
# Configuring 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication to prevent unauthorized client devices from gaining access to the network.

This chapter consists of the following major sections:



**Note** For complete syntax and usage information for the switch commands used in this chapter, refer to the *Cisco IOS Command Reference for the Catalyst 4000 Family Switch* and related publications at http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121cgcr/index.htm

## Understanding 802.1x Port-Based Authentication

**Note** 802.1x authentication will not work unless the switch is able to route packets to the configured RADIUS server. You can verify that the switch is able to route packets to the RADIUS server by pinging the server from the switch.

The IEEE 802.1x standard defines 802.1x port-based authentication as a client-server based access control and authentication protocol that restricts unauthorized clients from connecting to a LAN through publicly accessible ports. The authentication server validates each client connected to a switch port before making available any services offered by the switch or the LAN.

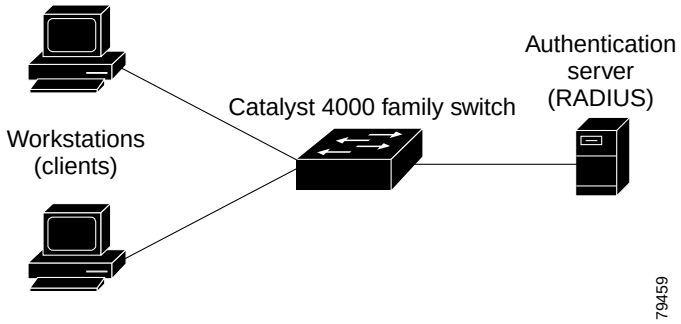
Until the client is authenticated, 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

To configure 802.1x on this switch, you need to understand the concepts covered in the following sections:

# Device Roles

With 802.1x port-based authentication, network devices have specific roles. Figure 22-1 shows the roles of each device.

***Figure 22-1 802.1x Device Roles***




- Client—The workstation that requests access to the LAN, switch services, and responds to requests from the switch. The workstation must be running 802.1x-compliant client software such as that offered with the Microsoft Windows XP operating system.

  **Note** For more information on Windows XP network connectivity and 802.1x authentication issues, see the Microsoft Knowledge Base article at this URL: 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 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. The Remote Authentication Dial-In User Service (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. 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 physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client. The switch includes the RADIUS client, which is responsible for encapsulating and decapsulating the EAP frames and interacting with the authentication server.

  When the switch receives EAPOL frames and relays them to the authentication server, the Ethernet header is stripped and the remaining EAP frame is re-encapsulated in the RADIUS format. The EAP frames are not modified or examined 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 other Catalyst 4000 family switches, the Catalyst 3550 multilayer switch, the Catalyst 2950 switch, or a wireless access point. These devices must be running software that supports the RADIUS client and 802.1x.

# Authentication Initiation and Message Exchange

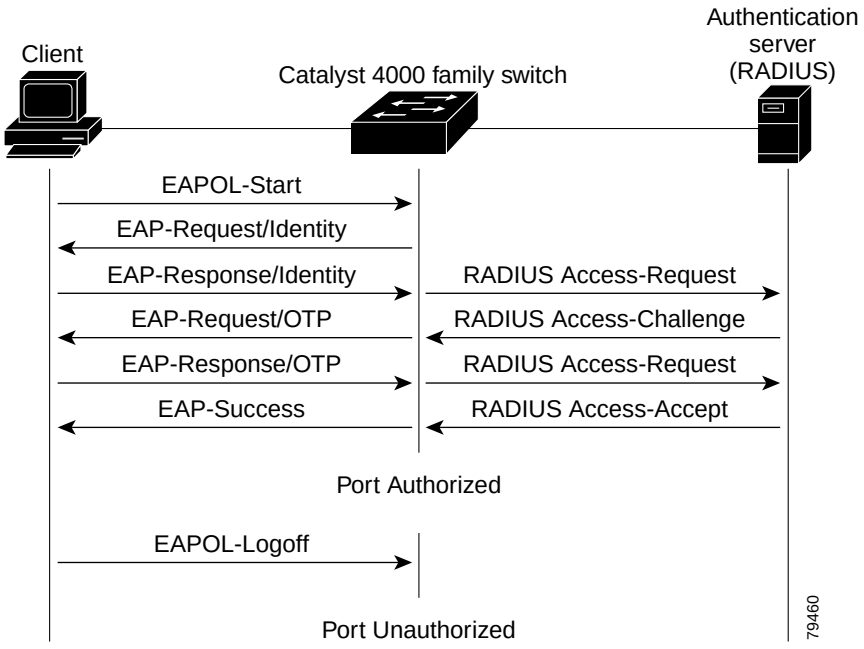
The switch or the client can initiate authentication. If you enable authentication on a port by using the **dot1x port-control auto** interface configuration command, the switch must initiate authentication when it determines that the port link state has changed. It then sends an EAP-request/identity frame to the client to request its identity (typically, the switch sends an initial identity/request frame followed by one or more requests for authentication information). Upon receipt of the frame, the client responds with an EAP-response/identity frame.

