Web-Based Authentication

This chapter describes how to configure web-based authentication on the device. It contains these sections:

- Web-Based Authentication Overview, on page 1
- How to Configure Web-Based Authentication, on page 10
- Verifying Web-Based Authentication Status, on page 22

Web-Based Authentication Overview

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

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

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

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

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

Note HTTPS traffic interception for central web authentication redirect is not supported.

Note You should use global parameter-map (for method-type, custom, and redirect) only for using the same web authentication methods like consent, web consent, and webauth, for all the clients and SSIDs. This ensures that all the clients have the same web-authentication method.

If the requirement is to use Consent for one SSID and Web-authentication for another SSID, then you should use two named parameter-maps. You should configure Consent in first parameter-map and configure webauth in second parameter-map.

Note The traceback that you receive when webauth client tries to do authentication does not have any performance or behavioral impact. It happens rarely when the context for which FFM replied back to EPM for ACL application is already dequeued (possibly due to timer expiry) and the session becomes 'unauthorized'.

Based on where the web pages are hosted, the local web authention can be categorozied as follows:

- *Internal*—The internal default HTML pages (Login, Success, Fail, and Expire) in the controller are used during the local web authentication.
- *Customized*—The customized web pages (Login, Success, Fail, and Expire) are downloaded onto the controller and used during the local web authentication.
- *External*—The customized web pages are hosted on the external web server instead of using the in-built or custom web pages.

Based on the various web authentication pages, the types of web authentication are as follows:

- *Webauth*—This is a basic web authentication. Herein, the controller presents a policy page with the user name and password. You need to enter the correct credentials to access the network.
- *Consent or web-passthrough*—Herein, the controller presents a policy page with the Accept or Deny buttons. You need to click the Accept button to access the network.
- *Webconsent*—This is a combination of webauth and consent web authentication types. Herein, the controller presents a policy page with Accept or Deny buttons along with user name or password. You need to enter the correct credentials and click the Accept button to access the network.

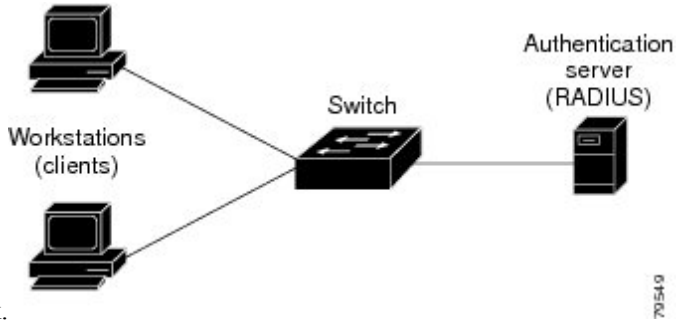
Device Roles

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

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

Figure 1: Web-Based Authentication Device Roles

This figure shows the roles of these devices in a



network.

Host Detection

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

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

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

Session Creation

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

- Reviews the exception list.

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

- Reviews for authorization bypass

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

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

- Sets up the HTTP intercept ACL

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

Authentication Process

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

- The user initiates an HTTP session.

- The HTTP traffic is intercepted, and authorization is initiated. The switch sends the login page to the user. The user enters a username and password, and the switch sends the entries to the authentication server.
- If the authentication succeeds, the switch downloads and activates the user's access policy from the authentication server. The login success page is sent to the user.
- If the authentication fails, the switch sends the login fail page. The user retries the login. If the maximum number of attempts fails, the switch sends the login expired page, and the host is placed in a watch list. After the watch list times out, the user can retry the authentication process.
- If the authentication server does not respond to the switch, and if an AAA fail policy is configured, the switch applies the failure access policy to the host. The login success page is sent to the user.
- The switch reauthenticates a client when the host does not respond to an ARP probe on a Layer 2 interface, or when the host does not send any traffic within the idle timeout on a Layer 3 interface.
- The feature applies the downloaded timeout or the locally configured session timeout.

Note Beginning with Cisco IOS XE Denali 16.1.1 and later, the default session timeout value for web-based authentication on WLC is 1800 seconds. The default session timeout value was infinite seconds, prior to Cisco IOS XE Denali 16.1.1.

- If the terminate action is RADIUS, the feature sends a nonresponsive host (NRH) request to the server. The terminate action is included in the response from the server.
- If the terminate action is default, the session is dismantled, and the applied policy is removed.

