Configuring Clientless SSL VPN Users

September 13, 2013

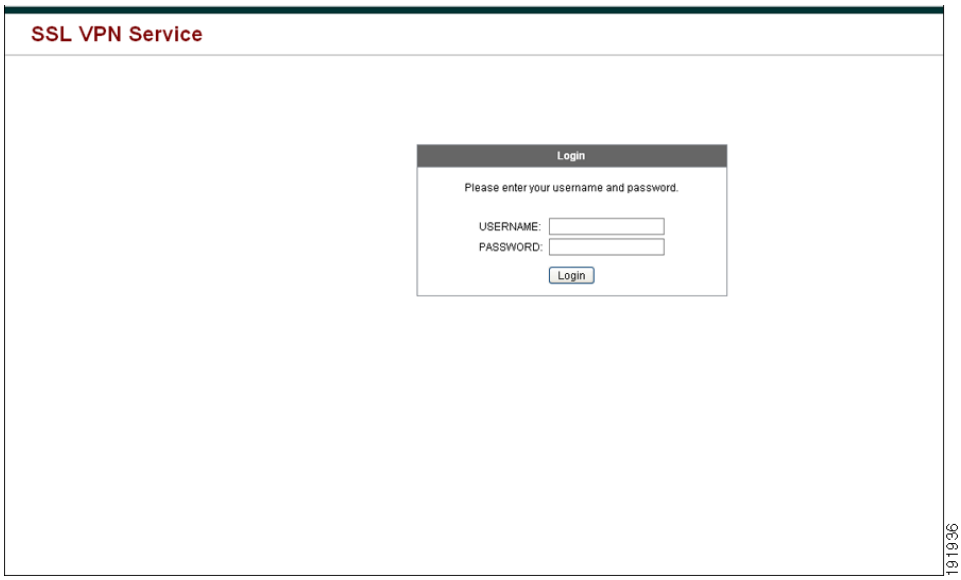
Overview

This section provides information to communicate to users to get them started using Clientless SSL VPN. It includes the following topics:

Defining the End User Interface

The Clientless SSL VPN end user interface consists of a series of HTML panels. A user logs on to Clientless SSL VPN by entering the IP address of an ASA interface in the format https://*address*. The first panel that displays is the login screen (Figure 19-1).

Figure 19-1 Clientless SSL VPN Login Screen



Viewing the Clientless SSL VPN Home Page

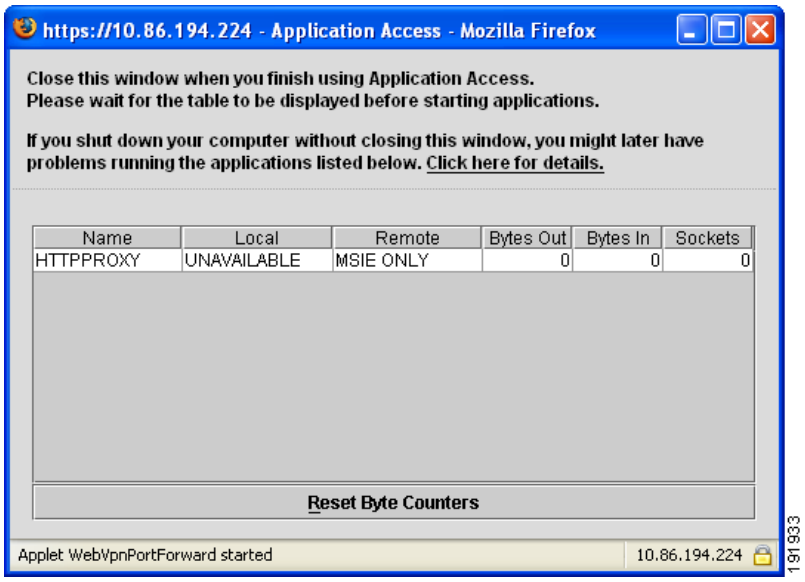
After the user logs in, the portal page opens.

The home page displays all of the Clientless SSL VPN features you have configured, and its appearance reflects the logo, text, and colors you have selected. This sample home page includes all available Clientless SSL VPN features with the exception of identifying specific file shares. It lets users browse the network, enter URLs, access specific websites, and use Application Access (port forwarding and smart tunnels) to access TCP applications.

Viewing the Clientless SSL VPN Application Access Panel

To start port forwarding or smart tunnels, a user clicks the **Go** button in the Application Access box. The Application Access window opens (Figure 19-2).

Figure 19-2 Clientless SSL VPN Application Access Window



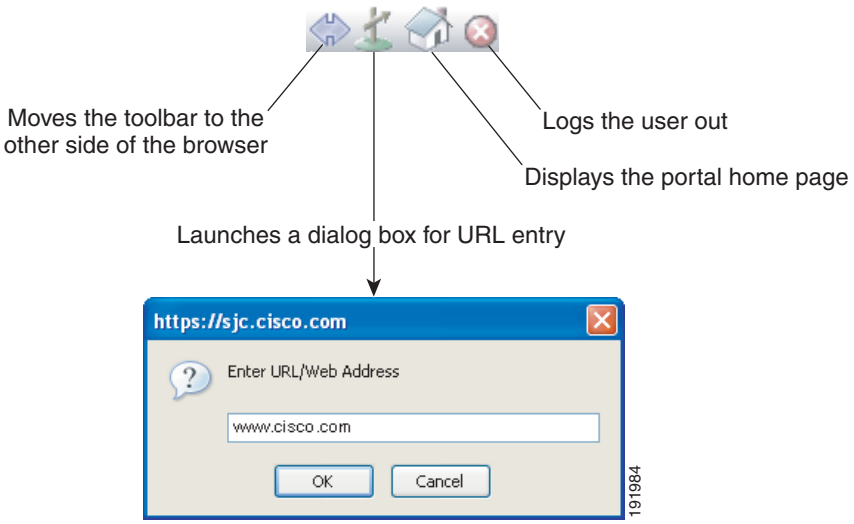
This window displays the TCP applications configured for this Clientless SSL VPN connection. To use an application with this panel open, the user starts the application in the normal way.

Note A stateful failover does not retain sessions established using Application Access. Users must reconnect following a failover.

Viewing the Floating Toolbar

The floating toolbar shown in Figure 19-3 represents the current Clientless SSL VPN session.

Figure 19-3 Clientless SSL VPN Floating Toolbar



Be aware of the following characteristics of the floating toolbar:

- The toolbar lets you enter URLs, browse file locations, and choose preconfigured Web connections without interfering with the main browser window.
- If you configure your browser to block popups, the floating toolbar cannot display.
- If you close the toolbar, the ASA prompts you to end the Clientless SSL VPN session.

See Table 19-2 on page 19-21 for detailed information about using Clientless SSL VPN.

Managing Passwords

Optionally, you can configure the ASA to warn end users when their passwords are about to expire.

The ASA supports password management for the RADIUS and LDAP protocols. It supports the "password-expire-in-days" option for LDAP only.

You can configure password management for IPsec remote access and SSL VPN tunnel-groups.

When you configure password management, the ASA notifies the remote user at login that the user's current password is about to expire or has expired. The ASA then offers the user the opportunity to change the password. If the current password has not yet expired, the user can still log in using that password.

This command is valid for AAA servers that support such notification.

The ASA, releases 7.1 and later, generally supports password management for the following connection types when authenticating with LDAP or with any RADIUS configuration that supports MS-CHAPv2:

- AnyConnect VPN Client
- IPsec VPN Client
- Clientless SSL VPN

The RADIUS server (for example, Cisco ACS) could proxy the authentication request to another authentication server. However, from the ASA perspective, it is talking only to a RADIUS server.

Prerequisites

- Native LDAP requires an SSL connection. You must enable LDAP over SSL before attempting to do password management for LDAP. By default, LDAP uses port 636.
- If you are using an LDAP directory server for authentication, password management is supported with the Sun Java System Directory Server (formerly named the Sun ONE Directory Server) and the Microsoft Active Directory.

  Sun—The DN configured on the ASA to access a Sun directory server must be able to access the default password policy on that server. We recommend using the directory administrator, or a user with directory administrator privileges, as the DN. Alternatively, you can place an ACI on the default password policy.