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

If 802.1x is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client transmits 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.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.

The specific exchange of EAP frames depends on the authentication method being used. Figure 22-2 shows a message exchange initiated by the client using the One-Time Password (OTP) authentication method with a RADIUS server.

***Figure 22-2 Message Exchange***



# Ports in Authorized and Unauthorized States

The switch port state determines whether or not the client is granted access to the network. The port starts in the unauthorized state. While in this state, the port disallows all ingress and egress traffic except for 802.1x protocol packets. When a client is successfully authenticated, the port transitions to the authorized state, allowing all traffic for the client to flow normally.

If a client that does not support 802.1x is connected 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 protocol, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request 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 can control the port authorization state with the **dot1x port-control** interface configuration command and these keywords:

- **force-authorized**—Disables 802.1x authentication and causes the port to transition to the authorized state without any authentication exchange required. The port transmits 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 interface.
- **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 transitions 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's 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 retransmit 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 transition to the unauthorized state.

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

# Supported Topologies

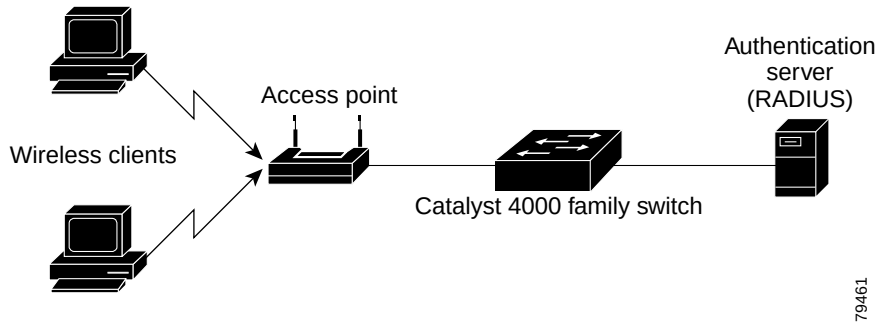
The 802.1x port-based authentication supports two topologies:

- Point-to-point
- Wireless LAN

In a point-to-point configuration (see Figure 22-1 on page 22-2), only one client can be connected to the 802.1x-enabled switch port. The switch detects the client 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.

Figure 22-3 shows 802.1x port-based authentication in a wireless LAN. The 802.1x port is configured as a multiple-host port that is authorized as soon as a client is authenticated. When the port is authorized, all other hosts indirectly attached to the port are granted access to the network. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), the switch denies access to the network to all attached clients. In this topology, the wireless access point is responsible for authenticating clients attached to it, and the wireless access point acts as a client to the switch.

***Figure 22-3 Wireless LAN Example***




# How to Configure 802.1x

These sections describe how to configure 802.1x:

- Default 802.1x Configuration, page 22-6
- 802.1x Configuration Guidelines, page 22-6
- Enabling 802.1x Authentication, page 22-7 (required)
- Configuring Switch-to-RADIUS-Server Communication, page 22-8 (required)
- Enabling Periodic Re-Authentication, page 22-9 (optional)
- Manually Re-Authenticating a Client Connected to a Port, page 22-10 (optional)
- Changing the Quiet Period, page 22-10 (optional)
- Changing the Switch-to-Client Retransmission Time, page 22-11 (optional)
- Setting the Switch-to-Client Frame-Retransmission Number, page 22-12 (optional)
- Enabling Multiple Hosts, page 22-12 (optional)
- Resetting the 802.1x Configuration to the Default Values, page 22-13 (optional)

# Default 802.1x Configuration

Table 22-1 shows the default 802.1x configuration.