Local Web Authentication Banner

With Web Authentication, you can create a default and customized web-browser banners that appears when you log in to a switch.

The banner appears on both the login page and the authentication-result pop-up pages. The default banner messages are as follows:

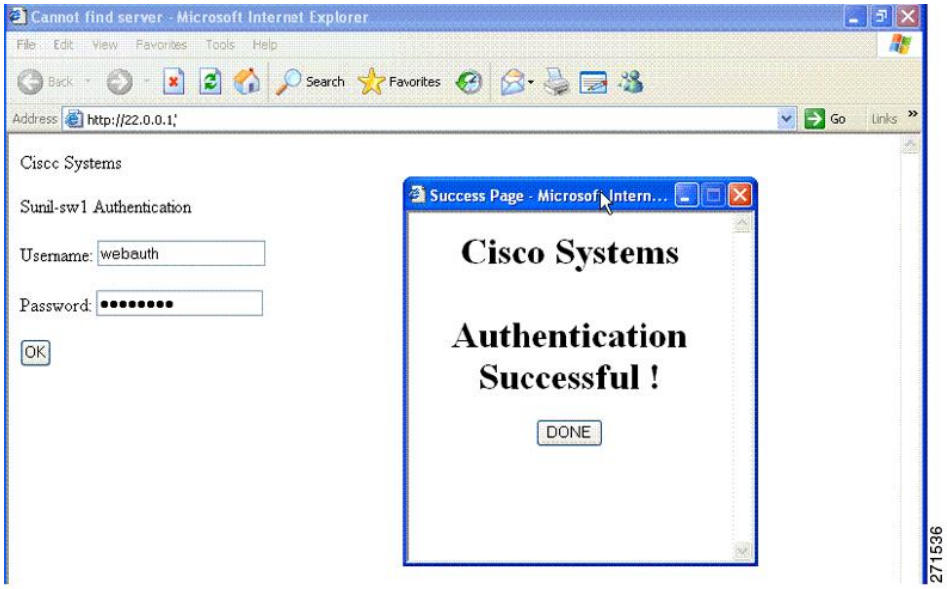
- *Authentication Successful*
- *Authentication Failed*
- *Authentication Expired*

The Local Web Authentication Banner can be configured in legacy and new-style (Session-aware) CLIs as follows:

- Legacy mode—Use the **ip admission auth-proxy-banner http** global configuration command.
- New-style mode—Use the **parameter-map type webauth global banner** global configuration command.

The default banner *Cisco Systems* and *Switch host-name Authentication* appear on the Login Page. *Cisco Systems* appears on the authentication result pop-up page.

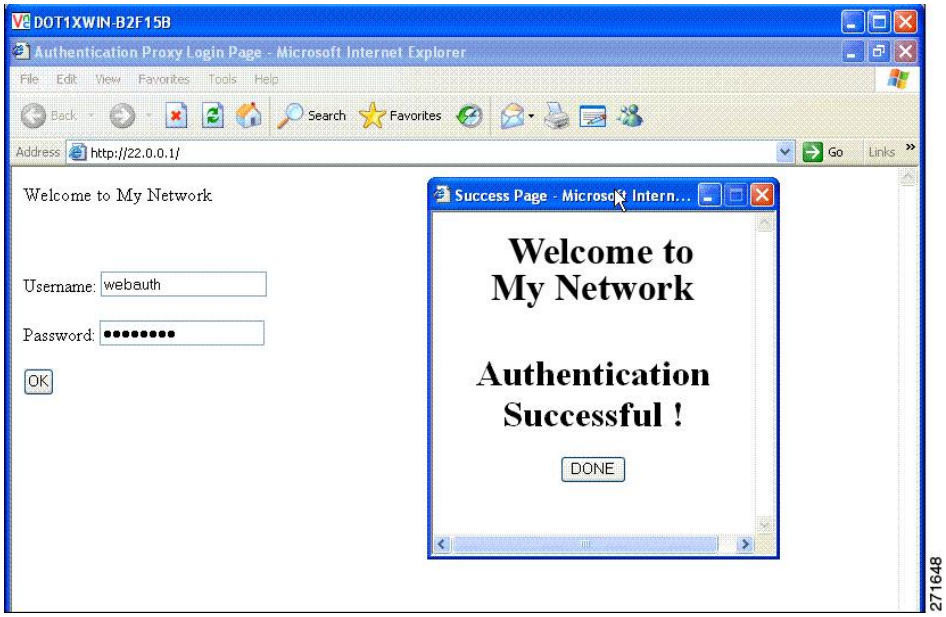
Figure 2: Authentication Successful Banner



