Configuring Switch-Based Authentication

This chapter describes how to configure switch-based authentication on the Cisco Industrial Ethernet 2000U Series Switches (IE 2000U) and includes the following sections:

- Information About Switch-Based Authentication, page 1-1
- Prerequisites, page 1-17
- Guidelines and Limitations, page 1-18
- Default Settings, page 1-19
- Configuring Switch-Based Authentication, page 1-19
- Verifying Configuration, page 1-58
- Related Documents, page 1-59

Information About Switch-Based Authentication

This section includes the following topics:

- Preventing Unauthorized Access to Your Switch, page 1-1
- TACACS+, page 1-2
- RADIUS, page 1-4
- RADIUS Change of Authorization, page 1-6
- Kerberos, page 1-11
- SSH, page 1-14
- Secure HTTP Servers and Clients, page 1-15
- Secure Copy, page 1-17

Preventing Unauthorized Access to Your Switch

You can prevent unauthorized users from reconfiguring your switch and viewing configuration information. Typically, you want network administrators to have access to your switch while you restrict access to users who dial in from outside the network through an asynchronous port, connect from outside the network through a serial port, or connect through a terminal or workstation from within the local network.

To prevent unauthorized access into your switch, you should configure one or more of these security features:

- At a minimum, you should configure passwords and privileges at each switch port. The software stores these passwords locally on the switch. When users attempt to access the switch through a port or line, they must enter the password specified for the port or line before they can access the switch. For more information, see the "Protecting Access to Privileged EXEC Commands" section on page 1-19.
- For an additional layer of security, you can also configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair. For more information, see the "Configuring Username and Password Pairs" section on page 1-24.
- If you want to use username and password pairs but you want to store them centrally on a server instead of locally, you can store them in a database on a security server. Multiple networking devices can then use the same database to obtain user authentication (and, if necessary, authorization) information. For more information, see the "TACACS+" section on page 1-2.

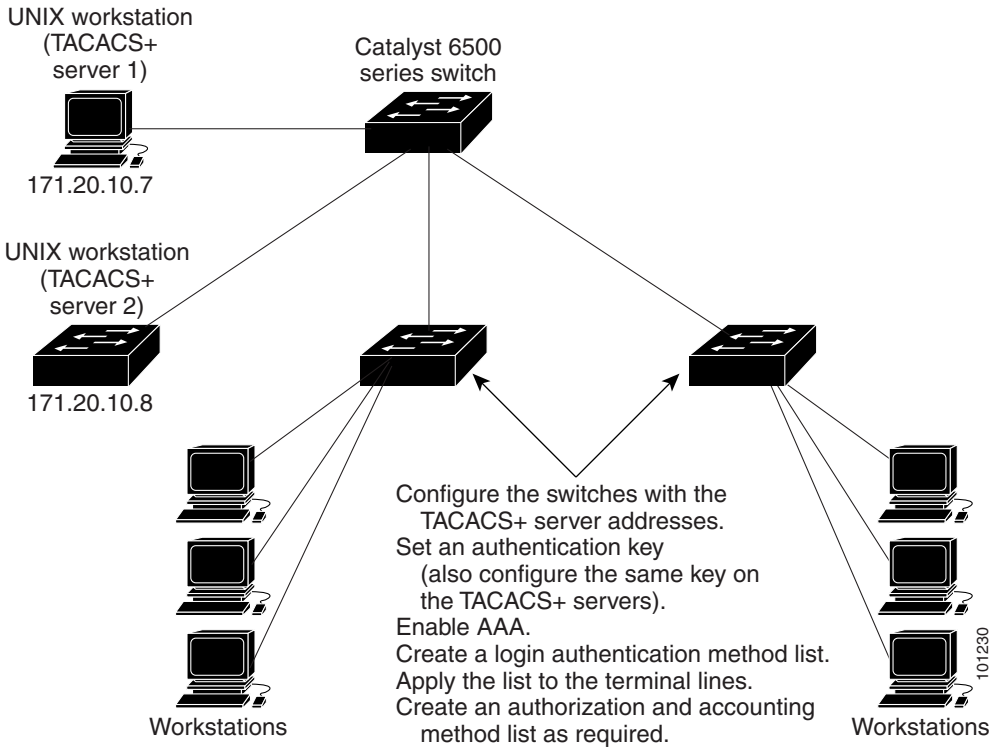
TACACS+

TACACS+ is a security application that provides centralized validation of users attempting to gain access to your switch. TACACS+ services are maintained in a database on a TACACS+ daemon typically running on a UNIX or Windows NT workstation. You should have access to and should configure a TACACS+ server before configuring TACACS+ features on your switch.

TACACS+ provides for separate and modular authentication, authorization, and accounting facilities. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide each service—authentication, authorization, and accounting—independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon.

The goal of TACACS+ is to provide a method for managing multiple network access points from a single management service. Your switch can be a network access server along with other Cisco routers and access servers. A network access server provides connections to a single user, to a network or subnetwork, and to interconnected networks as shown in Figure 1-1.

Figure 1-1 Typical TACACS+ Network Configuration



TACACS+, administered through the AAA security services, can provide these services:

- Authentication—Provides complete control of authentication through login and password dialog, challenge and response, and messaging support.

  The authentication facility can conduct a dialog with the user (for example, after a username and password are provided, to challenge a user with several questions, such as home address, mother's maiden name, service type, and social security number). The TACACS+ authentication service can also send messages to user screens. For example, a message could notify users that their passwords must be changed because of the company's password aging policy.

- Authorization—Provides fine-grained control over user capabilities for the duration of the user's session, including but not limited to setting autocommands, access control, session duration, or protocol support. You can also enforce restrictions on what commands a user can execute with the TACACS+ authorization feature.

- Accounting—Collects and sends information used for billing, auditing, and reporting to the TACACS+ daemon. Network managers can use the accounting facility to track user activity for a security audit or to provide information for user billing. Accounting records include user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

The TACACS+ protocol provides authentication between the switch and the TACACS+ daemon, and it ensures confidentiality because all protocol exchanges between the switch and the TACACS+ daemon are encrypted.

You need a system running the TACACS+ daemon software to use TACACS+ on your switch.

TACACS+ Operation

When a user attempts a simple ASCII login by authenticating to a switch using TACACS+, this process occurs:

1. When the connection is established, the switch contacts the TACACS+ daemon to obtain a username prompt to show to the user. The user enters a username, and the switch then contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, the user enters a password, and the password is then sent to the TACACS+ daemon.

   TACACS+ allows a dialog between the daemon and the user until the daemon receives enough information to authenticate the user. The daemon prompts for a username and password combination, but can include other items, such as the user's mother's maiden name.

2. The switch eventually receives one of these responses from the TACACS+ daemon:

   - ACCEPT—The user is authenticated and service can begin. If the switch is configured to require authorization, authorization begins at this time.
   - REJECT—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ daemon.
   - ERROR—An error occurred at some time during authentication with the daemon or in the network connection between the daemon and the switch. If an ERROR response is received, the switch typically tries to use an alternative method for authenticating the user.
   - CONTINUE—The user is prompted for additional authentication information.

   After authentication, the user undergoes an additional authorization phase if authorization has been enabled on the switch. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the TACACS+ daemon is again contacted, and it returns an ACCEPT or REJECT authorization response. If an ACCEPT response is returned, the response contains data in the form of attributes that direct the EXEC or NETWORK session for that user and the services that the user can access:

   - Telnet, Secure Shell (SSH), rlogin, or privileged EXEC services
   - Connection parameters, including the host or client IP address, access list, and user timeouts

RADIUS

RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information. The RADIUS host is normally a multiuser system running RADIUS server software from Cisco (Cisco Secure Access Control Server Version 3.0), Livingston, Merit, Microsoft, or another software provider. For more information, see the RADIUS server documentation.

Use RADIUS in these network environments that require access security:

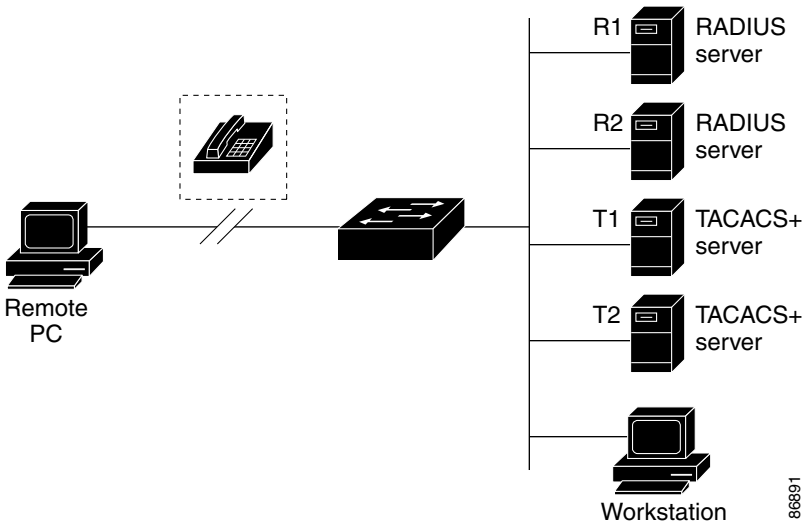
- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors' access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.
- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a smart card access control system. In one case, RADIUS has been used with Enigma's security cards to validate users and to grant access to network resources.

- Networks already using RADIUS. You can add a Cisco switch containing a RADIUS client to the network. This might be the first step when you make a transition to a TACACS+ server. See Figure 1-2 on page 1-5.
- Network in which the user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to the network, through a protocol such as IEEE 802.1x.
- Networks that require resource accounting. You can use RADIUS accounting independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, showing the amount of resources (such as time, packets, bytes, and so forth) used during the session. An Internet service provider might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

RADIUS is not suitable in these network security situations:

- Multiprotocol access environments. RADIUS does not support AppleTalk Remote Access (ARA), NetBIOS Frame Control Protocol (NBFCP), NetWare Asynchronous Services Interface (NASI), or X.25 PAD connections.
- Switch-to-switch or router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one device to a non-Cisco device if the non-Cisco device requires authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

Figure 1-2 Transitioning from RADIUS to TACACS+ Services



RADIUS Operation

When a user attempts to log in and authenticate to a switch that is access-controlled by a RADIUS server:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of these responses from the RADIUS server:
   - ACCEPT—The user is authenticated.

- REJECT—The user is either not authenticated and is prompted to reenter the username and password, or access is denied.
- CHALLENGE—A challenge requires additional data from the user.
- CHALLENGE PASSWORD—A response requests the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for privileged EXEC or network authorization. Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled. The additional data included with the ACCEPT or REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services
- Connection parameters, including the host or client IP address, access list, and user timeouts

RADIUS Change of Authorization

This section provides an overview of the RADIUS interface including available primitives and how they are used during a Change of Authorization (CoA).

- Overview, page 1-6
- Change-of-Authorization Requests, page 1-7
- CoA Request Response Code, page 1-8
- CoA Request Commands, page 1-9

Overview

A standard RADIUS interface is typically used in a pulled model, where the request originates from a network attached device and the response come from the queried servers. The switch supports the RADIUS Change of Authorization (CoA) extensions defined in RFC 5176 that are typically used in a pushed model and allow for the dynamic reconfiguring of sessions from external authentication, authorization, and accounting (AAA) or policy servers.

The switch supports these per-session CoA requests:

- Session reauthentication
- Session termination
- Session termination with port shutdown
- Session termination with port bounce

The RADIUS interface is enabled by default on the switch. However, some basic configuration is required for the following attributes:

- Security and Password—see the “Protecting Access to Privileged EXEC Commands” section on page 1-19.
- Accounting—see the “Starting RADIUS Accounting” section on page 1-42.

Change-of-Authorization Requests

Change of Authorization (CoA) requests, as described in RFC 5176, are used in a push model to allow for session identification, host reauthentication, and session termination. The model is comprised of one request (CoA-Request) and two possible response codes:

- CoA acknowledgement (ACK) [CoA-ACK]
- CoA non-acknowledgement (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a RADIUS or policy server) and directed to the switch that acts as a listener.

RFC 5176 Compliance

The Disconnect Request message, which is also referred to as Packet of Disconnect (POD), is supported by the switch for session termination. Table 1-1 shows the IETF attributes that are supported for the RADIUS CoA feature. Table 1-2 shows the possible values for the Error-Cause attribute.

Table 1-1 Supported IETF Attributes

Attribute Number	Attribute Name
24	State
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

Table 1-2 Error-Cause Values

Value	Explanation
201	Residual Session Context Removed
202	Invalid EAP Packet (Ignored)
401	Unsupported Attribute
402	Missing Attribute
403	NAS Identification Mismatch
404	Invalid Request
405	Unsupported Service
406	Unsupported Extension
407	Invalid Attribute Value
501	Administratively Prohibited
502	Request Not Routable (Proxy)
503	Session Context Not Found
504	Session Context Not Removable
505	Other Proxy Processing Error
506	Resources Unavailable

Table 1-2 Error-Cause Values (continued)

Value	Explanation
507	Request Initiated
508	Multiple Session Selection Unsupported

Preconditions

To use the CoA interface, a session must already exist on the switch. CoA can be used to identify a session and enforce a disconnect request. The update affects only the specified session.

CoA Request Response Code

The CoA Request response code can be used to convey a command to the switch. The supported commands are listed in Table 1-3 on page 1-9.

Session Identification

For disconnect and CoA requests targeted at a particular session, the switch locates the session based on one or more of the following attributes:

- Calling-Station-Id (IETF attribute 31 which contains the host MAC address)
- Audit-Session-Id (Cisco VSA)
- Acct-Session-Id (IETF attribute 44)

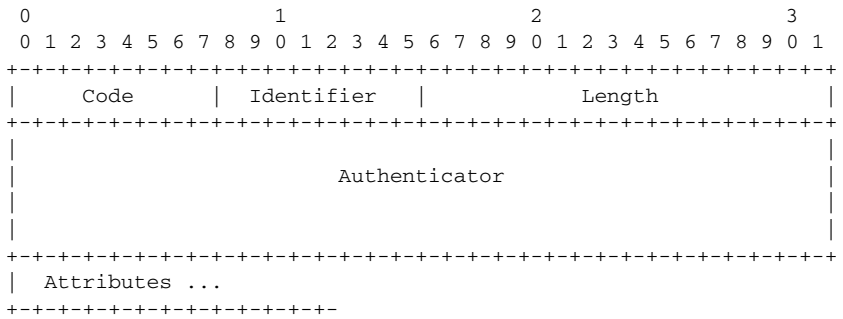
Unless all session identification attributes included in the CoA message match the session, the switch returns a Disconnect-NAK or CoA-NAK with the “Invalid Attribute Value” error-code attribute.

For disconnect and CoA requests targeted to a particular session, any one of the following session identifiers can be used:

- Calling-Station-ID (IETF attribute 31, which should contain the MAC address)
- Audit-Session-ID (Cisco vendor-specific attribute)
- Accounting-Session-ID (IETF attribute 44)

If more than one session identification attribute is included in the message, all the attributes must match the session or the switch returns a Disconnect- negative acknowledgement (NAK) or CoA-NAK with the error code “Invalid Attribute Value.”

The packet format for a CoA Request code as defined in RFC 5176 consists of the fields Code, Identifier, Length, Authenticator, and Attributes in Type:Length:Value (TLV) format.



The attributes field is used to carry Cisco VSAs.

CoA ACK Response Code

If the authorization state is changed successfully, a positive acknowledgement (ACK) is sent. The attributes returned within CoA ACK vary based on the CoA Request and are discussed in the individual CoA command descriptions.

CoA NAK Response Code

A negative acknowledgement (NAK) means that the authorization state did not change. The message can include attributes that show the reason for the failure. Use **show** commands to verify a successful CoA.

CoA Request Commands

- Session Reauthentication
- Session Termination
- CoA Disconnect-Request
- CoA Request: Disable Host Port
- CoA Request: Bounce-Port

The switch supports the commands shown in Table 1-3.

Table 1-3 CoA Commands Supported on the Switch

Command[1]	Cisco VSA
Reauthenticate host	Cisco:Avpair="subscriber:command=reauthenticate"
Terminate session	This is a standard disconnect request that does not require a VSA.
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"

1. All CoA commands must include the session identifier between the switch and the CoA client.

Session Reauthentication

The AAA server generates a session reauthentication request when a host with an unknown identity or posture joins the network and is associated with a restricted access authorization profile (such as a guest VLAN). A reauthentication request allows the host to be placed in the appropriate authorization group when its credentials are known.

To initiate session authentication, the AAA server sends a standard CoA-Request message that contains a Cisco vendor-specific attribute (VSA) in this form:
Cisco:Avpair="subscriber:command=reauthenticate" and one or more session identification attributes.

The current session state determines the switch response to the message. If the session is currently authenticated by 802.1x, the switch responds by sending an Extensible Authentication Protocol over LAN (EAPoL)-RequestId message to the server.

If the session is currently authenticated by MAC authentication bypass (MAB), the switch sends an access-request to the server, passing the same identity attributes used for the initial successful authentication.

If session authentication is in progress when the switch receives the command, the switch terminates the process, and restarts the authentication sequence, starting with the method configured to be attempted first.

If the session is not yet authorized or is authorized via guest VLAN, critical VLAN, or similar policies, the reauthentication message restarts the access control methods, beginning with the method configured to be attempted first. The current authorization of the session is maintained until the reauthentication leads to a different authorization result.

Session Termination

Three CoA types of requests can trigger session termination. A CoA Disconnect-Request terminates the session, without disabling the host port. This command causes re-initialization of the authenticator state machine for the specified host, but does not restrict its access to the network.

To restrict access to the network, use a CoA Request with the Cisco:Avpair="subscriber:command=disable-host-port" VSA. This command is useful when a host is known to be causing problems on the network and you need to immediately block network access. When you want to restore network access on the port, reenable it using a non-RADIUS mechanism.

When a device with no supplicant, such as a printer, needs a new IP address (for example, after a VLAN change), end the session on the host port with port-bounce (temporarily disable and then reenable the port).

CoA Disconnect-Request

Because this Disconnect-Request command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the "Session Identification" section on page 1-8. If the session cannot be located, the switch returns a Disconnect-NAK message with the "Session Context Not Found" error-code attribute. If the session *is* located, the switch terminates the session. After the session has been completely removed, the switch returns a Disconnect-ACK.

If the switch fails-over to a standby switch before returning a Disconnect-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the session is still not found, a Disconnect-ACK is sent with the "Session Context Not Found" error-code attribute.

CoA Request: Disable Host Port

This command is carried in a standard CoA-Request message that has this new VSA:

Cisco:Avpair="subscriber:command=disable-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the "Session Identification" section on page 1-8. If the session cannot be located, the switch returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch disables the hosting port and returns a CoA-ACK message.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is restarted on the new active switch.

Note A Disconnect-Request failure following command resending could be the result of either a successful session termination before change-over (if the Disconnect-ACK was not sent) or a session termination by other means (for example, a link failure) that occurred after the original command was issued and before the standby switch became active.

CoA Request: Bounce-Port

This command is carried in a standard CoA-Request message that contains the following new VSA:

Cisco:Avpair="subscriber:command=bounce-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the "Session Identification" section on page 1-8. If the session cannot be located, the switch returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch disables the hosting port for a period of 10 seconds, reenables it (port-bounce), and returns a CoA-ACK.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is re-started on the new active switch.

Kerberos

Kerberos is a secret-key network authentication protocol, which was developed at the Massachusetts Institute of Technology (MIT). It uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication and authenticates requests for network resources. Kerberos uses the concept of a trusted third party to perform secure verification of users and services. This trusted third party is called the *key distribution center* (KDC).

Kerberos verifies that users are who they claim to be and the network services that they use are what the services claim to be. To do this, a KDC or trusted Kerberos server issues tickets to users. These tickets, which have a limited lifespan, are stored in user credential caches. The Kerberos server uses the tickets instead of usernames and passwords to authenticate users and network services.

Note A Kerberos server can be an IE 2000U switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

The Kerberos credential scheme uses a process called *single logon*. This process authenticates a user once and then allows secure authentication (without encrypting another password) wherever that user credential is accepted.

This software release supports Kerberos 5, which allows organizations that are already using Kerberos 5 to use the same Kerberos authentication database on the KDC that they are already using on their other network hosts (such as UNIX servers and PCs).

In this software release, Kerberos supports these network services:

- Telnet
- rlogin
- rsh (Remote Shell Protocol)

Table 1-4 lists the common Kerberos-related terms and definitions.

Table 1-4 Kerberos Terms

Term	Definition
Authentication	A process by which a user or service identifies itself to another service. For example, a client can authenticate to a switch or a switch can authenticate to another switch.
Authorization	A means by which the switch identifies what privileges the user has in a network or on the switch and what actions the user can perform.
Credential	A general term that refers to authentication tickets, such as TGTs[1] and service credentials. Kerberos credentials verify the identity of a user or service. If a network service decides to trust the Kerberos server that issued a ticket, it can be used in place of reentering a username and password. Credentials have a default lifespan of eight hours.
Instance	An authorization-level label for Kerberos principals. Most Kerberos principals are of the form *user@REALM* (for example, smith@EXAMPLE.COM). A Kerberos principal with a Kerberos instance has the form *user/instance@REALM* (for example, smith/admin@EXAMPLE.COM). The Kerberos instance can be used to specify the authorization level for the user if authentication is successful. The server of each network service might implement and enforce the authorization mappings of Kerberos instances but is not required to do so. **Note** The Kerberos principal and instance names *must* be in all lowercase characters. **Note** The Kerberos realm name *must* be in all uppercase characters.
KDC[2]	Key distribution center that consists of a Kerberos server and database program that is running on a network host.
Kerberized	A term that describes applications and services that have been modified to support the Kerberos credential infrastructure.
Kerberos realm	A domain consisting of users, hosts, and network services that are registered to a Kerberos server. The Kerberos server is trusted to verify the identity of a user or network service to another user or network service. **Note** The Kerberos realm name *must* be in all uppercase characters.
Kerberos server	A daemon that is running on a network host. Users and network services register their identity with the Kerberos server. Network services query the Kerberos server to authenticate to other network services.
KEYTAB[3]	A password that a network service shares with the KDC. In Kerberos 5 and later Kerberos versions, the network service authenticates an encrypted service credential by using the KEYTAB to decrypt it. In Kerberos versions earlier than Kerberos 5, KEYTAB is referred to as SRVTAB[4].
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. **Note** The Kerberos principal name *must* be in all lowercase characters.

Table 1-4 Kerberos Terms (continued)

Term	Definition
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

1. TGT = ticket granting ticket
2. KDC = key distribution center
3. KEYTAB = key table
4. SRVTAB = server table

Kerberos Operation

A Kerberos server can be a Cisco IE 2000U switch that is configured as a network security server and that can authenticate remote users by using the Kerberos protocol. Although you can customize Kerberos in a number of ways, remote users attempting to access network services must pass through three layers of security before they can access network services.

To authenticate to network services by using a IE 2000U switch as a Kerberos server, remote users must follow these steps:

1. Authenticating to a Boundary Switch, page 1-13
2. Obtaining a TGT from a KDC, page 1-14
3. Authenticating to Network Services, page 1-14

Authenticating to a Boundary Switch

This section describes the first layer of security through which a remote user must pass. The user must first authenticate to the boundary switch. This process then occurs:

1. The user opens an un-Kerberized Telnet connection to the boundary switch.
2. The switch prompts the user for a username and password.
3. The switch requests a TGT from the KDC for this user.
4. The KDC sends an encrypted TGT that includes the user identity to the switch.
5. The switch attempts to decrypt the TGT by using the password that the user entered.
   - If the decryption is successful, the user is authenticated to the switch.
   - If the decryption is not successful, the user repeats Step 2 either by reentering the username and password (noting if Caps Lock or Num Lock is on or off) or by entering a different username and password.

A remote user who initiates a un-Kerberized Telnet session and authenticates to a boundary switch is inside the firewall, but the user must still authenticate directly to the KDC before getting access to the network services. The user must authenticate to the KDC because the TGT that the KDC issues is stored on the switch and cannot be used for additional authentication until the user logs on to the switch.

Obtaining a TGT from a KDC

This section describes the second layer of security through which a remote user must pass. The user must now authenticate to a KDC and obtain a TGT from the KDC to access network services.

For instructions about how to authenticate to a KDC, see the "Obtaining a TGT from a KDC" section in the "Configuring Kerberos" chapter of the User Security Configuration Guide, Cisco IOS Release 15MT.

Authenticating to Network Services

This section describes the third layer of security through which a remote user must pass. The user with a TGT must now authenticate to the network services in a Kerberos realm.

For instructions about how to authenticate to a network service, see the "Authenticating to Network Services" section in the "Configuring Kerberos" chapter of the User Security Configuration Guide, Cisco IOS Release 15MT.

SSH

SSH is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

SSH Servers, Integrated Clients, and Supported Versions

The SSH feature has an SSH server and an SSH integrated client, which are applications that run on the switch. You can use an SSH client to connect to a switch running the SSH server. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client also works with the SSH server supported in this release and with non-Cisco SSH servers.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.

SSH supports the Data Encryption Standard (DES) encryption algorithm, the Triple DES (3DES) encryption algorithm, and password-based user authentication.

SSH also supports these user authentication methods:

- TACACS+ (for more information, see the "Configuring TACACS+" section on page 1-28)
- RADIUS (for more information, see the "Configuring RADIUS" section on page 1-34)
- Local authentication and authorization (for more information, see the "Configuring the Switch for Local Authentication and Authorization" section on page 1-49)

Note This software release does not support IP Security (IPsec).

Secure HTTP Servers and Clients

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a web browser. Cisco uses an implementation of SSL Version 3.0 with application-layer encryption to run the secure HTTP server and secure HTTP client. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with https:// instead of http://.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 web server. The HTTP 1.1 server processes requests and passes responses (pages) back to the HTTP secure server, which, in turn, responds to the original request.

The primary role of the HTTP secure client (the web browser) is to respond to Cisco IOS application requests for HTTPS User Agent services, perform HTTPS User Agent services for the application, and pass the response back to the application.

Certificate Authority Trustpoints

Certificate authorities (CAs) manage certificate requests and issue certificates to participating network devices. These services provide centralized security key and certificate management for the participating devices. Specific CA servers are referred to as *trustpoints*.

When a connection attempt is made, the HTTPS server provides a secure connection by issuing a certified X.509v3 certificate, obtained from a specified CA trustpoint, to the client. The client (usually a web browser), in turn, has a public key that allows it to authenticate the certificate.

For secure HTTP connections, we highly recommend that you configure a CA trustpoint. If a CA trustpoint is not configured for the device running the HTTPS server, the server certifies itself and generates the needed RSA key pair. Because a self-certified (self-signed) certificate does not provide adequate security, the connecting client generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection. This option is useful for internal network topologies (such as testing).

If you do not configure a CA trustpoint, when you enable a secure HTTP connection, either a temporary or a persistent self-signed certificate for the secure HTTP server (or client) is automatically generated.

- If the switch is not configured with a hostname and a domain name, a temporary self-signed certificate is generated. If the switch reboots, any temporary self-signed certificate is lost, and a new temporary new self-signed certificate is assigned.
- If the switch has been configured with a host and domain name, a persistent self-signed certificate is generated. This certificate remains active if you reboot the switch or if you disable the secure HTTP server so that it will be there the next time you reenable a secure HTTP connection.

Note The certificate authorities and trustpoints must be configured on each device individually. Copying them from other devices makes them invalid on the switch.

If a self-signed certificate has been generated, this information is included in the output of the **show running-config** privileged EXEC command. This is a partial sample output from that command displaying a self-signed certificate.

```
Switch# show running-config
Building configuration...

<output truncated>
```

```
crypto pki trustpoint TP-self-signed-3080755072
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-3080755072
 revocation-check none
 rsakeypair TP-self-signed-3080755072
!
!
crypto ca certificate chain TP-self-signed-3080755072
 certificate self-signed 01
  3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109

<output truncated>
```

You can remove this self-signed certificate by disabling the secure HTTP server and entering the **no crypto pki trustpoint TP-self-signed-30890755072** global configuration command. If you later reenable a secure HTTP server, a new self-signed certificate is generated.

Note The values that follow *TP self-signed* depend on the serial number of the device.

You can use an optional command (**ip http secure-client-auth**) to allow the HTTPS server to request an X.509v3 certificate from the client. Authenticating the client provides more security than server authentication by itself.

For additional information on Certificate Authorities, see the "Configuring Certification Authority Interoperability" chapter in the *Public Key Infrastructure Configuration Guide, Cisco IOS Release 15MT*.

CipherSuites

A CipherSuite specifies the encryption algorithm and the digest algorithm to use on a SSL connection. When connecting to the HTTPS server, the client web browser offers a list of supported CipherSuites, and the client and server negotiate the best encryption algorithm.

Note To ensure the best results with encryption, please use the recommended web browsers listed in the release notes for this product.

The more secure and more complex CipherSuites require slightly more processing time. This list defines the CipherSuites supported by the switch and ranks them from fastest to slowest in terms of router processing load (speed):

1. SSL_RSA_WITH_DES_CBC_SHA—RSA key exchange (RSA Public Key Cryptography) with DES-CBC for message encryption and SHA for message digest
2. SSL_RSA_WITH_RC4_128_MD5—RSA key exchange with RC4 128-bit encryption and MD5 for message digest
3. SSL_RSA_WITH_RC4_128_SHA—RSA key exchange with RC4 128-bit encryption and SHA for message digest
4. SSL_RSA_WITH_3DES_EDE_CBC_SHA—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest

RSA (with the specified encryption and digest algorithm combinations) is used for both key generation and authentication on SSL connections. This usage is independent of whether or not a CA trustpoint is configured.

Secure Copy

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provides a secure replacement for the Berkeley r-tools.

For SSH to work, the switch needs an RSA public/private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH also relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.

Note When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.

A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.

For more information on how to configure and verify SCP, see the "Secure Copy Protocol" chapter of the *Secure Shell Configuration Guide, Cisco IOS Release 15M&T*.

Prerequisites

- You should have access to and should configure a TACACS+ server before configuring TACACS+ features on your switch.
- You should have access to and should configure a RADIUS server before configuring RADIUS features on your switch.
- To use Kerberos, SSH, and SSL, you must install the cryptographic (encrypted) software image on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information, see the release notes for this release.

Guidelines and Limitations

Setting Password Encryption

If you enable password encryption by using the **enable secret** command, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords. (See "Protecting Enable and Enable Secret Passwords with Encryption" section on page 1-21.)

- We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.
- If you configure the **enable secret** command, it takes precedence over the **enable password** (unencrypted) command; the two commands cannot be in effect simultaneously.

TACACS+

To prevent a lapse in security, you cannot configure TACACS+ through a network management application. When enabled, TACACS+ can authenticate users accessing the switch through the CLI.

Note Although TACACS+ configuration is performed through the CLI, the TACACS+ server authenticates HTTP connections that have been configured with a privilege level of 15.

RADIUS

To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the switch through the CLI.

SSH

These limitations apply to SSH:

- The switch supports RSA authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on DES (56-bit) and 3DES (168-bit) data encryption software.
- The switch supports the Advanced Encryption Standard (AES) encryption algorithm with a 128-bit key, 192-bit key, or 256-bit key. However, symmetric cipher AES to encrypt the keys is not supported.

SSL

- When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.
- Before you configure a CA trustpoint, ensure that the system clock is set. If the clock is not set, the certificate is rejected due to an incorrect date.

Default Settings

Feature	Default Setting
Default Password and Privilege Level	
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.
Security Application	
TACACS+ and AAA	Disabled.
RADIUS and AAA	Disabled.
SSL	The standard HTTP server is disabled. (Use the **ip http server** global configuration command to enable the standard HTTP server.) SSL is enabled. No CA trustpoints are configured. No self-signed certificates are generated.

Configuring Switch-Based Authentication

This section includes the following topics:

- Protecting Access to Privileged EXEC Commands, page 1-19
- Configuring TACACS+, page 1-28
- Configuring RADIUS, page 1-34
- Configuring Kerberos, page 1-48
- Configuring the Switch for Local Authentication and Authorization, page 1-49
- Configuring SSH, page 1-50
- Configuring Secure HTTP Servers and Clients, page 1-54

Protecting Access to Privileged EXEC Commands

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

- Setting or Changing a Static Enable Password, page 1-20
- Protecting Enable and Enable Secret Passwords with Encryption, page 1-21
- Disabling Password Recovery, page 1-22

- Setting a Telnet Password for a Terminal Line, page 1-23
- Configuring Username and Password Pairs, page 1-24
- Configuring Multiple Privilege Levels, page 1-25

Setting or Changing a Static Enable Password

The **enable password** command controls access to the privileged EXEC mode.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**enable password** *password*	Define a new password or change an existing password for access to privileged EXEC mode. By default, no password is defined. For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when you create the password; for example, to create the password abc?123, do this: Enter **abc**. Enter **Crtl-v**. Enter **?123**. When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file. The enable password is not encrypted and can be read in the switch configuration file.

To remove the password, use the **no enable password** global configuration command.

EXAMPLE

This example shows how to change the enable password to *l1u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Switch(config)# enable password l1u2c3k4y5
```

Protecting Enable and Enable Secret Passwords with Encryption

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a Trivial File Transfer Protocol (TFTP) server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}	Define a new password or change an existing password for access to privileged EXEC mode.
	or	or
	enable secret [**level** *level*] {*password* | *encryption-type encrypted-password*}	Define a secret password, which is saved using a nonreversible encryption method. • (Optional) For *level*, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges). • For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined. • (Optional) For *encryption-type*, only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password that you copy from another switch configuration. **Note** If you specify an encryption type and then enter a clear text password, you can not reenter privileged EXEC mode. You cannot recover a lost encrypted password by any method.

	Command	Purpose
Step 3	**service password-encryption**	(Optional) Encrypt the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If both the enable and enable secret passwords are defined, users must enter the enable secret password.

Use the **level** keyword to define a password for a specific privilege level. After you specify the level and set a password, give the password only to users who need to have access at this level. Use the **privilege level** global configuration command to specify commands accessible at various levels. For more information, see the "Configuring Multiple Privilege Levels" section on page 1-25.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

To remove a password and level, use the **no enable password** [**level** *level*] or **no enable secret** [**level** *level*] global configuration command. To disable password encryption, use the **no service password-encryption** global configuration command.

EXAMPLE

This example shows how to configure the encrypted password *1FaD0$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Switch(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Disabling Password Recovery

By default, any end user with physical access to the switch can recover from a lost password by interrupting the boot process while the switch is powering on and then entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

Note If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. We recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the XMODEM protocol.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no service password-recovery**	Disable password recovery. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show version**	Verify the configuration by checking the last few lines of the command output.

To reenable password recovery, use the **service password-recovery** global configuration command.

Note Disabling password recovery will not work if you have set the switch to boot manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:*) after the switch is power cycled.

EXAMPLE

```
Switch(config)# no service password-recovery
Switch(config)# end
```

Setting a Telnet Password for a Terminal Line

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you did not configure this password during the setup program, you can configure it now through the command-line interface (CLI).

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1		Attach a PC or workstation with emulation software to the switch console port. The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.
Step 2	**enable password** *password*	Enter privileged EXEC mode.
Step 3	**configure terminal**	Enter global configuration mode.

	Command	Purpose
Step 4	**line vty 0 15**	Configure the number of Telnet sessions (lines), and enter line configuration mode. There are 16 possible sessions on a command-capable switch. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
Step 5	**password** *password*	Enter a Telnet password for the line or lines. For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries. The password is listed under the command **line vty 0 15**.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the password, use the **no password** global configuration command.

EXAMPLE

This example shows how to set the Telnet password to *let45me67in89*:

```
Switch(config)# line vty 10
Switch(config-line)# password let45me67in89
```

Configuring Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	Enter the username, privilege level, and password for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. • For *password*, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
Step 3	**line console 0** or **line vty 0 15**	Enter line configuration mode, and configure the console port (line 0) or the VTY lines (line 0 to 15).
Step 4	**login local**	Enable local password checking at login time. Authentication is based on the username specified in Step 2.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable username authentication for a specific user, use the **no username** *name* global configuration command. To disable password checking and allow connections without a password, use the **no login** line configuration command.

EXAMPLE

```
Switch(config)# username user privilege 0 password 0 cisco
Switch(config)# line console 0
Switch(config-line)# login local
Switch(config-line)# end
```

Configuring Multiple Privilege Levels

By default, the Cisco IOS software has two modes of password security: user EXEC and privileged EXEC. You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

This section includes the following topics:

- Setting the Privilege Level for a Command, page 1-26
- Changing the Default Privilege Level for Lines, page 1-27
- Logging into and Exiting a Privilege Level, page 1-28

Setting the Privilege Level for a Command

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**privilege** *mode* **level** *level command*	Set the privilege level for a command. • For *mode*, enter **configure** for global configuration mode, **exec** for EXEC mode, **interface** for interface configuration mode, or **line** for line configuration mode. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password. • For *command*, specify the command to which you want to restrict access.
Step 3	**enable password level** *level password*	Specify the enable password for the privilege level. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. • For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config** or **show privilege**	Verify your entries. The first command shows the password and access level configuration. The second command shows the privilege level configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

To return to the default privilege for a given command, use the **no privilege** *mode* **level** *level command* global configuration command.

EXAMPLE

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Switch(config)# privilege exec level 14 configure
Switch(config)# enable password level 14 SecretPswd14
```

Changing the Default Privilege Level for Lines

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**line vty** *line*	Select the virtual terminal line on which to restrict access.
Step 3	**privilege level** *level*	Change the default privilege level for the line. For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config** or **show privilege**	Verify your entries. The first command shows the password and access level configuration. The second command shows the privilege level configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

To return to the default line privilege level, use the **no privilege level** line configuration command.

EXAMPLE

```
Switch(config)# line vty 1
Switch(config-line)# privilege level 1
Switch(config-line)# end
```

Logging into and Exiting a Privilege Level

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**enable** *level*	Log in to a specified privilege level. For *level*, the range is 0 to 15.
Step 2	**disable** *level*	Exit to a specified privilege level. For *level*, the range is 0 to 15.

EXAMPLE

In the following example, the user enters privileged EXEC mode (changes to privilege level 15) by using the **enable** command without a privilege-level argument. The system prompts the user for a password before allowing access to the privileged EXEC mode. The password is not printed to the screen. The user then exits back to user EXEC mode using the disable command. Note that the prompt for user EXEC mode is the greater than symbol (>), and the prompt for privileged EXEC mode is the number sign (#).

```
Switch> enable
Password: <letmein>
Switch# disable
Switch>
```

Configuring TACACS+

This section describes how to enable and configure Terminal Access Controller Access Control System Plus (TACACS+), which provides detailed accounting information and flexible administrative control over authentication and authorization processes. TACACS+ is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.

To configure your switch to support TACACS+, you must identify the host or hosts maintaining the TACACS+ daemon and define the method lists for TACACS+ authentication. You can optionally define method lists for TACACS+ authorization and accounting. A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used, thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

The **aaa authorization console** global configuration command allows you to enable AAA and TACACS+ to work on the console port.

This section includes the following topics:

- Identifying the TACACS+ Server Host and Setting the Authentication Key, page 1-29
- Configuring TACACS+ Login Authentication, page 1-30

- Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services, page 1-32
- Starting TACACS+ Accounting, page 1-33

Identifying the TACACS+ Server Host and Setting the Authentication Key

You can configure the switch to use a single server or AAA server groups to group existing server hosts for authentication. You can group servers to select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list and contains the list of IP addresses of the selected server hosts.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**tacacs-server host** *hostname* [**port** *integer*] [**timeout** *integer*] [**key** *string*]	Identify the IP host or hosts maintaining a TACACS+ server. Enter this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them. • For *hostname*, specify the name or IP address of the host. • (Optional) For **port** *integer*, specify a server port number. The default is port 49. The range is 1 to 65535. • (Optional) For **timeout** *integer*, specify a time, in seconds, that the switch waits for a response from the daemon before it times out and declares an error. The default is 5 seconds. The range is 1 to 1000 seconds. • (Optional) For **key** *string*, specify the encryption key for encrypting and decrypting all traffic between the switch and the TACACS+ daemon. You must configure the same key on the TACACS+ daemon for encryption to be successful.
Step 3	**aaa new-model**	Enable AAA.
Step 4	**aaa group server tacacs+** *group-name*	(Optional) Define the AAA server-group with a group name. This command puts the switch in a server group subconfiguration mode.
Step 5	**server** *ip-address*	(Optional) Associate a particular TACACS+ server with the defined server group. Repeat this step for each TACACS+ server in the AAA server group. Each server in the group must be previously defined in Step 2.

	Command	Purpose
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show tacacs**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified TACACS+ server name or address, use the **no tacacs-server host** *hostname* global configuration command. To remove a server group from the configuration list, use the **no aaa group server tacacs+** *group-name* global configuration command. To remove the IP address of a TACACS+ server, use the **no server ip-address** server group subconfiguration command.

EXAMPLE

The following example specifies that, for authentication, authorization, and accounting (AAA) confirmation, the router consults the TACACS+ server host named Sea_Cure on port number 51. The timeout value for requests on this connection is three seconds; the encryption key is a_secret.

```
Switch(config)# tacacs-server host Sea_Cure port 51 timeout 3 key a_secret
Switch(config)# aaa new-model
Switch(config)# aaa group server tacacs+
Switch(config)# tacgroup1
Switch(config)# server 0.1.1.1
```

Configuring TACACS+ Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined. A defined method list overrides the default method list.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

BEFORE YOU BEGIN

Complete the "Identifying the TACACS+ Server Host and Setting the Authentication Key" procedure on page 1-29.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.

	Command	Purpose
Step 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	Create a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: • **enable**—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. • **group tacacs+**—Use TACACS+ authentication. Before you can use this authentication method, you must configure the TACACS+ server. For more information, see the "Identifying the TACACS+ Server Host and Setting the Authentication Key" section on page 1-29. • **line**—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. • **local**—Use the local username database for authentication. You must enter username information in the database. Use the **username** *password* global configuration command. • **local-case**—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *name* **password** global configuration command. • **none**—Do not use any authentication for login.
Step 4	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enter line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 5	**login authentication** {**default** | *list-name*}	Apply the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 6	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable AAA authentication, use the **no aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] global configuration command. To either disable TACACS+ authentication for logins or to return to the default value, use the **no login authentication** {**default** | *list-name*} line configuration command.

EXAMPLE

The following example shows how to create an AAA authentication list called MIS-access. This authentication first tries to contact a TACACS+ server. If no server is found, TACACS+ returns an error and AAA tries to use the enable password. If this attempt also returns an error (because no enable password is configured on the server), the user is allowed access with no authentication.

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication login MIS-access group tacacs+ enable none
Switch(config)# line vty 0 4
Switch(config-line)# login authentication MIS-access
Switch(config-line)# end
```

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is located either in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

You can use the **aaa authorization** global configuration command with the **tacacs+** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec tacacs+ local** command sets these authorization parameters:

- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa authorization network tacacs+**	Configure the switch for user TACACS+ authorization for all network-related service requests.
Step 3	**aaa authorization exec tacacs+**	Configure the switch for user TACACS+ authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

EXAMPLE

```
Switch(config)# aaa authorization network tacacs+
Switch(config)# aaa authorization exec tacacs+
Switch(config-line)# end
```

Starting TACACS+ Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Follow this procedure to enable TACACS+ accounting for each Cisco IOS privilege level and for network services.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa accounting network start-stop tacacs+**	Enable TACACS+ accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop tacacs+**	Enable TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.

	Command	Purpose
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable accounting, use the **no aaa accounting** {**network** | **exec**} {**start-stop**} *method1*... global configuration command.

EXAMPLE

```
Switch(config)# aaa accounting network start-stop tacacs+
Switch(config)# aaa accounting exec start-stop tacacs+
Switch(config-line)# end
```

Configuring RADIUS

This section describes how to enable and configure RADIUS, which provides detailed accounting information and flexible administrative control over authentication and authorization processes. RADIUS is facilitated through AAA and can be enabled only through AAA commands.

To configure your switch to support RADIUS, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.

A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used (such as TACACS+ or local username lookup), thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

You should have access to and should configure a RADIUS server before configuring RADIUS features on your switch.

This section includes the following topics:

- Identifying the RADIUS Server Host, page 1-35 (required)
- Configuring RADIUS Login Authentication, page 1-37 (required)
- Defining AAA Server Groups, page 1-39 (optional)
- Configuring RADIUS Authorization for User Privileged Access and Network Services, page 1-41 (optional)
- Starting RADIUS Accounting, page 1-42 (optional)
- Configuring Settings for All RADIUS Servers, page 1-43 (optional)
- Configuring the Switch to Use Vendor-Specific RADIUS Attributes, page 1-44 (optional)
- Configuring the Switch for Vendor-Proprietary RADIUS Server Communication, page 1-46 (optional)
- Configuring CoA on the Switch, page 1-47
- Monitoring and Troubleshooting CoA Functionality, page 1-48
- Configuring RADIUS Server Load Balancing, page 1-48 (optional)

Identifying the RADIUS Server Host

Switch-to-RADIUS-server communication involves several components:

- Hostname or IP address
- Authentication destination port
- Accounting destination port
- Key string
- Timeout period
- Retransmission value

You identify RADIUS security servers by their hostname or IP address, hostname and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the switch tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order that they are configured.)

A RADIUS server and the switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the switch.

The timeout, retransmission, and encryption key values can be configured globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings. To apply these settings globally to all RADIUS servers communicating with the switch, use the three unique global configuration commands: **radius-server timeout**, **radius-server retransmit**, and **radius-server key**. To apply these values on a specific RADIUS server, use the **radius-server host** global configuration command.

You can configure the switch to use AAA server groups to group existing server hosts for authentication. For more information, see the "Defining AAA Server Groups" section on page 1-39.

Follow this procedure to configure per-server RADIUS server communication. This procedure is required.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA authentication.

	Command	Purpose
Step 3	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specify the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified RADIUS server, use the **no radius-server host** *hostname* | *ip-address* global configuration command.

EXAMPLE

This example shows how to configure one RADIUS server to be used for authentication and another to be used for accounting:

```
Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Switch(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

This example shows how to configure *host1* as the RADIUS server and to use the default ports for both authentication and accounting:

```
Switch(config)# radius-server host host1
```

Note You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Configuring RADIUS Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

Follow this procedure to configure login authentication. This procedure is required.