  Microsoft—You must configure LDAP over SSL to enable password management with Microsoft Active Directory.

  Restrictions

- Some RADIUS servers that support MSCHAP currently do not support MSCHAPv2. This command requires MSCHAPv2 so check with your vendor.

- Password management is *not* supported for any of these connection types for Kerberos/Active Directory (Windows password) or NT 4.0 Domain.
- For LDAP, the method to change a password is proprietary for the different LDAP servers on the market. Currently, the ASA implements the proprietary password management logic only for Microsoft Active Directory and Sun LDAP servers.
- The ASA ignores this command if RADIUS or LDAP authentication has not been configured.

DETAILED STEPS

Note The **password-management** command does not change the number of days before the password expires, but rather, the number of days ahead of expiration that the ASA starts warning the user that the password is about to expire.

	Command	Purpose
Step 1	`tunnel-group general-attributes`	Switches to general-attributes mode.
Step 2	`password-management`	Notifies remote users that their password is about to expire.
Step 3	`password-expire-in-days`	Specifies when the password expires.

Step 4	Enter number of days
	
	Example:
	`ciscoasa(config)# tunnel-group testgroup type webvpn`
	`ciscoasa(config)# tunnel-group testgroup general-attributes`
	`ciscoasa(config-general)# password-management password-expire-in-days 90`
		If you specify the keyword, you must also specify the number of days. If you set the number of days to 0, this command is switched off.
		Note The ASA does not notify the user of the pending expiration, but the user can change the password after it expires.
		
		Sets the days before password expiration to begin warning the user of the pending expiration to 90 for the connection profile “testgroup.”

Using Single Sign-On with Clientless SSL VPN

Single sign-on support lets users of Clientless SSL VPN enter a username and password only once to access multiple protected services and Web servers. In general, the SSO mechanism either starts as part of the AAA process or just after successful user authentication to a AAA server. The Clientless SSL VPN server running on the ASA acts as a proxy for the user to the authenticating server. When a user logs in, the Clientless SSL VPN server sends an SSO authentication request, including username and password, to the authenticating server. If the server approves the authentication request, it returns an SSO authentication cookie to the Clientless SSL VPN server. The ASA keeps this cookie on behalf of the user and uses it to authenticate the user to secure websites within the domain protected by the SSO server.

This section describes the four SSO authentication methods supported by Clientless SSL VPN: HTTP Basic and NTLMv1 (NT LAN Manager) authentication, the Computer Associates eTrust SiteMinder SSO server (formerly Netegrity SiteMinder), and Version 1.1 of Security Assertion Markup Language (SAML), the POST-type SSO server authentication.

This section includes:

Configuring SSO with HTTP Basic or NTLM Authentication

This section describes single sign-on with HTTP Basic or NTLM authentication. You can configure the ASA to implement SSO using either or both of these methods. The **auto-sign-on** command configures the ASA to automatically pass Clientless SSL VPN user login credentials (username and password) on to internal servers. You can enter multiple **auto-sign-on** commands. The ASA processes them according to the input order (early commands take precedence). You specify the servers to receive the login credentials using either IP address and IP mask, or URI mask.

Use the **auto-sign-on** command in any of three modes: Clientless SSL VPN configuration, Clientless SSL VPN group-policy mode, or Clientless SSL VPN username mode. Username supersedes group, and group supersedes global. Choose the mode with the required scope of authentication:

Mode	Scope
webvpn configuration	All Clientless SSL VPN users globally.
webvpn group-policy configuration	A subset of Clientless SSL VPN users defined by a group policy.
webvpn username configuration	An individual user of Clientless SSL VPN.

DETAILED STEPS

The following example commands present various possible combinations of modes and arguments.

	Command	Purpose

Step 1	**Example:**
	`ciscoasa(config)# webvpn`
	
	`ciscoasa(config-webvpn)# auto-sign-on allow ip 10.1.1.1 255.255.255.0 auth-type ntlm`
		Configures auto-sign-on for all users of Clientless SSL VPN to servers with IP addresses ranging from 10.1.1.0 to 10.1.1.255 using NTLM authentication.

Step 2	**Example:**
	`ciscoasa(config)# webvpn`
	`ciscoasa(config-webvpn)# auto-sign-on allow uri https://*.example.com/* auth-type basic`
		Configures auto-sign-on for all users of Clientless SSL VPN, using basic HTTP authentication, to servers defined by the URI mask https://*.example.com/*.

Step 3	**Example:**
	`ciscoasa(config)# group-policy ExamplePolicy attributes`
	`ciscoasa(config-group-policy)# webvpn`
	`ciscoasa(config-group-webvpn)# auto-sign-on allow uri https://*.example.com/* auth-type all`
		Configures auto-sign-on for Clientless SSL VPN sessions associated with the ExamplePolicy group policy, using either basic or NTLM authentication, to servers defined by the URI mask.

	Command	Purpose

Step 4	**Example:**
	`ciscoasa(config)# username Anyuser attributes`
	`ciscoasa(config-username)# webvpn`
	`ciscoasa(config-username-webvpn)# auto-sign-on allow ip 10.1.1.1 255.255.255.0 auth-type basic`
		Configures auto-sign-on for a user named Anyuser to servers with IP addresses ranging from 10.1.1.0 to 10.1.1.255 using HTTP Basic authentication.

Step 5	`(config-webvpn)# smart-tunnel auto-sign-on` *host-list* `[use-domain] [realm` *realm string*`] [port` *port num*`] [host` *host mask* `|` `ip` *address subnet mask*`]`	Configures auto-sign-on with a specific port and realm for authentication.

Configuring SSO Authentication Using SiteMinder

This section describes configuring the ASA to support SSO with SiteMinder. You would typically choose to implement SSO with SiteMinder if your website security infrastucture already incorporates SiteMinder. With this method, SSO authentication is separate from AAA and happens once the AAA process completes.

Prerequisites

- Specifying the SSO server.
- Specifying the URL of the SSO server to which the ASA makes SSO authentication requests.
- Specifying a secret key to secure the communication between the ASA and the SSO server. This key is similar to a password: you create it, save it, and enter it on both the ASA and the SiteMinder policy server using the Cisco Java plug-in authentication scheme.

Optionally, you can do the following configuration tasks in addition to the required tasks:

- Configuring the authentication request timeout.
- Configuring the number of authentication request retries.

Restrictions

To configure SSO for a user or group for Clientless SSL VPN access, you must first configure a AAA server, such as a RADIUS or LDAP server. You can then set up SSO support for Clientless SSL VPN.

DETAILED STEPS

This section presents specific steps for configuring the ASA to support SSO authentication with CA SiteMinder.

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**sso-server type** *type* **Example:** hostname(config)# **webvpn** hostname(config-webvpn)# **sso-server Example type siteminder** ciscoasa(config-webvpn-sso-siteminder)#	Creates an SSO server. Creates an SSO server named Example of type siteminder.
Step 3	**config-webvpn-sso-siteminder**	Switches to site minder configuration mode.
Step 4	**web-agent-url** **Example:** ciscoasa(config-webvpn-sso-siteminder)# **web-agent-url http://www.Example.com/webvpn** ciscoasa(config-webvpn-sso-siteminder)#	Specifies the authentication URL of the SSO server. Sends authentication requests to the URL http://www.Example.com/webvpn.
Step 5	**policy-server-secret** *secret* **Example:** ciscoasa(config-webvpn-sso-siteminder)# **policy-server-secret AtaL8rD8!** ciscoasa(config-webvpn-sso-siteminder)#	Specifies a secret key to secure the authentication communication between the ASA and SiteMinder. Creates a secret key AtaL8rD8!. You can create a key of any length using any regular or shifted alphanumeric character, but you must enter the same key on both the ASA and the SSO server.
Step 6	**request-timeout** *seconds* **Example:** ciscoasa(config-webvpn-sso-siteminder)# **request-timeout 8** ciscoasa(config-webvpn-sso-siteminder)#	Configures the number of seconds before a failed SSO authentication attempt times out. The default number of seconds is 5, and the possible range is 1 to 30. Changes the number of seconds before a request times out to 8.
Step 7	**max-retry-attempts** **Example:** ciscoasa(config-webvpn-sso-siteminder)# **max-retry-attempts 4** ciscoasa(config-webvpn-sso-siteminder)#	Configures the number of times the ASA retries a failed SSO authentication attempt before the authentication times out. The default is 3 retry attempts, and the possible range is 1 to 5 attempts. Configures the number of retries to 4.
Step 8	**username-webvpn** **group-policy-webvpn**	If specifying authentication for a user. If specifying authentication for a group.

