CHAPTER 1

Configuring IEEE 802.1x Port-Based Authentication

IEEE 802.1x port-based authentication prevents unauthorized devices (clients) from gaining access to the network. Unless otherwise noted, the term *switch* refers to a standalone switch and to a switch stack.

The Catalyst 3750 switch command reference and the "RADIUS Commands" section in the Cisco IOS Security Command Reference, Release 12.4, have command syntax and usage information.

The switch also supports Cisco TrustSec Security Group Tag (SCT) Exchange Protocol (SXP). This feature supports security group access control lists (SGACLs), which define ACL policies for a group of devices instead of an IP address. The SXP control protocol allows tagging packets with SCTs without a hardware upgrade, and runs between access layer devices at the Cisco TrustSec domain edge and distribution layer devices within the Cisco TrustSec domain. The Catalyst 3750switch operates as access layer switches in the Cisco TrustSec network.

For more information about Cisco TrustSec, see the *Cisco TrustSec Switch Configuration Guide*: http://www.cisco.com/en/US/docs/switches/lan/trustsec/configuration/guide/trustsec.html

The sections on SXP define the capabilities supported on the Catalyst 3750 switch.

This chapter includes these sections:

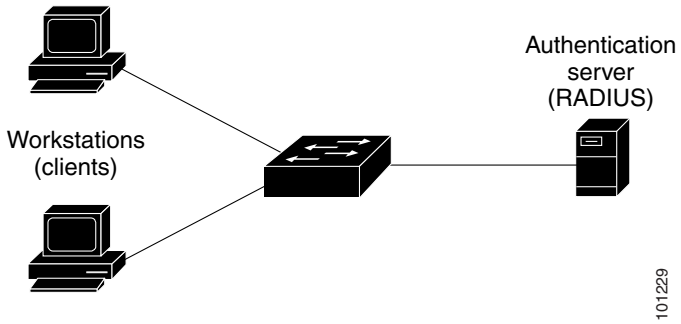
Understanding IEEE 802.1x Port-Based Authentication

The standard defines a client-server-based access control and authentication protocol to prevent unauthorized clients from connecting to a LAN through publicly accessible ports.The authentication server authenticates each client connected to a switch port before making available any switch or LAN services.

Until the client is authenticated, IEEE 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, normal traffic passes through the port.

Device Roles

Figure 1-1 802.1x 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 standard.)

  Note To resolve Windows XP network connectivity and 802.1x authentication issues, read the Microsoft Knowledge Base article: http://support.microsoft.com/support/kb/articles/Q303/5/97.ASP

- *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. (The switch is the *authenticator* in the 802.1x standard.)

  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.

  The devices that can act as intermediaries include the Catalyst 3750-E, Catalyst 3560-E, Catalyst 3750, Catalyst 3560, Catalyst 3550, Catalyst 2975, Catalyst 2970, Catalyst 2960, Catalyst 2955, Catalyst 2950, Catalyst 2940 switches, or a wireless access point. These devices must be running software that supports the RADIUS client and 802.1x authentication.

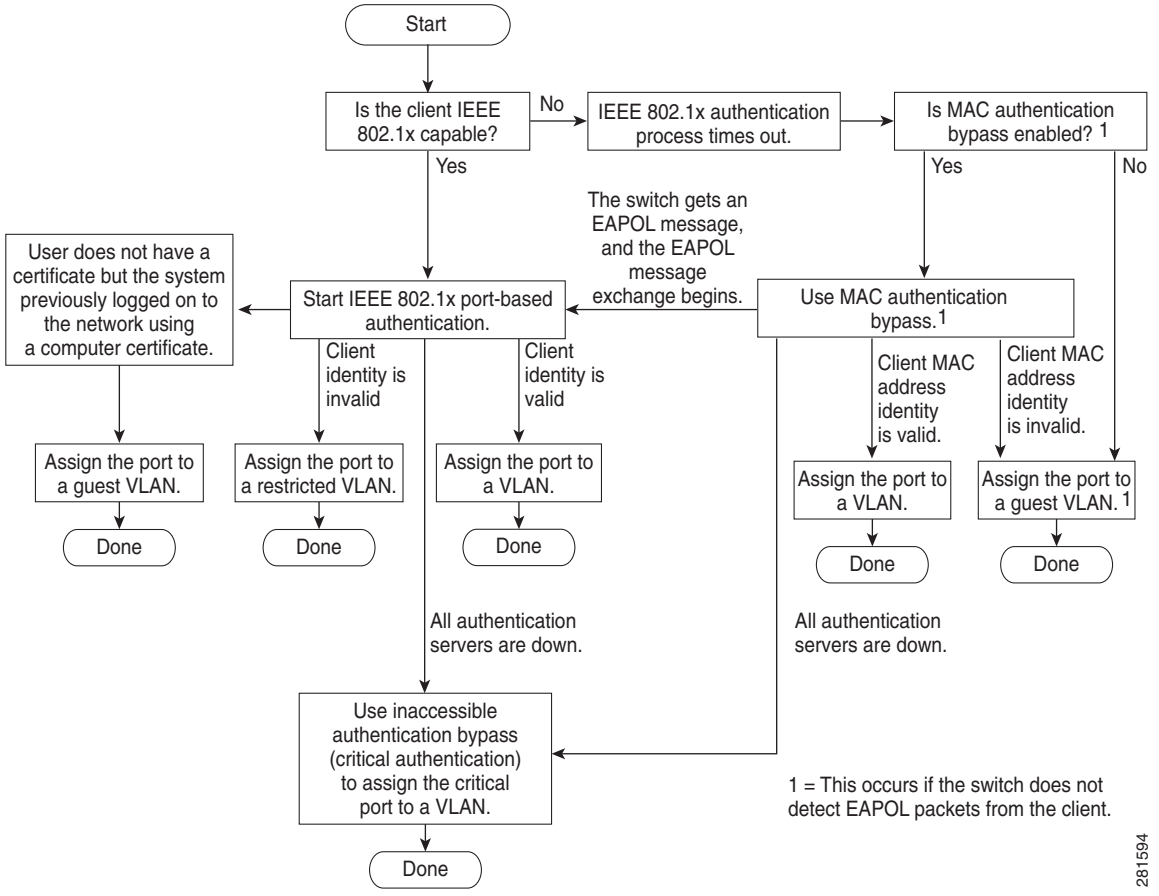
Authentication Process

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.

Figure 1-2 Authentication Flowchart



The switch reauthenticates a client when one of these situations occurs:

- Periodic reauthentication is enabled, and the reauthentication timer expires.

  You can configure the reauthentication 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 reauthentication occurs.

  The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during reauthentication. 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 reauthentication. When the *ReAuthenticate* action is set (the attribute value is *RADIUS-Request*), the session is not affected during reauthentication.

  We recommend that you specify the attribute value as *RADIUS-Request*.

- You manually reauthenticate the client by entering the **dot1x reauthenticate interface** *interface-id* privileged EXEC command.

If Multidomain authentication (MDA) is enabled on a port, this flow can be used with some exceptions that are applicable to voice authorization. For more information on MDA, see the "Multidomain Authentication" section on page 1-13.

Authentication Initiation and Message Exchange

During 802.1x authentication, 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.

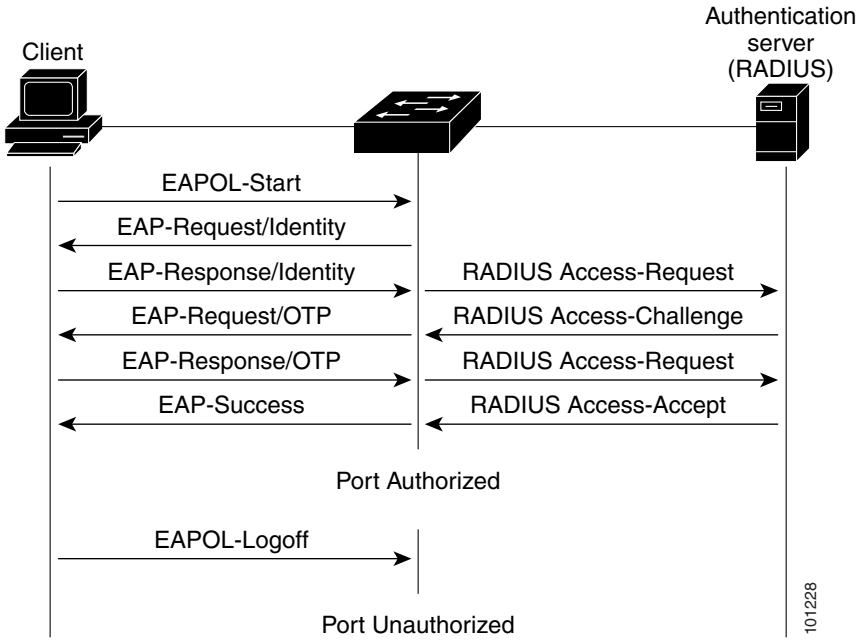
However, if during boot up, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.

Note If 802.1x authentication 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 1-10.

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. If the authentication fails, authentication can be retried, the port might be assigned to a VLAN that provides limited services, or network access is not granted. For more information, see the "Ports in Authorized and Unauthorized States" section on page 1-10.

The specific exchange of EAP frames depends on the authentication method being used. Figure 1-3 shows a message exchange initiated by the client when the client uses the One-Time-Password (OTP) authentication method with a RADIUS server.

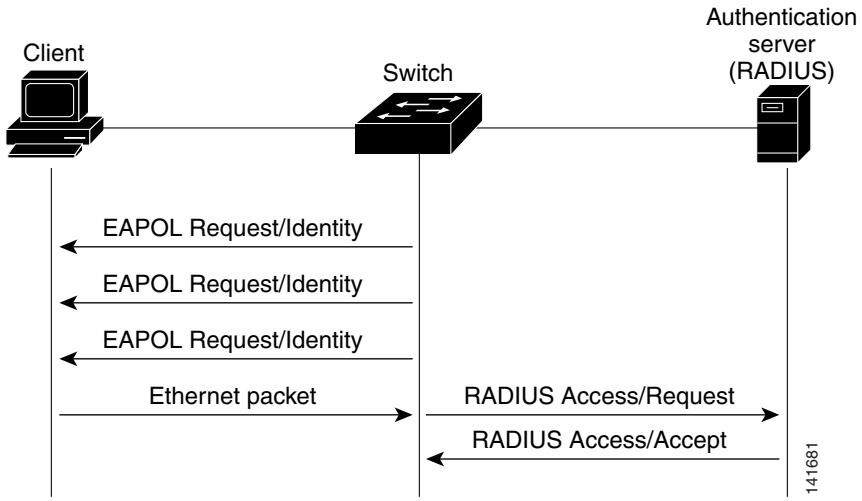
Figure 1-3 Message Exchange



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 1-4 shows the message exchange during MAC authentication bypass.

Figure 1-4 Message Exchange During MAC Authentication Bypass



Authentication Manager

In Cisco IOS Release 12.2(46)SE and earlier, you could not use the same authorization methods, including CLI commands and messages, on this switch and also on other network devices, such as a Catalyst 6000. You had to use separate authentication configurations. Cisco IOS Release 12.2(50)SE and later supports the same authorization methods on all Catalyst switches in a network.

