# Access Control Lists

Access control lists (ACLs) are used by many different features. When applied to interfaces or globally as access rules, they permit or deny traffic that flows through the appliance. For other features, the ACL selects the traffic to which the feature will apply, performing a matching service rather than a control service.

The following sections explain the basics of ACLs and how to configure and monitor them. Access rules, ACLs applied globally or to interfaces, are explained in more detail in Access Rules.



# About ACLs

Access control lists (ACLs) identify traffic flows by one or more characteristics, including source and destination IP address, IP protocol, ports, EtherType, and other parameters, depending on the type of ACL. ACLs are used in a variety of features. ACLs are made up of one or more access control entries (ACEs).

## ACL Types

The ASA uses the following types of ACLs:

- Extended ACLs—Extended ACLs are the main type that you will use. These ACLs are used for access rules to permit and deny traffic through the device, and for traffic matching by many features, including service policies, AAA rules, WCCP, Botnet Traffic Filter, and VPN group and DAP policies. In ASDM, many of these features have their own rules pages and they cannot use extended ACLs that you define in the ACL Manager, although ACL Manager will display the ACLs created on those pages. See Configure Extended ACLs, on page 7.
- EtherType ACLs—EtherType ACLs apply to non-IP layer-2 traffic on bridge group member interfaces only. You can use these rules to permit or drop traffic based on the EtherType value in the layer-2 packet. With EtherType ACLs, you can control the flow of non-IP traffic across the device. See Configure EtherType Rules.

- Webtype ACLs—Webtype ACLs are used for filtering clientless SSL VPN traffic. These ACLs can deny access based on URLs or destination addresses. See Configure Webtype ACLs, on page 12.
- Standard ACLs—Standard ACLs identify traffic by destination address only. There are few features that use them: route maps and VPN filters. Because VPN filters also allow extended access lists, limit standard ACL use to route maps. See Configure Standard ACLs, on page 11.

The following table lists some common uses for ACLs and the type to use.

***Table 1: ACL Types and Common Uses***

| ACL Use | ACL Type | Description |
|---|---|---|
| Control network access for IP traffic (routed and transparent mode) | Extended | The ASA does not allow any traffic from a lower security interface to a higher security interface unless it is explicitly permitted by an extended ACL. In routed mode, you must use an ACL to permit traffic between a bridge group member interface and an interface outside same the bridge group.<br>**Note** To access the ASA interface for management access, you do not also need an ACL allowing the host IP address. You only need to configure management access according to the general operations configuration guide. |
| Identify traffic for AAA rules | Extended | AAA rules use ACLs to identify traffic. |
| Augment network access control for IP traffic for a given user | Extended, downloaded from a AAA server per user | You can configure the RADIUS server to download a dynamic ACL to be applied to the user, or the server can send the name of an ACL that you already configured on the ASA. |
| VPN access and filtering | Extended<br>Standard | Group policies for remote access and site to site VPNs use standard or extended ACLs for filtering. Remote access VPNs also use extended ACLs for client firewall configurations and dynamic access policies. |
| Identify traffic in a traffic class map for Modular Policy Framework | Extended | ACLs can be used to identify traffic in a class map, which is used for features that support Modular Policy Framework. Features that support Modular Policy Framework include TCP and general connection settings, and inspection. |
| For bridge group member interfaces, control network access for non-IP traffic | EtherType | You can configure an ACL that controls traffic based on its EtherType for any interface that is a member of a bridge group. |
| Identify route filtering and redistribution | Standard<br>Extended | Various routing protocols use standard ACLs for route filtering and redistribution (through route maps) for IPv4 addresses, and extended ACLs for IPv6. |
| Filtering for clientless SSL VPN | Webtype | You can configure a webtype ACL to filter URLs and destinations. |

# The ACL Manager

The ACL Manager appears in two forms:

- In the main window, for example, by selecting **Configuration** > **Firewall** > **Advanced** > **ACL Manager**. In this case, the ACL Manager shows extended ACLs only. These ACLs include those generated by rules you create in the Access Rules, Service Policy Rules, and AAA Rules pages. Be careful that edits you make in ACL Manager do not negatively impact these rules; changes you make here will be reflected on those other pages.
- From a policy that requires an ACL, by clicking the **Manage** button next to the field. In this case, the ACL Manager can have separate tabs for standard and extended ACLs, if the policy allows either type. Otherwise, the view is filtered to show standard, extended, or webtype ACLs only. The ACL Manager never shows EtherType ACLs.