	Command	Purpose
Step 9	**sso-server value** *value* **Example:** ciscoasa(config)# **username Anyuser attributes** ciscoasa(config-username)# **webvpn** ciscoasa(config-username-webvpn)# **sso-server value** *value* ciscoasa(config-username-webvpn)#	Specifies the SSO authentication for either a group or a user. Assigns the SSO server named Example to the user named Anyuser.
Step 10	**test sso-server** *server* **username** *username* **Example:** ciscoasa# **test sso-server Example username Anyuser** INFO: Attempting authentication request to sso-server Example for user Anyuser INFO: STATUS: Success ciscoasa#	Tests the SSO server configuration. Tests the SSO server named Example using the username Anyuser.

Adding the Cisco Authentication Scheme to SiteMinder

In addition to configuring the ASA for SSO with SiteMinder, you must also configure your CA SiteMinder policy server with the Cisco authentication scheme, a Java plug-in you download from the Cisco website.

Prerequisites

Configuring the SiteMinder policy server requires experience with SiteMinder.

DETAILED STEPS

This section presents general tasks, not a complete procedure.

Step 1 With the SiteMinder Administration utility, create a custom authentication scheme, being sure to use the following specific arguments:

- In the Library field, enter **smjavaapi**.
- In the Secret field, enter the same secret configured on the ASA.

  You configure the secret on the ASA using the **policy-server-secret** command at the command-line interface.
- In the Parameter field, enter **CiscoAuthApi**.

Step 2 Using your Cisco.com login, download the file **cisco_vpn_auth.jar** from http://www.cisco.com/cisco/software/navigator.html and copy it to the default library directory for the SiteMinder server. This .jar file is also available on the Cisco ASA CD.

Configuring SSO Authentication Using SAML Browser Post Profile

This section describes configuring the ASA to support Security Assertion Markup Language (SAML), Version 1.1 POST profile Single Sign-On (SSO) for authorized users.

After a session is initiated, the ASA authenticates the user against a configured AAA method. Next, the ASA (the asserting party) generates an assertion to the relying party, the consumer URL service provided by the SAML server. If the SAML exchange succeeds, the user is allowed access to the protected resource.

Prerequisites

To configure SSO with an SAML Browser Post Profile, you must perform the following tasks:

- Specify the SSO server with the **sso-server** command.
- Specify the URL of the SSO server for authentication requests (the **assertion-consumer-url** command)
- Specify the ASA hostname as the component issuing the authentication request (the **issuer** command)
- Specify the trustpoint certificates use for signing SAML Post Profile assertions (the **trustpoint** command)

Optionally, in addition to these required tasks, you can do the following configuration tasks:

- Configure the authentication request timeout (the **request-timeout** command)
- Configure the number of authentication request retries (the **max-retry-attempts** command)

Restrictions

- SAML SSO is supported only for Clientless SSL VPN sessions.
- The ASA currently supports only the Browser Post Profile type of SAML SSO Server.
- The SAML Browser Artifact method of exchanging assertions is not supported.

DETAILED STEPS

This section presents specific steps for configuring the ASA to support SSO authentication with SAML-V1.1-POST Profile.

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**sso-server type** *type* **Example:** hostname(config)# **webvpn** hostname(config-webvpn)# **sso-server sample type SAML-V1.1-post** ciscoasa(config-webvpn-sso-saml)#	Creates an SSO server. Creates an SSO server named Sample of type SAML-V1.1-POST.
Step 3	**sso saml**	Switches to Clientless SSL VPN sso-saml configuration mode.

	Command	Purpose
Step 4	**assertion-consumer-url** *url* **Example:** ciscoasa(config-webvpn-sso-saml)# **assertion-consumer-url http://www.example.com/webvpn** ciscoasa(config-webvpn-sso-saml)#	Specifies the authentication URL of the SSO server. Sends authentication requests to the URL http://www.Example.com/webvpn.
Step 5	**issuer** *string* **Example:** ciscoasa(config-webvpn-sso-saml)# **issuer myasa** ciscoasa(config-webvpn-sso-saml)#	Identifies the ASA itself when it generates assertions. Typically, this issuer name is the hostname for the ASA.
Step 6	**trust-point** ciscoasa(config-webvpn-sso-saml)# **trust-point mytrustpoint**	Specifies the identification certificate for signing the assertion.
Step 7	(Optional) **request-timeout** **Example:** ciscoasa(config-webvpn-sso-saml)# **request-timeout 8** ciscoasa(config-webvpn-sso-saml)#	Configures the number of seconds before a failed SSO authentication attempt times out. Sets the number of seconds before a request times out to 8. The default number of seconds is 5, and the possible range is 1 to 30 seconds.
Step 8	(Optional) **max-retry-attempts** **Example:** ciscoasa(config-webvpn-sso-saml)# **max-retry-attempts 4** ciscoasa(config-webvpn-sso-saml)#	Configures the number of times the ASA retries a failed SSO authentication attempt before the authentication times out. Sets the number of retries to 4. The default is 3 retry attempts, and the possible range is 1 to 5 attempts.
Step 9	webvpn	Switches to Clientless SSL VPN configuration mode.
Step 10	**group-policy-webvpn** **username-webvpn**	If assigning an SSO server to a group policy. If assigning an SSO server to a user policy.
Step 11	**sso-server value** **Example:** ciscoasa(config)# **username Anyuser attributes** ciscoasa(config-username)# **webvpn** ciscoasa(config-username-webvpn)# **sso-server value sample** ciscoasa(config-username-webvpn)#	Specifies SSO authentication for either a group or a user. Assigns the SSO server named Example to the user named Anyuser.
Step 12	**test sso-server** **Example:** ciscoasa# **test sso-server Example username Anyuser** INFO: Attempting authentication request to sso-server sample for user Anyuser INFO: STATUS: Success	(Privileged exec mode) Tests the SSO server configuration. Tests the SSO server Example using the username Anyuser.

Configuring the SAML POST SSO Server

Use the SAML server documentation provided by the server software vendor to configure the SAML server in Relying Party mode.

DETAILED STEPS

Step 1 Configure the SAML server parameters to represent the asserting party (the ASA):

- Recipient consumer URL (same as the assertion consumer URL configured on the ASA)
- Issuer ID, a string, usually the hostname of appliance
- Profile type -Browser Post Profile

Step 2 Configure certificates.

Step 3 Specify that asserting party assertions must be signed.

Step 4 Select how the SAML server identifies the user:

- Subject Name Type is DN
- Subject Name format is uid=<user>

Configuring SSO with the HTTP Form Protocol

This section describes using the HTTP Form protocol for SSO. HTTP Form protocol is an approach to SSO authentication that can also qualify as a AAA method. It provides a secure method for exchanging authentication information between users of Clientless SSL VPN and authenticating Web servers. You can use it in conjunction with other AAA servers such as RADIUS or LDAP servers.**Prerequisites**

To configure SSO with the HTTP protocol correctly, you must have a thorough working knowledge of authentication and HTTP protocol exchanges.

Restrictions

As a common protocol, it is applicable only when the following conditions are met for the Web server application used for authentication:

- The authentication cookie must be set for successful request and not set for unauthorized logons. In this case, ASA cannot distinguish successful from failed authentication.

DETAILED STEPS

The ASA again serves as a proxy for users of Clientless SSL VPN to an authenticating Web server but, in this case, it uses HTTP Form protocol and the POST method for requests. You must configure the ASA to send and receive form data. Figure 19-4 illustrates the following SSO authentication steps:

Step 1 A user of Clientless SSL VPN first enters a username and password to log on to the Clientless SSL VPN server on the ASA.