Cisco IOS Release 12.2(55)SE supports filtering verbose system messages from the authentication manager. For details, see the "Authentication Manager CLI Commands" section on page 1-9.

Port-Based Authentication Methods

Table 1-1 lists the authentication methods supported in these host modes:

- Single host–Only one data or voice host (client) can be authenticated on a port.
- Multiple host–Multiple data hosts can be authenticated on the same port. (If a port becomes unauthorized in multiple-host mode, the switch denies network access to all of the attached clients.)
- Multidomain authentication (MDA) –Both a data device and voice device can be authenticated on the same switch port. The port is divided into a data domain and a voice domain.
- Multiple authentication–Multiple hosts can authenticate on the data VLAN. This mode also allows one client on the VLAN if a voice VLAN is configured.

Table 1-1 802.1x Features

	Mode			
Authentication method	**Single Host**	**Multiple Host**	**MDA[1]**	**Multiple Authentication[2]**
802.1x	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL[3] Redirect URL [3]	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL[4] Redirect URL [3]	VLAN assignment Per-user ACL[3] Filter-Id attribute[3] Downloadable ACL[3] Redirect URL[3]	Per-user ACL[3] Filter-Id attribute[3] Downloadable ACL[3] Redirect URL[3]
MAC authentication bypass	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL[3] Redirect URL[3]	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL[3] Redirect URL[3]	VLAN assignment Per-user ACL[3] Filter-Id attribute[3] Downloadable ACL[3] Redirect URL[3]	Per-user ACL[3] Filter-Id attribute[3] Downloadable ACL[3] Redirect URL[3]
Standalone web authentication[4]	Proxy ACL, Filter-Id attribute, downloadable ACL[2]			
NAC Layer 2 IP validation	Filter-Id attribute[3] Downloadable ACL Redirect URL	Filter-Id attribute[3] Downloadable ACL Redirect URL	Filter-Id attribute[3] Downloadable ACL Redirect URL	Filter-Id attribute[3] Downloadable ACL[3] Redirect URL[3]
Web authentication as fallback method[5]	Proxy ACL Filter-Id attribute[3] Downloadable ACL[3]	Proxy ACL Filter-Id attribute[3] Downloadable ACL[3]	Proxy ACL Filter-Id attribute[3] Downloadable ACL[3]	Proxy ACL[3] Filter-Id attribute[3] Downloadable ACL[3]

1. MDA = Multidomain authentication.
2. Also referred to as *multiauth*.
3. Supported in Cisco IOS Release 12.2(50)SE and later.
4. Supported in Cisco IOS Release 12.2(50)SE and later.
5. For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-Ids

In releases earlier than Cisco IOS Release 12.2(50)SE, per-user ACLs and filter Ids were only supported in single-host mode. In Cisco IOS Release 12.2(50), support was added for MDA- and multiauth-enabled ports. In 12.2(52)SE and later, support was added for ports in multihost mode.

In releases earlier than Cisco IOS Release 12.2(50)SE, an ACL configured on the switch is not compatible with an ACL configured on another device running Cisco IOS software, such as a Catalyst 6000 switch.

In Cisco IOS Release 12.2(50)SE or later, the ACLs configured on the switch are compatible with other devices running the Cisco IOS release.

Note You can only set **any** as the source in the ACL.

Note 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**.)

You must specify *any* in the source ports of any defined ACL. Otherwise, the ACL cannot be applied and authorization fails. Single host is the only exception to support backward compatibility.

More than one host can be authenticated on MDA-enabled and multiauth ports. The ACL policy applied for one host does not effect the traffic of another host.

If only one host is authenticated on a multi-host port, and the other hosts gain network access without authentication, the ACL policy for the first host can be applied to the other connected hosts by specifying *any* in the source address.

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 begin with the **dot1x** or **authentication** keyword. For example, the **authentication port-control auto** interface configuration command enables authentication on an interface. However, the **dot1x system-auth-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.

Table 1-2 Authentication Manager Commands and Earlier 802.1x Commands

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication control-direction {both | in}	**dot1x control-direction {both | in}**	Enable authentication with the wake-on-LAN (WoL) feature, and configure the port control as unidirectional or bidirectional.
authentication event	**dot1x auth-fail vlan**	Enable the restricted VLAN on a port.
	dot1x critical (interface configuration)	Enable the inaccessible-authentication-bypass feature.
	dot1x guest-vlan6	Specify an active VLAN as an guest VLAN.

Table 1-2 Authentication Manager Commands and Earlier 802.1x Commands (continued)

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication fallback *fallback-profile*	**dot1x fallback** *fallback-profile*	Configure a port to use web authentication as a fallback method for clients that do not support authentication.
authentication host-mode [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]	**dot1x host-mode** {**single-host** | **multi-host** | **multi-domain**}	Allow a single host (client) or multiple hosts on an authorized port.
authentication order	**dot1x mac-auth-bypass**	Provides the flexibility to define the order of authentication methods to be used.
authentication periodic	**dot1x reauthentication**	Enable periodic reauthentication of the client.
authentication port-control {**auto** | **force-authorized** | **force-un authorized**}	**dot1x port-control {auto** | **force-authorized** | **force-unauthorized}**	Enable manual control of the authorization state of the port.
authentication timer	**dot1x timeout**	Set the timers.
authentication violation {**protect** | **restrict** | **shutdown**}	**dot1x violation-mode** {**shutdown** | **restrict** | **protect**}	Configure the violation modes that occur when a new device connects to a port or when a new device connects to a port after the maximum number of devices are connected to that port.

Beginning with Cisco IOS Release 12.2(55)SE, you can filter out verbose system messages generated by the authentication manager. The filtered content typically relates to authentication success. You can also filter verbose messages for 802.1x authentication and MAB authentication. There is a separate command for each authentication method:

- The **no authentication logging verbose** global configuration command filters verbose messages from the authentication manager.
- The **no dot1x logging verbose** global configuration command filters 802.1x authentication verbose messages.
- The **no mab logging verbose** global configuration command filters MAC authentication bypass (MAB) verbose messages

For more information, see the command reference for this release.

Ports in Authorized and Unauthorized States

During 802.1x authentication, 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 that is not configured as a voice VLAN port disallows all ingress and egress traffic except for 802.1x authentication, 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 the port is configured as a voice VLAN port, the port allows VoIP traffic and 802.1x protocol packets before the client is successfully authenticated.

If a client that does not support 802.1x authentication 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 Authentication and Switch Stacks

If a switch is added to or removed from a switch stack, 802.1x authentication is not affected as long as the IP connectivity between the RADIUS server and the stack remains intact. This statement also applies if the stack master is removed from the switch stack. Note that if the stack master fails, a stack member becomes the new stack master by using the election process described in Chapter 1, “Managing Switch Stacks,” and the 802.1x authentication process continues as usual.

If IP connectivity to the RADIUS server is interrupted because the switch that was connected to the server is removed or fails, these events occur:

- Ports that are already authenticated and that do not have periodic reauthentication enabled remain in the authenticated state. Communication with the RADIUS server is not required.
- Ports that are already authenticated and that have periodic reauthentication enabled (with the **dot1x reauthentication** global configuration command) fail the authentication process when the reauthentication occurs. Ports return to the unauthenticated state during the reauthentication process. Communication with the RADIUS server is required.

  For an ongoing authentication, the authentication fails immediately because there is no server connectivity.

If the switch that failed comes up and rejoins the switch stack, the authentications might or might not fail depending on the boot-up time and whether the connectivity to the RADIUS server is re-established by the time the authentication is attempted.

To avoid loss of connectivity to the RADIUS server, you should ensure that there is a redundant connection to it. For example, you can have a redundant connection to the stack master and another to a stack member, and if the stack master fails, the switch stack still has connectivity to the RADIUS server.

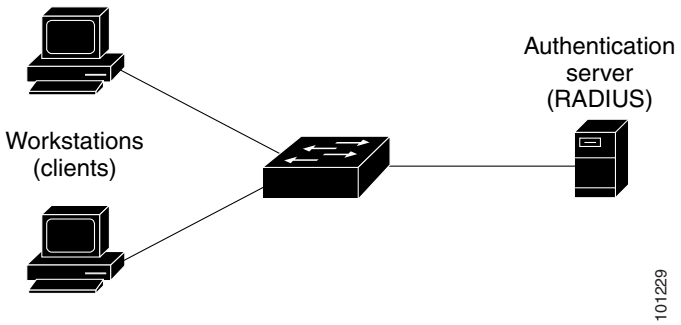
Note Standalone switches with 64MB of DRAM and stackable switches with 128MB of DRAM may encounter system memory exhaustion when 802.1x authentication is enabled concurrently with other features. Hence, switches stacks should be limited to around 5 switches. If more then 5 switches are required in a stack it advisable to keep the number of vlans present and features enable as low as possible.

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 1-1 on page 1-3), 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 1-5 on page 1-12 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 (reauthentication 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.

Figure 1-5 Multiple Host Mode Example



Note For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

The switch supports multidomain authentication (MDA), which allows both a data device and a voice device, such as an IP Phone (Cisco or non-Cisco), to connect to the same switch port. For more information, see the "Multidomain Authentication" section on page 1-13.

Multidomain Authentication

The switch supports multidomain authentication (MDA), which allows both a data device and voice device, such as an IP phone (Cisco or non-Cisco), to authenticate on the same switch port. The port is divided into a data domain and a voice domain.

MDA does not enforce the order of device authentication. However, for best results, we recommend that a voice device is authenticated before a data device on an MDA-enabled port.

The LED of a switch port blinks amber if MDA is configured, there is no phone behind the IP phone, and the default **authentication control-direction both** command is used. This is because the data VLAN spanning-tree is not in the forwarding state, which causes the LED to blink in amber.

Follow these guidelines for configuring MDA:

- To configure a switch port for MDA, see the "Configuring the Host Mode" section on page 1-47.
- You must configure the voice VLAN for the IP phone when the host mode is set to multidomain. For more information, see Chapter 14, "Configuring VLANs."
- Voice VLAN assignment on an MDA-enabled port is supported in Cisco IOS Release 12.2(40)SE and later.

Note If you use a dynamic VLAN to assign a voice VLAN on an MDA-enabled switch port on a switch running Cisco IOS Release 12.2(37)SE, the voice device fails authorization.