There are separate pages for standard ACLs and webtype ACLs, so that you can configure them in the main window. These pages are functionally equivalent to the ACL Manager without the name:

- Standard ACLs—**Configuration** > **Firewall** > **Advanced** > **Standard ACL**.
- Webtype ACLs—**Configuration** > **Remote Access VPN** > **Clientless SSL VPN Access** > **Advanced** > **Web ACLs**.

# ACL Names

Each ACL has a name or numeric ID, such as outside_in, OUTSIDE_IN, or 101. Limit the names to 241 characters or fewer.Consider using all uppercase letters to make it easier to find the name when viewing a running configuration.

Develop a naming convention that will help you identify the intended purpose of the ACL. For example, ASDM uses the convention *interface-name_purpose_direction*, such as “outside_access_in”, for an ACL applied to the “outside” interface in the inbound direction.

Traditionally, ACL IDs were numbers. Standard ACLs were in the range 1-99 or 1300-1999. Extended ACLs were in the range 100-199 or 2000-2699. The ASA does not enforce these ranges, but if you want to use numbers, you might want to stick to these conventions to maintain consistency with routers running IOS Software.

# Access Control Entry Order

An ACL is made up of one or more ACEs. Unless you explicitly insert an ACE at a given line, each ACE that you enter for a given ACL name is appended to the end of the ACL.

The order of ACEs is important. When the ASA decides whether to forward or drop a packet, the ASA tests the packet against each ACE in the order in which the entries are listed. After a match is found, no more ACEs are checked.

Thus, if you place a more specific rule after a more general rule, the more specific rule might never be hit. For example, if you want to permit network 10.1.1.0/24, but drop traffic from host 10.1.1.15 on that subnet, the ACE that denies 10.1.1.15 must come before the one that permits 10.1.1.0/24. If the permit 10.1.1.0/24 ACE comes first, 10.1.1.15 will be allowed, and the deny ACE will never be matched.

Use the Up and Down buttons to reposition rules as necessary.

# Permit/Deny vs. Match/Do Not Match

Access control entries either “permit” or “deny” traffic that matches the rule. When you apply an ACL to a feature that determines whether traffic is allowed through the ASA or is dropped, such as global and interface access rules, “permit” and “deny” mean what they say.

For other features, such as service policy rules, “permit” and “deny” actually mean “match” or “do not match.” In these cases, the ACL is selecting the traffic that should receive the services of that feature, such as application inspection or redirection to a service module. “Denied” traffic is simply traffic that does not match the ACL, and thus will not receive the service. (In ASDM, service policy rules actually use Match/Do Not Match, and AAA rules use Authenticate/Do Not Authenticate, for example, but in the CLI, it is always permit/deny.)

# Access Control Implicit Deny

ACLs that are used for through-the-box access rules have an implicit deny statement at the end. Thus, for traffic controlling ACLs such as those applied to interfaces, if you do not explicitly permit a type of traffic, that traffic is dropped. For example, if you want to allow all users to access a network through the ASA except for one or more particular addresses, then you need to deny those particular addresses and then permit all others.

For management (control plane) ACLs, which control to-the-box traffic, there is no implicit deny at the end of a set of management rules for an interface. Instead, any connection that does not match a management access rule is then evaluated by regular access control rules.

For ACLs used to select traffic for a service, you must explicitly “permit” the traffic; any traffic not “permitted” will not be matched for the service; “denied” traffic bypasses the service.

For EtherType ACLs, the implicit deny at the end of the ACL does not affect IP traffic or ARPs; for example, if you allow EtherType 8037, the implicit deny at the end of the ACL does not now block any IP traffic that you previously allowed with an extended ACL (or implicitly allowed from a high security interface to a low security interface). However, if you *explicitly* deny all traffic with an EtherType ACE, then IP and ARP traffic is denied; only physical protocol traffic, such as auto-negotiation, is still allowed.

# IP Addresses Used for Extended ACLs When You Use NAT

When you use NAT or PAT, you are translating addresses or ports, typically mapping between internal and external addresses. If you need to create an extended ACL that applies to addresses or ports that have been translated, you need to determine whether to use the real (untranslated) addresses or ports or the mapped ones. The requirement differs by feature.

Using the real address and port means that if the NAT configuration changes, you do not need to change the ACLs.

