How does Web-Based Authentication Work?

How does Web-Based Authentication Work?

When you initiate an HTTP session, web 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 authentication sends a Login-Successful HTML page to the host and applies the access policies returned by the AAA server. If authentication fails, web 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, a Login-Expired HTML page is sent to the host, and the user is excluded with the exclusion reason as Web authentication failure.

Database for Authentication

The user accounts used for authentication can either be added locally on the WLC, on a RADIUS server or on an LDAP server. The choice of server to be used for authentication and the precedence can be configured on the WLC by the administrator.

- **Local database**: The controller stores the credentials (username and password) of all the local network users and these credentials are then used to authenticate the users.
- **LDAP database**: Lightweight Directory Access Protocol or LDAP stores the credentials of the users in the LDAP backend database. The controller queries the LDAP server for the credentials of a particular user in the database and these credentials are then used to authenticate the users.
- **RADIUS database**: Remote Authentication Dial-In User Service (RADIUS) serves as the backend database for storing the credentials of the users. The controller requests the RADIUS server for credentials that can be used to authenticate the users.

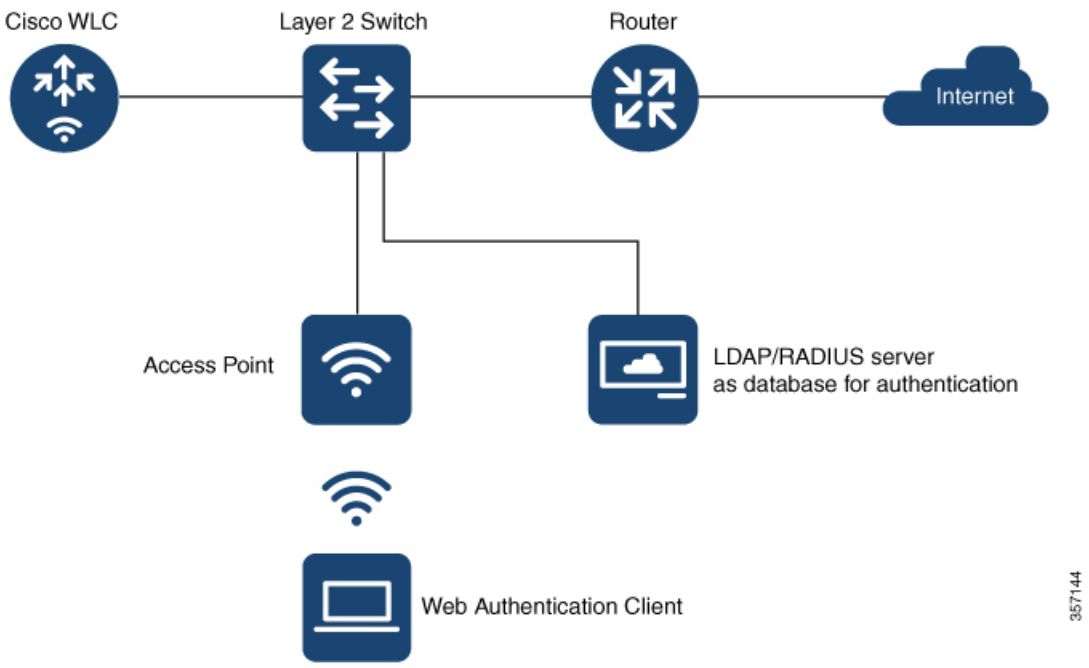
Based on where the web pages are hosted, web authentication can be categorized as follows:

- **Internal**: The internal default HTML pages (Login, Success, Fail, and Expire) in the controller are used during authentication.
- **Custom**: The customized web pages (Login, Success, Fail, and Expire) are downloaded onto the controller and used during authentication.

- **External**: The customized web pages are hosted on an external web server instead of using the in-built or customized web pages.

Devices and Roles in Local Web Authentication

Figure 1: Local Web Authentication Topology



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

- Client: Requests access to the WLAN and its services and responds to requests from the controller.
- Access Point: Connects directly to a switch through a wired Ethernet connection and provides wireless connection to client devices. It also restricts IP traffic except DHCP and DNS packets until the guest provides valid credentials.
- Controller: Manages the access points and clients. The WLC intercepts HTTP requests from the client and redirects the client to a login page for authentication. It authenticates the user after validating the credentials entered by the user against the local database or the external server. In addition, the controller hosts the virtual interface used for guest connectivity.
- 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 its services or that the client is denied access to the network.

