5

Wireless (for ISA550W and ISA570W only)

This chapter describes how to configure your wireless network. It includes the following sections:

To access the Wireless pages, click **Wireless** in the left hand navigation pane.

Viewing Wireless Status

This section describes how to view information for your wireless network. Refer to the following topics:

- Viewing Wireless Statistics, page 207
- Viewing Wireless Client Status, page 208

Viewing Wireless Statistics

Use the Wireless Status page to view the cumulative total of relevant wireless statistics for all SSIDs. This page is automatically updated every 10 seconds. Click Refresh to manually refresh the data.

Wireless > Wireless Status > Wireless Status

Field	Description
Wireless Status	
SSID Number	Number of the SSID.
SSID Name	Name of the SSID.
MAC Address	MAC address of the SSID.
VLAN	VLAN to which the SSID is mapped.
Client List	Number of client stations that are connected to the SSID.
Wireless Statistics	
Name	Name of the SSID.
Tx Packets	Number of transmitted packets on the SSID.
Rx Packets	Number of received packets on the SSID.
Collisions	Number of packet collisions reported to the SSID.
Tx Bytes/Sec	Number of transmitted bytes of information on the SSID.
Rx Bytes/Sec	Number of received bytes of information on the SSID.
Uptime	Time that the SSID has been active.

Viewing Wireless Client Status

Use the Client Status page to view information for all client stations that are already connected to each SSID. The MAC address and IP address for all connected client stations for each SSID are displayed. To open this page, click **Wireless > Wireless Status > Client Status**. This page is automatically updated every 10 seconds. Click **Refresh** to manually refresh the data.

Configuring the Basic Settings

Use the Basic Settings page to change the wireless mode to suit the devices in your network, specify the wireless channel and bandwidth for operation to resolve issues with interference from other access points in the area, or enable U-APSD and SSID Isolation if needed.

STEP 1 Click **Wireless > Basic Settings**.

STEP 2 Enter the following information:

- **Wireless Radio:** Click **On** to turn wireless radio on and hence enable the SSID called “cisco-data,” or click **Off** to turn wireless radio off. Enabling any SSID will turn on wireless radio. Disabling all SSIDs will turn off wireless radio.
- **Wireless Mode:** Choose the 802.11 modulation technique.
  - **802.11b/g mixed:** Choose this mode if some devices in the wireless network use 802.11b and others use 802.11g. Both 802.11b and 802.11g clients can connect to the access point.
  - **802.11b/g/n mixed:** Choose this mode to allow 802.11b, 802.11g, and 802.11n clients operating in the 2.4 GHz frequency to connect to the access point.
  - **802.11n only:** Choose this mode if all devices in the wireless network can support 802.11n. Only 802.11n clients operating in the 2.4 GHz frequency can connect to the access point.
- **Wireless Channel:** Choose a channel from a list of channels or choose **Auto** to let the system determine the optimal channel to use based on the environmental noise levels for the available channels.

- **Bandwidth Channel:** Choose 20 MHz channel bonding (spacing), or choose **Auto** to let the system determine the optimal channel spacing to use. This setting is specific to 802.11n traffic.
- **Extension Channel:** Choose either Lower or Upper if you choose Auto channel spacing.
- **Unscheduled Automatic Power Save Delivery (U-APSD):** Click **Enable** to enable U-APSD to conserve the power, or click **Disable** to disable it.
- **SSID Isolation:** Click **Enable** to enable the SSID Isolation feature so that the SSIDs will be unable to see each other when the SSIDs belong to the same VLAN, or click **Disable** to disable it. When you enable SSID Isolation (among the SSIDs), traffic on one SSID will not be forwarded to any other SSID.

STEP 3 In the **SSIDs** area, all predefined SSIDs on the security appliance appear in the table. You can configure the following properties for each predefined SSID:

- **Enable:** Check this box to enable a SSID, uncheck this box to disable a SSID. By default, all SSIDs are disabled.
- **SSID Name:** Enter the name for a SSID.
- **SSID Broadcast:** Check this box to broadcast the SSID in its beacon frames. All wireless devices within range are able to see the SSID when they scan for available networks. Uncheck this box to prevent auto-detection of the SSID. In this case, users must know the SSID to set up a wireless connection to this SSID.

  NOTE: Disabling SSID Broadcast is sufficient to prevent clients from accidentally connecting to your network, but it will not prevent even the simplest of attempts by a hacker to connect or monitor unencrypted traffic. Suppressing the SSID broadcast offers a very minimal level of protection on an otherwise exposed network (such as a guest network) where the priority is making it easy for clients to get a connection and where no sensitive information is available.
- **Security Mode:** Displays the security mode currently used for the SSID. To configure the security settings for the SSID, click the **Edit** (pencil) icon. See **Configuring Wireless Security, page 211**.
- **MAC Filtering:** Shows if the MAC Filtering feature is enabled or disabled on the SSID. MAC Filtering can permit or block access to the SSID by the MAC (hardware) address of the requesting device. To configure the MAC Filtering settings for the SSID, click the **Edit** (pencil) icon. See **Controlling Wireless Access Based on MAC Addresses, page 217**.

- **VLAN Mapping:** Displays the VLAN to which the SSID is mapped. All traffic from the wireless clients that are connected to this SSID will be directed to the selected VLAN. To associate the SSID to a specific VLAN, click the **Edit** (pencil) icon. See **Mapping the SSID to VLAN, page 218**.
- **Wi-Fi Multimedia:** Check this box to enable Wi-Fi Multimedia (WMM), which is designed to improve the user experience for audio, video, and voice applications over a Wi-Fi wireless connection. WMM refers to QoS over Wi-Fi. QoS enables Wi-Fi access points to prioritize traffic and optimizes the way shared network resources are allocated among different applications. By default, WMM is enabled when you choose a wireless mode that includes 802.11n.
- **Station Isolation:** Check so that the wireless clients on the same SSID will be unable to see each other.

STEP 4 Click **Save** to apply your settings.

Configuring SSID Profiles

ISA550W and ISA570W support four SSIDs. By default, all SSIDs are disabled. For security purposes, we strongly recommend that you configure each SSID with the highest level of security that is supported by the devices into your wireless network.

Multiple SSIDs can segment the wireless LAN into multiple broadcast domains. This configuration helps you to maintain better control over broadcast and multicast traffic, which affects network performance.

Refer to the following topics:

- **Configuring Wireless Security, page 211**
- **Controlling Wireless Access Based on MAC Addresses, page 217**
- **Mapping the SSID to VLAN, page 218**
- **Configuring SSID Schedule, page 218**

Configuring Wireless Security

This section describes how to configure the security mode for the SSID. All devices on this network must use the same security mode and settings to work correctly. Cisco recommends using the highest level of security that is supported by the devices in your network.

NOTE If the security mode is set as WEP or as WPA with TKIP encryption algorithm for the SSID that supports 802.11n, the transmit rate for its associated client stations will not exceed 54 Mbps.

STEP 1 Click **Wireless > Basic Settings**.

STEP 2 In the **SSIDs** area, click the **Edit** (pencil) icon to edit the settings for the SSID.

The SSID - Edit window opens.

STEP 3 In the **Security Mode** tab, specify the following information:

- **SSID Name:** The name of the SSID on which the security settings are applied.
- **User Limit:** Specify the maximum number of users that can simultaneously connect to this SSID. Enter a value in the range 0 to 200. The value of zero (0) indicates that there is no limit for this SSID.

  NOTE: The maximum number of users that can simultaneously connect to all enabled SSIDs is 200.
- **Security Mode:** Choose the type of security.

Security Mode	Description
Open	Any wireless device that is in range can connect to the SSID. This is the default setting but not recommended.

Security Mode	Description
WEP	Wired Equivalent Privacy (WEP) is a data encryption protocol for 802.11 wireless networks. All wireless stations and SSIDs on the network are configured with a static 64-bit or 128-bit Shared Key for data encryption. The higher the bit for data encryption, the more secure for your network. WEP encryption is an older encryption method that is not considered to be secure and can easily be broken. Choose this option only if you need to allow access to devices that do not support WPA or WPA2.
WPA	Wi-Fi Protected Access (WPA) provides better security than WEP because it uses dynamic key encryption. This standard was implemented as an intermediate measure to replace WEP, pending final completion of the 802.11i standard for WPA2. The security appliance supports the following WPA security modes. Choose one of them if you need to allow access to devices that do not support WPA2. ▪ **WPA-Personal:** Supports TKIP (Temporal Key Integrity Protocol) or AES (Advanced Encryption System) encryption mechanisms for data encryption (default is TKIP). TKIP uses dynamic keys and incorporates Message Integrity Code (MIC) to provide protection against hackers. AES uses symmetric 128-bit block data encryption. ▪ **WPA-Enterprise:** Uses WPA with RADIUS authentication. This mode supports TKIP and AES encryption mechanisms (default is TKIP) and requires the use of a RADIUS server to authenticate users.