### Features That Use Real IP Addresses

The following commands and features use real IP addresses in the ACLs, even if the address as seen on an interface is the mapped address:

- Access Rules (extended ACLs referenced by the **access-group** command)
- Service Policy Rules (Modular Policy Framework **match access-list** command)
- Botnet Traffic Filter traffic classification (**dynamic-filter enable classify-list** command)

- AAA Rules (**aaa ... match** commands)
- WCCP (**wccp redirect-list group-list** command)

For example, if you configure NAT for an inside server, 10.1.1.5, so that it has a publicly routable IP address on the outside, 209.165.201.5, then the access rule to allow the outside traffic to access the inside server needs to reference the server's real IP address (10.1.1.5), and not the mapped address (209.165.201.5).

### Features That Use Mapped IP Addresses

The following features use ACLs, but these ACLs use the mapped values as seen on an interface:

- IPsec ACLs
- **capture** command ACLs
- Per-user ACLs
- Routing protocol ACLs
- All other feature ACLs.

## Time-Based ACEs

You can apply time range objects to extended and webtype ACEs so that the rules are active for specific time periods only. These types of rules let you differentiate between activity that is acceptable at certain times of the day but that is unacceptable at other times. For example, you could provide additional restrictions during working hours, and relax them after work hours or at lunch. Conversely, you could essentially shut your network down during non-work hours.

You cannot create time-based rules that have the exact same protocol, source, destination, and service criteria of a rule that does not include a time range object. The non-time-based rule always overrides the duplicate time-based rule, as they are redundant.

**Note** Users could experience a delay of approximately 80 to 100 seconds after the specified end time for the ACL to become inactive. For example, if the specified end time is 3:50, because the end time is inclusive, the command is picked up anywhere between 3:51:00 and 3:51:59. After the command is picked up, the ASA finishes any currently running task and then services the command to deactivate the ACL.

# Licensing for Access Control Lists

Access control lists do not require a special license.

However, to use **sctp** as the protocol in an entry, you must have a Carrier license.

# Guidelines for ACLs

### Firewall Mode

- Extended and standard ACLs are supported in routed and transparent firewall modes.
- Webtype ACLs are supported in routed mode only.
- EtherType ACLs are supported for bridge group member interfaces only, in routed and transparent modes.

### Failover and Clustering

Configuration sessions are not synchronized across failover or clustered units. When you commit the changes in a session, they are made in all failover and cluster units as normal.

### IPv6

- Extended and webtype ACLs allow a mix of IPv4 and IPv6 addresses.
- Standard ACLs do not allow IPv6 addresses.
- EtherType ACLs do not contain IP addresses.

### Additional Guidelines

- When you specify a network mask, the method is different from the Cisco IOS software **access-list** command. The ASA uses a network mask (for example, 255.255.255.0 for a Class C mask). The Cisco IOS mask uses wildcard bits (for example, 0.0.0.255).
- Normally, you cannot reference an object or object group that does not exist in an ACL or object group, or delete one that is currently referenced. You also cannot reference an ACL that does not exist in an **access-group** command (to apply access rules). However, you can change this default behavior so that you can "forward reference" objects or ACLs before you create them. Until you create the objects or ACLs, any rules or access groups that reference them are ignored. To enable forward referencing, select the option in the access rules advanced settings; choose **Configuration > Access Rules** and click the **Advanced** button.
- If you enter more than one item in source or destination address, or source or destination service, ASDM automatically creates an object group for them with the prefix DM_INLINE. These objects are automatically expanded to their component parts in the rule table view, but you can see the object names if you deselect the **Auto-expand network and service objects with specified prefix** rule table preference in **Tools** > **Preferences**.
- (Extended ACL only) The following features use ACLs, but cannot accept an ACL with identity firewall (specifying user or group names), FQDN (fully-qualified domain names), or Cisco TrustSec values:
  - VPN **crypto map** command
  - VPN **group-policy** command, except for **vpn-filter**
  - WCCP
  - DAP

# Configure ACLs

The following sections explain how to configure the various types of generic ACL, except those used as access rules (including EtherType), service policy rules, AAA rules, and other uses where ASDM provides a special-purpose page for those rule-based policies.

## Configure Extended ACLs

An extended ACL is represented as a named container of ACEs. To create a new ACL, you must first create the container. Then, you can add ACEs, edit existing ACEs, and reorder the ACEs using the table in ACL Manager.

The extended ACL can include a mix of IPv4 and IPv6 addresses.

### Procedure

**Step 1** Choose **Configuration** > **Firewall** > **Advanced** > **ACL Manager**.

**Step 2** If you are creating a new ACL, choose **Add** > **Add ACL**, fill in a name, and click **OK**.

The ACL container is added to the table. You can later rename it by selecting it and clicking **Edit**.