- To authorize a voice device, the AAA server must be configured to send a Cisco Attribute-Value (AV) pair attribute with a value of `device-traffic-class=voice`. Without this value, the switch treats the voice device as a data device.
- The guest VLAN and restricted VLAN features only apply to the data devices on an MDA-enabled port. The switch treats a voice device that fails authorization as a data device.
- If more than one device attempts authorization on either the voice or the data domain of a port, it is error disabled.
- Until a device is authorized, the port drops its traffic. Non-Cisco IP phones or voice devices are allowed into both the data and voice VLANs. The data VLAN allows the voice device to contact a DHCP server to obtain an IP address and acquire the voice VLAN information. After the voice device starts sending on the voice VLAN, its access to the data VLAN is blocked.
- A voice device MAC address that is binding on the data VLAN is not counted towards the port security MAC address limit.
- MDA can use MAC authentication bypass as a fallback mechanism to allow the switch port to connect to devices that do not support 802.1x authentication. For more information, see the "MAC Authentication Bypass" section on page 1-40.
- When a *data* or a *voice* device is detected on a port, its MAC address is blocked until authorization succeeds. If the authorization fails, the MAC address remains blocked for 5 minutes.
- If more than five devices are detected on the *data* VLAN or more than one voice device is detected on the *voice* VLAN while a port is unauthorized, the port is error disabled.
- When a port host mode changes from single- or multihost to multidomain mode, an authorized data device remains authorized on the port. However, a Cisco IP phone on the port voice VLAN is automatically removed and must be reauthenticated on that port.
- Active fallback mechanisms such as guest VLAN and restricted VLAN remain configured after a port changes from single-host or multiple-host mode to multidomain mode.

- Switching a port host mode from multidomain to single-host or multiple-hosts mode removes all authorized devices from the port.
- If a data domain is authorized first and placed in the guest VLAN, non-802.1x-capable voice devices need their packets tagged on the voice VLAN to trigger authentication. The phone need not need to send tagged traffic. (The same is true for an 802.1x-capable phone.)
- We do not recommend per-user ACLs with an MDA-enabled port. An authorized device with a per-user ACL policy might impact traffic on both the port voice and data VLANs. You can use only one device on the port to enforce per-user ACLs.

For more information, see the "Configuring the Host Mode" section on page 1-47.

802.1x Multiple Authentication Mode

Multiple-authentication (multiauth) mode allows multiple authenticated clients on the data VLAN. Each host is individually authenticated. If a voice VLAN is configured, this mode also allows one client on the VLAN. (If the port detects any additional voice clients, they are discarded from the port, but no violation errors occur.)

If a hub or access point is connected to an 802.1x-enabled port, each connected client must be authenticated.

For non-802.1x devices, you can use MAC authentication bypass or web authentication as the per-host authentication fallback method to authenticate different hosts with different methods on a single port.

There is no limit to the number of data hosts can authenticate on a multiauthport. However, only one voice device is allowed if the voice VLAN is configured. Since there is no host limit defined violation will not be trigger, if a second voice is seen we silently discard it but do not trigger violation.

For MDA functionality on the voice VLAN, multiple-authentication mode assigns authenticated devices to either a data or a voice VLAN, depending on the VSAs received from the authentication server.

Note When a port is in multiple-authentication mode, the 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 1-24.

For more information about configuring multiauth mode on a port, see the "Configuring the Host Mode" section on page 1-47.

Beginning with Cisco IOS Release 12.2(55)SE, you can assign a RADIUS-server-supplied VLAN in multi-auth mode, under these conditions:

- The host is the first host authorized on the port, and the RADIUS server supplies VLAN information.
- Subsequent hosts are authorized with a VLAN that matches the operational VLAN.
- A host is authorized on the port with no VLAN assignment, and subsequent hosts either have no VLAN assignment, or their VLAN information matches the operational VLAN.
- The first host authorized on the port has a group VLAN assignment, and subsequent hosts either have no VLAN assignment, or their group VLAN matches the group VLAN on the port. Subsequent hosts must use the same VLAN from the VLAN group as the first host. If a VLAN list is used, all hosts are subject to the conditions specified in the VLAN list.
- Only one voice VLAN assignment is supported on a multi-auth port.

- After a VLAN is assigned to a host on the port, subsequent hosts must have matching VLAN information or be denied access to the port.
- You cannot configure a guest VLAN or an auth-fail VLAN in multi-auth mode.
- The behavior of the critical-auth VLAN is not changed for multi-auth mode. When a host tries to authenticate and the server is not reachable, all authorized hosts are reinitialized in the configured VLAN.

MAC Move

When a MAC address is authenticated on one switch port, that address is not allowed on another authentication manager-enabled port of the switch. If the switch detects that same MAC address on another authentication manager-enabled 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 reauthenticated on the new port. When a host moves to a second port, the session on the first port is deleted, and the host is reauthenticated on the new port.

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

When a MAC address moves from one port to another, the switch terminates the authenticated session on the original port and initiates a new authentication sequence on the new port.

The MAC move feature applies to both voice and data hosts.

Note In open authentication mode, a MAC address is immediately moved from the original port to the new port, with no requirement for authorization on the new port.

For more information see the "Enabling MAC Move" section on page 1-52.

MAC Replace

Beginning with Cisco IOS Release 12.2(55)SE, the MAC replace feature can be configured to address the violation that occurs when a host attempts to connect to a port where another host was previously authenticated.

Note This feature does not apply to ports in multi-auth mode, because violations are not triggered in that mode. It does not apply to ports in multiple host mode, because in that mode, only the first host requires authentication.

If you configure the **authentication violation** interface configuration command with the **replace** keyword, the authentication process on a port in multi-domain mode is:

- A new MAC address is received on a port with an existing authenticated MAC address.
- The authentication manager replaces the MAC address of the current data host on the port with the new MAC address.

- The authentication manager initiates the authentication process for the new MAC address.
- If the authentication manager determines that the new host is a voice host, the original voice host is removed.

If a port is in open authentication mode, any new MAC address is immediately added to the MAC address table.

For more information see the "Enabling MAC Replace" section on page 1-53.

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.
- Reauthentication successfully occurs.
- Reauthentication 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 in the Acct-Input-Octets or the Acct-Output-Octets attributes of a RADIUS packet.)

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

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

Table 1-3 lists the AV pairs and when they are sent are sent by the switch:

Table 1-3 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

Table 1-3 Accounting AV Pairs (continued)

Attribute Number	AV Pair Name	START	INTERIM	STOP
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 view the AV pairs that are being sent by the switch by entering the **debug radius accounting** privileged EXEC command. For more information about this command, see the *Cisco IOS Debug Command Reference:*
http://www.cisco.com/en/US/docs/ios/12_2/debug/command/reference/122debug.html

For more information about AV pairs, see RFC 3580, "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 the devices that do not support 802.1x functionality.

This feature only works 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 1-40.

802.1x Authentication with VLAN Assignment

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.

Voice device authentication is supported with multidomain host mode in Cisco IOS Release 12.2(37)SE. In Cisco IOS Release 12.2(40)SE and later, when a voice device is authorized and the RADIUS server returned an authorized VLAN, the voice VLAN on the port is configured to send and receive packets on the assigned voice VLAN. Voice VLAN assignment behaves the same as data VLAN assignment on multidomain authentication (MDA)-enabled ports. For more information, see the "Multidomain Authentication" section on page 1-13.

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

- If no VLAN is supplied by the RADIUS server or if 802.1x authentication 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 authentication is enabled but the VLAN information from the RADIUS server is not valid, authorization fails and configured VLAN remains in use. 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, a shut down or suspended VLAN. In the case of a mutlidomain 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).
- If 802.1x authentication is enabled and all information from the RADIUS server is valid, the authorized device is placed in the specified VLAN after authentication.
- If the 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 and configured voice 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. In the case of a multidomain host, the same applies to voice devices when the port is fully authorized with these exceptions:
  - If the VLAN configuration change of one device results in matching the other device configured or assigned VLAN, then authorization of all devices on the port is terminated and multidomain host mode is disabled until a valid configuration is restored where data and voice device configured VLANs no longer match.
  - If a voice device is authorized and is using a downloaded voice VLAN, the removal of the voice VLAN configuration, or modifying the configuration value to *dot1p* or *untagged* results in voice device un-authorization and the disablement of multi-domain host mode.

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

The 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic 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 in 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, VLAN ID, or VLAN-Group
- [83] Tunnel-Preference

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 1-36.

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 is over, 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 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. Incoming packets received on the port to which a port ACL is applied are filtered by the port ACL. Incoming routed packets received on other ports are filtered by the router ACL. Outgoing routed packets are filtered by the router ACL. 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. For more information, see Chapter 1, "Configuring Network Security with ACLs."

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).

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 1-36. For more information about configuring ACLs, see Chapter 1, "Configuring Network Security with ACLs."

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

To configure per-user ACLs, you need to 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.

For more configuration information, see the "Authentication Manager" section on page 1-7.

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 more than one host is authenticated and the host is in single-host, MDA, or multiple-authentication mode, the switch changes the source address of the ACL to 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. On a voice VLAN port configured in multi-auth or MDA mode, the switch applies the ACL only to the phone as part of the authorization policies.

Beginning with Cisco IOS Release 12.2(55)SE, if there is no static ACL on a port, a dynamic auth-default ACL is created, and policies are enforced before dACLs are downloaded and applied.

Note The auth-default-ACL does not appear in the running configuration.

The auth-default ACL is created when at least one host with an authorization policy is detected on the port. The auth-default ACL is removed from the port when the last authenticated session ends. You can configure the auth-default ACL by using the **ip access-list extended auth-default-acl** global configuration command.

Note The auth-default-ACL does not support Cisco Discovery Protocol (CDP) bypass in the single host mode. You must configure a static ACL on the interface to support CDP bypass.

The 802.1x and MAB authentication methods support two authentication modes, *open* and *closed*. If there is no static ACL on a port in *closed* authentication mode:

- An auth-default-ACL is created.
- The auth-default-ACL allows only DHCP traffic until policies are enforced.
- When the first host authenticates, the authorization policy is applied without IP address insertion.
- When a second host is detected, the policies for the first host are refreshed, and policies for the first and subsequent sessions are enforced with IP address insertion.

If there is no static ACL on a port in *open* authentication mode:

- An auth-default-ACL-OPEN is created and allows all traffic.
- Policies are enforced with IP address insertion to prevent security breaches.
- Web authentication is subject to the auth-default-ACL-OPEN.

To control access for hosts with no authorization policy, you can configure a directive. The supported values for the directive are *open* and *default*. When you configure the *open* directive, all traffic is allowed. The *default* directive subjects traffic to the access provided by the port. You can configure the directive either in the user profile on the AAA server or on the switch. To configure the directive on the AAA server, use the **authz-directive =<open/default>** global command. To configure the directive on the switch, use the **epm access-control open** global configuration command.

Note The default value of the directive is *default*.

If a host falls back to web authentication on a port without a configured ACL:

- If the port is in open authentication mode, the auth-default-ACL-OPEN is created.
- If the port is in closed authentication mode, the auth-default-ACL is created.

The access control entries (ACEs) in the fallback ACL are converted to per-user entries. If the configured fallback profile does not include a fallback ACL, the host is subject to the auth-default-ACL associated with the port.

Note If you use a custom logo with web authentication and it is stored on an external server, the port ACL must allow access to the external server before authentication. You must either configure a static port ACL or change the auth-default-ACL to provide appropriate access to the external server.

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 attribute value pair to intercept an HTTP or HTTPS request from the end point device. The switch then forwards the client web browser to the specified redirect address. The url-redirect attribute value pair on the Cisco Secure ACS contains the URL to which the web browser is redirected. The url-redirect-acl attribute value 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.

When security ACL/dACL and punt/redirect ACLs are applied together to the session, security ACL/dACL has higher priority.

For more information about using redirect ACLs, refer the document here.

Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs

You can set the CiscoSecure-Defined-ACL Attribute-Value 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.