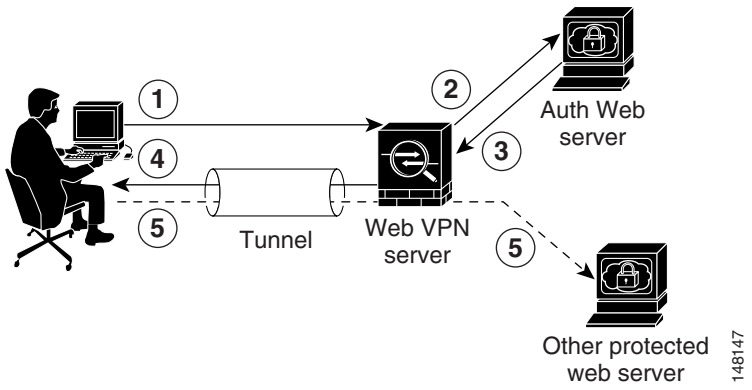
Step 2 The Clientless SSL VPN server acts as a proxy for the user and forwards the form data (username and password) to an authenticating Web server using a POST authentication request.

Step 3 If the authenticating Web server approves the user data, it returns an authentication cookie to the Clientless SSL VPN server where it is stored on behalf of the user.

Step 4 The Clientless SSL VPN server establishes a tunnel to the user.

Step 5 The user can now access other websites within the protected SSO environment without re-entering a username and password.

Figure 19-4 SSO Authentication Using HTTP Forms



While you would expect to configure form parameters that let the ASA include POST data such as the username and password, you initially may not be aware of additional hidden parameters that the Web server requires. Some authentication applications expect hidden data which is neither visible to nor entered by the user. You can, however, discover hidden parameters the authenticating Web server expects by making a direct authentication request to the Web server from your browser without the ASA in the middle acting as a proxy. Analyzing the Web server response using an HTTP header analyzer reveals hidden parameters in a format similar to the following:

```
<param name>=<URL encoded value>&<param name>=<URL encoded>
```

Some hidden parameters are mandatory and some are optional. If the Web server requires data for a hidden parameter, it rejects any authentication POST request that omits that data. Because a header analyzer does not tell you if a hidden parameter is mandatory or not, we recommend that you include all hidden parameters until you determine which are mandatory.

To configure SSO with the HTTP Form protocol, you must perform the following:

- Configure the uniform resource identifier on the authenticating Web server to receive and process the form data (**action-uri**).
- Configure the username parameter (**user-parameter**).
- Configure the user password parameter (**password-parameter**).

You may also need to do the following tasks depending upon the requirements of authenticating Web server:

- Configure a starting URL if the authenticating Web server requires a pre-login cookie exchange (**start-url**).
- Configure any hidden authentication parameters required by the authenticating Web server (**hidden-parameter**).
- Configure the name of an authentication cookie set by the authenticating Web server (**auth-cookie-name**).

	Command	Purpose
Step 1	**aaa-server-host**	Switches to the aaa-server-host configuration mode.
Step 2	**start-url** **Example:** ciscoasa(config)# **aaa-server testgrp1 protocol http-form** ciscoasa(config)# **aaa-server testgrp1 host 10.0.0.2** ciscoasa(config-aaa-server-host)# **start-url http://example.com/east/Area.do?Page-Grp1** ciscoasa(config-aaa-server-host)#	If the authenticating Web server requires it, specifies the URL from which to retrieve a pre-login cookie from the authenticating Web server. Specifies the authenticating Web server URL http://example.com/east/Area.do?Page-Grp1 in the testgrp1 server group with an IP address of 10.0.0.2.
Step 3	**action-uri** **Example:** http://www.example.com/auth/index.html/appdir/authc/forms/MCOlogin.fcc?TYPE=33554433&REALMOID=06-000a1311-a828-1185-ab41-8333b16a0008&GUID=&SMAUTHREASON=0&METHOD=GET&SMAGENTNAME=SM5FZmjnk3DRNwNjk2KcqVCFbIrNT9%2bJ0H0KPshFtg6rB1UV2PxkHqLw%3d%3d&TARGET=https%3A%2F%2Fauth.example.com To specify this action URI, enter the following commands: ciscoasa(config-aaa-server-host)# **action-uri http://www.example.com/auth/index.htm** ciscoasa(config-aaa-server-host)# **action-uri l/appdir/authc/forms/MCOlogin.fcc?TYP** ciscoasa(config-aaa-server-host)# **action-uri 554433&REALMOID=06-000a1311-a828-1185** ciscoasa(config-aaa-server-host)# **action-uri -ab41-8333b16a0008&GUID=&SMAUTHREASON** ciscoasa(config-aaa-server-host)# **action-uri =0&METHOD=GET&SMAGENTNAME=SM5FZmjnk** ciscoasa(config-aaa-server-host)# **action-uri 3DRNwNjk2KcqVCFbIrNT9%2bJ0H0KPshFtg6r** ciscoasa(config-aaa-server-host)# **action-uri B1UV2PxkHqLw%3d%3d&TARGET=https%3A%2F** ciscoasa(config-aaa-server-host)# **action-uri %2Fauth.example.com** ciscoasa(config-aaa-server-host)#	Specifies a URI for an authentication program on the authenticating Web server. A URI can be entered on multiple, sequential lines. The maximum number of characters per line is 255. The maximum number of characters for a complete URI is 2048. You must include the hostname and protocol in the action URI. In this example, these appear at the start of the URI in http://www.example.com.
Step 4	**user-parameter** **Example:** ciscoasa(config-aaa-server-host)# **user-parameter userid** ciscoasa(config-aaa-server-host)#	Configures the **userid** username parameter for the HTTP POST request.
Step 5	**password-parameter** **Example:** ciscoasa(config-aaa-server-host)# **password-parameter user_password** ciscoasa(config-aaa-server-host)#	Configures thew **user_password** user password parameter for the HTTP POST request.

	Command	Purpose
Step 6	**hidden-parameter** **Example:** SMENC=ISO-8859-1&SMLOCALE=US-EN&target=https%3A%2F%2Fwww.example.com%2Femco%2Fappdir%2FAreaRoot.do%3FEMCOPageCode%3DENG&smauthreason=0 To specify this hidden parameter, enter the following commands: ciscoasa(config)# **aaa-server testgrp1 host example.com** ciscoasa(config-aaa-server-host)# **hidden-parameter SMENC=ISO-8859-1&SMLOCALE=US-EN&targe** ciscoasa(config-aaa-server-host)# **hidden-parameter t=https%3A%2F%2Fwww.example.com%2Femc** ciscoasa(config-aaa-server-host)# **hidden-parameter o%2Fappdir%2FAreaRoot.do%3FEMCOPageCo** ciscoasa(config-aaa-server-host)# **hidden-parameter de%3DENG&smauthreason=0** ciscoasa(config-aaa-server-host)#	Specifies hidden parameters for exchange with the authenticating Web server. Shows an example hidden parameter excerpted from a POST request. This hidden parameter includes four form entries and their values, separated by &. The entries and their values are: • SMENC with a value of ISO-8859-1. • SMLOCALE with a value of US-EN. • target with a value of https%3A%2F%2Fwww.example.com%2Femco%2Fappdir%2FAreaRoot.do. • %3FEMCOPageCode%3DENG. • smauthreason with a value of 0.
Step 7	(Optional) **auth-cookie-name** *cookie-name* **Example:** ciscoasa(config-aaa-server-host)# **auth-cookie-name SsoAuthCookie** ciscoasa(config-aaa-server-host)#	Specifies the name for the authentication cookie. Specifies an authentication cookie name of SsoAuthCookie.
Step 8	**tunnel-group general-attributes**	Switches to tunnel-group general-attributes configuration mode.
Step 9	**authentication-server-group** **Example:** hostname(config)# **tunnel-group testgroup general-attributes** hostname(config-tunnel-general)**#authentication-server-group testgrp1**	Configures a tunnel-group to use the SSO server configured in the previous steps. Configures the tunnel-group named /testgroup/ to use the SSO server(s) named /testgrp1/”.
Step 10	**aaa-server-host**	Switches to AAA server host configuration mode.