**Step 3** Do any of the following:

- To add an ACE at the end of the ACL, select the ACL name or any ACE within it and choose **Add > Add ACE**.
- To insert an ACE at a specific location, select an existing ACE and choose **Add > Insert** to add the ACE above the rule, or choose **Add > Insert After**.
- To edit a rule, select it and click **Edit**.

**Step 4** Fill in the ACE properties. The primary options to select are:

- **Action: Permit/Deny**—Whether you are permitting (selecting) the described traffic or are denying (deselecting, not matching) it.
- **Source/Destination criteria**—A definition of the source (originating address) and destination (target address of the traffic flow). You typically configure IPv4 or IPv6 addresses of hosts or subnets, which you can represent with network or network object groups, or network-service object groups. You can also specify a user or user group name for the source. Additionally, you can use the Service field to identify the specific type of traffic if you want to focus the rule more narrowly than all IP traffic. If you include service specifications in a network-service object, specify IP in the Service field. If you implement Cisco TrustSec, you can use security groups to define source and destination.

For detailed information on all of the available options, see Extended ACE Properties, on page 8.

When you are finished defining the ACE, click **OK** to add the rule to the table.

**Step 5** Click **Apply**.

# Extended ACE Properties

When you add or edit an ACE in an extended ACL, you can configure the following properties. In many fields, you can click the "..." button on the right of the edit box to select, create, or edit objects that are available for the field.

**Action: Permit/Deny**

Whether you are permitting (selecting) the described traffic or are denying (deselecting, not matching) it.

**Source Criteria**

The characteristics of the originator of the traffic you are trying to match. You must configure Source, but the other properties are optional.

**Source**

The IPv4 or IPv6 address of the source. The default is **any**, which matches all IPv4 or IPv6 addresses; you can use **any4** to target IPv4 only, or **any6** to target IPv6 only. You can specify a single host address (such as 10.100.10.5 or 2001:DB8::0DB8:800:200C:417A), a subnet (in 10.100.10.0/24 or 10.100.10.0/255.255.255.0 format, or for IPv6, 2001:DB8:0:CD30::/60), the name of a network object or network object group, the name of a network-service object group, or the name of an interface.

**User**

If you enable the identity firewall, you can specify a user or user group as the traffic source. The IP address the user is currently using will match the rule. You can specify a username (DOMAIN\user), a user group (DOMAIN\\group, note the double \ indicates a group name), or a user object group. For this field, it is far easier to click "**...**" to select names from your AAA server group than to type them in.

**Security Group**

If you enable Cisco TrustSec, you can specify a security group name or tag (1-65533), or security group object.

**More Options > Source Service**

If you specify TCP, UDP, or SCTP as the destination service, you can optionally specify a predefined service object for TCP, UDP, TCP-UDP, or SCTP, or use your own object. Typically, you define the destination service only and not the source service. Note that if you define the source service, the destination service protocol must match it (for example, both TCP, with or without port definitions).

**Destination Criteria**

The characteristics of the target of the traffic you are trying to match. You must configure Destination, but the other properties are optional.

**Destination**

The IPv4 or IPv6 address of the destination. The default is **any**, which matches all IPv4 or IPv6 addresses; you can use **any4** to target IPv4 only, or **any6** to target IPv6 only. You can specify a single host address (such as 10.100.10.5 or 2001:DB8::0DB8:800:200C:417A), a subnet (in 10.100.10.0/24 or 10.100.10.0/255.255.255.0 format, or for IPv6, 2001:DB8:0:CD30::/60), the name of a network object or network object group, the name of a network-service object group, or the name of an interface.

**Security Group**

If you enable Cisco TrustSec, you can specify a security group name or tag (1-65533), or security group object.

**Service**

The protocol of the traffic, such as IP, TCP, UDP, and optionally ports for TCP, UDP, or SCTP. The default is IP, but you can select a more specific protocol to target traffic with more granularity. Typically, you would select some type of service object. For TCP, UDP, and SCTP, you can specify ports, for example, tcp/80, tcp/http, tcp/10-20 (for a range of ports), tcp-udp/80 (match any TCP or UDP traffic on port 80), sctp/diameter, and so forth. If you include service specifications in a network-service object, specify IP in the Service field. For detailed information on specifying services, see Service Specifications in Extended ACEs, on page 10.

**Description**

A explanation of the purpose of the ACE, up to 100 characters per line. You can enter multiple lines; each line is added as a remark in the CLI, and the remarks are placed before the ACE.

