで **access-list** *name* **remark** コマンドを使用します。コメントは各 ACL に 1 つだけ入力できます。入力したコメントは常に ACL の先頭に表示されます。このコマンドの構文は次のとおりです。

access-list *name* **remark** *text*

キーワードと引数は次のとおりです。

- *name* - ACL を識別する重複のない名称。「"」記号で囲まずに、64 文字までの英数字でテキスト文字列を入力します。
- **remark** *text* - ACL に関して残しておきたい任意のコメントを指定します。コメントは ACL の先頭に表示されます。引用符で囲まずに、100 文字までの英数字のテキスト文字列を入力します。テキストの先頭に複数のスペースを入れることができます。末尾のスペースは無視されます。

たとえば、次のように入力します。

```
host1/Admin(config)# access-list INBOUND remark This is a remark
```

たとえば、拡張 ACL からエントリ コメントを削除するには次のように入力します。

```
host1/Admin(config)# no access-list INBOUND line 200 remark
```

no access-list *name* コマンドを使って ACL を削除すると、注釈もすべて削除されます。

EtherType ACL の設定

EtherType に基づいてトラフィックを制御する ACL を設定できます。EtherType はサブプロトコルの識別子です。EtherType ACL は Ethernet V2 フレームをサポートします。802.3 形式のフレームはタイプ フィールドでなくレングス フィールドを使用するため、EtherType ACL は 802.3 形式のフレームをサポートしません。唯一の例外はブリッジ プロトコル データ ユニット（BPDU）で、SNAP によりカプセル化されています。そのため ACE は特別に BPDU を扱うことができます。

BPDU は許可または拒否できます。デフォルトでは、すべての BPDU が拒否されます。ACE のポートがトランク ポートであるため、ACE はトランク ポート（Cisco 独自）BPDU を受信します。トランク BPDU はペイロード内部に VLAN 情報を持っているため、BPDU を許可すると ACE は発信先 VLAN によりペイロードを変更します。冗長性を設定した場合は、ブリッジ ループを避けるために、EtherType ACL を使って BPDU をインターフェイスの両側で許可する必要があります。冗長性の設定の詳細については『*Cisco 4700 Series Application Control Engine Appliance Administration Guide*』を参照してください。

MPLS を許可する場合は、ACE が Label Distribution Protocol（LDP; ラベル配布プロトコル）と Tag Distribution Protocol（TDP; タグ配布プロトコル）の TCP コネクションを必ず確立できるように、ACE に接続されている両方の MPLS ルータが LDP セッションまたは TDP セッションに利用するルータ ID として、ACE インターフェイスの IP アドレスを使用するように設定してください。LDP および TDP を利用すると、パケット転送に使用するラベル（アドレス）を MPLS ルータのネゴシエーションに使用できるようになります。

（注） レイヤ 2 インターフェイスはアウトバウンド方向のみに EtherType ACL を設定できます。

Cisco IOS ルータでは、使用するプロトコル（LDP または TDP）に応じたコマンドを入力してください。次の例では、*interface* は ACE に接続されているインターフェイスです。

```
host1/Admin(config)# mpls ldp router-id interface force
```

または

```
host1/Admin(config)# tag-switching tdp router-id interface force
```

ヒント 設定作業中に見やすくなるように、ACL の名称を大文字で入力します。ACL にはインターフェイスによる名称（INBOUND など）か、目的別の名称（MPLS など）をつけることを推奨します。

EtherType ACL を設定するには、コンフィギュレーション モードで **access-list ethertype** コマンドを使用します。このコマンドの構文は次のとおりです。

access-list *name* **ethertype** {**deny** | **permit**} {**any** | **bpdu** | **ipv6** | **mpls**}

キーワードと引数は次のとおりです。

- *name* - ACL を識別する重複のない名称。「"」記号で囲まずに、最大 64 文字までの英数字でスペースを入れずにテキスト文字列を入力します。
- ethertype - 指定するサブプロトコルの名称。有効な値は次のとおりです。
  - **deny** - 割り当てたインターフェイスのコネクションを遮断します。
  - **permit** - 割り当てたインターフェイスのコネクションを許可します。
  - **any** - すべての EtherType を指定します。
  - **bpdu** - ブリッジ プロトコル データ ユニット（BPDU）を指定します。

（注）　ACE は最小スパニング ツリー（MST）BPDU を転送しません。

  - **ipv6** - インターネット プロトコル バージョン 6（IPv6）を指定します。
  - **mpls** - マルチプロトコル ラベル スイッチング（MPLS）を指定します。

（注）　EtherType ACL に **mpls** キーワードを指定すると、ACE は MPLS ユニキャスト トラフィックと MPLS マルチキャスト トラフィックの両方を許可または拒否します。

たとえば、MPLS に EtherType ACL を設定するには次のように入力します。

```
host1/Admin(config)# access-list INBOUND ethertype permit mpls
```

EtherType ACL からエントリを 1 つ削除するには次のように入力します。

```
host1/Admin(config)# no access-list INBOUND ethertype permit mpls
```

全エントリの番号付けの変更

ACL 内部のエントリの順序番号は、コンフィギュレーション モードで **access-list** *name* **resequence** コマンドに特定の開始番号と間隔を用いて変更できます。ACL 内のエントリを並べ替える機能は、拡張 ACL でしかサポートされません。

このコマンドの構文は次のとおりです。

access-list *name* **resequence** [*number1*] [*number2*]

キーワード、オプション、および引数は次のとおりです。

- *name* - ACL を識別する重複のない名称。「"」記号で囲まずに、64 文字までの英数字でテキスト文字列を入力します。
- **resequence** - ACL 内部のエントリの番号付けの変更を指定します。
- *number1* -（任意）ACL の最初のエントリに割り当てる番号です。任意の整数を入力します。デフォルトは 10 です。
- *number2* -（任意）ACL の最初のエントリから後の各エントリに追加する番号です。任意の整数を入力します。デフォルトは 10 です。

たとえば、次のように入力します。

```
host1/Admin(config)# access-list INBOUND resequence 5 15
```

オブジェクト グループによるアクセス制御リストの簡易化

ここでは、オブジェクト グループを使用して、ACL の作成と保守を簡易化する方法について説明します。内容は次のとおりです。

- オブジェクト グループの概要
- ネットワーク オブジェクト グループの設定
- サービス オブジェクト グループの設定
- ACL でのオブジェクト グループの使用
- インターフェイスへの ACL の適用
- コンテキスト内のすべてのインターフェイスへの ACL のグローバルな適用
- ACL によるトラフィックのフィルタリング

