



Configuration Guide

IPv4 ACLs in AOS

This configuration guide provides an overview of Internet Protocol version 4 (IPv4) access control lists (ACLs) and their operation in ADTRAN Operating System (AOS) products. Included in this guide is an overview of ACLs in AOS, a description of how IPv4 ACLs function and how to configure them using both the Web-based graphical user interface (GUI) and the command line interface (CLI), and common applications of IPv4 ACLs. Also included in this guide are tips for troubleshooting IPv4 ACLs, such as using the **show** and **debug** features of AOS.

This guide includes the following sections:

- *[ACL Overview on page 2](#)*
- *[IPv4 ACL Functionality in AOS on page 2](#)*
- *[Hardware and Software Requirements and Limitations on page 3](#)*
- *[Beginning ACL Configuration on page 4](#)*
- *[IPv4 ACL Configuration Using the CLI on page 11](#)*
- *[IPv4 ACL Configuration Examples on page 15](#)*
- *[IPv4 ACL Configuration Command Summary on page 18](#)*
- *[Troubleshooting on page 18](#)*
- *[Additional Resources on page 23](#)*

ACL Overview

ACLs compare IP traffic to a list of specified criteria and determine if that traffic is forwarded or discarded. ACLs are useful in many AOS filtering and security features, including access groups, policy classes, access classes, quality of service (QoS) maps, route maps, crypto maps, and demand routing. ACLs allow these features to logically inspect each IP packet, compare it to the set of criteria listed in the ACL, and then take the appropriate action with the packet.

There are three major types of ACLs: Hardware ACLs, IP (both IPv4 and Internet Protocol version 6 (IPv6) ACLs, and media access control (MAC) ACLs. This configuration guide deals with only IPv4 ACLs.



For more information regarding hardware ACLs, refer to the configuration guide [Hardware ACLs in AOS](#); for more information regarding IPv6 ACLs, refer to the configuration guide [IPv6 in AOS](#); for more information regarding MAC ACLs, refer to the configuration guide [Creating MAC Access Control Lists](#). All documents are available online at <https://supportforums.adtran.com>.

IP ACLs can be divided into two categories: standard and extended.

Standard ACLs

Standard ACLs are the simplest form of ACLs. They inspect only the IPv4 source address in the IPv4 packet. In standard ACLs, packets sent from specific subnets or specific hosts can be either permitted or denied. Standard ACLs use an ACL cache to speed up packet matching.

Extended ACLs

Extended ACLs offer more choices in discriminating traffic than standard ACLs provide. Extended ACLs can inspect the protocol, the source and destination addresses and ports, the Transmission Control Protocol (TCP) flags, and the Internet Control Message Protocol (ICMP) message types of the IPv4 packet.

IPv4 ACL Functionality in AOS

ACLs list criteria for IP packets in lines called ACL entries. Each ACL entry begins with either the keyword **permit** or **deny**. **Permit** indicates that packets matching the specific criteria are selected and handled according to the configuration of the AOS feature using the ACL. **Deny** indicates that packets matching the specific criteria are not selected and are handled accordingly. Packets are compared to the ACL entry from top-to-bottom. Because ACLs work this way, when creating an ACL with several entries, you should put the most specific entries at the beginning and put the most general entries at the end of the ACL. If a packet does not match the criteria specified by the first entry, then it is compared to the criteria in the next entry. When the IP packet is found to match an entry's specified criteria, then the packet is either categorized as **permit** or **deny** and the comparison of the ACL entries abruptly stops. At this point, the feature using the ACL takes the appropriate action. If a packet does not match any of the entries, it is implicitly denied. However, an empty ACL with no **permit** or **deny** entries will implicitly permit everything.

It is important to remember that ACLs are merely criteria lists that other features use to classify IPv4 packets. Other features that use ACLs include the following:

- Static filtering
- Policy based routing (PBR)
- AOS firewall
- Network monitoring
- Demand interfaces
- Integrated traffic monitoring (ITM)
- Port forwarding
- Network quality monitoring (NQM)
- Wide area network (WAN) failover applications
- Transparent proxy
- Voice quality monitoring (VQM)
- Network address translation (NAT)
- QoS
- Simple Network Management Protocol (SNMP) applications

For more information about any of these features, refer to *Additional Resources on page 23*.

Hardware and Software Requirements and Limitations

Most AOS products have the ability to use ACLs. For information about whether the previously mentioned features that use ACLs are available on your AOS platform, refer to the *Product Feature Matrix*, available online at <https://supportforums.adtran.com>.

It is important also to remember that specific features using ACLs will have limitations and requirements for the use of ACLs with that feature. You will need to refer to the documentation for that specific feature for the requirements and limitations of ACLs (refer to *Additional Resources on page 23*). It is also important to remember that extended and standard ACLs function differently, and cannot always be used in the same places or interchangeably.

Each ACL has an implicit **deny any** as the last criteria if there are other explicit criteria entries within the ACL. An empty ACL has an implicit **permit any** when there are no other explicit criteria entries within the ACL.

In AOS firmware release R11.10.2, the ability to match or block non-initial fragments in ACL entries was implemented.

Beginning ACL Configuration

Before you begin configuring an ACL, it is helpful to know a few things about the traffic you want to monitor. Keep these items in mind:

- What is the source IPv4 address, host name, or port that I want to monitor?
- What type of protocol do I want to monitor?
- What is the destination IPv4 address, host name, or port that I want to monitor?
- Does the feature that I am using require a standard or an extended ACL?
- If I need multiple entries in this ACL, which are the specific entries I need to put at the beginning, and which are the more general entries that I should put at the end?

**NOTE**

It is important to know ahead of time the order in which your match criteria needs to be arranged. If you are using the CLI to configure your ACL, new entries can only be added to the end of the list. If you are using the GUI to configure your ACL, the order of the entries can be altered using the up and down arrows as described on [<source> parameter on page 9](#).

Once you know the purpose and type of your ACL, you can begin to configure it.

ACL Configuration Using the GUI

ACLs can be configured in multiple ways using the GUI. Some AOS features will automatically create necessary ACLs in that feature's section in the GUI. For example, the Firewall security zone's policy has a section for editing the ACL to choose which data is matched, and the virtual private network (VPN) or firewall wizard creates ACLs automatically. If the feature you are using helps you to configure an ACL, you should follow that route in the GUI. This section demonstrates how to create an ACL independent of the features that may use it, and also shows how to edit any created ACL, whether or not it was created using another feature ([Editing ACLs Using the GUI on page 10](#)).

To configure ACLs using the GUI, follow these steps:

1. Open a new Web page in your Internet browser.
2. Type your AOS product's IP address in the Internet browser's address field in the following form: **http://<ip address>**, for example, **http://65.162.109.200**.
3. At the prompt, enter your user name and password and select **OK**.