**Note** If you add remarks with non-English characters on one platform (such as Windows) then try to remove them from another platform (such as Linux), you might not be able to edit or delete them because the original characters might not be correctly recognized. This limitation is due to an underlying platform dependency that encodes different language characters in different ways.

**Enable Logging; Logging Level; More Options > Logging Interval**

The logging options define how syslog messages will be generated for rules. These options apply to ACLs that are used as access rules only, that is, those attached to interfaces or applied globally. The options are ignored for ACLs used for other features. You can implement the following logging options:

**Deselect Enable Logging**

This will disable logging for the rule. No syslog messages of any type will be issued for connections that match this rule.

**Select Enable Logging with Logging Level = Default**

This provides the default logging for rules. Syslog message 106023 is issued for each denied connection. If the appliance comes under attack, the frequency of issuing this message could impact services.

**Select Enable Logging with Non-Default Logging Level**

This provides a summarized syslog message, 106100, instead of 106023. Message 106100 is issued upon first hit, then again at each interval configured in **More Options > Logging Interval** (default is every 300 seconds, you can specify 1-600), showing the number of hits during that interval. The recommended logging level is **Informational**.

Summarizing deny messages can reduce the impact of attacks and possibly make it easier for you to analyze messages. If you do come under a denial of service attack, you might see message 106101, which indicates that the number of cached deny flows used to produce the hit count for message 106100 has exceeded the maximum for an interval. At this point, the appliance stops collecting statistics until the next interval to mitigate the attack.

**More Options > Enable Rule**

Whether the rule is active on the device. Disabled rules appear with strike-through text in the rule table. Disabling a rule lets you stop its application to traffic without deleting it, so you can enable it again later if you decide you need it.

**More Options > Time Range**

The name of the time range object that defines the times of day and days of the week when the rule should be active. If you do not specify a time range, the rule is always active.

## Service Specifications in Extended ACEs

For the destination service in an extended ACE, you can specify any of the following criteria. The options are similar, but more limited, for source service, which is limited to TCP, UDP, TCP-UDP, or SCTP criteria. If you include service specifications in a network-service object, specify IP in the Service field.

**Object name**

The name of any type of service object or service object group. These objects can include many of the specifications explained below, allowing you to easily reuse service definitions among ACLs. There are many pre-defined objects, so you might find what you need without having to manually type the specification or create your own objects.

**Protocol**

A number between 1-255, or a well-known name, such as **ip**, **tcp**, **udp**, **gre**, and so forth.

**TCP, UDP, TCP-UDP, SCTP ports**

You can include port specifications on the **tcp**, **udp**, **tcp-udp**, and **sctp** keywords. The tcp-udp keyword lets you define ports for both protocols without having to specify them separately. You can use the following methods to specify ports:

- Single port—tcp/80, udp/80, tcp-udp/80, sctp/3868, or a well-known service name, such as tcp/www, udp/snmp, or sctp/diameter.
- Range of ports—tcp/1-100, udp/1-100, tcp-udp/1-100, sctp/1-100 matches ports 1-100 inclusive.
- Not equal to a port—Add != to the beginning of the specification, for example, !=tcp/80 to match any TCP traffic except TCP port 80 (HTTP).
- Less than a port number—Add <, for example <tcp/150 to match TCP traffic for any port below 150.
- Greater than a port number—Add >, for example, >tcp150 to match TCP traffic for any port above 150.

**Note** DNS, Discard, Echo, Ident, NTP, RPC, SUNRPC, and Talk each require one definition for TCP and one for UDP. TACACS+ requires one definition for port 49 on TCP.

**ICMP, ICMP6 messages**

You can target specific messages (such as ping echo request and reply messages) and even message codes. There are many pre-defined objects that cover ICMP (for IPv4) and ICMP6 (for IPv6), so you might not need to manually define the criteria. The format is:

**icmp**/*icmp_message_type*[/*icmp_message_code*]

**icmp6**/*icmp6_message_type*[/*icmp6_message_code*]

Where the message type is 1-255 or a well-known name, and the code is 0-255. Ensure that the number you select matches to an actual type/code or the ACE will never be matched.

# Configure Standard ACLs

A standard ACL is represented as a named container of ACEs. To create a new ACL, you must first create the container. Then, you can add ACEs, edit existing ACEs, and reorder the ACEs using the standard ACL table. The table can appear as a tab in the ACL Manager when you configure ACLs while configuring the policies that use them, in which case the procedures are the same except for how you get to the window.

A standard ACL uses IPv4 addresses only, and defines destination addresses only.