The banner can be customized as follows:

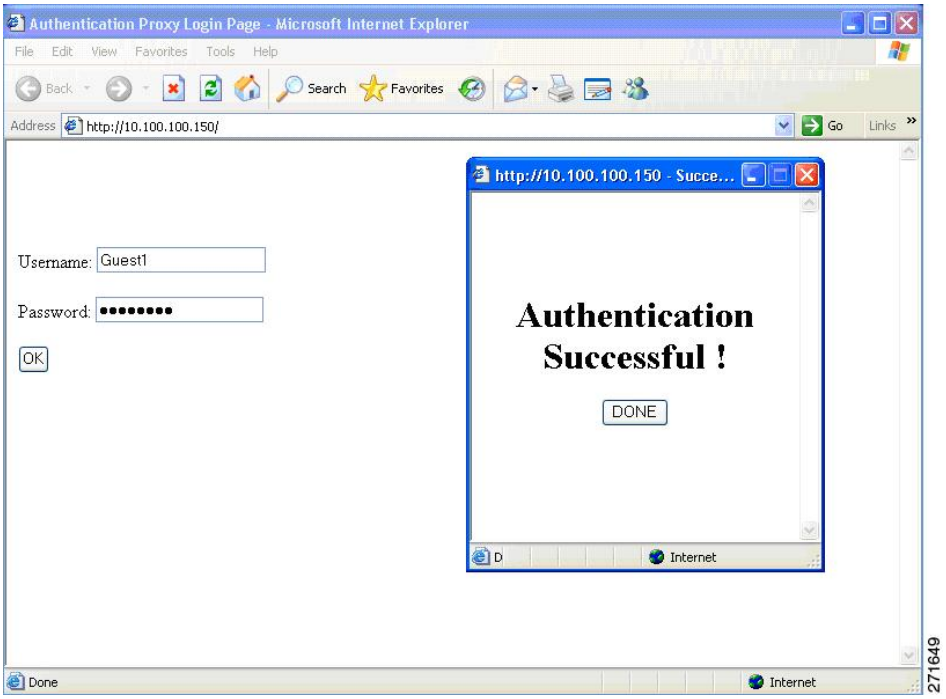
- Add a message, such as switch, router, or company name to the banner:
  - Legacy mode—Use the **ip admission auth-proxy-banner http** *banner-text*global configuration command.
  - New-style mode—Use the **parameter-map type webauth global banner** global configuration command.
- Add a logo or text file to the banner:
  - Legacy mode—Use the **ip admission auth-proxy-banner http** *file-path* global configuration command.
  - New-style mode—Use the **parameter-map type webauth global banner** global configuration command.

Figure 3: Customized Web Banner



If you do not enable a banner, only the username and password dialog boxes appear in the web authentication login screen, and no banner appears when you log into the switch.

Figure 4: Login Screen With No Banner



Web Authentication Customizable Web Pages

During the web-based authentication process, the switch internal HTTP server hosts four HTML pages to deliver to an authenticating client. The server uses these pages to notify you of these four-authentication process states:

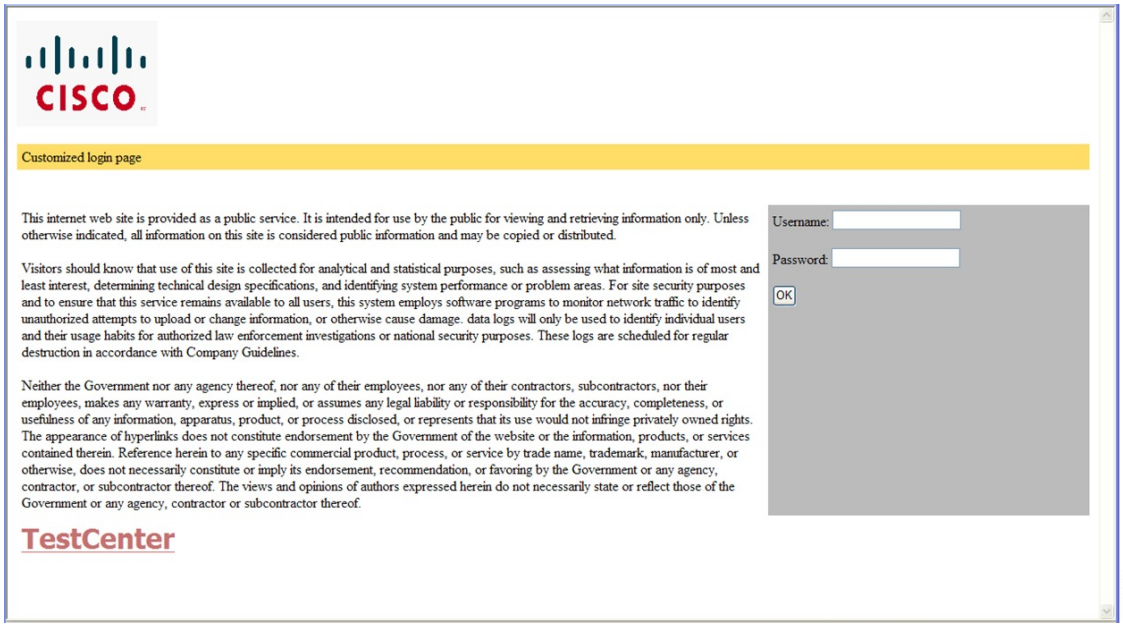
- Login—Your credentials are requested.
- Success—The login was successful.
- Fail—The login failed.
- Expire—The login session has expired because of excessive login failures.

Guidelines

- You can substitute your own HTML pages for the default internal HTML pages.
- You can use a logo or specify text in the *login*, *success*, *failure*, and *expire* web pages.
- On the banner page, you can specify text in the login page.
- The pages are in HTML.
- You must include an HTML redirect command in the success page to access a specific URL.
- The URL string must be a valid URL (for example, http://www.cisco.com). An incomplete URL might cause *page not found* or similar errors on a web browser.
- If you configure web pages for HTTP authentication, they must include the appropriate HTML commands (for example, to set the page time out, to set a hidden password, or to confirm that the same page is not submitted twice).
- The CLI command to redirect users to a specific URL is not available when the configured login form is enabled. The administrator should ensure that the redirection is configured in the web page.
- If the CLI command redirecting users to specific URL after authentication occurs is entered and then the command configuring web pages is entered, the CLI command redirecting users to a specific URL does not take effect.
- Configured web pages can be copied to the switch boot flash or flash.
- On stackable switches, configured pages can be accessed from the flash on the active switch or member switches.
- The login page can be on one flash, and the success and failure pages can be another flash (for example, the flash on the active switch or a member switch).
- You must configure all four pages.
- The banner page has no effect if it is configured with the web page.
- All of the logo files (image, flash, audio, video, and so on) that are stored in the system directory (for example, flash, disk0, or disk) and that must be displayed on the login page must use *web_auth_<filename>* as the file name.
- The configured authentication proxy feature supports both HTTP and SSL.

You can substitute your HTML pages for the default internal HTML pages. You can also specify a URL to which users are redirected after authentication occurs, which replaces the internal Success page.

Figure 5: Customizable Authentication Page



Authentication Proxy Web Page Guidelines

When configuring customized authentication proxy web pages, follow these guidelines:

- To enable the custom web pages feature, specify all four custom HTML files. If you specify fewer than four files, the internal default HTML pages are used.
- The four custom HTML files must be present on the flash memory of the switch. The maximum size of each HTML file is 8 KB.
- Any images on the custom pages must be on an accessible HTTP server. Configure an intercept ACL within the admission rule.
- Any external link from a custom page requires configuration of an intercept ACL within the admission rule.
- To access a valid DNS server, any name resolution required for external links or images requires configuration of an intercept ACL within the admission rule.
- If the custom web pages feature is enabled, a configured auth-proxy-banner is not used.
- If the custom web pages feature is enabled, the redirection URL for successful login feature is not available.
- To remove the specification of a custom file, use the **no** form of the command.

Because the custom login page is a public web form, consider these guidelines for the page:

- The login form must accept user entries for the username and password and must show them as **uname** and **pwd**.

- The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.