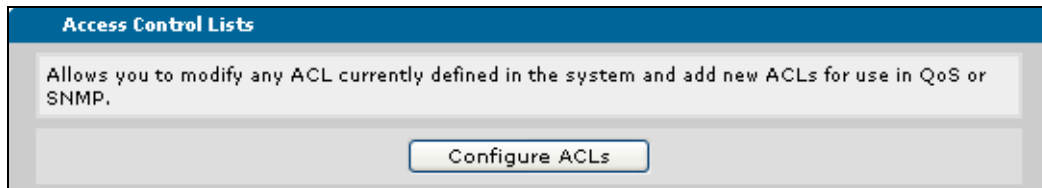
**NOTE**

*The default user name is **admin** and the default password is **password**.*

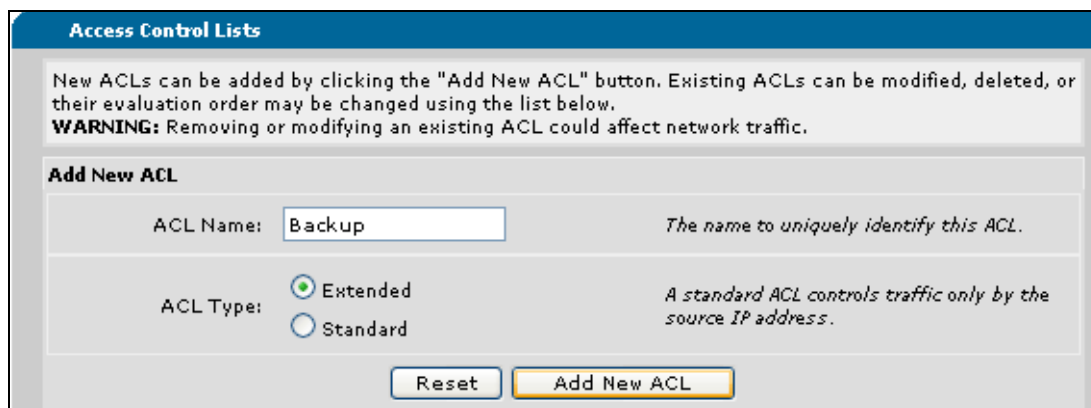
- Navigate to **Data > Firewall > Firewall/ACLs**.



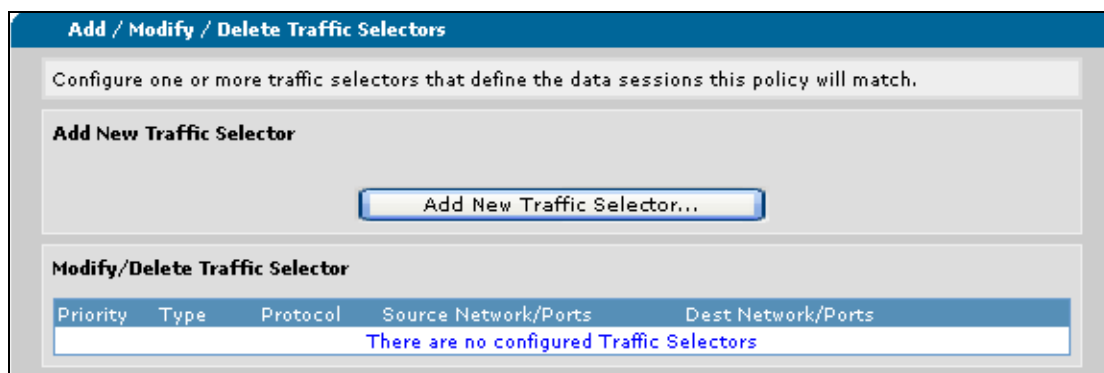
- Scroll down to the **Access Control Lists** dialog box, and select **Configure ACLs**.



- Enter the **ACL Name** and select the **ACL Type**. Remember that **Extended** ACLs can inspect the protocol, the source and destination addresses and ports, the TCP flags, and the ICMP message types of the IP packet, and that **Standard** ACLs only inspect the source IP address. Once you have selected the ACL type, select **Add New ACL**.



- Whether you have chosen an extended or standard ACL, you will be prompted to define the parameters of that ACL. Select **Add New Traffic Selector** to add the inspection criteria to the ACL.



8. If you are creating a **standard** ACL, you will be prompted to define whether the traffic will be permitted or denied, and to define the source information of the traffic.

Selecting **Permit** indicates that traffic matching the specified criteria is permitted by the application using the ACL. Selecting **Deny** indicates that traffic matching the specified criteria is ignored by the application using the ACL.

When specifying the **Source Data**, you can select **Any**, which indicates all traffic will be permitted, or you can define an IP address or host name. IP addresses require an address and a network mask.



*Using the **hostname** option to define source traffic relies on matching traffic based on a domain naming system (DNS) name. The unit must be configured with DNS servers for this function to work.*

Once you have entered the correct action and source information for the ACL, select **Apply** and your standard ACL is created.

9. If you are creating an **Extended** ACL, you will be prompted for much more information when defining the traffic selectors.

Add New Custom ACL Entry

Enter the information on this form to specify which packets will trigger the specified action.

Filter Type: ☒ Permit [?](#)
☐ Deny

Protocol: any [?](#)

ICMP Message Type (ICMP Only): ☐ Any [?](#)
☐ Well Known

Source Data

Source Host/Network: ☒ Any
☐ IP Address
☐ Hostname

Address: . . .
Mask: . . .

Source Ports (TCP/UDP Only): ☐ Any
☐ Well Known
☐ Specified

Source IP Address or a hostname of sessions originating in Security Zone 'VPN' that should be affected.

Source ports of sessions originating in Security Zone 'VPN' that should be affected.

Destination Data

Destination Host/Network: ☒ Any
☐ IP Address
☐ Hostname

Address: . . .
Mask: . . .

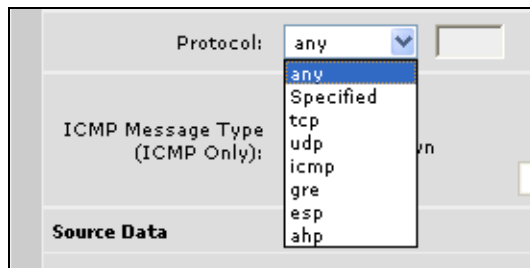
Destination IP Address of sessions originating in Security Zone 'VPN' that should be affected.

Destination Ports (TCP/UDP Only): ☐ Any
☐ Well Known
☐ Specified

Destination ports of sessions originating in Security Zone 'VPN' that should be affected.

To begin, you will define whether the ACL will **Permit** or **Deny** traffic that matches the criteria.

Next, you can define the protocol the ACL will use to match traffic. Selecting **any** indicates that any protocol type will match this selector. You can also specify a particular protocol from the **Protocol** drop-down menu.