Security Mode	Description
WPA2	WPA2 provides the best security for wireless transmissions. This method implements the security standards specified in the final version of 802.11i. The security appliance supports the following WPA2 security modes: ▪ **WPA2-Personal:** Always uses AES encryption mechanism for data encryption. ▪ **WPA2-Enterprise:** Uses WPA2 with RADIUS authentication. This mode always uses AES encryption mechanism for data encryption and requires the use of a RADIUS server to authenticate users.
WPA + WPA2	Allows both WPA and WPA2 clients to connect simultaneously. The SSID automatically chooses the encryption algorithm used by each client device. This security mode is a good choice to enable a higher level of security while allowing access by devices that might not support WPA2. The security appliance supports the following WPA+WPA2 security modes: ▪ **WPA/WPA2-Personal mixed:** Supports the transition from WPA-Personal to WPA2-Personal. You can have client devices that use either WPA-Personal or WPA2-Personal. ▪ **WPA/WPA2-Enterprise mixed:** Supports the transition from WPA-Enterprise to WPA2-Enterprise. You can have client devices that use either WPA-Enterprise or WPA2-Enterprise.
RADIUS	Uses RADIUS servers for client authentication and dynamic WEP key generation for data encryption.

STEP 4 If you choose **Open** as the security mode, no other options are configurable. This mode means that any data transferred to and from the SSID is not encrypted. This security mode can be useful during initial network configuration or for problem solving, but it is not recommended for regular use on the internal network because it is not secure.

STEP 5 If you choose **WEP** as the security mode, enter the following information:

- **Authentication Type:** Choose either **Open System** or **Shared key**, or choose **Auto** to let the security appliance accept both Open System and Shared Key schemes.
- **Default Transmit Key:** Choose a key index as the default transmit key. Key indexes 1 through 4 are available.
- **Encryption:** Choose the encryption type: 64 bits (10 hex digits), 64 bits (5 ASCII), 128 bits (26 hex digits), or 128 bits (13 ASCII). The default is 64 bits (10 hex digits). The larger size keys provide stronger encryption, thus making the key more difficult to crack.
- **Passphrase:** If you want to generate WEP keys by using a Passphrase, enter any alphanumeric phrase (between 4 to 63 characters) and then click **Generate** to generate 4 unique WEP keys. Select one key to use as the key that devices must have to use the wireless network.
- **Key 1-4:** If a WEP Passphrase is not specified, a key can be entered directly into one of the Key boxes. The length of the key should be 5 ASCII characters (or 10 hex characters) for 64-bit encryption and 13 ASCII characters (or 26 hex characters) for 128-bit encryption.

STEP 6 If you choose **WPA-Personal** as the security mode, enter the following information:

- **Encryption:** Choose either TKIP or TKIP_CCMP (AES) as the encryption algorithm for data encryption. The default is TKIP.
- **Shared Secret:** The Pre-shared Key (PSK) is the shared secret key for WPA. Enter a string of at least 8 characters to a maximum of 63 characters.
- **Key Renewal Timeout:** Enter a value to set the interval at which the key is refreshed for clients associated to this SSID. The valid range is 0 to 4194303 seconds. A value of zero (0) indicates that the key is not refreshed. The default value is 3600 seconds.

STEP 7 If you choose **WPA2-Personal** as the security mode, enter the following information:

- **Encryption:** Always use AES for data encryption.
- **Shared Secret:** The Pre-shared Key (PSK) is the shared secret key for WPA. Enter a string of at least 8 characters to a maximum of 63 characters.

- **Key Renewal Timeout:** Enter a value to set the interval at which the key is refreshed for clients associated to this SSID. The valid range is 0 to 4194303 seconds. A value of zero (0) indicates that the key is not refreshed. The default value is 3600 seconds.

STEP 8 If you choose **WPA/WPA2-Personal mixed** as the security mode, enter the following information:

- **Encryption:** Automatically choose TKIP or AES for data encryption.
- **Shared Secret:** The Pre-shared Key (PSK) is the shared secret key for WPA. Enter a string of at least 8 characters to a maximum of 63 characters.
- **Key Renewal Timeout:** Enter a value to set the interval at which the key is refreshed for clients associated to this SSID. The valid range is 0 to 4194303 seconds. A value of zero (0) indicates that the key is not refreshed. The default value is 3600 seconds.

STEP 9 If you choose **WPA-Enterprise** as the security mode, enter the following information:

- **Encryption:** Choose either TKIP or AES as the encryption algorithm for data encryption. The default is TKIP.
- **Key Renewal Timeout:** Enter a value to set the interval at which the key is refreshed for clients associated to this SSID. The valid range is 0 to 4194303 seconds. A value of zero (0) indicates that the key is not refreshed. The default value is 3600 seconds.
- **RADIUS Server ID:** The security appliance predefines three RADIUS groups. Choose an existing RADIUS group for client authentication. The following RADIUS server settings of the selected group are displayed.
  - **Primary RADIUS Server IP Address:** The IP address of the primary RADIUS server.
  - **Primary RADIUS Server Port:** The port number of the primary RADIUS server.
  - **Primary RADIUS Server Shared Secret:** The shared secret key of the primary RADIUS server.
  - **Secondary RADIUS Server IP Address:** The IP address of the secondary RADIUS server.
  - **Secondary RADIUS Server Port:** The port number of the secondary RADIUS server.

- **Secondary RADIUS Server Shared Secret:** The shared secret key of the secondary RADIUS server.

  NOTE: You can change the settings in the above fields but the RADIUS server settings you specify will replace the default settings of the selected group. To maintain the RADIUS servers, go to the Users > RADIUS Servers page. See Configuring RADIUS Servers, page 401.

STEP 10 If you choose **WPA2-Enterprise** as the security mode, enter the following information:

- **Encryption:** Always use AES encryption algorithm for data encryption.
- **Key Renewal Timeout:** Enter a value to set the interval at which the key is refreshed for clients associated to this SSID. The valid range is 0 to 4194303 seconds. A value of zero (0) indicates that the key is not refreshed. The default value is 3600 seconds.
- **RADIUS Server ID:** Choose an existing RADIUS group for client authentication. The RADIUS server settings of the selected group are displayed. You can change the RADIUS server settings but the settings you specify will replace the default settings of the selected group. To maintain the RADIUS servers, go to the Users > RADIUS Servers page. See Configuring RADIUS Servers, page 401.

STEP 11 If you choose **WPA/WPA2-Enterprise Mixed** as the security mode, enter the following information:

- **Encryption:** Automatically choose TKIP or AES encryption algorithm for data encryption.
- **Key Renewal Timeout:** Enter a value to set the interval at which the key is refreshed for clients associated to this SSID. The valid range is 0 to 4194303 seconds. A value of zero (0) indicates that the key is not refreshed. The default value is 3600 seconds.
- **RADIUS Server ID:** Choose an existing RADIUS group for client authentication. The RADIUS server settings of the selected group are displayed. You can change the RADIUS server settings but the settings you specify will replace the default settings of the selected group. To maintain the RADIUS servers, go to the Users > RADIUS Servers page. See Configuring RADIUS Servers, page 401.

STEP 12 If you choose **RADIUS** as the security mode, choose an existing RADIUS group for client authentication from the **RADIUS Server ID** drop-down list. The RADIUS server settings of the selected group are displayed. You can change the RADIUS server settings but the settings you specify will replace the default settings of the

selected group. To maintain the RADIUS servers, go to the Users > RADIUS Servers page. See Configuring RADIUS Servers, page 401.

STEP 13 Click **OK** to save your settings.

STEP 14 Click **Save** to apply your settings.

Controlling Wireless Access Based on MAC Addresses

MAC Filtering allows or blocks access to the SSID by the MAC (hardware) address of the requesting device. By default, MAC Filtering is disabled for each SSID.

MAC Filtering provides additional security, but it also adds to the complexity and maintenance. You need to specify the list of MAC addresses that you want to block or allow. Be sure to enter each MAC address correctly to ensure that the policy is applied as intended. Generally it is easier and more secure to use this feature to allow access to the specified MAC addresses, thereby denying access to unknown MAC addresses.

STEP 1 Click **Wireless > Basic Settings**.

STEP 2 In the **SSIDs** area, click the **Edit** (pencil) icon to edit the settings for the SSID.

The SSID - Edit window opens.

STEP 3 In the **MAC Filtering** tab, enter the following information:

- **SSID Name:** The name of the SSID on which the MAC Filtering settings are applied.
- **Connection Control:** Choose one of the following options as the MAC Filtering policy:
  - **Disable:** Disable MAC Filtering for the SSID.
  - **Allow only the following MAC addresses to connect to the wireless network:** All devices in list of MAC addresses are allowed to connect to this SSID. All other devices are blocked.
  - **Prevent the following MAC addresses from connecting to the wireless network:** All devices in list of MAC addresses are prevented from connecting to this SSID. All other devices are allowed.

STEP 4 In the **Connection Control List** area, specify the list of MAC addresses that you want to block or allow. You can add up to 16 MAC addresses.

STEP 5 Click **OK** to save your settings.

STEP 6 Click **Save** to apply your settings.