オブジェクト グループの概要

オブジェクト グループを使用すると、1 つの ACL に複数の ACL エントリを容易に設定できます。類似のオブジェクトをグループに分類することで、オブジェクトごとに個別に ACL エントリを入力しなくても、1 つの ACL エントリで特定のオブジェクト グループを使用できます。次の種類のオブジェクト グループを作成できます。

- ネットワーク オブジェクト グループ
- サービス オブジェクト グループ

たとえば、次の 3 つのオブジェクト グループについて考えてみます。

- MyServices - 内部ネットワークへのアクセスが許可されたサービス リクエストの TCP ポート番号と UDP ポート番号が含まれます。
- TrustedHosts - 最大範囲のサービスとサーバに対するアクセスが許可されたホストとネットワークのアドレスが含まれます。
- PublicServers - 最大限のアクセスが許可されたサーバのホスト アドレスが含まれます。

各グループを作成したあとは、単一の ACL エントリを使用して、信頼できるホストから、特定のサービス リクエストを公共サーバのグループに送ることができます。

（注）　1 つの ACE で、最大 4 K のオブジェクト グループを設定することができます。各オブジェクト グループには最大 64 K のエレメントを収めることができます。1 つの ACE に、最大 64,000 の ACL エントリが使用できます。

システム全体で 64,000 エントリという ACL の制限は、拡張 ACL エントリに適用されます。拡張 ACL エントリは、1 のオブジェクト グループ エレメントに対応して個別に入力するエントリのことです。ACL でオブジェクト グループを使用すると、実際に入力する ACL エントリの数は少なくなります。ACE が、オブジェクト グループを参照する ACL を拡張すると、内部的にはそのオブジェクト グループ内のエレメント数に従って、複数の ACL エントリが存在することになります。ACL 内の拡張 ACL エントリの数を確認するには、**show access-list** *name* コマンドを入力します。詳細については、「ACL の設定情報および統計情報の表示」を参照してください。

ネットワーク オブジェクト グループの設定

ここでは、1 つの ACL に複数の ACL エントリを容易に作成するためのオブジェクト グループの設定方法について説明します。内容は次のとおりです。

- ネットワーク オブジェクト グループの作成
- ネットワーク オブジェクト グループへの説明の追加
- ネットワーク オブジェクト グループのネットワーク IP アドレスの設定
- ホスト IP アドレスの設定

ネットワーク オブジェクト グループの作成

オブジェクト グループを作成するには、設定モードで **object-group** コマンドを使用します。このコマンドの構文は次のとおりです。

object-group network *name*

キーワードと引数は次のとおりです。

- **network** - 一連のホスト、またはサブネットの IP アドレスを指定します。
- *name* - オブジェクト グループの一意の識別子。「"」記号で囲まずに、最大 64 文字までの英数字でスペースを入れずにテキスト文字列を入力します。

たとえば、ネットワーク オブジェクト グループを作成するには、次のように入力します。

```
host1/Admin(config)# object-group network NET_OBJ_GROUP1
host1/Admin(config-objgrp-netw)#
```

設定からネットワーク オブジェクト グループを削除するには、次のように入力します。

```
host1/Admin(config)# no object-group network NET_OBJ_GROUP1
```

（注） サイズの大きい ACL ですでに使用されている既存のオブジェクト グループに対し、新規のエレメントを追加する場合、ACL のサイズやオブジェクト グループ内のエレメント数によっては、ACL を再コミットすると処理に長い時間を要することがあります。極端な場合、こうした ACL を再コミットすると、ACE によるコマンドへの応答が遅くなったり、一時的に応答しなくなったりすることがあります。まず、オブジェクト グループを参照する ACL エントリを削除し、該当するオブジェクト グループに修正を加えてから、ACL に再度、ACL エントリを追加することを推奨します。

ネットワーク オブジェクト グループへの説明の追加

ネットワーク オブジェクト グループに説明を任意で追加するには、オブジェクト グループのネットワーク設定モードで **description** コマンドを使用します。このコマンドの構文は次のとおりです。

description *text*

text 引数には、引用符を含まない 240 文字までの英数字のテキスト文字列を入力します。

たとえば、ネットワーク オブジェクト グループに説明を追加するには、次のように入力します。

```
host1/Admin(config-objgrp-netw)# description intranet network object
group
```

ネットワーク オブジェクト グループから説明を削除するには、次のように入力します。

```
host1/Admin(config-objgrp-netw)# no description intranet network
object group
```

ネットワーク オブジェクト グループのネットワーク IP アドレスの設定

ネットワークの IP アドレスとネットワーク オブジェクト グループを関連付けるには、オブジェクト グループのネットワーク設定モードで *ip_address* 引数を使用します。このコマンドの構文は次のとおりです。

ip_address netmask

引数は次のとおりです。

- *ip_address* - ネットワーク オブジェクト グループに割り当てられた IP アドレス。ドット付き 10 進表記で IP アドレスを入力します（たとえば、192.168.12.15）。
- *netmask* - IP アドレスに適用するネットワーク マスク。ドット付き 10 進表記でネットワーク マスクを入力します（たとえば、255.255.255.0）。

たとえば、ネットワーク オブジェクト グループに IP アドレス 192.168.12.15 とネットワーク マスク 255.255.255.0 を追加するには、次のように入力します。

```
host1/Admin(config-objgrp-netw)# 192.168.12.15 255.255.255.0
```

必要に応じて、その他のオブジェクト グループ IP アドレスを入力します。

ネットワーク オブジェクト グループから IP アドレスを削除するには、次のように入力します。

```
host1/Admin(config-objgrp-netw)# no 192.168.12.15 255.255.255.0
```

ホスト IP アドレスの設定

ホストの IP アドレスとネットワーク オブジェクト グループを関連付けるには、オブジェクト グループのネットワーク設定モードで **host** コマンドを使用します。このコマンドの構文は次のとおりです。

host *ip_address*

ip_address は、ホストの IP アドレスを指定します。この引数を使用して、1 つの IP アドレスを指定します。ドット付き 10 進表記で IP アドレスを入力します（たとえば、192.168.12.15）。

たとえば、3 つのホスト アドレスを含むネットワーク オブジェクト グループを作成するには、次のように入力します。

```
host1/Admin(config)# object-group network NET_OBJ_GROUP1
host1/Admin(config-objgrp-netw)# description Administrator Addresses
host1/Admin(config-objgrp-netw)# host 192.168.12.15
host1/Admin(config-objgrp-netw)# host 192.168.12.21
host1/Admin(config-objgrp-netw)# host 192.168.12.27
```

