Web-Based Authentication

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

Authentication Overview

Use the 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, authentication intercepts ingress HTTP packets from the host and sends an HTML login page to the users. The users enter their credentials, which the authentication feature sends to the authentication, authorization, and accounting (AAA) server for authentication.

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

If authentication fails, 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, authentication forwards a Login-Expired HTML page to the host, and the user is .

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 embedded wireless controller are used during the local web authentication.

- *Customized*—The customized web pages (Login, Success, Fail, and Expire) are downloaded onto the embedded wireless 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 embedded wireless 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 embedded wireless 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.

Note

- You can view the webauth parameter-map information using the **show running-config** command output.
- The wireless Web-Authentication feature does not support the bypass type.
- Change in web authentication parameter map redirect login URL does not occur until a AP rejoin happens. You must enable and disable the WLAN to apply the new URL redirection.

Note We recommend that you follow the Cisco guidelines to create a customized web authentication login page. If you have upgraded to the latest versions of Google Chrome or Mozilla Firefox browsers, ensure that your webauth bundle has the following line in the *login.html* file:

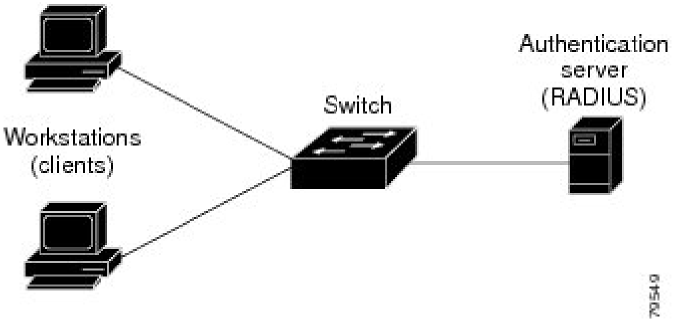
```
<body onload="loadAction();">
```

Device Roles

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

- *Client*—The device (workstation) that requests access to the network and the controller and responds to requests from the controller. 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 controller that the client is authorized to access the network and the controller services or that the client is denied.
- *Controller*—Controls the physical access to the network based on the authentication status of the client. The controller 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: Local Web Authentication Device Roles



Authentication Process

When you enable local web 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 authentication server is not available, after the web authentication retries, the client moves to the excluded state and the client receives an Authentication Server is Unavailabble page.
- 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.
- Web authentication sessions can not apply new VLAN as part of the authorization policy, as the client already has been assigned an IP address and you will not be able to change the IP address in the client, in case the VLAN changes.
- 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 the controller.

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*

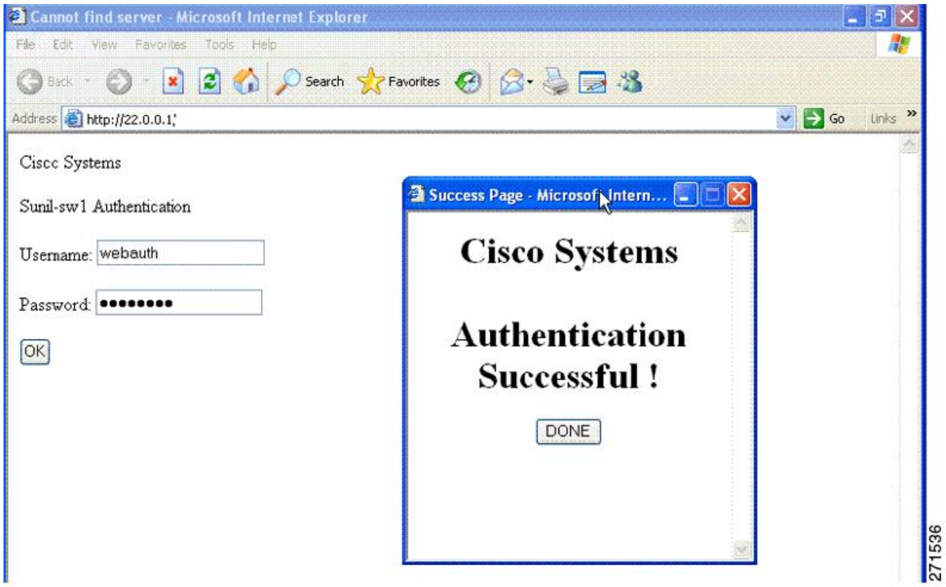
The Local Web Authentication Banner can be configured as follows:

- Use the following global configuration command:

```
Device(config)# parameter map type webauth global
Device(config-params-parameter-map)# banner ?
file <file-name>
text <Banner text>
title <Banner title>
```

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



Cannot find server - Microsoft Internet Explorer

Address http://22.0.0.1;

Cisco Systems

Sunil-sw1 Authentication

Username: webauth

Password: ••••••••

Success Page - Microsoft Intern...

Cisco Systems

Authentication Successful !

The banner can be customized as follows:

- Add a message, such as switch, router, or company name to the banner:
  - New-style mode—Use the following global configuration command:

    parameter-map type webauth global

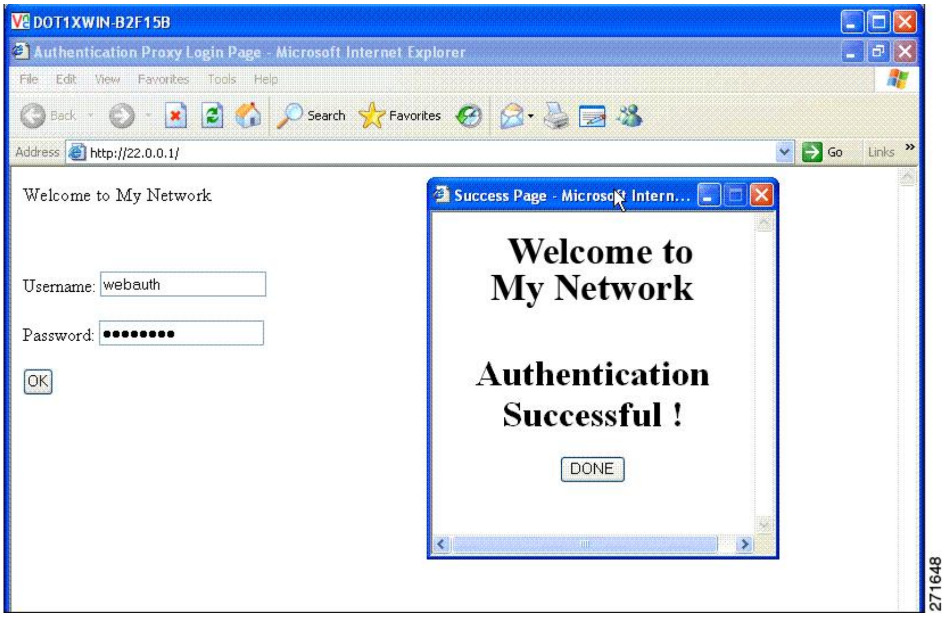
    banner text <text>

- Add a logo or text file to the banner:
  - New-style mode—Use the following global configuration command:

    parameter-map type webauth global

    banner file <filepath>

Figure 3: Customized Web Banner



DOT1XWIN-B2F15B

Authentication Proxy Login Page - Microsoft Internet Explorer

Address http://22.0.0.1/

Welcome to My Network

Username: webauth

Password: •••••••••

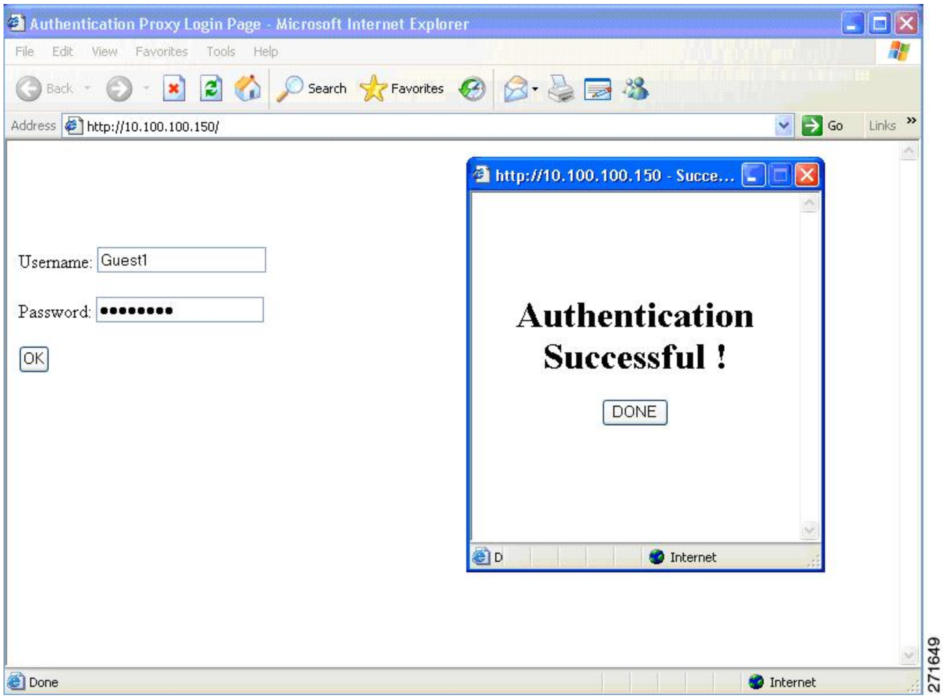
Success Page - Microsoft Intern...

Welcome to My Network

Authentication Successful !

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



Authentication Proxy Login Page - Microsoft Internet Explorer

Address http://10.100.100.150/

Username: Guest1

Password: •••••••••

http://10.100.100.150 - Succe...

Authentication Successful !

Customized Local Web Authentication

During the local web 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.

Note Virtual IP address is mandatory to configure custom web authentication.

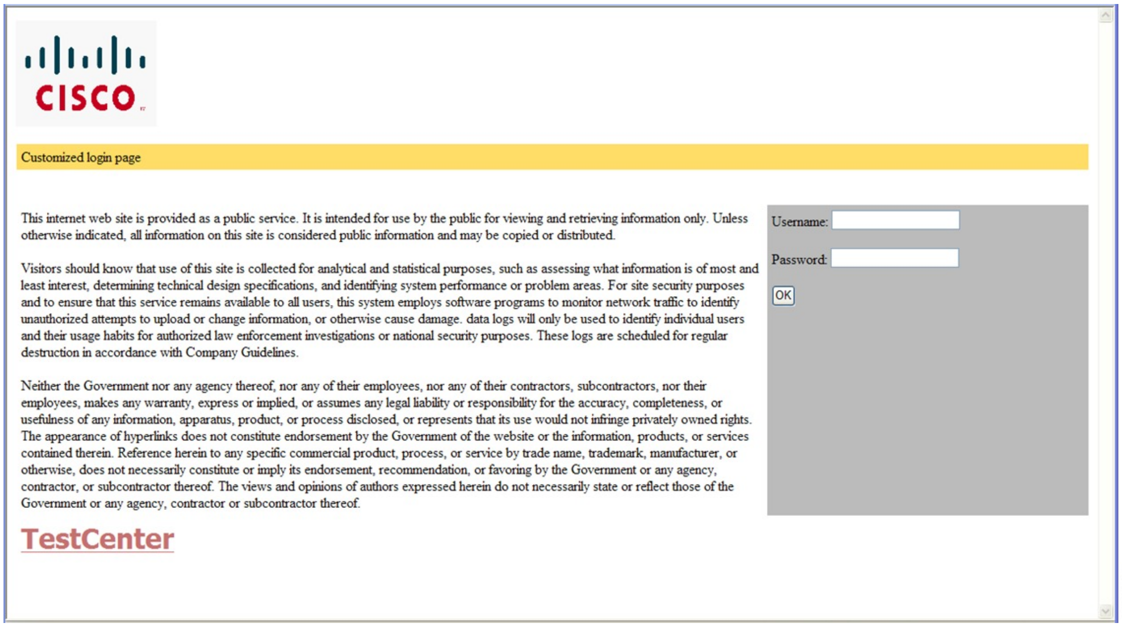
Guidelines

- 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.
- 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



Customized login page

This internet web site is provided as a public service. It is intended for use by the public for viewing and retrieving information only. Unless otherwise indicated, all information on this site is considered public information and may be copied or distributed.