Mapping the SSID to VLAN

STEP 1 Click **Wireless > Basic Settings**.

STEP 2 In the **SSIDs** area, click the **Edit** (pencil) icon to edit the settings for the SSID.

The SSID - Edit window opens.

STEP 3 In the **VLANs** tab, enter the following information:

- **SSID Name:** The name of the SSID to which the VLAN is mapped.
- **VLAN:** Choose the VLAN from the drop-down list. The SSID is mapped to the selected VLAN. All traffic from the wireless clients that are connected to this SSID will be directed to the selected VLAN.

STEP 4 Click **OK** to save your settings.

STEP 5 Click **Save** to apply your settings.

Configuring SSID Schedule

This section describes how to specify the schedule to keep the SSID active within a specific time per day.

STEP 1 Click **Wireless > Basic Settings**.

STEP 2 In the **SSIDs** area, click the **Edit** (pencil) icon to edit the settings for the SSID.

The SSID - Edit window opens.

STEP 3 In the **Scheduling** tab, specify the time per day to keep the SSID active:

- **SSID Name:** The name of the SSID on which the schedule setting is applied.

- **Active Time:** Click **On** to enable the schedule feature for the SSID, or click **Off** to disable it. Disabling the schedule feature will keep the SSID active in 24 hours per day. If you enable this feature, configure the time range per day to keep this SSID active.
  - **Start Time:** Enter the values in the hour and minute fields and choose AM or PM from the drop-down list.
  - **Stop Time:** Enter the values in the hour and minute fields and choose AM or PM from the drop-down list.

STEP 4 Click **OK** to save your settings.

STEP 5 Click **Save** to apply your settings.

Configuring Wi-Fi Protected Setup

Use the Wi-Fi Protected Setup page to configure Wi-Fi Protected Setup (WPS) on the security appliance to allow WPS-enabled devices to more easily connect to the wireless network.

STEP 1 Click **Wireless > Wi-Fi Protected Setup**.

The Wi-Fi Protected Setup window opens.

STEP 2 Click **On** to enable WPS, or click **Off** to disable it.

STEP 3 If you enable WPS, specify the following WPS settings:

- **WPS Configuration Status:** Determines whether to start a new configuration on the SSID before the wireless client establishes a WPS connection.
  - **Configured:** If you choose this option, the wireless clients will associate with the SSID by following the original security settings of the SSID, which may cause an un-secured connection if the SSID is not configured properly in advance, for example the security mode is set to “Open.” To provide a secured connection under the Configured status, you can manually change the security mode for the SSID in advance and then establish the WPS connection.

- **Unconfigured:** If you choose this option, the SSID will automatically configure its security settings such as the SSID name and the security mode before the wireless clients are associated to provide a secured connection. After the wireless clients are connected, the status will be automatically changed to "Configured." Any change for the SSID name, the security mode, or the WEP key or passphrase will change the status to "Configured."

- **Network Name (SSID):** Choose the SSID on which the WPS settings are applied.
- **Security:** The security mode currently used for the selected SSID.
- **Encryption:** The encryption method currently used for the selected SSID.

STEP 4 If the wireless client device has a WPS push button, follow these steps to establish the WPS connection:

a. Enable WPS on the security appliance.

b. Click the **WPS** button on this page.

c. Press the **WPS** push button on the wireless client device within 2 minutes.

d. Verify that the wireless client is connected to the SSID.

STEP 5 If the wireless client device has a WPS PIN number, follow these steps to establish the WPS connection:

a. Get the PIN number used on the wireless client device.

b. Enable WPS on the security appliance.

c. Enter the PIN number in the field and click **Apply** to register the PIN number.

d. Enable WPS on the wireless client device within 2 minutes.

e. Verify that the wireless client is connected to the SSID.

STEP 6 If the wireless client device asks for the PIN number of the security appliance, follow these steps to establish the WPS connection:

a. Enable WPS on the security appliance.

b. Click **Generate** to generate a PIN number.

c. Follow the instructions on the wireless client device to configure WPS within 2 minutes by using the registered PIN number.

d. Verify that the wireless client device is connected to the SSID.

NOTE: If the wireless client device does not connect to the SSID after 2 minutes, please manually disable WPS on the security appliance to prevent the WPS brute-force attack.

STEP 7 Click **Save** to apply your settings.

Configuring Captive Portal

You may want to direct users to a web portal before they can access the Internet through the security appliance. To achieve this goal, you can enable Captive Portal on a wireless network, a VLAN, or a DMZ.

When a user in a Captive Portal user group attempts to access the Internet via a web browser, a portal page appears. You can require a log in or the entry of payment information, for example, and you can set up the portal page to display information, usage guidelines, warning messages, and so on. After successfully logging in, paying, or acknowledging your messages, the user can use other applications on the PC to communicate with the network.

In addition to the portal options mentioned above, additional options make it easy to adapt the Captive Portal feature to your needs:

- You can specify certain domains that users can access without going through the portal.
- The portal page can be stored locally on the ISA500 device or on an external web server that you specify.

Requirements

This feature is compatible with these browsers:

- Internet Explorer (v 8.0 or above)
- Firefox (v 9.0 or above)
- Google Chrome
- Safari

A computer accessing the Captive Portal must have one of these operating systems:

- Windows 7
- Windows XP
- Mac OS

Captive Portal also can be used from a mobile device with one of these operating systems:

- iOS (iPhone, iPad)
- Android

Before You Begin

Before you configure your portal, you may need to configure VLANs, SSIDs, and users. Read the following information to determine what steps may be needed to achieve your goals.

VLAN Setup

No special VLAN configuration is required for a Captive Portal, but you may want to consider the points below before proceeding. To configure VLANs, use the Networking > VLAN page..

- Each SSID is associated with a VLAN. You can use the pre-configured VLANs (DEFAULT, GUEST, and VOICE) or add a custom VLAN.

- You may want to associate a VLAN, such as the GUEST VLAN, with a security zone so that you can configure appropriate security policies. For example, you can apply URL filtering policies to the zone to prevent access to certain types of websites.
- A Captive Portal must be associated either with a single SSID or with a VLAN. If you want to enable a portal for users of multiple SSIDs, you will need to assign them all to the same VLAN. You can use a pre-configured VLAN or can create a VLAN for this purpose.

Wireless Setup

For a Captive Portal on the wireless network, you must enable the wireless radio and at least one SSID before you can enable a Captive Portal. To configure these settings, use the Wireless > Basic Settings page. .

- Enable the wireless radio.
- Enable the SSID(s) that you want to use for the portal.
- If you created a special VLAN for use with your Captive Portal, assign it to the SSID(s) that you want to use for the portal.

User Authentication

If you want to require user authentication for your portal, the security appliance can authenticate the users by using the local database and an external AAA server (such as RADIUS, AD, and LDAP). The authentication method is derived from the user authentication settings that you specified in the Users > User Authentication page. See Configuring User Authentication Settings, page 393.

For the local database option, you need to set up a User Group with the Captive Portal service enabled, and add the users' names and passwords. .

Configuring a Captive Portal

You configure this feature separately for the wireless network (**Wireless > Captive Portal**) and for the wired network (**Networking > Captive Portal**).

STEP 1 **Enable Captive Portal:** Click **On** to enable the Captive Portal feature.

STEP 2 **Apply On:** Choose the SSID, VLAN, or DMZ interface on which to apply the Captive Portal settings.

STEP 3 **Web Authentication Type:** Choose one of the following methods for web authentication. The security appliance can authenticate the users by using the local database and external AAA server (such as RADIUS, AD, and LDAP). The authentication method is derived from the user authentication settings that you specified in the Users > User Authentication page.

- **Internal:** Uses the default HotSpot Login page and requires a login.
- **Internal, no auth with accept button:** Uses the default HotSpot Login page and does not require a login. A user simply clicks the **Accept** button to access the Internet.
- **External:** Uses a custom HotSpot Login page on the specified external web server and requires a login.
- **External, no auth with accept button:** Uses a custom HotSpot Login page on the specified external web server and does not require a login. A user simply clicks the **Accept** button to access the Internet.

Note: If you chose Internal or External, you will need to use the Users > Users and Groups page to create a User Group with Captive Portal service enabled, and to add users to the group.