	Command	Purpose
Step 11	**hidden-parameter** **Example:** SMENC=ISO-8859-1&SMLOCALE=US-EN&target=https%3A%2F%2Fwww.example.com%2Femco%2Fappdir%2FAreaRoot.do%3FEMCOPageCode%3DENG&smauthreason=0 To specify this hidden parameter, enter the following commands: ciscoasa(config)# **aaa-server testgrp1 host example.com** ciscoasa(config-aaa-server-host)# **hidden-parameter SMENC=ISO-8859-1&SMLOCALE=US-EN&targe** ciscoasa(config-aaa-server-host)# **hidden-parameter t=https%3A%2F%2Fwww.example.com%2Femc** ciscoasa(config-aaa-server-host)# **hidden-parameter o%2Fappdir%2FAreaRoot.do%3FEMCOPageCo** ciscoasa(config-aaa-server-host)# **hidden-parameter de%3DENG&smauthreason=0** ciscoasa(config-aaa-server-host)#	Specifies hidden parameters for exchange with the authenticating Web server. Shows an example hidden parameter excerpted from a POST request. This hidden parameter includes four form entries and their values, separated by &. The entries and their values are: • SMENC with a value of ISO-8859-1. • SMLOCALE with a value of US-EN. • target with a value of https%3A%2F%2Fwww.example.com%2Femco%2Fappdir%2FAreaRoot.do. • %3FEMCOPageCode%3DENG. • smauthreason with a value of 0.
Step 12	(Optional) **auth-cookie-name** *cookie-name* **Example:** ciscoasa(config-aaa-server-host)# **auth-cookie-name SsoAuthCookie** ciscoasa(config-aaa-server-host)#	Specifies the name for the authentication cookie. Specifies an authentication cookie name of SsoAuthCookie.
Step 13	**tunnel-group general-attributes**	Switches to tunnel-group general-attributes mode.
Step 14	**authentication-server-group** *group* **Example:** hostname(config)# **tunnel-group testgroup general-attributes** hostname(config-tunnel-general)**#authentication-server-group testgrp1**	Configures a tunnel-group to use the SSO server configured in the previous steps. Configures a tunnel-group named /testgroup/ to use the SSO server(s) named /testgrp1/”.

Gathering HTTP Form Data

This section presents the steps for discovering and gathering necessary HTTP Form data. If you do not know what parameters the authenticating Web server requires, you can gather parameter data by analyzing an authentication exchange.

Prerequisites

These steps require a browser and an HTTP header analyzer.

DETAILED STEPS

Step 1 Start your browser and HTTP header analyzer, and connect directly to the Web server login page without going through the ASA.

Step 2 After the Web server login page has loaded in your browser, examine the login sequence to determine if a cookie is being set during the exchange. If the Web server has loaded a cookie with the login page, configure this login page URL as the *start-URL*.

Step 3 Enter the username and password to log on to the Web server, and press **Enter**. This action generates the authentication POST request that you examine using the HTTP header analyzer.

An example POST request—with host HTTP header and body—follows:

```
POST
/emco/myemco/authc/forms/MCOlogin.fcc?TYPE=33554433&REALMOID=06-000430e1-7443-125c-ac05
-83846dc90034&GUID=&SMAUTHREASON=0&METHOD=GET&SMAGENTNAME=$SM$5FZmjnk3DRNwNjk2KcqVCFbIr
NT9%2bJ0H0KPshFtg6rB1UV2PxkHqLw%3d%3d&TARGET=https%3A%2F%2Fwww.example.com%2Femco%2Fmye
mco%2FHTTP/1.1

Host: www.example.com

(BODY)

SMENC=ISO-8859-1&SMLOCALE=US-EN&USERID=Anyuser&USER_PASSWORD=XXXXXX&target=https%3A%2F%
2Fwww.example.com%2Femco%2Fmyemco%2F&smauthreason=0
```

Step 4 Examine the POST request and copy the protocol, host, and the complete URL to configure the action-uri parameter.

Step 5 Examine the POST request body and copy the following:

a. Username parameter. In the preceding example, this parameter is *USERID*, not the value *anyuser*.

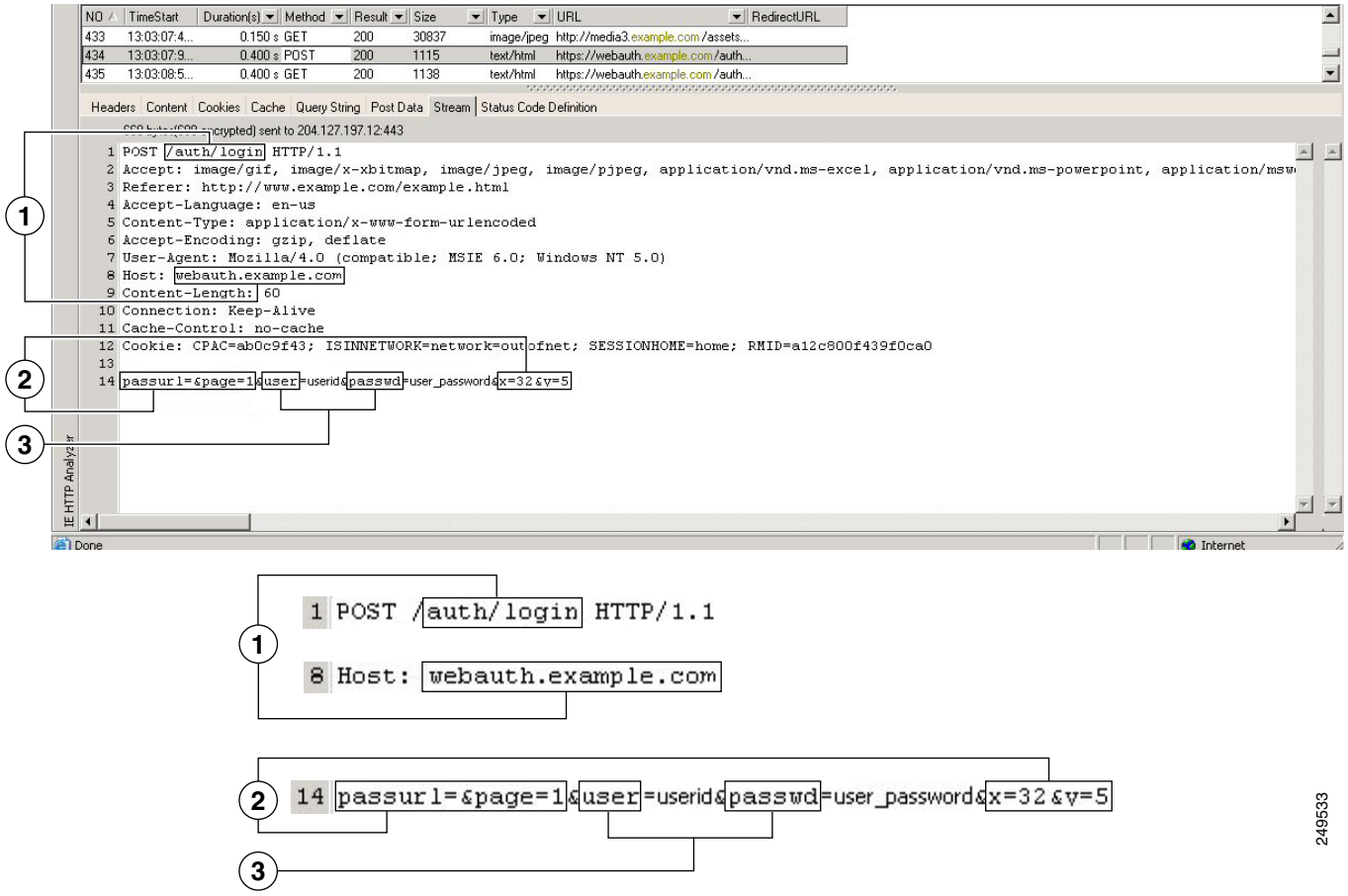
b. Password parameter. In the preceding example, this parameter is *USER_PASSWORD*.

c. Hidden parameter. This parameter is everything in the POST body except the username and password parameters. In the preceding example, the hidden parameter is:

SMENC=ISO-8859-1&SMLOCALE=US-EN&target=https%3A%2F%2Fwww.example.com%2Femco%2Fmyemco%2F&smauthreason=0

Figure 19-5 highlights the action URI, hidden, username and password parameters within sample output from an HTTP analyzer. This is only an example; output varies widely across different websites.