Protocol selections include the most widely used protocols. If you want to specify a protocol not included on the list, select **Specified** from the drop-down menu and enter the protocol ID in the given space. Protocol IDs range from **0** to **255**.

If you are using the ICMP, you can also specify what type of ICMP message you want to match. You can specify that **Any** ICMP message type is considered, or you can specify a well-known ICMP message type by selecting it from the drop-down menu.

Next, you define the **Source Data** of the IP packets the ACL will match. Similar to the configuration of the standard ACL, you can define the source information by IP address, by host name, or by specifying that **Any** sources are matched. Extended ACLs also give you the option to specify a source port for packet matching. These are TCP and User Datagram Protocol (UDP) ports, and you can specify that **Any** source port will be matched, that **Well Known** ports will be matched, or you can specify a source port. This option will not be available unless you are using UDP or TCP. **Well Known** ports are selected from the drop-down menu. To specify a port, select **Specified** and then select the appropriate criteria from the drop-down menu. You can specify a range of ports, ports greater than or less than a specified value, a port equal to a specified value, or ports not equal to a specified value by using the given parameters. Then enter the appropriate port number(s) in the given fields.

After defining the source data, you need to specify the **Destination Data**. Destination data is specified in the same way as source data: by specifying the IP address or host name and by optionally specifying a destination port.

Once the criteria have been entered, select **Apply** and the traffic selectors are created.



*In general, each ACL has an implicit **deny any** as the last criteria if there are other explicit criteria entries within the ACL. Each empty ACL has an implicit **permit any** when there are no other explicit criteria entries within the ACL.*

10. The traffic selectors you have configured appear in a list for the ACL. You can select the **Type** hyperlink to edit the ACL traffic selectors.

Add / Modify / Delete Policy Traffic Selectors

Configure one or more traffic selectors that define the data sessions this policy will match.

Add New Traffic Selector

Add New Traffic Selector...

Modify/Delete Traffic Selector

Priority	Type	Protocol	Source Network/Ports	Dest Network/Ports	
	Permit	any	any	any	Delete

11. As the traffic selectors are created for any ACL, they populate a traffic selector list for the particular ACL. Because ACLs work by matching criteria in top-to-bottom order, the order in which the criteria are listed is very important. You can move the match criteria order by using the up and down arrows in the traffic selector criteria list. This list appears below the **Add New Traffic Selector** button.

Add / Modify / Delete Policy Traffic Selectors

Configure one or more traffic selectors that define the data sessions this policy will match.

Add New Traffic Selector

Add New Traffic Selector...

Modify/Delete Traffic Selector

Priority	Type	Protocol	Source Network/Ports	Dest Network/Ports	
▲▼	Permit	UDP	any: = 69	any: any	Delete
▲▼	Deny	AH	any	any	Delete

You can also delete traffic selector criteria from this list by selecting the **Delete** button for the appropriate traffic selector.

The ACLs are now created and ready to be used by other AOS security and routing features. Keep in mind that although the ACL is created, it is inactive until used by another feature. For more information on these features and how they work with ACLs, refer to [IPv4 ACL Configuration Examples on page 15](#).

Editing ACLs Using the GUI

To edit any ACL, navigate to **Data > Firewall > Firewall/ACLs** and select **Configure ACLs**. At the bottom of the **Access Control Lists** menu, a list of all configured ACLs appears. Select the name hyperlink of the ACL you would like to edit.

Access Control Lists

New ACLs can be added by clicking the "Add New ACL" button. Existing ACLs can be modified, deleted, or their evaluation order may be changed using the list below.
WARNING: Removing or modifying an existing ACL could affect network traffic.

Add New ACL

ACL Name: *The name to uniquely identify this ACL.*

ACL Type: ☒ Extended *A standard ACL controls traffic only by the source IP address.*
☐ Standard

Modify/Delete ACLs

To view or modify an existing ACL, click the "Name" link in the desired row.

<input type="checkbox"/>	ACL Name	ACL Type	Security Zone(s)
<input type="checkbox"/>	Backup	Standard	---
<input type="checkbox"/>	BackupExt	Extended	---
<input type="checkbox"/>	MatchAll	Extended	---
<input type="checkbox"/>	primary_connection1	Extended	---
<input type="checkbox"/>	sec_conn	Extended	---
<input type="checkbox"/>	secondaryconnection	Extended	---
<input type="checkbox"/>	test	Extended	---

You can remove ACLs from your configuration by checking the box next to the appropriate ACL(s) and selecting **Remove Selected ACLs**.

IPv4 ACL Configuration Using the CLI

Before configuring your IPv4 ACL, remember to check the criteria outlined in the section [Beginning ACL Configuration on page 4](#). To configure IPv4 ACLs using the CLI, follow these steps:

1. Create and name an IPv4 ACL using the **ip access-list [extended | standard] <name>** command from the Global Configuration mode prompt. This command will determine whether you are creating a standard ACL (matching on source information) or an extended ACL (matching on a number of criteria) and also enter the configuration mode for the ACL. To create and name a standard IPv4 ACL, and enter the ACL's configuration mode, enter the command as follows:

```
(config)#ip access-list standard Trusted
Configuring New Standard ACL "Trusted"
(config-std-nacl)#
```

To create and name an extended IPv4 ACL, and enter the ACL's configuration mode, enter the command as follows:

```
(config)#ip access-list extended MATCHALL
Configuring New Extended ACL "MATCHALL"
(config-ext-nacl)#
```

If the ACL name already exists, the command enters the existing ACL's configuration mode. Using the **no** form of this command removes the ACL from the unit's configuration.

2. Specify an ACL comment using the **remark <text>** command. The ACL comments make it easier for another user to identify the purpose of the ACL. Enter the command from the ACL configuration mode as follows:

```
(config-ext-nacl)#remark permits Smith workstation
(config-ext-nacl)#
```

Using the **no** form of this command removes the remark from the ACL.

3. If you are creating a **Standard** ACL, you will then specify the packet source information and decide whether the ACL will **permit** or **deny** traffic based on a match. Use the following command to specify this information:

```
[permit | deny] <source> [log] [track <name>]
```

Permit indicates that traffic matching the criteria is allowed to be used by the feature using the ACL.

Deny indicates that traffic matching the criteria is not considered by the feature using the ACL.