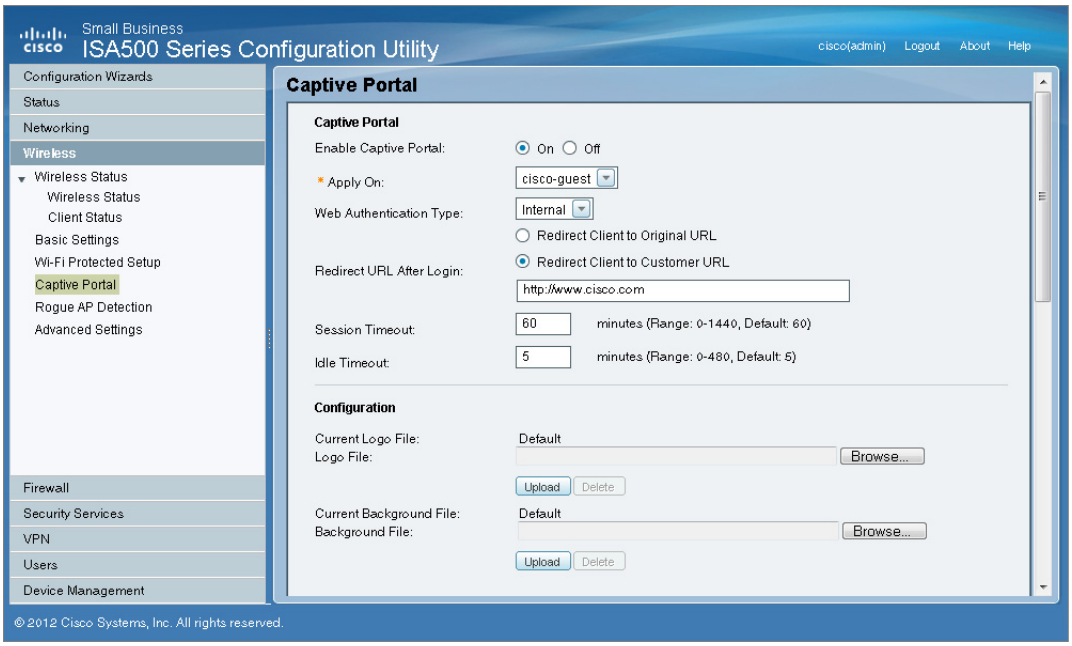
STEP 4 **Redirected URL After Login:** Choose one of the following options to determine what happens after a user leaves the portal page:

- **Redirect Client to Customer URL:** Directs the users to a particular URL (such as the URL for your company). If you choose this option, enter the desired URL in the field, including http:// or https://.
- **Redirect Client to Original URL:** Directs the users to the URL that they were trying to access originally.

STEP 5 Configure the timeout settings, or keep the default values.

- **Session Timeout:** Enter the maximum number of minutes that a wireless session can remain connected. After the timeout period elapses, the session will be terminated. Enter 0 to allow a user to remain connected without any limit. The default value is 60 minutes.

- **Idle Timeout:** Enter the maximum number of minutes that a wireless session can be idle. After the timeout period elapses, an idle session will be terminated. The default value is 5 minutes.

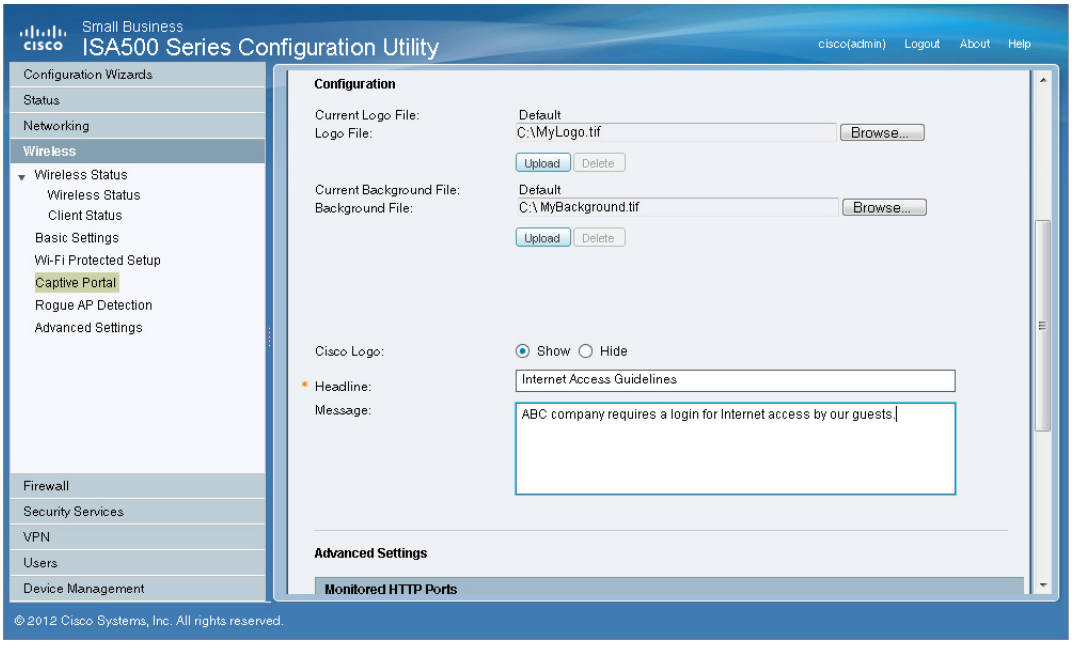


STEP 6 If you chose **Internal** or **Internal, no auth with accept button**, set up the default HotSpot Login page:

- **Logo File:** You can import an image, such as your corporate logo, to display on the login page. Click **Browse** to locate and select an image file from your local PC and then click **Upload**. To delete the loaded file, click **Delete**.
- **Background File:** You can import an image to display as the background for the login page. Click **Browse** to locate and select an image file (jpg, gif, or png) from your local PC and then click **Upload**. To delete the loaded file, click **Delete**.

  NOTE: When uploading a file, select a bmp, jpg, gif, or png file of 200KB or less. The Current Logo File field displays the filename of the file that is in use, or *Default* if no file has been uploaded for this purpose.
- **Cisco Logo:** If you want to hide the Cisco logo that appears on the login page, choose **Hide**. Otherwise, choose **Show**.
- **Headline:** If you want to create your own headline on the login page, enter the desired text in this field.

- **Message:** If you want to create your own message on the login page, enter the desired text in this field.



STEP 7 If you chose **External** or **External, no auth with accept button**, specify these settings for your external portal page:

- **Authentication Web Server:** Enter the full URL of the external web server (including https://), for example https://172.24.10.10/cgi-bin/PortalLogin.cgi.
- **Authentication Web Key:** Enter the key used to protect the username and password that the external web server sends to the security appliance for authentication.

STEP 8 If you want to use the portal for HTTP requests through other ports besides the default 80 and 443, add the ports in the **Advanced Settings > Monitored HTTP Ports** area.

NOTE: Captive Portal only monitors HTTPS requests through the port 443.

a. Click **Add**.

b. Enter the port number in the **Port** field.

c. Click **OK** to save your settings.

STEP 9 If you want to bypass the portal for certain IP addresses, add them in the **Advanced Settings > Open Domains** area.

a. Click **Add**.

b. Enter the IP address or domain name in the **Domain** field.

c. Click **OK** to save your settings.

STEP 10 Click **Save** to apply your settings.

Troubleshooting

Problem 1: User is not redirected to portal page when internal web authentication type is chosen.

Solution: Either of the following could resolve the problem:

- Check the device is connected to Captive Portals wireless network and the IP address is assigned to the device.
- Check Web Authentication Type is selected as Internal or Internal, no auth with accept button.
- Check the TCP ports on which HTTP requests are sent are added under Monitored HTTP Ports under Advanced Settings on Captive Portal page.

Problem 2: User is not redirected to portal page when internal web authentication type is chosen.

Solution: Either of the following could resolve the problem:

- Check the device is connected to Captive Portals wireless network and the IP address is assigned to the device. .
- Check Web Authentication Type is selected as External or External, no auth with accept button.
- Check the TCP ports on which HTTP requests are sent are added under Monitored HTTP Ports under Advanced Settings on Captive Portal page.
- Check the connectivity of Web-server from ISA500.
- Web-server should be able to accessed by the devices on the Captive Portal wireless network. In other words, the firewall rules associated with

the VLAN to which Captive Portal users join should be able to access the web-server.

- Check if the web-server has any issues.

Using External Web-Hosted CGI Scripts

Following is a CGI script which asks for the authentication information of a user.

