Cisco ASA Series VPN CLI Configuration Guide

Software Version 9.1
For the ASA 5505, ASA 5510, ASA 5520, ASA 5540, ASA 5550, ASA 5512-X, ASA 5515-X, ASA 5525-X, ASA 5545-X, ASA 5555-X, ASA 5580, ASA 5585-X, and the ASA Services Module

Released: December 3, 2012
Updated: March 31, 2014

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Text Part Number: N/A, Online only

Cisco ASA Series VPN CLI Configuration Guide

CONTENTS

About This Guide

This preface introduces *Cisco ASA Series VPN CLI Configuration Guide* and includes the following sections:

Document Objectives

The purpose of this guide is to help you configure VPN on the ASA using the command-line interface. This guide does not cover every feature, but describes only the most common configuration scenarios.

You can also configure and monitor the ASA by using ASDM, a web-based GUI application. ASDM includes configuration wizards to guide you through some common configuration scenarios, and online help for less common scenarios.

This guide applies to the Cisco ASA series. Throughout this guide, the term “ASA” applies generically to supported models, unless specified otherwise.

Related Documentation

For more information, see *Navigating the Cisco ASA Series Documentation* at http://www.cisco.com/go/asadocs.

Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands and keywords and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.

[ ]	Elements in square brackets are optional.
{x | y | z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
`courier` font	Terminal sessions and information the system displays appear in `courier` font.
`courier bold` font	Commands and keywords and user-entered text appear in **`bold courier`** font.
`courier italic` font	Arguments for which you supply values are in *`courier italic`* font.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means *reader take note*.

Tip Means *the following information will help you solve a problem.*

Caution Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.

Obtain Documentation and Submit a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation*.

To receive new and revised Cisco technical content directly to your desktop, you can subscribe to the What's New in Cisco Product Documentation RSS feed. The RSS feeds are a free service.

PART 1

Configuring Site-to-Site and Client VPN

Configuring IPsec and ISAKMP

This chapter describes how to configure Internet Protocol Security (IPsec) and the Internet Security Association and Key Management Protocol (ISAKMP) standards to build Virtual Private Networks (VPNs). It includes the following sections:

Information About Tunneling, IPsec, and ISAKMP

Tunneling makes it possible to use a public TCP/IP network, such as the Internet, to create secure connections between remote users and a private corporate network. Each secure connection is called a tunnel.

The ASA uses the ISAKMP and IPsec tunneling standards to build and manage tunnels. ISAKMP and IPsec accomplish the following:

- Negotiate tunnel parameters
- Establish tunnels
- Authenticate users and data
- Manage security keys
- Encrypt and decrypt data
- Manage data transfer across the tunnel
- Manage data transfer inbound and outbound as a tunnel endpoint or router

The ASA functions as a bidirectional tunnel endpoint. It can receive plain packets from the private network, encapsulate them, create a tunnel, and send them to the other end of the tunnel where they are unencapsulated and sent to their final destination. It can also receive encapsulated packets from the public network, unencapsulate them, and send them to their final destination on the private network.

IPsec Overview

The ASA uses IPsec for LAN-to-LAN VPN connections and provides the option of using IPsec for client-to-LAN VPN connections. In IPsec terminology, a *peer* is a remote-access client or another secure gateway. For both connection types, the ASA supports only Cisco peers. Because we adhere to VPN industry standards, ASAs can work with other vendors' peers; however, we do not support them.

During tunnel establishment, the two peers negotiate security associations that govern authentication, encryption, encapsulation, and key management. These negotiations involve two phases: first, to establish the tunnel (the IKE SA) and second, to govern traffic within the tunnel (the IPsec SA).

A LAN-to-LAN VPN connects networks in different geographic locations. In IPsec LAN-to-LAN connections, the ASA can function as initiator or responder. In IPsec client-to-LAN connections, the ASA functions only as responder. Initiators propose SAs; responders accept, reject, or make counter-proposals—all in accordance with configured SA parameters. To establish a connection, both entities must agree on the SAs.

Configuration for site to site tasks is performed in both single context mode and multiple context mode.

Note Multiple context mode only applies to IKEv2 and IKEv1 site to site and does not apply to AnyConnect, clientless SSL VPN, the legacy Cisco VPN client, the Apple native VPN client, the Microsoft native VPN client, or cTCP for IKEv1 IPsec.

ISAKMP and IKE Overview

ISAKMP is the negotiation protocol that lets two hosts agree on how to build an IPsec security association (SA). It provides a common framework for agreeing on the format of SA attributes. This security association includes negotiating with the peer about the SA and modifying or deleting the SA. ISAKMP separates negotiation into two phases: Phase 1 and Phase 2. Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages. Phase 2 creates the tunnel that protects data.

IKE uses ISAKMP to set up the SA for IPsec to use. IKE creates the cryptographic keys used to authenticate peers.

The ASA supports IKEv1 for connections from the legacy Cisco VPN client, and IKEv2 for the AnyConnect VPN client.

To set the terms of the ISAKMP negotiations, you create an IKE policy, which includes the following:

- The authentication type required of the IKEv1 peer, either RSA signature using certificates or preshared key (PSK).
- An encryption method to protect the data and ensure privacy.
- A Hashed Message Authentication Codes (HMAC) method to ensure the identity of the sender, and to ensure that the message has not been modified in transit.
- A Diffie-Hellman group to determine the strength of the encryption-key-determination algorithm. The ASA uses this algorithm to derive the encryption and hash keys.

- For IKEv2, a separate pseudo-random function (PRF) used as the algorithm to derive keying material and hashing operations required for the IKEv2 tunnel encryption and so on.
- A limit to the time the ASA uses an encryption key before replacing it.

With IKEv1 policies, you set one value for each parameter. For IKEv2, you can configure multiple encryption and authentication types, and multiple integrity algorithms for a single policy. The ASA orders the settings from the most secure to the least secure and negotiates with the peer using that order. This ordering allows you to potentially send a single proposal to convey all the allowed transforms instead of sending each allowed combination as with IKEv1.

Licensing Requirements for Remote Access IPsec VPNs

The following table shows the licensing requirements for this feature:

Note This feature is not available on No Payload Encryption models.