BEFORE YOU BEGIN

Before defining a method list that uses any of the following authentication methods, perform the required configuration for that method:

- Enable password—Define an enable password by using the **enable** *password* global configuration command.
- RADIUS authentication —Configure the RADIUS server. For more information, see the "Identifying the RADIUS Server Host" section on page 1-35.
- Line password—Define a line password by using the **password** *password* line configuration command.
- Local username database—Enter username information in the database using the **username** *name* **password** global configuration command.
- Case-sensitive local username database—Enter username information in the database by using the **username** *password* global configuration command.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	Create a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: – **enable**—Use the enable password for authentication. – **group radius**—Use RADIUS authentication. – **line**—Use the line password for authentication. – **local**—Use the local username database for authentication. – **local-case**—Use a case-sensitive local username database for authentication. – **none**—Do not use any authentication for login.
Step 4	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enter line configuration mode, and configure the lines to which you want to apply the authentication list.

	Command	Purpose
Step 5	**login authentication** {**default** | *list-name*}	Apply the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable AAA authentication, use the **no aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] global configuration command. To either disable RADIUS authentication for logins or to return to the default value, use the **no login authentication** {**default** | *list-name*} line configuration command.

EXAMPLE

The following example shows how to create a default AAA authentication list. This authentication first tries to contact a RADIUS server. If no server is found, RADIUS returns an error and AAA tries to use the password in the local username database.

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication login default group radius local
Switch(config)# login authentication default
Switch(config)# end
```

Defining AAA Server Groups

You can configure the switch to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. If you configure two different host entries on the same RADIUS server for the same service, (for example, accounting), the second configured host entry acts as a fail-over backup to the first one.

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specify the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 3	**aaa new-model**	Enable AAA.
Step 4	**aaa group server radius** *group-name*	Define the AAA server-group with a group name. This command puts the switch in a server group configuration mode.

	Command	Purpose
Step 5	**server** *ip-address*	Associate a particular RADIUS server with the defined server group. Repeat this step for each RADIUS server in the AAA server group. Each server in the group must be previously defined in Step 2.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 9		Enable RADIUS login authentication. See the "Configuring RADIUS Login Authentication" section on page 1-37.

To remove the specified RADIUS server, use the **no radius-server host** *hostname* | *ip-address* global configuration command. To remove a server group from the configuration list, use the **no aaa group server radius** *group-name* global configuration command. To remove the IP address of a RADIUS server, use the **no server** *ip-address* server group configuration command.

EXAMPLE

In this example, the switch is configured to recognize two different RADIUS group servers (*group1* and *group2*). Group1 has two different host entries on the same RADIUS server configured for the same services. The second host entry acts as a fail-over backup to the first entry.

```
Switch(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Switch(config)# aaa new-model
Switch(config)# aaa group server radius group1
Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config-sg-radius)# exit
Switch(config)# aaa group server radius group2
Switch(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Switch(config-sg-radius)# exit
```

Configuring RADIUS Authorization for User Privileged Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa authorization network radius**	Configure the switch for user RADIUS authorization for all network-related service requests.
Step 3	**aaa authorization exec radius**	Configure the switch for user RADIUS authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

EXAMPLE

```
Switch(config)# aaa authorization network radius
Switch(config)# aaa authorization exec radius
Switch(config)# end
```

Starting RADIUS Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Follow this procedure to enable RADIUS accounting for each Cisco IOS privilege level and for network services.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa accounting network start-stop radius**	Enable RADIUS accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop radius**	Enable RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable accounting, use the **no aaa accounting** {**network** | **exec**} {**start-stop**} *method1...* global configuration command.

EXAMPLE

```
Switch(config)# aaa accounting network start-stop radius
Switch(config)# aaa accounting exec start-stop radius
Switch(config)# end
```

Configuring Settings for All RADIUS Servers

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server key** *string*	Specify the shared secret text string used between the switch and all RADIUS servers. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 3	**radius-server retransmit** *retries*	Specify the number of times the switch sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
Step 4	**radius-server timeout** *seconds*	Specify the number of seconds a switch waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.

	Command	Purpose
Step 5	**radius-server deadtime** *minutes*	Specify the length of time, in minutes, for which a RADIUS server that is not responding to authentication requests is skipped over by transaction requests. Setting the deadtime avoids the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your settings.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting for the retransmit, timeout, and deadtime, use the **no** forms of these commands.

EXAMPLE

```
Switch(config)# radius-server key key1
Switch(config)# radius-server retransmit 5
Switch(config)# radius-server timeout 10
Switch(config)# radius-server deadtime 5
Switch(config)# end
```

Configuring the Switch to Use Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the switch and the RADIUS server by using the vendor-specific attribute (attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option by using the format recommended in the specification. Cisco's vendor-ID is 9, and the supported option has vendor-type 1, which is named *cisco-avpair*. The value is a string with this format:

```
protocol : attribute sep value *
```

Protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attribute-value (AV) pair defined in the Cisco TACACS+ specification, and *sep* is = for mandatory attributes and is * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

For example, this AV pair activates Cisco's *multiple named ip address pools* feature during IP authorization (during PPP IPCP address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

Other vendors have unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, "Remote Authentication Dial-In User Service (RADIUS)."

Follow this procedure to configure the switch to recognize and use VSAs.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server vsa send** [**accounting** | **authentication**]	Enable the switch to recognize and use VSAs as defined by RADIUS IETF attribute 26. • (Optional) Use the **accounting** keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the **authentication** keyword to limit the set of recognized vendor-specific attributes to only authentication attributes. If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

For a complete list of RADIUS attributes or more information about vendor-specific attribute 26, see the *RADIUS Attributes Configuration Guide, Cisco IOS Release 15M&T*.

EXAMPLE

This example shows how to provide a user logging in from a switch with immediate access to privileged EXEC commands:

```
cisco-avpair= "shell:priv-lvl=15"
```

This example shows how to specify an authorized VLAN in the RADIUS server database:

```
cisco-avpair= "tunnel-type(#64)=VLAN(13)"
cisco-avpair= "tunnel-medium-type(#65)=802 media(6)"
cisco-avpair= "tunnel-private-group-ID(#81)=vlanid"
```

This example shows how to apply an input ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"
cisco-avpair= "mac:inacl#3=deny any any decnet-iv"
```

This example shows how to apply an output ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

Although an IETF draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the switch and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the switch. You specify the RADIUS host and secret text string by using the **radius-server** global configuration commands.

Follow this procedure to specify a vendor-proprietary RADIUS server host and a shared secret text string.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} **non-standard**	Specify the IP address or hostname of the remote RADIUS server host and identify that it is using a vendor-proprietary implementation of RADIUS.
Step 3	**radius-server key** *string*	Specify the shared secret text string used between the switch and the vendor-proprietary RADIUS server. The switch and the RADIUS server use this text string to encrypt passwords and exchange responses. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the vendor-proprietary RADIUS host, use the **no radius-server host** {*hostname* | *ip-address*} **non-standard** global configuration command. To disable the key, use the **no radius-server key** global configuration command.

EXAMPLE

This example shows how to specify a vendor-proprietary RADIUS host and to use a secret key of *rad124* between the switch and the server:

```
Switch(config)# radius-server host 172.20.30.15 nonstandard
Switch(config)# radius-server key rad124
```

Configuring CoA on the Switch

This procedure is required.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa server radius dynamic-author**	Configure the switch as an authentication, authorization, and accounting (AAA) server to facilitate interaction with an external policy server.
Step 4	**client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]	Enter dynamic authorization local server configuration mode, and specify a RADIUS client from which a device accepts CoA and disconnect requests.
Step 5	**server-key** [**0** | **7**] *string*	Configure the RADIUS key to be shared between a device and RADIUS clients.
Step 6	**port** *port-number*	Specify the port on which a device listens for RADIUS requests from configured RADIUS clients.
Step 7	**auth-type** {**any** | **all** | **session-key**}	Specify the type of authorization the switch uses for RADIUS clients. The client must match all the configured attributes for authorization.
Step 8	**ignore session-key**	(Optional) Configure the switch to ignore the session-key. For more information about the **ignore** command, see the *Cisco IOS Intelligent Services Gateway Command Reference* on Cisco.com.
Step 9	**ignore server-key**	(Optional) Configure the switch to ignore the server-key. For more information about the **ignore** command, see the *Cisco IOS Intelligent Services Gateway Command Reference* on Cisco.com.
Step 10	**authentication command bounce-port ignore**	(Optional) Configure the switch to ignore a CoA request to temporarily disable the port hosting a session. The purpose of temporarily disabling the port is to trigger a DHCP renegotiation from the host when a VLAN change occurs and there is no supplicant on the endpoint to detect the change.
Step 11	**authentication command disable-port ignore**	(Optional) Configure the switch to ignore a nonstandard command requesting that the port hosting a session be administratively shut down. Shutting down the port terminates the session. Use standard CLI or SNMP commands to reenable the port.

	Command	Purpose
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show running-config**	Verify your entries.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable the AAA server functionality on the switch, use the **no aaa server radius dynamic authorization** global configuration command.

EXAMPLE

```
Switch(config)# aaa new-model
Switch(config)# aaa server radius dynamic-author
Switch(config-locsvr-da-radius)# client 10.0.0.1
Switch(config-locsvr-da-radius)# server-key cisco123
Switch(config-locsvr-da-radius)# port 3799
Switch(config-locsvr-da-radius)# auth-type all
Switch(config-locsvr-da-radius)# ignore session-key
Switch(config-locsvr-da-radius)# ignore server-key
Switch(config-locsvr-da-radius)# exit
```

Monitoring and Troubleshooting CoA Functionality

Use the following commands to monitor and troubleshoot CoA on the switch:

- **debug radius**
- **debug aaa coa**
- **debug aaa pod**
- **debug aaa subsys**
- **debug cmdhd** [**detail** | **error** | **events**]
- **show aaa attributes protocol radius**

Configuring RADIUS Server Load Balancing

This feature allows access and authentication requests to be evenly distributed across all RADIUS servers in a server group. For more information, see the “RADIUS Server Load Balancing” chapter of the *RADIUS Configuration Guide, Cisco IOS Release 15M&T*.

Configuring Kerberos

The Kerberos security system authenticates requests for network resources by using a trusted third party.

For Kerberos configuration examples, see the “Configuring Kerberos” chapter of the *User Security Configuration Guide, Cisco IOS Release 15MT*.

For complete syntax and usage information for the commands used to configure Kerberos, see the *Cisco IOS Security Command Reference*.

Note In the Kerberos configuration examples and in the *Cisco IOS Security Command Reference,* the trusted third party can be a IE 2000U switch that supports Kerberos, that is configured as a network security server, and that can authenticate users by using the Kerberos protocol.

So that remote users can authenticate to network services, you must configure the hosts and the KDC in the Kerberos realm to communicate and mutually authenticate users and network services. To do this, you must identify them to each other. You add entries for the hosts to the Kerberos database on the KDC and add KEYTAB files generated by the KDC to all hosts in the Kerberos realm. You also create entries for the users in the KDC database.

When you add or create entries for the hosts and users, follow these guidelines:

- The Kerberos principal name *must* be in all lowercase characters.
- The Kerberos instance name *must* be in all lowercase characters.
- The Kerberos realm name *must* be in all uppercase characters.

Note A Kerberos server can be a IE 2000U switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

To set up a Kerberos-authenticated server-client system, follow these steps:

1. Configure the KDC by using Kerberos commands.
2. Configure the switch to use the Kerberos protocol.

For instructions, see the "How to Configure Kerberos" section in the *User Security Configuration Guide, Cisco IOS Release 15MT*.

Configuring the Switch for Local Authentication and Authorization

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The switch then handles authentication and authorization. No accounting is available in this configuration.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication login default local**	Set the login authentication to use the local username database. The **default** keyword applies the local user database authentication to all ports.
Step 4	**aaa authorization exec local**	Configure user AAA authorization, check the local database, and allow the user to run an EXEC shell.

	Command	Purpose
Step 5	**aaa authorization network local**	Configure user AAA authorization for all network-related service requests.
Step 6	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	Enter the local database, and establish a username-based authentication system. Repeat this command for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. • For *password*, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config**	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

EXAMPLE

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication login default local
Switch(config)# aaa authorization exec local
Switch(config)# aaa authorization network local
Switch(config)# username user2 privilege 2 password 0 cisco
Switch(config)# end
```

Configuring SSH

This section includes the following topics:

- Setting Up the Switch to Run SSH, page 1-51 (required)
- Configuring the SSH Server, page 1-52 (required only if you are configuring the switch as an SSH server)
- Using SSH Keyboard Interactive Authentication, page 1-53

Setting Up the Switch to Run SSH

Follow these steps to set up your switch to run SSH:

1. Download the cryptographic software image from Cisco.com. This step is required. For more information, see the release notes for this release.
2. Configure a hostname and IP domain name for the switch. Follow this procedure only if you are configuring the switch as an SSH server.
3. Generate an RSA key pair for the switch, which automatically enables SSH. Follow this procedure only if you are configuring the switch as an SSH server.
4. Configure user authentication for local or remote access. This step is required. For more information, see the "Configuring the Switch for Local Authentication and Authorization" section on page 1-49.

Follow this procedure to configure a hostname and an IP domain name and to generate an RSA key pair. This procedure is required if you are configuring the switch as an SSH server.

BEFORE YOU BEGIN

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command. For more information, see the "Setting Up the Switch to Run SSH" section on page 1-51.
- When generating the RSA key pair, the message `No host name specified` might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message `No domain specified` might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.
- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**hostname** *hostname*	Configure a hostname for your switch.
Step 3	**ip domain-name** *domain_name*	Configure a host domain for your switch.
Step 4	**crypto key generate rsa**	Enable the SSH server for local and remote authentication on the switch and generate an RSA key pair. We recommend a minimum modulus size of 1024 bits. When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show ip ssh** or **show ssh**	Show the version and configuration information for your SSH server. Show the status of the SSH server on the switch.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the RSA key pair, use the **crypto key zeroize rsa** global configuration command. After the RSA key pair is deleted, the SSH server is automatically disabled.

EXAMPLE

```
Switch(config)# crypto key generate rsa
Switch(config)# end
```

Configuring the SSH Server

BEFORE YOU BEGIN

Complete the "Setting Up the Switch to Run SSH" procedure on page 1-51.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip ssh version** [**1** | **2**]	(Optional) Configure the switch to run SSH Version 1 or SSH Version 2. • **1**—Configure the switch to run SSH Version 1. • **2**—Configure the switch to run SSH Version 2. If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client supports SSHv1 and SSHv2, the SSH server selects SSHv2.

	Command	Purpose
Step 3	**ip ssh** {**timeout** *seconds* | **authentication-retries** *number*}	Configure the SSH control parameters: • Specify the timeout value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the switch uses the default timeout values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session timeout value returns to the default of 10 minutes. • Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5. Repeat this step when configuring both parameters.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip ssh** or **show ssh**	Show the version and configuration information for your SSH server. Show the status of the SSH server connections on the switch.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default SSH control parameters, use the **no ip ssh** {**timeout** | **authentication-retries**} global configuration command.

EXAMPLE

```
Switch(config)# ip ssh timeout 30
Switch(config)# end
```

Using SSH Keyboard Interactive Authentication

The SSH Keyboard Interactive Authentication feature, also known as Generic Message Authentication for SSH, is a method that can be used to implement different types of authentication mechanisms. Basically, any currently supported authentication method that requires only user input can be performed with this feature. The feature is automatically deployed.

Supported methods:

- Password
- SecurID and hardware tokens printing a number or a string in response to a challenge sent by the server
- Pluggable Authentication Module (PAM)
- S/KEY (and other One-Time-Pads)

For SSH keyboard interactive authentication to work, ensure that the **Apply password change rule** checkbox is checked on the Authentication Server Group Setup page on the RADIUS or TACACS server. The keyboard interactive authentication method works only with SSH V2 and the blank password mechanism is supported only with TACACS authentication.

Configuring Secure HTTP Servers and Clients

Secure Socket Layer (SSL) version 3.0 support for the HTTP 1.1 server and client provides server authentication, encryption, and message integrity, as well as HTTP client authentication, to allow secure HTTP communications.

This section includes the following topics:

- Configuring a CA Trustpoint, page 1-54
- Configuring the Secure HTTP Server, page 1-55
- Configuring the Secure HTTP Client, page 1-57

Configuring a CA Trustpoint

For secure HTTP connections, we recommend that you configure an official CA trustpoint. A CA trustpoint is more secure than a self-signed certificate.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**hostname** *hostname*	Specify the hostname of the switch (required only if you have not previously configured a hostname). The hostname is required for security keys and certificates.
Step 3	**ip domain-name** *domain-name*	Specify the IP domain name of the switch (required only if you have not previously configured an IP domain name). The domain name is required for security keys and certificates.
Step 4	**crypto key generate rsa**	(Optional) Generate an RSA key pair. RSA key pairs are required before you can obtain a certificate for the switch. RSA key pairs are generated automatically. You can use this command to regenerate the keys, if needed.
Step 5	**crypto ca trustpoint** *name*	Specify a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.
Step 6	**enrollment url** *url*	Specify the URL to which the switch should send certificate requests.
Step 7	**enrollment http-proxy** *host-name port-number*	(Optional) Configure the switch to obtain certificates from the CA through an HTTP proxy server.

	Command	Purpose
Step 8	**crl query** *url*	Configure the switch to request a certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.
Step 9	**primary**	(Optional) Specify that the trustpoint is the primary (default) trustpoint for CA requests.
Step 10	**exit**	Exit CA trustpoint configuration mode, and return to global configuration mode.
Step 11	**crypto ca authentication** *name*	Authenticate the CA by getting the public key of the CA. Use the same name used in Step 5.
Step 12	**crypto ca enroll** *name*	Obtain the certificate from the specified CA trustpoint. This command requests a signed certificate for each RSA key pair.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show crypto ca trustpoints**	Verify the configuration.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no crypto ca trustpoint** *name* global configuration command to delete all identity information and certificates associated with the CA.

EXAMPLE

```
Switch(config)# crypto key generate rsa
Switch(config)# crypto ca trustpoint your_trustpoint
Switch(ca-trustpoint)# enrollment url http://your_server:80
Switch(ca-trustpoint)# enrollment http-proxy your_host 49
Switch(ca-trustpoint)# crl query ldap://your_host:49
Switch(ca-trustpoint)# primary your_trustpoint
Switch(ca-trustpoint)# exit
Switch(config)# crypto ca authentication your_trustpoint
Switch(config)# crypto ca enroll your_trustpoint
Switch(config)# end
```

Configuring the Secure HTTP Server

If you are using a certificate authority for certification, you should use the previous procedure to configure the CA trustpoint on the switch before enabling the HTTP server. If you have not configured a CA trustpoint, a self-signed certificate is generated the first time that you enable the secure HTTP server. After you have configured the server, you can configure options (path, access list to apply, maximum number of connections, or timeout policy) that apply to both standard and secure HTTP servers.

BEFORE YOU BEGIN

Complete the "Configuring a CA Trustpoint" procedure on page 1-54.

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose

Step 1	**show ip http server status**	(Optional) Display the status of the HTTP server to determine if the secure HTTP server feature is supported in the software. You should see one of these lines in the output:
		`HTTP secure server capability: Present`
		`or`
		`HTTP secure server capability: Not present`

Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**ip http secure-server**	Enable the HTTPS server if it has been disabled. The HTTPS server is enabled by default.
Step 4	**ip http secure-port** *port-number*	(Optional) Specify the port number to be used for the HTTPS server. The default port number is 443. Valid options are 443 or any number in the range 1025 to 65535.
Step 5	**ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}	(Optional) Specify the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 6	**ip http secure-client-auth**	(Optional) Configure the HTTP server to request an X.509v3 certificate from the client for authentication during the connection process. The default is for the client to request a certificate from the server, but the server does not attempt to authenticate the client.
Step 7	**ip http secure-trustpoint** *name*	Specify the CA trustpoint to use to get an X.509v3 security certificate and to authenticate the client certificate connection. **Note** Use of this command assumes you have already configured a CA trustpoint according to the previous procedure.
Step 8	**ip http path** *path-name*	(Optional) Set a base HTTP path for HTML files. The path specifies the location of the HTTP server files on the local system (usually located in system flash memory).
Step 9	**ip http access-class** *access-list-number*	(Optional) Specify an access list to use to allow access to the HTTP server.
Step 10	**ip http max-connections** *value*	(Optional) Set the maximum number of concurrent connections that are allowed to the HTTP server. The range is 1 to 16; the default value is 5.

	Command	Purpose
Step 11	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*	(Optional) Specify how long a connection to the HTTP server can remain open under the defined circumstances: • **idle**—the maximum time period when no data is received or response data cannot be sent. The range is 1 to 600 seconds. The default is 180 seconds (3 minutes). • **life**—the maximum time period from the time that the connection is established. The range is 1 to 86400 seconds (24 hours). The default is 180 seconds. • **requests**—the maximum number of requests processed on a persistent connection. The maximum value is 86400. The default is 1.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show ip http server secure status**	Display the status of the HTTP secure server to verify the configuration.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip http server** global configuration command to disable the standard HTTP server. Use the **no ip http secure-server** global configuration command to disable the secure HTTP server. Use the **no ip http secure-port** and the **no ip http secure-ciphersuite** global configuration commands to return to the default settings. Use the **no ip http secure-client-auth** global configuration command to remove the requirement for client authentication.

To verify the secure HTTP connection by using a web browser, enter https://*URL*, where the URL is the IP address or hostname of the server switch. If you configure a port other than the default port, you must also specify the port number after the URL. For example:

https://209.165.129:1026
or

https://host.domain.com:1026

EXAMPLE

```
Switch# show ip http server status
Switch# configure terminal
Switch(config)# ip http secure-server
Switch(config)# ip http secure-port 443
Switch(config)# ip http secure-ciphersuite rc4-128-md5
Switch(config)# ip http secure-client-auth
Switch(config)# ip http secure-trustpoint your_trustpoint
Switch(config)# ip http path /your_server:80
Switch(config)# ip http access-class 2
Switch(config)# ip http timeout-policy idle 120 life 240 requests 1
Switch(config)# end
```

Configuring the Secure HTTP Client

The standard HTTP client and secure HTTP client are always enabled. A certificate authority is required for secure HTTP client certification.

BEFORE YOU BEGIN

This procedure assumes that you have previously configured a CA trustpoint on the switch (see "Configuring a CA Trustpoint" procedure on page 1-54). If a CA trustpoint is not configured and the remote HTTPS server requires client authentication, connections to the secure HTTP client fail.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip http client secure-trustpoint** *name*	(Optional) Specify the CA trustpoint to be used if the remote HTTP server requests client authentication. Using this command assumes that you have already configured a CA trustpoint as described in the "Configuring a CA Trustpoint" procedure on page 1-54. The command is optional if client authentication is not needed or if a primary trustpoint has been configured.
Step 3	**ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}	(Optional) Specify the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip http client secure status**	Display the status of the HTTP secure server to verify the configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip http client secure-trustpoint** *name* to remove a client trustpoint configuration. Use the **no ip http client secure-ciphersuite** to remove a previously configured CipherSuite specification for the client.

EXAMPLE