***Table 22-1 Default 802.1x Configuration***

| Feature | Default Setting |
|---|---|
| Authentication, authorization, and accounting (AAA) | Disabled |
| RADIUS server<br>• IP address<br>• UDP authentication port<br>• Key | <br>• None specified<br>• 1812<br>• None specified |
| Per-interface 802.1x protocol enable state | Disabled (force-authorized)<br>The port transmits and receives normal traffic without 802.1x-based authentication of the client. |
| Periodic re-authentication | Disabled |
| Time between re-authentication attempts | 3600 sec |
| Quiet period | 60 sec<br>Number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. |
| Retransmission time | 30 sec<br>Number of seconds that the switch should wait for a response to an EAP request/identity frame from the client before retransmitting the request. |
| Maximum retransmission number | 2<br>Number of times that the switch will send an EAP-request/identity frame before restarting the authentication process. |
| Multiple host support | Disabled |
| Client timeout period | 30 sec<br>When relaying a request from the authentication server to the client, the amount of time the switch waits for a response before retransmitting the request to the client. |
| Authentication server timeout period | 30 sec<br>When relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before retransmitting the response to the server. This setting is not configurable. |

# 802.1x Configuration Guidelines

Keep these guidelines in mind when configuring 802.1x authentication:

- When 802.1x is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.

- The 802.1x protocol is supported on both Layer 2 static-access ports and Layer 3 routed ports, but it is not supported on these port types:
  - Trunk port—If you try to enable 802.1x on a trunk port, an error message appears, and 802.1x is not enabled. If you try to change the mode of an 802.1x-enabled port to trunk, the port mode is not changed.
  - Default ports—All ports default as dynamic-access ports (auto). Use the **no switchport** command to access a router port.
  - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x on a dynamic port, an error message appears, and 802.1x is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, the port mode is not changed.
  - EtherChannel port—Before enabling 802.1x on the port, you must first remove it from the EtherChannel. If you try to enable 802.1x on an EtherChannel or on an active port in an EtherChannel, an error message appears, and 802.1x is not enabled. If you enable 802.1x on a not-yet active port of an EtherChannel, the port does not join the EtherChannel.
  - Switched Port Analyzer (SPAN) destination port—You can enable 802.1x on a port that is a SPAN destination port; however, 802.1x is disabled until the port is removed as a SPAN destination. You can enable 802.1x on a SPAN source port.

# Enabling 802.1x Authentication

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

The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle, the authentication process stops, and no other authentication methods are attempted.

Beginning in privileged EXEC mode, follow this procedure to configure 802.1x port-based authentication.

| | Task | Command |
|---|---|---|
| **Step 1** | Enter global configuration mode. | `Switch # configure terminal` |
| **Step 2** | Enable AAA. | `Switch(config)# aaa new-model` |
| **Step 3** | Create an 802.1x authentication method list.<br><br>To create a default list that is used when a named list is not specified in the **authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all interfaces.<br><br>Enter at least one of these keywords:<br>• **group radius**—Use the list of all RADIUS servers for authentication.<br>• **none**—Use no authentication. The client is automatically authenticated by the switch without using the information supplied by the client. | `Switch(config)# aaa authentication dot1x {default} method1 [method2...]` |

| | Task | Command |
|---|---|---|
| **Step 4** | Enter interface configuration mode and specify the interface to be enabled for 802.1x authentication. | `Switch(config)# interface interface-id` |
| **Step 5** | Enable 802.1x authentication on the interface.<br><br>For feature interaction information with trunk, dynamic, dynamic-access, EtherChannel, secure, and SPAN ports, see the "802.1x Configuration Guidelines" section on page 22-6. | `Switch(config-if)# dot1x port-control auto` |
| **Step 6** | Return to privileged EXEC mode. | `Switch(config-if)# end` |
| **Step 7** | Verify your entries.<br><br>Check the Status column in the 802.1x Port Summary section of the display. An enabled status means the port-control value is set either to **auto** or to **force-unauthorized**. | `Switch # show dot1x all` |
| **Step 8** | (Optional) Save your entries in the configuration file. | `Switch # copy running-config startup-config` |

To disable AAA, use the **no aaa new-model** global configuration command.

To disable 802.1x AAA authentication, use the **no aaa authentication dot1x** {**default** | *list-name*} *method1* [*method2*...] global configuration command.

To disable 802.1x authentication, use the **dot1x port-control force-authorized** or the **no dot1x port-control** interface configuration command.