Model	License Requirement[1]
ASA 5505	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license and Security Plus license: 2 sessions. *Optional permanent or time-based licenses: 10 or 25 sessions.* *Shared licenses are not supported.*[2] – AnyConnect Essentials license[3]: 25 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: – Base license: 10 sessions. – Security Plus license: 25 sessions.
ASA 5510	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base and Security Plus license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses*[2]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license and Security Plus license: 250 sessions.

Model	License Requirement[1]
ASA 5520	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 750 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 750 sessions.
ASA 5540	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 2500 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 2500 sessions.
ASA 5550	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.

Model	License Requirement[1]
ASA 5580	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.
ASA 5512-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 250 sessions.
ASA 5515-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 250 sessions.
ASA 5525-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 750 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 750 sessions.

Model	License Requirement[1]
ASA 5545-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 2500 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 2500 sessions.
ASA 5555-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.
ASA 5585-X with SSP-10	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.

Model	License Requirement[1]
ASA 5585-X with SSP-20, -40, and -60	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.
ASA SM	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.

1. The maximum combined VPN sessions of *all* types cannot exceed the maximum sessions shown in this table. For the ASA 5505, the maximum combined sessions is 10 for the Base license, and 25 for the Security Plus license.
2. A shared license lets the security appliance act as a shared license server for multiple client security appliances. The shared license pool is large, but the maximum number of sessions used by each individual security appliance cannot exceed the maximum number listed for permanent licenses.
3. The AnyConnect Essentials license enables AnyConnect VPN client access to the security appliance. This license does not support browser-based SSL VPN access or Cisco Secure Desktop. For these features, activate an AnyConnect Premium license instead of the AnyConnect Essentials license.

   Note: With the AnyConnect Essentials license, VPN users can use a Web browser to log in, and download and start (WebLaunch) the AnyConnect client.

   The AnyConnect client software offers the same set of client features, whether it is enabled by this license or an AnyConnect Premium SSL VPN Edition license.

   The AnyConnect Essentials license cannot be active at the same time as the following licenses on a given security appliance: AnyConnect Premium license (all types) or the Advanced Endpoint Assessment license. You can, however, run AnyConnect Essentials and AnyConnect Premium licenses on different security appliances in the same network.

   By default, the security appliance uses the AnyConnect Essentials license, but you can disable it to use other licenses by using the **no anyconnect-essentials** command.

   For a detailed list of the features supported by the AnyConnect Essentials license and AnyConnect Premium license, see *AnyConnect Secure Mobility Client Features, Licenses, and OSs*:

   http://www.cisco.com/en/US/products/ps10884/products_feature_guides_list.html

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

Supported in single or multiple context mode.

Firewall Mode Guidelines

Supported in routed firewall mode only. Does not support transparent firewall mode.

Failover Guidelines

IPsec VPN sessions are replicated in Active/Standby failover configurations only.

IPv6 Guidelines

Does not support IPv6.

Configuring ISAKMP

This section describes the Internet Security Association and Key Management Protocol (ISAKMP) and the Internet Key Exchange (IKE) protocol.

This section includes the following topics:

Configuring IKEv1 and IKEv2 Policies

To create an IKE policy, enter the **crypto** ***ikev1*** **|** ***ikev2*** **policy** command from global configuration mode in either single or multiple context mode. The prompt displays IKE policy configuration mode. For example:

```
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)#
```

After creating the policy, you can specify the settings for the policy.

Table 1-1 and Table 1-2 provide information about the IKEv1 and IKEv2 policy keywords and their values.

Table 1-1 IKEv1 Policy Keywords for CLI Commands

Command	Keyword	Meaning	Description
authentication	rsa-sig	A digital certificate with keys generated by the RSA signatures algorithm	Specifies the authentication method the ASA uses to establish the identity of each IPsec peer.
	crack	Challenge/Response for Authenticated Cryptographic Keys	CRACK provides strong mutual authentication when the client authenticates using a legacy method such as RADIUS, and the server uses public key authentication.
	pre-share (default)	Preshared keys	Preshared keys do not scale well with a growing network but are easier to set up in a small network.
encryption	**des**	56-bit DES-CBC	Specifies the symmetric encryption algorithm that protects data transmitted between two IPsec peers. The default is 168-bit Triple DES.
	3des (default)	168-bit Triple DES	
hash	**sha** (default)	SHA-1 (HMAC variant)	Specifies the hash algorithm used to ensure data integrity. It ensures that a packet comes from where it says it comes from and that it has not been modified in transit.
	md5	MD5 (HMAC variant)	The default is SHA-1. MD5 has a smaller digest and is considered to be slightly faster than SHA-1. A successful (but extremely difficult) attack against MD5 has occurred; however, the HMAC variant IKE uses prevents this attack.
group	1	Group 1 (768-bit)	Specifies the Diffie-Hellman group identifier, which the two IPsec peers use to derive a shared secret without transmitting it to each other. The lower the Diffie-Hellman group number, the less CPU time it requires to execute. The higher the Diffie-Hellman group number, the greater the security. AES support is available on security appliances licensed for VPN-3DES only. To support the large key sizes required by AES, ISAKMP negotiation should use Diffie-Hellman (DH) Group 5.
	2 (default)	Group 2 (1024-bit)	
	5	Group 5 (1536-bit)	
lifetime	integer value (86400 = default)	120 to 2147483647 seconds	Specifies the SA lifetime. The default is 86,400 seconds or 24 hours. As a general rule, a shorter lifetime provides more secure ISAKMP negotiations (up to a point). However, with shorter lifetimes, the ASA sets up future IPsec SAs more quickly.

Table 1-2 IKEv2 Policy Keywords for CLI Commands

Command	Keyword	Meaning	Description
integrity	**sha** (default)	SHA-1 (HMAC variant)	Specifies the hash algorithm used to ensure data integrity. It ensures that a packet comes from where it says it comes from and that it has not been modified in transit.
	md5	MD5 (HMAC variant)	The default is SHA-1. MD5 has a smaller digest and is considered to be slightly faster than SHA-1. A successful (but extremely difficult) attack against MD5 has occurred; however, the HMAC variant IKE user prevents this attack.