Authentication Process

Figure 2: Process Flow Diagram for Local Web Authentication

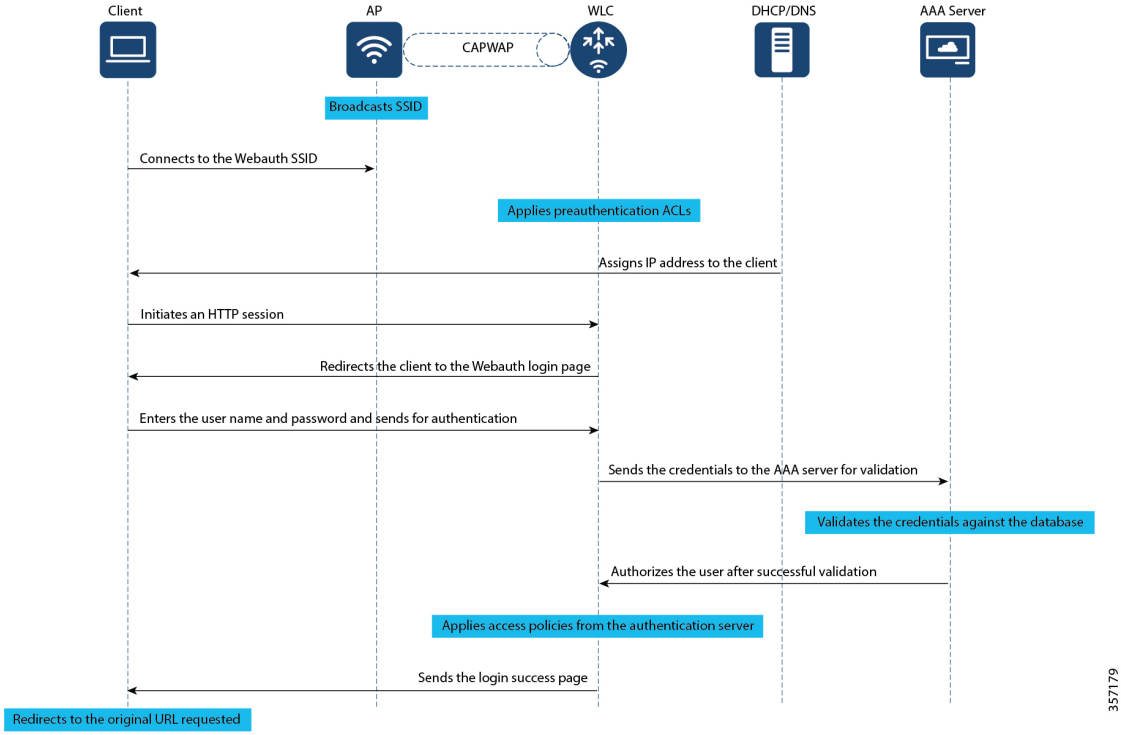
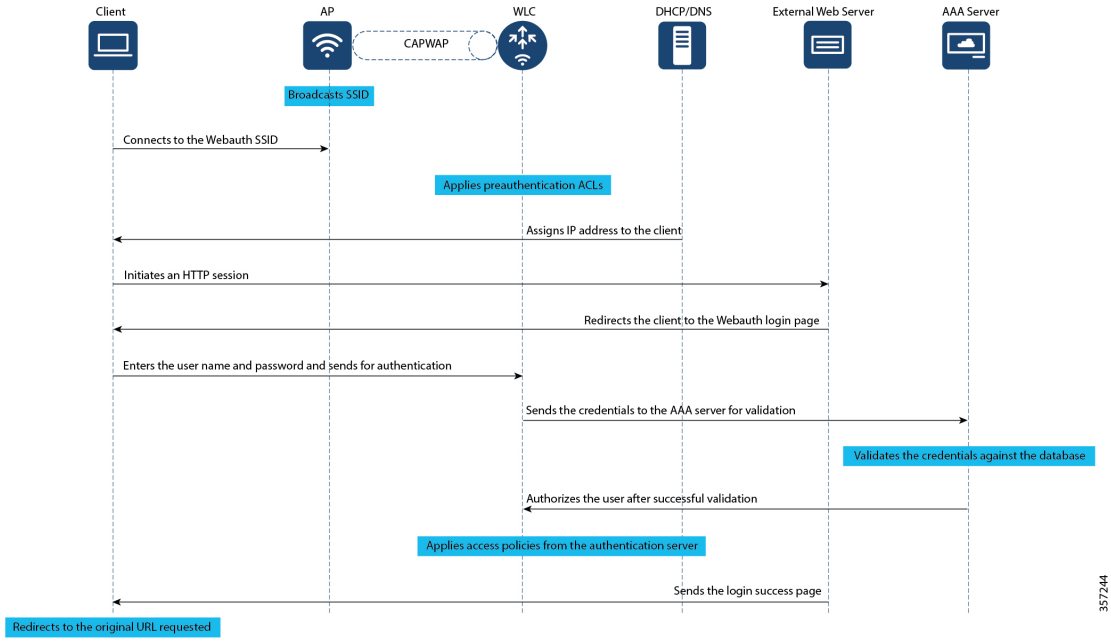


Figure 3: Process Flow Diagram for External Web Authentication



Web Authentication Types

Based on the various types of web authentication pages, web-based authentication is classified as follows:

- **Webauth**: This is the basic web authentication method where the controller presents a policy page with the user name and password. You need to enter the correct credentials to access the network.
- **Authbypass**: The controller uses the MAC address as the client identity and validates this with the authentication server that has a database of client MAC addresses that are allowed network access.
- **Consent (web-passthrough)**: The controller redirects you to a usage policy page without prompting you to enter any credentials for user authentication. To access the network, you must accept the policy.
- **Webconsent**: This is a combination of webauth and consent web authentication types in which the controller redirects you to a usage policy page with **Accept** or **Deny** buttons along with user name or password. You need to enter the correct credentials and accept the usage policy to access the network.