### Procedure

**Step 1** Choose **Configuration** > **Firewall** > **Advanced** > **Standard ACL**.

**Step 2** If you are creating a new ACL, choose **Add** > **Add ACL**, fill in a name, and click **OK**.

The ACL container is added to the table. You cannot rename a standard ACL.

**Step 3** Do any of the following:

- To add an ACE at the end of the ACL, select the ACL name or any ACE within it and choose **Add > Add ACE**.
- To insert an ACE at a specific location, select an existing ACE and choose **Add > Insert** to add the ACE above the rule, or choose **Add > Insert After**.
- To edit a rule, select it and click **Edit**.

**Step 4** Fill in the ACE properties. The options are:

- **Action: Permit/Deny**—Whether you are permitting (selecting) the described traffic or are denying (deselecting, not matching) it.
- **Address**—A definition of the destination or target address of the traffic flow. You can specify a host address such as 10.100.1.1, a network (in 10.100.1.0/24 or 10.100.1.0/255.255.255.0 format), or you can select a network object (which simply loads the contents of the object into the Address field).
- **Description**—A explanation of the purpose of the ACE, up to 100 characters per line. You can enter multiple lines; each line is added as a remark in the CLI, and the remarks are placed before the ACE.

  **Note** If you add remarks with non-English characters on one platform (such as Windows) then try to remove them from another platform (such as Linux), you might not be able to edit or delete them because the original characters might not be correctly recognized. This limitation is due to an underlying platform dependency that encodes different language characters in different ways.

When you are finished defining the ACE, click **OK** to add the rule to the table.

**Step 5** Click **Apply**.

# Configure Webtype ACLs

Webtype ACLs are used for filtering clientless SSL VPN traffic, constraining user access to specific networks, subnets, hosts, and Web servers. If you do not define a filter, all connections are allowed. A webtype ACL is represented as a named container of ACEs. To create a new ACL, you must first create the container. Then, you can add ACEs, edit existing ACEs, and reorder the ACEs using the Web ACL table. The table appears as the ACL Manager when you configure webtype ACLs while configuring the policies that use them, in which case the procedures are the same except for how you get to the window.

The webtype ACL can include a mix of IPv4 and IPv6 addresses in addition to URL specifications.

### Procedure

**Step 1** Choose **Configuration** > **Remote Access VPN** > **Clientless SSL VPN Access** > **Advanced** > **Web** > **ACLs**.

**Step 2** If you are creating a new ACL, choose **Add** > **Add ACL**, fill in a name, and click **OK**.

The ACL container is added to the table. You can later rename it by selecting it and clicking **Edit**.

**Step 3** Do any of the following:

- To add an ACE at the end of the ACL, select the ACL name or any ACE within it and choose **Add > Add ACE**.
- To insert an ACE at a specific location, select an existing ACE and choose **Add > Insert** to add the ACE above the rule, or choose **Add > Insert After**.
- To edit a rule, select it and click **Edit**.

**Step 4** Fill in the ACE properties. The primary options to select are:

- **Action: Permit/Deny**—Whether you are permitting (selecting) the described traffic or are denying (deselecting, not matching) it.
- **Filter**—The traffic matching criteria, based on the destination. You can either specify a URL by selecting the protocol and entering the server name and optionally, path and file name, or you can specify a destination IPv4 or IPv6 address and TCP service.

For detailed information on all of the available options, see Webtype ACE Properties, on page 13.

When you are finished defining the ACE, click **OK** to add the rule to the table.

**Step 5** Click **Apply**.

# Webtype ACE Properties

When you add or edit an ACE in a webtype ACL, you can configure the following properties. In many fields, you can click the "..." button on the right of the edit box to select, create, or edit objects that are available for the field.

For a given ACE, you can filter on URL or Address, but not both.