Table 1-2 IKEv2 Policy Keywords for CLI Commands (continued)

Command	Keyword	Meaning	Description
	sha256	SHA 2, 256-bit digest	Specifies the Secure Hash Algorithm SHA 2 with the 256-bit digest.
	sha384	SHA 2, 384-bit digest	Specifies the Secure Hash Algorithm SHA 2 with the 384-bit digest.
	sha512	SHA 2, 512-bit digest	Specifies the Secure Hash Algorithm SHA 2 with the 512-bit digest.
	null		When AES-GCM is specified as the encryption algorithm, an administrator can choose null as the IKEv2 integrity algorithm.
encryption	**des**	56-bit DES-CBC	Specifies the symmetric encryption algorithm that protects data transmitted between two IPsec peers. The default is 168-bit Triple DES.
	3des (default)	168-bit Triple DES	
	aes aes-192 aes-256		The Advanced Encryption Standard supports key lengths of 128, 192, 256 bits.
	aes-gcm aes-gcm-192 aes-gcm-256 null	AES-GCM algorithm options to use for IKEv2 encryption	The Advanced Encryption Standard supports key lengths of 128, 192, 256 bits.
policy_index			Accesses the IKEv2 policy sub-mode.
prf	**sha** (default)	SHA-1 (HMAC variant)	Specifies the pseudo random function (PRF)—the algorithm used to generate keying material.
	md5	MD5 (HMAC variant)	The default is SHA-1. MD5 has a smaller digest and is considered to be slightly faster than SHA-1. A successful (but extremely difficult) attack against MD5 has occurred; however, the HMAC variant IKE uses prevents this attack.
	sha256	SHA 2, 256-bit digest	Specifies the Secure Hash Algorithm SHA 2 with the 256-bit digest.
	sha384	SHA 2, 384-bit digest	Specifies the Secure Hash Algorithm SHA 2 with the 384-bit digest.
	sha512	SHA 2, 512-bit digest	Specifies the Secure Hash Algorithm SHA 2 with the 512-bit digest.
priority			Extends the policy mode to support the additional IPsec V3 features and makes the AES-GCM and ECDH settings part of the Suite B support.

Table 1-2 IKEv2 Policy Keywords for CLI Commands (continued)

Command	Keyword	Meaning	Description
group	1 **2** (default) 5 14 19 20 21 24	Group 1 (768-bit) Group 2 (1024-bit) Group 5 (1536-bit)	Specifies the Diffie-Hellman group identifier, which the two IPsec peers use to derive a shared secret without transmitting it to each other. The lower the Diffie-Hellman group number, the less CPU time it requires to execute. The higher the Diffie-Hellman group number, the greater the security. The AnyConnect client supports DH group 1, 2, and 5 in non-FIPS mode, and groups 2 and only in FIPS mode. AES support is available on security appliances licensed for VPN-3DES only. To support the large key sizes required by AES, ISAKMP negotiation should use Diffie-Hellman (DH) Group 5.
lifetime	integer value (86400 = default)	120 to 2147483647 seconds	Specifies the SA lifetime. The default is 86,400 seconds or 24 hours. As a general rule, a shorter lifetime provides more secure ISAKMP negotiations (up to a point). However, with shorter lifetimes, the ASA sets up future IPsec SAs more quickly.

IKEv1 and IKEv2 each support a maximum of 20 IKE policies, each with a different set of values. Assign a unique priority to each policy that you create. The lower the priority number, the higher the priority.

When IKE negotiations begin, the peer that initiates the negotiation sends all of its policies to the remote peer, and the remote peer tries to find a match. The remote peer checks all of the peer's policies against each of its configured policies in priority order (highest priority first) until it discovers a match.

A match exists when both policies from the two peers contain the same encryption, hash, authentication, and Diffie-Hellman parameter values. For IKEv1, the remote peer policy must also specify a lifetime less than or equal to the lifetime in the policy the initiator sent. If the lifetimes are not identical, the ASA uses the shorter lifetime. For IKEv2 the lifetime is not negotiated but managed locally between each peer, making it possible to configure lifetime independently on each peer. If no acceptable match exists, IKE refuses negotiation and the SA is not established.

There is an implicit trade-off between security and performance when you choose a specific value for each parameter. The level of security the default values provide is adequate for the security requirements of most organizations. If you are interoperating with a peer that supports only one of the values for a parameter, your choice is limited to that value.

Note New ASA configurations do not have a default IKEv1 or IKEv2 policy.

To configure IKE policies, in global configuration mode, use the **crypto *ikev1* | *ikev2* policy** *priority* command to enter IKE policy configuration mode.

You must include the priority in each of the ISAKMP commands. The priority number uniquely identifies the policy and determines the priority of the policy in IKE negotiations.

To enable and configure IKE, complete the following steps, using the IKEv1 examples as a guide:

Note If you do not specify a value for a given policy parameter, the default value applies.

Step 1 Enter IKEv1 policy configuration mode:

```
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)#
```

Step 2 Specify the encryption algorithm. The default is Triple DES. This example sets encryption to DES.

encryption [aes | aes-192 | aes-256 | des | 3des]

For example:

```
hostname(config-ikev1-policy)# encryption des
```

Step 3 Specify the hash algorithm. The default is SHA-1. This example configures MD5.

hash [md5 | sha]

For example:

```
hostname(config-ikev1-policy)# hash md5
```

Step 4 Specify the authentication method. The default is preshared keys. This example configures RSA signatures.

authentication [pre-share | crack | rsa-sig]

For example:

```
hostname(config-ikev1-policy)# authentication rsa-sig
```

Step 5 Specify the Diffie-Hellman group identifier. The default is Group 2. This example configures Group 5.

group [**1** | **2** | **5**]

For example:

```
hostname(config-ikev1-policy)# group 5
```

Step 6 Specify the SA lifetime. This examples sets a lifetime of 4 hours (14400 seconds). The default is 86400 seconds (24 hours).

lifetime *seconds*

For example:

```
hostname(config-ikev1-policy)# lifetime 14400
```

Enabling IKE on the Outside Interface

You must enable IKE on the interface that terminates the VPN tunnel. Typically this is the outside, or public interface. To enable IKEv1 or IKEv2, use the **crypto** ***ikev1*** **|** ***ikev2*** **enable** *interface-name* command from global configuration mode in either single or multiple context mode.