- 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 an 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 "Authentication Manager" section on page 1-7 and the "Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs" section on page 1-72.

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 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 1-74. Additional configuration is similar MAC authentication bypass, as described in the "Configuring MAC Authentication Bypass" section on page 1-67.

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 Windows 98 systems, 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 no longer allows 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 no longer allows other devices access to the guest VLAN. To prevent this situation, use one of these command sequences:

- Enter the **authentication event no-response action authorize vlan** *vlan-id* interface 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, multiple host, or multi-domain modes.

You can configure any active VLAN except an RSPAN VLAN, a private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or 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 1-28.

For more information, see the "Configuring a Guest VLAN" section on page 1-61.

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 stack or 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 reauthentication attempt. A port in the restricted VLAN tries to reauthenticate at configured intervals (the default is 60 seconds). If reauthentication fails, the port remains in the restricted VLAN. If reauthentication is successful, the port moves either to the configured VLAN or to a VLAN sent by the RADIUS server. You can disable reauthentication. 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 reauthentication 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, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

Other 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 1-62.

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

When a port is configured on any host mode and the AAA server is unavailable, the port is then configured to multi-host mode and moved to the critical VLAN. To support this inaccessible bypass on multiple-authentication (multiauth) ports, use the **authentication event server dead action reinitialize vlan** *vlan-id* command. 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.

This command is supported on 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 reauthentication 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 reauthenticated. For more information, see the command reference for this release and the "Configuring Inaccessible Authentication Bypass and Critical Voice VLAN" section on page 1-64.

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 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.
- Voice VLAN—Inaccessible authentication bypass is compatible with voice VLAN, but the RADIUS-configured or user-specified access VLAN and the voice VLAN must be different.
- Remote Switched Port Analyzer (RSPAN)—Do not configure an RSPAN VLAN as the RADIUS-configured or user-specified access VLAN for inaccessible authentication bypass.

In a switch stack:

- The stack master checks the status of the RADIUS servers by sending keepalive packets.

  When the status of a RADIUS server changes, the stack master sends the information to the stack members. The stack members can then check the status of RADIUS servers when reauthenticating critical ports.
- If the new stack master is elected, the link between the switch stack and RADIUS server might change, and the new stack immediately sends keepalive packets to update the status of the RADIUS servers.

  If the server status changes from *dead* to *alive*, the switch reauthenticates all switch ports in the critical-authentication state.
- When a member is added to the stack, the stack master sends the member the server status.

802.1x Critical Voice VLAN

When an IP phone connected to a port is authenticated by the access control server (ACS), the phone is put into the voice domain. If the ACS is not reachable, the switch cannot determine if the device is a voice device. If the server is unavailable, the phone cannot access the voice network and therefore cannot operate.

For data traffic, you can configure inaccessible authentication bypass, or critical authentication, to allow traffic to pass through on the native VLAN when the server is not available. If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network and puts the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN. When the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated, the switch connects those hosts to critical ports. A new host trying to connect to the critical port is moved to a user-specified access VLAN, the critical VLAN, and granted limited authentication.

With this release, you can enter the **authentication event server dead action authorize voice** interface configuration command to configure the critical voice VLAN feature. When the ACS does not respond, the port goes into critical authentication mode. When traffic coming from the host is tagged with the voice VLAN, the connected device (the phone) is put in the configured voice VLAN for the port. The IP phones learn the voice VLAN identification through CDP (Cisco devices) or through LLDP or DHCP.

You can configure the voice VLAN for a port by entering the **switchport voice vlan** *vlan-id* interface configuration command.

This feature is supported in multidomain and multi-auth host modes. Although you can enter the command when the switch in single-host or multi-host mode, the command has no effect unless the device changes to multidomain or multi-auth host mode.

802.1x Authentication with Voice VLAN Ports

A voice VLAN port is a special access port associated with two VLAN identifiers:

- VVID to carry voice traffic to and from the IP phone. The VVID is used to configure the IP phone connected to the port.
- PVID to carry the data traffic to and from the workstation connected to the switch through the IP phone. The PVID is the native VLAN of the port.

The IP phone uses the VVID for its voice traffic, regardless of the authorization state of the port. This allows the phone to work independently of 802.1x authentication.

In single-host mode, only the IP phone is allowed on the voice VLAN. In multiple-hosts mode, additional clients can send traffic on the voice VLAN after a supplicant is authenticated on the PVID. When multiple-hosts mode is enabled, the supplicant authentication affects both the PVID and the VVID.

A voice VLAN port becomes active when there is a link, and the device MAC address appears after the first CDP message from the IP phone. Cisco IP phones do not relay CDP messages from other devices. As a result, if several IP phones are connected in series, the switch recognizes only the one directly connected to it. When 802.1x authentication is enabled on a voice VLAN port, the switch drops packets from unrecognized IP phones more than one hop away.

When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.

When IP phones are connected to an 802.1x-enabled switch port that is in single host mode, the switch grants the phones network access without authenticating them. We recommend that you use multidomain authentication (MDA) on the port to authenticate both a data device and a voice device, such as an IP phone.

Note If you enable 802.1x authentication on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the Cisco IP phone loses connectivity to the switch for up to 30 seconds.

For more information about voice VLANs, see Chapter 1, “Configuring Voice VLAN.”

802.1x Authentication with Port Security

In general, Cisco does not recommend enabling port security when IEEE 802.1x is enabled. Since IEEE 802.1x enforces a single MAC address per port (or per VLAN when MDA is configured for IP telephony), port security is redundant and in some cases may interfere with expected IEEE 802.1x operations.

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 (see Figure 1-2 on page 1-4) 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. This process works for most client devices; however, it does not work for clients that use an alternate MAC address format. You can configure how MAB authentication is performed for clients with MAC addresses that deviate from the standard format or where the RADIUS configuration requires the user name and password to differ. See the "Configuring a MAC Authentication Bypass (MAB) Username and Password" section on page 1-67.

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 reauthentication occurs, the switch uses 802.1x authentication as the preferred reauthentication 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 reauthenticated. The reauthentication process is the same as that for clients that were authenticated with 802.1x. During reauthentication, the port remains in the previously assigned VLAN. If reauthentication is successful, the switch keeps the port in the same VLAN. If reauthentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If reauthentication 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 reauthentication. If MAC authentication bypass is enabled and the 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate reauthorization. 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 Authentication with Port Security" section on page 1-27.
- Voice VLAN—See the "802.1x Authentication with Voice VLAN Ports" section on page 1-27.
- VLAN Membership Policy Server (VMPS)—802.1x and VMPS are mutually exclusive.
- Private VLAN—You can assign a client to a private VLAN.
- Network admission control (NAC) Layer 2 IP validation—This feature takes effect after an 802.1x port is authenticated with MAC authentication bypass, including hosts in the exception list.
- Network Edge Access Topology (NEAT)—MAB and NEAT are mutually exclusive. You cannot enable MAB when NEAT is enabled on an interface, and you cannot enable NEAT when MAB is enabled on an interface.

For more configuration information, see the "Authentication Manager" section on page 1-7.

Cisco IOS Release 12.2(55)SE and later supports filtering of verbose MAB system messages. See the "Authentication Manager CLI Commands" section on page 1-9.

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.

802.1x User Distribution Configuration Guidelines

- 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 1-68.

Network Admission Control Layer 2 802.1x 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 reauthentication 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 reauthenticate the client by using the Termination-Action RADIUS attribute (Attribute[29]). If the value is the *DEFAULT* or is not set, the session ends. If the value is RADIUS-Request, the reauthentication process starts.
- Set the list of VLAN number or name or VLAN group name as the value of the Tunnel Group Private ID (Attribute[81]) and the preference for the VLAN number or name or VLAN group name as the value of the Tunnel Preference (Attribute[83]). If you do not configure the Tunnel Preference, the first Tunnel Group Private ID (Attribute[81]) attribute is picked up from the list.
- View the NAC posture token, which shows the posture of the client, by using the **show authentication** or **show dot1x** privileged EXEC command.
- 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 2 802.1x Validation" section on page 1-69 and the "Configuring Periodic Re-Authentication" section on page 1-49.

For more information about NAC, see the *Network Admission Control Software Configuration Guide*.

For more configuration information, see the "Authentication Manager" section on page 1-7.

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 1-74.

Open1x Authentication

Open1x authentication allows a device to access a port before that device is authenticated. When open authentication is configured, a new host can pass traffic according to the access control list (ACL) defined on the port. 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.
- MDA mode with open authentication–Only one user in the voice domain and one user in the data domain are allowed.
- 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 the Host Mode" section on page 1-47.

Note If open authentication is configured, it takes precedence over other authentication controls. This means that if you use the **authentication open** interface configuration command, the port will grant access to the host irrespective of the **authentication port-control** interface configuration command.

Using Voice Aware 802.1x Security

You use the voice aware 802.1x security feature to configure the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. When an attempt to authenticate the data client caused a security violation in previous releases, the entire port shut down, resulting in a complete loss of connectivity.

You can use this feature where a PC is connected to the IP phone. A security violation found on the data VLAN shuts down only the data VLAN. The traffic on the voice VLAN continues without interruption.

For information on configuring voice aware 802.1x security, see the "Configuring Voice Aware 802.1x Security" section on page 1-41.

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). This allows 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.

In the default state, when you connect a supplicant switch to an authenticator switch that has BPDU guard enabled, the authenticator port could be error-disabled if it receives a Spanning Tree Protocol (STP) bridge protocol data unit (BPDU) packets before the supplicant switch has authenticated. Beginning with Cisco IOS Release 15.0(1)SE, you can control traffic exiting the supplicant port during the authentication period. Entering the **dot1x supplicant controlled transient** global configuration command temporarily blocks the supplicant port during authentication to ensure that the authenticator port does not shut down before authentication completes. If authentication fails, the supplicant port opens. Entering the **no dot1x supplicant controlled transient** global configuration command opens the supplicant port during the authentication period. This is the default behavior.

We strongly recommend using the **dot1x supplicant controlled transient** command on a supplicant switch when BPDU guard is enabled on the authenticator switch port with the **spanning-tree bpduguard enable** interface onfiguration command.

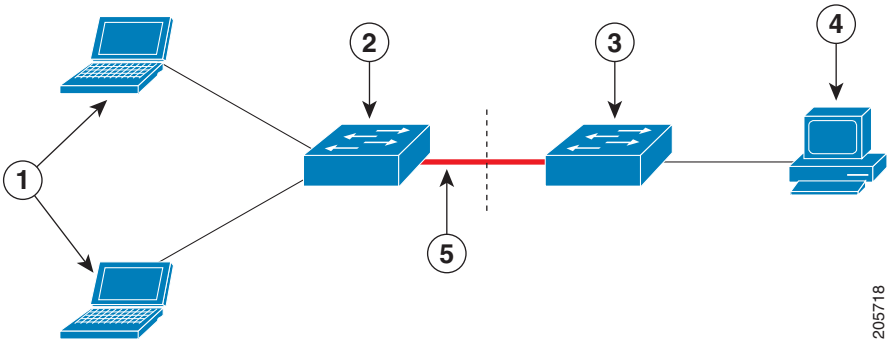
Note If you globally enable BPDU guard on the authenticator switch by using the **spanning-tree portfast bpduguard default** global configuration command, entering the **dot1x supplicant controlled transient** command does not prevent the BPDU violation.