サービス オブジェクト グループの設定

ここでは、1 つの ACL にプロトコル名とポート名を含む ACL エントリを容易に作成するためのオブジェクト グループの設定方法について説明します。内容は次のとおりです。

- サービス オブジェクト グループの作成
- サービス オブジェクト グループへの説明の追加
- サービス オブジェクト グループのプロトコル パラメータの定義

サービス オブジェクト グループの作成

サービス オブジェクト グループを作成するには、設定モードで **object-group** コマンドを使用します。このコマンドの構文は次のとおりです。

object-group service *name*

キーワードと引数は次のとおりです。

- **service** - 一連の IP プロトコルとポートを指定します。

- *name* - オブジェクト グループの一意の識別子。「"」記号で囲まずに、最大 64 文字までの英数字でスペースを入れずにテキスト文字列を入力します。

たとえば、サービス オブジェクト グループを作成するには、次のように入力します。

```
host1/Admin(config)# object-group service SERV_OBJ_GROUP1
host1/Admin(config-objgrp-serv)#
```

設定からサービス オブジェクト グループを削除するには、次のように入力します。

```
host1/Admin(config)# no object-group service SERV_OBJ_GROUP1
```

（注） サイズの大きい ACL ですでに使用されている既存のオブジェクト グループに対し、新規のエレメントを追加する場合、ACL のサイズやオブジェクト グループ内のエレメント数によっては、ACL を再コミットすると処理に長い時間を要することがあります。極端な場合、こうした ACL を再コミットすると、ACE によるコマンドへの応答が遅くなったり、一時的に応答しなくなったりすることがあります。まず、オブジェクト グループを参照する ACL エントリを削除し、該当するオブジェクト グループに修正を加えてから、ACL に再度、ACL エントリを追加することを推奨します。

サービス オブジェクト グループへの説明の追加

サービス オブジェクト グループに説明を任意で追加するには、オブジェクト グループのサービス設定モードで **description** コマンドを使用します。このコマンドの構文は次のとおりです。

description *text*

text 引数には、引用符を含まない 240 文字までの英数字のテキスト文字列を入力します。

たとえば、サービス オブジェクト グループに説明を追加するには、次のように入力します。

```
host1/Admin(config)# object-group service SERV_OBJ_GROUP1
host1/Admin(config-objgrp-serv)# description intranet network object
group
```

サービス オブジェクト グループから説明を削除するには、次のように入力します。

```
host1/Admin(config)# object-group service SERV_OBJ_GROUP1
```

```
host1/Admin(config-objgrp-serv)# no description intranet network
object group
```

サービス オブジェクト グループのプロトコル パラメータの定義

サービス オブジェクト グループのプロトコル パラメータを定義するには、オブジェクト グループのサービス設定モードで *protocol* 引数を使用します。TCP または UDP の場合のコマンド構文は次のとおりです。

protocol [**source** {{*operator*} *port1* | *port1 port2*}] [{{*operator*} *port3* | *port3 port4*}]

ICMP の場合のコマンド構文は次のとおりです。

icmp [*icmp-type*] [**code** {{*operator*} *icmp-code1* | **range** *icmp-code1 icmp-code2*}]

キーワード、引数、およびオプションは次のとおりです。

- *protocol* - IP プロトコルの名前または番号。プロトコル名、または IP プロトコル番号として 1 ～ 255 の整数を入力します。表 1-2 を参照してください。
- **source** -（任意）TCP、TCP-UDP、または UDP の送信元ポートを指定します。

（注） TCP または UDP の宛先ポートを指定するには、キーワードを前に入力せずに *operator* 引数を使用します。宛先のキーワードは暗黙で指定されます。

- *operator* - TCP および UDP の各プロトコルで送信元ポートと宛先ポートの番号を比較、また ICMP プロトコルで ICMP コード番号を比較するために使用するオペランドです。演算子は次のとおりです。
  - **lt** - より小さい
  - **gt** - より大きい
  - **eq** - 等しい
  - **neq** - 等しくない
  - **range** - ポート番号を含む範囲、または ICMP メッセージ コード。この演算子では 2 番めのポート番号、または 2 番めの ICMP メッセージ コードを指定して、範囲の上限を定義します。

- *port1 port2* - サービスへのアクセスを許可または拒否する IP プロトコルの送信元ポート名、またはポート番号。ポート名、または 0 ～ 65535 の整数を入力します。ポート番号を含む範囲を入力するには、**range** キーワードに続けて 2 つのポート番号を指定します。*port2* は *port1* より大きいか、等しくなければなりません。well-known TCP キーワードと番号については表 1-3 を、well-known UDP キーワードと番号については表 1-4 を参照してください。
- *port3 port4* - サービスへのアクセスを許可または拒否する IP プロトコルの宛先ポート名、またはポート番号。ポート番号を含む領域（任意）を入力するには、**range** キーワードのあとに続けて 2 つのポート番号を指定します。*port4* は *port3* より大きいか、等しくなければなりません。well-known TCP キーワードと番号については表 1-3 を、well-known UDP キーワードと番号については表 1-4 を参照してください。
- *icmp-type* - （任意）プロトコルとして ICMP を入力した場合は、ICMP メッセージングのタイプを指定します。ICMP コード番号、または表 1-5 に記載されたいずれかの ICMP タイプに対応する整数を入力します。
- **code** - （任意）数値演算子と ICMP コードを指定します。
- *icmp-code1 icmp-code2* - ICMP タイプに対応する ICMP コード番号を指定します。表 1-5 を参照してください。ICMP コードを含む範囲（任意）を入力するには、**range** キーワードに続けて 2 つの ICMP コード番号を指定します。*icmp-code1* は *icmp-code2* より大きいか、等しくなければなりません。ICMP コード、および対応する ICMP タイプのリストについては、表 1-5 を参照してください。

たとえば、サービス オブジェクト グループに宛先（宛先のキーワードは暗黙で指定されます）だけを追加する場合は、次のように入力します。

```
host1/Admin(config-objgrp-serv)# tcp eq 41
```

必要に応じて、その他のオブジェクト グループ プロトコルを入力します。

サービス オブジェクト グループから宛先 TCP ポートを削除するには、次のように入力します。

```
host1/Admin(config-objgrp-prot)# no tcp
```