For example:

```
hostname(config)# crypto ikev1 enable outside
```

Disabling IKEv1 Aggressive Mode

Phase 1 IKEv1 negotiations can use either main mode or aggressive mode. Both provide the same services, but aggressive mode requires only two exchanges between the peers totaling three messages, rather than three exchanges totaling six messages. Aggressive mode is faster, but does not provide identity protection for the communicating parties. Therefore, the peers must exchange identification information before establishing a secure SA. Aggressive mode is enabled by default.

- Main mode is slower, using more exchanges, but it protects the identities of the communicating peers.
- Aggressive mode is faster, but does not protect the identities of the peers.

To disable aggressive mode, enter the following command in either single or multiple context mode:

```
crypto ikev1 am-disable
```

For example:

```
hostname(config)# crypto ikev1 am-disable
```

If you have disabled aggressive mode, and want to revert to back to it, use the **no** form of the command. For example:

```
hostname(config)# no crypto ikev1 am-disable
```

Note Disabling aggressive mode prevents Cisco VPN clients from using preshared key authentication to establish tunnels to the ASA. However, they may use certificate-based authentication (that is, ASA or RSA) to establish tunnels.

Determining an ID Method for IKEv1 and IKEv2 ISAKMP Peers

During ISAKMP Phase I negotiations, either IKEv1 or IKEv2, the peers must identify themselves to each other. You can choose the identification method from the following options.

Address	Uses the IP addresses of the hosts exchanging ISAKMP identity information.
Automatic	Determines ISAKMP negotiation by connection type: • IP address for preshared key. • Cert Distinguished Name for certificate authentication.
Hostname	Uses the fully qualified domain name of the hosts exchanging ISAKMP identity information (default). This name comprises the hostname and the domain name.
Key ID *key_id_string*	Specifies the string used by the remote peer to look up the preshared key.

The ASA uses the Phase I ID to send to the peer. This is true for all VPN scenarios except LAN-to-LAN IKEv1 connections in main mode that authenticate with preshared keys.

The default setting is auto.

To change the peer identification method, enter the following command in either single or multiple context mode:

```
crypto isakmp identity {address | hostname | key-id id-string | auto}
```

For example, the following command sets the peer identification method to hostname:

```
hostname(config)# crypto isakmp identity hostname
```

Enabling IPsec over NAT-T

NAT-T lets IPsec peers establish a connection through a NAT device. It does this by encapsulating IPsec traffic in UDP datagrams, using port 4500, which provides NAT devices with port information. NAT-T auto-detects any NAT devices and only encapsulates IPsec traffic when necessary. This feature is disabled by default.

Note Due to a limitation of the AnyConnect client, you must enable NAT-T for the AnyConnect client to successfully connect using IKEv2. This requirement applies even if the client is not behind a NAT-T device.

With the exception of the home zone on the Cisco ASA 5505, the ASA can simultaneously support standard IPsec, IPsec over TCP, NAT-T, and IPsec over UDP, depending on the client with which it is exchanging data.

The following breakdown shows the connections with each option enabled.

Options	Enabled Feature	Client Position	Feature Used
Option 1	If NAT-T is enabled	and client is behind NAT, then	NAT-T is used
		and no NAT exists, then	Native IPsec (ESP) is used
Option 2	If IPsec over UDP is enabled	and client is behind NAT, then	IPsec over UDP is used
		and no NAT exists, then	IPsec over UDP is used
Option 3	If both NAT-T and IPsec over UDP are enabled	and client is behind NAT, then	NAT-T is used
		and no NAT exists, then	IPsec over UDP is used

Note When IPsec over TCP is enabled, it takes precedence over all other connection methods.

When you enable NAT-T, the ASA automatically opens port 4500 on all IPsec-enabled interfaces.

The ASA supports multiple IPsec peers behind a single NAT/PAT device operating in one of the following networks, but not both:

- LAN-to-LAN
- Remote access

In a mixed environment, the remote access tunnels fail the negotiation because all peers appear to be coming from the same public IP address, address of the NAT device. Also, remote access tunnels fail in a mixed environment because they often use the same name as the LAN-to-LAN tunnel group (that is, the IP address of the NAT device). This match can cause negotiation failures among multiple peers in a mixed LAN-to-LAN and remote access network of peers behind the NAT device.

Using NAT-T

To use NAT-T, you must perform the following site-to-site steps in either single or multiple context mode:

Step 1 Enter the following command to enable IPsec over NAT-T globally on the ASA:

crypto isakmp nat-traversal *natkeepalive*

The range for the *natkeepalive* argument is 10 to 3600 seconds. The default is 20 seconds.

For example, enter the following command to enable NAT-T and set the keepalive value to one hour.

```
hostname(config)# crypto isakmp nat-traversal 3600
```

Step 2 Select the before-encryption option for the IPsec fragmentation policy by entering this command:

```
hostname(config)# crypto ipsec fragmentation before-encryption
```

This option lets traffic travel across NAT devices that do not support IP fragmentation. It does not impede the operation of NAT devices that do support IP fragmentation.

Enabling IPsec with IKEv1 over TCP

IPsec/IKEv1 over TCP enables a Cisco VPN client to operate in an environment in which standard ESP or IKEv1 cannot function or can function only with modification to existing firewall rules. IPsec over TCP encapsulates both the IKEv1 and IPsec protocols within a TCP-like packet and enables secure tunneling through both NAT and PAT devices and firewalls. This feature is disabled by default.

Note This feature does not work with proxy-based firewalls.

IPsec over TCP works with remote access clients. You enable it globally, and it works on all IKEv1-enabled interfaces. It is a client to the ASA feature only. It does not work for LAN-to-LAN connections.

The ASA can simultaneously support standard IPsec, IPsec over TCP, NAT-Traversal, and IPsec over UDP, depending on the client with which it is exchanging data. IPsec over TCP, if enabled, takes precedence over all other connection methods.

The VPN 3002 hardware client, which supports one tunnel at a time, can connect using standard IPsec, IPsec over TCP, NAT-Traversal, or IPsec over UDP.

You enable IPsec over TCP on both the ASA and the client to which it connects.