```
Switch (config)# ip http client secure-trustpoint your_trustpoint
Switch(config)# ip http client secure-ciphersuite rc4-128-md5
Switch(config)# end
```

Verifying Configuration

Command	Purpose
show tacacs	Display TACACS+ server statistics.
show running-config	Display RADIUS configuration.
show ip ssh	Shows the version and configuration information for the SSH server.

Command	Purpose
show ssh	Shows the status of the SSH server.
show ip http client secure status	Shows the HTTP secure client configuration.
show ip http server secure status	Shows the HTTP secure server configuration.
show running-config	Shows the generated self-signed certificate for secure HTTP connections.

Related Documents

- Cisco IOS Master Command List, All Releases
- User Security Configuration Guide, Cisco IOS Release 15M&T
- Cisco IOS Security Command Reference
- RADIUS Attributes Configuration Guide, Cisco IOS Release 15M&T
- RADIUS Configuration Guide, Cisco IOS Release 15M&T
- Secure Shell Configuration Guide, Cisco IOS Release 15M&T
- Public Key Infrastructure Configuration Guide, Cisco IOS Release 15MT

Configuring IEEE 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication on the Cisco Industrial Ethernet 2000U Series Switches (IE 2000U).

As LANs extend to hotels, airports, and corporate lobbies and create insecure environments, 802.1x prevents unauthorized devices (clients) from gaining access to the network.

For complete syntax and usage information for the commands used in this chapter, see the documents listed in the "Related Documents" section on page 2-60.

This chapter consists of these sections:

- Information About 802.1x Port-Based Authentication, page 2-1
- Prerequisites, page 2-24
- Guidelines and Limitations, page 2-24
- Default Settings, page 2-26
- Configuring 802.1x Authentication, page 2-27
- Verifying Configuration, page 2-60
- Related Documents, page 2-60

Information About 802.1x Port-Based Authentication

The 802.1x standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN.

Until the client is authenticated, 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

Note CDP and STP are supported by default on network node interfaces (NNIs). You can enable CDP and STP on enhanced network interfaces (ENIs). User network nodes (UNIs) do not support CDP or STP.

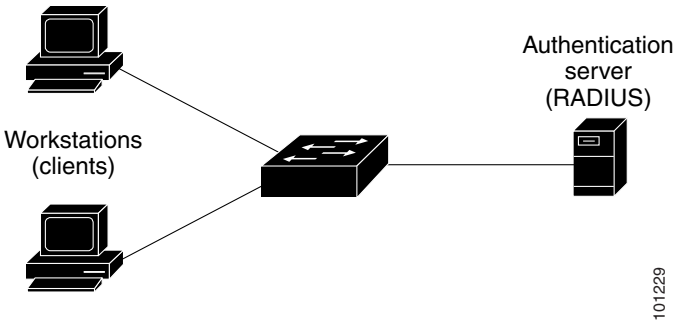
This section includes the following topics:

- Device Roles, page 2-2
- Authentication Process, page 2-3
- Authentication Initiation and Message Exchange, page 2-5
- Authentication Manager, page 2-7
- Ports in Authorized and Unauthorized States, page 2-8
- 802.1x Host Mode, page 2-9
- 802.1x Multiple-Authentication Mode, page 2-10
- MAC Move, page 2-10
- 802.1x Accounting, page 2-10
- 802.1x Accounting Attribute-Value Pairs, page 2-11
- 802.1x Readiness Check, page 2-12
- 802.1x Authentication with VLAN Assignment, page 2-12
- Using 802.1x Authentication with Per-User ACLs, page 2-13
- 802.1x Authentication with Downloadable ACLs and Redirect URLs, page 2-14
- VLAN ID-based MAC Authentication, page 2-15
- 802.1x Authentication with Guest VLAN, page 2-16
- 802.1x Authentication with Restricted VLAN, page 2-17
- 802.1x Authentication with Inaccessible Authentication Bypass, page 2-17
- 802.1x Authentication with Port Security, page 2-19
- 802.1x Authentication with Wake-on-LAN, page 2-20
- 802.1x Authentication with MAC Authentication Bypass, page 2-20
- 802.1x User Distribution, page 2-21
- Network Admission Control Layer 2 IEEE 802.1x Validation, page 2-22
- Flexible Authentication Ordering, page 2-22
- Open1x Authentication, page 2-22
- 802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT), page 2-23
- Common Session ID, page 2-24

Device Roles

With 802.1x port-based authentication, the devices in the network have specific roles as shown in Figure 2-1.

Figure 2-1 802.1x Device Roles



- *Client*—The device (workstation) that requests access to the LAN and switch services and responds to requests from the switch. The workstation must be running 802.1x-compliant client software such as that offered in the Microsoft Windows XP operating system. (The client is the *supplicant* in the 802.1x specification.)
- *Authentication server*—Performs the actual authentication of the client. The authentication server validates the identity of the client and notifies the switch whether or not the client is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the client.

  In this release, the RADIUS security system with Extensible Authentication Protocol (EAP) extensions is the only supported authentication server. It is available in Cisco Secure Access Control Server Version 3.0 or later. RADIUS operates in a client/server model in which secure authentication information is exchanged between the RADIUS server and one or more RADIUS clients.
- *Switch* (edge switch or wireless access point)—Controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client. The switch includes the RADIUS client, which is responsible for encapsulating and decapsulating the EAP frames and interacting with the authentication server.

  When the switch receives EAPOL frames and relays them to the authentication server, the Ethernet header is stripped, and the remaining EAP frame is re-encapsulated in the RADIUS format. The EAP frames are not modified during encapsulation, and the authentication server must support EAP within the native frame format. When the switch receives frames from the authentication server, the server's frame header is removed, leaving the EAP frame, which is then encapsulated for Ethernet and sent to the client.

Authentication Process

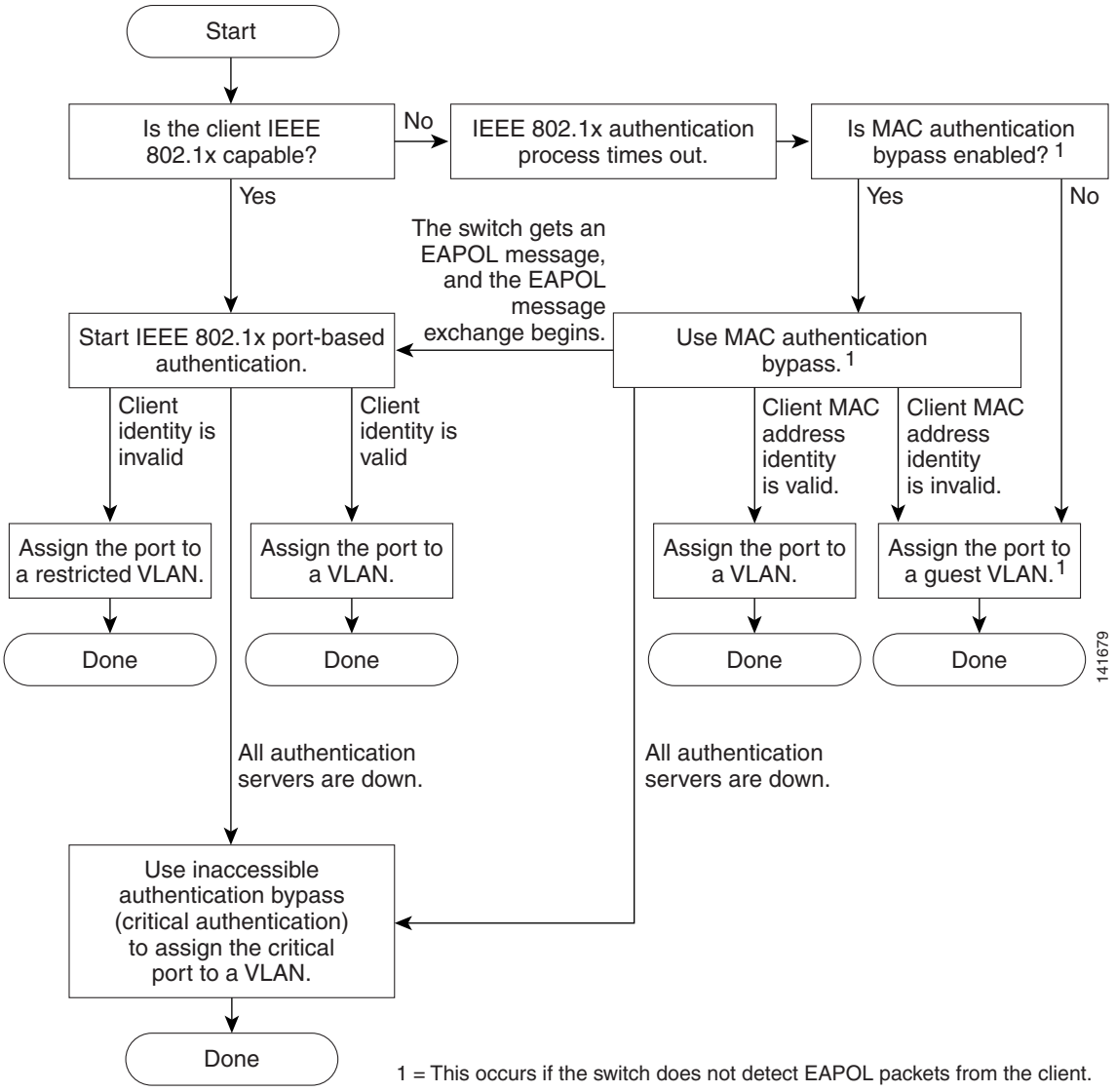
When 802.1x port-based authentication is enabled and the client supports 802.1x-compliant client software, these events occur:

- If the client identity is valid and the 802.1x authentication succeeds, the switch grants the client access to the network.
- If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can use the client MAC address for authorization. If the client MAC address is valid and the authorization succeeds, the switch grants the client access to the network. If the client MAC address is invalid and the authorization fails, the switch assigns the client to a guest VLAN that provides limited services if a guest VLAN is configured.

- If the switch gets an invalid identity from an 802.1x-capable client and a restricted VLAN is specified, the switch can assign the client to a restricted VLAN that provides limited services.
- If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network by putting the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN.

Note Inaccessible authentication bypass is also referred to as critical authentication or the AAA fail policy. See the "Configuring an AAA Fail Policy" section on page 3-16.

Figure 2-2 Authentication Flowchart



The switch re-authenticates a client when one of these situations occurs:

- Periodic re-authentication is enabled, and the re-authentication timer expires.

  You can configure the re-authentication timer to use a switch-specific value or to be based on values from the RADIUS server.

After 802.1x authentication using a RADIUS server is configured, the switch uses timers based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]).

The Session-Timeout RADIUS attribute (Attribute[27]) specifies the time after which re-authentication occurs.

The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during re-authentication. The actions are *Initialize* and *ReAuthenticate*. When the *Initialize* action is set (the attribute value is *DEFAULT*), the 802.1x session ends, and connectivity is lost during re-authentication. When the *ReAuthenticate* action is set (the attribute value is RADIUS-Request), the session is not affected during re-authentication.

- You manually re-authenticate the client by entering the **authentication periodic** interface configuration command.

Authentication Initiation and Message Exchange

The switch or the client can initiate authentication. If you enable authentication on a port by using the **authentication port-control auto** interface configuration command, the switch initiates authentication when the link state changes from down to up or periodically as long as the port remains up and unauthenticated. The switch sends an EAP-request/identity frame to the client to request its identity. Upon receipt of the frame, the client responds with an EAP-response/identity frame.

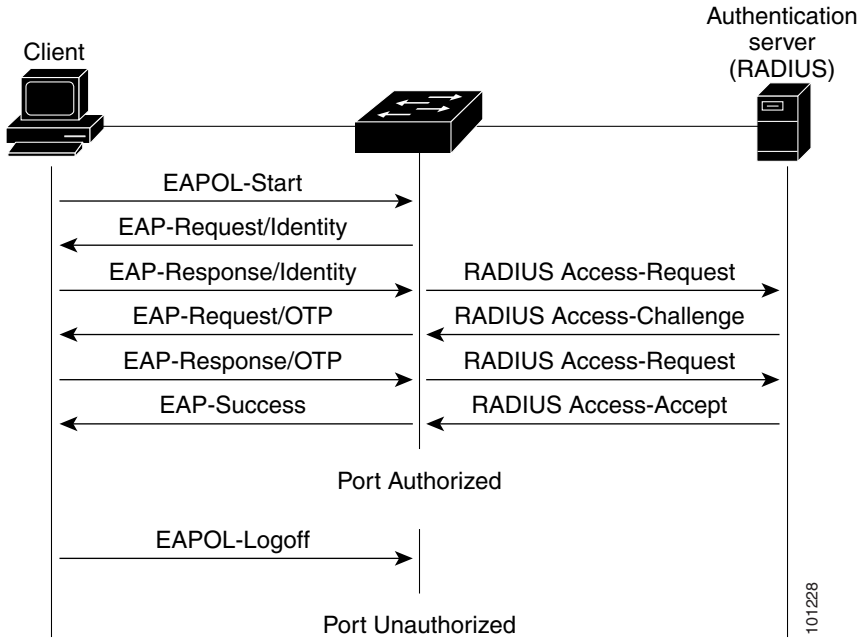
However, if during bootup, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.

Note If 802.1x is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized state. A port in the authorized state effectively means that the client has been successfully authenticated. For more information, see the "Ports in Authorized and Unauthorized States" section on page 2-8.

When the client supplies its identity, the switch begins its role as the intermediary, passing EAP frames between the client and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. For more information, see the "Ports in Authorized and Unauthorized States" section on page 2-8.

The specific exchange of EAP frames depends on the authentication method being used. Figure 2-3 shows a message exchange initiated by the client when the client uses the One-Time-Password (OTP) authentication method with a RADIUS server.

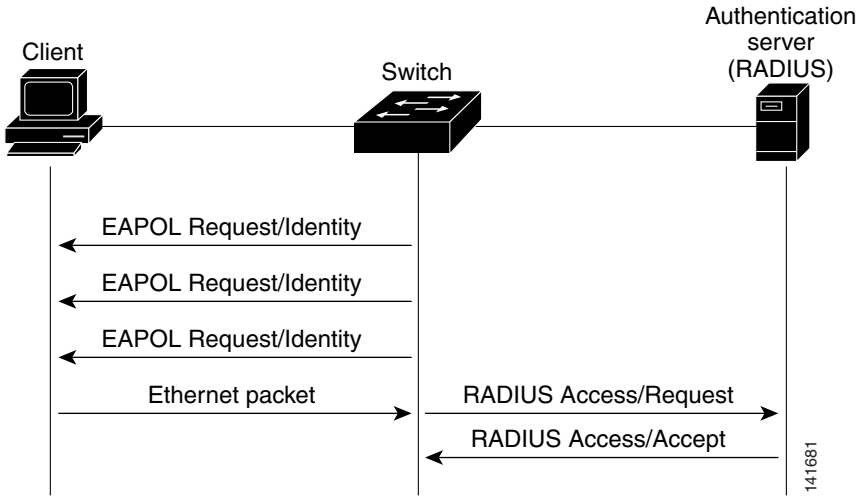
Figure 2-3 Message Exchange



If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the client. The switch uses the MAC address of the client as its identity and includes this information in the RADIUS-access/request frame that is sent to the RADIUS server. After the server sends the switch the RADIUS-access/accept frame (authorization is successful), the port becomes authorized. If authorization fails and a guest VLAN is specified, the switch assigns the port to the guest VLAN. If the switch detects an EAPOL packet while waiting for an Ethernet packet, the switch stops the MAC authentication bypass process and stops 802.1x authentication.

Figure 2-4 shows the message exchange during MAC authentication bypass.

Figure 2-4 Message Exchange During MAC Authentication Bypass



Authentication Manager

This section contains the following topics:

- Port-Based Authentication Methods, page 2-7
- Per-User ACLs and Filter-IDs, page 2-8
- Authentication Manager CLI Commands, page 2-8

Note Catalyst switches that are running Cisco IOS Release 12.2(50)SE or later in a network support the same authorization methods as the IE 2000U switch.

Note The IE 2000U switch does not support multidomain authentication (MDA) or 802.1x authentication with voice VLAN ports.

Port-Based Authentication Methods

Table 2-1 802.1x Features

Authentication method	Mode		
	Single host	Multiple host	Multiple Authentication[1]
802.1x	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment	Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
MAC authentication bypass	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment	Per-user ACL Filter-Id attribute Downloadable[2]
Standalone web authentication[2]	Proxy ACL, Filter-Id attribute, downloadable ACL		
NAC Layer 2 IP validation	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL
Web authentication as fallback method[2]	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL

1. Also referred to as *multiauth*.
2. For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-IDs

You can only set **any** as the source in the ACL. **For any ACL configured for multiple-host mode, the source portion of statement must be *any*. (For example, permit icmp *any* host 10.10.1.1.)**

Authentication Manager CLI Commands

The authentication-manager interface-configuration commands control all the authentication methods, such as 802.1x, MAC authentication bypass, and web authentication. The authentication manager commands determine the priority and order of authentication methods applied to a connected host.

The authentication manager commands control generic authentication features, such as host-mode, violation mode, and the authentication timer. Generic authentication commands include the **authentication host-mode**, **authentication violation**, and **authentication timer** interface configuration commands.

802.1x-specific commands at the global configuration mode begin with **aaa authentication dot1x**. For example, the **authentication port-control auto** interface configuration command enables authentication on an interface. However, the **dot1x system-authentication control** global configuration command only globally enables or disables 802.1x authentication.

Note If 802.1x authentication is globally disabled, other authentication methods are still enabled on that port, such as web authentication.

The authentication manager commands provide the same functionality as earlier 802.1x commands.

For more information, see the *Cisco IOS Security Command Reference*.

Ports in Authorized and Unauthorized States

Depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port disallows all incoming and outgoing traffic except for 802.1x, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally.

If a client that does not support 802.1x connects to an unauthorized 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an 802.1x-enabled client connects to a port that is not running the 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **authentication port-control** interface configuration command and these keywords:

- **force-authorized**—Disables 802.1x authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without 802.1x-based authentication of the client. This is the default setting.
- **force-unauthorized**—Causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the port.

- **auto**—Enables 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client MAC address.

If the client is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated client are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the authentication server cannot be reached, the switch can resend the request. If no response is received from the server after the specified number of attempts, authentication fails, and network access is not granted.

When a client logs off, it sends an EAPOL-logoff message, causing the switch port to change to the unauthorized state.

If the link state of a port changes from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

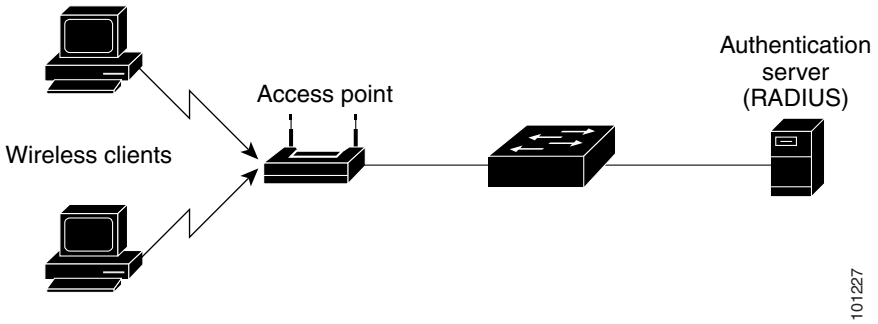
802.1x Host Mode

You can configure an 802.1x port for single-host or for multiple-hosts mode. In single-host mode (see Figure 2-1 on page 2-3), only one client can be connected to the 802.1x-enabled switch port. The switch detects the client by sending an EAPOL frame when the port link state changes to the up state. If a client leaves or is replaced with another client, the switch changes the port link state to down, and the port returns to the unauthorized state.

In multiple-hosts mode, you can attach multiple hosts to a single 802.1x-enabled port. Figure 2-5 on page 2-9 shows 802.1x port-based authentication in a wireless LAN. In this mode, only one of the attached clients must be authorized for all clients to be granted network access. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), the switch denies network access to all of the attached clients. In this topology, the wireless access point is responsible for authenticating the clients attached to it, and it also acts as a client to the switch.

With the multiple-hosts mode enabled, you can use 802.1x to authenticate the port and port security to manage network access for all MAC addresses, including that of the client.

Figure 2-5 Multiple Host Mode Example



802.1x Multiple-Authentication Mode

Multiple-authentication (multiauth) mode allows multiple authenticated clients on the data VLAN. When a hub or access point is connected to an 802.1x-enabled port, multiple-authentication mode provides enhanced security over multiple-hosts mode by requiring authentication of each connected client. For non-802.1x devices, you can use MAC authentication bypass or web authentication as the fallback method for individual host authentications to authenticate different hosts through different methods on a single port.

Note When a port is in multiple-authentication mode, the RADIUS-server-supplied VLAN assignment, guest VLAN, and the authentication-failed VLAN features do not activate.

For more information about critical authentication mode and the critical VLAN, see the "802.1x Authentication with Inaccessible Authentication Bypass" section on page 2-17.

For more information see the "Configuring 802.1x Accounting" section on page 2-41.

MAC Move

When a MAC address is authenticated on one switch port, that address is not allowed on another 802.1x port of the switch. If the switch detects that same MAC address on another 802.1x port, the address is not allowed.

There are situations where a MAC address might need to move from one port to another on the same switch. For example, when there is another device (for example a hub or an IP phone) between an authenticated host and a switch port, you might want to disconnect the host from the device and connect it directly to another port on the same switch.

You can globally enable MAC move so the device is re-authenticated on the new port. When a host moves to a second port, the session on the first port is deleted, and the host is re-authenticated on the new port.

MAC move is supported for all host modes. (The authenticated host can move to any port on the switch, no matter which host mode is enabled on the port.)

Note MAC move is not supported on port-security enabled 802.1x ports. If MAC move is globally configured on the switch and a port security-enabled host moves to an 802.1x-enabled port, a violation error occurs.

For more information see the "Enabling MAC Move" section on page 2-40.

802.1x Accounting

The 802.1x standard defines how users are authorized and authenticated for network access but does not keep track of network usage. 802.1x accounting is disabled by default. You can enable 802.1x accounting to monitor this activity on 802.1x-enabled ports:

- User successfully authenticates.
- User logs off.
- Link-down occurs.
- re-authentication successfully occurs.

- Re-authentication fails.

The switch does not log 802.1x accounting information. Instead, it sends this information to the RADIUS server, which must be configured to log accounting messages.

802.1x Accounting Attribute-Value Pairs

The information sent to the RADIUS server is represented in the form of Attribute-Value (AV) pairs. These AV pairs provide data for different applications. (For example, a billing application might require information that is contained within the Acct-Input-Octets or the Acct-Output-Octets of a packet.)

AV pairs are automatically sent by a switch that is configured for 802.1x accounting. The switch sends these types of RADIUS accounting packets:

- START–Sent when a new user session starts
- INTERIM–Sent during an existing session for updates
- STOP–Sent when a session terminates

Table 2-2 lists the AV pairs that might be sent by the switch:

Table 2-2 Accounting AV Pairs

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[1]	User-Name	Always	Always	Always
Attribute[4]	NAS-IP-Address	Always	Always	Always
Attribute[5]	NAS-Port	Always	Always	Always
Attribute[8]	Framed-IP-Address	Never	Sometimes[1]	Sometimes[1]
Attribute[25]	Class	Always	Always	Always
Attribute[30]	Called-Station-ID	Always	Always	Always
Attribute[31]	Calling-Station-ID	Always	Always	Always
Attribute[40]	Acct-Status-Type	Always	Always	Always
Attribute[41]	Acct-Delay-Time	Always	Always	Always
Attribute[42]	Acct-Input-Octets	Never	Always	Always
Attribute[43]	Acct-Output-Octets	Never	Always	Always
Attribute[44]	Acct-Session-ID	Always	Always	Always
Attribute[45]	Acct-Authentic	Always	Always	Always
Attribute[46]	Acct-Session-Time	Never	Always	Always
Attribute[49]	Acct-Terminate-Cause	Never	Never	Always
Attribute[61]	NAS-Port-Type	Always	Always	Always

1. The Framed-IP-Address AV pair is sent only if a valid Dynamic Host Control Protocol (DHCP) binding exists for the host in the DHCP snooping bindings table.

You can see the AV pairs that are being sent by the switch by enabling the **debug radius accounting** or **debug aaa accounting** privileged EXEC commands. For more information about these commands, see the Cisco IOS Debug Command Reference.

For more information about AV pairs, see RFC 3580, "IEEE 802.1x Remote Authentication Dial In User Service (RADIUS) Usage Guidelines."

802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable. You use an alternate authentication such as MAC authentication bypass or web authentication for devices that do not support 802.1x functionality.

This feature works only if the supplicant on the client supports a query with the NOTIFY EAP notification packet. The client must respond within the 802.1x timeout value.

For information on configuring the switch for the 802.1x readiness check, see the "Configuring 802.1x Readiness Check" section on page 2-28.

802.1x Authentication with VLAN Assignment

The switch supports 802.1x authentication with VLAN assignment. After 802.1x authentication of a port, the RADIUS server sends the VLAN assignment to configure the switch port. The RADIUS server database maintains the username-to-VLAN mappings, assigning the VLAN based on the username of the client connected to the switch port. You can use this feature to limit network access for certain users.

When configured on the switch and the RADIUS server, 802.1x with VLAN assignment has these characteristics:

- If no VLAN is supplied by the RADIUS server or if 802.1x authorization is disabled, the port is configured in its access VLAN after successful authentication. Recall that an access VLAN is a VLAN assigned to an access port. All packets sent from or received on this port belong to this VLAN.
- If 802.1x authorization is enabled but the VLAN information from the RADIUS server is not valid, the port returns to the unauthorized state and remains in the configured access VLAN. This prevents ports from appearing unexpectedly in an inappropriate VLAN because of a configuration error.

  Configuration errors could include specifying a VLAN for a routed port, a malformed VLAN ID, a nonexistent or internal (routed port) VLAN ID, an RSPAN VLAN, or a shut down or suspended VLAN. In the case of a multidomain host port, configuration errors can also be due to an attempted assignment of a data VLAN that matches the configured or assigned voice VLAN ID (or the reverse). Voice VLANs are not supported.
- If 802.1x authorization is enabled and all information from the RADIUS server is valid, the authorized device is placed in the specified VLAN.
- If multiple-hosts mode is enabled on an 802.1x port, all hosts are placed in the same VLAN (specified by the RADIUS server) as the first authenticated host.
- Enabling port security does not impact the RADIUS server-assigned VLAN behavior.
- If 802.1x authentication is disabled on the port, it is returned to the configured access VLAN.

When the port is in the force authorized, the force unauthorized, the unauthorized, or the shutdown state, it is put into the configured access VLAN.

If an 802.1x port is authenticated and put in the RADIUS server-assigned VLAN, any change to the port access VLAN configuration does not take effect.

The 802.1x with VLAN assignment feature is not supported on trunk ports or with dynamic-access port assignment through a VLAN Membership Policy Server (VMPS).

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication. (The VLAN assignment feature is automatically enabled when you configure 802.1x authentication on an access port).
- Assign vendor-specific tunnel attributes on the RADIUS server. The RADIUS server must return these attributes to the switch:
  - [64] Tunnel-Type = VLAN
  - [65] Tunnel-Medium-Type = 802
  - [81] Tunnel-Private-Group-ID = VLAN name or VLAN ID

  Attribute[64] must contain the value *VLAN* (type 13). Attribute[65] must contain the value *802* (type 6). Attribute[81] specifies the *VLAN name* or *VLAN ID* assigned to the 802.1x-authenticated user.

For examples of tunnel attributes, see the "Configuring the Switch to Use Vendor-Specific RADIUS Attributes" section on page 1-44.

Using 802.1x Authentication with Per-User ACLs

You can enable per-user access control lists (ACLs) to provide different levels of network access and service to an 802.1x-authenticated user. When the RADIUS server authenticates a user connected to an 802.1x port, it retrieves the ACL attributes based on the user identity and sends them to the switch. The switch applies the attributes to the 802.1x port for the duration of the user session. The switch removes the per-user ACL configuration when the session ends, if authentication fails, or if a link-down condition occurs. The switch does not save RADIUS-specified ACLs in the running configuration. When the port is unauthorized, the switch removes the ACL from the port.

You can configure router ACLs and input port ACLs on the same switch. However, a port ACL takes precedence over a router ACL. If you apply an input port ACL to an interface that belongs to a VLAN, the port ACL takes precedence over an input router ACL applied to the VLAN interface. The port ACL filters received packets. The router ACL filters received routed packets from other ports. The router ACL also filters sent routed packets. To avoid configuration conflicts, you should carefully plan the user profiles stored on the RADIUS server.

RADIUS supports per-user attributes, including vendor-specific attributes. These vendor-specific attributes (VSAs) are in octet-string format and are passed to the switch during the authentication process. The VSAs used for per-user ACLs are `inacl#<n>` for the ingress direction and `outacl#<n>` for the egress direction. MAC ACLs are supported only in the ingress direction. The switch supports VSAs only in the ingress direction. It does not support port ACLs in the egress direction on Layer 2 ports.

Use only the extended ACL syntax style to define the per-user configuration stored on the RADIUS server. When the definitions are passed from the RADIUS server, they are created by using the extended naming convention. However, if you use the Filter-Id attribute, it can point to a standard ACL.

You can use the Filter-Id attribute to specify an inbound or outbound ACL that is already configured on the switch. The attribute contains the ACL number followed by *.in* for ingress filtering or *.out* for egress filtering. If the RADIUS server does not allow the *.in* or *.out* syntax, the access list is applied to the

outbound ACL by default. Because of limited support of Cisco IOS access lists on the switch, the Filter-Id attribute is supported only for IP ACLs numbered 1 to 199 and 1300 to 2699 (IP standard and IP extended ACLs).

Only one 802.1x-authenticated user is supported on a port. If multiple-hosts mode is enabled on the port, the per-user ACL attribute is disabled for the associated port.

The maximum size of the per-user ACL is 4000 ASCII characters but is limited by the maximum size of RADIUS-server per-user ACLs.

For examples of vendor-specific attributes, see the "Configuring the Switch to Use Vendor-Specific RADIUS Attributes" section on page 1-44. For more information about configuring ACLs, see Chapter 6, "Configuring Network Security with ACLs."

To configure per-user ACLs, perform these tasks:

- Enable AAA authentication.
- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication.
- Configure the user profile and VSAs on the RADIUS server.
- Configure the 802.1x port for single-host mode.

Note Per-user ACLs are supported only in single-host mode.

802.1x Authentication with Downloadable ACLs and Redirect URLs

You can download ACLs and redirect URLs from a RADIUS server to the switch during 802.1x authentication or MAC authentication bypass of the host. You can also download ACLs during web authentication.

Note A downloadable ACL is also referred to as a *dACL*.

If the host mode is in single-host or multiple-authentication mode, the switch modifies the source address of the ACL to be the host IP address.

You can apply the ACLs and redirect URLs to all the devices connected to the 802.1x-enabled port.

If no ACLs are downloaded during 802.1x authentication, the switch applies the static default ACL on the port to the host.

Note If a downloadable ACL or redirect URL is configured for a client on the authentication server, you must also configure a default port ACL on the connected client switch port.

Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL

The switch uses these *cisco-av-pair* VSAs:

- url-redirect is the HTTP to HTTPS URL.
- url-redirect-acl is the switch ACL name or number.

The switch uses the CiscoSecure-Defined-ACL AV pair to intercept an HTTP or HTTPS request from the endpoint device. The switch then forwards the client web browser to the specified redirect address. The url-redirect AV pair on the Cisco Secure ACS contains the URL to which the web browser is redirected. The url-redirect-acl AV pair contains the name or number of an ACL that specifies the HTTP or HTTPS traffic to redirect. Traffic that matches a permit ACE in the ACL is redirected.

Note Define the URL redirect ACL and the default port ACL on the switch.

If a redirect URL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs

You can set the CiscoSecure-Defined-ACL Attribute-Value (AV) pair on the Cisco Secure ACS with the RADIUS cisco-av-pair vendor-specific attributes (VSAs). This pair specifies the names of the downloadable ACLs on the Cisco Secure ACS with the #ACL#-IP-name-number attribute, where:

- The *name* is the ACL name.
- The *number* is the version number (for example, 3f783768).

If a downloadable ACL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

If the default ACL is configured on the switch and the Cisco Secure ACS sends a host-access-policy to the switch, it applies the policy to traffic from the host connected to a switch port. If the policy does not apply, the switch applies the default ACL. If the Cisco Secure ACS sends the switch a downloadable ACL, this ACL takes precedence over the default ACL that is configured on the switch port. However, if the switch receives a host access policy from the Cisco Secure ACS but the default ACL is not configured, the authorization failure is declared.

For configuration details, see the "Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs" section on page 2-54.

VLAN ID-based MAC Authentication

You can use VLAN ID-based MAC authentication if you wish to authenticate hosts based on a static VLAN ID instead of a downloadable VLAN. When you have a static VLAN policy configured on your switch, VLAN information is sent to an IAS (Microsoft) RADIUS server along with the MAC address of each host for authentication. The VLAN ID configured on the connected port is used for MAC authentication. By using VLAN ID-based MAC authentication with an IAS server, you can have a fixed number of VLANs in the network.

The feature also limits the number of VLANs monitored and handled by STP. The network can be managed as a fixed VLAN.

Note This feature is not supported on the Cisco ACS server. (The ACS server ignores the sent VLAN-IDs for new hosts and only authenticates based on the MAC address.)

For configuration information, see the "Configuring VLAN ID-based MAC Authentication" section on page 2-57. Additional configuration is similar to MAC authentication bypass, as described in the "Configuring MAC Authentication Bypass" section on page 2-48.

802.1x Authentication with Guest VLAN

You can configure a guest VLAN for each 802.1x port on the switch to provide limited services to clients, such as downloading the 802.1x client. These clients might be upgrading their system for 802.1x authentication, and some hosts, such as systems before Windows XP, might not be 802.1x-capable.

When you enable a guest VLAN on an 802.1x port, the switch assigns clients to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.

The switch maintains the EAPOL packet history. If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an 802.1x-capable supplicant, and the interface does not change to the guest VLAN state. EAPOL history is cleared if the interface link status goes down. If no EAPOL packet is detected on the interface, the interface changes to the guest VLAN state.

If devices send EAPOL packets to the switch during the lifetime of the link, the switch does not allow clients that fail authentication access to the guest VLAN.

If the switch is trying to authorize an 802.1x-capable voice device and the AAA server is unavailable, the authorization attempt fails, but the detection of the EAPOL packet is saved in the EAPOL history. When the AAA server becomes available, the switch authorizes the voice device. However, the switch does not allow other devices access to the guest VLAN. To prevent this situation, use one of these command sequences:

- Enter the **dot1x guest-vlan supplicant** global configuration command to allow access to the guest VLAN.
- Enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command to restart the port.

Note If an EAPOL packet is detected after the interface has changed to the guest VLAN, the interface reverts to an unauthorized state, and 802.1x authentication restarts.

Any number of 802.1x-incapable clients are allowed access when the switch port is moved to the guest VLAN. If an 802.1x-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the user-configured access VLAN, and authentication is restarted.

Guest VLANs are supported on 802.1x ports in single-host or multiple-hosts mode.

You can configure any active VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on trunk ports; it is supported only on access ports.

The switch supports *MAC authentication bypass*. When MAC authentication bypass is enabled on an 802.1x port, the switch can authorize clients based on the client MAC address when 802.1x authentication times out while waiting for an EAPOL message exchange. After detecting a client on an 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is specified. For more information, see the "802.1x Authentication with MAC Authentication Bypass" section on page 2-20.

For more information, see the "Configuring a Guest VLAN" section on page 2-42.

802.1x Authentication with Restricted VLAN

You can configure a restricted VLAN (also referred to as an *authentication failed VLAN*) for each 802.1x port on a switch to provide limited services to clients that cannot access the guest VLAN. These clients are 802.1x-compliant and cannot access another VLAN because they fail the authentication process. A restricted VLAN allows users without valid credentials in an authentication server (typically, visitors to an enterprise) to access a limited set of services. The administrator can control the services available to the restricted VLAN.

Note You can configure a VLAN to be both the guest VLAN and the restricted VLAN if you want to provide the same services to both types of users.

Without this feature, the client attempts and fails authentication indefinitely, and the switch port remains in the spanning-tree blocking state. With this feature, you can configure the switch port to be in the restricted VLAN after a specified number of authentication attempts (the default value is 3 attempts).

The authenticator counts the failed authentication attempts for the client. When this count exceeds the configured maximum number of authentication attempts, the port moves to the restricted VLAN. The failed attempt count increments when the RADIUS server replies with either an *EAP failure* or an empty response without an EAP packet. When the port moves into the restricted VLAN, the failed attempt counter resets.

Users who fail authentication remain in the restricted VLAN until the next re-authentication attempt. A port in the restricted VLAN tries to re-authenticate at configured intervals (the default is 60 seconds). If re-authentication fails, the port remains in the restricted VLAN. If re-authentication is successful, the port moves either to the configured VLAN or to a VLAN sent by the RADIUS server. You can disable re-authentication. If you do this, the only way to restart the authentication process is for the port to receive a *link dow*n or *EAP logoff* event. We recommend that you keep re-authentication enabled if a client might connect through a hub. When a client disconnects from the hub, the port might not receive the *link down* or *EAP logoff* event.

After a port moves to the restricted VLAN, a simulated EAP success message is sent to the client. This prevents clients from indefinitely attempting authentication. Some clients (for example, devices running Windows XP) cannot implement DHCP without EAP success.

Restricted VLANs are supported only on 802.1x ports in single-host mode and on Layer 2 ports.

You can configure any active VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on trunk ports; it is supported only on access ports.

This feature works with port security. As soon as the port is authorized, a MAC address is provided to port security. If port security does not permit the MAC address or if the maximum secure address count is reached, the port becomes unauthorized and error disabled.

Other port security features such as dynamic ARP Inspection, DHCP snooping, and IP source guard can be configured independently on a restricted VLAN.

For more information, see the "Configuring a Restricted VLAN" section on page 2-43.

802.1x Authentication with Inaccessible Authentication Bypass

Use the inaccessible authentication bypass feature, also referred to as critical authentication or the AAA fail policy, when the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated. You can configure the switch to connect those hosts to critical ports.

When a new host tries to connect to the critical port, that host is moved to a user-specified access VLAN, the critical VLAN. The administrator gives limited authentication to the hosts.

When the switch tries to authenticate a host connected to a critical port, the switch checks the status of the configured RADIUS server. If a server is available, the switch can authenticate the host. However, if all the RADIUS servers are unavailable, the switch grants network access to the host and puts the port in the critical-authentication state, which is a special case of the authentication state.

Support on Multiple-Authentication Ports

To support inaccessible bypass on multiple-authentication (multiauth) ports, you can use the **authentication event server dead action reinitialize vlan** *vlan-id.* When a new host tries to connect to the critical port, that port is reinitialized and all the connected hosts are moved to the user-specified access VLAN.

The **authentication event server dead action reinitialize vlan** *vlan-id* interface configuration command is supported for all host modes.

Authentication Results

The behavior of the inaccessible authentication bypass feature depends on the authorization state of the port:

- If the port is unauthorized when a host connected to a critical port tries to authenticate and all servers are unavailable, the switch puts the port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
- If the port is already authorized and re-authentication occurs, the switch puts the critical port in the critical-authentication state in the current VLAN, which might be the one previously assigned by the RADIUS server.
- If the RADIUS server becomes unavailable during an authentication exchange, the current exchange times out, and the switch puts the critical port in the critical-authentication state during the next authentication attempt.

You can configure the critical port to reinitialize hosts and move them out of the critical VLAN when the RADIUS server is again available. When this is configured, all critical ports in the critical-authentication state are automatically re-authenticated. For more information, see the *Cisco IOS Security Command Reference* and the "Configuring the Inaccessible Authentication Bypass Feature" section on page 2-44.

Feature Interactions

Inaccessible authentication bypass interacts with these features:

- Guest VLAN—Inaccessible authentication bypass is compatible with guest VLAN. When a guest VLAN is enabled on an 8021.x port, the features interact as follows:
  - If at least one RADIUS server is available, the switch assigns a client to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.
  - If all the RADIUS servers are not available and the client is connected to a critical port, the switch authenticates the client and puts the critical port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.

  - If all the RADIUS servers are not available and the client is not connected to a critical port, the switch might not assign clients to the guest VLAN if one is configured.
  - If all the RADIUS servers are not available and if a client is connected to a critical port and was previously assigned to a guest VLAN, the switch keeps the port in the guest VLAN.
- Restricted VLAN—If the port is already authorized in a restricted VLAN and the RADIUS servers are unavailable, the switch puts the critical port in the critical-authentication state in the restricted VLAN.
- 802.1x accounting—Accounting is not affected if the RADIUS servers are unavailable.
- Private VLAN—You can configure inaccessible authentication bypass on a private VLAN host port. The access VLAN must be a secondary private VLAN.
- Remote Switched Port Analyzer (RSPAN)—Do not configure an RSPAN VLAN as the RADIUS-configured or user-specified access VLAN for inaccessible authentication bypass.

802.1x Authentication with Port Security

You can configure an 802.1x port with port security in either single-host or multiple-hosts mode. (You also must configure port security on the port by using the **switchport port-security** interface configuration command.) When you enable port security and 802.1x on a port, 802.1x authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through an 802.1x port.

These are some examples of the interaction between 802.1x and port security on the switch:

- When a client is authenticated, and the port security table is not full, the client MAC address is added to the port security list of secure hosts. The port then proceeds to come up normally.

  When a client is authenticated and manually configured for port security, it is guaranteed an entry in the secure host table (unless port security static aging has been enabled).

  A security violation occurs if the client is authenticated, but the port security table is full. This can happen if the maximum number of secure hosts has been statically configured or if the client ages out of the secure host table. If the client address is aged, its place in the secure host table can be taken by another host.

  If the security violation is caused by the first authenticated host, the port becomes error-disabled and immediately shuts down.

  The port security violation modes determine the action for security violations.
- When you manually remove an 802.1x client address from the port security table by using the **no switchport port-security mac-address** *mac-address* interface configuration command, you should re-authenticate the 802.1x client by using the **authentication periodic** interface configuration command.
- When an 802.1x client logs off, the port changes to an unauthenticated state, and all dynamic entries in the secure host table are cleared, including the entry for the client. Normal authentication then takes place.
- If the port is administratively shut down, the port becomes unauthenticated, and all dynamic entries are removed from the secure host table.
- You can configure the **authentication violation** interface configuration command so that a port shuts down, generates a syslog error, accepts, or discards packets from a new device when it connects to an IEEE 802.1x-enabled port or when the maximum number of allowed devices have been authenticated. For more information see the "Maximum Number of Allowed Devices Per Port"

section on page 2-25 and the *Cisco IOS Security Command Reference*.

802.1x Authentication with Wake-on-LAN

The 802.1x authentication with the wake-on-LAN (WoL) feature allows dormant PCs to be powered when the switch receives a specific Ethernet frame, known as the *magic packet.* You can use this feature in environments where administrators need to connect to systems that have been powered down.

When a host that uses WoL is attached through an 802.1x port and the host powers off, the 802.1x port becomes unauthorized. The port can only receive and send EAPOL packets, and WoL magic packets cannot reach the host. When the PC is powered off, it is not authorized, and the switch port is not opened.

When the switch uses 802.1x authentication with WoL, the switch forwards traffic to unauthorized 802.1x ports, including magic packets. While the port is unauthorized, the switch continues to block ingress traffic other than EAPOL packets. The host can receive packets but cannot send packets to other devices in the network.

Note If PortFast is not enabled on the port, the port is forced to the bidirectional state.

When you configure a port as unidirectional by using the **authentication control-direction in** interface configuration command, the port changes to the spanning-tree forwarding state. The port can send packets to the host but cannot receive packets from the host.

When you configure a port as bidirectional by using the **authentication control-direction both** interface configuration command, the port is access-controlled in both directions. The port does not receive packets from or send packets to the host.

802.1x Authentication with MAC Authentication Bypass

You can configure the switch to authorize clients based on the client MAC address by using the MAC authentication bypass feature. For example, you can enable this feature on 802.1x ports connected to devices such as printers.

If 802.1x authentication times out while waiting for an EAPOL response from the client, the switch tries to authorize the client by using MAC authentication bypass.

When the MAC authentication bypass feature is enabled on an 802.1x port, the switch uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. After detecting a client on an 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is configured.

If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an 802.1x-capable supplicant and uses 802.1x authentication (not MAC authentication bypass) to authorize the interface. EAPOL history is cleared if the interface link status goes down.

If the switch already authorized a port by using MAC authentication bypass and detects an 802.1x supplicant, the switch does not unauthorize the client connected to the port. When re-authentication occurs, the switch uses 802.1x authentication as the preferred re-authentication process if the previous session ended because the Termination-Action RADIUS attribute value is DEFAULT.

Clients that were authorized with MAC authentication bypass can be re-authenticated. The re-authentication process is the same as that for clients that were authenticated with 802.1x. During re-authentication, the port remains in the previously assigned VLAN. If re-authentication is successful, the switch keeps the port in the same VLAN. If re-authentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If re-authentication is based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]) and if the Termination-Action RADIUS attribute (Attribute [29]) action is *Initialize,* (the attribute value is *DEFAULT*), the MAC authentication bypass session ends, and connectivity is lost during re-authentication. If MAC authentication bypass is enabled and the 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate re-authorization. For more information about these AV pairs, see RFC 3580, "802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines."

MAC authentication bypass interacts with the features:

- 802.1x authentication—You can enable MAC authentication bypass only if 802.1x authentication is enabled on the port.
- Guest VLAN—If a client has an invalid MAC address identity, the switch assigns the client to a guest VLAN if one is configured.
- Restricted VLAN—This feature is not supported when the client connected to an 802.1x port is authenticated with MAC authentication bypass.
- Port security—See the "802.1x Readiness Check" section on page 2-12.
- VLAN Membership Policy Server (VMPS)—802.1x and VMPS are mutually exclusive.
- Private VLAN—You can assign a client to a private VLAN.

802.1x User Distribution

You can configure 802.1x user distribution to load-balance users with the same group name across multiple different VLANs.

The VLANs are either supplied by the RADIUS server or configured through the switch CLI under a VLAN group name.

- Configure the RADIUS server to send more than one VLAN name for a user. The multiple VLAN names can be sent as part of the response to the user. The 802.1x user distribution tracks all the users in a particular VLAN and achieves load balancing by moving the authorized user to the least populated VLAN.
- Configure the RADIUS server to send a VLAN group name for a user. The VLAN group name can be sent as part of the response to the user. You can search for the selected VLAN group name among the VLAN group names that you configured by using the switch CLI. If the VLAN group name is found, the corresponding VLANs under this VLAN group name are searched to find the least populated VLAN. Load balancing is achieved by moving the corresponding authorized user to that VLAN.

Note The RADIUS server can send the VLAN information in any combination of VLAN IDs, VLAN names, or VLAN groups.

Network Admission Control Layer 2 IEEE 802.1x Validation

The switch supports the Network Admission Control (NAC) Layer 2 802.1x validation, which checks the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access. With NAC Layer 2 802.1x validation, you can do these tasks:

- Download the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute[29]) from the authentication server.
- Set the number of seconds between re-authentication attempts as the value of the Session-Timeout RADIUS attribute (Attribute[27]) and get an access policy against the client from the RADIUS server.
- Set the action to be taken when the switch tries to re-authenticate the client by using the Termination-Action RADIUS attribute (Attribute[29]). If the value is *DEFAULT* or is not set, the session ends. If the value is RADIUS-Request, the re-authentication process starts.
- See the NAC posture token, which shows the posture of the client, by using the **show authentication** command in user EXEC or privileged EXEC mode.
- Configure secondary private VLANs as guest VLANs.

Configuring NAC Layer 2 802.1x validation is similar to configuring 802.1x port-based authentication except that you must configure a posture token on the RADIUS server. For information about configuring NAC Layer 2 802.1x validation, see the "Configuring NAC Layer 2 IEEE 802.1x Validation" section on page 2-50 and the "Configuring Periodic Re-Authentication" section on page 2-36.

For more information about NAC, see the *Network Admission Control Configuration Guide, Cisco IOS Release 15MT*.

For more configuration information, see the "Authentication Manager" section on page 2-7.

Flexible Authentication Ordering

You can use flexible authentication ordering to configure the order of methods that a port uses to authenticate a new host. MAC authentication bypass and 802.1x can be the primary or secondary authentication methods, and web authentication can be the fallback method if either or both of those authentication attempts fail. For more information see the "Configuring Flexible Authentication Ordering" section on page 2-58.

Open1x Authentication

Open1x authentication allows a device access to a port before that device is authenticated. When open authentication is configured, a new host on the port can only send traffic to the switch. After the host is authenticated, the policies configured on the RADIUS server are applied to that host.

You can configure open authentication with these scenarios:

- Single-host mode with open authentication–Only one user is allowed network access before and after authentication.
- Multiple-hosts mode with open authentication–Any host can access the network.
- Multiple-authentication mode with open authentication–Similar to MDA, except multiple hosts can be authenticated.

For more information see the "Configuring 802.1x Accounting" section on page 2-41.

Using Voice Aware 802.1x Security

You use the voice aware 802.1x security feature to configure the switch to disable only the data VLAN on which a security violation occurs rather than to shut down the entire port. You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN shuts down the data VLAN.

For information on configuring voice aware 802.1x security, see the "Configuring Voice Aware 802.1x Security" section on page 2-31.

802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)

The Network Edge Access Topology (NEAT) feature extends identity to areas outside the wiring closet (such as conference rooms), allowing any type of device to authenticate on the port.

- 802.1x switch supplicant: You can configure a switch to act as a supplicant to another switch by using the 802.1x supplicant feature. This configuration is helpful in a scenario, where, for example, a switch is outside a wiring closet and is connected to an upstream switch through a trunk port. A switch configured with the 802.1x switch supplicant feature authenticates with the upstream switch for secure connectivity.

  Once the supplicant switch authenticates successfully the port mode changes from access to trunk.

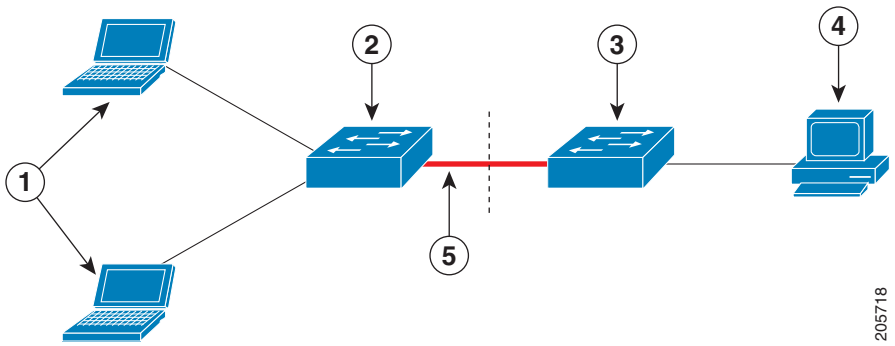
- If the access VLAN is configured on the authenticator switch, it becomes the native VLAN for the trunk port after successful authentication.

You can enable multiple-authentication mode on the authenticator switch interface that connects to one or more supplicant switches. Multiple-host mode is not supported on the authenticator switch interface.

Use the **dot1x supplicant force-multicast** global configuration command on the supplicant switch for Network Edge Access Topology (NEAT) to work in all host modes.

- Host Authorization: Ensures that only traffic from authorized hosts (connecting to the switch with supplicant) is allowed on the network. The switches use Client Information Signalling Protocol (CISP) to send the MAC addresses connecting to the supplicant switch to the authenticator switch, as shown in Figure 2-6.
- Auto enablement: Automatically enables trunk configuration on the authenticator switch, allowing user traffic from multiple VLANs coming from supplicant switches. Configure the cisco-av-pair as *device-traffic-class=switch* at the ACS. (You can configure this under the *group* or the *user* settings.)

Figure 2-6 Authenticator and Supplicant Switch using CISP



	Workstations (clients)		Supplicant switch (outside wiring closet)
	Authenticator switch		Access control server (ACS)
	Trunk port		

Common Session ID

Authentication manager uses a single session ID (referred to as a common session ID) for a client no matter which authentication method is used. This ID is used for all reporting purposes, such as the show commands and MIBs. The session ID appears with all per-session syslog messages.

The session ID includes:

- The IP address of the Network Access Device (NAD)
- A monotonically increasing unique 32-bit integer
- The session start time stamp (a 32-bit integer)

This example shows how the session ID appears in the output of the show authentication command. The session ID in this example is 160000050000000B288508E5:

```
Switch# show authentication sessions
Interface  MAC Address     Method   Domain   Status         Session ID
Fa4/0/4    0000.0000.0203  mab      DATA     Authz Success  160000050000000B288508E5
```

This is an example of how the session ID appears in the syslog output. The session ID in this example is also160000050000000B288508E5:

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on Interface
Fa4/0/4 AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client
(0000.0000.0203) on Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

The session ID is used by the NAD, the AAA server, and other report-analyzing applications to identify the client. The ID appears automatically. No configuration is required.

Prerequisites

Be sure to review the Guidelines and Limitations section and the Before You Begin section within each configuration section before configuring a feature.

Guidelines and Limitations

- When 802.1x is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.
- The 802.1x protocol is supported on Layer 2 static-access ports and Layer 3 routed ports, but it is not supported on these port types:

  - Trunk port—If you try to enable 802.1x on a trunk port, an error message appears, and 802.1x is not enabled. If you try to change the mode of an 802.1x-enabled port to trunk, an error message appears, and the port mode is not changed.
  - Dynamic-access ports—If you try to enable 802.1x on a dynamic-access (VLAN Query Protocol [VQP]) port, an error message appears, and 802.1x is not enabled. If you try to change an 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
  - EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x on an EtherChannel port, an error message appears, and 802.1x is not enabled.
  - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x on a port that is a SPAN or RSPAN destination port. However, 802.1x is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x on a SPAN or RSPAN source port.

- You can configure any VLAN except an RSPAN VLAN or a private VLAN.
- The 802.1x with VLAN assignment feature is not supported on private-VLAN ports, trunk ports, or ports with dynamic-access port assignment through a VMPS.
- You can configure 802.1x on a private-VLAN port, but do not configure 802.1x with port security on private-VLAN ports.
- Before globally enabling 802.1x on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x and EtherChannel are configured.

Maximum Number of Allowed Devices Per Port

This is the maximum number of devices allowed on an 802.1x-enabled port:

- In single-host mode, only one device is allowed on the access VLAN.
- In multiple-hosts mode, only one 802.1x supplicant is allowed on the port, but an unlimited number of non-802.1x hosts are allowed on the access VLAN.

802.1x User Distribution

- Confirm that at least one VLAN is mapped to the VLAN group.
- You can map more than one VLAN to a VLAN group.
- You can modify the VLAN group by adding or deleting a VLAN.
- When you clear an existing VLAN from the VLAN group name, none of the authenticated ports in the VLAN are cleared, but the mappings are removed from the existing VLAN group.
- If you clear the last VLAN from the VLAN group name, the VLAN group is cleared.
- You can clear a VLAN group even when the active VLANs are mapped to the group. When you clear a VLAN group, none of the ports or users that are in the authenticated state in any VLAN within the group are cleared, but the VLAN mappings to the VLAN group are cleared.

For more information, see the "Configuring 802.1x User Distribution" section on page 2-49.

NEAT

- You can configure NEAT ports with the same configurations as the other authentication ports. When the supplicant switch authenticates, the port mode changes from access to trunk based on the switch vendor-specific attributes (VSAs). (*device-traffic-class=switch*).

- The VSA changes the authenticator switch port mode from access to trunk and enables 802.1x trunk encapsulation. The access VLAN, if any, is then converted to a native trunk VLAN. The VSA does not change any of the port configurations on the supplicant.

For more information, see the "Configuring an Authenticator and a Supplicant Switch with NEAT" section on page 2-52.

Default Settings

Feature	Default Setting
AAA	Disabled.
RADIUS server • IP address • UDP authentication port • Key	 • None specified. • 1812. • None specified.
Switch 802.1x enable state	Disabled.
Per-port 802.1x enable state	Disabled (force-authorized). The port sends and receives normal traffic without 802.1x-based authentication of the client.
Periodic re-authentication	Disabled.
Number of seconds between re-authentication attempts	3600 seconds.
Re-authentication number	2 times (number of times that the switch restarts the authentication process before the port changes to the unauthorized state).
Quiet period	60 seconds (number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client).
Retransmission time	30 seconds (number of seconds that the switch should wait for a response to an EAP request/identity frame from the client before resending the request).
Maximum retransmission number	2 times (number of times that the switch will send an EAP-request/identity frame before restarting the authentication process).
Host mode	Single-host mode.

Feature	Default Setting
Client timeout period	30 seconds (when relaying a request from the authentication server to the client, the amount of time the switch waits for a response before resending the request to the client.)
Authentication server timeout period	30 seconds (when relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before resending the response to the server.) You can change this timeout period by using the authentication timer interface configuration command.

Configuring 802.1x Authentication

This section includes the following topics:

- Configuring 802.1x Readiness Check, page 2-28 (optional)
- Configuring the Switch-to-RADIUS Server Communication, page 2-29 (required)
- Configuring Voice Aware 802.1x Security, page 2-31 (optional)
- Configuring 802.1x Violation Modes, page 2-32
- Configuring 802.1x Authentication, page 2-33
- Configuring 802.1x Accounting, page 2-41 (optional)
- Configuring Periodic Re-Authentication, page 2-36 (optional)
- Manually Re-Authenticating a Client Connected to a Port, page 2-37 (optional)
- Changing the Quiet Period, page 2-37 (optional)
- Changing the Switch-to-Client Retransmission Time, page 2-38 (optional)
- Setting the Switch-to-Client Frame-Retransmission Number, page 2-38 (optional)
- Setting the Re-Authentication Number, page 2-39 (optional)
- Enabling MAC Move, page 2-40 (optional)
- Configuring 802.1x Accounting, page 2-41 (optional)
- Configuring a Guest VLAN, page 2-42
- Configuring a Restricted VLAN, page 2-43
- Configuring the Inaccessible Authentication Bypass Feature, page 2-44
- Configuring 802.1x Authentication with Wake-on-LAN, page 2-47
- Configuring MAC Authentication Bypass, page 2-48
- Configuring 802.1x User Distribution, page 2-49 (optional)
- Configuring NAC Layer 2 IEEE 802.1x Validation, page 2-50 (optional)
- Resetting the 802.1x Authentication Configuration to the Default Values, page 2-51 (optional)
- Disabling 802.1x Authentication on the Port, page 2-52 (optional)
- Configuring an Authenticator and a Supplicant Switch with NEAT, page 2-52 (optional)

- Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs, page 2-54 (optional)
- Configuring VLAN ID-based MAC Authentication, page 2-57 (optional)
- Configuring Flexible Authentication Ordering, page 2-58 (optional)
- Configuring Open1x, page 2-58 (optional)

Configuring 802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable.

The 802.1x readiness check is allowed on all ports that can be configured for 802.1x. The readiness check is not available on a port that is configured as **authentication port-control force-unauthorized**.

BEFORE YOU BEGIN

Follow these guidelines to enable the readiness check on the switch:

- The readiness check is typically used before 802.1x is enabled on the switch.
- If you use the **dot1x test eapol-capable** privileged EXEC command without specifying an interface, all the ports on the switch stack are tested.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated.
- The readiness check can be sent on a port that handles multiple hosts (for example, a PC that is connected to an IP phone). A syslog message is generated for each of the clients that respond to the readiness check within the timer period.

DETAILED STEPS

	Command	Purpose
Step 1	**dot1x test eapol-capable** [**interface** *interface-id*]	Enable the 802.1x readiness check on the switch. (Optional) For *interface-id* specify the port on which to check for IEEE 802.1x readiness. **Note** If you omit the optional **interface** keyword, all interfaces on the switch are tested.
Step 1	**configure terminal**	(Optional) Enter global configuration mode.
Step 2	**dot1x test timeout** *timeout*	(Optional) Configure the timeout used to wait for the EAPOL response. The range is from 1 to 65535 seconds. The default is 10 seconds.
Step 3	**end**	(Optional) Return to privileged EXEC mode.
Step 4	**show running-config**	(Optional) Verify your modified timeout values.

EXAMPLE

This example shows how to enable a readiness check on a switch to query a port. It also shows the response received from the queried port verifying that the device connected to it is 802.1x-capable.

```
switch# dot1x test eapol-capable interface gigabitethernet1/0/13

DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL
capable
```

Configuring the Switch-to-RADIUS Server Communication

RADIUS security servers are identified by their hostname or IP address, hostname and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, which enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service (for example, authentication), the second host entry configured acts as the failover backup to the first one. The RADIUS host entries are tried in the order that they were configured.

BEFORE YOU BEGIN

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string*	Configure the RADIUS server parameters. For *hostname* | *ip-address,* specify the hostname or IP address of the remote RADIUS server. For **auth-port** *port-number*, specify the UDP destination port for authentication requests. The default is 1812. The range is 0 to 65536. For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. **Note** Always configure the key as the last item in the **radius-server host** command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. If you want to use multiple RADIUS servers, reenter this command.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the specified RADIUS server, use the **no radius-server host** {*hostname* | *ip-address*} global configuration command.

EXAMPLE

This example shows how to specify the server with IP address 172.20.39.46 as the RADIUS server, to use port 1612 as the authorization port, and to set the encryption key to *rad123*, matching the key on the RADIUS server:

```
Switch(config)# radius-server host 172.120.39.46 auth-port 1612 key rad123
```

You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, **radius-server retransmit**, and the **radius-server key** global configuration commands. For more information, see the "Configuring Settings for All RADIUS Servers" section on page 1-43.

Configuring Voice Aware 802.1x Security

You use the voice aware 802.1x security feature to configure the switch to disable only the data VLAN on which a security violation occurs rather than to shut down the entire port. You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN shuts down the data VLAN.

BEFORE YOU BEGIN

Follow these guidelines to configure voice aware 802.1x voice security on the switch:

- You enable voice aware 802.1x security by entering the **reducible detect cause security-violation shutdown vlan** global configuration command. You disable voice aware 802.1x security by entering the **no** version of this command. This command applies to all 802.1x-configured ports in the switch.

Note If you do not include the **shutdown vlan** keywords, the entire port is shut down when it enters the error-disabled state.

- If you use the **errdisable recovery cause security-violation** global configuration command to configure error-disabled recovery, the port is automatically reenabled. If error-disabled recovery is not configured for the port, you reenable it by using the **shutdown** and **no-shutdown** interface configuration commands.
- You can reenable individual VLANs by using the **clear errdisable interface** *interface-id* **vlan** [*vlan-list*] privileged EXEC command. If you do not specify a range, all VLANs on the port are enabled.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**errdisable detect cause security-violation shutdown vlan**	Shut down any VLAN on which a security violation error occurs. **Note** If the **shutdown vlan** keywords are not included, the entire port enters the error-disabled state and shuts down.
Step 3	**errdisable recovery cause security-violation**	(Optional) Enable automatic per-VLAN error recovery.
Step 4	**clear errdisable interface** *interface-id* **vlan** [*vlan-list*]	(Optional) Reenable individual VLANs that have been error disabled. • For *interface-id* specify the port on which to reenable individual VLANs. • (Optional) For *vlan-list* specify a list of VLANs to be reenabled. If *vlan-list* is not specified, all VLANs are reenabled.
Step 5	**shutdown** **no-shutdown**	(Optional) Reenable an error-disabled VLAN, and clear all error-disable indications.

	Command	Purpose
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show errdisable detect**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure the switch to shut down any VLAN on which a security violation error occurs:

```
Switch(config)# errdisable detect cause security-violation shutdown vlan
```

This example shows how to reenable all VLANs that were error disabled on port GigabitEthernet2/02/2:

```
Switch# clear errdisable interface GigabitEthernet2/0/2 vlan
```

You can verify your settings by entering the **show errdisable detect** privileged EXEC command.

Configuring 802.1x Violation Modes

You can configure an 802.1x port so that it shuts down, generates a syslog error, or discards packets from a new device when:

- a device connects to an 802.1x-enabled port.
- the maximum number of allowed devices have been authenticated on the port.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1*	Create an 802.1x authentication method list. To create a default list to use when a named list is *not* specified in the **authentication** command, use the **default** keyword followed by the method to use in default situations. The default method list is automatically applied to all ports. For *method1*, enter the **group radius** keywords to apply the list of all RADIUS servers for authentication. **Note** Though other keywords are visible in the command-line help string, only the **group radius** keywords are supported.

	Command	Purpose
Step 4	**interface** *interface-id*	Specify the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 5	**switchport mode access**	Set the port to access mode.
Step 6	**authentication violation** {**shutdown** | **restrict** | **protect** | **replace**} or **dot1x violation-mode** {**shutdown** | **restrict** | **protect**}	Configure the violation mode. The keywords have these meanings: • **shutdown**–Error disable the port. • **restrict**–Generate a syslog error. • **protect**–Drop packets from any new device that sends traffic to the port. • **replace**–Tear down the old session and accept packets from any new device that sends traffic to the port.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show authentication**	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure an 802.1x port to report a syslog error when an authentication error is detected:

```
Switch# configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa authentication dot1x default
Switch(config)# interface ethernet0/2
Switch(config-if)# switchport mode access
Switch(config-if) authentication violation restrict
Switch(config-if) end
```

Configuring 802.1x Authentication

To configure 802.1x port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

To allow VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the 802.1x AAA process:

1. A user connects to a port on the switch.
2. Authentication is performed.
3. VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.
4. The switch sends a start message to an accounting server.
5. Re-authentication is performed, as necessary.
6. The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.

7. The user disconnects from the port.
8. The switch sends a stop message to the accounting server.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1*	Create an 802.1x authentication method list. To create a default list to use when a named list is *not* specified in the **authentication** command, use the **default** keyword followed by the method to use in default situations. The default method list is automatically applied to all ports. For *method1*, enter the **group radius** keywords to use the list of all RADIUS servers for authentication. **Note** Though other keywords are visible in the command-line help string, only the **group radius** keywords are supported.
Step 4	**dot1x system-auth-control**	Enable 802.1x authentication globally on the switch.
Step 5	**aaa authorization network** {**default**} **group radius**	(Optional) Configure the switch for user RADIUS authorization for all network-related service requests, such as VLAN assignment.
Step 6	**interface** *interface-id*	Specify the port connected to the client that is to be enabled for 802.1x authentication, and enter interface configuration mode.
Step 7	**authentication port-control auto**	Enable 802.1x authentication on the port.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show authentication**	Verify your entries.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure 802.1x authentication on a port:

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication dot1x default group radius
Switch(config)# dot1x system-auth-control
Switch(config)# interface ethernet0/2
Switch(config-if)# authentication port-control auto
Switch(config-if)# end
```

Configuring the Host Mode

Follow this procedure to allow a single host (client) or multiple hosts on an 802.1x-authorized port that has the **authentication port-control auto** interface configuration command set to **auto**.