たとえば、TCP（ソース ポートのみ）、UDP（送信元ポートと宛先ポート）、および ICMP に対するサービス オブジェクト グループを作成するには、次のように入力します。

```
host1/Admin(config)# object-group service TCP_UDP_ICMP
host1/Admin(config-objgrp-serv)# tcp source eq domain
host1/Admin(config-objgrp-serv)# udp source eq radius eq radius-acct
```

```
host1/Admin(config-objgrp-serv)# icmp echo code eq 0
```

上記のサービス オブジェクト グループから ICMP プロトコルを削除するには、次のように入力します。

```
host1/Admin(config-objgrp-prot)# no icmp echo code eq 0
```

ACL でのオブジェクト グループの使用

ACL 内のオブジェクト グループを使用するには、通常のネットワーク（*source_address mask* など）、サービス（*protocol operator port*）、または ICMP タイプ（*icmp_type*）引数を **object-group** *name* のキーワードと引数に置き換えます。

たとえば、**access-list extended** コマンド内で指定できる全パラメータに対してオブジェクト グループを使用する場合は、次のコマンドを入力します。

```
host1/Admin(config)# access-list acl_name extended {deny | permit}
object-group service_grp_name object-group network_grp_name
object-group network_grp_name
```

すべてのパラメータにオブジェクト グループを使用する必要はありません。たとえば、送信元アドレスにオブジェクト グループを指定できますが、宛先アドレスは IP アドレスとサブネット マスクで識別します。

ここでは、オブジェクト グループを使用するまたは使用しないで拡張 ACL を設定する例、およびオブジェクト グループ エントリを複数の ACL エントリに展開する例を示します。

- オブジェクト グループを使用せずに拡張 ACL を設定する例
- オブジェクト グループを使用して同様の拡張 ACL を設定する例
- オブジェクト グループを複数の ACL エントリに展開する例

オブジェクト グループを使用せずに拡張 ACL を設定する例

次の例では、オブジェクト グループを使用せずに拡張 ACL を設定して、内部ネットワーク上の複数のホストから Web サーバへのアクセスを制限する方法を示します。これ以外のトラフィックはすべて許可されます。

```
host1/Admin(config)# access-list ACL_IN remark "object-group acl to
deny specific hosts"
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.4 host 209.165.201.29 eq www
```

```
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.78 host 209.165.201.29 eq www
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.89 host 209.165.201.29 eq www
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.4 host 209.165.201.16 eq www
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.78 host 209.165.201.16 eq www
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.89 host 209.165.201.16 eq www
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.4 host 209.165.201.78 eq www
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.78 host 209.165.201.78 eq www
host1/Admin(config)# access-list ACL_IN extended deny tcp host
10.1.1.89 host 209.165.201.78 eq www
host1/Admin(config)# access-list ACL_IN extended permit ip any any
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input ACL_IN
```

オブジェクト グループを使用して同様の拡張 ACL を設定する例

次の例では、2 つのネットワーク オブジェクト グループ（1 つはホスト内のグループ、もう 1 つは Web サーバ用のグループ）を使用して、「オブジェクト グループを使用せずに拡張 ACL を設定する例」の場合と同じ拡張 ACL を設定する方法を示します。オブジェクト グループを使用することで、設定が容易になります。また、設定を簡単にして、より多くのホストを追加できます。

```
host1/Admin(config)# object-group network DENIED
host1/Admin(config-objgrp-network)# host 10.1.1.4
host1/Admin(config-objgrp-network)# host 10.1.1.78
host1/Admin(config-objgrp-network)# host 10.1.1.89

host1/Admin(config)# object-group network WEB
host1/Admin(config-objgrp-network)# host 209.165.201.29
host1/Admin(config-objgrp-network)# host 209.165.201.16
host1/Admin(config-objgrp-network)# host 209.165.201.78

host1/Admin(config)# access-list ACL_IN remark "object-group acl to
deny specific hosts"
host1/Admin(config)# access-list ACL_IN extended deny tcp object-group
DENIED object-group WEB eq www
host1/Admin(config)# access-list ACL_IN extended permit ip any any
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input ACL_IN
```

オブジェクト グループを複数の ACL エントリに展開する例

以下の **show** コマンドによる出力例は、オブジェクト グループ（ACE を参照）を持つ 1 つの ACL エントリが「オブジェクト グループを使用して同様の拡張 ACL を設定する例」によって複数の ACL エントリに展開される例を示したものです。**show running-config access-list** コマンドの出力には、ACL_IN ACL で展開していないオブジェクト グループの設定を示します。**show access-list ACL_IN** コマンドの出力には、展開された ACL エントリを表示します。

```
host1/Admin# show running-config access-list
Generating configuration....

access-list ACL_IN remark “object group acl to deny specific hosts”
access-list ACL_IN line 8 extended deny tcp object-group DENIED
object-group WEB eq www
access-list ACL_IN line 16 extended permit ip any any


host1/Admin# show access-list ACL_IN
access-list:ACL_IN, elements: 10, status: ACTIVE
  remark : “object group acl to deny specific hosts”
access-list ACL_IN line 8 extended deny tcp object-group DENIED
object-group WEB eq www
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.4 host
209.165.201.29 eq www (hitcount=0)
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.4 host
209.165.201.16 eq www (hitcount=0)
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.4 host
209.165.201.78 eq www (hitcount=0)
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.78 host
209.165.201.29 eq www (hitcount=0)
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.78 host
209.165.201.16 eq www (hitcount=0)
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.78 host
209.165.201.78 eq www (hitcount=0)
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.89 host
209.165.201.29 eq www (hitcount=0)
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.89 host
209.165.201.16 eq www (hitcount=0)
  access-list ACL_IN line 8 extended deny tcp host 10.1.1.89 host
209.165.201.78 eq www (hitcount=0)
access-list ACL_IN line 16 extended permit ip any any (hitcount=0)
```

インターフェイスへの ACL の適用

設定した ACL を使用する前に、ACL を 1 つまたは複数のインターフェイスに適用する必要があります。

ACL をインターフェイスのインバウンドまたはアウトバウンド方向に適用し、ACL をアクティブにするには、インターフェイス コンフィギュレーション モードで **access-group** コマンドを使用します。インターフェイスの両方向に対して、それぞれの種類（拡張および EtherType）の ACL を 1 つ適用できます。ACL 適用の方向についての詳細は「インバウンド ACL とアウトバウンド ACL」を参照してください。

（注） コンテキスト内のすべてのインターフェイスに対してグローバルに ACL を適用済みの場合は、そのコンテキスト内の個々のインターフェイスに別の ACL は適用できません。ACL のグローバルな設定の詳細については「コンテキスト内のすべてのインターフェイスへの ACL のグローバルな適用」を参照してください。