You can enable IPsec over TCP for up to 10 ports that you specify. If you enter a well-known port, for example port 80 (HTTP) or port 443 (HTTPS), the system displays a warning that the protocol associated with that port no longer works on the public interface. The consequence is that you can no longer use a browser to manage the ASA through the public interface. To solve this problem, reconfigure the HTTP/HTTPS management to different ports.

The default port is 10000.

You must configure TCP port(s) on the client as well as on the ASA. The client configuration must include at least one of the ports you set for the ASA.

To enable IPsec over TCP for IKEv1 globally on the ASA, perform the following command in either single or multiple context mode:

crypto ikev1 ipsec-over-tcp [port *port 1...port0]*

This example enables IPsec over TCP on port 45:

```
hostname(config)# crypto ikev1 ipsec-over-tcp port 45
```

Waiting for Active Sessions to Terminate Before Rebooting

You can schedule an ASA reboot to occur only when all active sessions have terminated voluntarily. This feature is disabled by default.

To enable waiting for all active sessions to voluntarily terminate before the ASA reboots, perform the following site-to-site task in either single or multiple context mode:

crypto isakmp reload-wait

For example:

```
hostname(config)# crypto isakmp reload-wait
```

Use the **reload** command to reboot the ASA. If you set the **reload-wait** command, you can use the **reload quick** command to override the **reload-wait** setting. The **reload** and **reload-wait** commands are available in privileged EXEC mode; neither includes the **isakmp** prefix.

Alerting Peers Before Disconnecting

Remote access or LAN-to-LAN sessions can drop for several reasons, such as an ASA shutdown or reboot, session idle timeout, maximum connection time exceeded, or administrator cut-off.

The ASA can notify qualified peers (in LAN-to-LAN configurations), Cisco VPN clients, and VPN 3002 hardware clients of sessions that are about to be disconnected. The peer or client receiving the alert decodes the reason and displays it in the event log or in a pop-up pane. This feature is disabled by default.

Qualified clients and peers include the following:

- Security appliances with Alerts enabled
- Cisco VPN clients running Version 4.0 or later software (no configuration required)
- VPN 3002 hardware clients running Version 4.0 or later software, with Alerts enabled
- VPN 3000 series concentrators running Version 4.0 or later software with Alerts enabled

To enable disconnect notification to IPsec peers, enter the **crypto isakmp disconnect-notify** command in either single or multiple context mode.

For example:

```
hostname(config)# crypto isakmp disconnect-notify
```

Configuring Certificate Group Matching for IKEv1

Tunnel groups define user connection terms and permissions. Certificate group matching lets you match a user to a tunnel group using either the Subject DN or Issuer DN of the user certificate.

Note Certificate group matching applies to IKEv1 and IKEv2 LAN-to-LAN connections only. IKEv2 remote access connections support the pull-down group selection configured in the webvpn-attributes of the tunnel-group and webvpn configuration mode for certificate-group-map, and so on.

To match users to tunnel groups based on these fields of the certificate, you must first create rules that define a matching criteria, and then associate each rule with the desired tunnel group.

To create a certificate map, **use the crypto ca certificate map** command. To define a tunnel group, use the **tunnel-group** command.

You must also configure a certificate group matching policy, specifying to match the group from the rules, or from the organizational unit (OU) field, or to use a default group for all certificate users. You can use any or all of these methods.

The following sections provide more information:

- Creating a Certificate Group Matching Rule and Policy, page 1-17
- Using the Tunnel-group-map default-group Command, page 1-18

Creating a Certificate Group Matching Rule and Policy

To configure the policy and rules by which certificate-based ISAKMP sessions map to tunnel groups, and to associate the certificate map entries with tunnel groups, enter the **tunnel-group-map** command in either single or multiple context mode.

The syntax follows:

tunnel-group-map enable {*rules* | *ou* | *ike-id* | *peer ip*}

tunnel-group-map [*rule-index*] **enable** *policy*

policy	Specifies the policy for deriving the tunnel group name from the certificate. *Policy* can be one of the following: *ike-id*—Indicates that if a tunnel group is not determined based on a rule lookup or taken from the OU, then the certificate-based ISAKMP sessions are mapped to a tunnel group based on the content of the phase1 ISAKMP ID. *ou*—Indicates that if a tunnel-group is not determined based on a rule lookup, then use the value of the OU in the subject distinguished name (DN). *peer-ip*—Indicates that if a tunnel group is not determined based on a rule lookup or taken from the OU or ike-id methods, then use the peer IP address. *rules*—Indicates that the certificate-based ISAKMP sessions are mapped to a tunnel group based on the certificate map associations configured by this command.
rule index	(Optional) Refers to parameters specified by the **crypto ca certificate map** command. The values are 1 to 65535.

Be aware of the following:

- You can invoke this command multiple times as long as each invocation is unique and you do not reference a map index more than once.
- Rules cannot be longer than 255 characters.

- You can assign multiple rules to the same group. To do that, you add the rule priority and group first. Then you define as many criteria statements as you need for each group. When multiple rules are assigned to the same group, a match results for the first rule that tests true.
- By creating a single rule, you can require all criteria to match before assigning a user to a specific tunnel group. Requiring all criteria to match is equivalent to a logical AND operation. Alternatively, create one rule for each criterion if you want to require that only one match before assigning a user to a specific tunnel group. Requiring only one criterion to match is equivalent to a logical OR operation.

The following example enables mapping of certificate-based ISAKMP sessions to a tunnel group based on the content of the phase1 ISAKMP ID:

```
ciscoasa(config)# tunnel-group-map enable ike-id
ciscoasa(config)#
```

The following example enables mapping of certificate-based ISAKMP sessions to a tunnel group based on the IP address of the peer:

```
ciscoasa(config)# tunnel-group-map enable peer-ip
ciscoasa(config)#
```

The following example enables mapping of certificate-based ISAKMP sessions based on the organizational unit (OU) in the subject distinguished name (DN):

```
ciscoasa(config)# tunnel-group-map enable ou
ciscoasa(config)#
```

The following example enables mapping of certificate-based ISAKMP sessions based on established rules:

```
ciscoasa(config)# tunnel-group-map enable rules
ciscoasa(config)#
```

Using the Tunnel-group-map default-group Command

This command specifies a default tunnel group to use when the configuration does not specify a tunnel group.