Figure 19-5 Action-uri, hidden, username and password parameters



1	Action URI parameter
2	Hidden parameters
3	Username and password parameters

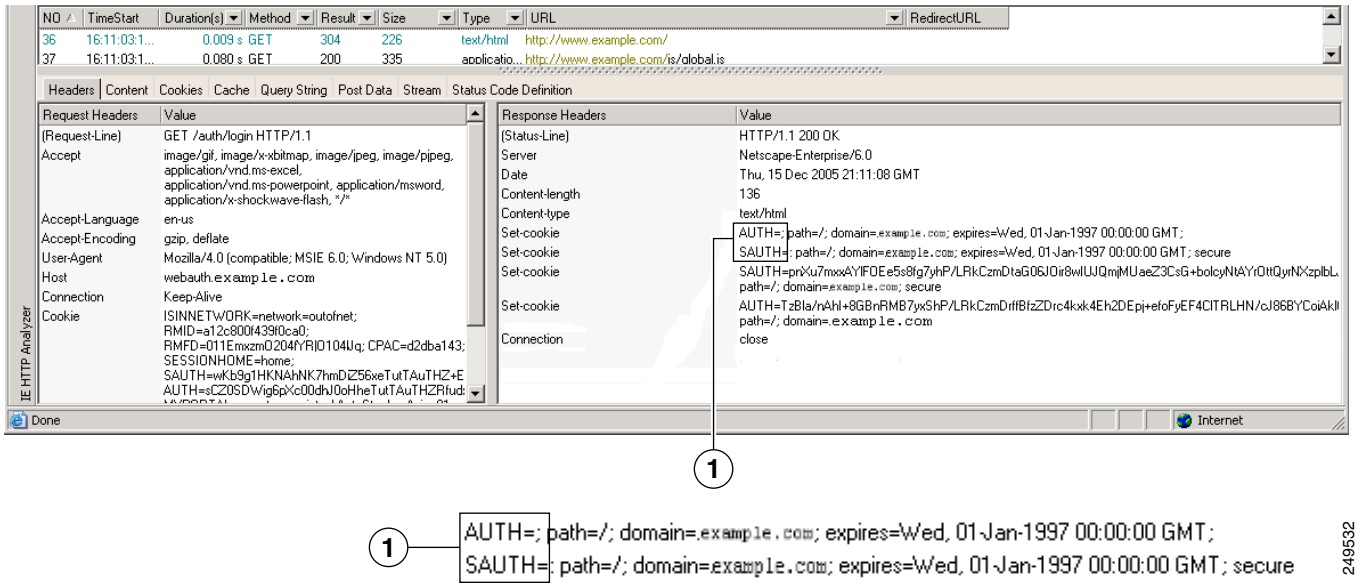
Step 6 If you successfully log on to the Web server, examine the server response with the HTTP header analyzer to locate the name of the session cookie set by the server in your browser. This is the **auth-cookie-name** parameter.

In the following server response header, the name of the session cookie is SMSESSION. You just need the name, not the value.

```
Set-Cookie:
SMSESSION=yN4Yp5hHVNDgs4FT8dn7+Rwev41hsE49XlKc+1twie0gqnjbhkTkUnR8XWP3hvDH6PZP
bHIHtWLDKTa8ngDB/lbYTjIxrbDx8WPWwaG3CxVa3adOxHFR8yjD55GevK3ZF4ujgU1lhO6fta0dSS
OSepWvnsCb7IFxCw+MGiw0o88uHa2t4l+SillqfJvcpuXfiIAO06D/gtDF40Ow5YKHEl2KhDEvv+yQ
zxwfEz2cl7Ef5iMr8LgGcDK7qvMcvrgUqx68JQOK2+RSwtHQ15bCZmsDU5vQVCvSQWC8OMHNGwpS25
3XwRLvd/h6S/tM0k98QMv+i3N8oOdj1V7flBqecH7+kVrU01F6oFzr0zM1kMyLr5HhlVDh7B0k9wp0
dUFZiAzaf43jupD5f6CEkuLeudYW1xgNzsR8eqtPK6t1gFJyOn0s7QdNQ7q9knsPJsekRAH9hrLBhW
BLTU/3B1QS94wEGD2YTuiW36TiP14hYwOlCAYRj2/bY3+lYzVu7EmzMQ+UefYxh4cF2gYD8RZL2Rwm
P9JV5148I3XBFPNUw/3V5jf7nRuLr/CdfK3OO8+Pa3V6/nNhokErSgyxjzMd88DVzM41LxxaUDhbcm
koHT9ImzBvKzJX0J+o7FoUDFOxEdIqlAN4GNqk49cpi2sXDbIarALp6Bl3+tbB4MlHGH+0CPscZXqo
i/kon9YmGauHyRs+0m6wthdlAmCnvlJCDfDoXtn8DpabgiW6VDTrvl3SGPyQtUv7Wdahuq5SxbUzjY
2JxQnrUtwB977NCzYu2sOtN+dsEReWJ6ueyJBbMzKyzUB4L3i5uSYN50B4PCv1w5KdRKa5p3N0Nfq6
RM6dfipMEJw0Ny1sZ7ohz3fbvQ/YZ7lw/k7ods/8VbaR15ivkE8dSCzuf/AInHtCzuQ6wApzEp9CUo
G8/dapWriHjNoi4llJOgCst33wEhxFxcWy2UWxs4EZSjsI5GyBnefSQTPVfma5dc/emWor9vWr0HnT
QaHP5rg5dTNqunkDEdMIHfbeP3F90cZejVzihM6igiS6P/CEJAjE;Domain=.example.com;Path=
/
```

Figure 19-6 shows an example of authorization cookies in HTTP analyzer output. This is only an example; output varies widely across different websites.

Figure 19-6 Authorization Cookies in Sample HTTP Analyzer Output



1	Authorization cookies

Step 7 In some cases, the server may set the same cookie regardless of whether the authentication was successful or not, and such a cookie is unacceptable for SSO purposes. To confirm that the cookies are different, repeat Step 1 through Step 6 using invalid login credentials and then compare the “failure” cookie with the “success” cookie. You now have the necessary parameter data to configure the ASA for SSO with HTTP Form protocol.

Configuring SSO for Plug-ins

Plug-ins support single sign-on (SSO). They use the same credentials (username and password) entered to authenticate the Clientless SSL VPN session. Because the plug-ins do not support macro substitution, you do not have the option to perform SSO on different fields, such as the internal domain password or the attribute on a RADIUS or LDAP server.

To configure SSO support for a plug-in, you install the plug-in and add a bookmark entry to display a link to the server, specifying SSO support using the csco_sso=1 parameter. The following examples show plug-in bookmarks enabled for SSO:

```
ssh://ssh-server/?cisco_sso=1
rdp://rdp-server/?Parameter1=value&Parameter2=value&csco_sso=1
```

Configuring SSO with Macro Substitution

This section describes using macro substitution for SSO. Configuring SSO with macro substitution allows for you to inject certain variables into bookmarks to substitute for dynamic values.

Note Smart tunnel bookmarks support auto-sign-on but not variable substitution. For example, a SharePoint bookmark configured for smart tunnel uses the same username and password credentials to log on to the application as the credentials used to log on to Clientless SSL VPN. You can use variable substitutions and auto sign-on simultaneously or separately.

You can now use bookmarks with macro substitutions for auto sign-on on some Web pages. The former POST plug-in approach was created so that administrators could specify a POST bookmark with sign-on macros and receive a kick-off page to load prior to posting the POST request. This POST plug-in approach eliminated those requests that required the presence of cookies or other header items. Now an an administrator determines the pre-load page and URL, which specifies where the post login request is sent. A pre-load page enables an endpoint browser to fetch certain information that is sent along to the webserver or Web application rather than just using a POST request with credentials.