コネクションレスのプロトコルについては、トラフィックを両方向に通過させる場合、ACL を送信元インターフェイスおよび宛先側インターフェイスに適用する必要があります。たとえば、ACL の中で BGP を透過モードで許可できますが、ACL を両方のインターフェイスに適用する必要があります。

このコマンドの構文は次のとおりです。

access-group {**input** | **output**} *acl_name*

キーワードと引数は次のとおりです。

- **input** | **output** - ACL を適用するインターフェイスの方向（インバウンドまたはアウトバウンド）を指定します。
- *acl_name* - インターフェイスに適用する既存 ACL の識別名です。スペースと引用符を使用しない、64 文字までの英数字でテキスト文字列を入力します。

たとえば、次のように入力します。

```
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input INBOUND
```

インターフェイスから ACL を 1 つ削除するには次のように入力します。

```
host1/Admin(config-if)# no access-group input INBOUND
```

コンテキスト内のすべてのインターフェイスへの ACL のグローバルな適用

次の条件において、1 つのコンテキストのすべてのインターフェイスに対して 1 つの ACL をまとめて適用できます。

- ACL を適用されているインターフェイスがそのコンテキスト内に存在しない。
- レイヤ 2 ACL 1 つとレイヤ 3 ACL 1 つをインバウンド方向にのみグローバルに適用できます。
- レイヤ 2 ブリッジ グループ 仮想インターフェイス（BVI）に対しては、レイヤ 3 ACL と レイヤ 2 ACL の両方を適用できます。
- レイヤ 3 仮想 LAN（VLAN）インターフェイスに対しては、レイヤ 3 ACL のみが適用できます。
- 冗長設定のもとでは、ACE は FT VLAN にグローバル ACL を適用しません。冗長性の詳細については『*Cisco 4700 Series Application Control Engine Appliance Administration Guide*』を参照してください。

インバウンド方向のコンテキストのすべてのインターフェイスに ACL をグローバルに適用するには、コンフィギュレーション モードで **access-group input** コマンドを使用します。このコマンドの構文は次のとおりです。

access-group input *acl_name*

引数 *acl_name* に対しては、「"」記号で囲まず 64 文字までのスペースの入っていない英数字のテキスト文字列で、既存の ACL の識別名を入力します。

このコマンドを次の例のように使用すると、あるコンテキスト内のすべてのインターフェイス上のすべてのトラフィックを許可できます。

```
host1/Admin(config)# access-list ALL_ACCESS permit ip any any
```

この後、次のように入力して ACL をグローバルに適用します。

```
host1/Admin(config)# access-group input ALL_ACCESS
```

このコンテキストのすべてのインターフェイスから ACL を削除するには、次のように入力します。

```
host1/Admin(config)# no access-group input ALL_ACCESS
```

ACL によるトラフィックのフィルタリング

ACE は ACL を利用すると、その ACL に定義したアクションに基づいて、インタレスティング トラフィックをフィルタリングし、そのトラフィックを許可または拒否させることができます。ACL を利用してトラフィックをフィルタリングするには、レイヤ 3 およびレイヤ 4 クラス マップで **match access-list** コマンドを使用します。

あるパケットが ACL の中の 1 つのエントリにマッチし、それが **許可** エントリの場合、ACE はマッチしたものを受け入れます。エントリが **deny** の場合、ACE によって照合結果がブロックされます。レイヤ 3 およびレイヤ 4 クラス マップ、およびポリシー マップの設定についての詳細は第 4 章「セキュリティ アクセス コントロール リストの設定」を参照してください。

ACL の設定例

ここでは、ACE として利用できる次の種類の ACL の例を示します。

- 拡張 ACL の例
- EtherType ACL の例

拡張 ACL の例

ここでは、拡張 ACL の例を示します。送信元 IP アドレスと宛先 IP アドレスの両方（IP）、ポート（TCP または UDP）、ICMP タイプを指定する場合は拡張 ACL を使用します。拡張 ACL の設定の詳細については「拡張 ACL の設定」を参照してください。

次に示す ACL は ACE を通るすべてのホスト（ACL を適用するインターフェイス上の）を許可します。

```
host1/Admin(config)# access-list ACL_IN extended permit ip any any
```

次の ACL は、192.168.1.0/24 のホストが 209.165.201.0/27 のネットワークにアクセスすることを拒否します。それ以外のアドレスはすべて許可されます。

```
host1/Admin(config)# access-list ACL_IN extended deny tcp 192.168.1.0
255.255.255.0 209.165.201.0 255.255.255.224
host1/Admin(config)# access-list ACL_IN extended permit ip any any
```

一部のホストのみに対してアクセスを制限する場合は、制限付き許可のエントリを入力します。デフォルトでは、明示的に許可されないかぎり、他のすべてのトラフィックが拒否されます。

```
host1/Admin(config)# access-list ACL_IN extended permit ip 192.168.1.0
255.255.255.0 209.165.201.0 255.255.255.224
```

使用できるキーワードと well-known ポートの割り当ては、表 1-3 で一覧できます。DNS、Discard、Echo、Ident、NTP、RPC、SUNRPC、Talk は、それぞれ TCP および UDP に対して 1 つずつの定義を必要とします。TACACS+ には TCP のポート 49 に対する定義が 1 つ必要です。

次の ACL 例では、すべてのホスト（この ACL を適用するインターフェイス上の）に対してアドレス 209.165.201.29 の Web サイトへのアクセスを制限しています。これ以外のトラフィックはすべて許可されます。

```
host1/Admin(config)# access-list ACL_IN extended deny tcp any host
209.165.201.29 eq www
host1/Admin(config)# access-list ACL_IN extended permit ip any any
```

次の ACL では、内部ホストすべてに対して外部ネットワークとの通信を許可しますが、内部ネットワークへのアクセスは特定の外部ホストだけが許可されます。

```
host1/Admin(config)# access-list OUT extended permit ip any any
host1/Admin(config)# access-list IN extended permit ip host
209.168.200.3 any
host1/Admin(config)# access-list IN extended permit ip host
209.168.200.4 any
```

次の例で ICMP ACL を設定する方法を示します。ICMP ACL の設定の詳細については「拡張 ACL の設定」を参照してください。

```
host1/Admin(config)# access-list INBOUND extended permit icmp any any
echo
host1/Admin(config)# access-list INBOUND extended permit icmp host
10.0.0.1 host 20.0.0.1 unreachable code range 0 3
```