The *<source>* parameter specifies the source used for packet matching. Sources can be expressed in one of four ways:

- Using the keyword **any** to match any IPv4 address.
- Using **host** *<ip address>* to specify a single host address. IPv4 addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**).
- Using the *<ip address>* *<wildcard mask>* format to match all IPv4 addresses in a range. The wildcard mask corresponds to a range of IPv4 addresses (network) or a specific host. Wildcard masks are also expressed in dotted decimal notation (for example, **0.0.0.255**) and they work in reverse logic from subnet masks. When broken out into binary form, a **0** indicates which bits of the IPv4v address to consider, and a **1** indicates which bits are disregarded. For example, specifying 255 in any octet of the wildcard mask equates to a “don’t care” for that octet in the IPv4 address. Additionally, a 30-bit mask would be represented with the wildcard string **0.0.0.3**, a 28-bit mask with **0.0.0.15**, a 24-bit mask with **0.0.0.255**, and so forth.
- Using the keyword **hostname** *<hostname>* to match traffic based on a DNS name. The unit must be configured with DNS servers for this function to work. Using **vrf** *<name>* in conjunction with the **hostname** parameter associates a nondefault VRF with the DNS host name for the source. The VRF is required if the router’s DNS server is on a nondefault VRF. This parameter can only be used with the **hostname** source. The command in this case would appear:

[permit | deny] hostname <hostname> [vrf <name>] [track <name>] [log]

The optional **track** *<track>* parameter associates the ACL entry with a particular track. This association causes the ACL entry to only be applied when a track is in the PASS state. The track is configured independently to be disabled or enabled if certain events take place. For an example of ACL use with a track, refer to [IPv4 ACL Configuration Examples on page 15](#).

The optional **log** parameter specifies that any entries that match the ACL criteria will be logged. Logging is beneficial when used in conjunction with the **debug ip access-list** command, which displays the number of times in the last five seconds that an inspected packet has matched the entry.

For example, to create a standard ACL entry that permits traffic from IPv4 address **190.72.22.248** and logs entries that match that address, the command is entered as follows:

```
(config-std-nacl)#permit host 190.72.22.248 log
```

These are the basic steps for configuring a standard ACL. You can enter as many criteria as you need to have the ACL match traffic for your network. It is important to remember that the order of your entries is crucial: the ACL will match traffic based on criteria from the top down. If the order of the entries is not correct, you will have to remove the applicable ACL entries and then reenter them in the correct order. If a new entry needs to be at the top of the entry list, all the previous entries must be removed. You should also remember that each ACL has an implicit **deny any** as the last criteria if there are other explicit criteria entries within the ACL and each empty ACL has an implicit **permit any** when there are no other explicit criteria entries within the ACL. Once you have configured the ACL, you can use it with the appropriate feature.

4. If you are configuring an **Extended ACL**, you will specify whether the ACL will **permit** or **deny** traffic based on protocol, source information, and destination information. Use the following command to specify this information:

```
[permit | deny] <protocol> <source> <source port> <destination> <destination port> [log]
[track <name>] [fragments]
```

Permit indicates that traffic matching the criteria is allowed to be used by the feature using the ACL. **Deny** indicates that traffic matching the criteria is not considered by the feature using the ACL.

The *<protocol>* parameter specifies the protocol used by the packet. You can select from **ip**, **icmp**, **tcp**, **udp**, **ahp**, **esp**, **gre** or you can enter a specific protocol. Specific protocol range is **0** to **255**.

The *<source>* parameter specifies the source used for packet matching. Sources can be expressed in one of four ways:

- Using the keyword **any** to match any IPv4 address.
- Using **host** *<ipv4 address>* to specify a single host address. IPv4 addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**).
- Using the *<ipv4 address> <wildcard mask>* format to match all IPv4 addresses in a range. The wildcard mask corresponds to a range of IPv4 addresses (network) or a specific host. Wildcard masks are also expressed in dotted decimal notation (for example, **0.0.0.255**) and they work in reverse logic from subnet masks. When broken out into binary form, a **0** indicates which bits of the IPv4 address to consider, and a **1** indicates which bits are disregarded. For example, specifying 255 in any octet of the wildcard mask equates to a “don’t care” for that octet in the IPv4 address. Additionally, a 30-bit mask would be represented with the wildcard string **0.0.0.3**, a 28-bit mask with **0.0.0.15**, a 24-bit mask with **0.0.0.255**, and so forth.
- Using the keyword **hostname** *<hostname>* to match traffic based on a DNS name. The unit must be configured with DNS servers for this function to work. Using **vrf** *<name>* in conjunction with the **hostname** parameter associates a nondefault VRF with the DNS host name for the source. The VRF is required if the router’s DNS server is on a nondefault VRF. This parameter can only be used with the **hostname** source. The command in this case would appear:

```
[permit | deny] hostname <hostname> [vrf <name>] <source port> <destination>
<destination port> [track <name>] [log] [fragments]
```

The *<source port>* parameter is optional, and allows you to specify the monitored traffic source port. The source port is used only when the *<protocol>* is specified as **tcp** or **udp**. The following selections are available for specifying source port information:

any matches any port.

eq *<port number/name>* matches only packets equal to a specified port number.

gt *<port number/name>* matches only packets with a port number greater than the specified port number.

lt *<port number/name>* matches only packets with a port number less than the specified number.

neq *<port number/name>* matches only packets that are not equal to the specified port number.

range *<starting port number/name> <ending port number/name>* matches only packets that contain a port number in the specified range.

Port numbers are those ports used by TCP or UDP to pass information to upper layers and range from **0** to **65535**. All ports below **1024** are considered well-known ports, and are controlled by the Internet Assigned Numbers Authority (IANA). All ports above **1024** are dynamically assigned ports that include registered ports for vendor-specific applications. UDP and TCP ports can also be entered by port name. [Table 1](#) lists the UDP port names and numbers. [Table 2](#) lists the TCP port names and numbers.

Table 1. UDP Port Names and Numbers

biff (Port 512)	bootpc (Port 68)	bootps (Port 67)
discard (Port 9)	dnsix (Port 195)	domain (Port 53)
echo (Port 7)	isakmp (Port 500)	dnsix (Port 195)
nameserver (Port 42)	mobile-ip (Port 434)	netbios-dgm (Port 138)
netbios-ns (Port 137)	netbios-ss (Port 139)	ntp (Port 123)
pim-auto-rp (Port 496)	rip (Port 520)	snmp (Port 161)
snmptrap (Port 162)	sunrpc (Port 111)	syslog (Port 514)
tacacs (Port 49)	talk (Port 517)	tftp (Port 69)
time (Port 37)	who (Port 513)	xdmcp (Port 177)

Table 2. TCP Port Names and Numbers

bgp (Port 179)	chargen (Port 19)	cmd (Port 514)
daytime (Port 13)	discard (Port 9)	domain (Port 53)
echo (Port 7)	exec (Port 512)	finger (Port 79)
ftp (Port 21)	gopher (Port 70)	hostname (Port 101)
ident (Port 113)	irc (Port 194)	klogin (Port 543)
kshell (Port 544)	login (Port 513)	lpd (Port 515)
nnrp (Port 119)	pim-auto-rp (Port 496)	pop2 (Port 109)
pop3 (Port 110)	smtp (Port 25)	sunrpc (Port 111)
syslog (Port 514)	tacacs (Port 49)	talk (Port 517)
tftp (Port 69)	telnet (Port 23)	time (Port 37)
uucp (Port 540)	whois (Port 43)	www (Port 80)

The *<destination>* parameter specifies the destination used for packet matching. Destinations can be expressed in the same four ways as the source information (described on [<source> parameter on page 12](#)).