The syntax is **tunnel-group-map** [*rule-index*] **default-group** *tunnel-group-name* where *rule-index* is the priority for the rule, and *tunnel-group name* must be for a tunnel group that already exists.

Configuring IPsec

This section provides background information about IPsec and describes the procedures required to configure the ASA when using IPsec to implement a VPN. It contains the following topics:

Understanding IPsec Tunnels

IPsec tunnels are sets of SAs that the ASA establishes between peers. The SAs specify the protocols and algorithms to apply to sensitive data and also specify the keying material that the peers use. IPsec SAs control the actual transmission of user traffic. SAs are unidirectional, but are generally established in pairs (inbound and outbound).

The peers negotiate the settings to use for each SA. Each SA consists of the following:

- IKEv1 transform sets or IKEv2 proposals
- Crypto maps
- ACLs
- Tunnel groups
- Prefragmentation policies

Understanding IKEv1 Transform Sets and IKEv2 Proposals

An IKEv1 transform set or an IKEv2 proposal is a combination of security protocols and algorithms that define how the ASA protects data. During IPsec SA negotiations, the peers must identify a transform set or proposal that is the same at both peers. The ASA then applies the matching transform set or proposal to create an SA that protects data flows in the ACL for that crypto map.

With IKEv1 transform sets, you set one value for each parameter. For IKEv2 proposals, you can configure multiple encryption and authentication types and multiple integrity algorithms for a single proposal. The ASA orders the settings from the most secure to the least secure and negotiates with the peer using that order. This allows you to potentially send a single proposal to convey all the allowed combinations instead of the need to send each allowed combination individually as with IKEv1.

The ASA tears down the tunnel if you change the definition of the transform set or proposal used to create its SA. See the "Clearing Security Associations" section on page 1-38" for further information.

Note If you clear or delete the only element in a transform set or proposal, the ASA automatically removes the crypto map references to it.

Defining Crypto Maps

Crypto maps define the IPsec policy to be negotiated in the IPsec SA. They include the following:

- ACL to identify the packets that the IPsec connection permits and protects.
- Peer identification.
- Local address for the IPsec traffic. (See "Applying Crypto Maps to Interfaces" for more details.)
- Up to 11 IKEv1 transform sets or IKEv2 proposals, with which to attempt to match the peer security settings.

A *crypto map set* consists of one or more crypto maps that have the same map name. You create a crypto map set when you create its first crypto map. The following site-to-site task creates or adds to a crypto map in either single or multiple context mode:

crypto map *map-name seq-num* **match address** *access-list-name*

Use the access-list-name to specify the ACL ID, as a string or integer up to 241 characters in length.

Tip Use all capital letters to more easily identify the ACL ID in your configuration.

You can continue to enter this command to add crypto maps to the crypto map set. In the following example, *mymap* is the name of the crypto map set to which you might want to add crypto maps:

```
crypto map mymap 10 match address 101
```

The *sequence number* (*seq-num*) shown in the syntax above distinguishes one crypto map from another one with the same name. The sequence number assigned to a crypto map also determines its priority among the other crypto maps within a crypto map set. The lower the sequence number, the higher the priority. After you assign a crypto map set to an interface, the ASA evaluates all IP traffic passing through the interface against the crypto maps in the set, beginning with the crypto map with the lowest sequence number.

```
[no] crypto map <map_name> <map_index> set pfs [group1 | group2 | group5 | group14 |
group19 | group20 | group21 | group24]
```

Specifies the ECDH group used for Perfect Forward Secrecy (FCS) for the cryptography map. Prevents you from onfiguring group14 and group24 options for a cryptography map (when using an IKEv1 policy).

```
[no]crypto map <name> <priority> set validate-icmp-errors
OR
[no]crypto dynamic-map <name> <priority> set validate-icmp-errors
```

Specifies whether incoming ICMP error messages are validated for the cryptography or dynamic cryptographyy map.

```
[no] crypto map <name> <priority> set df-bit [clear-df | copy-df | set-df}
OR
[no] crypto map dynamic-map <name> <priority> set df-bit [clear-df | copy-df | set-df]
```

Configures the existing do not fragment (DF) policy (at a security association level) for the cryptography or dynamic cryptography map.

- *clear-df*—Ignores the DF bit.
- *copy-df*—Maintains the DF bit.
- *set-df*—Sets and uses the DF bit.

```
[no] crypto map <name> <priority> set tfc-packets [burst <length | auto] [payload-size
<bytes | auto> [timeout <seconds | auto>
OR
[no] crypto dynamic-map <name> <priority> set tfc-packets [burst <length | auto]
[payload-size <bytes | auto> [timeout <seconds | auto>
```

An administrator can enable dummy Traffic Flow Confidentiality (TFC) packets at random lengths and intervals on an IPsec security association. You must have an IKEv2 IPsec proposal set before enabling TFC.

The ACL assigned to a crypto map consists of all of the ACEs that have the same ACL name, as shown in the following command syntax:

access-list *access-list-name* **{deny | permit} ip** *source source-netmask destination destination-netmask*

Each ACL consists of one or more ACEs that have the same ACL name. You create an ACL when you create its first ACE. The following command syntax creates or adds to an ACL:

access-list *access-list-name* **{deny | permit} ip** *source source-netmask destination destination-netmask*

In the following example, the ASA applies the IPsec protections assigned to the crypto map to all traffic flowing from the 10.0.0.0 subnet to the 10.1.1.0 subnet:

```
access-list 101 permit ip 10.0.0.0 255.255.255.0 10.1.1.0 255.255.255.0
```

The crypto map that matches the packet determines the security settings used in the SA negotiations. If the local ASA initiates the negotiation, it uses the policy specified in the static crypto map to create the offer to send to the specified peer. If the peer initiates the negotiation, the ASA attempts to match the policy to a static crypto map, and if that fails, then it attempts to match any dynamic crypto maps in the crypto map set, to decide whether to accept or reject the peer offer.