このセクションの内容は、次のとおりです。

- インバウンド ACL とアウトバウンド ACL
- NAT を利用する場合に ACL で用いる IP アドレス

インバウンド ACL とアウトバウンド ACL

ACE のインターフェイスを通過するトラフィックは 2 種類の方法で制御できます。

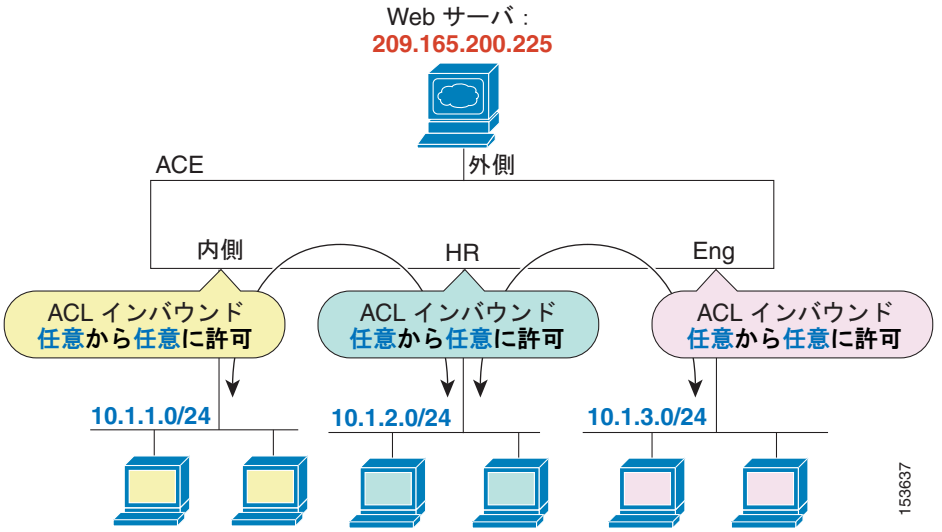
- インバウンド ACL を送信元のインターフェイスに適用すると、ACE が受信するトラフィックを制御できます。
- 一方、アウトバウンド ACL を宛先のインターフェイスに適用すると、ACE が送信するトラフィックを制御できます。

すべてのトラフィックに対して ACE に入ることを許可するには、許可用のインバウンド ACL をインターフェイスに適用する必要があります。これを行わないと、ACE はそのインターフェイスに入るすべてのトラフィックを自動的に拒否してしまいます。デフォルトでは、インバウンド ACL に設定済みの ACL に加えてアウトバウンド ACL を使ってトラフィックを制限しないかぎり、ACE のどのインターフェイスからもトラフィックが出て行くことが可能です。

（注） 「インバウンド」、「アウトバウンド」とは、あるインターフェイスに対する ACL の適用方法であり、あるインターフェイスを通って ACE に入るトラフィック、ACE から出て行くトラフィックを指します。

ACL の設定を単純化したい場合にアウトバウンド ACL を利用できます。たとえば、3 つのインターフェイス上の 3 つの内部ネットワークに相互のアクセスを許可する場合は、それぞれのインターフェイスに対して、内部側インターフェイスを通るすべてのトラフィックを許可するインバウンド ACL を作成すると簡単です（図 1-1 を参照）。

図 1-1　インバウンド ACL



それぞれの内部側インターフェイスを通るすべてのトラフィックを許可する 3 種類のインバウンド ACL は、次のコマンドで作成します。

```
host1/Admin(config)# access-list INSIDE extended permit ip any any
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input INSIDE

host1/Admin(config)# access-list HR extended permit ip any any
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input HR

host1/Admin(config)# access-list ENG extended permit ip any any
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input ENG
```

外部ネットワークにある Web サーバへのアクセスを、内部ネットワーク側の特定ホストのみに許可する場合は、指定したホストのみに許可する制限の強い ACL を作成し、外部側インターフェイスのアウトバウンド方向に適用することができます（図 1-2 を参照）。NAT と IP アドレスについての詳細は「NAT を利用する場合に ACL で用いる IP アドレス」を参照してください。他のすべてのホストは、アウトバウンド ACL により外部ネットワークから遮断されます。

次のコマンドを使用すると、特定のホストのみを許可する ACL を作成し、外部側インターフェイスのアウトバウンド方向に適用できます。

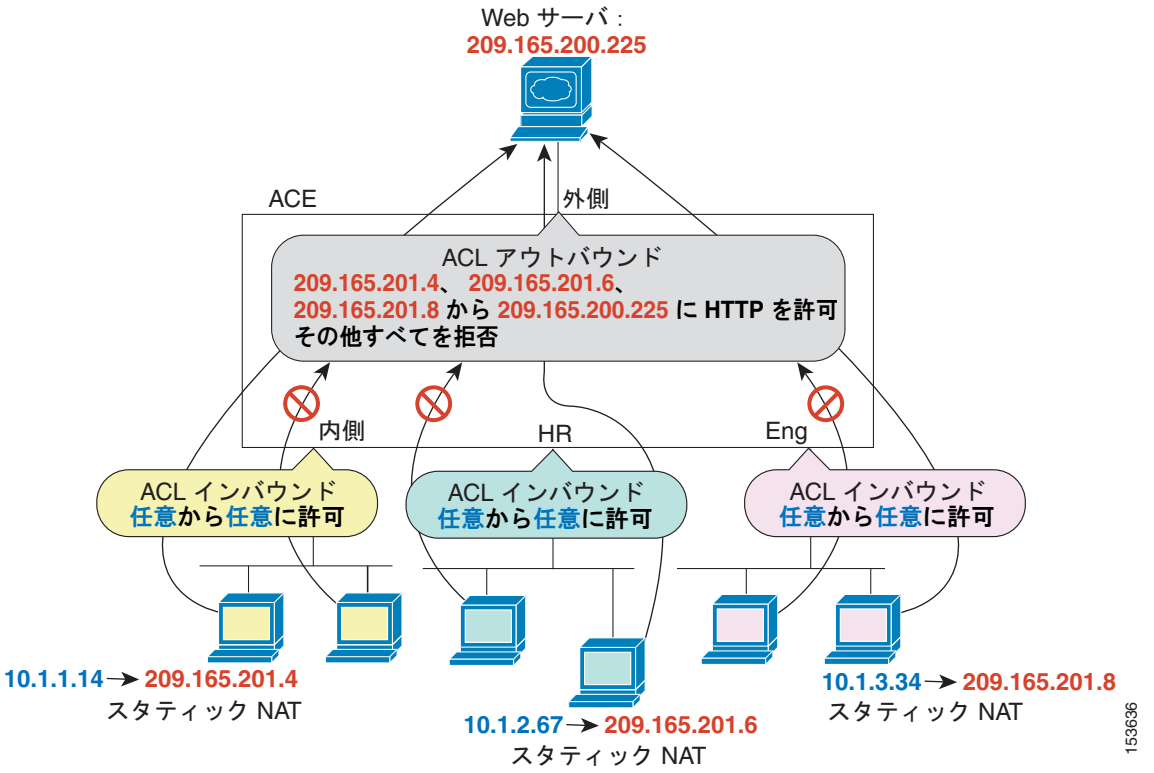
```
host1/Admin(config)# access-list OUTSIDE extended permit tcp host
209.165.201.4 host 209.165.200.225 eq www

host1/Admin(config)# access-list OUTSIDE extended permit tcp host
209.165.201.6 host 209.165.200.225 eq www

host1/Admin(config)# access-list OUTSIDE extended permit tcp host
209.165.201.8 host 209.165.200.225 eq www

host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group output OUTSIDE
```