Visitors should know that use of this site is collected for analytical and statistical purposes, such as assessing what information is of most and least interest, determining technical design specifications, and identifying system performance or problem areas. For site security purposes and to ensure that this service remains available to all users, this system employs software programs to monitor network traffic to identify unauthorized attempts to upload or change information, or otherwise cause damage. data logs will only be used to identify individual users and their usage habits for authorized law enforcement investigations or national security purposes. These logs are scheduled for regular destruction in accordance with Company Guidelines.

Neither the Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, nor their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. The appearance of hyperlinks does not constitute endorsement by the Government of the website or the information, products, or services contained therein. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the Government or any agency, contractor, or subcontractor thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the Government or any agency, contractor or subcontractor thereof.

Username:

Password:

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.

How to Configure Local Web Authentication

Configuring Default Local Web Authentication

The following table shows the default configurations required for local web authentication.

Table 1: Default Local Web Authentication Configuration

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

Configuring AAA Authentication (GUI)

Note The WebUI does not support the ipv6 radius source-interface under AAA radius server group configuration.

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA**.

Step 2 In the **Authentication** section, click **Add**.

Step 3 In the **Quick Setup: AAA Authentication** window that is displayed, enter a name for your method list.

Step 4 Choose the type of authentication you want to perform before allowing access to the network, in the **Type** drop-down list.

Step 5 Choose if you want to assign a group of servers as your access server, or if you want to use a local server to authenticate access, from the **Group** Type drop-down list.

Step 6 To configure a local server to act as a fallback method when servers in the group are unavailable, check the **Fallback** to local check box.

Step 7 Choose the server groups you want to use to authenticate access to your network, from the **Available Server Groups** list and click **>** icon to move them to the **Assigned Server Groups** list.

Step 8 Click **Save & Apply to Device**.

Configuring AAA Authentication (CLI)

Procedure

	Command or Action	Purpose
Step 1	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables AAA functionality.
Step 2	**aaa authentication login** {**default** | *named_authentication_list*} **group** *AAA_group_name* **Example:** `Device(config)# aaa authentication login default group group1`	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 3	**aaa authorization network** {**default** | **named**} **group** *AAA_group_name* **Example:** `Device(config)# aaa authorization network default group group1`	Creates an authorization method list for web-based authorization.

Step 4	**tacacs-server host** {*hostname* | *ip_address*}
	Example:
	`Device(config)# tacacs-server host 10.1.1.1`
		Specifies a AAA server.

Configuring the HTTP/HTTPS Server (GUI)

Procedure

Step 1 Choose **Administration** > **Management** > **HTTP/HTTPS/Netconf**.

Step 2 In the **HTTP/HTTPS Access Configuration** section, enable HTTP Access and enter the port that will listen for HTTP requests. The default port is 80. Valid values are 80, and ports between 1025 and 65535.

Step 3 Enable **HTTPS Access** on the device and enter the designated port to listen for HTTPS requests. The default port is 1025. Valid values are 443, and ports between 1025 and 65535. On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser.

Step 4 Choose the **Personal Identity Verification** as enabled or disabled.

Step 5 In the **HTTP Trust Point Configuration** section, enable **Enable Trust Point** to use Certificate Authority servers as trustpoints.

Step 6 From the **Trust Points** drop-down list, choose a trust point.

Step 7 In the **Timeout Policy Configuration** section, enter the HTTP timeout policy in seconds. Valid values can range from 1 to 600 seconds.

Step 8 Enter the number of minutes of inactivity allowed before the session times out. Valid values can range from 180 to 1200 seconds.

Step 9 Enter the server life time in seconds. Valid values can range from 1 to 86400 seconds.

Step 10 Enter the maximum number of requests the device can accept. Valid values range from 1 to 86400 requests.