You can enable MDA or multiauth mode on the authenticator switch interface that connects to one more supplicant switches. Multihost 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 1-6.
- 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 1-6 Authenticator and Supplicant Switch using CISP



1	Workstations (clients)	**2**	Supplicant switch (outside wiring closet)
3	Authenticator switch	**4**	Access control server (ACS)
5	Trunk port		

Guidelines

- You can configure NEAT ports with the same configurations as the other authentication ports. When the supplicant switch authenticates, the port mode is changed 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 and the access VLAN if any would be converted to a native trunk VLAN. VSA does not change any of the port configurations on the supplicant
- To change the host mode *and* the apply a standard port configuration on the authenticator switch port, you can also use Auto Smartports user-defined macros, instead of the switch VSA. This allows you to remove unsupported configurations on the authenticator switch port and to change the port mode from *access* to *trunk*. For information, see the *AutoSmartports Configuration Guide*.

For more information, see the "Configuring an Authenticator and a Supplicant Switch with NEAT" section on page 1-70.

Using IEEE 802.1x Authentication with ACLs and the RADIUS Filter-Id Attribute

The switch supports both IP standard and IP extended port access control lists (ACLs) applied to ingress ports.

- ACLs that you configure
- ACLs from the Access Control Server (ACS)

An IEEE 802.1x port in single-host mode uses ACLs from the ACS to provide different levels of service to an IEEE 802.1x-authenticated user. When the RADIUS server authenticates this type of user and port, it sends ACL attributes based on the user identity to the switch. The switch applies the attributes to the port for the duration of the user session. If the session is over, authentication fails, or a link fails, the port becomes unauthorized, and the switch removes the ACL from the port.

Only IP standard and IP extended port ACLs from the ACS support the Filter-Id attribute. It specifies the name or number of an ACL. The Filter-id attribute can also specify the direction (inbound or outbound) and a user or a group to which the user belongs.

- The Filter-Id attribute for the user takes precedence over that for the group.
- If a Filter-Id attribute from the ACS specifies an ACL that is already configure, it takes precedence over a user-configured ACL.
- If the RADIUS server sends more than one Filter-Id attribute, only the last attribute is applied.

If the Filter-Id attribute is not defined on the switch, authentication fails, and the port returns to the unauthorized state.

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.

Device Sensor

Device Sensor uses protocols such as Cisco Discovery Protocol (CDP), Link Layer Discovery Protocol (LLDP), and DHCP to obtain endpoint information from network devices and make this information available to its clients. Device Sensor has internal clients, such as the embedded Device Classifier (local analyzer), Auto Smartports (ASP), MediaNet Service Interface (MSI)-Proxy, and EnergyWise. Device Sensor also has an external client, Identity Services Engine (ISE), which uses RADIUS accounting to receive and analyze endpoint data. When integrated with ISE, Device Sensor provides central policy management and device-profiling capabilities.

Device profiling capability consists of two parts:

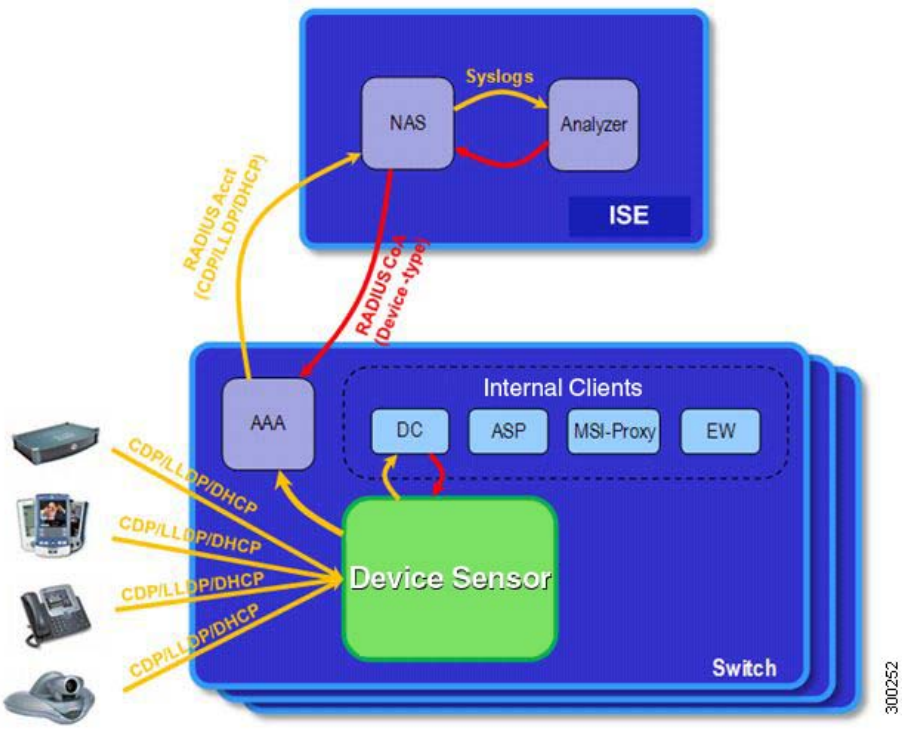
- Collector--Gathers endpoint data from network devices.
- Analyzer--Processes the data and determines the type of device.

For more information about device profiling, see the "Configuring Endpoint Profiling Policies" chapter in the *Cisco Identity Services Engine User Guide* at this URL:

http://www.cisco.com/en/US/docs/security/ise/1.1/user_guide/ise_prof_pol.html

Device Sensor represents the embedded collector functionality. Figure 1-7shows Device Sensor in the context of its internal clients and the ISE.

Figure 1-7 Device Sensor and Clients



Client notifications and accounting messages that contain profiling data and other session-related data are generated and sent to the internal clients and the ISE. By default, client notifications and accounting events are generated only when an incoming packet includes a Type-Length-Value (TLV) that has not previously been received within a given access session. You can enable client notifications and accounting events for TLV changes; that is, when a previously received TLV is received with a different value.

Device Sensor port security protects the switch from consuming memory and failing during deliberate or unintentional denial-of-service (DoS)-type attacks.

Guidelines

- Device Sensor limits the maximum number of device monitoring sessions to 32 per port.
- In the case of lack of activity from hosts, the age session limit is 12 hours.
- The length of one TLV must not be more than 1024 and the total length of TLVs (combined length of TLVs) of all protocols must not be more than 4096.
- Device Sensor profiles devices that are only one hop away.
- Only CDP, LLDP, and DHCP protocols are supported.
- To troubleshoot Device Sensor, use the **debug device-sensor** and the **debug authentication all** privileged EXEC commands.

For more information, see the "Configuring Device Sensor" section on page 1-55.

Configuring 802.1x Authentication

- Configuring a Guest VLAN, page 1-61 (optional)
- Configuring a Restricted VLAN, page 1-62 (optional)
- Configuring Inaccessible Authentication Bypass and Critical Voice VLAN, page 1-64 (optional)
- Configuring 802.1x Authentication with Wake-on-LAN, page 1-66 (optional)
- Configuring MAC Authentication Bypass, page 1-67 (optional)
- Configuring NAC Layer 2 802.1x Validation, page 1-69 (optional)
- Configuring an Authenticator and a Supplicant Switch with NEAT, page 1-70 (optional)
- Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs, page 1-72 (optional)
- Configuring Flexible Authentication Ordering, page 1-74 (optional)
- Disabling 802.1x Authentication on the Port, page 1-76 (optional)
- Resetting the 802.1x Authentication Configuration to the Default Values, page 1-76 (optional)

Default 802.1x Authentication Configuration

Table 1-4 Default 802.1x Authentication Configuration

Feature	Default Setting
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.
AAA	Disabled.
RADIUS server • IP address • UDP authentication port • Key	 • None specified. • 1812. • None specified.
Host mode	Single-host mode.
Control direction	Bidirectional control.
Periodic reauthentication	Disabled.
Number of seconds between reauthentication attempts	3600 seconds.
Reauthentication 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).

Table 1-4 Default 802.1x Authentication Configuration (continued)

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 server** interface configuration command.
Inactivity timeout	Disabled.
Guest VLAN	None specified.
Inaccessible authentication bypass	Disabled.
Restricted VLAN	None specified.
Authenticator (switch) mode	None specified.
MAC authentication bypass	Disabled.
Voice-aware security	Disabled

802.1x Authentication Configuration Guidelines

802.1x Authentication

- When 802.1x authentication is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.
- If the VLAN to which an 802.1x-enabled port is assigned changes, this change is transparent and does not affect the switch. For example, this change occurs if a port is assigned to a RADIUS server-assigned VLAN and is then assigned to a different VLAN after reauthentication.

  If the VLAN to which an 802.1x port is assigned to shut down, disabled, or removed, the port becomes unauthorized. For example, the port is unauthorized after the access VLAN to which a port is assigned shuts down or is removed.
- The 802.1x protocol is supported on Layer 2 static-access ports, voice VLAN ports, and Layer 3 routed ports, but it is not supported on these port types:
  - Trunk port—If you try to enable 802.1x authentication on a trunk port, an error message appears, and 802.1x authentication 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 ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x authentication on a dynamic port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, an error message appears, and the port mode is not changed.
  - Dynamic-access ports—If you try to enable 802.1x authentication on a dynamic-access (VLAN Query Protocol [VQP]) port, an error message appears, and 802.1x authentication 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 authentication on an EtherChannel port, an error message appears, and 802.1x authentication is not enabled.
  - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x authentication on a port that is a SPAN or RSPAN destination port. However, 802.1x authentication is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x authentication on a SPAN or RSPAN source port.

- Before globally enabling 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x authentication and EtherChannel are configured.
- Cisco IOS Release 12.2(55)SE and later supports filtering of system messages related to 802.1x authentication. See the "Authentication Manager CLI Commands" section on page 1-9.

VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass

- When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.
- The 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VMPS.
- You can configure 802.1x authentication on a private-VLAN port, but do not configure 802.1x authentication with port security, a voice VLAN, a guest VLAN, a restricted VLAN, or a per-user ACL on private-VLAN ports.
- You can configure any VLAN except an RSPAN VLAN, private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- After you configure a guest VLAN for an 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can change the settings for restarting the 802.1x authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. Decrease the settings for the 802.1x authentication process (**authentication timer inactivity** and **authentication timer reauthentication** interface configuration commands). The amount to decrease the settings depends on the connected 802.1x client type.
- When configuring the inaccessible authentication bypass feature, follow these guidelines:
  - The feature is supported on 802.1x port in single-host mode and multihosts mode.
  - If the client is running Windows XP and the port to which the client is connected is in the critical-authentication state, Windows XP might report that the interface is not authenticated.
  - If the Windows XP client is configured for DHCP and has an IP address from the DHCP server, receiving an EAP-Success message on a critical port might not re-initiate the DHCP configuration process.