The secret string programmed in the **uamsecret** variable should be configured as Authentication Web Key on the Captive portal page. Replace the **MySMB** string in the following section with your company name.

```
# !/usr/bin/perl
# chilli - ChilliSpot.org. A Wireless LAN Access Point Controller
# Copyright (C) 2003, 2004 Mondru AB.
#
# The contents of this file may be used under the terms of the GNU
# General Public License Version 2, provided that the above copyright
# notice and this permission notice is included in all copies or
# substantial portions of the software.

# Redirects from ChilliSpot daemon:
#
# Redirection when not yet or already authenticated
#   notyet:  ChilliSpot daemon redirects to login page.
#   already: ChilliSpot daemon redirects to success status page.
#
# Response to login:
#   already: Attempt to login when already logged in.
#   failed:  Login failed
#   success: Login succeded
#
# logoff:  Response to a logout


# Shared secret used to encrypt challenge with. Prevents dictionary attacks.
# You should change this to your own shared secret.
$uamsecret = "ht2eb8ej6s4et3rg1ulp";

# Uncomment the following line if you want to use ordinary user-password
# for radius authentication. Must be used together with $uamsecret.
$userpassword=1; [1]

# Our own path
$loginpath = $ENV{'SCRIPT_URL'};

use Digest::MD5  qw(md5 md5_hex md5_base64);

# Make sure that the form parameters are clean
```

```
$OK_CHARS='-a-zA-Z0-9_.@&=%!';
$| = 1;
if ($ENV{'CONTENT_LENGTH'}) {
    read (STDIN, $_, $ENV{'CONTENT_LENGTH'});
}
s/[^$OK_CHARS]/_/go;
$input = $_;


# Make sure that the get query parameters are clean
$OK_CHARS='-a-zA-Z0-9_.@&=%!';
$_ = $query=$ENV{QUERY_STRING};
s/[^$OK_CHARS]/_/go;
$query = $_;


# If she did not use https tell her that it was wrong.
if (!($ENV{HTTPS} =~ /^on$/)) {
    print "Content-type: text/html\n\n
<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\">
<html>
<head>
  <title>MySMB Login Failed</title>[7.1]
  <meta http-equiv=\"Cache-control\" content=\"no-cache\">
  <meta http-equiv=\"Pragma\" content=\"no-cache\">
</head>
<body bgColor = '#c0d8f4'>
  <h1 style=\"text-align: center;\">MySMB Login Failed</h1>[7.2]
  <center>
    Login must use encrypted connection.
  </center>
</body>
<!--
<?xml version=\"1.0\" encoding=\"UTF-8\"?>
<WISPAccessGatewayParam
  xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"
  xsi:noNamespaceSchemaLocation=
\"http://www.acmewisp.com/WISPAccessGatewayParam.xsd\">
<AuthenticationReply>
<MessageType>120</MessageType>
<ResponseCode>102</ResponseCode>
<ReplyMessage>Login must use encrypted connection</ReplyMessage>[7.3]
</AuthenticationReply>
</WISPAccessGatewayParam>
-->
</html>
";
    exit(0);
}


#Read form parameters which we care about
@array = split('&',$input);
foreach $var ( @array )
{
```

```
    @array2 = split('=',$var);
    if ($array2[0] =~ /^UserName$/) { $username = $array2[1]; }
    if ($array2[0] =~ /^Password$/) { $password = $array2[1]; }
    if ($array2[0] =~ /^challenge$/) { $challenge = $array2[1]; }
    if ($array2[0] =~ /^button$/) { $button = $array2[1]; }
    if ($array2[0] =~ /^logout$/) { $logout = $array2[1]; }
    if ($array2[0] =~ /^prelogin$/) { $prelogin = $array2[1]; }
    if ($array2[0] =~ /^res$/) { $res = $array2[1]; }
    if ($array2[0] =~ /^uamip$/) { $uamip = $array2[1]; }
    if ($array2[0] =~ /^uamport$/) { $uamport = $array2[1]; }
    if ($array2[0] =~ /^userurl$/)   { $userurl = $array2[1]; }
    if ($array2[0] =~ /^timeleft$/)  { $timeleft = $array2[1]; }
    if ($array2[0] =~ /^redirurl$/)  { $redirurl = $array2[1]; }
}

#Read query parameters which we care about
@array = split('&',$query);
foreach $var ( @array )
{
    @array2 = split('=',$var);
    if ($array2[0] =~ /^res$/)       { $res = $array2[1]; }
    if ($array2[0] =~ /^challenge$/) { $challenge = $array2[1]; }
    if ($array2[0] =~ /^uamip$/)     { $uamip = $array2[1]; }
    if ($array2[0] =~ /^uamport$/)   { $uamport = $array2[1]; }
    if ($array2[0] =~ /^reply$/)     { $reply = $array2[1]; }
    if ($array2[0] =~ /^userurl$/)   { $userurl = $array2[1]; }
    if ($array2[0] =~ /^timeleft$/)  { $timeleft = $array2[1]; }
    if ($array2[0] =~ /^redirurl$/)  { $redirurl = $array2[1]; }
}


$reply =~ s/\+/ /g;
$reply =~s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/seg;

$userurldecode = $userurl;
$userurldecode =~ s/\+/ /g;
$userurldecode =~s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/seg;

$redirurldecode = $redirurl;
$redirurldecode =~ s/\+/ /g;
$redirurldecode =~s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/seg;

$password =~ s/\+/ /g;
$password =~s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/seg;

# If attempt to login
if ($button =~ /^Login$/) {
    $hexchal  = pack "H32", $challenge;
    if (defined $uamsecret) {
   $newchal  = md5($hexchal, $uamsecret);
    }
    else {
   $newchal  = $hexchal;
    }
    $response = md5_hex("\0", $password, $newchal);
```

```
    $pappassword = unpack "H32", ($password ^ $newchal);
#sleep 5;
print "Content-type: text/html\n\n";
print "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\">
<html>
<head>
  <title>MySMB Login</title>
  <meta http-equiv=\"Cache-control\" content=\"no-cache\">
  <meta http-equiv=\"Pragma\" content=\"no-cache\">";
    if ((defined $uamsecret) && defined($userpassword)) {
   print "  <meta http-equiv=\"refresh\" content=\"0;url=
http://$uamip:$uamport/logon?username=$username&password=
$pappassword&userurl=$userurl\">";
    } else {
   print "  <meta http-equiv=\"refresh\" content=\"0;url=
http://$uamip:$uamport/logon?username=$username&response=$response&userurl=
$userurl\">";
    }
print "</head>
<body bgColor = '#c0d8f4'>";
  print "<h1 style=\"text-align: center;\">Logging in to MySMB</h1>";
  print "
  <center>
    Please wait......
  </center>
</body>
<!--
<?xml version=\"1.0\" encoding=\"UTF-8\"?>
<WISPAccessGatewayParam
  xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"
  xsi:noNamespaceSchemaLocation=
\"http://www.acmewisp.com/WISPAccessGatewayParam.xsd\">
<AuthenticationReply>
<MessageType>120</MessageType>
<ResponseCode>201</ResponseCode>
";
    if ((defined $uamsecret) && defined($userpassword)) {
   print "<LoginResultsURL>http://$uamip:$uamport/logon?username=
$username&password=$pappassword</LoginResultsURL>";
    } else {
   print "<LoginResultsURL>http://$uamip:$uamport/logon?username=
$username&response=$response&userurl=$userurl</LoginResultsURL>";
    }
print "</AuthenticationReply>
</WISPAccessGatewayParam>
-->
</html>
";
    exit(0);
}


# Default: It was not a form request
$result = 0;
```

```
# If login successful
if ($res =~ /^success$/) {
    $result = 1;
}

# If login failed
if ($res =~ /^failed$/) {
    $result = 2;
}

# If logout successful
if ($res =~ /^logoff$/) {
    $result = 3;
}

# If tried to login while already logged in
if ($res =~ /^already$/) {
    $result = 4;
}

# If not logged in yet
if ($res =~ /^notyet$/) {