Redirection URL for Successful Login Guidelines

When configuring a redirection URL for successful login, consider these guidelines:

- If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and is not available in the CLI. You can perform redirection in the custom-login success page.
- If the redirection URL feature is enabled, a configured auth-proxy-banner is not used
- To remove the specification of a redirection URL, use the **no** form of the command.
- If the redirection URL is required after the web-based authentication client is successfully authenticated, then the URL string must start with a valid URL (for example, http://) followed by the URL information. If only the URL is given without http://, then the redirection URL on successful authentication might cause page not found or similar errors on a web browser.

Web-based Authentication Interactions with Other Features

Port Security

You can configure web-based authentication and port security on the same port. Web-based authentication authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through the port.

LAN Port IP

You can configure LAN port IP (LPIP) and Layer 2 web-based authentication on the same port. The host is authenticated by using web-based authentication first, followed by LPIP posture validation. The LPIP host policy overrides the web-based authentication host policy.

If the web-based authentication idle timer expires, the NAC policy is removed. The host is authenticated, and posture is validated again.

Gateway IP

You cannot configure Gateway IP (GWIP) on a Layer 3 VLAN interface if web-based authentication is configured on any of the switch ports in the VLAN.

You can configure web-based authentication on the same Layer 3 interface as Gateway IP. The host policies for both features are applied in software. The GWIP policy overrides the web-based authentication host policy.

ACLs

If you configure a VLAN ACL or a Cisco IOS ACL on an interface, the ACL is applied to the host traffic only after the web-based authentication host policy is applied.

For Layer 2 web-based authentication, it is more secure, though not required, to configure a port ACL (PACL) as the default access policy for ingress traffic from hosts connected to the port. After authentication, the web-based authentication host policy overrides the PACL. The Policy ACL is applied to the session even if there is no ACL configured on the port.

You cannot configure a MAC ACL and web-based authentication on the same interface.

You cannot configure web-based authentication on a port whose access VLAN is configured for VACL capture.

Context-Based Access Control

Web-based authentication cannot be configured on a Layer 2 port if context-based access control (CBAC) is configured on the Layer 3 VLAN interface of the port VLAN.

EtherChannel

You can configure web-based authentication on a Layer 2 EtherChannel interface. The web-based authentication configuration applies to all member channels.

How to Configure Web-Based Authentication

Default Web-Based Authentication Configuration

The following table shows the default web-based authentication configuration.

Table 1: Default Web-based Authentication Configuration

Feature	Default Setting
AAA	Disabled
RADIUS server • IP address • UDP authentication port • Key	• None specified • None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Web-Based Authentication Configuration Guidelines and Restrictions

- Web-based authentication is an ingress-only feature.
- You can configure web-based authentication only on access ports. Web-based authentication is not supported on trunk ports, EtherChannel member ports, or dynamic trunk ports.
- External web authentication, where the switch redirects a client to a particular host or web server for displaying login message, is not supported.
- You cannot authenticate hosts on Layer 2 interfaces with static ARP cache assignment. These hosts are not detected by the web-based authentication feature because they do not send ARP messages.

- By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.
- You must enable SISF-Based device tracking to use web-based authentication. By default, SISF-Based device tracking is disabled on a switch.
- You must configure at least one IP address to run the switch HTTP server. You must also configure routes to reach each host IP address. The HTTP server sends the HTTP login page to the host.
- Hosts that are more than one hop away might experience traffic disruption if an STP topology change results in the host traffic arriving on a different port. This occurs because the ARP and DHCP updates might not be sent after a Layer 2 (STP) topology change.
- Web-based authentication does not support VLAN assignment as a downloadable-host policy.
- Web-based authentication supports IPv6 in Session-aware policy mode. IPv6 Web-authentication requires at least one IPv6 address configured on the switch and IPv6 Snooping configured on the switchport.
- Web-based authentication and Network Edge Access Topology (NEAT) are mutually exclusive. You cannot use web-based authentication when NEAT is enabled on an interface, and you cannot use NEAT when web-based authentication is running on an interface.
- Identify the following RADIUS security server settings that will be used while configuring switch-to-RADIUS-server communication:
    - Host name
    - Host IP address
    - Host name and specific UDP port numbers
    - IP address and specific UDP port numbers

  The combination of the IP address and UDP port number creates a unique identifier, that enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service (for example, authentication) the second host entry that is configured functions as the failover backup to the first one. The RADIUS host entries are chosen in the order that they were configured.
- When you configure the RADIUS server parameters:
    - Specify the **key** *string* on a separate command line.
    - For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server.
    - When you specify the **key** *string*, use spaces within and at the end of the key. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon.
    - You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using with 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 transmit, and the radius-server key global configuration commands.