For two peers to succeed in establishing an SA, they must have at least one compatible crypto map. To be compatible, a crypto map must meet the following criteria:

- The crypto map must contain compatible crypto ACLs (for example, mirror image ACLs). If the responding peer uses dynamic crypto maps, so the ASA also must contain compatible crypto ACLs as a requirement to apply IPsec.
- Each crypto map identifies the other peer (unless the responding peer uses dynamic crypto maps).
- The crypto maps have at least one transform set or proposal in common.

You can apply only one crypto map set to a single interface. Create more than one crypto map for a particular interface on the ASA if any of the following conditions exist:

- You want specific peers to handle different data flows.
- You want different IPsec security to apply to different types of traffic.

For example, create a crypto map and assign an ACL to identify traffic between two subnets and assign one IKEv1 transform set or IKEv2 proposal. Create another crypto map with a different ACL to identify traffic between another two subnets and apply a transform set or proposal with different VPN parameters.

If you create more than one crypto map for an interface, specify a sequence number (seq-num) for each map entry to determine its priority within the crypto map set.

Each ACE contains a permit or deny statement. Table 1-3 explains the special meanings of permit and deny ACEs in ACLs applied to crypto maps.

Table 1-3 Special Meanings of Permit and Deny in Crypto ACLs Applied to Outbound Traffic

Result of Crypto Map Evaluation	Response
Match criterion in an ACE containing a permit statement	Halt further evaluation of the packet against the remaining ACEs in the crypto map set, and evaluate the packet security settings against those in the IKEv1 transform sets or IKEv2 proposals assigned to the crypto map. After matching the security settings to those in a transform set or proposal, the ASA applies the associated IPsec settings. Typically for outbound traffic, this means that it decrypts, authenticates, and routes the packet.
Match criterion in an ACE containing a deny statement	Interrupt further evaluation of the packet against the remaining ACEs in the crypto map under evaluation, and resume evaluation against the ACEs in the next crypto map, as determined by the next seq-num assigned to it.
Fail to match all tested permit ACEs in the crypto map set	Route the packet without encrypting it.

ACEs containing deny statements filter out outbound traffic that does not require IPsec protection (for example, routing protocol traffic). Therefore, insert initial deny statements to filter outbound traffic that should not be evaluated against permit statements in a crypto ACL.

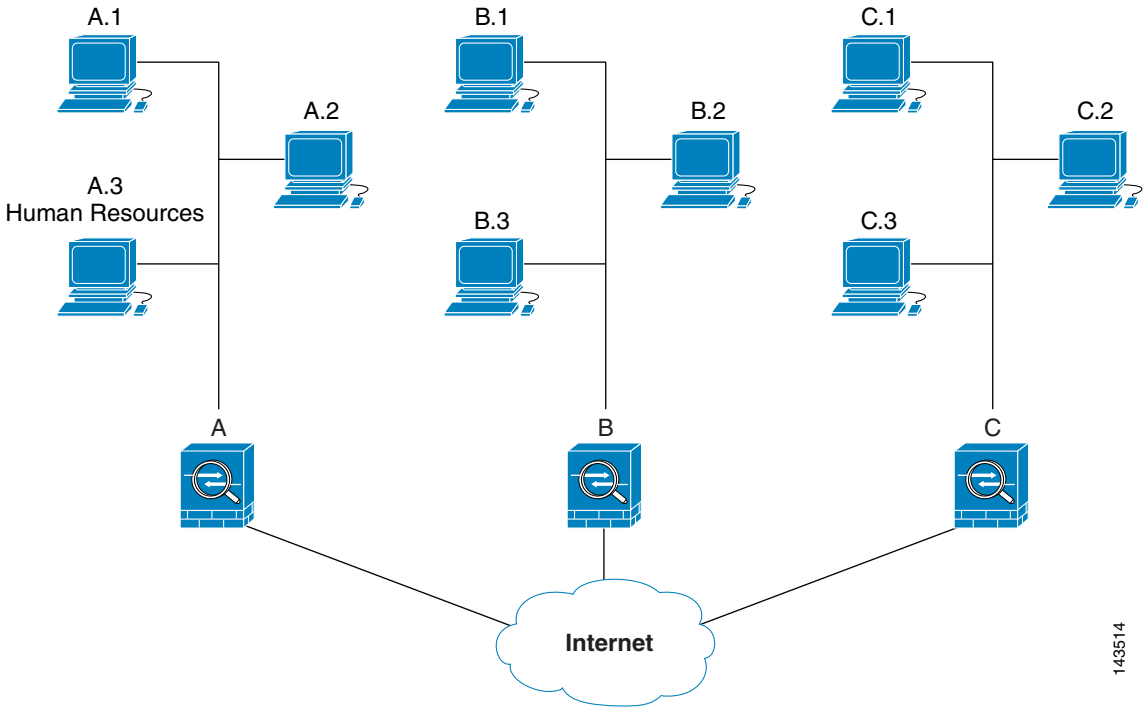
For an inbound, encrypted packet, the security appliance uses the source address and ESP SPI to determine the decryption parameters. After the security appliance decrypts the packet, it compares the inner header of the decrypted packet to the permit ACEs in the ACL associated with the packet SA. If the inner header fails to match the proxy, the security appliance drops the packet. It the inner header matches the proxy, the security appliance routes the packet.

When comparing the inner header of an inbound packet that was not encrypted, the security appliance ignores all deny rules because they would prevent the establishment of a Phase 2 SA.

Note To route inbound, unencrypted traffic as clear text, insert deny ACEs before permit ACEs.

Figure 1-1 shows an example LAN-to-LAN network of ASAs.

Figure 1-1 Effect of Permit and Deny ACEs on Traffic (Conceptual Addresses)



The simple address notation shown in this figure and used in the following explanation is an abstraction. An example with real IP addresses follows the explanation.

The objective in configuring Security Appliances A, B, and C in this example LAN-to-LAN network is to permit tunneling of all traffic originating from one of the hosts shown in Figure 1-1 and destined for one of the other hosts. However, because traffic from Host A.3 contains sensitive data from the Human Resources department, it requires strong encryption and more frequent rekeying than the other traffic. So you will want to assign a special transform set for traffic from Host A.3.