- **Action: Permit/Deny**—Whether you are permitting (selecting) the described traffic or are denying (deselecting, not matching) it.
  - **Filter on URL**—Match traffic based on destination URL. Select the protocol and enter the server name and optionally, path and file name. For example, http://www.example.com or to cover all servers, http://*.example.com. Following are some tips and limitations on specifying URLs:
    - Select **any** to match all URLs.
    - 'Permit url any' will allow all the URLs that have the format protocol://server-ip/path and will block traffic that does not match this pattern, such as port-forwarding. There should be an ACE to allow connections to the required port (port 1494 in the case of Citrix) so that an implicit deny does not occur.
    - Smart tunnel and ica plug-ins are not affected by an ACL with 'permit url any' because they match smart-tunnel:// and ica:// types only.
    - You can use these protocols: cifs://, citrix://, citrixs://, ftp://, http://, https://, imap4://, nfs://, pop3://, smart-tunnel://, and smtp://. You can also use wildcards in the protocol; for example, htt* matches http and https, and an asterisk * matches all protocols. For example, *://*.example.com matches any type URL-based traffic to the example.com network.
    - If you specify a smart-tunnel:// URL, you can include the server name only. The URL cannot contain a path. For example, smart-tunnel://www.example.com is acceptable, but smart-tunnel://www.example.com/index.html is not.
    - An asterisk * matches none or any number of characters. To match any http URL, enter http://*/*.
    - A question mark ? matches any one character exactly.
    - Square brackets [] are range operators, matching any character in the range. For example, to match both http://www.cisco.com:80/ and http://www.cisco.com:81/, enter **http://www.cisco.com:8[01]/**.
  - **Filter on Address and Service**—Match traffic based on destination address and service.
    - **Address**—The IPv4 or IPv6 address of the destination. To match all addresses, you can use **any**, which matches all IPv4 or IPv6 addresses, **any4** to match IPv4 only, or **any6** to match IPv6 only. You can specify a single host address (such as 10.100.10.5 or 2001:DB8::0DB8:800:200C:417A), a subnet (in 10.100.10.0/24 or 10.100.10.0/255.255.255.0 format, or for IPv6, 2001:DB8:0:CD30::/60), or select a network object, which fills in the field with the contents of the object.
    - **Service**—A single TCP service specification. The default is **tcp** with no ports, but you can specify a single port (such as tcp/80 or tcp/www) or port range (such as tcp/1-100). You can include operators; for example, !=tcp/80 excludes port 80; <tcp/80 is all ports less than 80; >tcp/80 is all ports greater than 80.

- **Enable Logging**; **Logging Level**; **More Options > Logging Interval**—The logging options define how syslog messages will be generated for rules that actually deny traffic. You can implement the following logging options:
  - **Deselect Enable Logging**—This will disable logging for the rule. No syslog messages of any type will be issued for traffic denied by this rule.
  - **Select Enable Logging with Logging Level = Default**—This provides the default logging for rules. Syslog message 106103 is issued for each denied packet. If the appliance comes under attack, the frequency of issuing this message could impact services.
  - **Select Enable Logging with Non-Default Logging Level**—This provides a summarized syslog message, 106102, instead of 106103. Message 106102 is issued upon first hit, then again at each interval configured in **More Options > Logging Interval** (default is every 300 seconds, you can specify 1-600), showing the number of hits during that interval. The recommended logging level is **Informational**.
- **More Options > Time Range**—The name of the time range object that defines the times of day and days of the week when the rule should be active. If you do not specify a time range, the rule is always active.

## Examples for Webtype ACLs

Following are some examples of URL-based rules for webtype ACLs.

| | Filter | Effect |
|---|---|---|
| Deny | url http://*.yahoo.com/ | Denies access to all of Yahoo! |
| Deny | url cifs://fileserver/share/directory | Denies access to all files in the specified location. |
| Deny | url https://www.example.com/directory/file.html | Denies access to the specified file. |
| Permit | url https://www.example.com/directory | Permits access to the specified location. |
| Deny | url http://*:8080/ | Denies HTTPS access to anywhere via port 8080. |
| Deny | url http://10.10.10.10 | Denies HTTP access to 10.10.10.10. |
| Permit | url any | Permits access to any URL. Usually used after an ACL that denies url access. |

# Monitoring ACLs

The ACL Manager, Standard ACL, Web ACL, and EtherType ACL tables show a consolidated view of ACLs. But to see exactly what is configured on the device, you can use the following commands. Choose **Tools > Command Line Interface** to enter the commands.

- **show access-list** [*name*]—Displays the access lists, including the line number for each ACE and hit counts. Include an ACL name or you will see all access lists.
- **show running-config access-list** [*name*]—Displays the current running access-list configuration. Include an ACL name or you will see all access lists.