図 1-2　アウトバウンド ACL

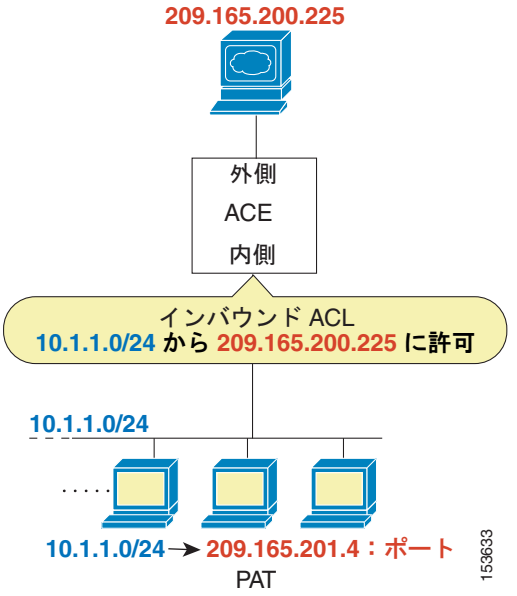


NAT を利用する場合に ACL で用いる IP アドレス

NAT を利用する場合、ACL で指定する IP アドレスは ACL を適用するインターフェイスにより決まります。そのインターフェイスに接続されているネットワークにとって有効なアドレスを使用しなければなりません。この規則はインバウンドとアウトバウンドの両方にあてはまります。使用すべきアドレスは ACL の方向では決まらず、ACL が適用されるインターフェイスのみで決まります。

たとえば、ある ACL をインターフェイスのインバウンド方向に適用すると仮定します。また、内部送信元アドレスが外部アドレスにアクセスする際に、内部送信元アドレスに対して NAT を実行するように ACE が設定されているとします。この場合、ACL を内部インターフェイスに適用するため、送信元アドレスとして変換前の元のアドレスを使用します。外部のアドレスは変換しないため、ACL で用いる宛先アドレスもそのままです（図 1-3 を参照）。

図 1-3　　ACL 内の IP アドレス：送信元アドレスに使用される NAT



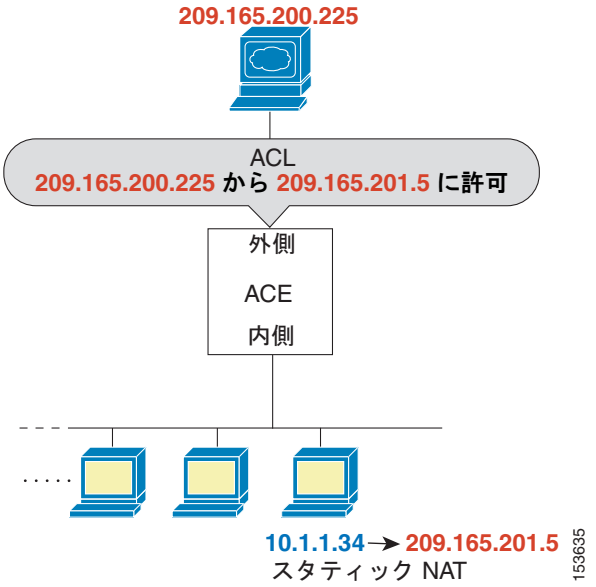
次のコマンドを使用すると、内部ネットワークの 10.1.1.0/24 に対して外部の宛先ホスト 209.165.200.225 へのアクセスを許可する ACL を作成し、VLAN インターフェイス 100 に適用できます。

```
host1/Admin(config)# access-list INSIDE extended permit ip 10.1.1.0
255.255.255.0 host 209.165.200.225
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input INSIDE
```

外部のホストに内部のホストへのアクセスを許可する場合は、外部インターフェイスにインバウンド ACL を適用できます。この ACL では、外部ネットワークから利用できるアドレスとして、内部ホストの変換後のアドレスを指定しなけれ

ばなりません（図 1-4 を参照）。

図 1-4　ACL 内の IP アドレス：宛先アドレスに使用される NAT

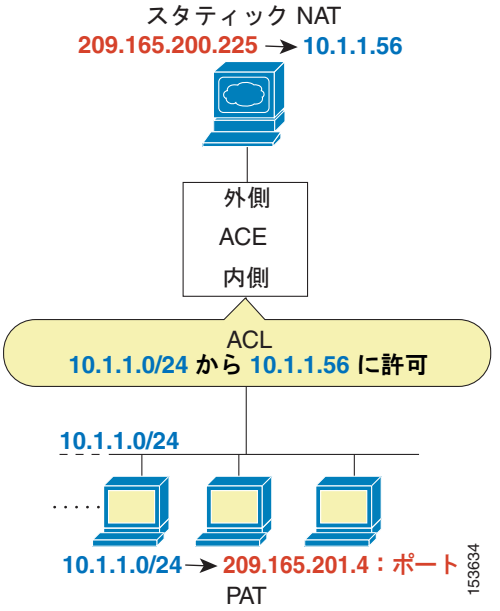


次のコマンドを利用すると、外部ホスト 209.165.200.225 に対して、内部ホスト 209.165.201.5（ホスト 10.1.1.34 の変換後アドレス）へのアクセスを許可できます。最後のコマンドでこの ACL を VLAN インターフェイス 100 に適用しています。

```
host1/Admin(config)# access-list OUTSIDE extended permit ip host
209.165.200.225 host 209.165.201.5
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input OUTSIDE
```