The *<destination port>* parameter is optional, and allows you to specify the monitored traffic source port. The destination port is also used only when the *<protocol>* is specified as **tcp** or **udp**. The same selections available for source port selection are available for destination port selection.

The optional **track** *<track>* parameter associates the ACL entry with a particular track. This association allows the ACL entry to be applied when a track changes states (from PASS to FAIL or vice versa). The track is configured independently, but the association between the track and ACL entry allows the entry to be disabled or enabled if certain events take place. For additional information about ACL use with a track, refer to [Additional Resources on page 23](#).

The optional **log** parameter specifies that any entries that match the ACL criteria will be logged. Logging is beneficial when used in conjunction with the **debug ip access-list** command, which displays the number of times in the last five seconds that an inspected packet has matched the entry.

For example, to create an extended ACL entry that permits all Internet key exchange (IKE) (UDP Port 500) packets from the **190.72.22.0 /24** network and logs matching entries, the command is entered as follows:

```
(config-ext-nacl)#permit udp 190.72.22.0 0.0.0.255 eq 500 any eq 500 log
```

The optional **fragments** parameter is used to specify that the ACL entry is only matched by non-initial fragments. The **fragments** keyword is only available when the specified protocol is **ip**. IPv4 ACLs match non-initial fragments in the following manner:

- Non-initial fragments can match entries with the **fragments** keyword, provided the other Layer 3 information specified in the entry matches the packet.
- Non-initial fragments can match entries with the **ip** protocol specified, provided the other Layer 3 information specified in the entry matches the packet.
- Non-initial fragments are implicitly permitted by access groups if the fragments did not match an explicit entry in the ACL.

These are the basic steps for configuring an extended IPv4 ACL. You can enter as many criteria as you need to have the ACL monitor traffic for your network. It is important to remember that the order of your entries is crucial: the ACL will match traffic based on criteria from the top down. If the order of the entries is not correct, you will have to remove the applicable ACL entries and then reenter them in the correct order. If a new entry needs to be at the top of the entry list, all the previous entries must be removed. You should also remember that each ACL has an implicit **deny any** as the last criteria if there are other explicit criteria entries within the ACL and each empty ACL has an implicit **permit any** when there are no other explicit criteria entries within the ACL. Once you have configured the ACL, you can use it with the appropriate feature.

IPv4 ACL Configuration Examples

The following sections describe typical ACL applications in real-world settings. All of the following configurations are done using the CLI. The configuration parameters entered in these examples are sample configurations only. You should configure these applications in a manner consistent with the needs of your particular network. CLI prompts have been removed from the configuration examples to provide you with a method of copying and pasting configurations directly from this guide into the CLI. You should make the necessary adjustments to these configurations before adding them to your configuration to ensure they will function properly in your network.

Configuration Example 1

In the following example, two extended IPv4 ACLs (**Deny1** and **Deny2**) are configured to allow specific traffic while denying other traffic. **Deny1** is configured to allow only traffic from IPv4 address **172.16.1.2** while denying all other traffic (through the implicit **deny**), and **Deny2** is configured to deny traffic from IPv4 address **172.16.1.2** while allowing all other traffic. This example demonstrates the ability of ACLs to both deny and permit in the same ACL. You should note the order of the ACL entries, and also note how they work from general to specific. In **Deny1**, incoming traffic is matched against the IPv4 address. If the traffic matches, it is permitted. If the traffic does not match, it is compared to the second entry, and if the traffic is sourced from any IP matching **172.16.1.0 0.0.0.255**, it is denied. In **Deny2**, incoming traffic is first matched against source IP address **172.16.1.2**. If the traffic matches this source IPv4 address, it is denied. If the traffic does not match, it is compared to the second entry and is permitted. To configure these ACLs, use the following configuration:

```
!
ip access-list extended Deny1
    remark Allows 172.16.1.2 and denies all other IPs from 172.16.1.0/24
    permit ip host 172.16.1.2 any
!
ip access-list extended Deny2
    remark Denies 172.16.1.2 and allows all other traffic
    deny ip host 172.16.1.2 any
    permit ip any any
!
```

Configuration Example 2

In the following example, there are two private local area networks (LANs): a corporate headquarter LAN (**10.25.15.0**) and a remote branch office LAN (**10.10.4.0**). ACLs and an ACP are used to provide connectivity for traffic between the private LANs, to grant access to the public Internet connection for all users, both from the branch site and headquarters, and to hide private IPv4 addresses for all traffic transmitted to the public domain.

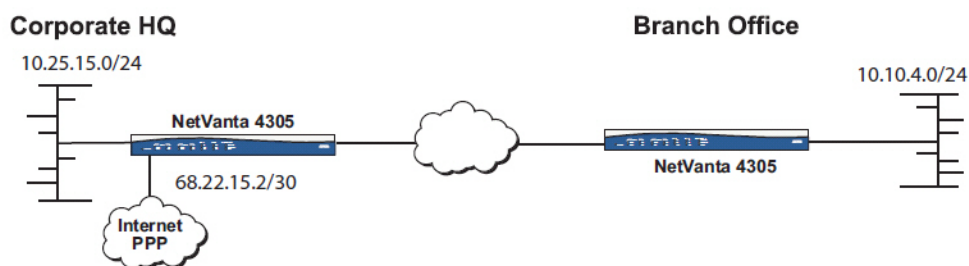


Figure 1. Corporate Headquarters and Branch Office Network Diagram

In order to configure the ACLs for this purpose, the first step is to define the ACL entries that select traffic received on the connection to the branch office. These ACL entries sort the traffic into two categories: traffic destined for the corporate LAN or traffic destined for the public Internet. Each category requires an extended ACL to select the appropriate traffic. Next, another set of ACL entries needs to be defined. These

entries also sort traffic into two categories for the corporate network: traffic destined for the branch office LAN or traffic destined for the public Internet. Again, each category requires an extended ACL to select the appropriate traffic. The first ACL created, **InterLAN**, matches traffic destined for the corporate LAN and traffic destined for the branch office LAN. The second ACL, **Internet**, matches traffic destined from either LAN for the Internet. Because the branch office and corporate headquarters contain different IP subnets on the 10.0.0.0 network, the IP addresses listed in the **permit** statements are modified using the wildcard bits **0.255.255.255** to encompass the entire **10.0.0.0** network. Once the ACLs are created, an ACP is created (**InterLANwNAT**) referencing the ACLs **InterLAN** and **Internet**. The ACP is then applied to Frame Relay interface **1.16** (private virtual circuit (PVC) to branch office) and Ethernet interface **0/1** (local network connection). The configuration for this scenario is as follows:

```
!  
ip firewall  
!  
ip access-list extended InterLAN  
    permit ip 10.0.0.0 0.255.255.255 10.0.0.0 0.255.255.255  
!  
ip access-list extended Internet  
    permit ip 10.0.0.0 0.255.255.255 any  
!  
ip policy-class InterLANwNAT  
    allow list InterLAN stateless  
    nat source list Internet interface ppp 1 overload  
!  
interface fr 1.16  
    access-policy InterLANwNAT  
!  
interface eth 0/1  
    access-policy InterLANwNAT  
!  
!
```

IPv4 ACL Configuration Command Summary

The following table summarizes the basic commands used with the creation and configuration of IPv4 ACLs. For more specific details about the parameters of these commands, refer to [IPv4 ACL Configuration Using the CLI on page 11](#).