Note This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to which multiple hosts are indirectly attached, and enter interface configuration mode.
Step 3	**authentication host-mode** [**multi-auth** | **multi-host** | **single-host**]	Set the authentication mode. The keywords have these meanings: • **multi-auth**–Allow multiple authenticated clients on the data VLAN. Each host is individually authenticated. **Note** The **multi-auth** keyword is only available with the **authentication host-mode** command. • **multi-host**–Allow multiple hosts on an 802.1x-authorized port after a single host has been authenticated. • **single-host**–Allow a single host (client) on an 802.1x-authorized port. Make sure that the **authentication port-control auto** interface configuration command set is set to **auto** for the specified interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show authentication** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note Although visible in the command-line interface help, the **authentication host-mode multi-domain** interface configuration command is not supported. Configuring this command on an interface puts it in the error-disabled state.

To disable multiple hosts on the port, use the **no authentication host-mode multi-host** interface configuration command.

EXAMPLE

This example shows how to enable 802.1x and to allow multiple hosts:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# authentication port-control auto
Switch(config-if)# authentication host-mode multi-host
```

Configuring Periodic Re-Authentication

You can enable periodic 802.1x client re-authentication and specify how often it occurs. If you do not specify a time period before enabling re-authentication, the number of seconds between re-authentication attempts is 3600. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**authentication periodic**	Enable periodic re-authentication of the client, which is disabled by default.
Step 4	**authentication timer** {{[**inactivity** | **reauthenticate**]} {**restart** *value*}} **or** **dot1x timeout reauth-period** *seconds*	Set the number of seconds between re-authentication attempts. The range is 1 to 65535; the default is 3600 seconds. This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show authentication** *interface-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable periodic re-authentication, use the **no authentication periodic** interface configuration command. To return to the default number of seconds between re-authentication attempts, use the **no authentication timer reauthenticate** interface configuration command.

EXAMPLE

This example shows how to enable periodic re-authentication and set the number of seconds between re-authentication attempts to 4000:

```
Switch(config-if)# authentication periodic
Switch(config-if)# authentication timer reauthenticate 4000
```

Manually Re-Authenticating a Client Connected to a Port

You can manually re-authenticate the client connected to a specific port at any time by entering the **authentication periodic** interface configuration command. This step is optional. If you want to enable or disable periodic re-authentication, see the "Configuring Periodic Re-Authentication" section on page 2-36.

EXAMPLE

This example shows how to manually re-authenticate the client connected to a port:

```
Switch(config-if)# authentication periodic
```

Changing the Quiet Period

When the switch cannot authenticate the client, the switch remains idle for a set period of time and then tries again. The **authentication timer reauthenticate** interface configuration command controls the quiet period. A failed authentication of the client might occur because the client provided an invalid password. You can provide a faster response time to the user by entering a smaller number than the default. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**authentication timer reauthenticate** *value*	Time, in seconds, after which an automatic re-authentication attempt starts. The range is 1 to 65535 seconds; the default is 3600.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show authentication interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default quiet period, use the **no authentication timer reauthenticate** interface configuration command.

EXAMPLE

This example shows how to set the quiet period on the switch to 30 seconds:

```
Switch(config-if)# authentication timer reauthenticate 30
```

Changing the Switch-to-Client Retransmission Time

The client responds to the EAP-request/identity frame from the switch with an EAP-response/identity frame. If the switch does not receive this response, it waits a set period of time (known as the retransmission time) and then resends the frame.

Follow this procedure to change the amount of time that the switch waits for client notification.This procedure is optional.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**dot1x timeout tx-period** *seconds*	Set the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 1 to 65535 seconds; the default is 30.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show authentication** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default retransmission time, use the **no dot1x timeout tx-period** interface configuration command.

EXAMPLE

This example shows how to set 60 as the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request:

```
Switch(config-if)# dot1x timeout tx-period 60
```

Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP frame (assuming no response is received) to the client before restarting the authentication process. This procedure is optional.

Note Only change the default value of this command to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**dot1x max-req** *count*	Set the number of times that the switch sends an EAP frame to the client before restarting the authentication process. The range is 1 to 10; the default is 2.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default retransmission number, use the **no dot1x max-req** interface configuration command.

EXAMPLE

This example shows how to set 5 as the number of times that the switch sends an EAP request before restarting the authentication process:

```
Switch(config-if)# dot1x max-req 5
```

Setting the Re-Authentication Number

You can also change the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. This procedure is optional.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**dot1x max-reauth-req** *count*	Set the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. The range is 1 to 10; the default is 2.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show authentication** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default re-authentication number, use the **no dot1x max-reauth-req** interface configuration command.

EXAMPLE

This example shows how to set 4 as the number of times that the switch restarts the authentication process before the port changes to the unauthorized state:

```
Switch(config-if)# dot1x max-reauth-req 4
```

Enabling MAC Move

MAC move allows an authenticated host to move from one port on the switch to another. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**authentication mac-move permit**	Enable the feature.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show run**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to globally enable MAC move on a switch:

```
Switch(config)# authentication mac-move permit
```

Configuring 802.1x Accounting

Enabling AAA system accounting with 802.1x accounting allows system reload events to be sent to the accounting RADIUS server for logging. The server can then infer that all active 802.1x sessions are closed.

Because RADIUS uses the unreliable UDP transport protocol, accounting messages might be lost due to poor network conditions. If the switch does not receive the accounting response message from the RADIUS server after a configurable number of retransmissions of an accounting request, this system message appears:

```
Accounting message %s for session %s failed to receive Accounting Response.
```

When the stop message is not sent successfully, this message appears:

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```

Follow this procedure to configure 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

BEFORE YOU BEGIN

You must configure the RADIUS server to perform accounting tasks, such as logging start, stop, and interim-update messages and time stamps.

Tip To allow your RADIUS server to perform accounting tasks, enable logging of "Update/Watchdog packets from this AAA client" in your RADIUS server Network Configuration tab. Next, enable "CVS RADIUS Accounting" in your RADIUS server System Configuration tab.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa accounting dot1x default start-stop group radius**	Enable 802.1x accounting using the list of all RADIUS servers.
Step 3	**aaa accounting system default start-stop group radius**	(Optional) Enables system accounting (using the list of all RADIUS servers) and generates system accounting reload event messages when the switch reloads.
Step 4	**end**	Return to privileged EXEc mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **show radius statistics** privileged EXEC command to display the number of RADIUS messages that do not receive the accounting response message.

EXAMPLE

This example shows how to configure 802.1x accounting. The first command configures the RADIUS server, specifying 1813 as the UDP port for accounting.

```
Switch(config)# radius-server host 172.120.39.46 auth-port 1812 acct-port 1813 key rad123
Switch(config)# aaa accounting dot1x default start-stop group radius
Switch(config)# aaa accounting system default start-stop group radius
```

Configuring a Guest VLAN

When you configure a guest VLAN, clients that are not 802.1x-capable are put into the guest VLAN when the server does not receive a response to its EAP request/identity frame. Clients that are 802.1x-capable but that fail authentication are not granted network access. The switch supports guest VLANs in single-host or multiple-hosts mode. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access**	Set the port to access mode.
Step 4	**authentication port-control auto**	Enable 802.1x authentication on the port.
Step 5	**authentication event no-response action authorize vlan** *vlan-id*	Specify an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN. Voice VLANs are not supported.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show authentication** *interface-id*	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable and remove the guest VLAN, use the **no authentication event no-response action authorize vlan** *vlan-id* interface configuration command. The port returns to the unauthorized state.

EXAMPLE

This example shows how to enable VLAN 2 as an 802.1x guest VLAN:

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# authentication event no-response action authorize vlan 2
```

This example shows how to set 3 as the quiet period on the switch, to set 15 as the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request, and to enable VLAN 2 as an 802.1x guest VLAN when an 802.1x port is connected to a DHCP client:

```
Switch(config-if)# authentication timer reauthenticate 3
Switch(config-if)# dot1x timeout tx-period 15
Switch(config-if)# authentication event no-response action authorize vlan 2
```

Configuring a Restricted VLAN

When you configure a restricted VLAN on a switch, clients that are 802.1x-compliant are moved into the restricted VLAN when the authentication server does not receive a valid username and password. The switch supports restricted VLANs only in single-host mode. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access**	Set the port to access mode.
Step 4	**authentication port-control auto**	Enable 802.1x authentication on the port.
Step 5	**authentication event fail action authorize** *vlan-id*	Specify an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. Voice VLANs are not supported.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show authentication** *interface-id*	(Optional) Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable and remove the restricted VLAN, use the **no authentication event fail** interface configuration command. The port returns to the unauthorized state.

EXAMPLE

This example shows how to enable *VLAN 2* as an 802.1x restricted VLAN:

```
Switch(config-if)# authentication event fail action authorize vlan 2
```

Configuring the Number of Authentication Attempts

You can configure the maximum number of authentication attempts allowed before a user is assigned to the restricted VLAN by using the **authentication event fail retry** *retry count* interface configuration command. The range of allowable authentication attempts is 1 to 5. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access**	Set the port to access mode.
Step 4	**authentication port-control auto**	Enable 802.1x authentication on the port.
Step 5	**authentication event fail action authorize vlan** *vlan-id*	Specify an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. Voice VLANs are not supported.
Step 6	**authentication event retry** *retry count*	Specify the number of authentication attempts to allow before a port moves to the restricted VLAN. The range is 1 to 5.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show authentication** *interface-id*	(Optional) Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default value, use the **no authentication event fail action authorize vlan** *vlan-id* **retry** interface configuration command.

EXAMPLE

This example shows how to set 4 as the number of authentication attempts allowed before the port moves to the restricted VLAN 2:

```
Switch(config-if)# authentication event fail action authorize vlan 2 retry 4
```

Configuring the Inaccessible Authentication Bypass Feature

You can configure the inaccessible bypass feature, also referred to as critical authentication or the AAA fail policy. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server dead-criteria time** *time* **tries** *tries*	(Optional) Set the conditions that are used to decide when a RADIUS server is considered unavailable or *dead*. The range for *time* is from 1 to 120 seconds. The switch dynamically determines the default *time* value that is 10 to 60 seconds. The range for *tries* is from 1 to 100. The switch dynamically determines the default *tries* parameter that is 10 to 100.
Step 3	**radius-server deadtime** *minutes*	(Optional) Set the number of minutes that a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.

	Command	Purpose
Step 4	**radius-server host** *ip-address* **[acct-port** *udp-port*] [**auth-port** *udp-port*] [**test username** *name* [**idle-time** *time*] [**ignore-acct-port**] [**ignore-auth-port**]] [**key** *string*]	(Optional) Configure the RADIUS server parameters by using these keywords: • **acct-port** *udp-port*—Specify the UDP port for the RADIUS accounting server. The range for the UDP port number is from 0 to 65536. The default is 1646. • **auth-port** *udp-port*—Specify the UDP port for the RADIUS authentication server. The range for the UDP port number is from 0 to 65536. The default is 1645. **Note** You should configure the UDP port for the RADIUS accounting server and the UDP port for the RADIUS authentication server to nondefault values. • **test username** *name*—Enable automated testing of the RADIUS server status, and specify the username to be used. • **idle-time** *time*—Set the interval of time in minutes after which the switch sends test packets to the server. The range is from 1 to 35791 minutes. The default is 60 minutes (1 hour). • **ignore-acct-port**—Disable testing on the RADIUS-server accounting port. • **ignore-auth-port**—Disable testing on the RADIUS-server authentication port. • **key** *string*—Specify the authentication and encryption key for all RADIUS communication between the switch and the RADIUS daemon. **Note** Always configure the key as the last item in the **radius-server host** command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. You can also configure the authentication and encryption key by using the **radius-server key** {**0** *string* | **7** *string* | *string*} global configuration command.
Step 5	**dot1x critical eapol**	(Optional) Specify that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port.

	Command	Purpose
Step 6	**authentication critical recovery delay** *milliseconds*	(Optional) Set the recovery delay period during which the switch waits to re-initialize a critical port when a RADIUS server that was unavailable becomes available. The range is from 1 to 10000 milliseconds. The default is 1000 milliseconds (a port can be re-initialized every second).
Step 7	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 8	**authentication event server dead action authorize vlan** *vlan-id*	Enable the inaccessible authentication bypass feature.
Step 9	**authentication event server alive action reinitialize**	Reinitialize all clients on the port.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show authentication interface** *interface-id*	(Optional) Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the RADIUS server default settings, use the **no radius-server dead-criteria**, the **no radius-server deadtime**, and the **no radius-server host** global configuration commands. To return to the default settings of inaccessible authentication bypass, use the **no dot1x critical eapol** and **no authentication critical recovery delay** global configuration commands. To disable inaccessible authentication bypass, use the **no authentication event server alive action reinitialize** interface configuration command.

EXAMPLE

This example shows how to configure the inaccessible authentication bypass feature:

```
Switch(config)# radius-server dead-criteria time 30 tries 20
Switch(config)# radius-server deadtime 60
Switch(config)# radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 test username
user1 idle-time 30 key abc1234
Switch(config)# dot1x critical eapol
Switch(config)# authentication critical recovery delay 2000
Switch(config)# interface gigabitethernet0/2
Switch(config)# radius-server deadtime 60
Switch(config-if)# interface gigabitethernet2/0/1
Switch(config-if)# authentication event server dead action authorize vlan 20
Switch(config-if)# authentication event server alive action reinitialize
Switch(config-if)# end
```

Configuring 802.1x Authentication with Wake-on-LAN

This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**authentication control-direction** {**both** | **in**}	Enable 802.1x authentication with Wake-on-LAN (WoL) on the port, and use these keywords to configure the port as bidirectional or unidirectional. • **both**—Sets the port as bidirectional. The port cannot receive packets from or send packets to the host. By default, the port is bidirectional. • **in**—Sets the port as unidirectional. The port can send packets to the host but cannot receive packets from the host.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show authentication interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable 802.1x authentication with WoL, use the **no authentication control-direction** interface configuration command.

EXAMPLE

This example shows how to enable 802.1x authentication with Wake-on-LAN (WoL) and set the port as bidirectional:

```
Switch(config-if)# authentication control-direction both
```

Configuring MAC Authentication Bypass

This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode. For the supported port types, see the "Guidelines and Limitations" section on page 2-24.

	Command	Purpose
Step 3	**authentication port-control auto**	Enable 802.1x authentication on the port.
Step 4	**mab** [**eap**]	Enable MAC authentication bypass (MAB). (Optional) Use the **eap** keyword to configure the switch to use EAP for authorization.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show authentication** *interface-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable MAC authentication bypass, use the **no mab** interface configuration command.

EXAMPLE

This example shows how to enable MAC authentication bypass:

```
Switch(config-if)# mab
```

Configuring 802.1x User Distribution

Follow this procedure to configure a VLAN group and to map a VLAN to it.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**vlan group** *vlan-group-name* **vlan-list** *vlan-list*	Configure a VLAN group, and map a single VLAN or a range of VLANs to it.
Step 2	**show vlan group all** *vlan-group-name*	Verify the configuration.
Step 3	**no vlan group** *vlan-group-name* **vlan-list** *vlan-list*	Clear the VLAN group configuration or elements of the VLAN group configuration.

EXAMPLE

This example shows how to configure the VLAN groups, map the VLANs to the groups, and verify the VLAN group configurations and mapping to the specified VLANs:

```
switch(config)# vlan group eng-dept vlan-list 10
end
switch# show vlan group group-name eng-dept
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10

switch# show vlan-group all
Group Name                      Vlans Mapped
```

```
-------------                    --------------
eng-dept                         10
hr-dept                          20
```

This example shows how to add a VLAN to an existing VLAN group and verify that the VLAN was added:

```
switch(config)# vlan group eng-dept vlan-list 30
end
switch# show vlan group eng-dept
Group Name                       Vlans Mapped
-------------                    --------------
eng-dept                         10,30
```

This example shows how to remove a VLAN from a VLAN group:

```
switch(config)# no vlan group eng-dept vlan-list 10
```

This example shows that when all the VLANs are cleared from a VLAN group, the VLAN group is cleared:

```
switch(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.
end
switch# show vlan group group-name eng-dept
```

This example shows how to clear all the VLAN groups:

```
switch(config)# no vlan group end-dept vlan-list all
end
switch# show vlan-group all
```

Configuring NAC Layer 2 IEEE 802.1x Validation

You can configure NAC Layer 2 802.1x validation, which is also referred to as 802.1x authentication with a RADIUS server. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**authentication event no-response action authorize vlan** *vlan-id*	Specify an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, or a voice VLAN as an 802.1x guest VLAN. Voice VLANs are not supported.

	Command	Purpose
Step 4	**authentication periodic**	Enable periodic re-authentication of the client, which is disabled by default.
Step 5	**authentication timer reauthenticate** *value*	Set the number of seconds between re-authentication attempts. The keyword has this meaning: • *value*—Sets the number of seconds from 1 to 65535. The default is 3600 seconds. This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show authentication** *interface-id*	Verify your 802.1x authentication configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure NAC Layer 2 802.1x validation:

```
Switch# configure terminal
Switch(config)# interface fastethernetethernet0/3
Switch(config-if)# authentication periodic
```

Resetting the 802.1x Authentication Configuration to the Default Values

This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the port to be configured.
Step 3	**dot1x default**	Reset the configurable 802.1x parameters to the default values.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

```
Switch(config)# interface fastethernetethernet0/3
Switch(config-if)# dot1x default
```

```
Switch(config-if)# end
```

Disabling 802.1x Authentication on the Port

You can disable 802.1x authentication on the port by using the **no dot1x pae** interface configuration command. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**no dot1x pae**	Disable 802.1x authentication on the port.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show authentication** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To configure the port as an 802.1x port access entity (PAE) authenticator, which enables IEEE 802.1x on the port but does not allow connected clients to be authorized, use the **dot1x pae authenticator** interface configuration command.

EXAMPLE

This example shows how to disable 802.1x authentication on the port:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# no dot1x pae authenticator
```

Configuring an Authenticator and a Supplicant Switch with NEAT

Configuring this feature requires that one switch outside a wiring closet is configured as a supplicant and is connected to an authenticator switch.

For overview information, see the "802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)" section on page 2-23.

Note The *cisco-av-pairs* must be configured as *device-traffic-class=switch* on the ACS, which sets the interface as a trunk after the supplicant is successfully authenticated.

Configuring the Authenticator

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cisp enable**	Enable CISP.
Step 3	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 4	**switchport mode access**	Set the port mode to **access**.
Step 5	**authentication port-control auto**	Set the port-authentication mode to auto.
Step 6	**dot1x pae authenticator**	Configure the interface as a port access entity (PAE) authenticator.
Step 7	**spanning-tree portfast trunk**	Enable Port Fast on an access port connected to a single workstation or server.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show running-config interface** *interface-id*	Verify your configuration.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure a switch as an 802.1x authenticator:

```
Switch# configure terminal
Switch(config)# cisp enable
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# authentication port-control auto
Switch(config-if)# dot1x pae authenticator
Switch(config-if)# spanning-tree portfast trunk
```

Configuring the Supplicant

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cisp enable**	Enable CISP.
Step 3	**dot1x credentials** *profile*	Create 802.1x credentials profile. This must be attached to the port that is configured as supplicant.
Step 4	**username** *suppswitch*	Create a username.
Step 5	**password** *password*	Create a password for the new username.
Step 6	**dot1x supplicant force-multicast**	Force the switch to send *only* multicast EAPOL packets when it receives either unicast or multicast packets. This also allows NEAT to work on the supplicant switch in all host modes.
Step 7	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 8	**switchport trunk encapsulation dot1q**	Set the port to trunk mode.
Step 9	**switchport mode trunk**	Configure the interface as a VLAN trunk port.
Step 10	**dot1x pae supplicant**	Configure the interface as a port access entity (PAE) supplicant.
Step 11	**dot1x credentials** *profile-name*	Attach the 802.1x credentials profile to the interface.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show running-config interface** *interface-id*	Verify your configuration.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure a switch as a supplicant:

```
Switch# configure terminal
Switch(config)# cisp enable
Switch(config)# dot1x credentials test
Switch(config)# username suppswitch
Switch(config)# password myswitch
Switch(config)# dot1x supplicant force-multicast
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# dot1x pae supplicant
Switch(config-if)# dot1x credentials test
Switch(config-if)# end
```

Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs

In addition to configuring 802.1x authentication on the switch, you need to configure the ACS. For more information, see the Cisco Secure ACS configuration guides.

Note You must configure a downloadable ACL on the ACS before downloading it to the switch.

After authentication on the port, you can use the **show ip access-list** privileged EXEC command to display the downloaded ACLs on the port.

Configuring Downloadable ACLs

The policies take effect after client authentication and the client IP address addition to the IP device tracking table. The switch then applies the downloadable ACL to the port.

BEFORE YOU BEGIN

Configure an ACL. See Chapter 6, "Configuring Network Security with ACLs" or Chapter 7, "Configuring IPv6 ACLs."

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip device tracking**	Configure the ip device tracking table.
Step 3	**aaa new-model**	Enable AAA.
Step 4	**aaa authorization network default group radius**	Set the authorization method to local. To remove the authorization method, use the **no aaa authorization network default group radius** command.
Step 5	**radius-server vsa send authentication**	Configure the network access server (NAS) to recognize and use only authentication vendor-specific attributes (VSAs).
Step 6	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 7	**ip access-group** *acl-id* **in**	Configure the default ACL on the port in the input direction. **Note** The *acl-id* is an access list name or number.
Step 8	**show running-config interface** *interface-id*	Verify your configuration.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

```
Switch(config)# ip device tracking
Switch(config)# aaa new-model
Switch(config)# aaa authorization network default group radius
Switch(config)# radius-server vsa send authentication
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group default_acl in
```

Configuring a Downloadable Policy

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* **deny source** *source-wildcard* **log**	Defines the default port ACL by using a source address and wildcard. The access-list-number is a decimal number from 1 to 99 or 1300 to 1999. Enter **deny** or **permit** to specify whether to deny or permit access if conditions are matched. The *source* is the source address of the network or host that sends a packet, such as this: • The 32-bit quantity in dotted-decimal format. • The keyword any as an abbreviation for source and source-wildcard value of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard value. • The keyword host as an abbreviation for source and source-wildcard of source 0.0.0.0. (Optional) Applies the source-wildcard wildcard bits to the source. (Optional) Enters log to cause an informational logging message about the packet that matches the entry to be sent to the console.
Step 3	**interface** *interface-id*	Enter interface configuration mode.
Step 4	**ip access-group** *acl-id* **in**	Configure the default ACL on the port in the input direction. **Note** The *acl-id* is an access list name or number.
Step 5	**exit**	Returns to global configuration mode.
Step 6	**aaa new-model**	Enables AAA.
Step 7	**aaa authorization network default group radius**	Sets the authorization method to local. To remove the authorization method, use the **no aaa authorization network default group radius** command.
Step 8	**ip device tracking**	Enables the IP device tracking table. To disable the IP device tracking table, use the **no ip device tracking** global configuration command.

	Command	Purpose
Step 9	**ip device tracking probe count** *count*	(Optional) Configures the IP device tracking table: • **count** *count*–Sets the number of times that the switch sends the ARP probe. The range is from 1 to 5. The default is 3.
Step 10	**radius-server vsa send authentication**	Configures the network access server to recognize and use vendor-specific attributes (VSAs). **Note** The downloadable ACL must be operational.
Step 11	**end**	Returns to privileged EXEC mode.
Step 12	**show ip device tracking all**	Displays information about the entries in the IP device tracking table.
Step 13	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

EXAMPLE

This example shows how to configure a switch for a downloadable policy:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# aaa new-model
Switch(config)# aaa authorization network default group radius
Switch(config)# ip device tracking
Switch(config)# ip access-list extended default_acl
Switch(config-ext-nacl)# permit ip any any
Switch(config-ext-nacl)# exit
Switch(config)# radius-server vsa send authentication
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group default_acl in
Switch(config-if)# exit
```

Configuring VLAN ID-based MAC Authentication

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mab request format attribute 32 vlan access-vlan**	Enable VLAN ID-based MAC authentication.
Step 3	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

There is no show command to confirm the status of VLAN ID-based MAC authentication. You can use the **debug radius accounting** privileged EXEC command to confirm the RADIUS attribute 32. For more information about this command, see the Cisco IOS Debug Command Reference.

EXAMPLE

This example shows how to globally enable VLAN ID-based MAC authentication on a switch:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# mab request format attribute 32 vlan access-vlan
Switch(config-if)# exit
```

Configuring Flexible Authentication Ordering

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**authentication order** [**dot1x** | **mab**] | {**webauth**}	(Optional) Set the order of authentication methods used on a port.
Step 4	**authentication priority** [**dot1x** | **mab**] | {**webauth**}	(Optional) Add an authentication method to the port-priority list.
Step 5	**show authentication**	(Optional) Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure a port attempt 802.1x authentication first, followed by web authentication as fallback method:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config)# authentication order dot1x webauth
```

Configuring Open1x

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**authentication control-direction** {**both** | **in**}	(Optional) Configure the port control as unidirectional or bidirectional.
Step 4	**authentication fallback** *name*	(Optional) Configure a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
Step 5	**authentication host-mode** [**multi-auth** | **multi-host** | **single-host**]	(Optional) Set the authorization manager mode on a port. **Note** Although visible in the command-line interface help, the **authentication host-mode multi-domain** interface configuration command is not supported. Configuring this command on an interface causes the interface to go into the error-disabled state.
Step 6	**authentication open**	(Optional) Enable or disable open access on a port.
Step 7	**authentication order** [**dot1x** | **mab**] | {**webauth**}	(Optional) Set the order of authentication methods used on a port.
Step 8	**authentication periodic**	(Optional) Enable or disable re-authentication on a port.
Step 9	**authentication port-control** {**auto** | **force-authorized** | **force-un authorized**}	(Optional) Enable manual control of the port authorization state.
Step 10	**show authentication**	(Optional) Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure Open1x on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config)# authentication control-direction both
Switch(config)# au ten tic at ion fallback profile1
Switch(config)# authentication host-mode multi-auth
Switch(config)# authentication open
Switch(config)# authentication order dot1x webauth
Switch(config)# authentication periodic
Switch(config)# authentication port-control auto
```

Verifying Configuration

Command	Purpose
show dot1x all statistics	Display 802.1x statistics for all ports.
show dot1x statistics interface *interface-id*	Display 802.1x statistics for a specific port.
show dot1x all or **show authentication method dotx**	Display the 802.1x administrative and operational status for the switch.
show dot1x interface *interface-id* or **show authentication interface** *interface-id*	Display the 802.1x administrative and operational status for a specific port.

Related Documents

- Cisco IOS Master Command List, All Releases
- Cisco IOS Debug Command Reference
- Cisco IOS Security Command Reference
- *Cisco Connected Grid Switches Security Software Configuration Guide*
  - "Configuring the Switch to Use Vendor-Specific RADIUS Attributes" section within the "Configuring Switch-Based Authentication" chapter
  - "Configuring Network Security with ACLs"

Configuring Web-Based Authentication

This chapter describes how to configure web-based authentication on the Cisco Industrial Ethernet 2000U Series Switches (IE 2000U).

- Information About Web-Based Authentication, page 3-1
- Prerequisites, page 3-7
- Guidelines and Limitations, page 3-7
- Default Settings, page 3-10
- Configuring Web-Based Authentication, page 3-10
- Verifying Configuration, page 3-19
- Configuration Example, page 3-19
- Related Documents, page 3-21

Note For complete syntax and usage information for the switch commands used in this chapter, refer to the documents listed in the "Related Documents" section on page 3-21.

Information About Web-Based Authentication

Use the web-based authentication feature, known as *web authentication proxy*, to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.

Note You can configure web-based authentication on Layer 2 and Layer 3 interfaces.

When you initiate an HTTP session, web-based authentication intercepts ingress HTTP packets from the host and sends an HTML login page to the users. The users enter their credentials, which the web-based authentication feature sends to the authentication, authorization, and accounting (AAA) server for authentication.

If authentication succeeds, web-based authentication sends a Login-Successful HTML page to the host and applies the access policies returned by the AAA server.

If authentication fails, web-based authentication forwards a Login-Fail HTML page to the user, prompting the user to retry the login. If the user exceeds the maximum number of attempts, web-based authentication forwards a Login-Expired HTML page to the host, and the user is placed on a watch list for a waiting period.

These sections describe the role of web-based authentication as part of AAA:

- Device Roles, page 3-2
- Host Detection, page 3-2
- Session Creation, page 3-3
- Authentication Process, page 3-3
- Web-Based Authentication Customizable Web Pages, page 3-4
- Local Web-Based Authentication Banner, page 3-4

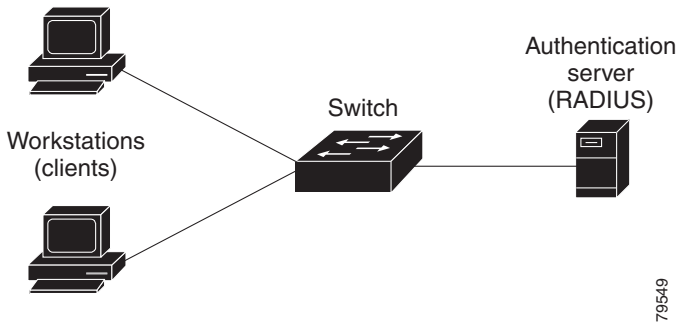
Device Roles

With web-based authentication, the devices in the network have these specific roles:

- *Client*—The device (workstation) that requests access to the LAN and the services and responds to requests from the switch. The workstation must be running an HTML browser with Java Script enabled.
- *Authentication server*—Authenticates the client. The authentication server validates the identity of the client and notifies the switch that the client is authorized to access the LAN and the switch services or that the client is denied.
- *Switch*—Controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client.

Figure 3-1 shows the roles of these devices in a network:

Figure 3-1 Web-Based Authentication Device Roles



Host Detection

The switch maintains an IP device tracking table to store information about detected hosts.

Note By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.

For Layer 2 interfaces, web-based authentication detects IP hosts by using these mechanisms:

- ARP based trigger—ARP redirect ACL allows web-based authentication to detect hosts with a static IP address or a dynamic IP address.
- Dynamic ARP inspection—A security feature that validates ARP packets in a network.
- DHCP snooping—Web-based authentication is notified when the switch creates a DHCP-binding entry for the host.

Session Creation

When web-based authentication detects a new host, it creates a session as follows:

- Reviews the exception list

  If the host IP is included in the exception list, the policy from the exception list entry is applied, and the session is established.

- Reviews for authorization bypass

  If the host IP is not on the exception list, web-based authentication sends a nonresponsive-host (NRH) request to the server.

  If the server response is *access accepted*, authorization is bypassed for this host. The session is established.

- Sets up the HTTP intercept ACL

  If the server response to the NRH request is *access rejected*, the HTTP intercept ACL is activated, and the session waits for HTTP traffic from the host.

Authentication Process

When you enable web-based authentication, these events occur:

1. The user initiates an HTTP session.
2. The HTTP traffic is intercepted, and authorization is initiated. The switch sends the login page to the user. The user enters a username and password, and the switch sends the entries to the authentication server.
3. If the authentication succeeds, the switch downloads and activates the user's access policy from the authentication server. The login success page is sent to the user.
4. If the authentication fails, the switch sends the login fail page. The user retries the login. If the number of failed attempts reaches the maximum, the switch sends the login expired page, and the host is placed in a watch list. After the watch list times out, the user can retry the authentication process.
5. If the authentication server does not respond to the switch, and if an AAA fail policy is configured, the switch applies the failure access policy to the host. The login success page is sent to the user. (See the "Local Web-Based Authentication Banner" section on page 3-4.)
6. The switch reauthenticates a client when the host does not respond to an ARP probe on a Layer 2 interface, or when the host does not send any traffic within the idle timeout on a Layer 3 interface.
7. The feature applies the downloaded timeout or the locally configured session timeout.
8. If the terminate action is RADIUS, the feature sends a nonresponsive host (NRH) request to the server. The terminate action is included in the response from the server.

9. If the terminate action is default, the session is dismantled, and the applied policy is removed.

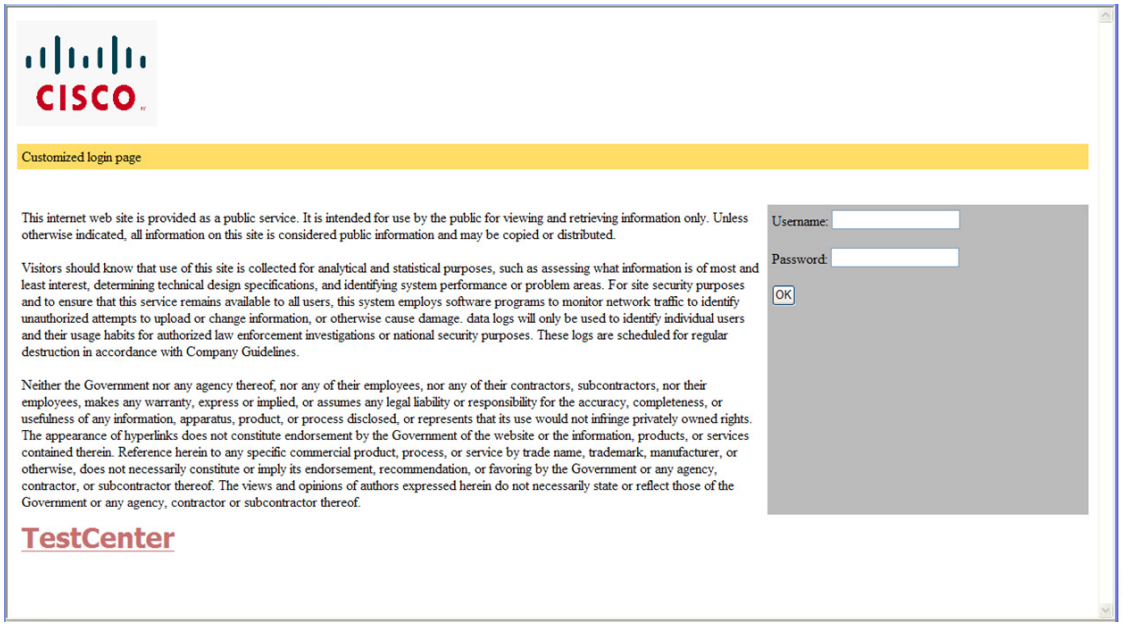
Web-Based Authentication Customizable Web Pages

During the web-based authentication process, the switch internal HTTP server hosts four HTML pages to deliver to an authenticating client. The server uses these pages to notify you of these four authentication-process states:

- Login—Your credentials are requested.
- Success—The login was successful.
- Fail—The login failed.
- Expire—The login session has expired because of excessive login failures.

You can substitute your HTML pages, as shown in Figure 3-2, for the default internal HTML pages. You can also specify a URL to which users are redirected after authentication occurs, which replaces the internal Success page.

Figure 3-2 Customizeable Authentication Page



For more information, see the "Customizing the Authentication Proxy Web Pages" section on page 3-14.

Local Web-Based Authentication Banner

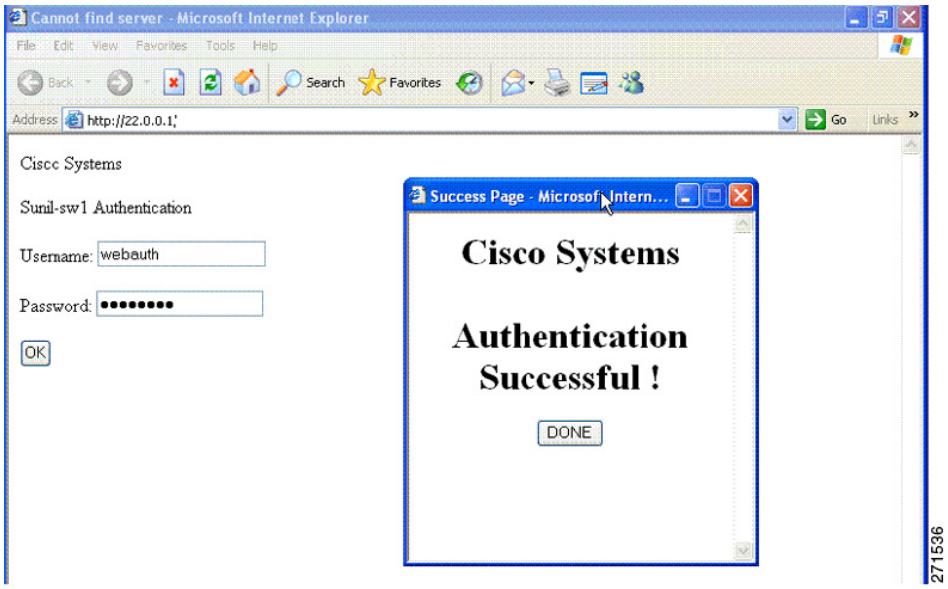
You can create a banner that will appear when you log in to a switch by using web-based authentication.

The banner appears on both the login page and the authentication-result pop-up pages.

- *Authentication Successful*
- *Authentication Failed*
- *Authentication Expired*

You create a banner by using the **ip admission auth-proxy-banner http** global configuration command. The default banner *Cisco Systems* and *Switch host-name Authentication* appear on the login page. *Cisco Systems* appears on the authentication result pop-up page, as shown in Figure 3-3.

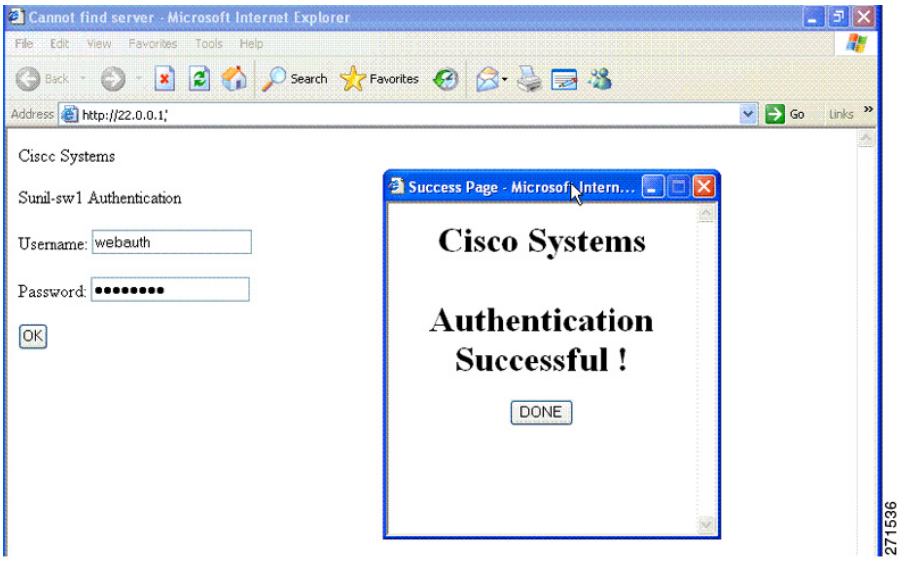
Figure 3-3 Authentication Successful Banner



You can also customize the banner, as shown in Figure 3-4.

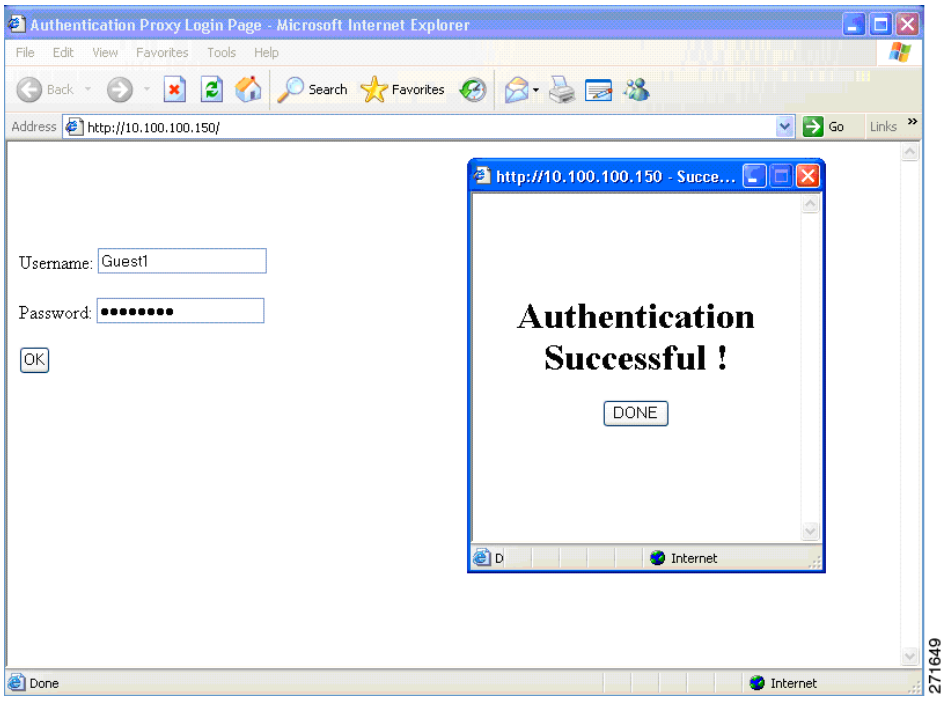
- Add a switch, router, or company name to the banner by using the **ip admission auth-proxy-banner http** *banner-text* global configuration command.
- Add a logo or text file to the banner by using the **ip admission auth-proxy-banner http** *file-path* global configuration command.

Figure 3-4 Customized Web Banner



If you do not enable a banner, only the username and password dialog boxes appear in the web authentication login page, and no banner appears when you log in to the switch, as shown in Figure 3-5.

Figure 3-5 Login Page With No Banner



For more information, see the *Cisco IOS Security Command Reference* and the "Configuring a Web Authentication Local Banner" section on page 3-18.

Prerequisites

Be sure to review the Guidelines and Limitations section and the Before You Begin section within each configuration section before configuring a feature.

Guidelines and Limitations

- Web-Based Authentication, page 3-7
- Customized Web Pages and Redirection URL, page 3-7
- Web-based Authentication Interactions with Other Features, page 3-8

Web-Based Authentication

- Web-based authentication is an ingress-only feature.
- You can configure web-based authentication only on access ports. Web-based authentication is not supported on trunk ports, EtherChannel member ports, or dynamic trunk ports.
- You must configure the default ACL on the interface before configuring web-based authentication. Configure a port ACL for a Layer 2 interface or a Cisco IOS ACL for a Layer 3 interface.
- You cannot authenticate hosts on Layer 2 interfaces with static ARP cache assignment. These hosts are not detected by the web-based authentication feature because they do not send ARP messages.
- By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.
- You must configure at least one IP address to run the switch HTTP server. You must also configure routes to reach each host IP address. The HTTP server sends the HTTP login page to the host.
- Hosts that are more than one hop away might experience traffic disruption if an STP topology change results in the host traffic arriving on a different port. This occurs because the ARP and DHCP updates might not be sent after a Layer 2 (STP) topology change.
- Web-based authentication does not support VLAN assignment as a downloadable-host policy.
- Web-based authentication is not supported for IPv6 traffic.

Customized Web Pages and Redirection URL

Follow these guidelines for customizing the authentication proxy web pages and specifying a redirection URL:

- You can use a logo or specify text in the *login*, *success*, *failure*, and *expire* web pages.
- On the banner page, you can specify text in the login page.
- The pages are in HTML.
- You must include an HTML redirect command in the success page to access a specific URL.
- The URL string must be a valid URL (for example, http://www.cisco.com). An incomplete URL might cause *page not found* or similar errors on a web browser.
- The login form must accept user entries for the username and password and must show them as **uname** and **pwd**.

- The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.
- If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and is not available in the CLI. You can perform redirection in the custom-login success page.
- If the CLI command redirecting users to specific URL after authentication occurs is entered and then the command configuring web pages is entered, the CLI command redirecting users to a specific URL does not take effect.
- If the custom web pages feature is enabled, a configured auth-proxy-banner is not used.
- If the redirection URL feature is enabled, a configured auth-proxy-banner is not used.
- The four custom HTML files must be present on the flash memory of the switch. The maximum size of each HTML file is 8 KB.
- Configured web pages can be copied to the switch boot flash or flash.
- Configured pages can be accessed from the flash on the stack master or members.
- The login page can be on one flash, and the success and failure pages can be another flash (for example, the flash on the stack master or a member).
- You must configure all four pages. If you specify fewer than four files, the internal default HTML pages are used.
- All of the logo files (image, flash, audio, video, and so on) that are stored in the system directory (for example, flash, disk0, or disk) and that must be displayed on the login page must use *web_auth_<filename>* as the file name.
- The configured authentication proxy feature supports both HTTP and SSL.
- Any images on the custom pages must be on an accessible HTTP server. Configure an intercept ACL within the admission rule.
- Any external link from a custom page requires configuration of an intercept ACL within the admission rule.
- To access a valid DNS server, any name resolution required for external links or images requires configuration of an intercept ACL within the admission rule.

Web-based Authentication Interactions with Other Features

- Port Security, page 3-9
- LAN Port IP, page 3-9
- Gateway IP, page 3-9
- ACLs, page 3-9
- Context-Based Access Control, page 3-9
- 802.1x Authentication, page 3-9

- EtherChannel, page 3-10

Port Security

You can configure web-based authentication and port security on the same port. Web-based authentication authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through the port.

For more information about enabling port security, see the "Configuring Port Security" section in the *Cisco Connected Grid Switches System Management Software Configuration Guide*.

LAN Port IP

You can configure LAN port IP (LPIP) and Layer 2 web-based authentication on the same port. The host is authenticated by using web-based authentication first, followed by LPIP posture validation. The LPIP host policy overrides the web-based authentication host policy.

If the web-based authentication idle timer expires, the NAC policy is removed, the host is authenticated, and posture is validated again.

Gateway IP

You cannot configure Gateway IP (GWIP) on a Layer 3 VLAN interface if web-based authentication is configured on any of the switch ports in the VLAN.

You can configure web-based authentication on the same Layer 3 interface as Gateway IP. The host policies for both features are applied in software. The GWIP policy overrides the web-based authentication host policy.

ACLs

If you configure a VLAN ACL or a Cisco IOS ACL on an interface, the ACL is applied to the host traffic only after the web-based authentication host policy is applied.

For Layer 2 web-based authentication, you must configure a port ACL (PACL) as the default access policy for ingress traffic from hosts connected to the port. After authentication, the web-based authentication host policy overrides the PACL.

You cannot configure a MAC ACL and web-based authentication on the same interface.

You cannot configure web-based authentication on a port whose access VLAN is configured for VACL capture.

Context-Based Access Control

Web-based authentication cannot be configured on a Layer 2 port if context-based access control (CBAC) is configured on the Layer 3 VLAN interface of the port VLAN.

802.1x Authentication

You cannot configure web-based authentication on the same port as 802.1x authentication except as a fallback authentication method.

EtherChannel

You can configure web-based authentication on a Layer 2 EtherChannel interface. The web-based authentication configuration applies to all member channels.

Default Settings

Feature	Default Setting
AAA	Disabled
RADIUS server • IP address • UDP authentication port • Key	 • None specified • 1812 • None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Configuring Web-Based Authentication

- Configuring the Authentication Rule and Interfaces, page 3-10
- Configuring AAA Authentication, page 3-11
- Configuring Switch-to-RADIUS Server Communication, page 3-12
- Configuring the HTTP Server, page 3-14
- Customizing the Authentication Proxy Web Pages, page 3-14
- Specifying a Redirection URL for Successful Login, page 3-15
- Configuring an AAA Fail Policy, page 3-16
- Configuring the Web-Based Authentication Parameters, page 3-17
- Removing Web-Based Authentication Cache Entries, page 3-18

Configuring the Authentication Rule and Interfaces

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**ip admission name** *name* **proxy http**	Configure an authentication rule for web-based authorization.

	Command	Purpose
Step 2	**interface** *type slot/port*	Enter interface configuration mode and specify the ingress Layer 2 or Layer 3 interface to be enabled for web-based authentication. *type* can be fastethernet, gigabit ethernet, or tengigabitethernet.
Step 3	**ip access-group** *name*	Apply the default ACL.
Step 4	**ip admission** *name*	Configure web-based authentication on the specified interface.
Step 5	**exit**	Return to configuration mode.
Step 6	**ip device tracking**	Enable the IP device tracking table.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ip admission configuration**	Display the configuration.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to enable web-based authentication on Fast Ethernet port 5/1:

```
Switch(config)# ip admission name webauth1 proxy http
Switch(config)# interface fastethernet 5/1
Switch(config-if)# ip admission webauth1
Switch(config-if)# exit
Switch(config)# ip device tracking
```

This example shows how to verify the configuration:

```
Switch# show ip admission configuration
Authentication Proxy Banner not configured
Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Watch-list is disabled

Authentication Proxy Rule Configuration
 Auth-proxy name webauth1
    http list not specified inactivity-time 60 minutes

Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

Configuring AAA Authentication

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**aaa new-model**	Enable AAA functionality.
Step 2	**aaa authentication login default group** {*tacacs+* | *radius*}	Define the list of authentication methods at login.
Step 3	**aaa authorization auth-proxy default group** {*tacacs+* | *radius*}	Create an authorization method list for web-based authorization.
Step 4	**tacacs-server host** {*hostname* | *ip_address*}	Specify an AAA server. For RADIUS servers, see the "Configuring Switch-to-RADIUS Server Communication" section on page 3-12.
Step 5	**tacacs-server key** {*key-data*}	Configure the authorization and encryption key used between the switch and the TACACS server.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to enable AAA:

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication login default group tacacs+
Switch(config)# aaa authorization auth-proxy default group tacacs+
```

Configuring Switch-to-RADIUS Server Communication

RADIUS security servers identification includes:

- Host name
- Host IP address
- Host name and specific UDP port numbers
- IP address and specific UDP port numbers

The combination of the IP address and UDP port number creates a unique identifier that enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service (for example, authentication) the second host entry that is configured functions as the failover backup to the first one. The RADIUS host entries are chosen in the order that they were configured.

BEFORE YOU BEGIN

When you configure the RADIUS server parameters:

- Specify the **key** *string* on a separate command line.
- For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server.
- When you specify the **key** *string*, use spaces within and at the end of the key. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon.

- You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, **radius-server retransmit**, and the **radius-server key** global configuration commands. For more information, see the *RADIUS Configuration Guide, Cisco IOS Release 15M&T* and the *Cisco IOS Security Command Reference*.

Note You need to configure some settings on the RADIUS server, including the switch IP address, the key string to be shared by both the server and the switch, and the downloadable ACL (DACL). For more information, see the RADIUS server documentation.

DETAILED STEPS

	Command	Purpose
Step 1	**ip radius source-interface** *interface_name*	Specify that the RADIUS packets have the IP address of the indicated interface.
Step 2	**radius-server host** {*hostname* | *ip-address*} **test username** *username*	Specify the host name or IP address of the remote RADIUS server. The **test username** *username* option enables automated testing of the RADIUS server connection. The specified *username* does not need to be a valid user name. The **key** option specifies an authentication and encryption key to use between the switch and the RADIUS server. To use multiple RADIUS servers, reenter this command for each server.
Step 3	**radius-server key** *string*	Configure the authorization and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 4	**radius-server vsa send authentication**	Enable downloading of an ACL from the RADIUS server.
Step 5	**radius-server dead-criteria tries** *num-tries*	Specify the number of unanswered messages sent to a RADIUS server before considering the server to be inactive. The range of *num-tries* is 1 to 100.

EXAMPLE

This example shows how to configure the RADIUS server parameters on a switch:

```
Switch(config)# ip radius source-interface Vlan80
Switch(config)# radius-server host 172.120.39.46 test username user1
Switch(config)# radius-server key rad123
Switch(config)# radius-server dead-criteria tries 2
```

Configuring the HTTP Server

To use web-based authentication, you must enable the HTTP server within the switch. You can enable the server for either HTTP or HTTPS.

Note To ensure secure authentication when you enter the **ip http secure-server** command, the login page is always in HTTPS (secure HTTP) even if the user sends an HTTP request.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**ip http server**	Enable the HTTP server. The web-based authentication feature uses the HTTP server to communicate with the hosts for user authentication.
Step 2	**ip http secure-server**	Enable HTTPS.

EXAMPLE

This example shows how to enable HTTPS:

```
Switch(config)# ip http secure-server
```

Customizing the Authentication Proxy Web Pages

You can configure web authentication to display four substitute HTML pages to the user in place of the switch default HTML pages during web-based authentication.

BEFORE YOU BEGIN

- To specify the use of your custom authentication proxy web pages, first store your custom HTML files on the switch flash memory.
- Be familiar with the guidelines listed in the "Customized Web Pages and Redirection URL" section on page 3-7.

DETAILED STEPS

	Command	Purpose
Step 1	**ip admission proxy http login page file** *device:login-filename*	Specify the location in the switch memory file system of the custom HTML file to use in place of the default login page. The *device:* is flash memory.
Step 2	**ip admission proxy http success page file** *device:success-filename*	Specify the location of the custom HTML file to use in place of the default login success page.
Step 3	**ip admission proxy http failure page file** *device:fail-filename*	Specify the location of the custom HTML file to use in place of the default login failure page.
Step 4	**ip admission proxy http login expired page file** *device:expired-filename*	Specify the location of the custom HTML file to use in place of the default login expired page.

To remove the specification of a custom file, use the **no** form of the command.

EXAMPLE

This example shows how to configure custom authentication proxy web pages:

```
Switch(config)# ip admission proxy http login page file flash:login.htm
Switch(config)# ip admission proxy http success page file flash:success.htm
Switch(config)# ip admission proxy http fail page file flash:fail.htm
Switch(config)# ip admission proxy http login expired page flash flash:expired.htm
```

This example shows how to verify the configuration of custom authentication proxy web pages:

```
Switch# show ip admission configuration
Authentication proxy webpage
 Login page           : flash:login.htm
 Success page         : flash:success.htm
 Fail Page            : flash:fail.htm
 Login expired Page   : flash:expired.htm

Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Session ratelimit is 100
Authentication Proxy Watch-list is disabled
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

Specifying a Redirection URL for Successful Login

You can specify a URL to which the user is redirected after authentication, effectively replacing the internal S*uccess* HTML page.

BEFORE YOU BEGIN

Be familiar with the guidelines listed in the "Customized Web Pages and Redirection URL" section on page 3-7.

DETAILED STEPS

Command	Purpose
ip admission proxy http success redirect *url-string*	Specify a URL for redirection of the user in place of the default login success page.

To remove the specification of a redirection URL, use the **no** form of the command.

EXAMPLE

This example shows how to configure a redirection URL for successful login:

```
Switch(config)# ip admission proxy http success redirect www.cisco.com
```

This example shows how to verify the redirection URL for successful login:

```
Switch# show ip admission configuration
Authentication Proxy Banner not configured
Customizable Authentication Proxy webpage not configured
HTTP Authentication success redirect to URL: http://www.cisco.com
Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Watch-list is disabled
Authentication Proxy Max HTTP process is 7
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

Configuring an AAA Fail Policy

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**ip admission name** *rule-name* **proxy http event timeout aaa policy identity** *identity_policy_name*	Create an AAA failure rule and associate an identity policy to apply to sessions when the AAA server is unreachable. **Note** To remove the rule, use the **no ip admission name** *rule-name* **proxy http event timeout aaa policy identity** global configuration command.
Step 2	**ip admission ratelimit aaa-down** *number_of_sessions*	(Optional) Limit the rate of authentication attempts from hosts in the AAA down state to avoid flooding the AAA server when it returns to service.

EXAMPLE

This example shows how to apply an AAA failure policy:

```
Switch(config)# ip admission name AAA_FAIL_POLICY proxy http event timeout aaa policy
identity GLOBAL_POLICY1
```

This example shows how to determine whether any connected hosts are in the AAA Down state:

```
Switch# show ip admission cache
Authentication Proxy Cache
 Client IP 209.165.201.11 Port 0, timeout 60, state ESTAB (AAA Down)
```

This example shows how to view detailed information about a particular session based on the host IP address:

```
Switch# show ip admission cache 209.165.201.11
Address              : 209.165.201.11
MAC Address          : 0000.0000.0000
Interface            : Vlan333
Port                 : 3999
Timeout              : 60
Age                  : 1
State                : AAA Down
AAA Down policy      : AAA_FAIL_POLICY
```

Configuring the Web-Based Authentication Parameters

You can configure the maximum number of failed login attempts before the client is placed in a watch list for a waiting period.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**ip admission max-login-attempts** *number*	Set the maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
Step 2	**end**	Returns to privileged EXEC mode.
Step 3	**show ip admission configuration**	Display the authentication proxy configuration.
Step 4	**show ip admission cache**	Display the list of authentication entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to set the maximum number of failed login attempts to 10:

```
Switch(config)# ip admission max-login-attempts 10
```

Configuring a Web Authentication Local Banner

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip admission auth-proxy-banner http** *[banner-text | file-path]*	Enable the local banner. (Optional) Create a custom banner by entering *C banner-text C,* where *C* is a delimiting character or a file-path that indicates a file (for example, a logo or text file) that appears in the banner.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to configure a local banner with the custom message *My Switch*:

```
Switch(config) configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa ip auth-proxy auth-proxy-banner C My Switch C
Switch(config) end
```

Removing Web-Based Authentication Cache Entries

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

Command	Purpose
clear ip auth-proxy cache {* | *host ip address*}	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.
clear ip admission cache {* | *host ip address*}	Delete IP admission cache entries. Use an asterisk to delete all cache entries and associated dynamic access lists. Enter a specific IP address to delete the entry for a single host.

EXAMPLE

This example shows how to remove the web-based authentication session for the client at the IP address 209.165.201.1:

```
Switch# clear ip auth-proxy cache 209.165.201.1
```

Verifying Configuration

Command	Purpose
show authentication sessions [**interface** *type slot/port*]	Displays the web-based authentication settings. type = fastethernet, gigabitethernet, or tengigabitethernet (Optional) Use the **interface** keyword to display the web-based authentication settings for a specific interface.

Configuration Example

This example shows how to enable web-based authentication on Fast Ethernet port 5/1:

```
Switch(config)# ip admission name webauth1 proxy http
Switch(config)# interface fastethernet 5/1
Switch(config-if)# ip admission webauth1
Switch(config-if)# exit
Switch(config)# ip device tracking
```

This example shows how to verify the configuration:

```
Switch# show ip admission configuration
Authentication Proxy Banner not configured
Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Watch-list is disabled

Authentication Proxy Rule Configuration
 Auth-proxy name webauth1
    http list not specified inactivity-time 60 minutes

Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

This example shows how to enable AAA:

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication login default group tacacs+
Switch(config)# aaa authorization auth-proxy default group tacacs+
```

This example shows how to configure the RADIUS server parameters on a switch:

```
Switch(config)# ip radius source-interface Vlan80
Switch(config)# radius-server host 172.120.39.46 test username user1
Switch(config)# radius-server key rad123
Switch(config)# radius-server dead-criteria tries 2
```

This example shows how to enable HTTPS:

```
Switch(config)# ip http secure-server
```

This example shows how to configure custom authentication proxy web pages:

```
Switch(config)# ip admission proxy http login page file flash:login.htm
Switch(config)# ip admission proxy http success page file flash:success.htm
Switch(config)# ip admission proxy http fail page file flash:fail.htm
Switch(config)# ip admission proxy http login expired page flash flash:expired.htm
```

This example shows how to verify the configuration of custom authentication proxy web pages:

```
Switch# show ip admission configuration
Authentication proxy webpage
 Login page           : flash:login.htm
 Success page         : flash:success.htm
 Fail Page            : flash:fail.htm
 Login expired Page   : flash:expired.htm

Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Session ratelimit is 100
Authentication Proxy Watch-list is disabled
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

This example shows how to configure a redirection URL for successful login:

```
Switch(config)# ip admission proxy http success redirect www.cisco.com
```

This example shows how to verify the redirection URL for successful login:

```
Switch# show ip admission configuration
Authentication Proxy Banner not configured
Customizable Authentication Proxy webpage not configured
HTTP Authentication success redirect to URL: http://www.cisco.com
Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Watch-list is disabled
Authentication Proxy Max HTTP process is 7
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

This example shows how to apply an AAA failure policy:

```
Switch(config)# ip admission name AAA_FAIL_POLICY proxy http event timeout aaa policy
identity GLOBAL_POLICY1
```

This example shows how to determine whether any connected hosts are in the AAA Down state:

```
Switch# show ip admission cache
Authentication Proxy Cache
 Client IP 209.165.201.11 Port 0, timeout 60, state ESTAB (AAA Down)
```

This example shows how to view detailed information about a particular session based on the host IP address:

```
Switch# show ip admission cache 209.165.201.11
Address             : 209.165.201.11
MAC Address         : 0000.0000.0000
Interface           : Vlan333
Port                : 3999
```

```
Timeout             : 60
Age                 : 1
State               : AAA Down
AAA Down policy     : AAA_FAIL_POLICY
```

This example shows how to set the maximum number of failed login attempts to 10:

```
Switch(config)# ip admission max-login-attempts 10
```

This example shows how to configure a local banner with the custom message *My Switch*:

```
Switch(config) configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa ip auth-proxy auth-proxy-banner C My Switch C
Switch(config) end
```

This example shows how to remove the web-based authentication session for the client at the IP address 209.165.201.1:

```
Switch# clear ip auth-proxy cache 209.165.201.1
```

Related Documents

- Cisco IOS Security Command Reference
- RADIUS Configuration Guide, Cisco IOS Release 15M&T
- Cisco IOS HTTP Services Command Reference

Configuring DHCP Features and IP Source Guard

This chapter describes how to configure DHCP snooping and option-82 data insertion, and the DHCP server port-based address allocation features on the Cisco Industrial Ethernet 2000U Series Switches (IE 2000U). This chapter also describes how to configure the IP source guard feature.

Note For complete syntax and usage information for the commands used in this chapter, see the documents listed in the "Related Documents" section on page 4-25.

- Information About DHCP Features, page 4-1
- Information About DHCP Server Port-Based Address Allocation, page 4-7
- Information About IP Source Guard, page 4-8
- Prerequisites, page 4-10
- Guidelines and Limitations, page 4-10
- Default Settings, page 4-13
- Configuring DHCP Features, page 4-14
- Configuring DHCP Server Port-Based Address Allocation, page 4-20
- Configuring IP Source Guard, page 4-23
- Verifying Configuration, page 4-24
- Related Documents, page 4-25

Information About DHCP Features

DHCP is widely used in LAN environments to dynamically assign host IP addresses from a centralized server, which significantly reduces the overhead of administration of IP addresses. DHCP also helps conserve the limited IP address space because IP addresses no longer need to be permanently assigned to hosts; only those hosts that are connected to the network consume IP addresses.

- DHCP Server, page 4-2
- DHCP Relay Agent, page 4-2
- DHCP Snooping, page 4-2
- Option-82 Data Insertion, page 4-3
- Cisco IOS DHCP Server Database, page 4-6

- DHCP Snooping Binding Database, page 4-6

Tip For information about the DHCP client, see "Related Documents" section on page 4-25.

DHCP Server

The DHCP server assigns IP addresses from specified address pools on a switch or router to DHCP clients and manages them. If the DHCP server cannot give the DHCP client the requested configuration parameters from its database, it can forward the request to one or more secondary DHCP servers defined by the network administrator.

DHCP Relay Agent

A DHCP relay agent is a Layer 3 device that forwards DHCP packets between clients and servers. Relay agents forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is different from the normal Layer 2 forwarding, in which IP datagrams are switched transparently between networks. Relay agents receive DHCP messages and generate new DHCP messages to send on egress interfaces.

DHCP Snooping

DHCP snooping is a DHCP security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, also referred to as a DHCP snooping binding table. For more information about this database, see the "Configuring DHCP Server Port-Based Address Allocation" section on page 4-20.

DHCP snooping acts like a firewall between untrusted hosts and DHCP servers. You use DHCP snooping to differentiate between untrusted interfaces connected to the end user and trusted interfaces connected to the DHCP server or another switch.

Note For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces.

An untrusted DHCP message is a message that is received from outside the network or firewall. When you use DHCP snooping in a service-provider environment, an untrusted message is sent from a device that is not in the service-provider network, such as a customer's switch. Messages from unknown devices are untrusted because they can be sources of traffic attacks.

The DHCP snooping binding database has the MAC address, the IP address, the lease time, the binding type, the VLAN number, and the interface information that corresponds to the local untrusted interfaces of a switch. It does not have information regarding hosts interconnected with a trusted interface.

In a service-provider network, a trusted interface is connected to a port on a device in the same network. An untrusted interface is connected to an untrusted interface in the network or to an interface on a device that is not in the network.

When a switch receives a packet on an untrusted interface and the interface belongs to a VLAN in which DHCP snooping is enabled, the switch compares the source MAC address and the DHCP client hardware address. If the addresses match (the default), the switch forwards the packet. If the addresses do not match, the switch drops the packet.

The switch drops a DHCP packet when one of these situations occurs:

- A packet from a DHCP server, such as a DHCPOFFER, DHCPACK, DHCPNAK, or DHCPLEASEQUERY packet, is received from outside the network or firewall.
- A packet is received on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match.
- The switch receives a DHCPRELEASE or DHCPDECLINE broadcast message that has a MAC address in the DHCP snooping binding database, but the interface information in the binding database does not match the interface on which the message was received.
- A DHCP relay agent forwards a DHCP packet that includes a relay-agent IP address that is not 0.0.0.0, or the relay agent forwards a packet that includes option-82 information to an untrusted port.

If the switch is an aggregation switch supporting DHCP snooping and is connected to an edge switch that is inserting DHCP option-82 information, the switch drops packets with option-82 information when packets are received on an untrusted interface. If DHCP snooping is enabled and packets are received on a trusted port, the aggregation switch does not learn the DHCP snooping bindings for connected devices and cannot build a complete DHCP snooping binding database.

When an aggregation switch can be connected to an edge switch through an untrusted interface and you enter the **ip dhcp snooping information option allowed-trust** global configuration command, the aggregation switch accepts packets with option-82 information from the edge switch. The aggregation switch learns the bindings for hosts connected through an untrusted switch interface. The DHCP security features, such as dynamic ARP inspection or IP source guard, can still be enabled on the aggregation switch while the switch receives packets with option-82 information on ingress untrusted interfaces to which hosts are connected. The port on the edge switch that connects to the aggregation switch must be configured as a trusted interface.

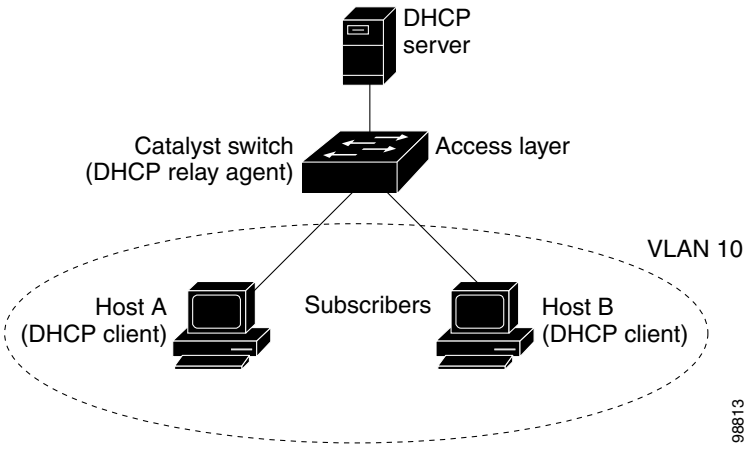
Option-82 Data Insertion

In residential, metropolitan Ethernet-access environments, DHCP can centrally manage the IP address assignments for a large number of subscribers. When the DHCP option-82 feature is enabled on the switch, a subscriber device is identified by the switch port through which it connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can be connected to the same port on the access switch and are uniquely identified.

Note The DHCP option-82 feature is supported only when DHCP snooping is globally enabled and on the VLANs to which subscriber devices using this feature are assigned.

Figure 4-1 is an example of a metropolitan Ethernet network in which a centralized DHCP server assigns IP addresses to subscribers connected to the switch at the access layer. Because the DHCP clients and their associated DHCP server do not reside on the same IP network or subnet, a DHCP relay agent (the Catalyst switch) is configured with a helper address to enable broadcast forwarding and to transfer DHCP messages between the clients and the server.

Figure 4-1 DHCP Relay Agent in a Metropolitan Ethernet Network



When you enable the DHCP snooping information option-82 on the switch, this sequence of events occurs:

- The host (DHCP client) generates a DHCP request and broadcasts it on the network.
- When the switch receives the DHCP request, it adds the option-82 information in the packet. By default, the remote-ID suboption is the switch MAC address, and the circuit-ID suboption is the port identifier, **vlan-mod-port**, from which the packet is received. You can also configure the remote ID and circuit ID. For information on configuring these suboptions, see the "Enabling DHCP Snooping and Option 82" section on page 4-16.
- If the IP address of the relay agent is configured, the switch adds this IP address in the DHCP packet.
- The switch forwards the DHCP request that includes the option-82 field to the DHCP server.
- The DHCP server receives the packet. If the server is option-82-capable, it can use the remote ID, the circuit ID, or both to assign IP addresses and implement policies, such as restricting the number of IP addresses that can be assigned to a single remote ID or circuit ID. Then the DHCP server echoes the option-82 field in the DHCP reply.
- The DHCP server unicasts the reply to the switch if the request was relayed to the server by the switch. The switch verifies that it originally inserted the option-82 data by inspecting the remote ID and possibly the circuit ID fields. The switch removes the option-82 field and forwards the packet to the switch port that connects to the DHCP client that sent the DHCP request.

In the default suboption configuration, when the described sequence of events occurs, the values in these fields in Figure 4-2 do not change:

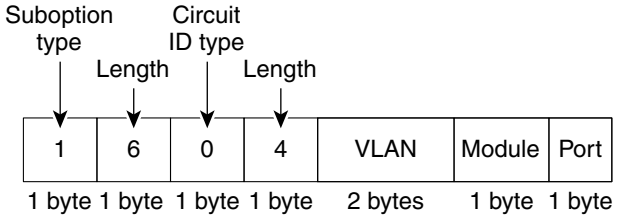
- Circuit ID suboption fields
  - Suboption type
  - Length of the suboption type
  - Circuit ID type
  - Length of the circuit ID type
- Remote ID suboption fields
  - Suboption type
  - Length of the suboption type
  - Remote ID type
  - Length of the circuit ID type

In the port field of the circuit ID suboption, the port numbers start at 3. For example, on a switch with 24 10/100 ports and small form-factor pluggable (SFP) module slots, port 3 is the Fast Ethernet 0/1 port, port 4 is the Fast Ethernet 0/2 port, and so forth. Port 27 is the SFP module slot 0/1, and so forth.

Figure 4-2 shows the packet formats for the remote ID suboption and the circuit ID suboption when the default suboption configuration is used. The switch uses the packet formats when DHCP snooping is globally enabled and when the **ip dhcp snooping information option** global configuration command is entered.

Figure 4-2 Suboption Packet Formats

Circuit ID Suboption Frame Format



Remote ID Suboption Frame Format

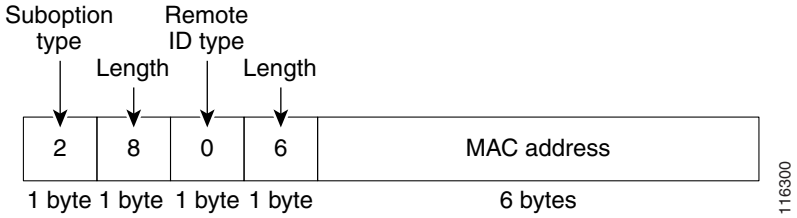


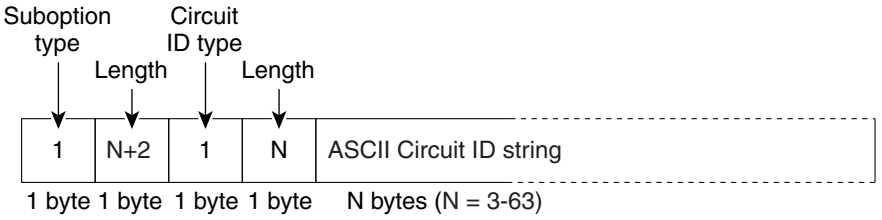
Figure 4-3 shows the packet formats for user-configured remote ID and circuit ID suboptions The switch uses these packet formats when you globally enable DHCP snooping and enter the **ip dhcp snooping information option format remote-id** global configuration command **and the ip dhcp snooping vlan information option format-type circuit-id string** interface configuration command.

The values for these fields in the packets change from the default values when you configure the remote-ID and circuit-ID suboptions:

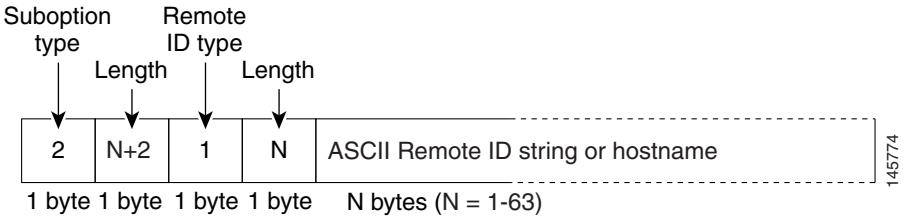
- Circuit-ID suboption fields
  - The circuit-ID type is 1.
  - The length values are variable, depending on the length of the string that you configure.
- Remote-ID suboption fields
  - The remote-ID type is 1.
  - The length values are variable, depending on the length of the string that you configure.

Figure 4-3 User-Configured Suboption Packet Formats

Circuit ID Suboption Frame Format (for user-configured string):



Remote ID Suboption Frame Format (for user-configured string):



Cisco IOS DHCP Server Database

During the DHCP-based autoconfiguration process, the designated DHCP server uses the Cisco IOS DHCP server database. It has IP addresses, *address bindings*, and configuration parameters, such as the boot file.

An address binding is a mapping between an IP address and a MAC address of a host in the Cisco IOS DHCP server database. You can manually assign the client IP address, or the DHCP server can allocate an IP address from a DHCP address pool. For more information about manual and automatic address bindings, see the "Related Documents" section on page 4-25.

DHCP Snooping Binding Database

When DHCP snooping is enabled, the switch uses the DHCP snooping binding database to store information about untrusted interfaces. The database can have up to 8192 bindings.

Each database entry (*binding*) has an IP address, an associated MAC address, the lease time (in hexadecimal format), the interface to which the binding applies, and the VLAN to which the interface belongs. The database agent stores the bindings in a file at a configured location. At the end of each entry is a *checksum* value that accounts for all the bytes associated with the entry. Each entry is 72 bytes, followed by a space and then the checksum value.

To keep the bindings when the switch reloads, you must use the DHCP snooping database agent. If the agent is disabled, dynamic ARP inspection or IP source guard is enabled, and the DHCP snooping binding database has dynamic bindings, the switch loses its connectivity. If the agent is disabled and only DHCP snooping is enabled, the switch does not lose its connectivity, but DHCP snooping might not prevent DHCP spoofing attacks.

When reloading, the switch reads the binding file to build the DHCP snooping binding database. The switch keeps the file current by updating it when the database changes.

When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the binding file. The frequency at which the file is updated is based on a configurable delay, and the updates are batched. If the file is not updated in a specified time (set by the write-delay and abort-timeout values), the update stops.

This is the format of the file that has the bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-..-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The *initial-checksum* entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Fa1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Fa1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Fa1/0/4 584a38f0
END
```

When the switch starts and the calculated checksum value equals the stored checksum value, the switch reads entries from the binding file and adds the bindings to its DHCP snooping binding database. The switch ignores an entry when one of these situations occurs:

- The switch reads the entry and the calculated checksum value does not equal the stored checksum value. The entry and the ones following it are ignored.
- An entry has an expired lease time (the switch might not remove a binding entry when the lease time expires).
- The interface in the entry no longer exists on the system.
- The interface is a routed interface or a DHCP snooping-trusted interface.

Information About DHCP Server Port-Based Address Allocation

DHCP server port-based address allocation is a feature that enables DHCP to maintain the same IP address on an Ethernet switch port regardless of the attached device client identifier or client hardware address.

When Ethernet switches are deployed in the network, they offer connectivity to the directly connected devices. In some environments, such as on a factory floor, if a device fails, the replacement device must be working immediately in the existing network. With the current DHCP implementation, there is no guarantee that DHCP would offer the same IP address to the replacement device. Control, monitoring, and other software expect a stable IP address associated with each device. If a device is replaced, the address assignment should remain stable even though the DHCP client has changed.

When configured, the DHCP server port-based address allocation feature ensures that the same IP address is always offered to the same connected port even as the client identifier or client hardware address changes in the DHCP messages received on that port. The DHCP protocol recognizes DHCP clients by the client identifier option in the DHCP packet. Clients that do not include the client identifier option are identified by the client hardware address. When you configure this feature, the port name of the interface overrides the client identifier or hardware address and the actual point of connection, the switch port, becomes the client identifier.

In all cases, by connecting the Ethernet cable to the same port, the same IP address is allocated through DHCP to the attached device.

The DHCP server port-based address allocation feature is only supported on a Cisco IOS DHCP server and not a third-party server.

Information About IP Source Guard

IPSG is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering traffic based on the DHCP snooping binding database and on manually configured IP source bindings. You can use IP source guard to prevent traffic attacks if a host tries to use the IP address of its neighbor.

You can enable IP source guard when DHCP snooping is enabled on an untrusted interface. After IPSG is enabled on an interface, the switch blocks all IP traffic received on the interface except for DHCP packets allowed by DHCP snooping. A port access control list (ACL) is applied to the interface. The port ACL allows only IP traffic with a source IP address in the IP source binding table and denies all other traffic.

Note The port ACL takes precedence over any VLAN maps that affect the same interface.

The IP source binding table has bindings that are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address, its associated MAC address, and its associated VLAN number. The switch uses the IP source binding table only when IP source guard is enabled.

IP source guard is supported only on Layer 2 ports, including access and trunk ports.You can configure IP source guard with source IP address filtering or with source IP and MAC address filtering.

This section includes the following topics:

- Source IP Address Filtering, page 4-8
- Source IP and MAC Address Filtering, page 4-9
- IP Source Guard for Static Hosts, page 4-9

Source IP Address Filtering

When IP source guard is enabled with this option, IP traffic is filtered based on the source IP address. The switch forwards IP traffic when the source IP address matches an entry in the DHCP snooping binding database or a binding in the IP source binding table.

When a DHCP snooping binding or static IP source binding is added, changed, or deleted on an interface, the switch modifies the port ACL using the IP source binding changes, and re-applies the port ACL to the interface.

If you enable IP source guard on an interface on which IP source bindings are not configured (dynamically learned by DHCP snooping or manually configured), the switch creates and applies a port ACL that denies all IP traffic on the interface. If you disable IP source guard, the switch removes the port ACL from the interface.

Source IP and MAC Address Filtering

When IP source guard is enabled with this option, IP traffic is filtered based on the source IP and MAC addresses. The switch forwards traffic only when the source IP and MAC addresses match an entry in the IP source binding table.

When IP source guard with source IP and MAC address filtering is enabled, the switch filters IP and non-IP traffic. If the source MAC address of an IP or non-IP packet matches a valid IP source binding, the switch forwards the packet. The switch drops all other types of packets except DHCP packets.

The switch uses port security to filter source MAC addresses. The interface can shut down when a port-security violation occurs.

IP Source Guard for Static Hosts

Note Do not use IPSG (IP source guard) for static hosts on uplink ports or trunk ports.

IPSG for static hosts extends the IPSG capability to non-DHCP and static environments. The previous IPSG used the entries created by DHCP snooping to validate the hosts connected to a switch. Any traffic received from a host without a valid DHCP binding entry is dropped. This security feature restricts IP traffic on nonrouted Layer 2 interfaces. It filters traffic based on the DHCP snooping binding database and on manually configured IP source bindings. The previous version of IPSG required a DHCP environment for IPSG to work.

IPSG for static hosts allows IPSG to work without DHCP. IPSG for static hosts relies on IP device tracking-table entries to install port ACLs. The switch creates static entries based on ARP requests or other IP packets to maintain the list of valid hosts for a given port. You can also specify the number of hosts allowed to send traffic to a given port. This is equivalent to port security at Layer 3.

IPSG for static hosts also supports dynamic hosts. If a dynamic host receives a DHCP-assigned IP address that is available in the IP DHCP snooping table, the same entry is learned by the IP device tracking table. When you enter the **show ip device tracking all** EXEC command, the IP device tracking table displays the entries as ACTIVE.

Note Some IP hosts with multiple network interfaces can inject some invalid packets into a network interface. The invalid packets contain the IP or MAC address for another network interface of the host as the source address. The invalid packets can cause IPSG for static hosts to connect to the host, to learn the invalid IP or MAC address bindings, and to reject the valid bindings. Consult the vendor of the corresponding operating system and the network interface to prevent the host from injecting invalid packets.

IPSG for static hosts initially learns IP or MAC bindings dynamically through an ACL-based snooping mechanism. IP or MAC bindings are learned from static hosts by ARP and IP packets. They are stored in the device tracking database. When the number of IP addresses that have been dynamically learned or statically configured on a given port reaches a maximum, the hardware drops any packet with a new IP

address. To resolve hosts that have moved or gone away for any reason, IPSG for static hosts leverages IP device tracking to age out dynamically learned IP address bindings. This feature can be used with DHCP snooping. Multiple bindings are established on a port that is connected to both DHCP and static hosts. For example, bindings are stored in both the device tracking database as well as in the DHCP snooping binding database.

Prerequisites

DHCP Server

- Before you configure DHCP features and IP source guard, you must understand the concepts documented in the "Information About DHCP Features" section on page 4-1, "Information About DHCP Server Port-Based Address Allocation" section on page 4-7, and "Information About IP Source Guard" section on page 4-8.
- The Cisco DHCP server and the relay agent are enabled by default. Use the **no service dhcp** command to disable the Cisco DHCP server and the relay agent and the **service dhcp** command to reenable the functionality.
- Port 67 (the DHCP server port) is closed in the Cisco DHCP/BOOTP default configuration. There are two logical parts to the **service dhcp** command: service enabled and service running. The DHCP service is enabled by default, but port 67 does not open until the DHCP service is running. If the DHCP service is running, the **show ip sockets details** or the **show sockets detail** command displays port 67 as open.

DHCP Relay Agent

- You must enable the Cisco DHCP relay agent on an interface by using the **ip helper-address** command. This command enables a DHCP broadcast to be forwarded to the configured DHCP server.

Guidelines and Limitations

- DHCP Snooping, page 4-10
- Port-Based Address Allocation, page 4-11
- IP Source Guard, page 4-12

DHCP Snooping

These are the configuration guidelines for DHCP snooping:

- You must globally enable DHCP snooping on the switch.
- DHCP snooping is not active until DHCP snooping is enabled on a VLAN.
- Before globally enabling DHCP snooping on the switch, make sure that the devices acting as the DHCP server and the DHCP relay agent are configured and enabled.
- When you globally enable DHCP snooping on the switch, these Cisco IOS commands are not available until snooping is disabled. If you enter these commands, the switch returns an error message, and the configuration is not applied.
  - **ip dhcp relay information check** global configuration command

  - **ip dhcp relay information policy** global configuration command
  - **ip dhcp relay information trust-all** global configuration command
  - **ip dhcp relay information trusted** interface configuration command
- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. For example, you must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.
- When configuring a large number of circuit IDs on a switch, consider the impact of lengthy character strings on the NVRAM or the flash memory. If the circuit-ID configurations, combined with other data, exceed the capacity of the NVRAM or the flash memory, an error message appears.
- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. For example, you must specify the IP addresses that the DHCP server can assign or exclude, configure DHCP options for devices, or set up the DHCP database agent.
- If the DHCP relay agent is enabled but DHCP snooping is disabled, the DHCP option-82 data insertion feature is not supported.
- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust** interface configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.
- Follow these guidelines when configuring the DHCP snooping binding database:
  - Because both the NVRAM and the flash memory have limited storage capacity, we recommend that you store the binding file on a TFTP server.
  - For network-based URLs (such as TFTP and FTP), you must create an empty file at the configured URL before the switch can write bindings to the binding file at that URL. See the documentation for your TFTP server to determine whether you must first create an empty file on the server; some TFTP servers cannot be configured this way.
  - To ensure that the lease time in the database is accurate, we recommend that NTP be enabled and configured. For more information, see the "Administering the Switch" chapter in the *Cisco Connected Grid Switches System Management Software Configuration Guide*.
  - If NTP is configured, the switch writes binding changes to the binding file only when the switch system clock is synchronized with NTP.
- Do not enter the **ip dhcp snooping information option allowed-untrusted** command on an aggregation switch to which an untrusted device is connected. If you enter this command, an untrusted device might spoof the option-82 information.
- You can display DHCP snooping statistics by entering the **show ip dhcp snooping statistics** user EXEC command, and you can clear the snooping statistics counters by entering the **clear ip dhcp snooping statistics** privileged EXEC command.

Note Do not enable Dynamic Host Configuration Protocol (DHCP) snooping on RSPAN VLANs. If DHCP snooping is enabled on RSPAN VLANs, DHCP packets might not reach the RSPAN destination port.

Port-Based Address Allocation

These are the configuration guidelines for DHCP port-based address allocation:

- Only one IP address can be assigned per port.

- Reserved addresses (preassigned) cannot be cleared by using the **clear ip dhcp binding** global configuration command.
- Preassigned addresses are automatically excluded from normal dynamic IP address assignment. Preassigned addresses cannot be used in host pools, but there can be multiple preassigned addresses per DHCP address pool.
- To restrict assignments from the DHCP pool to preconfigured reservations (unreserved addresses are not offered to the client and other clients are not served by the pool), you can enter the **reserved-only** DHCP pool configuration command.

IP Source Guard

These are the configuration guidelines for IP source guard:

- You can configure static IP bindings only on nonrouted ports. If you enter the **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* global configuration command on a routed interface, this error message appears:

  ```
  Static IP source binding can only be configured on switch port.
  ```

- When IP source guard with source IP filtering is enabled on an interface, DHCP snooping must be enabled on the access VLAN to which the interface belongs.
- If you are enabling IP source guard on a trunk interface with multiple VLANs and DHCP snooping is enabled on all the VLANs, the source IP address filter is applied on all the VLANs.

  Note If IP source guard is enabled and you enable or disable DHCP snooping on a VLAN on the trunk interface, the switch might not properly filter traffic.

- If you enable IP source guard with source IP and MAC address filtering, DHCP snooping and port security must be enabled on the interface. You must also enter the **ip dhcp snooping information option** global configuration command and ensure that the DHCP server supports option 82. When IP source guard is enabled with MAC address filtering, the DHCP host MAC address is not learned until the host is granted a lease. When forwarding packets from the server to the host, DHCP snooping uses option-82 data to identify the host port.
- When configuring IP source guard on interfaces on which a private VLAN is configured, port security is not supported.
- IP source guard is not supported on EtherChannels.
- You can enable this feature when IEEE 802.1x port-based authentication is enabled.
- If the number of ternary content addressable memory (TCAM) entries exceeds the maximum available, the CPU usage increases.

Default Settings

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software; requires configuration. **Note** The switch responds to DHCP requests only if it is configured as a DHCP server.
DHCP relay agent	Enabled. **Note** The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.
DHCP packet forwarding address	None configured.
Checking the relay agent information	Enabled (invalid messages are dropped). **Note** The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.
DHCP relay agent forwarding policy	Replace the existing relay agent information. **Note** The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.
DHCP snooping enabled globally	Disabled.
DHCP snooping information option	Enabled.
DHCP snooping option to accept packets on untrusted ingress interfaces	Disabled. **Note** Use this feature when the switch is an aggregation switch that receives packets with option-82 information from an edge switch.
DHCP snooping limit rate	None configured.
DHCP snooping trust	Untrusted.
DHCP snooping VLAN	Disabled.
DHCP snooping MAC address verification	Enabled.
Cisco IOS DHCP server binding database	Enabled in Cisco IOS software; requires configuration. **Note** The switch gets network addresses and configuration parameters only from a device configured as a DHCP server.
DHCP snooping binding database agent	Enabled in Cisco IOS software; requires configuration. This feature is operational only when a destination is configured.
DHCP server port-based address allocation	Disabled.
IP source guard	Disabled.