This example shows how to enable AAA and 802.1x on Fast Ethernet port 0/1:

```
Switch# configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa authentication dot1x default group radius
Switch(config)# interface fastethernet0/1
Switch(config-if)# dot1x port-control auto
Switch(config-if)# end
```

# Configuring Switch-to-RADIUS-Server Communication

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

Beginning in privileged EXEC mode, follow this procedure to configure the RADIUS server parameters on the switch.

| | Task | Command |
|---|---|---|
| **Step 1** | Enter global configuration mode. | `Switch# configure terminal` |
| **Step 2** | Configure the RADIUS server parameters on the switch.<br><br>For *hostname* \| *ip-address*, specify the hostname or IP address of the remote RADIUS server.<br><br>For **auth-port** *port-number*, specify the UDP destination port for authentication requests. The default is 1812.<br><br>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.<br><br>**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.<br><br>If you want to use multiple RADIUS servers, re-enter this command. | `Switch(config)# radius-server host {hostname \| ip-address} auth-port port-number key string` |
| **Step 3** | Return to privileged EXEC mode. | `Switch(config)# end` |
| **Step 4** | Verify your entries. | `Switch# show running-config` |
| **Step 5** | (Optional) Save your entries in the configuration file. | `Switch# copy running-config startup-config` |

To delete 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.

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.

# Enabling Periodic Re-Authentication

You can enable periodic 802.1x client re-authentication and specify how often it occurs. If you do not specify a time value before enabling re-authentication, the time between re-authentication attempts is 3600 sec.

Automatic 802.1x client re-authentication is a per interface setting and can be set for clients connected to individual ports. To manually re-authenticate the client connected to a specific port, see the "Manually Re-Authenticating a Client Connected to a Port" section on page 22-10.

Beginning in privileged EXEC mode, follow this procedure to enable periodic re-authentication of the client and to configure the number of seconds between re-authentication attempts:

| | Task | Command |
|---|---|---|
| **Step 1** | Enter global configuration mode. | `Switch# configure terminal` |
| **Step 1** | Enter interface configuration mode and specify the interface to be enabled for periodic re-authentication. | `Switch(config)# interface` *interface-id* |
| **Step 2** | Enable periodic re-authentication of the client, which is disabled by default. | `Switch(config-if)# dot1x re-authentication` |
| **Step 3** | Set the number of seconds between re-authentication attempts.<br>The range is 1 to 65,535; the default is 3600 seconds.<br>This command affects the behavior of the switch only if periodic re-authentication is enabled. | `Switch(config)# dot1x timeout reauth-period` *seconds* |
| **Step 4** | Return to privileged EXEC mode. | `Switch(config)# end` |
| **Step 5** | Verify your entries. | `Switch# show dot1x all` |
| **Step 6** | (Optional) Save your entries in the configuration file. | `Switch(config)# copy running-config startup-config` |

To disable periodic re-authentication, use the **no dot1x re-authentication** interface configuration command. To return to the default number of seconds between re-authentication attempts, use the **no dot1x timeout reauth-period** global configuration command.

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

```
Switch(config)# dot1x re-authentication
Switch(config)# dot1x timeout reauth-period 4000
```

# Manually Re-Authenticating a Client Connected to a Port

You can manually re-authenticate the client connected to a specific port at any time by entering the **dot1x re-authenticate interface** *interface-id* privileged EXEC command. If you want to enable or disable periodic re-authentication, see the "Enabling Periodic Re-Authentication" section on page 22-9.

This example shows how to manually re-authenticate the client connected to Fast Ethernet port 0/1:

```
Switch# dot1x re-authenticate interface fastethernet0/1
Starting reauthentication on FastEthernet0/1
```

# 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 idle time is determined by the **quiet-period** value. 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 this procedure to change the quiet period:

| | Task | Command |
|---|---|---|
| **Step 1** | Enter global configuration mode. | `Switch# configure terminal` |
| **Step 2** | Enter interface configuration mode and specify the interface to be enabled for timeout **quiet-period**. | `Switch(config)# interface` *interface-id* |
| **Step 3** | Set the number of seconds that the switch remains in the **quiet-period** following a failed authentication exchange with the client.<br>The range is 0 to 65,535 seconds; the default is 60. | `Switch(config)# dot1x timeout quiet-period` *seconds* |
| **Step 4** | Return to privileged EXEC mode. | `Switch(config)# end` |
| **Step 5** | Verify your entries. | `Switch# show dot1x all` |
| **Step 6** | (Optional) Save your entries in the configuration file. | `Switch# copy running-config startup-config` |