Step 11 Save the configuration.

Configuring the HTTP Server (CLI)

To use local web 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	**configure terminal** **Example:** `Device# `**`Device# configure terminal`**	Enters global configuration mode.
Step 2	**ip http server** **Example:** `Device(config)# `**`ip http server`**	Enables the HTTP server. The local web authentication feature uses the HTTP server to communicate with the hosts for user authentication.
Step 3	**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.

	Command or Action	Purpose
		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 4	**end**
	Example:
	`Device(config)# end`
		Exits configuration mode.

Creating a Parameter Map (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 Click **Add**.

Step 3 Click **Policy Map**.

Step 4 Enter **Parameter Name**, **Maximum HTTP connections**, **Init-State Timeout(secs)** and choose **webauth** in the **Type** drop-down list.

Step 5 Click **Apply to Device**.

Configuring the Maximum Web Authentication Request Retries

Follow these steps to configure the maximum web authentication request retries:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**wireless security web-auth retries** *number* **Example:** `Device(config)# wireless security web-auth retries 2`	*number* is the maximum number of web auth request retries. The valid range is 0 to 20.

Step 4	**end**
	Example:
	`Device(config)# end`
		Returns to privileged EXEC mode.

Configuring a Local Banner in Web Authentication Page (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

Step 3 In the **General** tab and choose the required Banner Type:

- If you choose **Banner Text**, enter the required banner text to be displayed.
- If you choose **File Name**, specify the path of the file from which the banner text has to be picked up.

Step 4 Click **Update & Apply**.

Configuring a Local Banner in Web Authentication Page (CLI)

Follow the procedure given below to configure a local banner in web authentication pages.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *param-map* **Example:** `Device(config)# parameter-map type webauth` *`param-map`*	Configures the web authentication parameters. Enters the parameter map configuration mode.

	Command or Action	Purpose
Step 3	**banner** [*file* | *banner-text* | *title*] **Example:** `Device(config-params-parameter-map)# banner http C My Switch C`	Enables the local banner. Create a custom banner by entering *C banner-text C* (where *C* is a delimiting character), or *file* that indicates a file (for example, a logo or text file) that appears in the banner, or *title* that indicates the title of the banner.

Step 4	**end**
	Example:
	`Device(config-params-parameter-map)# end`
		Returns to privileged EXEC mode.

Configuring TrustPoint for Local Web Authentication

Before you begin

Ensure that a certificate is installed on your embedded wireless controller.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth global** **Example:** `Device (config)# parameter-map type webauth global`	Creates the parameter map.
Step 3	**trustpoint** *trustpoint-name* **Example:** `Device (config-params-parameter-map)# trustpoint trustpoint-name`	Configures trustpoint for local web authentication.

Step 4	**end**
	Example:
	`Device (config-params-parameter-map)# end`
		Returns to privileged EXEC mode.

Information About Management over Wireless

The management over wireless feature allows you to monitor and configure local embedded wireless controllers using a wireless client. You can perform all the management tasks except uploads to and downloads from (transfers to and from) the embedded wireless controller.

Restrictions on Management over Wireless

- Management over wireless can be disabled only if clients are on central switching.

Configuring Management over Wireless (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Wireless Global**.

Step 2 Check the **Management Via Wireless** check box to enable the feature.

Step 3 Click **Apply**.

Configuring Management over Wireless (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**[no] wireless mgmt-via-wireless** **Example:** `Device(config)# wireless mgmt-via-wireless`	Enables management access over wireless clients.
Step 3	**end**	Exits the global configuration mode and returns to the privileged EXEC mode.
Step 4	**show running-config | include mgmt-via-wireless** **Example:** `Device# show running-config | include mgmt-via-wireless`	Verifies the status of management access over wireless clients.