Configuring DHCP Features

- Configuring the DHCP Server, page 4-14
- Configuring the DHCP Relay Agent, page 4-14
- Specifying the Packet Forwarding Address, page 4-15
- Enabling DHCP Snooping and Option 82, page 4-16
- Enabling DHCP Snooping on Private VLANs, page 4-18
- Enabling the Cisco IOS DHCP Server Database, page 4-18
- Enabling the DHCP Snooping Binding Database Agent, page 4-18

Configuring the DHCP Server

The switch can act as a DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your switch but are not configured. These features are not operational.

For procedures to configure the switch as a DHCP server, see the "Related Documents" section on page 4-25.

Configuring the DHCP Relay Agent

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**service dhcp**	Enable the DHCP relay agent on your switch. By default, this feature is enabled.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the DHCP relay agent, use the **no service dhcp** global configuration command.

EXAMPLE

The following example shows to enable DHCP services on the DHCP server:

```
Switch(config)# service dhcp
```

Specifying the Packet Forwarding Address

If the DHCP server and the DHCP clients are on different networks or subnets and the switch is running the IP services image, you must configure the switch with the **ip helper-address** *address* interface configuration command. The general rule is to configure the command on the Layer 3 interface closest to the client. The address used in the **ip helper-address** command can be a specific DHCP server IP address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables any DHCP server to respond to requests.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface vlan** *vlan-id*	Create a switch virtual interface by entering a VLAN ID, and enter interface configuration mode.
Step 3	**ip address** *ip-address subnet-mask*	Configure the interface with an IP address and an IP subnet.
Step 4	**ip helper-address** *address*	Specify the DHCP packet forwarding address. The helper address can be a specific DHCP server address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables other servers to respond to DHCP requests. If you have multiple servers, you can configure one helper address for each server.
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface range** *port-range* or **interface** *interface-id*	Configure multiple physical ports that are connected to the DHCP clients, and enter interface range configuration mode. or Configure a single physical port that is connected to the DHCP client, and enter interface configuration mode.
Step 7	**no shutdown**	Enable the interface(s), if necessary. By default, user network interfaces (UNIs) and enhanced network interfaces (ENIs) are disabled and network node interfaces (NNIs) are enabled.
Step 8	**switchport mode access**	Define the VLAN membership mode for the port.
Step 9	**switchport access vlan** *vlan-id*	Assign the ports to the same VLAN as configured in Step 2.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show running-config**	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the DHCP packet forwarding address, use the **no ip helper-address** *address* interface configuration command.

EXAMPLE

```
Switch(config)# interface vlan 10
Switch(config-if)# ip address 10.108.1.27 255.255.255.0
Switch(config-if)# ip helper-address 10.24.43.2
Switch(config-if)# exit
Switch(config)# interface range gigabitethernet 1/1 - 6
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# end
```

Enabling DHCP Snooping and Option 82

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp snooping**	Enable DHCP snooping globally.
Step 3	**ip dhcp snooping vlan** *vlan-range*	Enable DHCP snooping on a VLAN or range of VLANs. The range is 1 to 4094. You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space.
Step 4	**ip dhcp snooping information option**	Enable the switch to insert and remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. This is the default setting.
Step 5	**ip dhcp snooping information option format remote-id [string** *ASCII-string* **| hostname]**	(Optional) Configure the remote-ID suboption. You can configure the remote ID to be: • String of up to 63 ASCII characters (no spaces) • Configured hostname for the switch **Note** If the hostname is longer than 63 characters, it is truncated to 63 characters in the remote-ID configuration. The default remote ID is the switch MAC address.

	Command	Purpose
Step 6	**ip dhcp snooping information option allowed-untrusted**	(Optional) If the switch is an aggregation switch connected to an edge switch, enable the switch to accept incoming DHCP snooping packets with option-82 information from the edge switch. The default is disabled. **Note** You must enter this command only on aggregation switches that are connected to trusted devices.
Step 7	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 8	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
Step 9	**ip dhcp snooping vlan** *vlan* **information option format-type circuit-id string [override]** *ASCII-string*	(Optional) Configure the circuit-ID suboption for the specified interface. Specify the VLAN and port identifier, using a VLAN ID in the range of 1 to 4094. The default circuit ID is the port identifier, in the format **vlan-mod-port**. You can configure the circuit ID to be a string of 3 to 63 ASCII characters (no spaces). (Optional) Use the **override** keyword when you do not want the circuit-ID suboption inserted in TLV format to define subscriber information.
Step 10	**ip dhcp snooping trust**	(Optional) Configure the interface as trusted or untrusted. You can use the **no** keyword to configure an interface to receive messages from an untrusted client. The default is untrusted.
Step 11	**ip dhcp snooping limit rate** *rate*	(Optional) Configure the number of DHCP packets per second that an interface can receive. The range is 1 to 2048. By default, no rate limit is configured. **Note** We recommend an untrusted rate limit of not more than 100 packets per second. If you configure rate limiting for trusted interfaces, you might need to increase the rate limit if the port is a trunk port assigned to more than one VLAN on which DHCP snooping is enabled.
Step 12	**exit**	Return to global configuration mode.
Step 13	**ip dhcp snooping verify mac-address**	(Optional) Configure the switch to verify that the source MAC address in a DHCP packet that is received on untrusted ports matches the client hardware address in the packet. The default is to verify that the source MAC address matches the client hardware address in the packet.
Step 14	**end**	Return to privileged EXEC mode.
Step 15	**show running-config**	Verify your entries.
Step 16	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable DHCP snooping, use the **no ip dhcp snooping** global configuration command. To disable DHCP snooping on a VLAN or range of VLANs, use the **no ip dhcp snooping vlan** *vlan-range* global configuration command. To disable the insertion and removal of the option-82 field, use the **no ip dhcp snooping information option** global configuration command. To configure an aggregation switch to drop incoming DHCP snooping packets with option-82 information from an edge switch, use the **no ip dhcp snooping information option allowed-untrusted** global configuration command.

EXAMPLE

This example shows how to enable DHCP snooping globally and on VLAN 10 and to configure a rate limit of 100 packets per second on a port:

```
Switch(config)# ip dhcp snooping
Switch(config)# ip dhcp snooping vlan 10
Switch(config)# ip dhcp snooping information option
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip dhcp snooping limit rate 100
```

Enabling DHCP Snooping on Private VLANs

You can enable DHCP snooping on private VLANs. If DHCP snooping is enabled, the configuration is propagated to both a primary VLAN and its associated secondary VLANs. If DHCP snooping is enabled on the primary VLAN, it is also configured on the secondary VLANs.

If DHCP snooping is already configured on the primary VLAN and you configure DHCP snooping with different settings on a secondary VLAN, the configuration for the secondary VLAN does not take effect. You must configure DHCP snooping on the primary VLAN. If DHCP snooping is not configured on the primary VLAN, this message appears when you are configuring DHCP snooping on the secondary VLAN, such as VLAN 200:

```
2w5d:%DHCP_SNOOPING-4-DHCP_SNOOPING_PVLAN_WARNING:DHCP Snooping configuration may not take
effect on secondary vlan 200. DHCP Snooping configuration on secondary vlan is derived
from its primary vlan.
```

The **show ip dhcp snooping** privileged EXEC command output shows all VLANs, including primary and secondary private VLANs, on which DHCP snooping is enabled.

Enabling the Cisco IOS DHCP Server Database

For procedures, refer to the "Related Documents" section on page 4-25.

Enabling the DHCP Snooping Binding Database Agent

BEFORE YOU BEGIN

You must enable DHCP snooping on the interface before entering this command as described in the "Enabling DHCP Snooping and Option 82" procedure on page 4-16.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp snooping database** {**flash:**/*filename* | **ftp://***user*:*password*@*host*/*filename* | **http://**[[*username*:*password*]@]{*hostname* | *host-ip*}[/*directory*] /*image-name*.**tar** | **rcp://***user*@*host*/*filename*}| **tftp://***host*/*filename*	Specify the URL for the database agent or the binding file by using one of these forms: • **flash:**/*filename* • **ftp://***user*:*password*@*host*/*filename* • **http://**[[*username*:*password*]@]{*hostname* | *host-ip*}[/*directory*] /*image-name*.**tar** • **rcp://***user*@*host*/*filename* • **tftp://***host*/*filename*
Step 3	**ip dhcp snooping database timeout** *seconds*	Specify when to stop the database transfer process after the binding database changes. The range is from 0 to 86400. Use 0 for an infinite duration. The default is 300 seconds (5 minutes).
Step 4	**ip dhcp snooping database write-delay** *seconds*	Specify the duration for which the transfer should be delayed after the binding database changes. The range is from 15 to 86400 seconds. The default is 300 seconds (5 minutes).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*	(Optional) Add binding entries to the DHCP snooping binding database. The *vlan-id* range is from 1 to 4904. The *seconds* range is from 1 to 4294967295. Enter this command for each entry that you add. **Note** Use this command when you are testing or debugging the switch.
Step 7	**show ip dhcp snooping database** [**detail**]	Display the status and statistics of the DHCP snooping binding database agent.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To stop using the database agent and binding files, use the **no ip dhcp snooping database** global configuration command. To reset the timeout or delay values, use the **ip dhcp snooping database timeout** *seconds* or the **ip dhcp snooping database write-delay** *seconds* global configuration command.

To clear the statistics of the DHCP snooping binding database agent, use the **clear ip dhcp snooping database statistics** privileged EXEC command. To renew the database, use the **renew ip dhcp snooping database** privileged EXEC command.

To delete binding entries from the DHCP snooping binding database, use the **no ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* privileged EXEC command. Enter this command for each entry that you delete.

EXAMPLE

This example shows how to specify the database URL using TFTP:

```
Switch(config)# ip dhcp snooping database tftp://10.90.90.90/snooping-rp2
```

This example shows how to specify the amount of time before writing DHCP snooping entries to an external server:

```
Switch(config)# ip dhcp snooping database write-delay 15
```

Configuring DHCP Server Port-Based Address Allocation

- Enabling Port-Based Address Allocation, page 4-20
- Preassigning an IP Address and Associating it to a Client, page 4-21

Enabling Port-Based Address Allocation

Follow this procedure to globally enable port-based address allocation and to automatically generate a subscriber identifier on an interface.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp use subscriber-id client-id**	Configure the DHCP server to globally use the subscriber identifier as the client identifier on all incoming DHCP messages.
Step 3	**ip dhcp subscriber-id interface-name**	Automatically generate a subscriber identifier based on the short name of the interface. A subscriber identifier configured on a specific interface takes precedence over this command.
Step 4	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 5	**ip dhcp server use subscriber-id client-id**	Configure the DHCP server to use the subscriber identifier as the client identifier on all incoming DHCP messages on the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

In the following example, the DHCP server will ignore any client identifier fields in the DHCP messages and use the subscriber ID as the client identifier. The DHCP server uses the subscriber identifier as the client identifier for all incoming messages received on Ethernet interface 0/0.

```
Switch(config)# ip dhcp use subscriber-id client-id
Switch(config)# ip dhcp subscriber-id interface-name
Switch(config)# interface Ethernet 0/0
Switch(config-if)# ip dhcp server use subscriber-id client-id
Switch(config)# end
```

Preassigning an IP Address and Associating it to a Client

After enabling DHCP port-based address allocation on the switch, use the **ip dhcp pool** global configuration command to preassign IP addresses and to associate them to clients. To restrict assignments from the DHCP pool to preconfigured reservations, you can enter the **reserved-only** DHCP pool configuration command. Unreserved addresses that are part of the network or on pool ranges are not offered to the client, and other clients are not served by the pool. By entering this command, users can configure a group of switches with DHCP pools that share a common IP subnet and that ignore requests from clients of other switches.

BEFORE YOU BEGIN

Review the "Guidelines and Limitations" section on page 4-10 for port-based address allocation configuration guidelines.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp pool** *poolname*	Enter DHCP pool configuration mode, and define the name for the DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0).
Step 3	**network** *network-number* [*mask* | */prefix-length*]	Specify the subnet network number and mask of the DHCP address pool.
Step 4	**address** *ip-address* **client-id** *string* *[***ascii***]*	Reserve an IP address for a DHCP client identified by the interface name. *string*—can be an ASCII value or a hexadecimal value.
Step 5	**reserved-only**	(Optional) Use only reserved addresses in the DHCP address pool. The default is to not restrict pool addresses.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip dhcp pool**	Verify DHCP pool configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable DHCP port-based address allocation, use the **no ip dhcp use subscriber-id client-id** global configuration command. To disable the automatic generation of a subscriber identifier, use the **no ip dhcp subscriber-id interface-name** global configuration command. To disable the subscriber identifier on an interface, use the **no ip dhcp server use subscriber-id client-id** interface configuration command.

To remove an IP address reservation from a DHCP pool, use the **no address** *ip-address* **client-id** *string* DHCP pool configuration command. To change the address pool to nonrestricted, enter the **no reserved-only** DHCP pool configuration command.

EXAMPLE

In this example, a subscriber identifier is automatically generated, and the DHCP server ignores any client identifier fields in the DHCP messages and uses the subscriber identifier instead. The subscriber identifier is based on the short name of the interface and the client preassigned IP address 10.1.1.7.

```
IE-2000U-4T# show running config

Building configuration...

Current configuration : 3018 bytes
!
! Last configuration change at 02:55:14 UTC Mon Sept 1 2013
!
version 15.0
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname IE-2000U-4T
!
boot-start-marker
boot-end-marker
!
!
!
no aaa new-model
system mtu routing 1998
ip routing
no ip domain-lookup
ip name-server 69.78.134.231
!
ip dhcp relay information policy keep
ip dhcp relay information trust-all
!
ip dhcp pool test1
 network 192.168.10.0 255.255.255.0
 default-router 192.168.10.1
 dns-server 69.78.134.231

<output truncated>
```

This example shows that the preassigned address was correctly reserved in the DHCP pool:

```
IE-2000U-4T# show ip dhcp pool dhcppool

Pool test1 :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254
 Leased addresses               : 1
 Excluded addresses             : 0
```

```
Pending event                  : none
1 subnet is currently in the pool :
Current index        IP address range                    Leased/Excluded/Total
192.168.10.4         192.168.10.1     - 192.168.10.254    1     / 0     / 254
```

Configuring IP Source Guard

Default IP Source Guard Configuration

By default, IP source guard is disabled.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
Step 4	**ip verify source** or **ip verify source port-security**	Enable IP source guard with source IP address filtering. Enable IP source guard with source IP and MAC address filtering. **Note** When you enable both IP Source Guard and Port Security by using the **ip verify source port-security** interface configuration command, there are two caveats: • The DHCP server must support option 82, or the client is not assigned an IP address. • The MAC address in the DHCP packet is not learned as a secure address. The MAC address of the DHCP client is learned as a secure address only when the switch receives non-DHCP data traffic.
Step 5	**exit**	Return to global configuration mode.
Step 6	**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **inteface** *interface-id*	Add a static IP source binding. Enter this command for each static binding.
Step 7	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 8	**show ip verify source** [**interface** *interface-id*]	Display the IP source guard configuration for all interfaces or for a specific interface.
Step 9	**show ip source binding** [*ip-address*] [*mac-address*] [**dhcp-snooping** | **static**] [**inteface** *interface-id*] [**vlan** *vlan-id*]	Display the IP source bindings on the switch, on a specific VLAN, or on a specific interface.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IP source guard with source IP address filtering, use the **no ip verify source** interface configuration command.

To delete a static IP source binding entry, use the **no ip source** global configuration command.

EXAMPLE

This example shows how to enable IP source guard with source IP and MAC filtering on VLANs 10 and 11:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip verify source port-security
Switch(config-if)# exit
Switch(config)# ip source binding 0100.0022.0010 vlan 10 10.0.0.2 interface
gigabitethernet0/1
Switch(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface
gigabitethernet0/1
Switch(config)# end
```

Verifying Configuration

Command	Purpose
DHCP Snooping	
show ip dhcp snooping	Displays the DHCP snooping configuration for a switch
show ip dhcp snooping binding	Displays only the dynamically configured bindings in the DHCP snooping binding database, also referred to as a binding table. **Note** If DHCP snooping is enabled and an interface changes to the down state, the switch does not delete the manually configured bindings.
show ip dhcp snooping database	Displays the DHCP snooping binding database status and statistics.
show ip dhcp snooping statistics	Displays the DHCP snooping statistics in summary or detail form.
show ip source binding	Display the dynamically and statically configured bindings.
DHCP Server Port-Based Address Allocation	
show interface *interface id*	Display the status and configuration of a specific interface.
show ip dhcp pool	Display the DHCP address pools.

Command	Purpose
show ip dhcp binding	Display address bindings on the Cisco IOS DHCP server.
IP Source Guard	
show ip source binding	Display the IP source bindings on a switch.
show ip verify source	Display the IP source guard configuration on the switch.

Related Documents

- Cisco IOS Master Command List, All Releases
- Cisco IOS IP Addressing Services Command Reference
- IP Addressing: DHCP Configuration Guide, Cisco IOS Release 15M&T

  You can find the following information in the above referenced guide:
  - DHCP clients
  - Manual and automatic address bindings
  - Configuring the switch as a DHCP server
  - Enabling and configuring the DHCP server database
  - Validating the DHCP relay agent information and configuring the relay agent forwarding policy

Configuring Dynamic ARP Inspection

This chapter describes how to configure dynamic Address Resolution Protocol inspection (dynamic ARP inspection) on the Cisco Industrial Ethernet 2000U Series Switches (IE 2000U).

This feature helps prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN.

Note For complete syntax and usage information for the commands used in this chapter, see the documents listed in the "Related Documents" section on page 5-18.

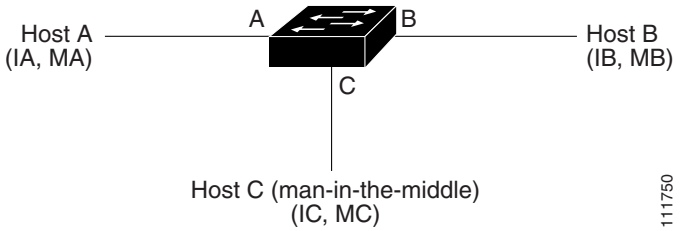
- Information About Dynamic ARP Inspection, page 5-1
- Prerequisites, page 5-5
- Guidelines and Limitations, page 5-5
- Default Settings, page 5-6
- Configuring Dynamic ARP Inspection, page 5-6
- Verifying Configuration, page 5-17
- Configuration Example, page 5-17
- Related Documents, page 5-18

Information About Dynamic ARP Inspection

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A but does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address. However, because ARP allows a gratuitous reply from a host even if an ARP request was not received, an ARP spoofing attack and the poisoning of ARP caches can occur. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

A malicious user can attack hosts, switches, and routers connected to your Layer 2 network by poisoning the ARP caches of systems connected to the subnet and by intercepting traffic intended for other hosts on the subnet. Figure 5-1 shows an example of ARP cache poisoning.

Figure 5-1 ARP Cache Poisoning



Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP request for the MAC address associated with IP address IB. When the switch and Host B receive the ARP request, they populate their ARP caches with an ARP binding for a host with the IP address IA and a MAC address MA; for example, IP address IA is bound to MAC address MA. When Host B responds, the switch and Host A populate their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can poison the ARP caches of the switch, Host A, and Host B by broadcasting forged ARP responses with bindings for a host with an IP address of IA (or IB) and a MAC address of MC. Hosts with poisoned ARP caches use the MAC address MC as the destination MAC address for traffic intended for IA or IB. This means that Host C intercepts that traffic. Because Host C knows the true MAC addresses associated with IA and IB, it can forward the intercepted traffic to those hosts by using the correct MAC address as the destination. Host C has inserted itself into the traffic stream from Host A to Host B, the classic *man-in-the middle* attack.

Dynamic ARP inspection is a security feature that validates ARP packets in a network. It intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

Dynamic ARP inspection ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

- Intercepts all ARP requests and responses on untrusted ports
- Verifies that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination
- Drops invalid ARP packets

Dynamic ARP inspection determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces, the switch forwards the packet only if it is valid.

You enable dynamic ARP inspection on a per-VLAN basis by using the **ip arp inspection vlan** *vlan-range* global configuration command. For configuration information, see the "Configuring Dynamic ARP Inspection in DHCP Environments" section on page 5-7.

In non-DHCP environments, dynamic ARP inspection can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses. You define an ARP ACL by using the **arp access-list** *acl-name* global configuration command. For configuration information, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 5-8. The switch logs dropped packets. For more information about the log buffer, see the "Logging of Dropped Packets" section on page 5-4.

Use the **ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]} global configuration command to configure dynamic ARP inspection to drop ARP packets when the IP addresses in the packets are invalid or when the MAC addresses in the body of the ARP packets do not match the addresses specified in the Ethernet header. For more information, see the "Performing Validation Checks" section on page 5-12.

Interface Trust States and Network Security

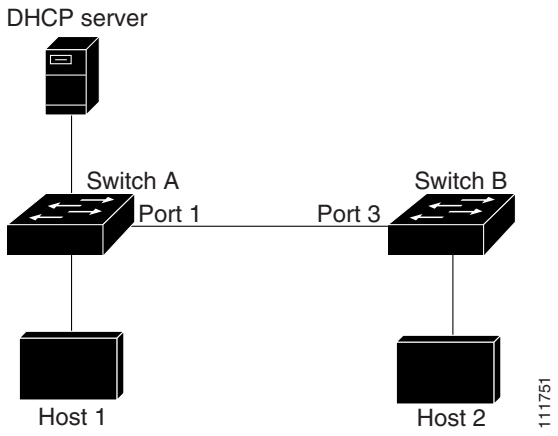
Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection validation process.

In a typical network configuration, you configure all switch ports connected to host ports as untrusted and configure all switch ports connected to switches as trusted. With this configuration, all ARP packets entering the network from a given switch bypass the security check. No other validation is needed at any other place in the VLAN or in the network. You configure the trust setting by using the **ip arp inspection trust** interface configuration command.

Caution Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

In Figure 5-2, assume that both Switch A and Switch B are running dynamic ARP inspection on the VLAN that includes Host 1 and Host 2. If Host 1 and Host 2 acquire their IP addresses from the DHCP server connected to Switch A, only Switch A binds the IP-to-MAC address of Host 1. Therefore, if the interface between Switch A and Switch B is untrusted, the ARP packets from Host 1 are dropped by Switch B. Connectivity between Host 1 and Host 2 is lost.

Figure 5-2 ARP Packet Validation on a VLAN Enabled for Dynamic ARP Inspection



Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running dynamic ARP inspection, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running dynamic ARP inspection.

Dynamic ARP inspection ensures that hosts (on untrusted interfaces) connected to a switch running dynamic ARP inspection do not poison the ARP caches of other hosts in the network. However, dynamic ARP inspection does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running dynamic ARP inspection.

In cases in which some switches in a VLAN run dynamic ARP inspection and other switches do not, configure the interfaces connecting such switches as untrusted. However, to validate the bindings of packets from nondynamic ARP inspection switches, configure the switch running dynamic ARP inspection with ARP ACLs. When you cannot determine such bindings, at Layer 3, isolate switches running dynamic ARP inspection from switches not running dynamic ARP inspection switches. For configuration information, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 5-8.

Note Depending on the setup of the DHCP server and the network, it might not be possible to validate a given ARP packet on all switches in the VLAN.

Rate Limiting of ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack. By default, the rate for untrusted interfaces is 15 packets per second (pps). Trusted interfaces are not rate-limited. You can change this setting by using the **ip arp inspection limit** interface configuration command.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you intervene. You can use the **errdisable recovery** global configuration command to enable error disable recovery so that ports automatically emerge from this state after a specified timeout period.

For configuration information, see the "Limiting the Rate of Incoming ARP Packets" section on page 5-10.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the **ip arp inspection filter vlan** global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Logging of Dropped Packets

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

You use the **ip arp inspection log-buffer** global configuration command to configure the number of entries in the buffer and the number of entries needed in the specified interval to generate system messages. You specify the type of packets that are logged by using the **ip arp inspection vlan logging** global configuration command. For configuration information, see the "Configuring the Log Buffer" section on page 5-14.

Prerequisites

Be sure to review the Guidelines and Limitations section and the Before You Begin section within each configuration section before configuring a feature.

Guidelines and Limitations

- Dynamic ARP inspection is an ingress security feature; it does not perform any egress checking.
- Dynamic ARP inspection is not effective for hosts connected to switches that do not support dynamic ARP inspection or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, separate the domain with dynamic ARP inspection checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for dynamic ARP inspection.
- Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses. For configuration information, see Chapter 4, "Configuring DHCP Features and IP Source Guard."

  When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.
- Dynamic ARP inspection is supported on access ports, trunk ports, EtherChannel ports, and private VLAN ports.

Note Do not enable Dynamic ARP inspection on RSPAN VLANs. If Dynamic ARP inspection is enabled on RSPAN VLANs, Dynamic ARP inspection packets might not reach the RSPAN destination port.

- A physical port can join an EtherChannel port channel only when the trust state of the physical port and the channel port match. Otherwise, the physical port remains suspended in the port channel. A port channel inherits its trust state from the first physical port that joins the channel. Consequently, the trust state of the first physical port need not match the trust state of the channel.

  Conversely, when you change the trust state on the port channel, the switch configures a new trust state on all the physical ports that comprise the channel.
- The operating rate for the port channel is cumulative across all the physical ports within the channel. For example, if you configure the port channel with an ARP rate-limit of 400 pps, all the interfaces combined on the channel receive an aggregate 400 pps. The rate of incoming ARP packets on EtherChannel ports is equal to the sum of the incoming rate of packets from all the channel members. Configure the rate limit for EtherChannel ports only after examining the rate of incoming ARP packets on the channel-port members.

  The rate of incoming packets on a physical port is checked against the port-channel configuration rather than the physical-ports configuration. The rate-limit configuration on a port channel is independent of the configuration on its physical ports.

  If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.

- Limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle packets across multiple dynamic ARP inspection-enabled VLANs. You also can use the **ip arp inspection limit none** interface configuration command to make the rate unlimited. A high rate-limit on one VLAN can cause a denial-of-service attack to other VLANs when the software places the port in the error-disabled state.
- When you enable dynamic ARP inspection on the switch, policers that were configured to police ARP traffic are no longer effective. The result is that all ARP traffic is sent to the CPU.

Default Settings

Feature	Default Setting
Dynamic ARP inspection	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Rate limit of incoming ARP packets	The rate is 15 pps on untrusted interfaces, assuming that the network is a switched network with a host connecting to as many as 15 new hosts per second. The rate is unlimited on all trusted interfaces. The burst interval is 1 second.
ARP ACLs for non-DHCP environments	No ARP ACLs are defined.
Validation checks	No checks are performed.
Log buffer	When you enable dynamic ARP inspection on the switch, the switch logs all denied or dropped ARP packets. The number of entries in the log is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second.
Per-VLAN logging	The switch logs all denied or dropped ARP packets.

Configuring Dynamic ARP Inspection

This section includes the following topics:

- Configuring Dynamic ARP Inspection in DHCP Environments, page 5-7 (required in DHCP environments)
- Configuring ARP ACLs for Non-DHCP Environments, page 5-8 (required in non-DHCP environments)
- Limiting the Rate of Incoming ARP Packets, page 5-10 (optional)
- Performing Validation Checks, page 5-12 (optional)
- Configuring the Log Buffer, page 5-14 (optional)

Configuring Dynamic ARP Inspection in DHCP Environments

This procedure shows how to configure dynamic ARP inspection when two switches support this feature. You must perform this procedure on both switches. This procedure is required.

Host 1 is connected to Switch A, and Host 2 is connected to Switch B as shown in Figure 5-2 on page 5-3. Both switches are running dynamic ARP inspection on VLAN 1 where the hosts are located. A DHCP server is connected to Switch A. Both hosts acquire their IP addresses from the same DHCP server. Therefore, Switch A has the bindings for Host 1 and Host 2, and Switch B has the binding for Host 2.

For information on how to configure dynamic ARP inspection when only one switch supports the feature, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 5-8.

BEFORE YOU BEGIN

Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses. For configuration information, see Chapter 4, "Configuring DHCP Features and IP Source Guard."

DETAILED STEPS

	Command	Purpose
Step 1	**show cdp neighbors**	Verify the connection between the switches.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**ip arp inspection *vlan*** *vlan-range*	Enable dynamic ARP inspection on a per-VLAN basis. By default, dynamic ARP inspection is disabled on all VLANs. For *vlan-range*, specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. Specify the same VLAN ID for both switches.
Step 4	**interface** *interface-id*	Specify the interface connected to the other switch, and enter interface configuration mode.
Step 5	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) and enhanced network interfaces (ENIs) are disabled, and network node interfaces (NNIs) are enabled.

	Command	Purpose
Step 6	**ip arp inspection trust**	Configure the connection between the switches as trusted. By default, all interfaces are untrusted. The switch does not check ARP packets that it receives from the other switch on the trusted interface. It simply forwards the packets. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 5-14.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ip arp inspection interfaces** **show ip arp inspection vlan** *vlan-range*	Verify the dynamic ARP inspection configuration.
Step 9	**show ip dhcp snooping binding**	Verify the DHCP bindings.
Step 10	**show ip arp inspection statistics vlan** *vlan-range*	Check the dynamic ARP inspection statistics.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable dynamic ARP inspection, use the **no ip arp inspection vlan** *vlan-range* global configuration command. To return the interfaces to an untrusted state, use the **no ip arp inspection trust** interface configuration command.

EXAMPLE

This example shows how to configure dynamic ARP inspection on Switch A in VLAN 1. You would perform a similar procedure on Switch B:

```
Switch(config)# ip arp inspection vlan 1
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip arp inspection trust
```

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure dynamic ARP inspection when Switch B shown in Figure 5-2 on page 5-3 does not support dynamic ARP inspection or DHCP snooping. This procedure is required in non-DHCP environments.

If you configure port 1 on Switch A as trusted, a security hole is created because both Switch A and Host 1 could be attacked by either Switch B or Host 2. To prevent this possibility, you must configure port 1 on Switch A as untrusted. To permit ARP packets from Host 2, you must set up an ARP ACL and apply it to VLAN 1. If the IP address of Host 2 is not static (it is impossible to apply the ACL configuration on Switch A) you must separate Switch A from Switch B at Layer 3 and use a router to route packets between them.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**arp access-list** *acl-name*	Define an ARP ACL, and enter ARP access-list configuration mode. By default, no ARP access lists are defined. **Note** At the end of the ARP access list, there is an implicit **deny ip any mac any** command.
Step 3	**permit ip host** *sender-ip* **mac host** *sender-mac* [**log**]	Permit ARP packets from the specified host (Host 2). • For *sender-ip*, enter the IP address of Host 2. • For *sender-mac*, enter the MAC address of Host 2. • (Optional) Specify **log** to log a packet in the log buffer when it matches the access control entry (ACE). Matches are logged if you also configure the **matchlog** keyword in the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 5-14.
Step 4	**exit**	Return to global configuration mode.
Step 5	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	Apply the ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN. • For *arp-acl-name*, specify the name of the ACL created in Step 2. • For *vlan-range*, specify the VLAN that the switches and hosts are in. You can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) Specify **static** to treat implicit denies in the ARP ACL as explicit denies and to drop packets that do not match any previous clauses in the ACL. DHCP bindings are not used. If you do not specify this keyword, it means that there is no explicit deny in the ACL that denies the packet, and DHCP bindings determine whether a packet is permitted or denied if the packet does not match any clauses in the ACL. ARP packets containing only IP-to-MAC address bindings are compared against the ACL. Packets are permitted only if the access list permits them.
Step 6	**interface** *interface-id*	Specify the Switch A interface that is connected to Switch B, and enter interface configuration mode.

	Command	Purpose
Step 7	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 8	**no ip arp inspection trust**	Configure the Switch A interface that is connected to Switch B as untrusted. By default, all interfaces are untrusted. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 5-14.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show arp access-list** [*acl-name*] **show ip arp inspection vlan** *vlan-range* **show ip arp inspection interfaces**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the ARP ACL, use the **no arp access-list** global configuration command. To remove the ARP ACL attached to a VLAN, use the **no ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* global configuration command.

EXAMPLE

This example shows how to configure an ARP ACL called *host2* on Switch A, to permit ARP packets from Host 2 (IP address 1.1.1.1 and MAC address 0001.0001.0001), to apply the ACL to VLAN 1, and to configure port 1 on Switch A as untrusted:

```
Switch(config)# arp access-list host2
Switch(config-arp-acl)# permit ip host 1.1.1.1 mac host 1.1.1
Switch(config-arp-acl)# exit
Switch(config)# ip arp inspection filter host2 vlan 1
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no ip arp inspection trust
```

Limiting the Rate of Incoming ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you enable error-disabled recovery so that ports automatically emerge from this state after a specified timeout period.

Note Unless you configure a rate limit on an interface, changing the trust state of the interface also changes its rate limit to the default value for that trust state. After you configure the rate limit, the interface retains the rate limit even when its trust state is changed. If you enter the **no ip arp inspection limit** interface configuration command, the interface reverts to its default rate limit.

For configuration guidelines for rate limiting trunk ports and EtherChannel ports, see the "Guidelines and Limitations" section on page 5-5.

Follow this procedure to limit the rate of incoming ARP packets. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be rate-limited, and enter interface configuration mode.
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4	**ip arp inspection limit** {**rate** *pps* [**burst interval** *seconds*] | **none**}	Limit the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted interfaces and unlimited on trusted interfaces. The burst interval is 1 second. The keywords have these meanings: • For **rate** *pps*, specify an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps. • (Optional) For **burst interval** *seconds*, specify the consecutive interval in seconds, over which the interface is monitored for a high rate of ARP packets. The range is 1 to 15. • For **rate none**, specify no upper limit for the rate of incoming ARP packets that can be processed.
Step 5	**exit**	Return to global configuration mode.
Step 6	**errdisable recovery cause arp-inspection interval** *interval*	(Optional) Enable error recovery from the dynamic ARP inspection error-disable state. By default, recovery is disabled, and the recovery interval is 300 seconds. For **interval** *interval*, specify the time, in seconds, to recover from the error-disable state. The range is 30 to 86400.

	Command	Purpose
Step 7	**exit**	Return to privileged EXEC mode.
Step 8	**show ip arp inspection interfaces** **show errdisable recovery**	Verify your settings.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default rate-limit configuration, use the **no ip arp inspection limit** interface configuration command. To disable error recovery for dynamic ARP inspection, use the **no errdisable recovery cause arp-inspection** global configuration command.

EXAMPLE

This example shows how to limit the rate of the incoming ARP requests to 25 packets per second and enables the recovery timer for the arp inspection error-disable cause:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip arp inspection limit rate 25
Switch(config-if)# exit
Switch(config)# errdisable recovery cause arp-inspection
```

Performing Validation Checks

Dynamic ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can configure the switch to perform additional checks on the destination MAC address, the sender and target IP addresses, and the source MAC address. This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]}	Perform a specific check on incoming ARP packets. By default, no checks are performed. The keywords have these meanings: • For **src-mac**, check the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For **dst-mac**, check the destination MAC address in the Ethernet header against the target MAC address in ARP body. This check is performed for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For **ip**, check the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses. You must specify at least one of the keywords. Each command overrides the configuration of the previous command; that is, if a command enables **src** and **dst mac** validations, and a second command enables IP validation only, the **src** and **dst mac** validations are disabled as a result of the second command.
Step 3	**exit**	Return to privileged EXEC mode.
Step 4	**show ip arp inspection vlan** *vlan-range*	Verify your settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable checking, use the **no ip arp inspection validate** [**src-mac**] [**dst-mac**] [**ip**] global configuration command. To display statistics for forwarded, dropped, and MAC and IP validation failure packets, use the **show ip arp inspection statistics** privileged EXEC command.

EXAMPLE

This example shows how to enable the source MAC validation:

```
Switch(config)# ip arp inspection validate src-mac
```

Configuring the Log Buffer

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

A log-buffer entry can represent more than one packet. For example, if an interface receives many packets on the same VLAN with the same ARP parameters, the switch combines the packets as one entry in the log buffer and generates a single system message for the entry.

If the log buffer overflows, it means that a log event does not fit into the log buffer, and the display for the **show ip arp inspection log** privileged EXEC command is affected. A -- in the display appears in place of all data except the packet count and the time. No other statistics are provided for the entry. If you see this entry in the display, increase the number of entries in the log buffer or increase the logging rate.

This procedure is optional.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip arp inspection log-buffer** {**entries** *number* | **logs** *number* **interval** *seconds*}	Configure the dynamic ARP inspection logging buffer. By default, when dynamic ARP inspection is enabled, denied or dropped ARP packets are logged. The number of log entries is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second. The keywords have these meanings: • For **entries** *number*, specify the number of entries to be logged in the buffer. The range is 0 to 1024. • For **logs** *number* **interval** *seconds*, specify the number of entries to generate system messages in the specified interval. For **logs** *number*, the range is 0 to 1024. A 0 value means that the entry is placed in the log buffer, but a system message is not generated. For **interval** *seconds*, the range is 0 to 86400 seconds (1 day). A 0 value means that a system message is immediately generated (and the log buffer is always empty). An interval setting of 0 overrides a log setting of 0. The **logs** and **interval** settings interact. If the **logs** *number* X is greater than **interval** *seconds* Y, X divided by Y (X/Y) system messages are sent every second. Otherwise, one system message is sent every Y divided by X (Y/X) seconds.

	Command	Purpose
Step 3	**ip arp inspection vlan** *vlan-range* **logging** {**acl-match** {**matchlog** | **none**} | **dhcp-bindings** {**all** | **none** | **permit**}}	Control the type of packets that are logged per VLAN. By default, all denied or all dropped packets are logged. The term *logged* means the entry is placed in the log buffer and a system message is generated. The keywords have these meanings: • For *vlan-range*, specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For **acl-match matchlog**, log packets based on the ACE logging configuration. If you specify the **matchlog** keyword in this command and the **log** keyword in the **permit** or **deny** ARP access-list configuration command, ARP packets permitted or denied by the ACL are logged. • For **acl-match none**, do not log packets that match ACLs. • For **dhcp-bindings all**, log all packets that match DHCP bindings. • For **dhcp-bindings none**, do not log packets that match DHCP bindings. • For **dhcp-bindings permit**, log DHCP-binding permitted packets.
Step 4	**exit**	Return to privileged EXEC mode.
Step 5	**show ip arp inspection log**	Verify your settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default log buffer settings, use the **no ip arp inspection log-buffer** {**entries** | **logs**} global configuration command. To return to the default VLAN log settings, use the **no ip arp inspection vlan** *vlan-range* **logging** {**acl-match** | **dhcp-bindings**} global configuration command. To clear the log buffer, use the **clear ip arp inspection log** privileged EXEC command.

EXAMPLE

This example shows how to configure the logging buffer to hold up to 45 entries:

```
Switch(config)# ip arp inspection log-buffer entries 45
```

This example shows how to configure the logging rate for 10 logs per 3 seconds:

```
Switch(config)# ip arp inspection log-buffer logs 10 interval 3
```

This example shows how to configure an ARP inspection on VLAN 1 to log packets that match the ACLs:

```
Switch(config)# ip arp inspection vlan 1 logging acl-match matchlog
```

Verifying Configuration

Command	Description
show arp access-list [*acl-name*]	Displays detailed information about ARP ACLs.
show ip arp inspection interfaces [*interface-id*]	Displays the trust state and the rate limit of ARP packets for the specified interface or all interfaces.
show ip arp inspection vlan *vlan-range*	Displays the configuration and the operating state of dynamic ARP inspection for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).
clear ip arp inspection statistics	Clears dynamic ARP inspection statistics.
show ip arp inspection statistics [**vlan** *vlan-range*]	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).
clear ip arp inspection log	Clears the dynamic ARP inspection log buffer.
show ip arp inspection log	Displays the configuration and contents of the dynamic ARP inspection log buffer.

For the **show ip arp inspection statistics** command, the switch increments the number of forwarded packets for each ARP request and response packet on a trusted dynamic ARP inspection port. The switch increments the number of ACL or DHCP permitted packets for each packet that is denied by source MAC, destination MAC, or IP validation checks, and the switch increments the appropriate failure count.

Configuration Example

This example shows how to configure dynamic ARP inspection on Switch A in VLAN 1. You would perform a similar procedure on Switch B:

```
Switch(config)# ip arp inspection vlan 1
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip arp inspection trust
```

This example shows how to configure an ARP ACL called *host2* on Switch A, to permit ARP packets from Host 2 (IP address 1.1.1.1 and MAC address 0001.0001.0001), to apply the ACL to VLAN 1, and to configure port 1 on Switch A as untrusted:

```
Switch(config)# arp access-list host2
Switch(config-arp-acl)# permit ip host 1.1.1.1 mac host 1.1.1
Switch(config-arp-acl)# exit
Switch(config)# ip arp inspection filter host2 vlan 1
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no ip arp inspection trust
```

This example shows how to limit the rate of the incoming ARP requests to 25 packets per second and enables the recovery timer for the arp inspection error-disable cause:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip arp inspection limit rate 25
Switch(config-if)# exit
Switch(config)# errdisable recovery cause arp-inspection
```

This example shows how to enable the source MAC validation:

```
Switch(config)# ip arp inspection validate src-mac
```

This example shows how to configure the logging buffer to hold up to 45 entries:

```
Switch(config)# ip arp inspection log-buffer entries 45
```

This example shows how to configure the logging rate for 10 logs per 3 seconds:

```
Switch(config)# ip arp inspection log-buffer logs 10 interval 3
```

This example shows how to configure an ARP inspection on VLAN 1 to log packets that match the ACLs:

```
Switch(config)# ip arp inspection vlan 1 logging acl-match matchlog
```

Related Documents

- Cisco IOS Master Command List, All Releases
- Cisco IOS IP Addressing Services Command Reference

Configuring Network Security with ACLs

This chapter describes how to configure network security on the Cisco Industrial Ethernet 2000U Series Switch (IE 2000U) by using access control lists (ACLs), which are also referred to in commands and tables as access lists.

Note Information in this chapter about IP ACLs is specific to IP Version 4 (IPv4). For information about configuring IPv6 ACLs, see Chapter 7, "Configuring IPv6 ACLs".

For complete syntax and usage information for the commands used in this chapter, see the documents listed in the "Related Documents" section on page 6-42.

This chapter includes the following sections:

- Information About ACLs, page 6-1
- Prerequisites, page 6-9
- Guidelines and Limitations, page 6-9
- Default Settings, page 6-12
- Configuring IPv4 ACLs, page 6-12
- Creating Named MAC Extended ACLs, page 6-28
- Configuring VLAN Maps, page 6-31
- Verifying Configuration, page 6-35
- Verifying Configuration, page 6-35
- Configuration Example, page 6-35
- Related Documents, page 6-42

Information About ACLs

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces or VLANs. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch accepts or rejects the packets. Because the switch stops

testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both.

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

The switch supports IPv4 ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs. For more information, see the "Information About QoS" section in the *Cisco Connected Grid Switches QoS Software Configuration Guide*.

This section includes the following topics:

- Supported ACLs, page 6-2
- Handling Fragmented and Unfragmented Traffic, page 6-5
- Hardware and Software Treatment of IP ACLs, page 6-6
- Using VLAN Maps with Router ACLs, page 6-7

Supported ACLs

The switch supports three applications of ACLs to filter traffic:

- Port ACLs access-control traffic entering a Layer 2 interface. The switch does not support port ACLs in the outbound direction. You can apply only one IP access list and one MAC access list to a Layer 2 interface.
- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces in a specific direction (inbound or outbound). The switch must be running the IP services image to support router ACLs.
- VLAN ACLs or VLAN maps access-control all packets (forwarded and routed). You can use VLAN maps to filter traffic between devices in the same VLAN. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv4. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets entering the VLAN are checked against the VLAN map. Packets can either enter the VLAN through a switch port or through a routed port after being routed.

You can use input port ACLs, router ACLs, and VLAN maps on the same switch. However, a port ACL takes precedence over a router ACL or VLAN map.

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map.

- When an input router ACL and input port ACL exist in an switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming routed IPv4 packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IPv4 packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IPv4 packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IPv4 packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

If IEEE 802.1Q tunneling is configured on an interface, any 802.1Q encapsulated IPv4 packets received on the tunnel port can be filtered by MAC ACLs, but not by IP v4 ACLs. This is because the switch does not recognize the protocol inside the 802.1Q header. This restriction applies to router ACLs, port ACLs, and VLAN maps. For more information about 802.1Q tunneling, refer to the “Configuring IEEE 802.1Q Tunneling” and “Configuring Layer 2 Protocol Tunneling” chapters in the *Cisco Connected Grid Switches Layer 2 Switching Software Configuration Guide*.

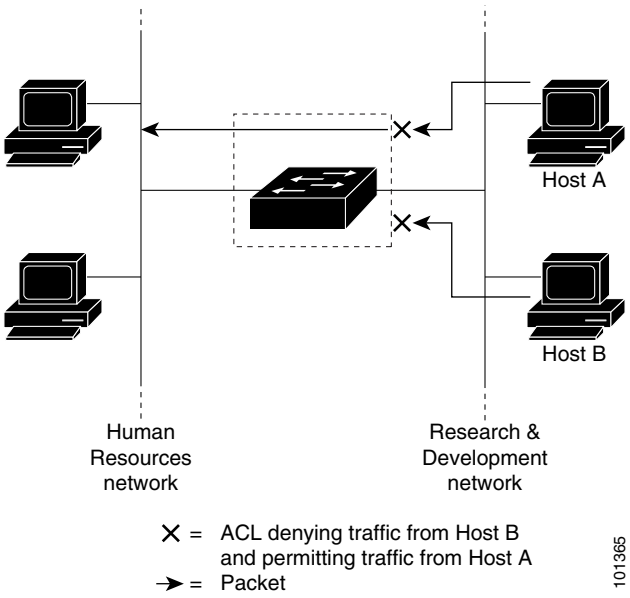
Port ACLs

Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces, not on EtherChannel interfaces, and can be applied only on interfaces in the inbound direction. These access lists are supported on Layer 2 interfaces:

- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

The switch examines ACLs associated with all inbound features configured on a given interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs are used to control access to a network or to part of a network. Figure 6-1 is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but prevent Host B from accessing the same network. Port ACLs can only be applied to Layer 2 interfaces in the inbound direction.

Figure 6-1 Using ACLs to Control Traffic to a Network



When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.

Note You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

If the switch is running the IP services image, you can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. You apply router ACLs on interfaces for specific directions (inbound or outbound). You can apply one router ACL in each direction on an interface.

One ACL can be used with multiple features for a given interface, and one feature can use multiple ACLs. When a single router ACL is used by multiple features, it is examined multiple times.

The switch supports these access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. However, router ACLs are supported in both directions. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL, and can be used to control access to a network or to part of a network. In Figure 6-1, ACLs applied at the router input allow Host A to access the Human Resources network, but prevent Host B from accessing the same network.

VLAN Maps

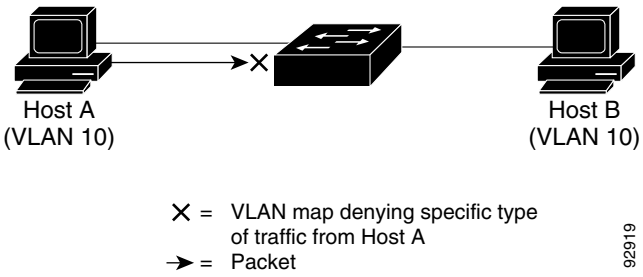
VLAN ACLs or VLAN maps can access-control *all* traffic. You can apply VLAN maps to all packets that are routed into or out of a VLAN or are forwarded within a VLAN in the switch. VLAN maps are used for security packet filtering and are not defined by direction (input or output).

You can configure VLAN maps to match Layer 3 addresses for IPv4 traffic.

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic *is not* access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied based on the action specified in the map. Figure 6-2 shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.

Figure 6-2 Using VLAN Maps to Control Traffic



Handling Fragmented and Unfragmented Traffic

IPv4 packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some ACEs do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IPv4 packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) and are considered to match the fragment regardless of what the missing Layer 4 information might have been.

- Deny ACEs that check Layer 4 information and never match a fragment unless the fragment contains Layer 4 information.

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Switch(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Switch(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Switch(config)# access-list 102 permit tcp any host 10.1.1.2
Switch(config)# access-list 102 deny tcp any any
```

Note In the first and second ACEs in the examples, the *eq* keyword after the destination address means to test for the TCP-destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2., port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer 3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.
- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

  Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.
- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.