    $result = 5;
}

# If login from smart client
if ($res =~ /^smartclient$/) {
    $result = 6;
}

# If requested a logging in pop up window
if ($res =~ /^popup1$/) {
    $result = 11;
}

# If requested a success pop up window
if ($res =~ /^popup2$/) {
    $result = 12;
}

# If requested a logout pop up window
if ($res =~ /^popup3$/) {
    $result = 13;
}


# Otherwise it was not a form request
# Send out an error message
if ($result == 0) {
    print "Content-type: text/html\n\n
<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\">
<html>
<head>
  <title>MySMB Login Failed</title>
  <meta http-equiv=\"Cache-control\" content=\"no-cache\">
```

```
  <meta http-equiv=\"Pragma\" content=\"no-cache\">
</head>
<body bgColor = '#c0d8f4'>
  <h1 style=\"text-align: center;\">MySMB Login Failed</h1>
  <center>
    Login must be performed through MySMB daemon.
  </center>
</body>
</html>
";
    exit(0);
}

#Generate the output
print "Content-type: text/html\n\n
<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\">
<html>
<head>
  <title>MySMB Login</title>[2.1]
  <meta http-equiv=\"Cache-control\" content=\"no-cache\">
  <meta http-equiv=\"Pragma\" content=\"no-cache\">
  <SCRIPT LANGUAGE=\"JavaScript\">
    var blur = 0;
    var starttime = new Date();
    var startclock = starttime.getTime();
    var mytimeleft = 0;

    function doTime() {
      window.setTimeout( \"doTime()\", 1000 );
      t = new Date();
      time = Math.round((t.getTime() - starttime.getTime())/1000);
      if (mytimeleft) {
        time = mytimeleft - time;
        if (time <= 0) {
          window.location = \"$loginpath?res=popup3&uamip=$uamip&uamport=
$uamport\";
        }
      }
      if (time < 0) time = 0;
      hours = (time - (time % 3600)) / 3600;
      time = time - (hours * 3600);
      mins = (time - (time % 60)) / 60;
      secs = time - (mins * 60);
      if (hours < 10) hours = \"0\" + hours;
      if (mins < 10) mins = \"0\" + mins;
      if (secs < 10) secs = \"0\" + secs;
      title = \"Online time: \" + hours + \":\" + mins + \":\" + secs;
      if (mytimeleft) {
        title = \"Remaining time: \" + hours + \":\" + mins + \":\" + secs;
      }
      if(document.all || document.getElementById){
         document.title = title;
      }
      else {
        self.status = title;
```

```
      }
    }

    function popUp(URL) {
      if (self.name != \"chillispot_popup\") {
        chillispot_popup = window.open(URL, 'chillispot_popup', 'toolbar=
0,scrollbars=0,location=0,statusbar=0,menubar=0,resizable=0,width=
500,height=375');
      }
    }

    function doOnLoad(result, URL, userurl, redirurl, timeleft) {
      if (timeleft) {
        mytimeleft = timeleft;
      }
      if ((result == 1) && (self.name == \"chillispot_popup\")) {
        doTime();
      }
      if ((result == 1) && (self.name != \"chillispot_popup\")) {
        chillispot_popup = window.open(URL, 'chillispot_popup', 'toolbar=
0,scrollbars=0,location=0,statusbar=0,menubar=0,resizable=0,width=
500,height=375');
      }
      if ((result == 2) || result == 5) {
        document.form1.UserName.focus()
      }
      if ((result == 2) && (self.name != \"chillispot_popup\")) {
        chillispot_popup = window.open('', 'chillispot_popup', 'toolbar=
0,scrollbars=0,location=0,statusbar=0,menubar=0,resizable=0,width=
400,height=200');
        chillispot_popup.close();
      }
      if ((result == 12) && (self.name == \"chillispot_popup\")) {
        doTime();
        if (redirurl) {
          opener.location = redirurl;
        }
        else if (userurl) {
          opener.location = userurl;
        }
        else if (opener.home) {
          opener.home();
        }
        else {
          opener.location = \"about:home\";
        }
        self.focus();
        blur = 0;
      }
      if ((result == 13) && (self.name == \"chillispot_popup\")) {
        self.focus();
        blur = 1;
      }
    }
```

```
    function doOnBlur(result) {
      if ((result == 12) && (self.name == \"chillispot_popup\")) {
        if (blur == 0) {
          blur = 1;
          self.focus();
        }
      }
    }
  </script>
</head>
<body onLoad=\"javascript:doOnLoad($result, '$loginpath?res=popup2&uamip=
$uamip&uamport=$uamport&userurl=$userurl&redirurl=$redirurl&timeleft=
$timeleft','$userurldecode', '$redirurldecode', '$timeleft')\" onBlur =
\"javascript:doOnBlur($result)\" bgColor = '#c0d8f4'>";


#      if (!window.opener) {
#        document.bgColor = '#c0d8f4';
#      }

#print "THE INPUT: $input";
#foreach $key (sort (keys %ENV)) {
#  print $key, ' = ', $ENV{$key}, "<br>\n";
#}

if ($result == 2) {
    print "
  <h1 style=\"text-align: center;\">MySMB Login Failed</h1>";[6.1]
    if ($reply) {
    print "<center> $reply </BR></BR></center>";
    }
}

if ($result == 5) {
    print "
  <h1 style=\"text-align: center;\">MySMB Login</h1>";[2.2]
}

if ($result == 2 || $result == 5) {
  print "
  <form name=\"form1\" method=\"post\" action=\"$loginpath\">
  <INPUT TYPE=\"hidden\" NAME=\"challenge\" VALUE=\"$challenge\">
  <INPUT TYPE=\"hidden\" NAME=\"uamip\" VALUE=\"$uamip\">
  <INPUT TYPE=\"hidden\" NAME=\"uamport\" VALUE=\"$uamport\">
  <INPUT TYPE=\"hidden\" NAME=\"userurl\" VALUE=\"$userurldecode\">
  <center>
  <table border=\"0\" cellpadding=\"5\" cellspacing=\"0\" style=\"width:
217px;\">
    <tbody>
      <tr>
        <td align=\"right\">Username:</td>[2.3]
        <td><input STYLE=\"font-family: Arial\" type=\"text\" name=
\"UserName\" size=\"20\" maxlength=\"128\"></td>
      </tr>
      <tr>
```

```
        <td align=\"right\">Password:</td>[2.4]
        <td><input STYLE=\"font-family: Arial\" type=\"password\" name=
\"Password\" size=\"20\" maxlength=\"128\"></td>
      </tr>
      <tr>
        <td align=\"center\" colspan=\"2\" height=\"23\"><input type=
\"submit\" name=\"button\" value=\"Login\"[2.5] onClick=
\"javascript:popUp('$loginpath?res=popup1&uamip=$uamip&uamport=
$uamport')\"></td>
      </tr>
    </tbody>
  </table>
  </center>
  </form>
</body>
</html>";
}

if ($result == 1) {
  print "
  <h1 style=\"text-align: center;\">Logged in to MySMB</h1>";[8.1]

  if ($reply) {
      print "<center> $reply </BR></BR></center>";
  }

  print "
  <center>
    <a href=\"http://$uamip:$uamport/logoff\">Logout</a>[8.2]
  </center>
</body>
</html>";
}

if (($result == 4) || ($result == 12)) {
  print "
  <h1 style=\"text-align: center;\">Logged in to MySMB</h1>[4.1]
  <center>
    <a href=\"http://$uamip:$uamport/logoff\">Logout</a>[4.2]
  </center>
</body>
</html>";
}


if ($result == 11) {
  print "<h1 style=\"text-align: center;\">Logging in to MySMB</h1>[3.1]";
  print "
  <center>
    Please wait..... [3.2]
  </center>
</body>
</html>";
}
```

```
if (($result == 3) || ($result == 13)) {
    print "
  <h1 style=\"text-align: center;\">Logged out from MySMB</h1>[5.1]
  <center>
    <a href=\"http://$uamip:$uamport/prelogin\">Login</a>[5.2]
  </center>
</body>
</html>";
}


exit(0);
```