To return to the default quiet-period, use the **no dot1x timeout quiet-period** configuration command. This example shows how to set the **quiet-period** on the switch to 30 seconds:

```
Switch(config)# 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 retransmits 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:

| | Task | Command |
|---|---|---|
| **Step 1** | Enter global configuration mode. | `Switch# configure terminal` |
| **Step 2** | Enter interface configuration mode and specify the interface to be enabled for timeout tx-period. | `Switch(config)# interface` *interface-id* |
| **Step 3** | Set the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before retransmitting the request.<br>The range is 1 to 65,535 seconds; the default is 30. | `Switch(config-if)# dot1x timeout tx-period` *seconds* |
| **Step 4** | Return to privileged EXEC mode. | `Switch(config)# end` |
| **Step 5** | Verify your entries. | `Switch# show dot1x all` |
| **Step 6** | (Optional) Save your entries in the configuration file. | `Switch# copy running-config startup-config` |

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

This example shows how to set the retransmission time to 60 seconds:

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

# Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP-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:

| | Task | Command |
|---|---|---|
| **Step 1** | Enter global configuration mode. | `Switch# configure terminal` |
| **Step 2** | Enter interface configuration mode and specify the interface to be enabled for **max-req**. | `Switch(config)# interface` *interface-id* |
| **Step 3** | 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. | `Switch(config-if)# dot1x max-req` *count* |
| **Step 4** | Return to privileged EXEC mode. | `Switch(config)# end` |
| **Step 5** | Verify your entries. | `Switch# show dot1x all` |
| **Step 6** | (Optional) Save your entries in the configuration file. | `Switch# copy running-config startup-config` |

To return to the default retransmission number, use the **no dot1x max-req** global 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)# dot1x max-req 5
```

# Enabling Multiple Hosts

You can attach multiple hosts to a single 802.1x-enabled port as shown in Figure 22-3 on page 22-5. In this mode, only one of the attached hosts must be successfully authorized for all hosts to be granted network access. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), all attached clients are denied access to the network.

Beginning in privileged EXEC mode, follow these steps to allow multiple hosts (clients) on an 802.1x-authorized port that has the **dot1x port-control** interface configuration command set to **auto**.

| | Task | Command |
|---|---|---|
| **Step 1** | Enter global configuration mode. | `Switch# configure terminal` |
| **Step 2** | Enter interface configuration mode and specify the interface to which multiple hosts are indirectly attached. | `Switch(config)# interface` *interface-id* |
| **Step 3** | Allow multiple hosts (clients) on an 802.1x-authorized port.<br>Make sure that the **dot1x port-control** interface configuration command set is set to **auto** for the specified interface. | `Switch(config-if)# dot1x multiple-hosts` |
| **Step 4** | Return to privileged EXEC mode. | `Switch(config-if)# end` |
| **Step 5** | Verify your entries. | `Switch# show dot1x all interface` *interface-id* |
| **Step 6** | (Optional) Save your entries in the configuration file. | `Switch# copy running-config startup-config` |

To disable multiple hosts on the port, use the **no dot1x multiple-hosts** interface configuration command.

This example shows how to enable 802.1x on Fast Ethernet interface 0/1 and to allow multiple hosts:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# dot1x port-control auto
Switch(config-if)# dot1x multiple-hosts
```

## Resetting the 802.1x Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the 802.1x configuration to the default values:

| | Task | Command |
|---|---|---|
| **Step 1** | Enter global configuration mode. | `Switch# configure terminal` |
| **Step 2** | Reset the configurable 802.1x parameters to the default values. | `Switch(config)# dot1x default` |
| **Step 3** | Return to privileged EXEC mode. | `Switch(config)# end` |
| **Step 4** | Verify your entries. | `Switch# show dot1x all` |
| **Step 5** | (Optional) Save your entries in the configuration file. | `Switch# copy running-config startup-config` |

# Displaying 802.1x Statistics and Status

To display 802.1x statistics for all interfaces, use the **show dot1x statistics** privileged EXEC command. To display 802.1x statistics for a specific interface, 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** privileged EXEC command. To display the 802.1x administrative and operational status for a specific interface, use the **show dot1x interface** *interface-id* privileged EXEC command.

For detailed information about the fields in these displays, refer to the *Cisco IOS Command Reference for the Catalyst 4000 Family Switch*.