両側のインターフェイスで NAT を実行する場合は、ACL を作成して適用する際に、アドレスが各インターフェイス上で可視的であることを確認する必要があります。図 1-5 では、外部サーバが変換アドレスにより内部ネットワークに対して可視的であるように、スタティック NAT を使用しています。

図 1-5　ACL 内の IP アドレス：送信元アドレスと宛先アドレスに使用される NAT



次のコマンドを使用すると、内部の送信元ネットワークの 10.1.1.0/24 に対して外部の宛先ホスト 10.1.1.56（ホスト 209.165.200.225 を変換したアドレス）へのアクセスを許可する ACL を作成できます。最後のコマンドでこの ACL を VLAN インターフェイス 100 に適用しています。

```
host1/Admin(config)# access-list INSIDE extended permit ip 10.1.1.0
255.255.255.0 host 10.1.1.56
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input INSIDE
```

アウトバウンド ACL に用いる IP アドレスの例については、図 1-2 を参照してください。

EtherType ACL の例

ここでは、EtherType ACL の例を示します。EtherType ACL の設定の詳細については「EtherType ACL の設定」を参照してください。

次に、ACL が一般的な EtherType が内部インターフェイスを送信元にすることを許可する例を示します。

```
host1/Admin(config)# access-list ETHER ethertype permit ipv6
host1/Admin(config)# access-list ETHER ethertype permit bpdu
host1/Admin(config)# access-list ETHER ethertype permit mpls
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group output ethertype ETHER
```

次に、ACL が特定の EtherType に ACE を通過することは許可しますが、IPv6 は拒否する例を示します。

```
host1/Admin(config)# access-list ETHER ethertype deny ipv6
host1/Admin(config)# access-list ETHER ethertype permit bpdu
host1/Admin(config)# access-list ETHER ethertype permit mpls
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input ethertype ETHER
```

次に、ACL が EtherType BPDU のトラフィックを拒否し、それ以外のトラフィックは両方のインターフェイスで許可する例を示します。

```
host1/Admin(config)# access-list nonIP ethertype deny bpdu
host1/Admin(config)# access-list nonIP ethertype permit any
host1/Admin(config)# interface vlan 100
host1/Admin(config-if)# access-group input ethertype nonIP
```

ACL の設定情報および統計情報の表示

ここでは、ACL の設定と統計情報の表示に利用できる **show** コマンドについて説明します。内容は次のとおりです。

- ACL 設定情報の表示
- ACL 統計情報の表示

ACL 設定情報の表示

show running-config コマンドを使用すると、ACL を適用したインターフェイスなど、ACL 設定情報をすべて表示できます。このコマンドの構文は次のとおりです。

show running-config

ACL とそのエントリのみを表示するには、EXEC モードで **show running-config access-list** コマンドを使用します。このコマンドの構文は次のとおりです。

show running-config access-list

ACL 統計情報の表示

show access-list コマンドを使用すると、特定の ACL の統計情報を表示できます。このコマンドの構文は次のとおりです。

show access-list *name* [**detail**]

引数とオプションのキーワードの内容は次のとおりです。

- *name* - 既存の ACL の識別子。「"」記号で囲まずに、最大 64 文字までの英数字でスペースを入れずにテキスト文字列を入力します。
- **detail** -（任意）ACE が、deny syslog（106023）の原因となった ACL エントリを識別する際に使用する 4 バイト MD5 ハッシュ値を含む詳細な ACL 情報を表示します。表 1-6 の 0x*nnnnnnnn* 出力フィールドの説明を参照してください。

表 1-6 に **show access-list detail** コマンドが出力する各フィールドを示します。

表 1-6　show access-list detail コマンドが出力するフィールドの内容

フィールド	説明
Access-list	このセキュリティ ACL の名称
Elements	ACL 内のエントリ数
Status	ACL の現在のステータス：ACTIVE の場合はこの ACL が 1 つ以上のインターフェイスに関連付けられています。NOT-ACTIVE の場合はどのインターフェイスにも関連付けられていません。
Remark	ACL 設定時に作成された説明用のコメント
Entries	ACL 中の全エントリの全文
Hitcounts	ACL の各エントリのヒット カウント
hash 1 (0x*nnnnnnnn*)	ACL を設定すると即座に ACE が **access-list** コマンドにより計算する 32 ビットの 16 進表記による MD5 ハッシュ値。ACE では、deny syslog メッセージにこのハッシュ値が取り込まれるため、コマンド出力より syslog の原因となった ACL エントリを特定することができます。このハッシュ値は、行番号には依存しません。deny syslog メッセージ中のハッシュ値と、リブート後に実行するこのコマンドの出力の間に矛盾が生じないように、ACL 中の個々のエントリを設定するときに、必ずタブ補完を使用するか、またはキーワード全体を入力するようにします。
hash 2 (0x*nnnn*)	ACL で設定するオブジェクト グループから生成される拡張アクセス リスト エントリに基づいて ACE が計算する 16 ビットの 16 進表記（0xnnnn）による MD5 ハッシュ値。インターフェイス上で ACL を起動すると、ACE は、*hash 2* 値を計算します。オブジェクト グループのない ACL の場合、*hash 2* は常に 0x0 になります。ACE では、deny syslog メッセージにこの *hash 2* 値が取り込まれるため、この値をもとに syslog の原因となった 拡張 ACL エントリを特定することができます。このハッシュ値も、行番号には依存しません。syslog の原因となった拡張 ACL エントリを個別に特定するには、このコマンド出力から、*hash 1* と *hash 2* のいずれの 16 進値にも一致するエントリを探す必要があります。

ACL 統計情報のクリア

EXEC モードで **clear access-list** コマンドを使用すると、ACL の統計情報（各 ACL エントリのヒット カウント）をクリアできます。このコマンドの構文は次のとおりです。

clear access-list *name*

引数 *name* には既存の ACL を入力します。「"」記号で囲まずに、最大 64 文字までの英数字でスペースを入れずにテキスト文字列を入力します。

たとえば、次のように入力します。

```
host1/Admin# clear access-list acl1
```

（注） 冗長性を設定した場合は、アクティブ側とスタンバイ側 の両方の ACE に関する ACL 統計情報（ヒット カウント）を明示的にクリアする必要があります。アクティブ アプライアンスでのみ統計情報をクリアすると、スタンバイ アプライアンスの統計情報が古い値のまま残ります。