Configuration Examples for Local Web Authentication

Example: Obtaining Web Authentication Certificate

This example shows how to obtain web authentication certificate.

```
Device# configure terminal
Device(config)# crypto pki import cert pkcs12 tftp://9.1.0.100/ldapserver-cert.p12 cisco
Device(config)# end
Device# show crypto pki trustpoints cert
 Trustpoint cert:
    Subject Name:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
          Serial Number (hex): 00
    Certificate configured.
Device# show  crypto pki certificates cert
Certificate
  Status: Available
  Certificate Serial Number (hex): 04
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Subject:
    Name: ldapserver
    e=rkannajr@cisco.com
    cn=ldapserver
    ou=WNBU
    o=Cisco
    st=California
    c=US
  Validity Date:
    start date: 07:35:23 UTC Jan 31 2012
    end   date: 07:35:23 UTC Jan 28 2022
  Associated Trustpoints: cert ldap12
  Storage: nvram:rkannajrcisc#4.cer

CA Certificate
  Status: Available
  Certificate Serial Number (hex): 00
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Subject:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Validity Date:
    start date: 07:27:56 UTC Jan 31 2012
    end   date: 07:27:56 UTC Jan 28 2022
  Associated Trustpoints: cert ldap12 ldap
  Storage: nvram:rkannajrcisc#0CA.cer
```

Example: Displaying a Web Authentication Certificate

This example shows how to display a web authentication certificate.

```
Device# show crypto ca certificate verb
     Certificate
     Status: Available
     Version: 3
     Certificate Serial Number (hex): 2A9636AC00000000858B
     Certificate Usage: General Purpose
     Issuer:
    cn=Cisco Manufacturing CA
    o=Cisco Systems
    Subject:
    Name: WS-C3780-6DS-S-2037064C0E80
    Serial Number: PID:WS-C3780-6DS-S SN:FOC1534X12Q
    cn=WS-C3780-6DS-S-2037064C0E80
    serialNumber=PID:WS-C3780-6DS-S SN:FOC1534X12Q
    CRL Distribution Points:
    http://www.cisco.com/security/pki/crl/cmca.crl
    Validity Date:
    start date: 15:43:22 UTC Aug 21 2011
    end   date: 15:53:22 UTC Aug 21 2021
    Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (1024 bit)
    Signature Algorithm: SHA1 with RSA Encryption
    Fingerprint MD5: A310B856 A41565F1 1D9410B5 7284CB21
    Fingerprint SHA1: 04F180F6 CA1A67AF 9D7F561A 2BB397A1 0F5EB3C9
    X509v3 extensions:
    X509v3 Key Usage: F0000000
      Digital Signature
      Non Repudiation
      Key Encipherment
      Data Encipherment
    X509v3 Subject Key ID: B9EEB123 5A3764B4 5E9C54A7 46E6EECA 02D283F7
    X509v3 Authority Key ID: D0C52226 AB4F4660 ECAE0591 C7DC5AD1 B047F76C
    Authority Info Access:
    Associated Trustpoints: CISCO_IDEVID_SUDI
    Key Label: CISCO_IDEVID_SUDI
```

Example: Choosing the Default Web Authentication Login Page

This example shows how to choose a default web authentication login page.

```
Device# configure terminal
Device(config)# parameter-map type webauth test
This operation will permanently convert all relevant authentication commands to their CPL
control-policy equivalents. As this conversion is irreversible and will
disable the conversion CLI 'authentication display [legacy|new-style]', you are strongly
advised to back up your current configuration before proceeding.
Do you wish to continue? [yes]: yes
Device(config)# wlan wlan50
Device(config-wlan)# shutdown
Device(config-wlan)# security web-auth authentication-list test
Device(config-wlan)# security web-auth parameter-map test
Device(config-wlan)# no shutdown
Device(config-wlan)# end
Device# show running-config | section wlan50
wlan wlan50 50 wlan50
 security wpa akm cckm
 security wpa wpa1
 security wpa wpa1 ciphers aes
 security wpa wpa1 ciphers tkip
 security web-auth authentication-list test
 security web-auth parameter-map test
 session-timeout 1800
 no shutdown

Device# show running-config | section parameter-map type webauth test
parameter-map type webauth test
 type webauth
```

Example: Choosing a Customized Web Authentication Login Page from an IPv4 External Web Server