  - You can configure the inaccessible authentication bypass feature and the restricted VLAN on an 802.1x port. If the switch tries to reauthenticate a critical port in a restricted VLAN and all the RADIUS servers are unavailable, switch changes the port state to the critical authentication state and remains in the restricted VLAN.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

MAC Authentication Bypass

- Unless otherwise stated, the MAC authentication bypass guidelines are the same as the 802.1x authentication guidelines. For more information, see the "802.1x Authentication" section on page 1-38.
- If you disable MAC authentication bypass from a port after the port has been authorized with its MAC address, the port state is not affected.
- If the port is in the unauthorized state and the client MAC address is not the authentication-server database, the port remains in the unauthorized state. However, if the client MAC address is added to the database, the switch can use MAC authentication bypass to reauthorize the port.
- If the port is in the authorized state, the port remains in this state until reauthorization occurs.
- You can configure a timeout period for hosts that are connected by MAC authentication bypass but are inactive. The range is 1to 65535 seconds.

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. If the port is also configured with a voice VLAN, an unlimited number of Cisco IP phones can send and receive traffic through the voice VLAN.
- In multidomain authentication (MDA) mode, one device is allowed for the access VLAN, and one IP phone is allowed for the voice VLAN.
- In multiple-host 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. An unlimited number of devices are allowed on the voice VLAN.

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 **dot1x force-unauthorized**.

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.

Beginning in privileged EXEC mode, follow these steps to enable the 802.1x readiness check on the switch:

	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 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 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.

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 Voice Aware 802.1x Security

You use the voice aware 802.1x security feature on the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. 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 results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

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

- You enable voice aware 802.1x security by entering the **errdisable 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 re-enabled. If error-disabled recovery is not configured for the port, you re-enable it by using the **shutdown** and **no-shutdown** interface configuration commands.
- You can re-enable 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.

Beginning in privileged EXEC mode, follow these steps to enable voice aware 802.1x security:

	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 re-enabled. If *vlan-list* is not specified, all VLANs are re-enabled.
Step 5	**shutdown** **no-shutdown**	(Optional) Re-enable an error-disabled VLAN, and clear all error-disable indications.
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.

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 re-enable all VLANs that were error disabled on port Gigabit Ethernet 4/0/2.

```
Switch# clear errdisable interface gigabitethernet4/0/2 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 about devices have been authenticated on the port

Beginning in privileged EXEC mode, follow these steps to configure the security violation actions on the switch:

	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 that is to be used 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	**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 5	**switchport mode access**	Set the port to access mode.
Step 6	**authentication violation** {**shutdown** | **restrict** | **protect** | **replace**}	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**–Removes the current session and authenticates with the new host.
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.

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 per-user ACLs or VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the 802.1x AAA process:

Step 1 A user connects to a port on the switch.

Step 2 Authentication is performed.

Step 3 VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.

Step 4 The switch sends a start message to an accounting server.

Step 5 Reauthentication is performed, as necessary.

Step 6 The switch sends an interim accounting update to the accounting server, that is based on the result of reauthentication.

Step 7 The user disconnects from the port.

Step 8 The switch sends a stop message to the accounting server.

Beginning in privileged EXEC mode, follow these steps to configure 802.1x port-based authentication:

	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 to use user-RADIUS authorization for all network-related service requests, such as per-user ACLs or VLAN assignment. For per-user ACLs, single-host mode must be configured. This setting is the default.
Step 6	**radius-server host** *ip-address*	(Optional) Specify the IP address of the RADIUS server.
Step 7	**radius-server key** *string*	(Optional) Specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.

	Command	Purpose
Step 8	**interface** *interface-id*	Specify the port connected to the client to enable for 802.1x authentication, and enter interface configuration mode.
Step 9	**switchport mode access**	(Optional) Set the port to access mode only if you configured the RADIUS server in Step 6 and Step 7.
Step 10	**authentication port-control auto**	Enable 802.1x authentication on the port. For feature interaction information, see the "802.1x Authentication Configuration Guidelines" section on page 1-38.
Step 11	**dot1x pae authenticator**	Set the interface Port Access Entity to act only as an authenticator and ignore messages meant for a supplicant.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show authentication**	Verify your entries.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

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 the 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 fail-over backup to the first one. The RADIUS host entries are tried in the order in which they were configured.

Beginning in privileged EXEC mode, follow these steps to configure the RADIUS server parameters on the switch. This procedure is required.

	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, re-enter this command. **Note** To configure an IPv4 or IPv6 RADIUS server, use the commands as mentioned below: • If you have configured an IPv4 RADIUS server, you can use either the radius-server host command or the **radius server** *name* command. • If you have configured an IPv6 RADIUS server, use the **radius server** *name* command. For more information about the **radius server** command, see *Cisco IOS Security Command Reference: Commands M to R*.
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 clear the specified RADIUS server, use the **no radius-server host** {*hostname* | *ip-address*} global configuration command.

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 1-36.

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 the Host Mode

Beginning in privileged EXEC mode, follow these steps to allow a single host (client) or multiple hosts on an 802.1x-authorized port. Use the **multi-domain** keyword to configure multidomain authentication (MDA) to enable authentication of both a host and a voice device, such as an IP phone (Cisco or non-Cisco) on the same switch port.

This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server vsa send authentication**	Configure the network access server to recognize and use vendor-specific attributes (VSAs).
Step 3	**interface** *interface-id*	Specify the port to which multiple hosts are indirectly attached, and enter interface configuration mode.
Step 4	**authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]	The keywords have these meanings: • **multi-auth**–Allow one client on the voice VLAN and 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. • **multi-domain**–Allow both a host and a voice device, such as an IP phone (Cisco or non-Cisco), to be authenticated on an 802.1x-authorized port. **Note** You must configure the voice VLAN for the IP phone when the host mode is set to **multi-domain**. For more information, see Chapter 1, "Configuring Voice VLAN." • **single-host**–Allow a single host (client) on an 802.1x-authorized port. Make sure that the **authentication port-control** interface configuration command set is set to **auto** for the specified interface.
Step 5	**switchport voice vlan** *vlan-id*	(Optional) Configure the voice VLAN.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show authentication interface** *interface-id*	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

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

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

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# authentication port-control auto
Switch(config-if)# authentication host-mode multi-host
Switch(config-if)# end
```

This example shows how to enable MDA and to allow both a host and a voice device on the port:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# authentication port-control auto
Switch(config-if)# authentication host-mode multi-domain
Switch(config-if)# switchport voice vlan 101
Switch(config-if)# end
```

Configuring Periodic Re-Authentication

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

Beginning in privileged EXEC mode, follow these steps to enable periodic reauthentication of the client and to configure the number of seconds between reauthentication attempts. This procedure is optional.

	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 reauthentication of the client, which is disabled by default. **Note** The default value is 3600 seconds. To change the value of the reauthentication timer or to have the switch use a RADIUS-provided session timeout, enter the **authentication timer reauthenticate** command.
Step 4	**authentication timer** {{[**inactivity** | **reauthenticate**]} {**restart** *value*}}	Set the number of seconds between reauthentication attempts. The **authentication timer** keywords have these meanings: • **inactivity**—Interval in seconds after which if there is no activity from the client then it is unauthorized • **reauthenticate**—Time in seconds after which an automatic reauthentication attempt is be initiated • **restart** *value*—Interval in seconds after which an attempt is made to authenticate an unauthorized port This command affects the behavior of the switch only if periodic reauthentication is enabled.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show authentication interface** *interface-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

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

This example shows how to enable periodic reauthentication and set the number of seconds between reauthentication 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 reauthenticate the client connected to a specific port at any time by entering the **dot1x reauthenticate interface** *interface-id* privileged EXEC command. This step is optional. If you want to enable or disable periodic reauthentication, see the "Configuring Periodic Re-Authentication" section

on page 1-49.

This example shows how to manually reauthenticate the client connected to a port:

```
Switch# dot1x reauthenticate interface gigabitethernet2/0/1
```

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 **dot1x timeout quiet-period** interface configuration command controls the idle 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 number smaller than the default.

Beginning in privileged EXEC mode, follow these steps to change the quiet period. This procedure is optional.

	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 quiet-period** *seconds*	Set the number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show authentication** *interface-id* or **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 quiet time, use the **no dot1x timeout quiet-period** interface configuration command.

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

```
Switch(config-if)# dot1x timeout quiet-period 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.

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.

Beginning in privileged EXEC mode, follow these steps to change the amount of time that the switch waits for client notification. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to configure, and enter interface configuration mode.
Step 3	**authentication timer reauthenticate** *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 5.
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 retransmission time, use the **no authentication timer reauthenticate** interface configuration command.

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)# authentication timer reauthenticate 60
```

Setting the Switch-to-Client Frame-Retransmission Number

You can change the number of times that the switch sends an EAP-request/identity frame (assuming no response is received) to the client before restarting the authentication process.

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.

Beginning in privileged EXEC mode, follow these steps to set the switch-to-client frame-retransmission number. This procedure is optional.

	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-request/identity 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 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 retransmission number, use the **no dot1x max-req** interface configuration command.

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

```
Switch(config-if)# dot1x max-reauth-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.

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.

Beginning in privileged EXEC mode, follow these steps to set the reauthentication number. This procedure is optional.

	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 restarts the authentication process before the port changes to the unauthorized state. The range is 0 to 10; the default is 2.
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 reauthentication number, use the **no dot1x max-reauth-req** interface configuration command.

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.

Beginning in privileged EXEC mode, follow these steps to globally enable MAC move on the switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**authentication mac-move permit**	Enable MAC move on the switch.
Step 3	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 4	**show running-config**	(Optional) Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

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

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

Enabling MAC Replace

MAC replace allows a host to replace an authenticated host on a port.

Beginning in privileged EXEC mode, follow these steps to enable MAC replace on an interface. This procedure is optional.