Web Authentication Features

Secure HTTP Access for Web Authentication

For local web authentication to work, you must enable HTTP access on the controller. By default, web authentication starts when the controller intercepts the first TCP HTTP GET packet from the client. When the client sends the first HTTP GET to TCP port 80, the controller redirects the client to https: <virtual IP>/login.html and displays the web authentication login page.

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. With secure authentication enabled, the login page always uses HTTPS even if the client sends an HTTP request.

The virtual interface IP address (IPv4 or IPv6) is used only in communications between the controller and wireless clients. It serves as the redirect address for the web authentication login page. It is recommended that you configure a nonroutable IP address for the virtual interface, ideally not overlapping with the network infrastructure addresses. Use one of the options proposed in RFC 5737, for example, 192.0.2.0/24, 198.51.100.0/24, and 203.0.113.0/24 networks.

WLC Certificate Validation

In a wireless network, each wireless device (controller, access point, and client) has its own authentication certificate that validates its identity. The WLC and access points are shipped with a Cisco-installed Certificate Authority (CA) certificate that is used to sign and validate device certificates. When the guest users try to access a URL, they receive a security warning because of the standard self-signed certificate that is installed on the WLC, by default. To avoid this warning, we recommend you install a third-party certificate signed by a trusted certificate authority on the controller.

To use a third-party certificate issued by a trusted CA, you must download the following certificates from the CA server to the controller and the clients.

- Device Certificate for the WLC and the clients
- Root Certificate of the Public Key Infrastructure (PKI) for the WLC
- CA Certificate for the clients

These digital certificates are configured and held in containers called trustpoints and used when the devices initiate a secure communication with the other network devices. A trustpoint includes the identity of the CA that signed the device certificate, CA-specific trustpoint configuration parameters, and an association with one, enrolled identity (device) certificate.

For more information about certificates, see Trustpoints in Cisco 9800 chapter.

Custom Webauth Login Portal

Web authentication allows customization of the login portal for user login. The administrator has an option to have the user view a Cisco default login page from the internal web server of the controller, a customized login page from the internal web server of the controller or a customized page residing on an external web server. The custom login pages allow you to create web pages that include your corporate logo, backgrounds, fonts, cascading style sheets, legalese and so on.

To create custom pages, you must download the webauth bundle from the software downloads page on cisco.com. The webauth bundle is a GNU tar file that contains HTML and GIF files. The main file in the web auth bundle is the login.html file which you can modify to create your custom web page. After you have created your HTML pages or modified the existing login.html file from the webauth bundle, you have to bundle the files in a GNU standard tar file and upload to the controller or to the local web server. We recommend you to customize a bundle that exists instead of creating a new bundle.

Custom Web Browser Banners

You can customize the web browser banners that appears when you log on to a switch, by modifying the title and the body of the default banner. The banner appears on both the login page and the authentication result pop-up page.

The banner title, **Welcome to the Cisco Web-Authentication network** and a body text appears by default when you are redirected to the login page. However, you can customize the title and the body text using the CLI or the Web user interface.

Cisco Systems and one of the following authentication messages appear on the authentication result pop-up page.

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

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 in to the switch.

Custom Authentication Proxy Web Pages

During local web authentication process, the internal HTTP server on the controller hosts four HTML pages to deliver to an authenticating client. The server uses these pages to notify the client of the four authentication process states. Custom Authentication Proxy Web Pages lets you display four user-defined HTML pages to users in place of the switch's internal default HTML pages during web-based authentication. You can also specify a URL to which users are redirected after authentication occurs, which replaces the internal Success page.

- 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.

The configured authentication proxy web pages supports both HTTP and SSL.

Preauthentication Access Control Lists and URLFilters

To determine the data requests that are allowed or blocked for a guest user, you need to configure preauthentication ACLs on the controller. You need to configure the allowed URLs or denied URLs for the ACLs. The URLs need to be pre-configured on the ACL. With Preauthentication ACLs configured, the client when in registration phase is allowed to connect to the configured URLs.

When using an external web server for web authentication, you must configure a pre-authentication ACL for permitting the clients to access the external web server. For local web authentication, configuring pre-authentication ACL is not mandatory. However, it is a good practice to configure a pre-authentication ACL if you want to give the client access to any non-HTTP resources before authentication.

Configuring URLfilter lists helps you to add specific URLs to the allowed list on the controller or the AP. Authentication is not required to access the allowed list of URLs. When you try to access sites that are not in allowed list, you are redirected to the Login page.