The following variables (or macros) allow for substitutions in bookmarks and forms-based HTTP POST operations:

- CSCO_WEBVPN_USERNAME—User login ID
- CSCO_WEBVPN_PASSWORD—User login password
- CSCO_WEBVPN_INTERNAL_PASSWORD—User internal (or domain) password. This cached credential is not authenticated against a AAA server. When you enter this value, the security appliance uses it as the password for auto sign-on, instead of the password/primary password value.

  Note You cannot use any of these three variables in GET-based http(s) bookmarks. Only POST-based http(s) and cifs bookmarks can use these variables.

- CSCO_WEBVPN_CONNECTION_PROFILE—User login group drop-down (connection profile alias)

- CSCO_WEBVPN_MACRO1—Set with the RADIUS-LDAP Vendor Specific Attribute (VSA). If you are mapping from LDAP with an ldap-attribute-map command, use the WebVPN-Macro-Substitution-Value1 Cisco attribute for this macro. See the Active Directory ldap-attribute-mapping examples at http://www.cisco.com/en/US/docs/security/asa/asa83/configuration/guide/ref_extserver.html#wp1572118.

  The CSCO_WEBVPN_MACRO1 macro substitution with RADIUS is performed by VSA#223 (see Table 19-1).

Table 19-1 VSA#223

WebVPN-Macro-Value1	Y	223	String	Single	Unbounded
WebVPN-Macro-Value2	Y	224	String	Single	Unbounded

  A value such as www.cisco.com/email dynamically populates a bookmark on the Clientless SSL VPN portal, such as https://CSCO_WEBVPN_MACRO1 or https://CSCO_WEBVPN_MACRO2 for the particular DAP or group policy.

- CSCO_WEBVPN_MACRO2—set with RADIUS-LDAP Vendor Specific Attribute (VSA). If you are mapping from LDAP with an ldap-attribute-map command, use the WebVPN-Macro-Substitution-Value2 Cisco attribute for this macro. See the Active Directory ldap-attribute-mapping examples at http://www.cisco.com/en/US/docs/security/asa/asa83/configuration/guide/ref_extserver.html#wp1572118.

  The CSCO_WEBVPN_MACRO2 macro substitution with RADIUS is performed by VSA#224 (see Table 19-1).

Each time Clientless SSL VPN recognizes one of these six strings in an end-user request (in the form of a bookmark or Post Form), it replaces the string with the user-specified value and then passes the request to a remote server.

If the lookup of the username and password fails on the ASA, an empty string is substituted, and the behavior converts back as if no auto sign-in is available.

Requiring Usernames and Passwords

Depending on your network, during a remote session users may have to log on to any or all of the following: the computer itself, an Internet service provider, Clientless SSL VPN, mail or file servers, or corporate applications. Users may have to authenticate in many different contexts, requiring different information, such as a unique username, password, or PIN.

Table 19-2 lists the type of usernames and passwords that Clientless SSL VPN users may need to know.

Table 19-2 Usernames and Passwords to Give to Users of Clientless SSL VPN Sessions

Login Username/ Password Type	Purpose	Entered When
Computer	Access the computer	Starting the computer
Internet Service Provider	Access the Internet	Connecting to an Internet service provider
Clientless SSL VPN	Access remote network	Starting Clientless SSL VPN

Table 19-2 Usernames and Passwords to Give to Users of Clientless SSL VPN Sessions

Login Username/ Password Type	Purpose	Entered When
File Server	Access remote file server	Using the Clientless SSL VPN file browsing feature to access a remote file server
Corporate Application Login	Access firewall-protected internal server	Using the Clientless SSL VPN Web browsing feature to access an internal protected website
Mail Server	Access remote mail server via Clientless SSL VPN	Sending or receiving email messages

Communicating Security Tips

Advise users to always click the logout icon on the toolbar to close the Clientless SSL VPN session. (Closing the browser window does not close the session.)

Clientless SSL VPN ensures the security of data transmission between the remote PC or workstation and the ASA on the corporate network. Advise users that using Clientless SSL VPN does not ensure that communication with every site is secure. If a user then accesses a non-HTTPS Web resource (located on the Internet or on the internal network), the communication from the corporate ASA to the destination Web server is not private because it is not encrypted.

"Clientless SSL VPN Security Precautions" on page 1 addresses an additional tip to communicate with users, depending on the steps you follow within that section.

Configuring Remote Systems to Use Clientless SSL VPN Features

This section describes how to set up remote systems to use Clientless SSL VPN and includes the following topics:

You may configure user accounts differently and different Clientless SSL VPN features can be available to each user.

Starting Clientless SSL VPN

You can connect to the internet using any supported connection including:

- Home DSL, cable, or dial-ups.
- Public kiosks.
- Hotel hotspots.
- Airport wireless nodes.
- Internet cafes.

Note See the *Supported VPN Platforms, Cisco ASA 5500 Series* for the list of Web browsers supported by Clientless SSL VPN.

Prerequisites

- Cookies must be enabled on the browser in order to access applications via port forwarding.
- You must have a URL for Clientless SSL VPN. The URL must be an https address in the following form: https://*address*, where *address* is the IP address or DNS hostname of an interface of the ASA (or load balancing cluster) on which SSL VPN is enabled. For example, https://cisco.example.com.
- You must have a Clientless SSL VPN username and password.

Restrictions

- Clientless SSL VPN supports local printing, but it does not support printing through the VPN to a printer on the corporate network.

Using the Clientless SSL VPN Floating Toolbar

A floating toolbar is available to simplify the use of Clientless SSL VPN. The toolbar lets you enter URLs, browse file locations, and choose preconfigured Web connections without interfering with the main browser window.

The floating toolbar represents the current Clientless SSL VPN session. If you click the **Close** button, the ASA prompts you to close the Clientless SSL VPN session.

Tip To paste text into a text field, use **Ctrl-V**. (Right-clicking is switched off on the toolbar displayed during the Clientless SSL VPN session.)

Restrictions

If you configure your browser to block popups, the floating toolbar cannot display.

Browsing the Web

Using Clientless SSL VPN does not ensure that communication with every site is secure. See Communicating Security Tips.

The look and feel of Web browsing with Clientless SSL VPN may be different from what users are accustomed to. For example:

- The title bar for Clientless SSL VPN appears above each Web page.
- You access websites by:
  - Entering the URL in the **Enter Web Address** field on the Clientless SSL VPN Home page
  - Clicking on a preconfigured website link on the Clientless SSL VPN Home page
  - Clicking a link on a webpage accessed via one of the previous two methods

Also, depending on how you configured a particular account, it may be that:

- Some websites are blocked
- Only the websites that appear as links on the Clientless SSL VPN Home page are available

Prerequisites

You need the username and password for protected websites.

Restrictions

Also, depending on how you configured a particular account, it may be that:

- Some websites are blocked
- Only the websites that appear as links on the Clientless SSL VPN Home page are available

Browsing the Network (File Management)

Users may not be familiar with how to locate their files through your organization network.

Note Do not interrupt the **Copy File to Server** command or navigate to a different screen while the copying is in progress. Interrupting the operation can cause an incomplete file to be saved on the server.

Prerequisites

- You must configure file permissions for shared remote access.
- You must have the server names and passwords for protected file servers.
- You must have the domain, workgroup, and server names where folders and files reside.