	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 violation** {**protect** | **replace** | **restrict** | **shutdown**}	Use the **replace** keyword to enable MAC replace on the interface. The port removes the current session and initiates authentication with the new host. The other keywords have these effects: • **protect**: the port drops packets with unexpected MAC addresses without generating a system message. • **restrict**: violating packets are dropped by the CPU and a system message is generated. • **shutdown**: the port is error disabled when it receives an unexpected MAC address.
Step 4	**end**	Return to privileged EXEc mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

This example shows how to enable MAC replace on an interface:

```
Switch(config)# interface gigabitethernet2/0/2
Switch(config-if)# authentication violation replace
```

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.
```

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

Beginning in privileged EXEC mode, follow these steps to configure 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

	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	**aaa accounting dot1x default start-stop group radius**	Enable 802.1x accounting using the list of all RADIUS servers.
Step 4	**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 5	**end**	Return to privileged EXEc mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Saves 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.

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 Device Sensor

Device Sensor is enabled by default. Complete the following tasks when you want Device Sensor to include or exclude a list of TLVs for a particular protocol. These lists are called filter lists.

Note If you do not perform any Device Sensor configuration tasks, the following TLVs are included by default:

- CDP filter--secondport-status-type and powernet-event-type (types 28 and 29)
- LLDP filter--organizationally-specific (type 127)
- DHCP filter--message-type (type 53)

Enabling Accounting Augmentation

For the Device Sensor protocol data to be added to accounting messages, you must first enable session accounting by using the following standard Authentication, Authorization, and Accounting (AAA) and RADIUS configuration commands:

```
Switch(config)# aaa new-model
Switch(config)# aaa accounting dot1x default start-stop group radius
Switch(config)# radius-server host{hostname|ip-address}[auth-port
port-number][acct-port port-number] [timeout seconds][retransmit retries][key string]
Switch(config)# radius-server vsa send accounting
```

Beginning in privileged EXEC mode, follow these steps to add Device Sensor protocol data to accounting records:

	Command	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command	Purpose
Step 2	**device-sensor accounting** **Example:** `Switch(config)# device-sensor accounting`	Enables the addition of sensor protocol data to accounting records and also enables the generation of additional accounting events when new sensor data is detected.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Creating a Cisco Discovery Protocol Filter

Beginning in privileged EXEC mode, follow these steps to create a CDP filter containing a list of TLVs that can be included or excluded in the Device Sensor output:

	Command	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**device-sensor filter-list cdp list tlv-list-name** **Example:** `Switch(config)# device-sensor filter-list cdp list cdp-list`	Creates a TLV list and enters CDP sensor configuration mode, where you can configure individual TLVs.
Step 3	**tlv** {**name** *tlv-name* | **number** *tlv-number*} **Example:** `Switch(config-sensor-cdplist)# tlv number 10`	Adds individual CDP TLVs to the TLV list. You can delete the TLV list without individually removing TLVs from the list by using the **no device-sensor filter-list cdp list tlv-list-name** command.
Step 4	**end** **Example:** `Switch(config-sensor-cdplist)# end`	Returns to privileged EXEC mode.

Creating an LLDP Filter

Beginning in privileged EXEC mode, follow these steps to create an LLDP filter containing a list of TLVs that can be included or excluded in the Device Sensor output:

	Command	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**device-sensor filter-list lldp list tlv-list-name** **Example:** `Switch(config)# device-sensor filter-list lldp list lldp-list`	Creates a TLV list and enters LLDP sensor configuration mode, where you can configure individual TLVs.
Step 3	**tlv** {**name** *tlv-name* | **number** *tlv-number*} **Example:** `Switch(config-sensor-cdplist)# tlv number 10`	Adds individual LLDP TLVs to the TLV list. You can delete the TLV list without individually removing TLVs from the list by using the **no device-sensor filter-list lldp list tlv-list-name** command.
Step 4	**end** **Example:** `Switch(config-sensor-lldplist)# end`	Returns to privileged EXEC mode.

Creating a DHCP Filter

Beginning in privileged EXEC mode, follow these steps to create a DHCP filter containing a list of DHCP options that can be included or excluded in the Device Sensor output:

	Command	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**device-sensor filter-list dhcp list option-list-name** **Example:** `Switch(config)# device-sensor filter-list dhcp list dhcp-list`	Creates an options list and enters DHCP sensor configuration mode, where you can specify individual DHCP options.

	Command	Purpose
Step 3	**option** {**name** *option-name* | **number** *option-number*} **Example:** `Switch(config-sensor-dhcplist)# option number 50`	Adds individual DHCP options to the option list. You can delete the entire option list without removing options individually from the list by using the **no device-sensor filter-list dhcp list option-list-name** command.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Applying a Protocol Filter to the Device Sensor Output

Beginning in privileged EXEC mode, follow these steps to apply a CDP, LLDP, or DHCP filter to the sensor output. The output is session notifications to internal sensor clients and accounting requests to the RADIUS server.

Note Only one filter list can be included or excluded at a time.

	Command	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**device-sensor filter-spec** {**cdp** | **dhcp** | **lldp**} {**exclude** {**all** | **list** *list-name*} | **include list** *list-name*} **Example:** `Switch(config)# device-sensor filter-spec cdp include list list1`	Applies a specific protocol filter containing a list of protocol TLV fields or DHCP options to the Device Sensor output. • **cdp**--Applies a CDP TLV filter list to the device sensor output. • **lldp**--Applies an LLDP TLV filter list to the device sensor output. • **dhcp**--Applies a DHCP option filter list to the device sensor output. • **exclude**--Specifies the TLVs that must be excluded from the device sensor output. • **include**--Specifies the TLVs that must be included from the device sensor output. • **all**--Disables all notifications for the associated protocol. • **list** *list-name*--Specifies the protocol TLV filter list name.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Tracking TLV Changes

By default, client notifications and accounting events are generated only when an incoming packet includes a TLV that has not previously been received within a given session.

To enable client notifications and accounting events for TLV changes, begin in privileged EXCEC mode and follow these steps.

	Command	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**device-sensor notify all-changes** **Example:** `Switch(config)# device-sensor notify all-changes`	Enables client notifications and accounting events for all TLV changes, that is, where either a new TLV is received or a previously received TLV is received with a new value in the context of a given session. **Note** Use the default device-sensor notify or the device-sensor notify new-tlvs command to return to the default TLV.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Verifying the Device Sensor Configuration

Beginning in privileged EXEC mode, follow these steps to verify the sensor cache entries for all devices.

	Command	Purpose
Step 1	**show device-sensor cache mac** *mac-address*	Displays sensor cache entries (the list of protocol TLVs or options received from a device) for a specific device. • **mac-address** is the MAC address of the endpoint
Step 2	**show device-sensor cache all** **Example:** `Switch(config)# device-sensor notify all-changes`	Displays sensor cache entries for all devices.

This is an example of the **show device-sensor cache mac mac-address** privileged EXEC command output:

```
Switch# show device-sensor cache mac 0024.14dc.df4d

Device: 0024.14dc.df4d on port GigabitEthernet1/0/24
-------------------------------------------------
Proto Type:Name                          Len Value
cdp     26:power-available-type           16 00 1A 00 10 00 00 00 01 00 00 00 00 FF FF FF FF
cdp     22:mgmt-address-type              17 00 16 00 11 00 00 00 01 01 01 CC 00 04 09 1B 65
                                             0E
```

```
cdp     11:duplex-type                  5 00 0B 00 05 01
cdp      9:vtp-mgmt-domain-type         4 00 09 00 04
cdp      4:capabilities-type            8 00 04 00 08 00 00 00 28
cdp      1:device-name                 14 00 01 00 0E 73 75 70 70 6C 69 63 61 6E 74
lldp     0:end-of-lldpdu                2 00 00
lldp     8:management-address          14 10 0C 05 01 09 1B 65 0E 03 00 00 00 01 00
lldp     7:system-capabilities          6 0E 04 00 14 00 04
lldp     4:port-description            23 08 15 47 69 67 61 62 69 74 45 74 68 65 72 6E 65
                                           74 31 2F 30 2F 32 34
lldp     5:system-name                 12 0A 0A 73 75 70 70 6C 69 63 61 6E 74
dhcp    82:relay-agent-info            20 52 12 01 06 00 04 00 18 01 18 02 08 00 06 00 24
                                           14 DC DF 80
dhcp    12:host-name                   12 0C 0A 73 75 70 70 6C 69 63 61 6E 74
dhcp    61:client-identifier           32 3D 1E 00 63 69 73 63 6F 2D 30 30 32 34 2E 31 34
                                           64 63 2E 64 66 34 64 2D 47 69 31 2F 30 2F 32 34
dhcp    57:max-message-size             4 39 02 04 80
```

This is an example of the **show device-sensor cache all** privileged EXEC command output:

```
Switch# show device-sensor cache all

Device: 001c.0f74.8480 on port GigabitEthernet2/1
-------------------------------------------------
Proto Type:Name Len Value
dhcp 52:option-overload 3 34 01 03
dhcp 60:class-identifier 11 3C 09 64 6F 63 73 69 73 31 2E 30
dhcp 55:parameter-request-list 8 37 06 01 42 06 03 43 96
dhcp 61:client-identifier 27 3D 19 00 63 69 73 63 6F 2D 30 30 31 63 2E 30 66
37 34 2E 38 34 38 30 2D 56 6C 31
dhcp 57:max-message-size 4 39 02 04 80
Device: 000f.f7a7.234f on port GigabitEthernet2/1
-------------------------------------------------
Proto Type:Name Len Value
cdp 22:mgmt-address-type 8 00 16 00 08 00 00 00 00
cdp 19:cos-type 5 00 13 00 05 00
cdp 18:trust-type 5 00 12 00 05 00
cdp 11:duplex-type 5 00 0B 00 05 01
cdp 10:native-vlan-type 6 00 0A 00 06 00 01
cdp 9:vtp-mgmt-domain-type 9 00 09 00 09 63 69 73 63 6F
```

Configuration Examples for the Device Sensor Feature

The following example shows how to create a CDP filter containing a list of TLVs:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor filter-list cdp list cdp-list
Switch(config-sensor-cdplist)# tlv name address-type
Switch(config-sensor-cdplist)# tlv name device-name
Switch(config-sensor-cdplist)# tlv number 34
Switch(config-sensor-cdplist)# end
```

The following example shows how to create an LLDP filter containing a list of TLVs:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor filter-list lldp list lldp-list
Switch(config-sensor-lldplist)# tlv name chassis-id
Switch(config-sensor-lldplist)# tlv name management-address
Switch(config-sensor-lldplist)# tlv number 28
Switch(config-sensor-lldplist)# end
```

The following example shows how to create a DHCP filter containing a list of options:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor filter-list dhcp list dhcp-list
Switch(config)# device-sensor filter-list dhcp list dhcp-list
Switch(config-sensor-dhcplist)# option name domain-name
Switch(config-sensor-dhcplist)# option name host-name
Switch(config-sensor-dhcplist)# option number 50
Switch(config-sensor-dhcplist)# end
```

The following example shows how to apply a CDP TLV filter list to the Device Sensor output:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor filter-spec cdp include cdp-list1
```

The following example shows how to enable client notifications and accounting events for all TLV changes:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor notify all-changes
```

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.

Beginning in privileged EXEC mode, follow these steps to configure a guest VLAN. This procedure is optional.

	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 "802.1x Authentication Configuration Guidelines" section on page 1-38.
Step 3	**switchport mode access** or **switchport mode private-vlan host**	Set the port to access mode or Configure the Layer 2 port as a private-VLAN host port.
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 internal VLAN (routed port), an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x guest VLAN.
Step 6	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 7	**show authentication interface** *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.

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

```
Switch(config)# interface gigabitethernet2/0/2
Switch(config-if)# authentication event no-response action authorize vlan 2
```

This example shows how to set 3 as the quiet time 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 re-sending 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 inactivity 3
Switch(config-if)# authentication timer reauthenticate 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 stack or 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.

Beginning in privileged EXEC mode, follow these steps to configure a restricted VLAN. This procedure is optional.

	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 "802.1x Authentication Configuration Guidelines" section on page 1-38.
Step 3	**switchport mode access** or **switchport mode private-vlan host**	Set the port to access mode, or Configure the Layer 2 port as a private-VLAN host port.
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 internal VLAN (routed port), an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN.
Step 6	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 7	**show authentication interface** *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 action authorize** *vlan-id* interface configuration command. The port returns to the unauthorized state.

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

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

You can configure the maximum number of authentication attempts allowed before a user is assigned to the restricted VLAN by using the **authentication event retry** *retry count* interface configuration command. The range of allowable authentication attempts is 1 to 3. The default is 3 attempts.