Hardware and Software Treatment of IP ACLs

ACL processing is primarily accomplished in hardware, but requires forwarding of some traffic flows to the CPU for software processing. If the hardware reaches its capacity to store ACL configurations, packets are sent to the CPU for forwarding. The forwarding rate for software-forwarded traffic is substantially less than for hardware-forwarded traffic.

Note If an ACL configuration cannot be implemented in hardware due to an out-of-resource condition on a switch, then only the traffic in that VLAN arriving on that switch is affected (forwarded in software). Software forwarding of packets might adversely impact the performance of the switch, depending on the number of CPU cycles that this consumes.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the **log** keyword
- Generating ICMP unreachable messages

When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of great enough bandwidth, not all of the packets that are forwarded can be logged.

If router ACL configuration cannot be applied in hardware, packets arriving in a VLAN that must be routed are routed in software. If ACLs cause large numbers of packets to be sent to the CPU, the switch performance can be negatively affected.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show access-lists hardware counters** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

Router ACLs function as follows:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If **log** has not been specified, the flows that match a *deny* statement in a security ACL are dropped by the hardware if *ip unreachables* is disabled. The flows matching a *permit* statement are switched in hardware.
- Adding the **log** keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU for logging only. If the ACE is a *permit* statement, the packet is still switched and routed in hardware.

Using VLAN Maps with Router ACLs

To access control routed traffic, you can use VLAN maps only or a combination of router ACLs and VLAN maps. You can define router ACLs on both input and output routed VLAN interfaces. If a packet flow matches a VLAN-map deny clause in the ACL, regardless of the router ACL configuration, the packet flow is denied.

Note When you use router ACLs with VLAN maps, packets that require logging in the router ACLs are not logged if they are denied by a VLAN map.

If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map, and no action specified, the packet is forwarded if it does not match any VLAN map entry.

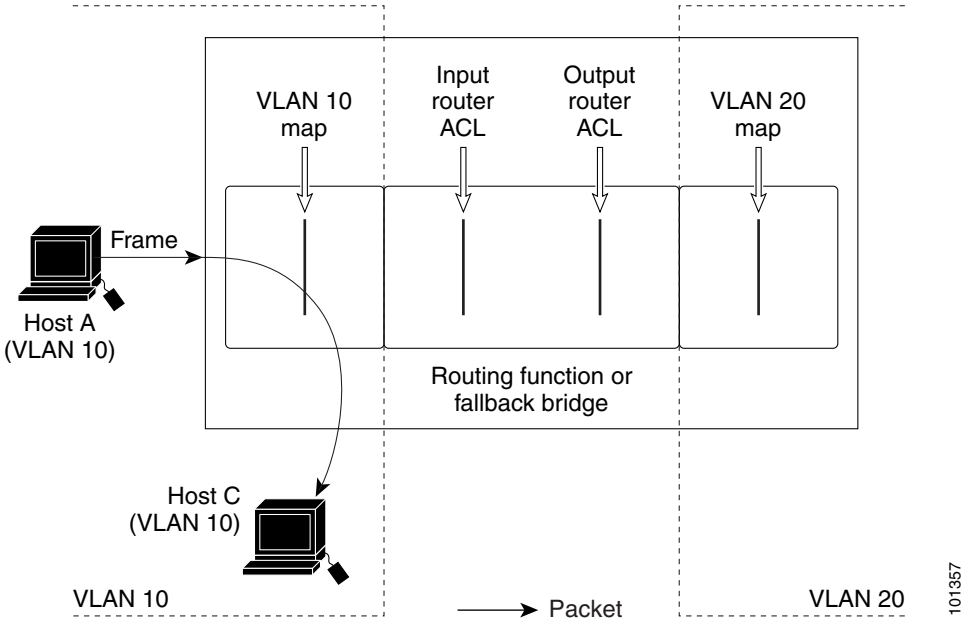
Examples of Router ACLs and VLAN Maps Applied to VLANs

This section gives examples of applying router ACLs and VLAN maps to a VLAN for switched, routed, and multicast packets. Although the following illustrations show packets being forwarded to their destination, each time the packet's path crosses a line indicating a VLAN map or an ACL, it is also possible that the packet might be dropped, rather than forwarded.

ACLs and Switched Packets

Figure 6-3 shows how an ACL is applied on packets that are switched within a VLAN. Packets switched within the VLAN without being routed or forwarded are only subject to the VLAN map of the input VLAN.

Figure 6-3 Applying ACLs on Switched Packets

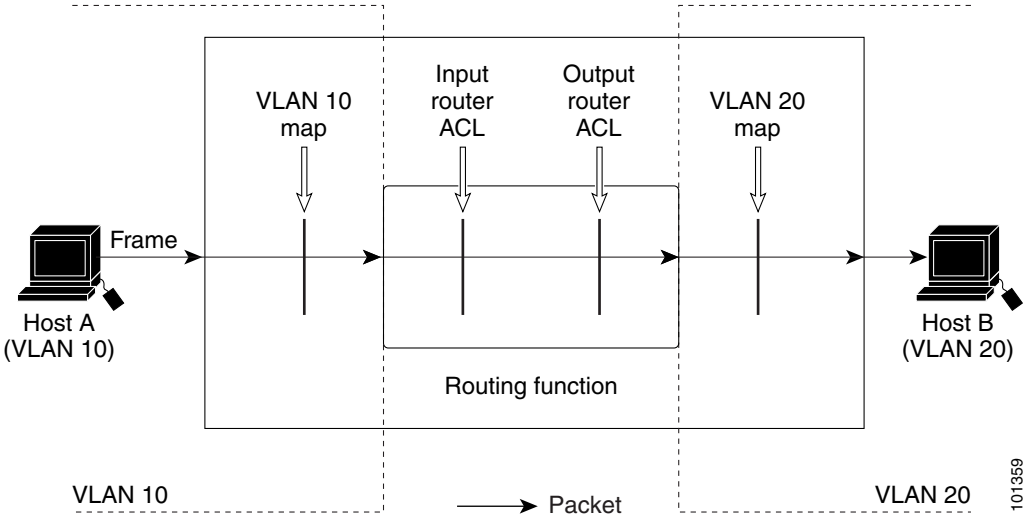


ACLs and Routed Packets

Figure 6-4 shows how ACLs are applied on routed packets. For routed packets, the ACLs are applied in this order:

1. VLAN map for input VLAN
2. Input router ACL
3. Output router ACL
4. VLAN map for output VLAN

Figure 6-4 Applying ACLs on Routed Packets

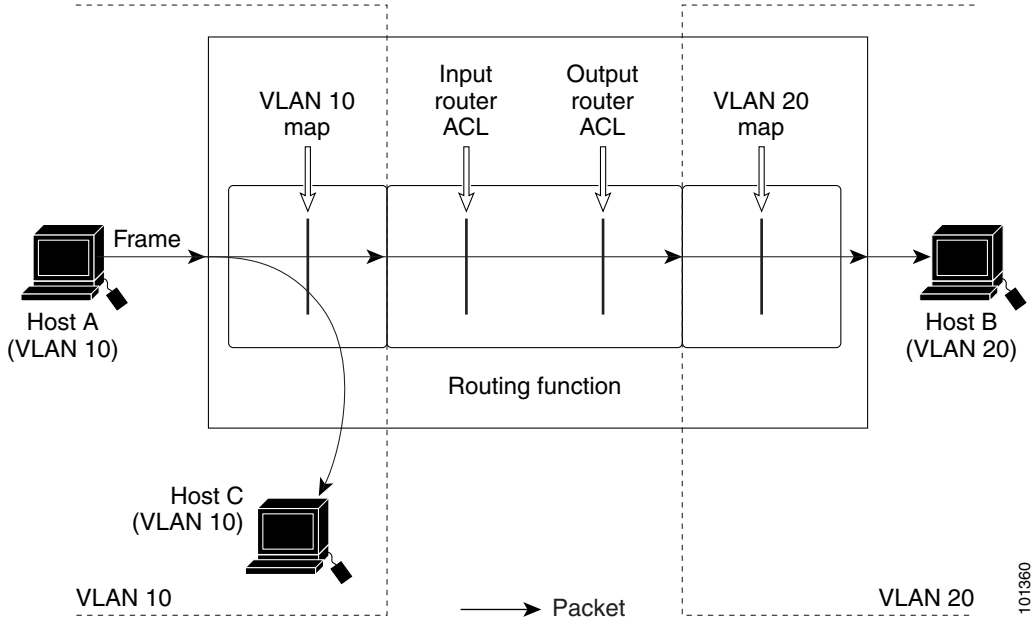


ACLs and Multicast Packets

Figure 6-5 shows how ACLs are applied on packets that are replicated for IP multicasting. A multicast packet being routed has two different kinds of filters applied: one for destinations that are other ports in the input VLAN and another for each of the destinations that are in other VLANs to which the packet has been routed. The packet might be routed to more than one output VLAN, in which case a different router output ACL and VLAN map would apply for each destination VLAN.

The final result is that the packet might be permitted in some of the output VLANs and not in others. A copy of the packet is forwarded to those destinations where it is permitted. However, if the input VLAN map (VLAN 10 map in Figure 6-5) drops the packet, no destination receives a copy of the packet.

Figure 6-5 Applying ACLs on Multicast Packets



Prerequisites

Before you create or apply an IP access list, you should understand the concepts in the "Information About ACLs" section on page 6-1.

You should also have IP running in your network.

Guidelines and Limitations

The switch does not support these Cisco IOS router ACL-related features:

- Non-IP protocol ACLs (see Table 6-1 on page 6-13) or bridge-group ACLs
- IP accounting
- Inbound and outbound rate limiting (except with QoS ACLs)
- Reflexive ACLs or dynamic ACLs

- ACL logging for port ACLs and VLAN maps

Configuring Named ACLs

Follow these guidelines when configuring standard or extended named ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name. VLAN maps also accept a name.
- A standard ACL and an extended ACL cannot have the same name.
- Numbered ACLs are also available, as described in the "Creating Standard and Extended IPv4 ACLs" section on page 6-13.
- You can use standard and extended ACLs (named or numbered) in VLAN maps.

Applying an ACL to an Interface

Follow these guidelines when you apply an ACL to an interface:

- When controlling access to an interface, you can use a named or numbered ACL.
- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface or a VLAN map applied to the VLAN. Incoming packets received on the Layer 2 port are always filtered by the port ACL.
- If you apply an ACL to a Layer 3 interface and routing is not enabled on the switch, the ACL only filters packets that are intended for the CPU, such as SNMP, Telnet, or web traffic. You do not have to enable routing to apply ACLs to Layer 2 interfaces.
- When private VLANs are configured, you can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.

Applying a MAC ACL to an Interface

Follow these guidelines when you apply a MAC ACL to an interface:

- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface or a VLAN map applied to the VLAN. Incoming packets received on the Layer 2 port are always filtered by the port ACL.
- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.

Configuring VLAN Maps

Follow these guidelines when configuring VLAN maps:

- If there is no ACL configured to deny traffic on an interface and *no* VLAN map is configured, all traffic is permitted.
- Each VLAN map consists of a series of entries. The order of entries in an VLAN map is important. A packet that comes into the switch is tested against the first entry in the VLAN map. If it matches, the action specified for that part of the VLAN map is taken. If there is no match, the packet is tested against the next entry in the map.
- If the VLAN map has at least one match clause for the type of packet (IP or MAC) and the packet does not match any of these match clauses, the default is to drop the packet. If there is no match clause for that type of packet in the VLAN map, the default is to forward the packet.
- The system might take longer to boot if you have configured a very large number of ACLs.

- Logging is not supported for VLAN maps.
- If VLAN map configuration cannot be applied in hardware, all packets in that VLAN must be routed by software.
- When a switch has an IP access list or MAC access list applied to a Layer 2 interface, and you apply a VLAN map to a VLAN that the port belongs to, the port ACL takes precedence over the VLAN map.
- You can configure VLAN maps on primary and secondary VLANs. However, we recommend that you configure the same VLAN maps on private-VLAN primary and secondary VLANs.
- When a frame is Layer-2 forwarded within a private VLAN, the same VLAN map is applied at the ingress side and at the egress side. When a frame is routed from inside a private VLAN to an external port, the private-VLAN map is applied at the ingress side.
  - For frames going upstream from a host port to a promiscuous port, the VLAN map configured on the secondary VLAN is applied.
  - For frames going downstream from a promiscuous port to a host port, the VLAN map configured on the primary VLAN is applied.

  To filter out specific IP traffic for a private VLAN, you should apply the VLAN map to both the primary and secondary VLANs. For more information about private VLANs, see the chapter "Configuring Private VLANs" in the *Cisco Connected Grid Switches Layer 2 Switching Software Configuration Guide*.
- See the "Verifying Configuration" section on page 6-35 for configuration examples.
- For information about using both router ACLs and VLAN maps, see the "VLAN Maps and Router ACL" section on page 6-11.

VLAN Maps and Router ACL

These guidelines are for configurations where you need to have an router ACL *and* a VLAN map on the same VLAN. These guidelines do not apply to configurations where you are mapping router ACLs and VLAN maps on different VLANs.

The switch hardware provides one lookup for security ACLs for each direction (input and output); therefore, you must merge a router ACL and a VLAN map when they are configured on the same VLAN. Merging the router ACL with the VLAN map might significantly increase the number of ACEs.

If you must configure a router ACL and a VLAN map on the same VLAN, use these guidelines for both router ACL and VLAN map configuration:

- You can configure only one VLAN map and one router ACL in each direction (input/output) on a VLAN interface.
- Whenever possible, try to write the ACL with all entries having a single action except for the final, default action of the other type. That is, write the ACL using one of these two forms:

  permit...
  permit...
  permit...
  deny ip any any

  or

deny...
deny...
deny...
permit ip any any

- To define multiple actions in an ACL (permit, deny), group each action type together to reduce the number of entries.
- Avoid including Layer 4 information in an ACL; adding this information complicates the merging process. The best merge results are obtained if the ACLs are filtered based on IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports). It is also helpful to use *don't care* bits in the IP address, whenever possible.

  If you need to specify the full-flow mode and the ACL contains both IP ACEs and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list. This gives priority to the filtering of traffic based on IP addresses.

Default Settings

Feature	Default Setting
Numbered standard and extended access lists	Defaults to a list that denies everything. Access lists are terminated by an implicit deny statement.
Named standard and extended access lists	No IP access list or object-group ACL is defined, and outbound ACLs do not match and filter IP helper relayed traffic.
Named MAC extended ACLs	No extended ACLs are defined.
VLAN MAP	No VLAN map is configured.

Configuring IPv4 ACLs

Configuring IP v4 ACLs on the switch is the same as configuring IPv4 ACLs on other Cisco switches and routers. The process is briefly described here. For more detailed information on configuring ACLs, see the *Security Configuration Guide: Access Control Lists, Cisco IOS Release 15M&T*.

These are the steps to use IP ACLs on the switch:

Step 1 Create an ACL by specifying an access list number or name and the access conditions.

Step 2 Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.

This section includes the following topics:

- Creating Standard and Extended IPv4 ACLs, page 6-13
- Applying an IPv4 ACL to a Terminal Line, page 6-25
- Applying an IPv4 ACL to an Interface, page 6-26
- Troubleshooting ACLs, page 6-27

Creating Standard and Extended IPv4 ACLs

This section describes IP ACLs. An ACL is a sequential collection of permit and deny conditions. One by one, the switch tests packets against the conditions in an access list. The first match determines whether the switch accepts or rejects the packet. Because the switch stops testing after the first match, the order of the conditions is critical. If no conditions match, the switch denies the packet.

The software supports these types of ACLs or access lists for IPv4:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and optional protocol-type information for finer granularity of control.

This section includes the following topics:

- IPv4 Access List Numbers, page 6-13
- ACL Logging, page 6-14
- Creating a Numbered Standard ACL, page 6-14
- Creating a Numbered Extended ACL, page 6-16
- Resequencing ACEs in an ACL, page 6-20
- Creating Named Standard and Extended ACLs, page 6-20
- Using Time Ranges with ACLs, page 6-22
- Including Comments in ACLs, page 6-24

IPv4 Access List Numbers

The number you use to denote your IPv4 ACL shows the type of access list that you are creating. Table 6-1 lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 6-1 Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No
300–399	DECnet access list	No
400–499	XNS standard access list	No
500–599	XNS extended access list	No
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No

Table 6-1 Access List Numbers (continued)

Access List Number	Type	Supported
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

Note In addition to numbered standard and extended IPv4 ACLs, you can also create standard and extended named IPv4 ACLs by using the supported numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

ACL Logging

The switch software can provide logging messages about packets permitted or denied by a standard IP access list. That is, any packet that matches the ACL causes an informational logging message about the packet to be sent to the console. The level of messages logged to the console is controlled by the logging console commands controlling the syslog messages.

Note Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL causes a logging message to be generated immediately, and subsequent packets are collected over 5-minute intervals before they appear or are logged. The logging message includes the access list number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.

Creating a Numbered Standard ACL

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a *don't care* mask of 0.0.0.0 are moved to the top of the list, above any entries with non-zero *don't care* masks. Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

After creating a numbered standard IPv4 ACL, you can apply it to terminal lines (see the "Applying an IPv4 ACL to a Terminal Line" section on page 6-25), to interfaces (see the "Applying an IPv4 ACL to an Interface" section on page 6-26), or to VLANs (see the "Configuring VLAN Maps" section on page 6-31).

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] [**log**]	Define a standard IPv4 access list by using a source address and wildcard. The *access-list-number* is a decimal number from 1 to 99 or 1300 to 1999. Enter **deny** or **permit** to specify whether to deny or permit access if conditions are matched. The *source* is the source address of the network or host from which the packet is being sent specified as: • The 32-bit quantity in dotted-decimal format. • The keyword **any** as an abbreviation for *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. • The keyword **host** as an abbreviation for source and source-wildcard of *source* 0.0.0.0. (Optional) The *source-wildcard* applies wildcard bits to the source. (Optional) Enter **log** to cause an informational logging message about the packet that matches the entry to be sent to the console.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists** [*number* | *name*]	Show the access list configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no access-list** *access-list-number* global configuration command to delete the entire ACL. You cannot delete individual ACEs from numbered access lists.

EXAMPLE

This example shows how to create a standard ACL to deny access to IP host 171.69.198.102, permit access to any others, and display the results.

```
Switch (config)# access-list 2 deny host 171.69.198.102
Switch (config)# access-list 2 permit any
Switch(config)# end
Switch# show access-lists
Standard IP access list 2
    10 deny   171.69.198.102
    20 permit any
```

Creating a Numbered Extended ACL

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

Some protocols also have specific parameters and keywords that apply to that protocol.

These IP protocols are supported (protocol keywords are in parentheses in bold):

Authentication Header Protocol (**ahp**), Enhanced Interior Gateway Routing Protocol (**eigrp**), Encapsulation Security Payload (**esp**), generic routing encapsulation (**gre**), Internet Control Message Protocol (**icmp**), Internet Group Management Protocol (**igmp**), any Interior Protocol (**ip**), IP in IP tunneling (**ipinip**), KA9Q NOS-compatible IP over IP tunneling (**nos**), Open Shortest Path First routing (**ospf**), Payload Compression Protocol (**pcp**), Protocol Independent Multicast (**pim**), Transmission Control Protocol (**tcp**), or User Datagram Protocol (**udp**).

Note ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

For more details on the specific keywords for each protocol, see the *Cisco IOS Security Command Reference*.

Note The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Supported parameters can be grouped into these categories: TCP, UDP, ICMP, IGMP, or other IP.

After an ACL is created, any additions (possibly entered from the terminal) are placed at the end of the list. You cannot selectively add or remove access list entries from a numbered access list.

Note When you are creating an ACL, remember that, by default, the end of the access list contains an implicit deny statement for all packets if it did not find a match before reaching the end.

After creating a numbered extended ACL, you can apply it to terminal lines (see the "Applying an IPv4 ACL to a Terminal Line" section on page 6-25), to interfaces (see the "Applying an IPv4 ACL to an Interface" section on page 6-26), or to VLANs (see the "Configuring VLAN Maps" section on page 6-31).

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2a	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Note** If you enter a **dscp** value, you cannot enter **tos** or **precedence**. You can enter both a **tos** and a **precedence** value with no **dscp**.	Define an extended IPv4 access list and the access conditions. The *access-list-number* is a decimal number from 100 to 199 or 2000 to 2699. Enter **deny** or **permit** to specify whether to deny or permit the packet when conditions are matched. For *protocol*, enter the name or number of an IP protocol: **ahp**, **eigrp**, **esp**, **gre**, **icmp**, **igmp**, **igrp**, **ip**, **ipinip**, **nos**, **ospf**, **pcp**, **pim**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP) use the keyword **ip**. **Note** This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see steps 2b through 2e. • *source*–Number of the network or host from which the packet is sent. • *source-wildcard*–Applies wildcard bits to the source. • *destination*–Network or host number to which the packet is sent. • *destination-wildcard*–Applies wildcard bits to the destination. Specifications for source, source-wildcard, destination, and destination-wildcard: • The 32-bit quantity in dotted-decimal format. • The keyword **any** for 0.0.0.0 255.255.255.255 (any host). • The keyword **host** for a single host 0.0.0.0. The other keywords are optional and have these meanings: • **precedence**—Match packets with a precedence level specified as a number from 0 to 7 or by name: **routine** (**0**), ***priority*** (**1**), **immediate** (**2**), **flash** (**3**), **flash-override** (**4**), **critical** (**5**), **internet** (6), **network** (**7**). • **fragments**–Check non-initial fragments. • **tos**—Match by type of service level, specified by a number from 0 to 15 or a name: **normal** (**0**), **max-reliability** (**2**), **max-throughput** (**4**), **min-delay** (**8**). • **log**—Create an informational logging message to be sent to the console about the packet that matches the entry or **log-input** to include the input interface in the log entry.

	Command	Purpose
		• **time-range**—For an explanation of this keyword, see the "Using Time Ranges with ACLs" section on page 6-22. • **dscp**—Match packets with the DSCP value (0 to 63), or use the question mark (?) to see a list of available values.
or	**access-list** *access-list-number* {**deny** | **permit**} *protocol* **any any** [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	In access-list configuration mode, define an extended IP access list using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255 and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255. You can use the **any** keyword in place of source and destination address and wildcard.
or	**access-list** *access-list-number* {**deny** | **permit**} *protocol* **host** *source* **host** *destination* [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	Define an extended IP access list by using an abbreviation for a source and a source wildcard of *source* 0.0.0.0 and an abbreviation for a destination and destination wildcard of *destination* 0.0.0.0. You can use the **host** keyword in place of the source and destination wildcard or mask.
Step 2b	**access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**established**] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*]	(Optional) Define an extended TCP access list and the access conditions. Enter **tcp** for Transmission Control Protocol. The parameters are the same as those described in Step 2a, with these exceptions: (Optional) Enter an *operator* and *port* to compare source (if positioned after *source source-wildcard*) or destination (if positioned after *destination destination-wildcard*) port. Possible operators include **eq** (equal), **gt** (greater than), **lt** (less than), **neq** (not equal), and **range** (inclusive range). Operators require a port number (**range** requires two port numbers separated by a space). Enter the *port* number as a decimal number (from 0 to 65535) or the name of a TCP port. To see TCP port names, use the ? or see the "Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*. Use only TCP port numbers or names when filtering TCP. The other optional keywords have these meanings: • **established**—Enter to match an established connection. This has the same function as matching on the **ack** or **rst** flag. • *flag*—Enter one of these flags to match by the specified TCP header bits: **ack** (acknowledge), **fin** (finish), **psh** (push), **rst** (reset), **syn** (synchronize), or **urg** (urgent).

	Command	Purpose
Step 2c	**access-list** *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended UDP access list and the access conditions. Enter **udp** for the User Datagram Protocol. The UDP parameters are the same as those described for TCP except that the [*operator* [*port*]] port number or name must be a UDP port number or name, and the **flag** and **established** parameters are not valid for UDP.
Step 2d	**access-list** *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended ICMP access list and the access conditions. Enter **icmp** for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 2a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: • *icmp-type*—Enter to filter by ICMP message type, a number from 0 to 255. • *icmp-code*—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • *icmp-message*—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type names and code names, use the ?, or see the "Configuring IP Services" section of the *Cisco IOS IP Configuration Guide, Release 12.2*.
Step 2e	**access-list** *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended IGMP access list and the access conditions. Enter **igmp** for Internet Group Management Protocol. The IGMP parameters are the same as those described for most IP protocols in Step 2a, with this optional parameter. *igmp-type*—To match IGMP message type, enter a number from 0 to 15, or enter the message name (**host-query**, **host-report**, **pim,** or **trace**). **Note** Although visible in the command-line help, **dvmrp** is not supported.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists** [*number* | *name*]	Verify the access list configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no access-list** *access-list-number* global configuration command to delete the entire access list. You cannot delete individual ACEs from numbered access lists.

EXAMPLE

This example shows how to create and display an extended access list to deny Telnet access from any host in network 171.69.198.0 to any host in network 172.20.52.0 and to permit any others. (The **eq** keyword after the destination address means to test for the TCP destination port number equaling Telnet.)

```
Switch(config)# access-list 102 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq
telnet
Switch(config)# access-list 102 permit tcp any any
Switch(config)# end
Switch# show access-lists
Extended IP access list 102
    10 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq telnet
    20 permit tcp any any
```

Resequencing ACEs in an ACL

Sequence numbers for the entries in an access list are automatically generated when you create a new ACL.You can use the **ip access-list resequence** global configuration command to edit the sequence numbers in an ACL and change the order in which ACEs are applied. For example, if you add a new ACE to an ACL, it is placed at the bottom of the list. By changing the sequence number, you can move the ACE to a different position in the ACL.

For more information about the **ip access-list resequence** command, see the "IP Access List Entry Sequence Numbering" section in the *Security Configuration Guide: Access Control Lists, Cisco IOS Release 15M&T*.

Creating Named Standard and Extended ACLs

You can identify IPv4 ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IPv4 access lists in a router than if you were to use numbered access lists. If you identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, not all commands that use IP access lists accept a named access list.

Note The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

When you are creating standard or extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL.

After creating a named ACL, you can apply it to interfaces (see the "Applying an IPv4 ACL to an Interface" section on page 6-26) or to VLANs (see the "Configuring VLAN Maps" section on page 6-31).

Creating Named Standard ACLs

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

See the "Configuring Named ACLs" section on page 6-10 for configuration guidelines.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip access-list standard** *name*	Define a standard IPv4 access list using a name, and enter access-list configuration mode. **Note** The name can be a number from 1 to 99.
Step 3	**deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**] or **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]	In access-list configuration mode, specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • **any**—A source and source wildcard of 0.0.0.0 255.255.255.255.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* | *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a named standard ACL, use the **no ip access-list standard** *name* global configuration command.

EXAMPLE

The following example defines a standard access list named Internetfilter:

```
Switch(config)# ip access-list standard Internetfilter
Switch(config-std-nacl)# permit 192.168.255.0 0.0.0.255
Switch(config-std-nacl)# permit 10.88.0.0 0.0.255.255
Switch(config-std-nacl)# permit 10.0.0.0 0.255.255.255
```

Creating Named Extended ACLs

BEFORE YOU BEGIN

See the "Configuring Named ACLs" section on page 6-10 for configuration guidelines.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip access-list extended** *name*	Define an extended IPv4 access list using a name and enter access-list configuration mode. **Note** The name can be a number from 100 to 199.
Step 3	{**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | **host** *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*]	In access-list configuration mode, specify the conditions allowed or denied. Use the **log** keyword to get access list logging messages, including violations. See the "Creating a Numbered Extended ACL" section on page 6-16 for definitions of protocols and other keywords. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • ***host*** *destination*—A destination and destination wildcard of *destination* 0.0.0.0. • **any**—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* | *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a named extended ACL, use the **no ip access-list extended** *name* global configuration command.

EXAMPLE

This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Switch(config)# ip access-list extended border-list
Switch(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Using Time Ranges with ACLs

You can selectively apply extended ACLs based on the time of day and week by using the **time-range** global configuration command. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when applying an ACL to set restrictions to the access list. You can use the time range to define when the permit or deny statements in the ACL are in effect, for example, during a specified time period or on specified days of the week. The **time-range** keyword and argument are referenced in the named and numbered extended ACL task tables in the previous sections, the "Creating Standard and Extended IPv4 ACLs" section on page 6-13,

and the "Creating Named Standard and Extended ACLs" section on page 6-20.

These are some of the many possible benefits of using time ranges:

- You have more control over permitting or denying a user access to resources, such as an application (identified by an IP address/mask pair and a port number).
- You can control logging messages. ACL entries can be set to log traffic only at certain times of the day. Therefore, you can simply deny access without needing to analyze many logs generated during peak hours.

Time-based access lists trigger CPU activity because the new configuration of the access list must be merged with other features and the combined configuration loaded into the TCAM. For this reason, you should be careful not to have several access lists configured to take affect in close succession (within a small number of minutes of each other.)

BEFORE YOU BEGIN

The time range relies on the switch system clock; therefore, you need a reliable clock source. We recommend that you use Network Time Protocol (NTP) to synchronize the switch clock. For more information, see the "Configuring the System Time and Date" chapter in the *Cisco Connected Grid Switches System Management Software Configuration Guide*.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**time-range** *time-range-name*	Assign a meaningful name (for example, *workhours*) to the time range to be created, and enter time-range configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.
Step 3	**absolute** [**start** *time date*] [**end** *time date*] or **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm* or **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm*	Specify when the function it will be applied to is operational. • You can use only one **absolute** statement in the time range. If you configure more than one absolute statement, only the one configured last is executed. • You can enter multiple **periodic** statements. For example, you could configure different hours for weekdays and weekends. See the example configurations.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show time-range**	Verify the time-range configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Repeat the steps if you want multiple items in effect at different times. To remove a configured time-range limitation, use the **no time-range** *time-range-name* global configuration command.

EXAMPLE

This example shows how to configure time ranges for *workhours* and to configure January 1, 2013 as a company holiday and to verify your configuration:

```
Switch(config)# time-range workhours
Switch(config-time-range)# periodic weekdays 8:00 to 12:00
Switch(config-time-range)# periodic weekdays 13:00 to 17:00
Switch(config-time-range)# exit
Switch(config)# time-range new_year_day_2013
Switch(config-time-range)# absolute start 00:00 1 Jan 2013 end 23:59 1 Jan 2013
Switch(config-time-range)# end
Switch# show time-range
time-range entry: new_year_day_2013 (inactive)
   absolute start 00:00 01 January 2013 end 23:59 01 January 2013
time-range entry: workhours (inactive)
   periodic weekdays 8:00 to 12:00
   periodic weekdays 13:00 to 17:00
```

To apply a time-range, enter the time-range name in an extended ACL that can implement time ranges. This example shows how to create and verify extended access list 188 that denies TCP traffic from any source to any destination during the defined holiday times and permits all TCP traffic during work hours:

```
Switch(config)# access-list 188 deny tcp any any time-range new_year_day_2013
Switch(config)# access-list 188 permit tcp any any time-range workhours
Switch(config)# end
Switch# show access-lists
Extended IP access list 188
   10 deny tcp any any time-range new_year_day_2013 (inactive)
   20 permit tcp any any time-range workhours (inactive)
```

This example uses named ACLs to permit and deny the same traffic:

```
Switch(config)# ip access-list extended deny_access
Switch(config-ext-nacl)# deny tcp any any time-range new_year_day_2013
Switch(config-ext-nacl)# exit
Switch(config)# ip access-list extended may_access
Switch(config-ext-nacl)# permit tcp any any time-range workhours
Switch(config-ext-nacl)# end
Switch# show ip access-lists
Extended IP access list deny_access
    10 deny tcp any any time-range new_year_day_2013 (inactive)
Extended IP access list may_access
    10 permit tcp any any time-range workhours (inactive)
```

Including Comments in ACLs

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. You should be consistent about where you put the remark so that it is clear which remark describes which permit or deny statement. For example, it would be confusing to have some remarks before the associated permit or deny statements and some remarks after the associated statements.

To include a comment for IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command. To remove the remark, use the **no** form of this command.

In this example, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith through
Switch(config)# access-list 1 deny 171.69.3.13
```

For an entry in a named IP ACL, use the **remark** access-list configuration command. To remove the remark, use the **no** form of this command.

In this example, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

Applying an IPv4 ACL to a Terminal Line

You can use numbered ACLs to control access to one or more terminal lines. You cannot apply named ACLs to lines. You must set identical restrictions on all the virtual terminal lines because a user can attempt to connect to any of them. Follow this procedure to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL.

For procedures for applying ACLs to interfaces, see the "Applying an IPv4 ACL to an Interface" section on page 6-26. For applying ACLs to VLANs, see the "Configuring VLAN Maps" section on page 6-31.

BEFORE YOU BEGIN

Configure an IPv4 ACL as described in the "Creating a Numbered Standard ACL" section on page 6-14 or "Creating a Numbered Extended ACL" section on page 6-16.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**line** [**console** | **vty**] *line-number*	Identify a specific line to configure, and enter in-line configuration mode. • **console**—Specify the console terminal line. The console port is DCE. • **vty**—Specify a virtual terminal for remote console access. The *line-number* is the first line number in a contiguous group that you want to configure when the line type is specified. The range is from 0 to 16.
Step 3	**access-class** *access-list-number* {**in** | **out**}	Restrict incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Display the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an ACL from a terminal line, use the **no access-class** *access-list-number* {**in** | **out**} line configuration command.

EXAMPLE

The following example defines an access list that permits only hosts on network 192.89.55.0 to connect to the virtual terminal ports on the switch:

```
access-list 12 permit 192.89.55.0 0.0.0.255
 line 1 5
 access-class 12 in
```

The following example defines an access list that denies connections to networks other than network 10.0.0.0 on terminal lines 1 through 5:

```
access-list 10 permit 10.0.0.0 0.255.255.255
 line 1 5
 access-class 10 out
```

Applying an IPv4 ACL to an Interface

This section describes how to apply IPv4 ACLs to network interfaces. You can apply an ACL to *either* outbound or inbound Layer 3 interfaces. You can apply ACLs only to inbound Layer 2 interfaces.

Note By default, the router sends Internet Control Message Protocol (ICMP) unreachable messages when a packet is denied by an access group. These access-group denied packets are not dropped in hardware but are bridged to the switch CPU so that it can generate the ICMP-unreachable message.

Note When you apply the **ip access-group** interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access groups filter packets that are routed or are received by Layer 3 processes on the CPU.

For inbound ACLs, after receiving a packet, the switch checks the packet against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards the packet.

For outbound ACLs, after receiving and routing a packet to a controlled interface, the switch checks the packet against the ACL. If the ACL permits the packet, the switch sends the packet. If the ACL rejects the packet, the switch discards the packet.

By default, the input interface sends ICMP unreachable messages whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP unreachable messages are normally limited to no more than one every one-half second per input interface, but this can be changed by using the **ip icmp rate-limit unreachable** global configuration command.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

BEFORE YOU BEGIN

Configure an IPv4 ACL as described in the "Creating a Numbered Standard ACL" section on page 6-14, "Creating a Numbered Extended ACL" section on page 6-16, or "Creating Named Standard and Extended ACLs" section on page 6-20.

See the "Applying an ACL to an Interface" section on page 6-10 for configuration guidelines.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Identify a specific interface for configuration, and enter interface configuration mode. The interface can be a Layer 2 interface (port ACL), or a Layer 3 interface (router ACL).
Step 3	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) and enhanced network interfaces (ENIs) are disabled, and network node interfaces (NNIs) are enabled.
Step 4	**ip access-group** {*access-list-number* | *name*} {**in** | **out**}	Control access to the specified interface. The **out** keyword is not supported for Layer 2 interfaces (port ACLs).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Display the access list configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified access group, use the **no ip access-group** {*access-list-number* | *name*} {**in** | **out**} interface configuration command.

EXAMPLE

This example shows how to apply access list 2 to a port to filter packets entering the port:

```
Switch(config)# interface gigabitethernet0/1
Router(config-if)# ip access-group 2 in
```

Troubleshooting ACLs

If this ACL manager message appears and [chars] is the access-list name,

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

The switch has insufficient resources to create a hardware representation of the ACL. The resources include hardware memory and label space but not CPU memory. A lack of available logical operation units or specialized hardware resources causes this problem. Logical operation units are needed for a TCP flag match or a test other than **eq** (**ne**, **gt**, **lt**, or **range**) on TCP, UDP, or SCTP port numbers.

Use one of these workarounds:

- Modify the ACL configuration to use fewer resources.
- Rename the ACL with a name or number that alphanumerically precedes the ACL names or numbers.

To determine the specialized hardware resources, enter the **show platform layer4 acl map** privileged EXEC command. If the switch does not have available resources, the output shows that index 0 to index 15 are not available.

For more information about configuring ACLs with insufficient resources, see CSCsq63926 in the Bug Toolkit.

For example, if you apply this ACL to an interface:

```
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
permit tcp source source-wildcard destination destination-wildcard
```

And if this message appears:

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

The flag-related operators are not available. To avoid this issue,

- Move the fourth ACE before the first ACE by using **ip access-list resequence** global configuration command:

  ```
  permit tcp source source-wildcard destination destination-wildcard
  permit tcp source source-wildcard destination destination-wildcard range 5 60
  permit tcp source source-wildcard destination destination-wildcard range 15 160
  permit tcp source source-wildcard destination destination-wildcard range 115 1660
  ```

  or

- Rename the ACL with a name or number that alphanumerically precedes the other ACLs (for example, rename ACL *79* to ACL *1*).

You can now apply the first ACE in the ACL to the interface. The switch allocates the ACE to available mapping bits in the Opselect index and then allocates flag-related operators to use the same bits in the TCAM.

Creating Named MAC Extended ACLs

You can filter non-IPv4 traffic on a VLAN or on a Layer 2 interface by using MAC addresses and named MAC extended ACLs. The procedure is similar to that of configuring other extended named ACLs.

Note You cannot apply named MAC extended ACLs to Layer 3 interfaces.

For more information about the supported non-IP protocols in the **mac access-list extended** command, see the *Cisco IOS LAN Switching Command Reference*.

Note Though visible in the command-line help strings, **appletalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac access-list extended** *name*	Define an extended MAC access list using a name.
Step 3	{**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** |**vines-ip** | **xns-idp** | *0-65535*] [**cos** *cos*]	In extended MAC access-list configuration mode, specify to **permit** or **deny any** source MAC address, a source MAC address with a mask, or a specific **host** source MAC address and **any** destination MAC address, destination MAC address with a mask, or a specific destination MAC address. (Optional) You can also enter these options: • *type mask*—An arbitrary EtherType number of a packet with Ethernet II or SNAP encapsulation in decimal, hexadecimal, or octal with optional mask of *don't care* bits applied to the EtherType before testing for a match. • **lsap** *lsap mask*—An LSAP number of a packet with IEEE 802.2 encapsulation in decimal, hexadecimal, or octal with optional mask of *don't care* bits. • **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** |**vines-ip** | **xns-idp**—A non-IP protocol. • **cos** *cos*—An 802.1Q cost of service number from 0 to 7 used to set priority.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* | *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no mac access-list extended** *name* global configuration command to delete the entire ACL. You can also delete individual ACEs from named MAC extended ACLs.

EXAMPLE

This example shows how to create and display an access list named *mac1*, denying only EtherType DECnet Phase IV traffic, but permitting all other types of traffic:

```
Switch(config)# mac access-list extended mac1
Switch(config-ext-macl)# deny any any decnet-iv
Switch(config-ext-macl)# permit any any
Switch(config-ext-macl)# end
Switch # show access-lists
Extended MAC access list mac1
    10 deny   any any decnet-iv
```

```
20 permit any any
```

Applying a MAC ACL to a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming in that interface.

After receiving a packet, the switch checks it against the inbound ACL. If the ACL permits it, the switch continues to process the packet. If the ACL rejects the packet, the switch discards it. When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Note The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface. You cannot use the command on EtherChannel port channels.

BEFORE YOU BEGIN

Configure a MAC ACL as described in the "Creating Named MAC Extended ACLs" procedure on page 6-28.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Identify a specific interface, and enter interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).
Step 3	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
Step 4	**mac access-group** {*name*} {**in**}	Control access to the specified interface by using the MAC access list. **Note** Port ACLs are supported only in the inbound direction.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mac access-group** [**interface** *interface-id*]	Display the MAC access list applied to the interface or all Layer 2 interfaces.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified access group, use the **no mac access-group** {*name*} interface configuration command.

EXAMPLE

This example shows how to apply MAC access list *mac1* to a port to filter packets entering the port:

```
Switch(config)# interface gigabitethernet0/2
Router(config-if)# mac access-group mac1 in
```

Configuring VLAN Maps

This section describes how to configure VLAN maps, which is the only way to control filtering within a VLAN. VLAN maps have no direction. To filter traffic in a specific direction by using a VLAN map, you need to include an ACL with specific source or destination addresses. If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet if the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Step 1 Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN. See the "Creating Standard and Extended IPv4 ACLs" section on page 6-13 and the "Creating Named MAC Extended ACLs" section on page 6-28.

Step 2 Enter the **vlan access-map** global configuration command to create a VLAN ACL map entry.

Step 3 In access-map configuration mode, optionally enter an **action**—**forward** (the default) or **drop**—and enter the **match** command to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended).

Note If the VLAN map has a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause and the configured action is drop, then all IP and Layer 2 packets are dropped.

Step 4 Use the **vlan filter** global configuration command to apply a VLAN map to one or more VLANs.

This section includes the following topics:

- Creating a VLAN Map, page 6-31
- Applying a VLAN Map to a VLAN, page 6-34
- Verifying Configuration, page 6-35

Creating a VLAN Map

Each VLAN map consists of an ordered series of entries.

BEFORE YOU BEGIN

Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN as described in the "Creating Standard and Extended IPv4 ACLs" section on page 6-13 or "Creating Named MAC Extended ACLs" section on page 6-28.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan access-map** *name* [*number*]	Create a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map. When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete. Entering this command changes to access-map configuration mode.
Step 3	**action** {**drop** | **forward**}	(Optional) Set the action for the map entry. The default is to forward.
Step 4	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.
Step 5	**end**	Return to global configuration mode.
Step 6	**show running-config**	Display the access list configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no vlan access-map** *name* global configuration command to delete a map.

Use the **no vlan access-map** *name number* global configuration command to delete a single sequence entry from within the map.

Use the **no action** access-map configuration command to enforce the default action, which is to forward.

VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.

EXAMPLE

Example 1

This example shows how to create an ACL and a VLAN map to deny a packet. In the first map, any packets that match the *ip1* ACL (TCP packets) would be dropped. You first create the *ip1*ACL to permit any TCP packet and no other packets. Because there is a match clause for IP packets in the VLAN map, the default action is to drop any IP packet that does not match any of the match clauses.