Table 3. IPv4 ACL Configuration Command Summary

Access Prompt	Command	Command Description
(config)#	ip access-list [extended standard] <name>	Creates an extended or standard ACL and enters the ACL configuration mode.
(config-std-nacl)# (config-ext-nacl)#	remark <text>	Specifies an ACL remark for identification purposes.
(config-std-nacl)#	[permit deny] <source> [log] [track <name>]	Configures the action and source data for a standard ACL. Also, optionally enables match logging or optionally associates the ACL with a track.
(config-ext-nacl)#	[permit deny] <protocol/> <source> <source port> <destination> <destination port> [log] [track <name>] [fragments]	Configures the parameters of an extended ACL. Includes the action, the protocol, source data, source port, destination data, and destination port. Optionally enables logging (log), associates the ACL with a track (track), or specifies that the line matches only non-initial fragments (fragments).

Troubleshooting

ACL configurations can be viewed using either the GUI or the CLI. The GUI debug options and the CLI **show** and **debug** commands aid in troubleshooting as they allow a quick viewing of ACL component configurations. The following sections describe the GUI and CLI troubleshooting techniques.

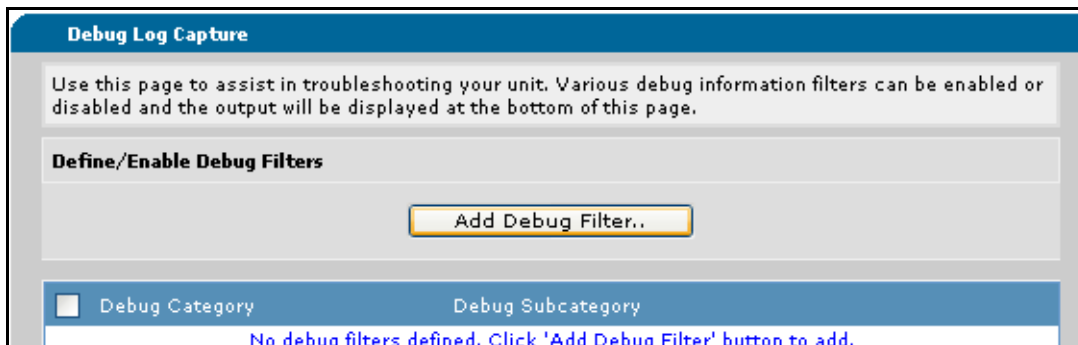
GUI Troubleshooting

To access ACL debug information, or to view ACL configuration using the GUI, connect to the GUI and follow the steps outlined in the following sections.

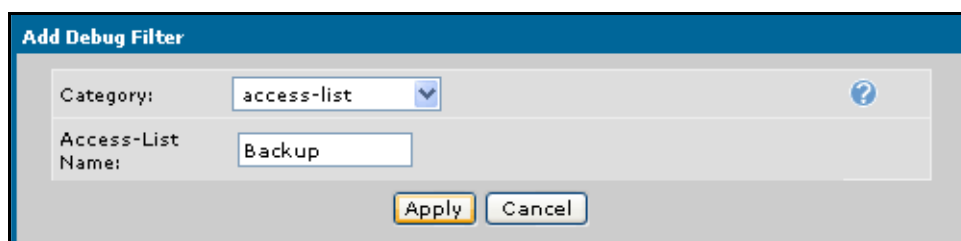
Debugging ACLs

Enabling ACL debug messaging displays information containing the logged matches recorded by the ACL (using the **log** option). To use the GUI to activate ACL debug, follow these steps:

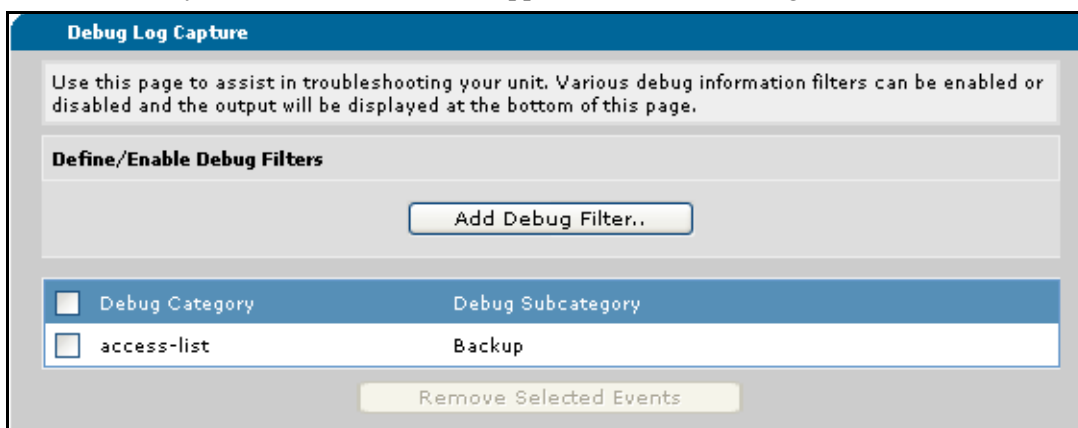
1. Navigate to **Utilities > Debug Unit**. Select **Add Debug Filter**.



2. Select **access-list** from the drop-down menu, and enter the ACL name in the appropriate field. Then select **Apply**.