CGI Source Code Example: No Authentication and Accept Button

Following is a CGI script which presents a Accept button on the portal page.

The secret string programmed in **uamsecret** variable should be configured as Authentication Web Key on the Captive portal page. Replace the **MySMB** string in the following section with your company name.

```
#!/usr/bin/perl

# chilli - ChilliSpot.org. A Wireless LAN Access Point Controller
# Copyright (C) 2003, 2004 Mondru AB.
#
# The contents of this file may be used under the terms of the GNU
# General Public License Version 2, provided that the above copyright
# notice and this permission notice is included in all copies or
# substantial portions of the software.

# Redirects from ChilliSpot daemon:
#
# Redirection when not yet or already authenticated
#   notyet:  ChilliSpot daemon redirects to login page.
#   already: ChilliSpot daemon redirects to success status page.
#
# Response to login:
#   already: Attempt to login when already logged in.
#   failed:  Login failed
#   success: Login succeded
#
# logoff:  Response to a logout


# Shared secret used to encrypt challenge with. Prevents dictionary attacks.
# You should change this to your own shared secret.
#$uamsecret = "ht2eb8ej6s4et3rg1ulp";
```

```
$uamsecret = "gemteksmb";

# Uncomment the following line if you want to use ordinary user-password
# for radius authentication. Must be used together with $uamsecret.
$userpassword=1;

# Our own path
$loginpath = $ENV{'SCRIPT_URL'};

use Digest::MD5  qw(md5 md5_hex md5_base64);

# Make sure that the form parameters are clean
$OK_CHARS='-a-zA-Z0-9_.@&=%!';
$| = 1;
if ($ENV{'CONTENT_LENGTH'}) {
    read (STDIN, $_, $ENV{'CONTENT_LENGTH'});
}
s/[^$OK_CHARS]/_/go;
$input = $_;


# Make sure that the get query parameters are clean
$OK_CHARS='-a-zA-Z0-9_.@&=%!';
$_ = $query=$ENV{QUERY_STRING};
s/[^$OK_CHARS]/_/go;
$query = $_;


# If she did not use https tell her that it was wrong.
if (!($ENV{HTTPS} =~ /^on$/)) {
    print "Content-type: text/html\n\n
<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\">
<html>
<head>
  <title>MySMB Login Failed</title>
  <meta http-equiv=\"Cache-control\" content=\"no-cache\">
  <meta http-equiv=\"Pragma\" content=\"no-cache\">
</head>
<body bgColor = '#c0d8f4'>
  <h1 style=\"text-align: center;\">MySMB Login Failed</h1>
  <center>
    Login must use encrypted connection.
  </center>
</body>
<!--
<?xml version=\"1.0\" encoding=\"UTF-8\"?>
<WISPAccessGatewayParam
  xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"
  xsi:noNamespaceSchemaLocation=
\"http://www.acmewisp.com/WISPAccessGatewayParam.xsd\">
<AuthenticationReply>
<MessageType>120</MessageType>
<ResponseCode>102</ResponseCode>
<ReplyMessage>Login must use encrypted connection</ReplyMessage>
</AuthenticationReply>
```

```
</WISPAccessGatewayParam>
-->
</html>
";
    exit(0);
}


#Read form parameters which we care about
@array = split('&',$input);
foreach $var ( @array )
{
    @array2 = split('=',$var);
    if ($array2[0] =~ /^UserName$/) { $username = $array2[1]; }
    if ($array2[0] =~ /^Password$/) { $password = $array2[1]; }
    if ($array2[0] =~ /^challenge$/) { $challenge = $array2[1]; }
    if ($array2[0] =~ /^button$/) { $button = $array2[1]; }
    if ($array2[0] =~ /^logout$/) { $logout = $array2[1]; }
    if ($array2[0] =~ /^prelogin$/) { $prelogin = $array2[1]; }
    if ($array2[0] =~ /^res$/) { $res = $array2[1]; }
    if ($array2[0] =~ /^uamip$/) { $uamip = $array2[1]; }
    if ($array2[0] =~ /^uamport$/) { $uamport = $array2[1]; }
    if ($array2[0] =~ /^userurl$/)   { $userurl = $array2[1]; }
    if ($array2[0] =~ /^timeleft$/)  { $timeleft = $array2[1]; }
    if ($array2[0] =~ /^redirurl$/)  { $redirurl = $array2[1]; }
}

#Read query parameters which we care about
@array = split('&',$query);
foreach $var ( @array )
{
    @array2 = split('=',$var);
    if ($array2[0] =~ /^res$/)       { $res = $array2[1]; }
    if ($array2[0] =~ /^challenge$/) { $challenge = $array2[1]; }
    if ($array2[0] =~ /^uamip$/)     { $uamip = $array2[1]; }
    if ($array2[0] =~ /^uamport$/)   { $uamport = $array2[1]; }
    if ($array2[0] =~ /^reply$/)     { $reply = $array2[1]; }
    if ($array2[0] =~ /^userurl$/)   { $userurl = $array2[1]; }
    if ($array2[0] =~ /^timeleft$/)  { $timeleft = $array2[1]; }
    if ($array2[0] =~ /^redirurl$/)  { $redirurl = $array2[1]; }
}


$reply =~ s/\+/ /g;
$reply =~s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/seg;

$userurldecode = $userurl;
$userurldecode =~ s/\+/ /g;
$userurldecode =~s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/seg;

$redirurldecode = $redirurl;
$redirurldecode =~ s/\+/ /g;
$redirurldecode =~s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/seg;

$password =~ s/\+/ /g;
```

```
$password =~s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/seg;

# If attempt to login
if ($button =~ /^Accept$/) {
    $hexchal  = pack "H32", $challenge;
    if (defined $uamsecret) {
   $newchal  = md5($hexchal, $uamsecret);
    }
    else {
   $newchal  = $hexchal;
    }
    $response = md5_hex("\0", $password, $newchal);
    $pappassword = unpack "H32", ($password ^ $newchal);
#sleep 5;
print "Content-type: text/html\n\n";
print "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\">
<html>
<head>
  <title>MySMB Login</title>
  <meta http-equiv=\"Cache-control\" content=\"no-cache\">
  <meta http-equiv=\"Pragma\" content=\"no-cache\">";
    if ((defined $uamsecret) && defined($userpassword)) {
   print "  <meta http-equiv=\"refresh\" content=\"0;url=
http://$uamip:$uamport/logon?username=$username&password=
$pappassword&userurl=$userurl\">";
    } else {
   print "  <meta http-equiv=\"refresh\" content=\"0;url=
http://$uamip:$uamport/logon?username=$username&response=$response&userurl=
$userurl\">";
    }
print "</head>
<body bgColor = '#c0d8f4'>";
  print "<h1 style=\"text-align: center;\">Logging in to MySMB</h1>";
  print "
  <center>
    Please wait......
  </center>
</body>
<!--
<?xml version=\"1.0\" encoding=\"UTF-8\"?>
<WISPAccessGatewayParam
  xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"
  xsi:noNamespaceSchemaLocation=
\"http://www.acmewisp.com/WISPAccessGatewayParam.xsd\">
<AuthenticationReply>
<MessageType>120</MessageType>
<ResponseCode>201</ResponseCode>
";
    if ((defined $uamsecret) && defined($userpassword)) {
   print "<LoginResultsURL>http://$uamip:$uamport/logon?username=
$username&password=$pappassword</LoginResultsURL>";
    } else {
   print "<LoginResultsURL>http://$uamip:$uamport/logon?username=
$username&response=$response&userurl=$userurl</LoginResultsURL>";
    }
```

```
print "</AuthenticationReply>
</WISPAccessGatewayParam>
-->
</html>
";
    exit(0);
}


# Default: It was not a form request
$result = 0;

# If login successful
if ($res =~ /^success$/) {
    $result = 1;
}

# If login failed
if ($res =~ /^failed$/) {
    $result = 2;
}

# If logout successful
if ($res =~ /^logoff$/) {
    $result = 3;
}

# If tried to login while already logged in
if ($res =~ /^already$/) {
    $result = 4;
}

# If not logged in yet
if ($res =~ /^notyet$/) {
    $result = 5;
}

# If login from smart client
if ($res =~ /^smartclient$/) {
    $result = 6;
}

# If requested a logging in pop up window
if ($res =~ /^popup1$/) {
    $result = 11;
}

# If requested a success pop up window
if ($res =~ /^popup2$/) {
    $result = 12;
}

# If requested a logout pop up window
if ($res =~ /^popup3$/) {
    $result = 13;
```

```
}


# Otherwise it was not a form request
# Send out an error message
if ($result == 0) {
    print "Content-type: text/html\n\n
<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\">
<html>
<head>
  <title>MySMB Login Failed</title>
  <meta http-equiv=\"Cache-control\" content=\"no-cache\">
  <meta http-equiv=\"Pragma\" content=\"no-cache\">
</head>
<body bgColor = '#c0d8f4'>
  <h1 style=\"text-align: center;\">MySMB Login Failed</h1>
  <center>
    Login must be performed through MySMB daemon.
  </center>
</body>
</html>
";
    exit(0);
}

#Generate the output
print "Content-type: text/html\n\n
<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\">
<html>
<head>
  <title>MySMB Login</title>
  <meta http-equiv=\"Cache-control\" content=\"no-cache\">
  <meta http-equiv=\"Pragma\" content=\"no-cache\">
  <SCRIPT LANGUAGE=\"JavaScript\">
    var blur = 0;
    var starttime = new Date();
    var startclock = starttime.getTime();
    var mytimeleft = 0;

    function doTime() {
      window.setTimeout( \"doTime()\", 1000 );
      t = new Date();
      time = Math.round((t.getTime() - starttime.getTime())/1000);
      if (mytimeleft) {
        time = mytimeleft - time;
        if (time <= 0) {
          window.location = \"$loginpath?res=popup3&uamip=$uamip&uamport=
$uamport\";
        }
      }
      if (time < 0) time = 0;
      hours = (time - (time % 3600)) / 3600;
      time = time - (hours * 3600);
      mins = (time - (time % 60)) / 60;
      secs = time - (mins * 60);
```

```
      if (hours < 10) hours = \"0\" + hours;
      if (mins < 10) mins = \"0\" + mins;
      if (secs < 10) secs = \"0\" + secs;
      title = \"Online time: \" + hours + \":\" + mins + \":\" + secs;
      if (mytimeleft) {
        title = \"Remaining time: \" + hours + \":\" + mins + \":\" + secs;
      }
      if(document.all || document.getElementById){
         document.title = title;
      }
      else {
        self.status = title;
      }
    }

    function popUp(URL) {
      if (self.name != \"chillispot_popup\") {
        chillispot_popup = window.open(URL, 'chillispot_popup', 'toolbar=
0,scrollbars=0,location=0,statusbar=0,menubar=0,resizable=0,width=
500,height=375');
      }
    }

    function doOnLoad(result, URL, userurl, redirurl, timeleft) {
   if (timeleft) {
        mytimeleft = timeleft;
      }
      if ((result == 1) && (self.name == \"chillispot_popup\")) {
        doTime();
      }
      if ((result == 1) && (self.name != \"chillispot_popup\")) {
        chillispot_popup = window.open(URL, 'chillispot_popup', 'toolbar=
0,scrollbars=0,location=0,statusbar=0,menubar=0,resizable=0,width=
500,height=375');
      }
      if ((result == 2) || result == 5) {
        //document.form1.UserName.focus()
      }
      if ((result == 2) && (self.name != \"chillispot_popup\")) {
        chillispot_popup = window.open('', 'chillispot_popup', 'toolbar=
0,scrollbars=0,location=0,statusbar=0,menubar=0,resizable=0,width=
400,height=200');
        chillispot_popup.close();
      }
      if ((result == 12) && (self.name == \"chillispot_popup\")) {
        doTime();
        if (redirurl) {
          opener.location = redirurl;
        }
        else if (userurl) {
          opener.location = userurl;
        }
        else if (opener.home) {
          opener.home();
        }
```

```
        else {
          opener.location = \"about:home\";
        }
        self.focus();
        blur = 0;
      }
      if ((result == 13) && (self.name == \"chillispot_popup\")) {
        self.focus();
        blur = 1;
      }
    }

    function doOnBlur(result) {
      if ((result == 12) && (self.name == \"chillispot_popup\")) {
        if (blur == 0) {
          blur = 1;
          self.focus();
        }
      }
    }
  </script>
</head>
<body onLoad=\"javascript:doOnLoad($result, '$loginpath?res=popup2&uamip=
$uamip&uamport=$uamport&userurl=$userurl&redirurl=$redirurl&timeleft=
$timeleft','$userurldecode', '$redirurldecode', '$timeleft')\" onBlur =
\"javascript:doOnBlur($result)\" bgColor = '#c0d8f4'>";


#      if (!window.opener) {
#        document.bgColor = '#c0d8f4';
#      }

#print "THE INPUT: $input";
#foreach $key (sort (keys %ENV)) {
#  print $key, ' = ', $ENV{$key}, "<br>\n";
#}

if ($result == 2) {
    print "
  <h1 style=\"text-align: center;\">MySMB Login Failed</h1>";
    if ($reply) {
    print "<center> $reply </BR></BR></center>";
    }
}

if ($result == 5) {
    print "
  <h1 style=\"text-align: center;\">MySMB Login</h1>";
}

if ($result == 2 || $result == 5) {
  print "
  <form name=\"form1\" method=\"post\" action=\"$loginpath\">
  <INPUT TYPE=\"hidden\" NAME=\"challenge\" VALUE=\"$challenge\">
  <INPUT TYPE=\"hidden\" NAME=\"uamip\" VALUE=\"$uamip\">
```

```
  <INPUT TYPE=\"hidden\" NAME=\"uamport\" VALUE=\"$uamport\">
  <INPUT TYPE=\"hidden\" NAME=\"userurl\" VALUE=\"$userurldecode\">
  <INPUT TYPE=\"hidden\" NAME=\"UserName\" VALUE=\"\">
  <INPUT TYPE=\"hidden\" NAME=\"Password\" VALUE=\"\">
  <center>
  <table border=\"0\" cellpadding=\"5\" cellspacing=\"0\" style=\"width:
217px;\">
    <tbody>
      <tr>
        <td align=\"center\" colspan=\"2\" height=\"23\"><input type=
\"submit\" name=\"button\" value=\"Accept\" onClick=
\"javascript:popUp('$loginpath?res=popup1&uamip=$uamip&uamport=
$uamport')\"></td>
      </tr>
    </tbody>
  </table>
  </center>
  </form>
</body>
</html>";
}

if ($result == 1) {
  print "
  <h1 style=\"text-align: center;\">Logged in to MySMB</h1>";

  if ($reply) {
      print "<center> $reply </BR></BR></center>";
  }

  print "
  <center>
    <a href=\"http://$uamip:$uamport/logoff\">Logout</a>
  </center>
</body>
</html>";
}

if (($result == 4) || ($result == 12)) {
  print "
  <h1 style=\"text-align: center;\">Logged in to MySMB</h1>
  <center>
    <a href=\"http://$uamip:$uamport/logoff\">Logout</a>
  </center>
</body>
</html>";
}


if ($result == 11) {
  print "<h1 style=\"text-align: center;\">Logging in to MySMB</h1>";
  print "
  <center>
    Please wait......
  </center>
```

```
</body>
</html>";
}


if (($result == 3) || ($result == 13)) {
    print "
  <h1 style=\"text-align: center;\">Logged out from MySMB</h1>
  <center>
    <a href=\"http://$uamip:$uamport/prelogin\">Login</a>
  </center>
</body>
</html>";
}


exit(0);
```