Note You need to configure some settings on the RADIUS server, including: the switch IP address, the key string to be shared by both the server and the switch, and the downloadable ACL (DACL). For more information, see the RADIUS server documentation.

- For a URL redirect ACL:
  - Packets that match a permit access control entry (ACE) rule are sent to the CPU for forwarding to the AAA server.
  - Packets that match a deny ACE rule are forwarded through the switch.
  - Packets that match neither the permit ACE rule or deny ACE rule are processed by the next dACL, and if there is no dACL, the packets hit the implicit-deny ACL and are dropped.

Configuring the Authentication Rule and Interfaces

Follow these steps to configure the authentication rule and interfaces:

Before you begin

SISF-Based device tracking is a prerequisite to Web Authentication. Ensure that you have enabled device tracking programmatically or manually.

For more information, see *Confguring SISF-Based Tracking*.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip admission name** *name* **proxy http** **Example:** `Device(config)# ip admission name webauth1 proxy http`	Configures an authentication rule for web-based authorization.

	Command or Action	Purpose
Step 4	**interface** *type slot/port* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode and specifies the ingress Layer 2 or Layer 3 interface to be enabled for web-based authentication. *type* can be fastethernet, gigabit ethernet, or tengigabitethernet.
Step 5	**ip access-group** *name* **Example:** `Device(config-if)# ip access-group webauthag`	Applies the default ACL.
Step 6	**ip admission name** **Example:** `Device(config)# ip admission name`	Configures an authentication rule for web-based authorization for the interface.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 8	**show ip admission** **Example:** `Device# show ip admission`	Displays the configuration.
Step 9	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring AAA Authentication

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Device> enable`	
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables AAA functionality.
Step 4	**aaa authentication login default group** {**tacacs+** | **radius**} **Example:** `Device(config)# aaa authentication login default group tacacs+`	Defines the list of authentication methods at login. **named_authentication_list** refers to any name that is not greater than 31 characters. **AAA_group_name** refers to the server group name. You need to define the server-group **server_name** at the beginning itself.
Step 5	**aaa authorization auth-proxy default group** {**tacacs+** | **radius**} **Example:** `Device(config)# aaa authorization auth-proxy default group tacacs+`	Creates an authorization method list for web-based authorization.
Step 6	**tacacs server** *server-name* **Example:** `Device(config)# tacacs server yourserver`	Specifies an AAA server.
Step 7	**address** {**ipv4 | ipv6**} *ip address* **Example:** `Device(config-server-tacacs)# address ipv4 10.0.1.12`	Configures the IP address for the TACACS server.

	Command or Action	Purpose
Step 8	**key** *string* **Example:** `Device(config-server-tacacs)# key cisco123`	Configures the authorization and encryption key used between the switch and the TACACS server.
Step 9	**exit** **Example:** `Device(config-server-tacacs)# exit`	Exits the TACACS server mode and enters the global configuration mode.
Step 10	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 11	**show running-config** **Example:** `Device# show running-config`	Verifies your entries.
Step 12	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Switch-to-RADIUS-Server Communication

Follow these steps to configure the RADIUS server parameters:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**ip radius source-interface** **Example:** `Device(config)# ip radius source-interface vlan 80`	Specifies that the RADIUS packets have the IP address of the indicated interface.
Step 4	**radius server** *server name* **Example:** `Device(config)# radius server rsim address ipv4 124.2.2.12`	(Optional) Specifies the IP address of the RADIUS server.
Step 5	**address** {**ipv4** | **ipv6**} *ip address* **Example:** `Device(config-radius-server)# address ipv4 10.0.1.2 auth-port 1550 acct-port 1560`	Configures the IP address for the RADIUS server.
Step 6	**key** *string* **Example:** `Device(config-radius-server)# key rad123`	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 7	**exit** **Example:** `Device(config-radius-server)# exit`	Exits the RADIUS server mode and enters the global configuration mode.
Step 8	**radius-server dead-criteria tries** *num-tries* **Example:** `Device(config)# radius-server dead-criteria tries 30`	Specifies the number of unanswered sent messages to a RADIUS server before considering the server to be inactive. The range of *num-tries* is 1 to 100.
Step 9	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring the HTTP Server

To use web-based authentication, you must enable the HTTP server within the Device. You can enable the server for either HTTP or HTTPS.