# History for ACLs

| Feature Name | Releases | Description |
|---|---|---|
| Extended, standard, webtype ACLs | 7.0(1) | ACLs are used to control network access or to specify traffic for many features to act upon. An extended access control list is used for through-the-box access control and several other features. Standard ACLs are used in route maps and VPN filters. Webtype ACLs are used in clientless SSL VPN filtering. EtherType ACLs control non-IP layer 2 traffic.<br>We added the ACL Manager and other pages for configuring ACLs. |
| Real IP addresses in extended ACLs | 8.3(1) | When using NAT or PAT, mapped addresses and ports are no longer used in an ACL for several features. You must use the real, untranslated addresses and ports for these features. Using the real address and port means that if the NAT configuration changes, you do not need to change the ACLs. |
| Support for Identity Firewall in extended ACLs | 8.4(2) | You can now use identity firewall users and groups for the source and destination. You can use an identity firewall ACL with access rules, AAA rules, and for VPN authentication. |
| EtherType ACL support for IS-IS traffic | 8.4(5), 9.1(2) | In transparent firewall mode, the ASA can now control IS-IS traffic using an EtherType ACL.<br>We modified the following screen: Configuration > Device Management > Management Access > EtherType Rules. |
| Support for Cisco TrustSec in extended ACLs | 9.0(1) | You can now use Cisco TrustSec security groups for the source and destination. You can use an identity firewall ACL with access rules. |

| Feature Name | Releases | Description |
|---|---|---|
| Unified extended and webtype ACLs for IPv4 and IPv6 | 9.0(1) | Extended and webtype ACLs now support IPv4 and IPv6 addresses. You can even specify a mix of IPv4 and IPv6 addresses for the source and destination. The **any** keyword was changed to represent IPv4 and IPv6 traffic. The **any4** and **any6** keywords were added to represent IPv4-only and IPv6-only traffic, respectively. The IPv6-specific ACLs are deprecated. Existing IPv6 ACLs are migrated to extended ACLs. See the release notes for more information about migration.<br>We modified the following screens:<br>Configuration > Firewall > Access Rules<br>Configuration > Remote Access VPN > Network (Client) Access > Group Policies > General > More Options |
| Extended ACL and object enhancement to filter ICMP traffic by ICMP code | 9.0(1) | ICMP traffic can now be permitted/denied based on ICMP code.<br>We introduced or modified the following screens:<br>Configuration > Firewall > Objects > Service Objects/Groups<br>Configuration > Firewall > Access Rule |
| Configuration session for editing ACLs and objects.<br>Forward referencing of objects and ACLs in access rules. | 9.3(2) | You can now edit ACLs and objects in an isolated configuration session. You can also forward reference objects and ACLs, that is, configure rules and access groups for objects or ACLs that do not yet exist.<br>We modified the Advanced settings for access rules. |
| ACL support for Stream Control Transmission Protocol (SCTP) | 9.5(2) | You can now create ACL rules using the **sctp** protocol, including port specifications.<br>We modified the add/edit dialog boxes for access control entries on the **Configuration** > **Firewall** > **Advanced** > **ACL Manager** page. |
| Ethertype rule support for the IEEE 802.2 Logical Link Control packet's Destination Service Access Point address. | 9.6(2) | You can now write Ethertype access control rules for the IEEE 802.2 Logical Link Control packet's Destination Service Access Point address. Because of this addition, the **bpdu** keyword no longer matches the intended traffic. Rewrite **bpdu** rules for **dsap 0x42**.<br>We modified the following screen: **Configuration** > **Firewall** > **EtherType Rules**. |
| Support in routed mode for Ethertype rules on bridge group member interfaces and extended access rules on Bridge Group Virtual Interfaces (BVI). | 9.7(1) | You can now create Ethertype ACLs and apply them to bridge group member interfaces in routed mode. You can also apply extended access rules to the Bridge Virtual Interface (BVI) in addition to the member interfaces.<br>We modified the following screens: **Configuration** > **Firewall** > **Access Rules**, **Configuration** > **Firewall** > **Ethertype Rules**. |

| Feature Name | Releases | Description |
|---|---|---|
| EtherType access control list changes. | 9.9(1) | EtherType access control lists now support Ethernet II IPX (EII IPX). In addition, new keywords are added to the DSAP keyword to support common DSAP values: BPDU (0x42), IPX (0xE0), Raw IPX (0xFF), and ISIS (0xFE). Consequently, existing EtherType access control entries that use the BPDU or ISIS keywords will be converted automatically to use the DSAP specification, and rules for IPX will be converted to 3 rules (DSAP IPX, DSAP Raw IPX, and EII IPX). In addition, packet capture that uses IPX as an EtherType value has been deprecated, because IPX corresponds to 3 separate EtherTypes.<br><br>We modified the following screens: **Configuration** > **Firewall** > **Ethertype Rules**. |
| Support for network-service objects in extended ACLs. | 9.17(1) | You can use network-service objects as the source and destination criteria in extended ACLs and access control rules.<br><br>We changed the following screen: Add/Edit extended ACE or access rule on the Firewall page. |