3. The **access-list** keyword and the ACL name appear in the list of debug filters.



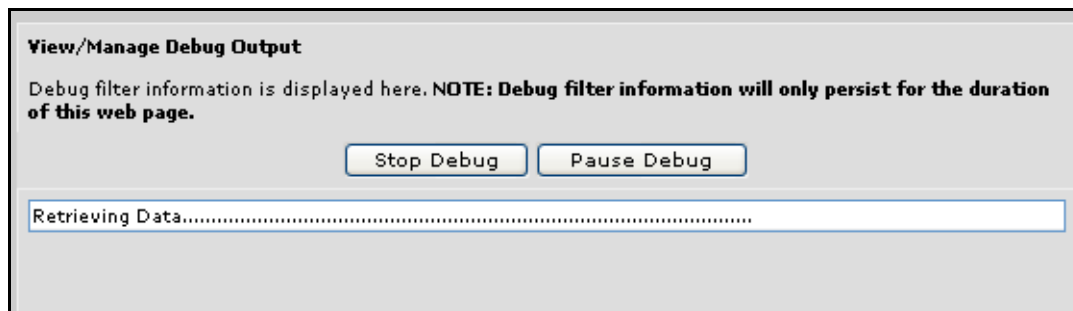
4. Select **Start Debug** and debug messaging is activated for the specified ACL.



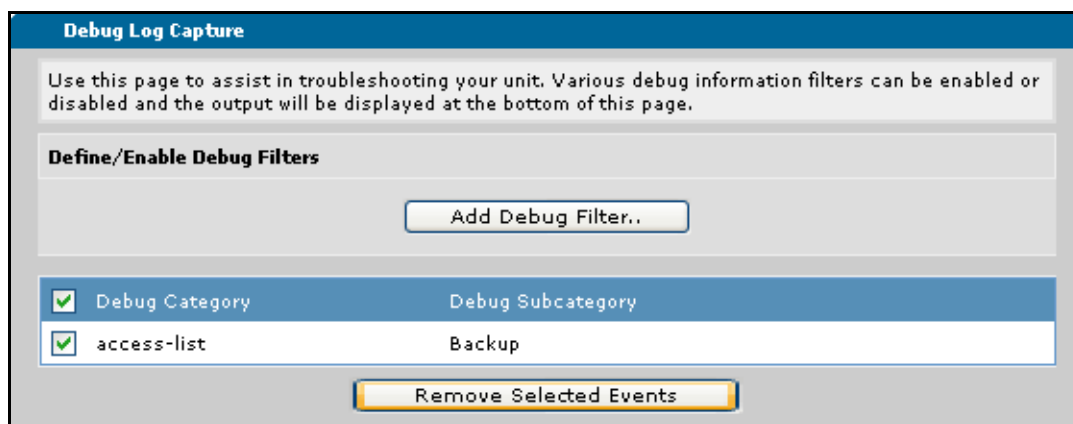
Turning on a large amount of debug information can adversely affect the performance of your unit.

Any events are displayed in the **View/Manage Debug Output** portion of the menu. From this area, you

can also **Stop Debug** or **Pause Debug** by using the appropriate buttons.



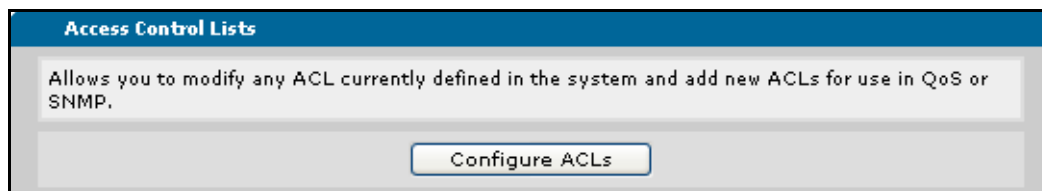
5. If you want to remove the ACL from the debug menu, select the box next to the **access-list** parameter and then select **Remove Selected Events**.



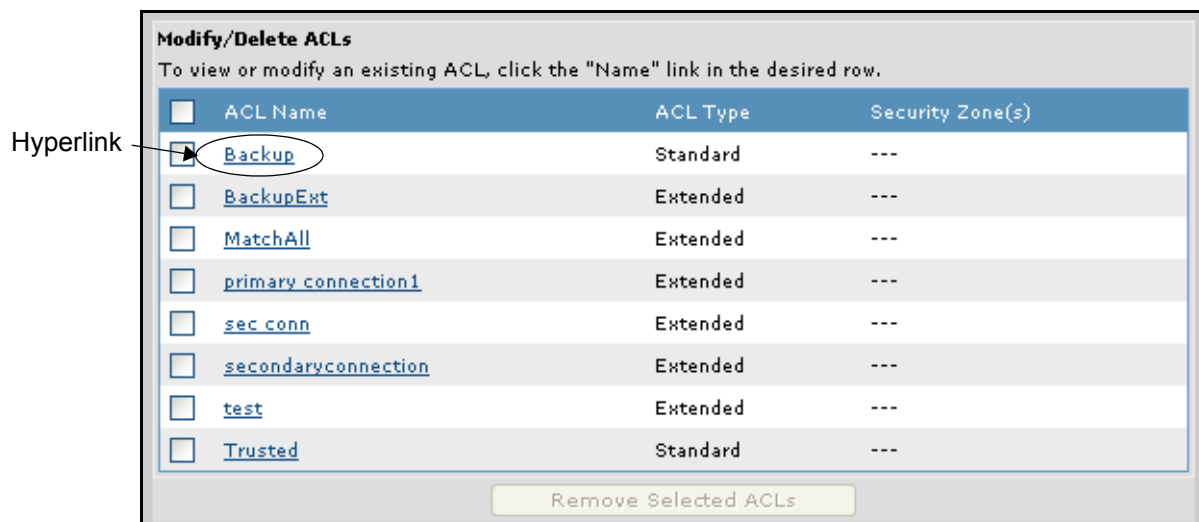
Viewing ACL Configuration

You can also view current ACL configuration using the GUI. To view configuration for an ACL, follow these steps (these are the same steps used to edit an ACL configuration as described on [source parameter on page 10](#)):

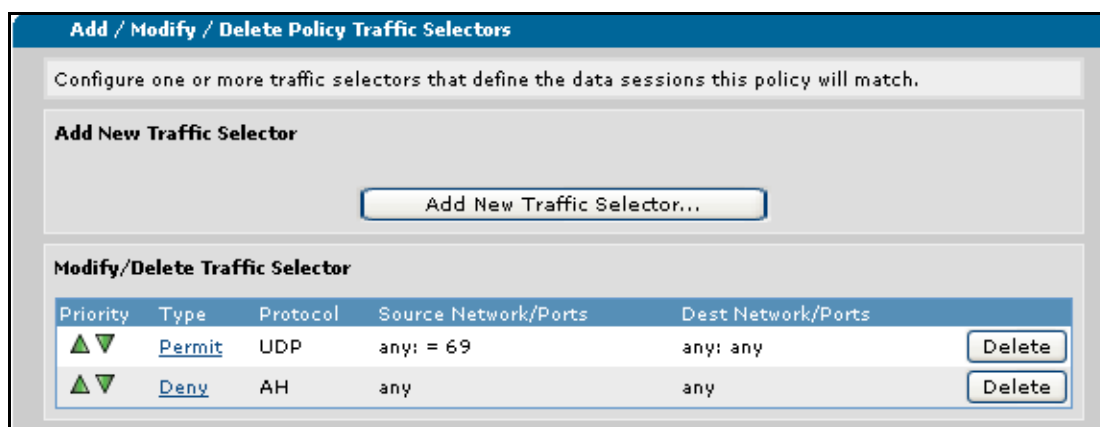
1. Navigate to **Data > Firewall > Firewall/ACLs**. Select **Configure ACLs** from the bottom of the menu.