```
Switch(config)# ip access-list extended ip1
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 10
Switch(config-access-map)# match ip address ip1
Switch(config-access-map)# action drop
```

This example shows how to create a VLAN map to permit a packet. ACL *ip2* permits UDP packets and any packets that match the *ip2* ACL are forwarded. In this map, any IP packets that did not match any of the previous ACLs (that is, packets that are not TCP packets or UDP packets) would get dropped.

```
Switch(config)# ip access-list extended ip2
Switch(config-ext-nacl)# permit udp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 20
Switch(config-access-map)# match ip address ip2
Switch(config-access-map)# action forward
```

Example 2

In this example, the VLAN map has a default action of drop for IP packets and a default action of forward for MAC packets. Used with standard ACL 101 and extended named access lists **igmp-match** and **tcp-match**, the map will have the following results:

- Forward all UDP packets
- Drop all IGMP packets
- Forward all TCP packets
- Drop all other IP packets
- Forward all non-IP packets

```
Switch(config)# access-list 101 permit udp any any
Switch(config)# ip access-list extended igmp-match
Switch(config-ext-nacl)# permit igmp any any
Switch(config)# ip access-list extended tcp-match
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-ip-default 10
Switch(config-access-map)# match ip address 101
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 20
Switch(config-access-map)# match ip address igmp-match
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 30
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
```

Example 3

In this example, the VLAN map has a default action of drop for MAC packets and a default action of forward for IP packets. Used with MAC extended access lists **good-hosts** and **good-protocols**, the map will have the following results:

- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Forward MAC packets with decnet-iv or vines-ip protocols
- Drop all other non-IP packets
- Forward all IP packets

```
Switch(config)# mac access-list extended good-hosts
Switch(config-ext-macl)# permit host 000.0c00.0111 any
Switch(config-ext-macl)# permit host 000.0c00.0211 any
Switch(config-ext-macl)# exit
Switch(config)# mac access-list extended good-protocols
Switch(config-ext-macl)# permit any any decnet-ip
Switch(config-ext-macl)# permit any any vines-ip
Switch(config-ext-nacl)# exit
```

```
Switch(config)# vlan access-map drop-mac-default 10
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-mac-default 20
Switch(config-access-map)# match mac address good-protocols
Switch(config-access-map)# action forward
```

Example 4

In this example, the VLAN map has a default action of drop for all packets (IP and non-IP). Used with access lists **tcp-match** and **good-hosts** from Examples 2 and 3, the map will have the following results:

- Forward all TCP packets
- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Drop all other IP packets
- Drop all other MAC packets

```
Switch(config)# vlan access-map drop-all-default 10
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-all-default 20
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
```

Applying a VLAN Map to a VLAN

BEFORE YOU BEGIN

Configure a VLAN map as described in the "Configuring VLAN Maps" section on page 6-31.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan filter** *mapname* **vlan-list** *list*	Apply the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.
Step 3	**show running-config**	Display the access list configuration.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the VLAN map, use the **no vlan filter** *mapname* **vlan-list** *list* global configuration command.

EXAMPLE

This example shows how to apply VLAN map 1 to VLANs 20 through 22:

```
Switch(config)# vlan filter map 1 vlan-list 20-22
```

Verifying Configuration

Command	Purpose
show access-lists [*number* | *name*]	Displays the contents of one or all current IP and MAC address access lists or a specific access list (numbered or named).
show ip access-lists [*number* | *name*]	Displays the contents of all current IP access lists or a specific IP access list (numbered or named).
show running-config [**interface** *interface-id*]	Displays the contents of the configuration file for the switch or the specified interface, including all configured MAC and IP access lists and which access groups are applied to an interface.
show mac access-group [**interface** *interface-id*]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.
show vlan access-map [*mapname*]	Shows information about all VLAN access-maps or the specified access map.
show vlan filter [**access-map** *name* | **vlan** *vlan-id*]	Shows information about all VLAN filters or about a specified VLAN or VLAN access map.

Configuration Example

- IPv4 ACL Configuration Examples, page 6-35
- Using VLAN Maps in Your Network, page 6-40

IPv4 ACL Configuration Examples

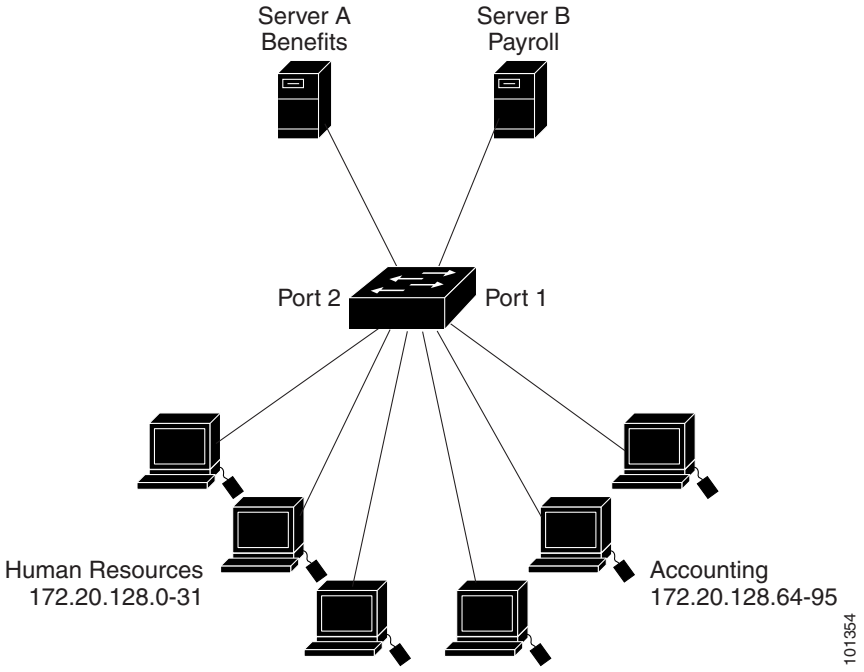
This section provides examples of configuring and applying IPv4 ACLs. For detailed information about compiling ACLs, see the *Security Configuration Guide: Access Control Lists, Cisco IOS Release 15M&T*.

Figure 6-6 shows a small networked office environment with routed Port 2 connected to Server A, containing benefits and other information that all employees can access, and routed Port 1 connected to Server B, containing confidential payroll data. All users can access Server A, but Server B has restricted access.

Use router ACLs to do this in one of two ways:

- Create a standard ACL, and filter traffic coming to the server from Port 1.
- Create an extended ACL, and filter traffic coming from the server into Port 1.

Figure 6-6 Using Router ACLs to Control Traffic



This example uses a standard ACL to filter traffic coming into Server B from a port, permitting traffic only from Accounting's source addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic coming out of routed Port 1 from the specified source address.

```
Switch(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Standard IP access list 6
    10 permit 172.20.128.64, wildcard bits 0.0.0.31
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 6 out
```

This example uses an extended ACL to filter traffic coming from Server B into a port, permitting traffic from any source address (in this case Server B) to only the Accounting destination addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic going into routed Port 1, permitting it to go only to the specified destination addresses. Note that with extended ACLs, you must enter the protocol (IP) before the source and destination information.

```
Switch(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Extended IP access list 106
    10 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 106 in
```

Numbered ACLs

In this example, network 36.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 36.0.0.0 address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and reject all others on that subnet. The last line of the list shows that the switch accepts addresses on all other network 36.0.0.0 subnets. The ACL is applied to packets entering a port.

```
Switch(config)# access-list 2 permit 36.48.0.3
Switch(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Switch(config)# access-list 2 permit 36.0.0.0 0.255.255.255
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 2 in
```

Extended ACLs

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# access-list 102 permit icmp any any
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 102 in
```

For another example of using an extended ACL, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets have the port numbers reversed. Because the secure system of the network always accepts mail connections on port 25, the incoming and outgoing services are separately controlled. The ACL must be configured as an input ACL on the outbound interface and an output ACL on the inbound interface.

In this example, the network is a Class B network with the address 128.88.0.0, and the mail host address is 128.88.1.2. The **established** keyword is used only for the TCP to show an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which show that the packet belongs to an existing connection. Gigabit Ethernet interface 1 is the interface that connects the router to the Internet.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 102 in
```

Named ACLs

This example creates a standard ACL named *internet_filter* and an extended ACL named *marketing_group*. The *internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Switch(config)# ip access-list standard Internet_filter
Switch(config-ext-nacl)# permit 1.2.3.4
Switch(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Switch(config)# ip access-list extended marketing_group
Switch(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Switch(config-ext-nacl)# deny tcp any any
Switch(config-ext-nacl)# permit icmp any any
Switch(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Switch(config-ext-nacl)# deny ip any any log
Switch(config-ext-nacl)# exit
```

The *Internet_filter* ACL is applied to outgoing traffic and the *marketing_group* ACL is applied to incoming traffic on a Layer 3 port.

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 2.0.5.1 255.255.255.0
Switch(config-if)# ip access-group Internet_filter out
Switch(config-if)# ip access-group marketing_group in
```

Time Range Applied to an IP ACL

This example denies HTTP traffic on IP on Monday through Friday between the hours of 8:00 a.m. and 6:00 p.m (18:00). The example allows UDP traffic only on Saturday and Sunday from noon to 8:00 p.m. (20:00).

```
Switch(config)# time-range no-http
Switch(config)# periodic weekdays 8:00 to 18:00
!
Switch(config)# time-range udp-yes
Switch(config)# periodic weekend 12:00 to 20:00
!
Switch(config)# ip access-list extended strict
Switch(config-ext-nacl)# deny tcp any any eq www time-range no-http
Switch(config-ext-nacl)# permit udp any any time-range udp-yes
!
Switch(config-ext-nacl)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group strict in
```

Commented IP ACL Entries

In this example of a numbered ACL, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith workstation through
Switch(config)# access-list 1 deny 171.69.3.13
```

In this example of a numbered ACL, the Winter and Smith workstations are not allowed to browse the web:

```
Switch(config)# access-list 100 remark Do not allow Winter to browse the web
Switch(config)# access-list 100 deny host 171.69.3.85 any eq www
Switch(config)# access-list 100 remark Do not allow Smith to browse the web
Switch(config)# access-list 100 deny host 171.69.3.13 any eq www
```

In this example of a named ACL, the Jones subnet is not allowed access:

```
Switch(config)# ip access-list standard prevention
Switch(config-std-nacl)# remark Do not allow Jones subnet through
Switch(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

In this example of a named ACL, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

ACL Logging

Two variations of logging are supported on router ACLs. The **log** keyword sends an informational logging message to the console about the packet that matches the entry; the **log-input** keyword includes the input interface in the log entry.

In this example, standard named access list *stan1* denies traffic from 10.1.1.0 0.0.0.255, allows traffic from all other sources, and includes the **log** keyword.

```
Switch(config)# ip access-list standard stan1
Switch(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Switch(config-std-nacl)# permit any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group stan1 in
Switch(config-if)# end
Switch# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 37 messages logged
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 37 messages logged
    File logging: disabled
    Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

This example is a named extended access list *ext1* that permits ICMP packets from any source to 10.1.1.0 0.0.0.255 and denies all UDP packets:

```
Switch(config)# ip access-list extended ext1
Switch(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Switch(config-ext-nacl)# deny udp any any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip access-group ext1 in
```

This is a an example of a log for an extended ACL:

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
```

```
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1
packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8
packets
```

Note that all logging entries for IP ACLs start with `%SEC-6-IPACCESSLOG` with minor variations in format depending on the kind of ACL and the access entry that has been matched.

This is an example of an output message when the **log-input** keyword is entered:

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1
0001.42ef.a400) -> 10.1.1.61 (0/0), 1 packet
```

A log message for the same sort of packet using the **log** keyword does not include the input interface information:

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0), 1
packet
```

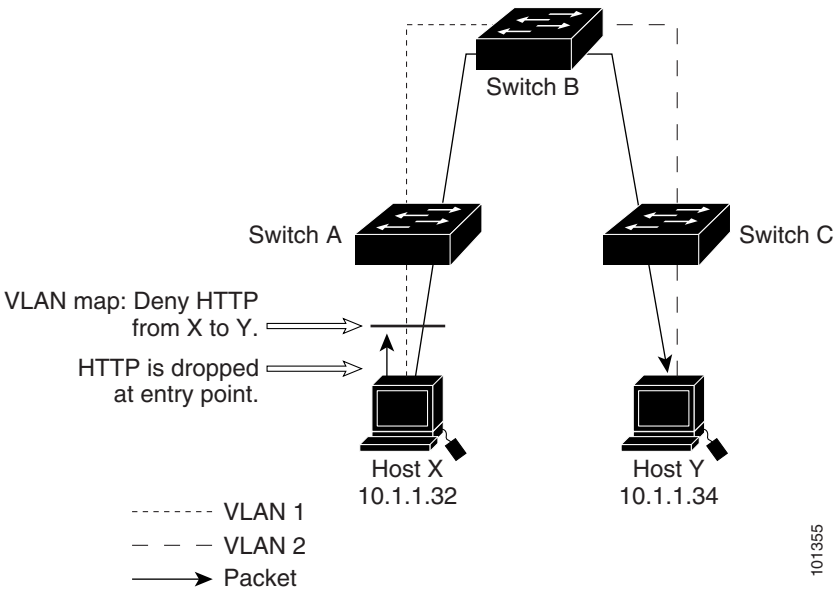
Using VLAN Maps in Your Network

- Wiring Closet Configuration, page 6-40
- Denying Access to a Server on Another VLAN, page 6-41

Wiring Closet Configuration

In a wiring closet configuration, routing might not be enabled on the switch. In this configuration, the switch can still support a VLAN map and a QoS classification ACL. In Figure 6-7, assume that Host X and Host Y are in different VLANs and are connected to wiring closet switches A and C. Traffic from Host X to Host Y is eventually being routed by Switch B, a Layer 3 switch with routing enabled. Traffic from Host X to Host Y can be access-controlled at the traffic entry point, Switch A.

Figure 6-7 Wiring Closet Configuration



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not forward it to Switch B.

1. First, define the IP access list *http* that permits (matches) any TCP traffic on the HTTP port.

```
Switch(config)# ip access-list extended http
Switch(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Switch(config-ext-nacl)# exit
```

2. Next, create VLAN access map *map2* so that traffic that matches the *http* access list is dropped and all other IP traffic is forwarded.

```
Switch(config)# vlan access-map map2 10
Switch(config-access-map)# match ip address http
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# ip access-list extended match_all
Switch(config-ext-nacl)# permit ip any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map2 20
Switch(config-access-map)# match ip address match_all
Switch(config-access-map)# action forward
```

3. Then, apply VLAN access map *map2* to VLAN 1.

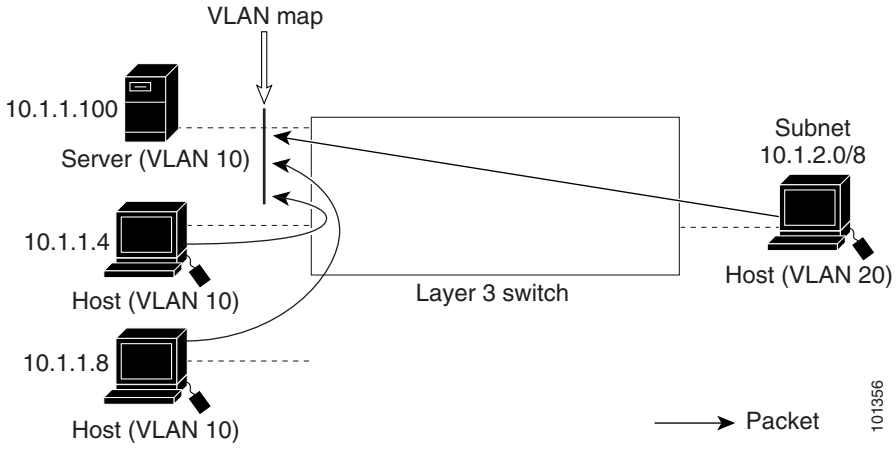
```
Switch(config)# vlan filter map2 vlan 1
```

Denying Access to a Server on Another VLAN

You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access denied to these hosts (see Figure 6-8):

- Hosts in subnet 10.1.2.0/8 in VLAN 20 should not have access.
- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.

Figure 6-8 Deny Access to a Server on Another VLAN



This example shows how to deny access to a server on another VLAN by creating the VLAN map SERVER 1 that denies access to hosts in subnet 10.1.2.0.8, host 10.1.1.4, and host 10.1.1.8 and permits other IP traffic. The final step is to apply the map SERVER1 to VLAN 10.

Step 1 Define the IP ACL that will match the correct packets.

```
Switch(config)# ip access-list extended SERVER1_ACL
Switch(config-ext-nacl))# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.4 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.8 host 10.1.1.100
Switch(config-ext-nacl))# exit
```

Step 2 Define a VLAN map using this ACL that will drop IP packets that match SERVER1_ACL and forward IP packets that do not match the ACL.

```
Switch(config)# vlan access-map SERVER1_MAP
Switch(config-access-map)# match ip address SERVER1_ACL
Switch(config-access-map)# action drop
Switch(config)# vlan access-map SERVER1_MAP 20
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
```

Step 3 Apply the VLAN map to VLAN 10.

```
Switch(config)# vlan filter SERVER1_MAP vlan-list 10.
```

Related Documents

- Cisco IOS Master Command List, All Releases
- Security Configuration Guide: Access Control Lists, Cisco IOS Release 15M&T
- Cisco IOS Security Command Reference
- Cisco IOS LAN Switching Command Reference
- *Security Configuration Guide: Access Control Lists, Cisco IOS Release 15M&T.*
  - "IP Access List Entry Sequence Numbering"

Configuring IPv6 ACLs

This chapter provides details about configuring IPv6 access control lists (ACLs) on the Cisco Industrial Ethernet 2000U Series Switch (IE 2000U), hereafter referred to as IE 2000U or switch.

When the IE2000U is running the IP services image:

- You can filter IPv6 traffic by creating IPv6 ACLs and applying them to interfaces
- You can create and apply input router ACLs to filter Layer 3 management traffic

This chapter contains these sections:

- Information About IPv6 ACLs, page 7-1
- Prerequisites, page 7-2
- Guidelines and Limitations, page 7-2
- Default Settings, page 7-3
- Configuring IPv6 ACLs, page 7-3
- Verifying IPv6 ACLs, page 7-8
- Configuration Example, page 7-9
- Related Documents, page 7-9

Information About IPv6 ACLs

A switch running the IP services image supports two types of IPv6 ACLs:

- IPv6 *router ACLs* on outbound or inbound traffic on Layer 3 interfaces only, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels.

  IPv6 router ACLs apply only to routed IPv6 packets.
- *IPv6 port ACLs* on inbound traffic on Layer 2 interfaces only. The switch applies IPv6 port ACLs to all IPv6 packets entering the interface.

Note For more information about IPv4 ACL support on the switch, see Related Documents.

You can apply both IPv4 and IPv6 ACLs to an interface.

As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs:

- When an input router ACL and input port ACL exist in an SVI, packets received on ports to which a port ACL is applied are filtered by the port ACL. Routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IPv6 packets are filtered by the router ACL. Other packets are not filtered.

Note When you apply *any* port ACL (IPv4, IPv6, or MAC) to an interface, that port ACL filters packets, and ignores any router ACLs attached to the SVI of the port VLAN.

Supported ACL Features

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the **fragments** keyword as in IPv4) are supported.
- The same statistics supported in IPv4 are supported for IPv6 ACLs.
- If the switch runs out of hardware space, packets associated with the ACL are forwarded to the CPU, and the software applies the ACLs.
- Routed or bridged packets with hop-by-hop options have IPv6 ACLs applied in software.
- Logging is supported for router ACLs, but not for port ACLs.
- The switch supports IPv6 address-matching for a full range of prefix-lengths.

Note For items not supported for IPv6 ACLS, see Guidelines and Limitations.

Prerequisites

Be sure to review Guidelines and Limitations and the Before You Begin section within each configuration section before configuring a feature.

Guidelines and Limitations

ACLs for IPv6 Traffic Not Supported

- The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.
- The switch does not apply MAC-based ACLs on IPv6 frames.
- You cannot apply IPv6 port ACLs to Layer 2 EtherChannels.
- The switch does not support output port ACLs.

Cisco IOS IPv6 ACLs Functions Not Supported

- The switch does not support matching on these keywords: **flowlabel**, **routing header**, and **undetermined-transport**.

- The switch does not support reflexive ACLs (the **reflect** keyword).

Access Control Entry (ACE) and ACLs

- When you apply an ACL to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the switch does not allow the ACE to be added to the attached ACL.

Named ACLs

- IPv6 supports only named ACLs.

IPv6 ACLs Interactions With Other Switches or Features

- When you configure an IPv6 router ACL to deny a packet, the software does not route the packet. Instead, the software forwards a copy of the packet to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.
- You can create both IPv4 and IPv6 ACLs on a switch, and you can apply both IPv4 and IPv6 ACLs to the same interface.
  - Each ACL must have a unique name; and, an error message appears if you try to use a name that already exists on the switch.
  - You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface.

    If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.
- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, for any additional configured ACLs, the switch forwards the packets to the CPU, and the software applies the ACLs.

Default Settings

Parameters	Default
IPv6 ACLs	There are no default IPv6 ACLs configured or applied on the switch.

Configuring IPv6 ACLs

This section includes the following topics:

- Creating IPv6 ACLs, page 7-4
- Applying an IPv6 ACL to an Interface, page 7-8

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

Select one of the dual IPv4 and IPv6 SDM templates. (See Related Documents for configuration details.)

Creating IPv6 ACLs

Note When you configure an unsupported IPv6 ACL, an error message appears, and the configuration does not take affect.

Use the **no** {**deny** | **permit**} IPv6 access-list configuration commands with keywords to remove the deny or permit conditions from the specified access list for the commands below.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 access-list** *access-list-name*	Define an IPv6 access list using a name, and enter IPv6 access-list configuration mode.

	Command	Purpose
Step 3a	{**deny** | **permit**} *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix*/ *prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]	Deny or permit the packet, when specified conditions are matched. These are the conditions: • *protocol*–Name or number of an Internet protocol: **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **stcp**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IPv6 protocol number. For additional specific parameters for ICMP, TCP, and UDP, see Steps 3b through 3d. • *source-ipv6-prefix*/*prefix-length* or *destination-ipv6-prefix*/ *prefix-length*–Source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons. • Enter **any** as an abbreviation for the IPv6 prefix ::/0. • **host** *source-ipv6-address* or *destination-ipv6-address*– Define source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons. • (Optional) *operator*–Operand that compares the source or destination ports of the specified protocol such as **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range**. If the operator follows the *source-ipv6-prefix*/*prefix-length* argument, it must match the source port. If the operator follows the *destination-ipv6- prefix*/*prefix-length* argument, it must match the destination port. • (Optional) *port-number*– Value of 0 to 65535 or TCP or UDP port name. Use TCP port names only when filtering TCP. Use UDP port names only when filtering UDP. • (Optional) **dscp** *value*–Match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) **fragments**–Check noninitial fragments. Keyword is only visible when the protocol is **ipv6**. • (Optional, router ACLs only) **log**–Send a logging message to the console about the packet that matches the entry. Enter **log-input** to include the input interface in the log entry. • (Optional) **routing**–Specify routing of IPv6 packets. • (Optional) **sequence** *value*–Specify the sequence number for the access list statement. Value range is from 1 to 4294967295. • (Optional) **time-range** *name*–Specify the time range that applies to the deny or permit statement.

	Command	Purpose
Step 3b	{**deny** | **permit**} **tcp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**ack**] [**dscp** *value*] [**established**] [**fin**] [**log**] [**log-input**] [**neq** {*port* | *protocol*}] [**psh**] [**range** {*port* | *protocol*}] [**rst**] [**routing**] [**sequence** *value*] [**syn**] [**time-range** *name*] [**urg**]	(Optional) Define a TCP access list and the access conditions. Enter **tcp** for Transmission Control Protocol. The parameters are the same as those described in Step 3a, with these additional optional parameters: • **ack**—Acknowledgment bit set. • **established**—An established connection. A match occurs if the TCP datagram has the ACK or RST bits set. • **fin**—Finished bit set; no more data from sender. • **neq** {*port* | *protocol*}—Match only packets that are not on a given port number. • **psh**—Push function bit set. • **range** {*port* | *protocol*}—Match only packets in the port number range. • **rst**—Reset bit set. • **syn**—Synchronize bit set. • **urg**—Urgent pointer bit set.
Step 3c	{**deny** | **permit**} **udp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**log**] [**log-input**] [**neq** {*port* | *protocol*}] [**range** {*port* | *protocol*}] [**routing**] [**sequence** *value*] [**time-range** *name*]	(Optional) Define a UDP access list and the access conditions. Enter **udp** for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the [*operator* [*port*]] port number or name must be a UDP port number or name, and the **established** parameter is not valid for UDP.

	Command	Purpose
Step 3d	{**deny** | **permit**} **icmp** {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix*/*prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [*icmp-type* [*icmp-code*] | *icmp-message*] [**dscp** *value*] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]	(Optional) Define an ICMP access list and the access conditions. Enter **icmp** for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 3a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: • *icmp-type*—Enter to filter by ICMP message type, a number from 0 to 255. • *icmp-code*—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • *icmp-message*—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type names and code names, use the ? key or see command reference for this release.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ipv6 access-list**	Verify the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example:

- Creates an IPv6 ACL named CISCO.
- Defines one deny entry that denies all packets that have a destination TCP port number greater than 5000 and a second deny entry that denies packets that have a source UDP port number less than 5000. The second deny entry also logs all matches to the console.
- Defines a permit entry to permit all ICMP packets and another permit entry that allows all other traffic. The second permit entry is necessary because an implicit deny-all condition is at the end of each IPv6 access list.

```
Switch(config)# ipv6 access-list CISCO
Switch(config-ipv6-acl)# deny tcp any any gt 5000
Switch config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Switch(config-ipv6-acl)# permit icmp any any
Switch(config-ipv6-acl)# permit any any
```

Applying an IPv6 ACL to an Interface

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Identify a Layer 2 interface (for port ACLs) or Layer 3 interface (for router ACLs) on which to apply an access list, and enter interface configuration mode.
Step 3	**no switchport**	If applying a router ACL, change the interface from Layer 2 mode (the default) to Layer 3 mode.
Step 4	**ipv6 address** *ipv6-address*	Configure an IPv6 address on a Layer 3 interface (for router ACLs). **Note** This command is not required on Layer 2 interfaces or if the interface is already configured with an explicit IPv6 address. Use the **no ipv6 traffic-filter** *access-list-name* interface configuration command to remove an access list from an interface.
Step 5	**ipv6 traffic-filter** *access-list-name* {**in** | **out**}	Apply the access list to incoming or outgoing traffic on the interface. **Note** The **out** keyword is not supported for Layer 2 interfaces (port ACLs).
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify the access list configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to apply the access list CISCO to outbound traffic on a Layer 3 interface:

```
Switch(config)# interface gigabitethernet 0/3
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```

Verifying IPv6 ACLs

You can display information about all configured access lists, all IPv6 access lists, or a specific access list by using one or more of the following privileged EXEC commands.

show access-lists	Display all access lists configured on the switch.
show ipv6 access-list [*access-list-name*]	Display all configured IPv6 access list or the access list specified by name.

Configuration Example

The following example:

- Creates an IPv6 ACL named CISCO.
- Defines one deny entry that denies all packets that have a destination TCP port number greater than 5000 and a second deny entry that denies packets that have a source UDP port number less than 5000. The second deny entry also logs all matches to the console.
- Defines a permit entry to permit all ICMP packets and another permit entry that allows all other traffic. The second permit entry is necessary because an implicit deny-all condition is at the end of each IPv6 access list.
- Applies the access list CISCO to outbound traffic on a Layer 3 interface.

```
Switch(config)# ipv6 access-list CISCO
Switch(config-ipv6-acl)# deny tcp any any gt 5000
Switch config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Switch(config-ipv6-acl)# permit icmp any any
Switch(config-ipv6-acl)# permit any any
Switch(config-ipv6-acl)# exit
Switch(config)# interface gigabitethernet 0/3
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```

Related Documents

You can find all of the related documents at www.cisco.com/go/ie2000u-docs.

- *Cisco Connected Grid Switches Security Software Configuration Guide*
  - "Configuring Network Security with ACLs"
- *Cisco Connected Grid Switches System Management Software Configuration Guide*
  - "Configuring SDM Templates"
- *Cisco Connected Grid Switches Unicast Routing Software Configuration Guide*
  - "Configuring IPv6 Unicast Routing

Configuring Control-Plane Security

This chapter provides details about configuring Control-Plane Security on the Cisco Industrial Ethernet 2000U Series Switches (IE 2000U) and includes the following sections:

- Information About Control-Plane Security, page 8-1
- Prerequisites, page 8-5
- Guidelines and Limitations, page 8-5
- Default Settings, page 8-6
- Configuring Control-Plane Security, page 8-6
- Verifying Configuration, page 8-8
- Configuration Example, page 8-8
- Related Documents, page 8-8

Information About Control-Plane Security

In any network, Layer 2 and Layer 3 switches exchange control packets with other switches in the network.

The switch, which acts as a transition between the customer network and the service-provider network, uses control-plane security to isolate the topology information between the two networks. This mechanism protects against a possible denial-of-service (DoS) attack from another customer network.

NNIs, UNIs, and ENIs

In the switch, ports configured as network node interfaces (NNIs) connect to the service-provider network. The switch communicates with the rest of the network through these ports, exchanging protocol control packets as well as regular traffic. Other ports on the switch are user network interfaces (UNIs) that are used as customer-facing ports. Each port is connected to a single customer, and exchanging network protocol control packets between the switch and the customer is not usually required. Most Layer 2 protocols are not supported on UNIs. To protect against accidental or intentional CPU overload, the switch provides control-plane security automatically by dropping or rate-limiting a predefined set of Layer 2 control packets and some Layer 3 control packets for UNIs.

You can also configure a third port type, an enhanced network interface (ENI). An ENI, like a UNI, is a customer-facing interface. By default on an ENI, Layer 2 control protocols, such as Cisco Discovery Protocol (CDP), Spanning-Tree Protocol (STP), and Link Layer Discovery Protocol (LLDP), are

disabled. On ENIs, unlike UNIs, you can enable these protocols. When configuring ENIs in port channels, you can also enable Link Aggregation Control Protocol (LACP) and Port Aggregation Protocol (PAgP). ENIs drop or rate-limit the protocol packets, depending on whether the protocol is enabled or disabled on the interface. For all other control protocols on ENIs, the switch drops or rate-limits packets the same way as it does for UNIs.

CPU protection, enabled by default, uses 19 policers per port. When enabled, you can configure a maximum of 45 policers per port. If you need to configure more policers per port, you can disable CPU protection by entering the **no policer cpu uni all** global configuration command and reloading the switch. When CPU protection is disabled, you can configure a maximum of 63 policers per port for user-defined classes and one for class-default.

Note When you disable CPU protection on the switch, protocol packets can reach the CPU, which could cause CPU processing overload and storm control through software.

Control-plane security is supported on a port for Layer 2 control packets and non-IP packets with router MAC addresses, regardless of whether the port is in routing or nonrouting mode. (A port is in routing mode when global IP routing is enabled and the port is configured with the **no switchport** interface configuration command or is associated with a VLAN that has an active switch virtual interface [SVI].) These packets are either dropped or rate-limited, depending upon the Layer 2 protocol configuration.

For Layer 3 control packets, on a port in routing mode (whether or not a Layer 3 service policy is attached), control-plane security supports rate-limiting only for Internet Group Management Protocol (IGMP) control packets. For Layer 3 packets, on a port in nonrouting mode (whether or not a Layer 2 service policy is attached), only IP packets with router MAC addresses are dropped.

These types of control packets are dropped or rate-limited:

- Layer 2 protocol control packets:
  - Control packets that are always dropped on UNIs and ENIs, such as Dynamic Trunking Protocol (DTP) packets and some bridge protocol data units (BPDUs).
  - Control packets that are dropped by default but can be enabled or tunneled, such as CDP, STP, LLDP, VLAN Trunking Protocol (VTP), UniDirectional Link Detection (UDLD) Protocol, LACP, and PAgP packets. When enabled, these protocol packets are rate-limited and tunneled through the switch.
  - Control or management packets that are required by the switch, such as keepalive packets. These control packets are processed by the CPU but are rate-limited to normal and safe limits to prevent CPU overload.
- Non-IP packets with router MAC addresses
- IP packets with router MAC addresses
- IGMP control packets that are enabled by default and need to be rate-limited. However, when IGMP snooping and IP multicast routing are disabled, the packets are treated like data packets, and no policers are assigned to them.

The switch uses policing to accomplish control-plane security by either dropping or rate-limiting Layer 2 control packets. If a Layer 2 protocol is enabled on a UNI or ENI port or tunneled on the switch, those protocol packets are rate-limited; otherwise control packets are dropped.

By default, some protocol traffic is dropped by the CPU, and some is rate-limited. Table 8-1 shows the default action and the action taken for Layer 2 protocol packets when the feature is enabled or when Layer 2 protocol tunneling is enabled for the protocol. Note that some features cannot be enabled on UNIs, and not all protocols can be tunneled (shown by dashes). If Layer 2 protocol tunneling is enabled

for *any* of the supported protocols (CDP, STP, VTP, LLDP, LACP, PAgP, or UDLD), the switch Layer 2 protocol tunneling protocol uses the rate-limiting policer on every port. If UDLD is enabled on a port or UDLD tunneling is enabled, UDLD packets are rate-limited.

STP	Dropped	Rate limited	Rate-limited
RSVD_STP (reserved IEEE 802.1D addresses)	Dropped	When the Ethernet Link Management Interface (ELMI) is enabled, globally or on a per-port basis whichever is configured last, a throttle policer is assigned to a port. When ELMI is disabled (globally or on a port, whichever is configured last), a drop policer is assigned to a port.	–
PVST+	Dropped	–	Rate limited
LACP	Dropped	Rate limited	Rate limited
PAgP	Dropped	Rate limited	Rate limited
IEEE 802.1x	Dropped	Rate limited	–
CDP	Dropped	Rate limited	Rate limited
LLDP	Dropped	Rate limited	Rate limited
DTP	Dropped	–	–
UDLD	Dropped	Rate limited	Rate limited
VTP	Dropped	–	Rate limited
CISCO_L2 (any other Cisco Layer 2 protocols with the MAC address 01:00:0c:cc:cc:cc)	Dropped	–	Rate limited if CDP, DTP, UDLD, PAGP, or VTP are Layer 2 tunneled
KEEPALIVE (MAC address, SNAP encapsulation, LLC, Org ID, or HDLC packets)	Rate-limited	–	–

The switch automatically allocates 27 control-plane security policers for CPU protection. At system bootup, it assigns a policer to each port numbered 0 to 26. The policer assigned to a port determines if the protocol packets arriving on the port are rate-limited or dropped. On the switch, a policer of 26 means a drop policer and is a global policer; any traffic type shown as 26 on any port is dropped. A policer of

a value of 0 to 25 means that a rate-limiting policer is assigned to the port for the protocol. The policers 0 to 23 are logical identifiers for Fast Ethernet ports 1 to 24; policers 24 and 25 refer to Gigabit Ethernet ports 1 and 2, respectively. A policer value of 255 means that no policer is assigned to a protocol.

To see what policer actions are assigned to the protocols on an interface, enter the **show platform policer cpu interface** *interface-id* privileged EXEC command.

This example shows the default policer configuration for a UNI. Because the port is Fast Ethernet 1, the identifier for rate-limited protocols is 0; a display for Fast Ethernet port 5 would display an identifier of 4. The *Policer Index* refers to the specific protocol. The ASIC number shows when the policer is on a different ASIC.

Because UNIs do not support STP, CDP, LLDP, LACP, or PAgP, these packets are dropped (physical policer of 26). These protocols are disabled by default on ENIs as well, but you can enable them. When enabled on ENIs, the control packets are rate limited and a rate-limiting policer is assigned to the port for these protocols (physical policer of 22).

```
Switch# show platform policer cpu interface fastethernet 0/3
Policers assigned for CPU protection
=====================================================================
 Feature                          Policer        Physical      Asic
                                  Index          Policer       Num
=====================================================================
Fa0/1
STP                                  1            26             0
LACP                                 2            26             0
8021X                                3            26             0
RSVD_STP                             4            26             0
PVST_PLUS                            5            26             0
CDP                                  6            26             0
LLDP                                 7            26             0
DTP                                  8            26             0
UDLD                                 9            26             0
PAGP                                10            26             0
VTP                                 11            26             0
CISCO_L2                            12            26             0
KEEPALIVE                           13             0             0
CFM                                 14           255             0
SWITCH_MAC                          15            26             0
SWITCH_ROUTER_MAC                   16            26             0
SWITCH_IGMP                         17             0             0
SWITCH_L2PT                         18            26             0
```

This example shows the policers assigned to a ENI when control protocols are enabled on the interface. A value of 22 shows that protocol packets are rate limited for that protocol. When the protocol is not enabled, the defaults are the same as for a UNI.

```
Switch# show platform policer cpu interface fastethernet0/23
Policers assigned for CPU protection
=====================================================================
 Feature                          Policer        Physical      Asic
                                  Index          Policer       Num
=====================================================================
Fa0/23
STP                                  1            26             0
LACP                                 2            22             0
8021X                                3            26             0
RSVD_STP                             4            26             0
PVST_PLUS                            5            26             0
CDP                                  6            22             0
LLDP                                 7            26             0
DTP                                  8            26             0
UDLD                                 9            26             0
```

```
PAGP                                10            26            0
VTP                                 11            26            0
CISCO_L2                            12            22            0
KEEPALIVE                           13            22            0
CFM                                 14           255            0
SWITCH_MAC                          15            26            0
SWITCH_ROUTER_MAC                   16            26            0
SWITCH_IGMP                         17            22            0
SWITCH_L2PT                         18            22            0
```

This example shows the default policers assigned to NNIs. Most protocols have no policers assigned to NNIs. A value of 255 means that no policer is assigned to the port for the protocol.

```
Switch #show platform policer cpu interface gigabitethernet 0/1
Policers assigned for CPU protection
=====================================================================
 Feature                          Policer        Physical      Asic
                                  Index          Policer       Num
=====================================================================
Gi0/1
STP                                 1            255            0
LACP                                2            255            0
8021X                               3            255            0
RSVD_STP                            4            255            0
PVST_PLUS                           5            255            0
CDP                                 6            255            0
LLDP                                7            255            0
DTP                                 8            255            0
UDLD                                9            255            0
PAGP                               10            255            0
VTP                                11            255            0
CISCO_L2                           12            255            0
KEEPALIVE                          13            255            0
CFM                                14            255            0
SWITCH_MAC                         15            255            0
SWITCH_ROUTER_MAC                  16            255            0
SWITCH_IGMP                        17            255            0
SWITCH_L2PT                        18            255            0
```

Prerequisites

Be sure to review the Guidelines and Limitations section and the Before You Begin section for each configuration.

Guidelines and Limitations

Reload Required When You Disable or Reenable CPU Protection

When you disable or enable CPU protection (**[no] policer cpu uni all**), you must reload the switch by entering the **reload** privileged EXEC command before the configuration takes effect.

Policer Support Per Port for CPU Protection

- When you enable (system default) CPU protection on the switch, you can configure a maximum of 45 policers per port.

- When you disable CPU protection allows, you to configure a maximum of 63 policers per port for user-defined classes and one for class-default.
  - Due to hardware limitations, you can attach 64 per-port, per-VLAN policers to a maximum of 6 ports. If you attempt to attach more than 6 per-port per VLAN 64-policer policy maps, the attachment fails with a *VLAN labels exceeded* error message.
  - When you disable CPU protection and attach a policy map with more than 45 policers, and then enable CPU protection again and reload, 19 policers per port are again required for CPU protection. During reload, the policers 46 and above will reach the *policer resources exceeded* error condition and no policers are attached to those classes.

Rate-limiting Threshold

The rate-limiting threshold applies to all supported control protocols on all UNIs and ENIs. It also applies to STP, CDP, LLDP, LACP, and PAgP when the protocol is enabled on an ENI.

You can configure only the rate-limiting threshold.

Ping Limitations

During normal Layer 2 operation, you cannot ping (**ping** [*host*] | *address*] the switch through a UNI or ENI.

This restriction does not apply to NNIs.

Default Settings

Parameters	Default
CPU protection	Enabled.
CPU policiers	A total of 27 pre-allocated to each port (0 to 26).

Configuring Control-Plane Security

You can set the threshold rate for CPU protection.

BEFORE YOU BEGIN

Review the Guidelines and Limitations for this feature.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**policer cpu uni** *rate*	Configure the CPU protection policing threshold rate. The range is from 8000 to 409500 bits per second (b/s). The default, if none is configured, is 160000 b/s. **Note** The configured rate applies to all supported and enabled control protocols on all UNIs and ENIs. To return to the default threshold rate, use the **no policer cpu uni** global configuration command. To disable CPU protection, enter the **no policer cpu uni all** global configuration command, and **reload** the switch.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show policer cpu uni-eni rate**	Verify the configured CPU policer rate.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to set the CPU protection threshold to 10000 b/s and verify the configuration:

```
Switch# config t
Switch(config)# policer cpu uni 10000
Switch(config)# end
Switch# show policer cpu uni-eni rate
CPU UNI/ENI port police rate = 10000 bps
```

This is an example of the **show** command output when you disable CPU protection on the switch:

```
Switch# show policer cpu uni-eni rate
CPU Protection feature is not enabled
```

Verifying Configuration

You can verify control-plane security settings on the switch or on an interface.

For details on **clear** and **debug** commands, see Clear and Debug Commands.

Command	Purpose
show platform policer cpu {**classification** | **interface** *interface-id*}	Display control-plane policer information. • **classification**—show classification statistics. • **interface** *interface-id*—show policer indexes for the specified interface.

show policer cpu uni-eni {**drop** [**interface** *interface-id*] | **rate**}	Display CPU policer information for the switch.
	• **drop** [**interface** *interface-id*] —shows the number of dropped frames for all interfaces or the specified interface.
	• **rate**—shows the configured threshold rate for CPU policers.
	If CPU protection is disabled on the switch, this message appears in the output:
	`Switch# show policer cpu uni drop`
	`CPU Protection feature is not enabled`

Clear and Debug Commands

Command	Purpose
clear policer cpu uni-eni counters {**classification** | **drop**}	Clear all control-plane statistics per feature (**classification**) or all statistics maintained by the control-plane policer (**drop**).
debug platform policer cpu uni-eni	Enable debugging of the control-plane policer. This command displays information messages when any changes are made to CPU protection.

Configuration Example

See example in the "Configuring Control-Plane Security" section on page 8-6.

Related Documents

- *Cisco Connected Grid Switches Layer 2 Switching Software Configuration Guide*
- *Cisco Connected Grid Switches System Management Software Configuration Guide*

You can find all of the related documents at www.cisco.com/go/ie2000u-docs.