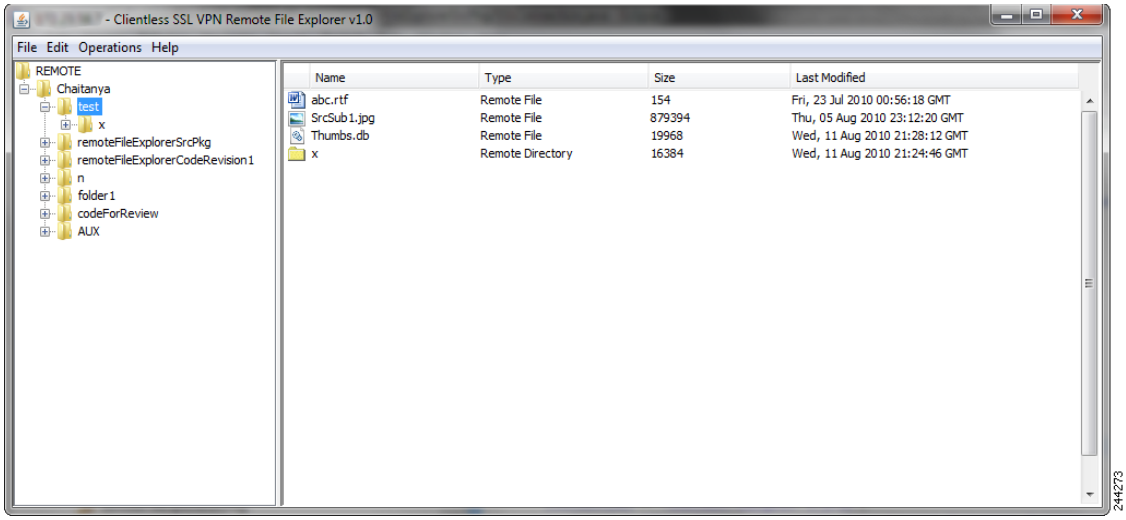
Restrictions

Only shared folders and files are accessible via Clientless SSL VPN.

Using the Remote File Explorer

The Remote File Explorer provides the user with a way to browse the corporate network from their Web browser. When the users clicks the Remote File System icon on the Cisco SSL VPN portal page, an applet is launched on the user's system displaying the remote file system in a tree and folder view.

Figure 19-7 Clientless SSL VPN Remote File Explorer



The browser enables the user to:

- Browse the remote file system.
- Rename files.
- Move or copy files within the remote file system and between the remote and local file systems.
- Perform bulk uploads and downloads of files.

Note This functionality requires that the Oracle Java Runtime Environment (JRE) 1.4 or later is installed on the user's machine and that Java is enabled in the Web browser. Launching remote files requires JRE 1.6 or later.

Renaming a File or Folder

To rename a file or folder:

Step 1 Click the file or folder to be renamed.

Step 2 Select **Edit** > **Rename**.

Step 3 When prompted, enter the new name in the dialog.

Step 4 Click **OK** to rename the file or folder. Alternative, click **Cancel** to leave the name unchanged.

Moving or Copying Files or Folders on the Remote Server

To move or copy a file or folder on the remote server:

Step 1 Navigate to the source folder containing the file or folder to be moved or copied.

Step 2 Click the file or folder.

Step 3 To copy the file select **Edit** > **Copy**. Alternatively, to move the file select **Edit** > **Cut**.

Step 4 Navigate to the destination folder.

Step 5 Select **Edit** > **Paste**.

Copying Files from the Local System Drive to the Remote Folder

You can copy files between the local file system and the remote file system by dragging and dropping them between the right pane of the Remote File Browser and your local file manager application.

Uploading and Downloading Files

You can download a file by clicking it in the browser, selecting **Operations** > **Download**, and providing a location and name to save the file in the **Save** dialog.

You can upload a file by clicking the destination folder, selecting **Operations** > **Upload**, and providing the location and name of the file in the **Open** dialog,

This functionality has the following restrictions:

- The user cannot view sub-folders for which they are not permitted access.
- Files that the user is not permitted to access cannot be moved or copied, even though they are displayed in the browser.
- The maximum depth of nested folders is 32.
- The tree view does not support drag and drop copying.
- When moving files between multiple instances of the Remote File Explorer, all instances must be exploring the same server (root share).
- The Remote File Explorer can display a maximum of 1500 files and folders in a single folder. If a folder exceeds this limit the folder cannot be displayed.

Using Port Forwarding

Note Users should always close the Application Access window when they finish using applications by clicking the **Close** icon. Failure to quit the window properly can cause Application Access or the applications themselves to be switched off. See the "Recovering from Hosts File Errors When Using Application Access" section on page 22-1 for details.

Prerequisites

- On Mac OS X, only the Safari browser supports this feature.
- You must have client applications installed.
- You must have Cookies enabled on the browser.
- You must have administrator access on the PC if you use DNS names to specify servers, because modifying the hosts file requires it.
- You must have Oracle Java Runtime Environment (JRE) version 1.4.x and 1.5.x installed.

If JRE is not installed, a pop-up window displays, directing users to a site where it is available. On rare occasions, the port forwarding applet fails with Java exception errors. If this happens, do the following:

a. Clear the browser cache and close the browser.
b. Verify that no Java icons are in the computer task bar.
c. Close all instances of Java.
d. Establish a Clientless SSL VPN session and launch the port forwarding Java applet.

- You must have JavaScript enabled on the browser. By default, it is enabled.
- If necessary, you must configure client applications.

Note The Microsoft Outlook client does not require this configuration step. All non-Windows client applications require configuration. To determine if configuration is necessary for a Windows application, check the value of the Remote Server field. If the Remote Server field contains the server hostname, you do not need to configure the client application. If the Remote Server field contains an IP address, you must configure the client application.

Restrictions

Because this feature requires installing Oracle Java Runtime Environment (JRE) and configuring the local clients, and because doing so requires administrator permissions on the local system or full control of C:\windows\System32\drivers\etc, it is unlikely that users will be able to use applications when they connect from public remote systems.

DETAILED STEPS

To configure the client application, use the server's locally mapped IP address and port number. To find this information:

1. Start a Clientless SSL VPN session and click the **Application Access** link on the Home page. The Application Access window appears.
2. In the Name column, find the name of the server to use, then identify its corresponding client IP address and port number (in the Local column).
3. Use this IP address and port number to configure the client application. Configuration steps vary for each client application.

Note Clicking a URL (such as one in an -email message) in an application running over a Clientless SSL VPN session does not open the site over that session. To open a site over the session, paste the URL into the Enter Clientless SSL VPN (URL) Address field.

Using email Via Port Forwarding

To use email, start Application Access from the Clientless SSL VPN home page. The mail client is then available for use.

Note If you are using an IMAP client and you lose your mail server connection or are unable to make a new connection, close the IMAP application and restart Clientless SSL VPN.

Prerequisites

You must fulfill requirements for application access and other mail clients.

Restrictions

We have tested Microsoft Outlook Express versions 5.5 and 6.0.

Clientless SSL VPN should support other SMTPS, POP3S, or IMAP4S email programs via port forwarding, such as Lotus Notes and Eudora, but we have not verified them.

Using email Via Web Access

The following email applications are supported:

- Microsoft Outlook Web App to Exchange Server 2010.

  OWA requires Internet Explorer 7 or later, or Firefox 3.01 or later.
- Microsoft Outlook Web Access to Exchange Server 2007, 2003, and 2000.

  For best results, use OWA on Internet Explorer 8.x or later, or Firefox 8.x.
- Lotus iNotes

Prerequisites

You must have the web-based email product installed.

Restrictions

Other web-based email applications should also work, but we have not verified them.

Using email Via email Proxy

The following legacy email applications are supported:

- Microsoft Outlook 2000 and 2002
- Microsoft Outlook Express 5.5 and 6.0

See the instructions and examples for your mail application in "Using Email over Clientless SSL VPN" section on page 16-14.

Prerequisites

- You must have the SSL-enabled mail application installed.
- Do not set the ASA SSL version to TLSv1 Only. Outlook and Outlook Express do not support TLS.
- You must have your mail application properly configured.

Restrictions

Other SSL-enabled clients should also work, but we have not verified them.

Using Smart Tunnel

Administration privileges are not required to use Smart Tunnel.

Note Java is not automatically downloaded for you as in port forwarder.

Prerequisites

- Smart tunnel requires either ActiveX or JRE (1.4x and 1.5x) on Windows and Java Web Start on Mac OS X.
- You must ensure cookies enabled on the browser.
- You must ensure JavaScript is enabled on the browser.

Restrictions

- Mac OS X does not support a front-side proxy.
- Supports only the operating systems and browsers specified in "Configuring Smart Tunnel Access" section on page 17-4.
- Only TCP socket-based applications are supported.