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of allowed authentication attempts. This procedure is optional.

	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 "802.1x Authentication Configuration Guidelines" section on page 1-38.
Step 3	**switchport mode access** or **switchport mode private-vlan host**	Set the port to access mode, or Configure the Layer 2 port as a private-VLAN host port.
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 internal VLAN (routed port), an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN.
Step 6	**authentication event retry** *retry count*	Specify a number of authentication attempts to allow before a port moves to the restricted VLAN. The range is 1 to 3, and the default is 3.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show authentication interface** *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 retry** interface configuration command.

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

```
Switch(config-if)# authentication event retry 2
```

Configuring Inaccessible Authentication Bypass and Critical Voice VLAN

You can configure the inaccessible bypass feature, also referred to as critical authentication or the AAA fail policy to allow data traffic to pass through on the native VLAN when the server is not available. You can also configure the critical voice VLAN feature so that if the server is not available and traffic from the host is tagged with the voice VLAN, the connected device (the phone) is put in the configured voice VLAN for the port.

Beginning in privileged EXEC mode, follow these steps to configure critical voice VLAN on a port and enable the inaccessible authentication bypass feature.

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**radius-server dead-criteria time** *time* **tries** *tries*	Sets the conditions that are used to decide when a RADIUS server is considered unavailable or down (*dead*). • The range for *time* is from 1 to 120 seconds. The switch dynamically determines a default *seconds* value between 10 and 60 seconds. • The range for *tries* is from 1 to 100. The switch dynamically determines a default *tries* parameter between 10 and 100.
Step 3	**radius-server deadtime** *minutes*	(Optional) Sets the number of minutes during which a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.
Step 4	**radius-server host** *ip-address* [**acct-port** *udp-port*] [**auth-port** *udp-port*] [**test username** *name* [**idle-time** *time*] [**ignore-acct-port**] [**ignoreauth-port**]] [**key** *string*]	Configures the RADIUS server parameters: • **acct-port** *udp-port*—Specifies 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*—Specifies 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*—Enables automatic testing of the RADIUS server status, and specifies the username to be used. • **idle-time** *time*—Sets 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**—Disables testing on the RADIUS-server accounting port. • **ignoreauth-port**—Disables testing on the RADIUS-server authentication port. • For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running 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. You can also configure the authentication and encryption key by using the **radius-server key** {**0** *string* | **7** *string* | *string*} global configuration command.

	Command	Purpose
Step 5	**interface** *interface-id*	Specifies the port to be configured and enters interface configuration mode.
Step 6	**authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*	Configures a critical VLAN to move hosts on the port if the RADIUS server is unreachable: • **authorize**—Moves any new hosts trying to authenticate to the user-specified critical VLAN. • **reinitialize**—Moves all authorized hosts on the port to the user-specified critical VLAN.
Step 7	**switchport voice vlan** *vlan-id*	Specifies the voice VLAN for the port. The voice VLAN cannot be the same as the critical data VLAN configured in Step 6.
Step 8	**authentication event server dead action authorize voice**	Configures critical voice VLAN to move data traffic on the port to the voice VLAN if the RADIUS server is unreachable.
Step 9	**end**	Returns to privileged EXEC mode.
Step 10	**show authentication interface** *interface-id*	(Optional) Verifies your entries.

This example shows how to configure the inaccessible authentication bypass feature and configure the critical voice VLAN:

```
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)# interface gigabitethernet 1/0/1
Switch(config)# radius-server deadtime 60
Switch(config-if)# authentication event server dead action reinitialicze vlan 20
Switch(config-if)# switchport voice vlan
Switch(config-if)# authentication event server dead action authorize voice
Switch(config-if)# end
```

Configuring 802.1x Authentication with Wake-on-LAN

Beginning in privileged EXEC mode, follow these steps to enable 802.1x authentication with WoL. This procedure is optional.

	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 "802.1x Authentication Configuration Guidelines" section on page 1-38.
Step 3	**authentication control-direction {both | in}**	Enable 802.1x authentication with 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.

These examples show how to enable 802.1x authentication with WoL and set the port as bidirectional:

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

Configuring MAC Authentication Bypass

Beginning in privileged EXEC mode, follow these steps to enable MAC authentication bypass. This procedure is optional.

	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 "802.1x Authentication Configuration Guidelines" section on page 1-38.
Step 3	**authentication port-control auto**	Enable 802.1x authentication on the port.
Step 4	**authentication order** [**mab**] {**webauth**}	Set the order of authentication methods. • **mab**—Add MAC authentication bypass (MAB) to the order of authentication methods. • **webauth**—Add web authentication to the order of authentication methods.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show authentication interface** *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 authentication order** interface configuration command.

This example shows how to enable MAC authentication bypass:

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

Configuring a MAC Authentication Bypass (MAB) Username and Password

Beginning in privileged EXEC mode, follow these steps to configure a MAB username and password. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**mab request format attribute 1 groupsize** {**1** | **2** | **4** | **12**} **separator**{**-** | **:** | **.**} {**lowercase** | **uppercase**}	Specifies the format of the MAC address in the User-Name attribute of MAB-generated Access-Request packets. group size—The number of hex nibbles to concatenate before insertion of a separator. A valid groupsize must be either 1, 2, 4, or 12. separator—The character that separates the hex nibbles according to group size. A valid separator must be either a hyphen, colon, or period. No separator is used for a group size of 12. {lowercase | uppercase}—Specifies if nonnumeric hex nibbles should be in lowercase or uppercase.

	Command	Purpose
Step 3	**mab request format attribute 2** {**0** | **7**} *<LINE>*	Specifies a custom (nondefault) value for the User-Password attribute in MAB-generated Access-Request packets. 0—Specifies a cleartext password. 7—Specifies an encrypted password. *<LINE>*—Specifies the password to be used in the User-Password attribute.
Step 4	**end**	Returns to privileged EXEC mode.

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

This example shows how to enable configurable MAC authentication bypass:

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

Configuring 802.1x User Distribution

Beginning in global configuration, follow these steps to configure a VLAN group and to map a VLAN to it:

	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.

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

```
Switch(config)# vlan group eng-dept vlan-list 10

switch(config)# show vlan group group-name eng-dept
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10
switch# show dot1x 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 to verify that the VLAN was added:

```
switch(config)# vlan group eng-dept vlan-list 30
switch(config)# 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# 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.

switch(config)# 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
switch(config)# show vlan-group all
```

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

Configuring NAC Layer 2 802.1x Validation

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

Beginning in privileged EXEC mode, follow these steps to configure NAC Layer 2 802.1x validation. The procedure is optional.

	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.
Step 4	**authentication periodic**	Enable periodic reauthentication of the client, which is disabled by default.
Step 5	**authentication timer reauthenticate**	Set reauthentication attempt for the client (set to one hour). This command affects the behavior of the switch only if periodic reauthentication is enabled.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show authentication interface** *interface-id*	Verify your 802.1x authentication configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

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

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# authentication periodic
Switch(config-if)# authentication timer reauthenticate
```

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 1-32.

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.

Beginning in privileged EXEC mode, follow these steps to configure a switch as an authenticator:

	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**	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.

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
```

Beginning in privileged EXEC mode, follow these steps to configure a switch as a supplicant:

	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.

	Command	Purpose
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	**dot1x supplicant controlled transient**	(Optional) Configure the switch to block traffic exiting the supplicant port during the authentication period. **Note** We strongly recommend using this command when BPDU guard is enabled on the authenticator switch.
Step 8	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 9	**switchport trunk encapsulation dot1q**	Set the port to trunk mode.
Step 10	**switchport mode trunk**	Configure the interface as a VLAN trunk port.
Step 11	**dot1x pae supplicant**	Configure the interface as a port access entity (PAE) supplicant.
Step 12	**dot1x credentials** *profile-name*	Attach the 802.1x credentials profile to the interface.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show running-config interface** *interface-id*	Verify your configuration.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

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)# dot1x supplicant controlled transient
Switch(config)# interface gigabitethernet2/0/1
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 NEAT with Auto Smartports Macros

You can also use an Auto Smartports user-defined macro instead of the switch VSA to configure the authenticator switch. For information, see the *Auto Smartports Configuration Guide*.

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.

Beginning in privileged EXEC mode:

	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**	Enables AAA.
Step 4	**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 5	**radius-server vsa send authentication**	Configure the radius vsa send authentication.
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.

Configuring a Downloadable Policy

Beginning in privileged EXEC mode:

	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 commands.
Step 9	**ip device tracking probe** [**count** | **interval** | **use-svi**]	(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. • **interval** *interval*—Sets the number of seconds that the switch waits for a response before resending the ARP probe. The range is from 30 to 300 seconds. The default is 30 seconds. • **use-svi**—Uses the switch virtual interface (SVI) IP address as source of ARP probes.
Step 10	**radius-server vsa send authentication**	Configures the network access server to recognize and use vendor-specific attributes. **Note** The downloadable ACL must be operational.
Step 11	**end**	Returns to privileged EXEC mode.

	Command	Purpose
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.

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 gigabitethernet1/0/2
Switch(config-if)# ip access-group default_acl in
Switch(config-if)# exit
```

Configuring VLAN ID-based MAC Authentication

Beginning in privileged EXEC mode, follow these 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*: http://www.cisco.com/en/US/docs/ios/debug/command/reference/db_q1.html#wp1123741

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

```
Switch# configure 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

Beginning in privileged EXEC mode, follow these 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.

	Command	Purpose
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.

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 gigabitethernet1/0/2
Switch(config)# authentication order dot1x webauth
```

Configuring Open1x

Beginning in privileged EXEC mode:

	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-domain** | **multi-host** | **single-host**]	(Optional) Set the authorization manager mode on a port.
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 reauthentication on a port.
Step 9	**authentication port-control** {**auto** | **force-authorized** | f**orce-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.

This example shows how to configure open 1x on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/2
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
```

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.

Beginning in privileged EXEC mode, follow these steps to disable 802.1x authentication on the port. This procedure is optional.

	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** *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 802.1x on the port but does not allow clients connected to the port to be authorized, use the **dot1x pae authenticator** interface configuration command.

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

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# no dot1x pae authenticator
```

Resetting the 802.1x Authentication Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the 802.1x authentication configuration to the default values. This procedure is optional.

	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 802.1x parameters to the default values.
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.

Displaying 802.1x Statistics and Status

To display 802.1x statistics for all ports, use the **show dot1x all statistics** privileged EXEC command. To display 802.1x statistics for a specific port, use the **show dot1x statistics interface** *interface-id* privileged EXEC command.

To display the 802.1x administrative and operational status for the switch, use the **show dot1x all** [**details** | **statistics** | **summary**] privileged EXEC command. To display the 802.1x administrative and operational status for a specific port, use the **show dot1x interface** *interface-id* privileged EXEC command.

Beginning with Cisco IOS Release 12.2(55)SE, you can use the **no dot1x logging verbose** global configuration command to filter verbose 802.1x authentication messages. See the "Authentication Manager CLI Commands" section on page 1-9.

For detailed information about the fields in these displays, see the command reference for this release.