2. Select the name of the ACL you would like to view from the list by using the ACL name hyperlink.



3. The configuration parameters for the ACL are displayed.



CLI Troubleshooting

The following table provides a quick look at the ACL CLI troubleshooting commands.

Table 4. IPv4 ACL Troubleshooting Commands

Access Prompt	Command	Description
#	show ip access-lists [<name>]	Displays ACL configuration and statistics for all ACLs or a specified ACL.
#	show running-config access-lists [verbose]	Displays the current running configuration for all configured IP ACLs.
#	clear access-list [<name>]	Clears all statistics associated with all ACLs or a specified ACL.
#	debug access-list <name>	Displays the logged matches of a specified ACL.

Show Commands

Show commands are issued from the Enable mode prompt, and display configuration information for all ACLs or for a specified ACL. Using the *<name>* parameter displays only the information about a specific ACL, rather than all configured ACLs. Using the **verbose** keyword displays the entire running configuration to the terminal screen, rather than only showing the default values.

The following is sample output from the **show ip access-lists** command:

```
>enable
#show ip access-lists
* - Indicates access list entry disabled by track.
Standard IP access list Backup
    permit any log (0 matches)
Standard IP access list Trusted
Extended IP access list BackupExt
    permit udp any eq tftp any log (0 matches)
    deny  ahp any any log (0 matches)
Extended IP access list MatchAll
Extended IP access list primary connection1
    permit icmp any hostname 10.200.1.123 (10.200.1.123) (0 matches)
Extended IP access list sec conn
    permit icmp 10.200.1.0 0.0.0.255 any echo-reply log (0 matches)
Extended IP access list secondaryconnection
Extended IP access list test
```

The following is sample output from the **show running-config access-lists** command:

```
>enable
#show running-config access-lists
ip access-list extended MatchAll
    Implicit permit (only for empty ACLs)
!
ip access-list extended Primaryconnection1
    permit icmp any host 10.200.1.123
!
ip access-list extended Secconn
    permit icmp 10.200.1.0 0.0.0.255 any echo-reply log
!
```

Clear Commands

Clear commands are issued from the Enable mode prompt, and clear all statistics associated with all ACLs or a specified ACL. Using the *<name>* parameter clears the statistics only for a specific ACL, rather than all configured ACLs. The example clears all statistics for the ACL **Backup**:

```
>enable
#clear access-list Backup
```

Debug Commands

Debug commands are issued from the Enable mode prompt, and display the matches logged by the ACL. Debug messages are printed every **5** seconds, and display the number of entry matches (**permit** or **deny**) since the last debug message was printed. Logging must be enabled on the ACL for any match information to be stored or displayed. Refer to [parameter on page 12](#) for more information about enabling logging.



Depending on the application in which the ACL is used, every packet may not be counted as an ACL match. Datapath applications like policy sessions and access groups with Rapid Route enabled make a single ACL comparison for the first packet of traffic flow only. Further packets in the flow are understood to match the ACL, so extra comparisons are not made. In these cases, the ACL hit count will not match the packet count.



Turning on a large amount of debug information can adversely affect the performance of your unit.

Enter the command as follows to enable debug messaging for the ACL **HOST**:

>enable

#debug access-list HOST

2009.06.09 14:15:03 ACCESS_LIST.HOST

permit host 192.168.0.1 log (1 matches)

2009.06.09 14:15:13 ACCESS_LIST.HOST

permit host 192.168.0.1 log (3 matches)

2009.06.09 14:15:57 ACCESS_LIST.HOST

permit host 192.168.0.1 log (1 matches)

Additional Resources

[Table 5](#) lists common features that use ACLs, and their accompanying documentation. For more information about any of these features and how they use ACLs, visit the ADTRAN support community online at <https://supportforums.adtran.com>.

Table 5. Common Features that Use ACLs

Feature	Article Title/Description
AOS IPv4 Firewall	IPv4 Firewall Configuration Guide describes how ACLs help the firewall to protect your network by filtering sessions from unrecognized origins and monitoring traffic. Also explains the relationship between ACLs and ACPs.
Demand Interface	Configuring Demand Routing in AOS describes how ACLs define interesting traffic for the demand interface.
ITM	Integrated Traffic Monitoring describes how to configure ACLs to help ITM sample and filter traffic flows.

Table 5. Common Features that Use ACLs (Continued)

Feature	Article Title/Description
NAT	Configuring Internet Access (Many to one NAT) with the Firewall Wizard in AOS describes how ACLs and ACPs work together in the CLI and how to use the firewall wizard.
	NAT Pools in AOS describes how ACLs are used with NAT configurations to limit forwarded traffic.
Network Monitoring	Configuring Network Monitor in AOS describes how network monitor can monitor and direct traffic based on an ACL attached to a schedule and a track.
NQM	Configuring Network Quality Monitoring in AOS describes how ACLs can be used to limit access to the probe responder used in NQM.
PBR	Policy Based Routing in AOS describes how PBR can manipulate traffic paths based on ACL entry matching.
Port Forwarding	Port Forwarding describes how ACLs work with port forwarding by defining the remote subnets that are allowed to connect using port forwarding.
QoS	Configuring QoS in AOS describes how ACLs can be used to specify how traffic is matched to criteria outlined in the QoS map.
SNMP	Configuring SNMP in AOS describes how ACLs function with SNMP when enabling read-only and read-write SNMP access.
SIP Proxy	Configuring SIP Proxy in AOS describes how ACLs can limit incoming voice traffic that is allowed to reach the SIP stack.
VPN	Configuring a VPN Using Main Mode in AOS describes how ACLs are used in the configuration of VPN tunnels.
	Configuring a VPN Using Aggressive Mode in AOS describes how to configure ACLs for use with crypto maps and VPN tunnels.
	Configuring Main Mode and Remote Client VPN in AOS CLI describes how ACLs are used to define the traffic sent through VPN tunnels.
	VPN Based WAN Failover describes how ACLs are used by ACPs to define traffic moving across specified connections.
VQM	Configuring Voice Quality Monitoring in AOS describes how ACLs can work to limit the calls monitored by VQM.
WAN	Configuring T1 and E1 WAN Interfaces describes the configuration of ACLs for use with ACPs and how both work together to restrict or allow certain traffic connections.
	WAN Failover Using Network Monitor describes using ACLs to match ICMP traffic sent from a probe to determine connectivity, and describes how those ACLs can be used to direct traffic to another connection if a route fails.