Note The Apple psuedo-browser will not open if you configure only the **ip http secure-server** command. You should also configure the **ip http server** command.

Follow the procedure given below to enable the server for either HTTP or HTTPS:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip http server** **Example:** `Device(config)# ip http server`	Enables the HTTP server. The web-based authentication feature uses the HTTP server to communicate with the hosts for user authentication.
Step 4	**ip http secure-server** **Example:** `Device(config)# ip http secure-server`	Enables HTTPS. You can configure custom authentication proxy web pages or specify a redirection URL for successful login. **Note** To ensure secure authentication when you enter the **ip http secure-server** command, the login page is always in HTTPS (secure HTTP) even if the user sends an HTTP request.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Customizing the Authentication Proxy Web Pages

You can configure web authentication to display four substitute HTML pages to the user in place of the Device default HTML pages during web-based authentication.

Follow these steps to specify the use of your custom authentication proxy web pages:

Before you begin

Store your custom HTML files on the Device flash memory.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip admission proxy http login page file** *device:login-filename* **Example:** `Device(config)# ip admission proxy http login page file disk1:login.htm`	Specifies the location in the Device memory file system of the custom HTML file to use in place of the default login page. The *device:* is flash memory.
Step 4	**ip admission proxy http success page file** *device:success-filename* **Example:** `Device(config)# ip admission proxy http success page file disk1:success.htm`	Specifies the location of the custom HTML file to use in place of the default login success page.
Step 5	**ip admission proxy http failure page file** *device:fail-filename* **Example:** `Device(config)# ip admission proxy http fail page file disk1:fail.htm`	Specifies the location of the custom HTML file to use in place of the default login failure page.

	Command or Action	Purpose
Step 6	**ip admission proxy http login expired page file** *device:expired-filename* **Example:** `Device(config)# ip admission proxy http login expired page file disk1:expired.htm`	Specifies the location of the custom HTML file to use in place of the default login expired page.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Specifying a Redirection URL for Successful Login

Follow these steps to specify a URL to which the user is redirected after authentication, effectively replacing the internal Success HTML page:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip admission proxy http success redirect** *url-string* **Example:** `Device(config)# ip admission proxy http success redirect www.example.com`	Specifies a URL for redirection of the user in place of the default login success page.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring the Web-Based Authentication Parameters

Follow these steps to configure the maximum number of failed login attempts before the client is placed in a watch list for a waiting period:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip admission max-login-attempts** *number* **Example:** `Device(config)# ip admission max-login-attempts 10`	Sets the maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Device# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Web-Based Authentication Local Banner

Follow these steps to configure a local banner on a switch that has web authentication configured.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip admission auth-proxy-banner http** [*banner-text* | *file-path*] **Example:** `Device(config)# ip admission auth-proxy-banner http C My Switch C`	Enables the local banner. (Optional) Create a custom banner by entering *C banner-text C* (where *C* is a delimiting character), or *file-path* that indicates a file (for example, a logo or text file) that appears in the banner.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Device# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Removing Web-Based Authentication Cache Entries

Follow these steps to remove web-based authentication cache entries:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**clear ip auth-proxy cache** {* | *host ip address*} **Example:** `Device# clear ip auth-proxy cache 192.168.4.5`	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.
Step 3	**clear ip admission cache** {* | *host ip address*} **Example:** `Device# clear ip admission cache 192.168.4.5`	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.

Verifying Web-Based Authentication Status

Use the commands in this topic to display the web-based authentication settings for all interfaces or for specific ports.

Table 2: Privileged EXEC show Commands

Command	Purpose
show authentication sessions method webauth	Displays the web-based authentication settings for all interfaces for fastethernet, gigabitethernet, or tengigabitethernet
show authentication sessions interface *type slot/port*[**details**]	Displays the web-based authentication settings for the specified interface for fastethernet, gigabitethernet, or tengigabitethernet. In Session Aware Networking mode, use the **show access-session interface** command.