Related Information

Support	
Cisco Small Business Support Community	www.cisco.com/go/smallbizsupport
Cisco Small Business Support and Resources	www.cisco.com/go/smallbizhelp
Phone Support Contacts	www.cisco.com/go/sbsc
Cisco Small Business Firmware Downloads	www.cisco.com/go/isa500software
Cisco Small Business Open Source Requests	www.cisco.com/go/smallbiz_opensource_request
Documentation	
Product Documentation	www.cisco.com/go/isa500resources
Cisco Small Business	
Cisco Partner Central for Small Business (Partner Login Required)	www.cisco.com/web/partners/sell/smb

Cisco Small Business Home	www.cisco.com/smb

78-21182-01

Configuring Wireless Rogue AP Detection

A Rogue AP is an access point connected to your network without authorization. It is not under the management of your network administrators and does not necessarily conform to your network security policies.

The security appliance provides proactive Rogue AP Detection in the 2.4-GHz band. Rogue AP Detection is able to discover, detect, and report unauthorized access points. You can specify an authorized access point by its MAC address.

STEP 1 Click **Wireless > Rogue AP Detection**.

The Rogue AP Detection window opens.

STEP 2 Click **On** to enable Rogue AP Detection, or click **Off** to disable it.

STEP 3 If you enable Rogue AP Detection, all rogue access points detected by the security appliance in the vicinity of the network appear in the list of Detected Rogue Access Points. The MAC address of the detected access point is displayed. You can locate the rogue access points by their MAC addresses and monitor them until they are eliminated or authorized. Click **Refresh** to update the data.

STEP 4 If an access point listed as a rogue is actually a legitimate access point, you can click **Grant Access** to set it as an authorized access point. The granted access point is moved to the list of Authorized Access Points.

STEP 5 The security appliance will not detect the authorized access points.

- To add an authorized access point, click **Add**. Enter the MAC address of the access point and click **OK**. You can specify up to 128 authorized access points.
- To delete an authorized access point from the list, click the **Delete** (x) icon.

- To change the MAC address of an authorized access point, click the **Edit** (pencil) icon.
- To export the list of authorized access points to a file, click **Export**.
- To import the list of authorized access points from a file, click **Import**.

  Choose whether to replace the existing list of Authorized Access Points or add the entries in the imported file to the list of Authorized Access Points.

  - Click **Replace** to import the list and replace the entire contents of the list of Authorized Access Points. Click **Browse** to locate the file and click **OK**.
  - Click **Merge** to import the list and add the access points in the imported file to the access points currently displayed in the list of Authorized Access Points. Click **Browse** to locate the file and click **OK**.

  After the import is complete, the screen refreshes and the MAC addresses of the imported access points appear in the list of Authorized Access Points.

STEP 6 Click **Save** to apply your settings.

Advanced Radio Settings

Use the Advanced Settings page to specify the advanced radio settings.

NOTE This page is available if the wireless radio is enabled on the Basic Settings page.

STEP 1 Click **Wireless > Advanced Settings**.

STEP 2 Enter the following information:

- **Guard Interval:** Choose either Long (800 ns) or Short (400 ns) that the security appliance will retry a frame transmission that fails.

  NOTE: The short frame is only available when the specified wireless mode includes 802.11n.

- **CTS Protection Mode:** CTS (Clear-To-Send) Protection Mode function boosts the ability of the SSID to catch all Wireless-G transmissions but will severely decrease performance.

- Select the **AUTO** radio button if you want the security appliance to perform a CTS handshake before transmitting a packet. This mode can minimize collisions among hidden stations.
- Select the **Disabled** radio button if you want to permanently disable this feature.

- **Power Output:** You can adjust the output power of the access point to get the appropriate coverage for your wireless network. Choose the level that you need for your environment. If you are not sure of which setting to select, then use the default setting, 100%.
- **Beacon Interval:** Beacon frames are transmitted by the access point at regular intervals to announce the existence of the wireless network. Set the interval by entering a value in milliseconds. Enter a value from 20 to 999 ms. The default value is 100 ms, which means that beacon frames are sent every 100 ms.
- **DTIM Interval:** The Delivery Traffic Information Map (DTIM) message is an element that is included in some beacon frames. It indicates that the client stations are currently sleeping in low-power mode and have buffered data on the access point awaiting pickup. Set the interval by entering a value in beacon frames. Enter a value from 1 to 255. The default value is 1, which means that the DTIM message is included in every second beacon frame.
- **RTS Threshold:** The RTS threshold determines the packet size that requires a Request To Send (RTS)/Clear To Send (CTS) handshake before sending. A low threshold setting can be useful in areas where many client devices are associating with the wireless device, or in areas where the clients are far apart and can detect only the access point but not other clients. Although a low threshold value consumes more bandwidth and reduces the throughput of the packet, frequent RTS packets can help the network recover from interference or collisions. Set the threshold by entering the packet size in bytes. Enter a value from 1 to 2347. The default value is 2347, which effectively disables RTS.
- **Fragmentation Threshold:** The fragmentation threshold is the frame length that requires packets to be broken up (fragmented) into two or more frames. Setting a lower value can reduce collisions because collisions occur more often in the transmission of long frames, which occupy the channel for a longer time. Use a low setting in areas where communication is poor or where there is a great deal of radio interference. Set the threshold by entering the frame length in bytes. Enter a value from 256 to 2346. The default value is 2346, which effectively disables fragmentation.

STEP 3 Click **Save** to apply your settings.