This example shows how to choose a customized web authentication login page from an IPv4 external web server.

```
Device# configure terminal
Device(config)# parameter-map type webauth global
Device(config-params-parameter-map)# virtual-ip ipv4 1.1.1.1
Device(config-params-parameter-map)# parameter-map type webauth test
Device(config-params-parameter-map)# type webauth
Device(config-params-parameter-map)# redirect for-login http://9.1.0.100/login.html
Device(config-params-parameter-map)# redirect portal ipv4 9.1.0.100
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv4 1.1.1.1
parameter-map type webauth test
type webauth
redirect for-login http://9.1.0.100/login.html
redirect portal ipv4 9.1.0.100
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: Choosing a Customized Web Authentication Login Page from an IPv6 External Web Server

This example shows how to choose a customized web authentication login page from an IPv6 external web server.

```
Device# configure terminal
Device(config)# parameter-map type webauth global
Device(config-params-parameter-map)# virtual-ip ipv6 1:1:1::1
Device(config-params-parameter-map)# parameter-map type webauth test
Device(config-params-parameter-map)# type webauth
Device(config-params-parameter-map)# redirect for-login http://9:1:1::100/login.html
Device(config-params-parameter-map)# redirect portal ipv6 9:1:1::100
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv6 1:1:1::1
parameter-map type webauth test
type webauth
redirect for-login http://9:1:1::100/login.html
redirect portal ipv6 9:1:1::100
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: Assigning Login, Login Failure, and Logout Pages per WLAN

This example shows how to assign login, login failure and logout pages per WLAN.

```
Device# configure terminal
Device(config)# parameter-map type webauth test
Device(config-params-parameter-map)# custom-page login device flash:loginsantosh.html
Device(config-params-parameter-map)# custom-page login expired device flash:loginexpire.html
Device(config-params-parameter-map)# custom-page failure device flash:loginfail.html
Device(config-params-parameter-map)# custom-page success device flash:loginsucess.html
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://9.1.0.100/login.html
 redirect portal ipv4 9.1.0.100
 custom-page login device flash:loginsantosh.html
 custom-page success device flash:loginsucess.html
 custom-page failure device flash:loginfail.html
 custom-page login expired device flash:loginexpire.html
```

Example: Configuring Preauthentication ACL

This example shows how to configure preauthentication ACL.

```
Device# configure terminal
Device(config)# wlan fff
Device(config-wlan)# shutdown
Device(config-wlan)# ip access-group web preauthrule
Device(config-wlan)# no shutdown
Device(config-wlan)# end
Device# show wlan name fff
```

Example: Configuring Webpassthrough

This example shows how to configure webpassthrough.

```
Device# configure terminal
Device(config)# parameter-map type webauth webparalocal
Device(config-params-parameter-map)# type consent
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://9.1.0.100/login.html
 redirect portal ipv4 9.1.0.100
```

Verifying Web Authentication Type

To verify the web authentication type, run the following command:

```
Device# show parameter-map type webauth all
Type Name
---------------------------------
Global global
Named webauth
Named ext
Named redirect
Named abc
Named glbal
Named ewa-2
wlc-tunga#sh parameter-map type webauth global
Parameter Map Name : global
Banner:
Text : CisCo
Type : webauth
Auth-proxy Init State time : 120 sec
Webauth max-http connection : 100
Webauth logout-window : Enabled
Webauth success-window : Enabled
Consent Email : Disabled
Sleeping-Client : Enabled
Sleeping-Client timeout : 60 min
Virtual-ipv4 : 1.1.1.1
Virtual-ipv4 hostname :
Webauth intercept https : Disabled
Webauth Captive Bypass : Disabled
Webauth bypass intercept ACL :
Trustpoint name :
HTTP Port : 80
Watch-list:
Enabled : no
Webauth login-auth-bypass:
```

```
Device# show parameter-map type webauth name global
Parameter Map Name : global
Type : webauth
Auth-proxy Init State time : 120 sec
Webauth max-http connection : 100
Webauth logout-window : Enabled
Webauth success-window : Enabled
```

```
Consent Email : Disabled
Sleeping-Client : Disabled
Webauth login-auth-bypass:
```