To configure Security Appliance A for outbound traffic, you create two crypto maps, one for traffic from Host A.3 and the other for traffic from the other hosts in Network A, as shown in the following example:

```
Crypto Map Seq_No_1
    deny packets from A.3 to B
    deny packets from A.3 to C
    permit packets from A to B
    permit packets from A to C
Crypto Map Seq_No_2
    permit packets from A.3 to B
    permit packets from A.3 to C
```

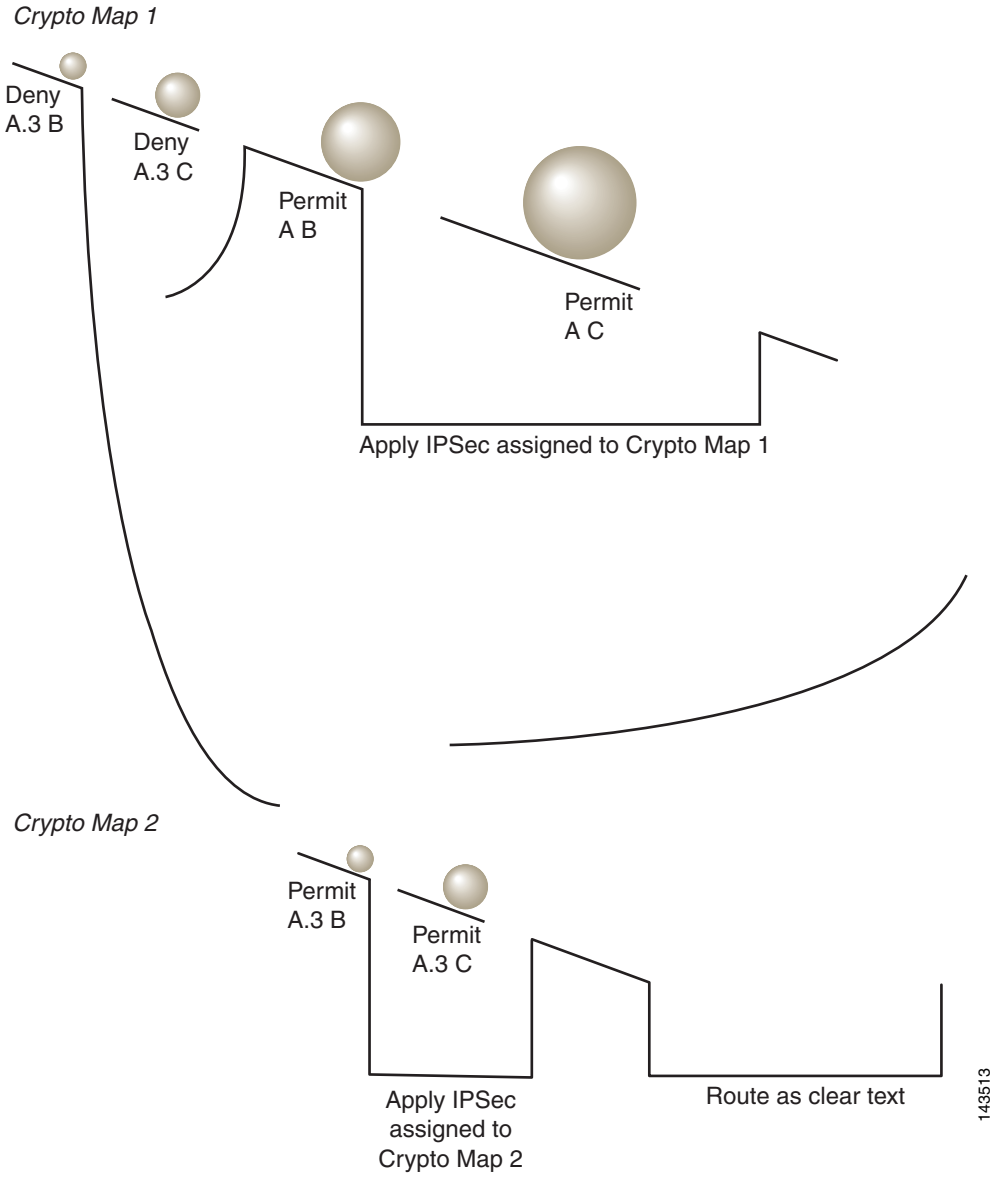
After creating the ACLs, you assign a transform set to each crypto map to apply the required IPsec to each matching packet.

Cascading ACLs involves the insertion of deny ACEs to bypass evaluation against an ACL and resume evaluation against a subsequent ACL in the crypto map set. Because you can associate each crypto map with different IPsec settings, you can use deny ACEs to exclude special traffic from further evaluation in the corresponding crypto map, and match the special traffic to permit statements in another crypto map to provide or require different security. The sequence number assigned to the crypto ACL determines its position in the evaluation sequence within the crypto map set.

Figure 1-2 shows the cascading ACLs created from the conceptual ACEs in this example. The meaning of each symbol in the figure follows.

	Crypto map within a crypto map set.
	(Gap in a straight line) Exit from a crypto map when a packet matches an ACE.
	Packet that fits the description of one ACE. Each size ball represents a different packet matching the respective ACE in the figure. The differences in size merely represent differences in the source and destination of each packet.
	Redirection to the next crypto map in the crypto map set.
	Response when a packet either matches an ACE or fails to match all of the permit ACEs in a crypto map set.

Figure 1-2 Cascading ACLs in a Crypto Map Set



Security Appliance A evaluates a packet originating from Host A.3 until it matches a permit ACE and attempts to assign the IPsec security associated with the crypto map. Whenever the packet matches a deny ACE, the ASA ignores the remaining ACEs in the crypto map and resumes evaluation against the next crypto map, as determined by the sequence number assigned to it. So in the example, if Security Appliance A receives a packet from Host A.3, it matches the packet to a deny ACE in the first crypto map and resumes evaluation of the packet against the next crypto map. When it matches the packet to the permit ACE in that crypto map, it applies the associated IPsec security (strong encryption and frequent rekeying).

To complete the security appliance configuration in the example network, we assign mirror crypto maps to Security Appliances B and C. However, because security appliances ignore deny ACEs when evaluating inbound, encrypted traffic, we can omit the mirror equivalents of the deny A.3 B and deny A.3 C ACEs, and therefore omit the mirror equivalents of Crypto Map 2. So the configuration of cascading ACLs in Security Appliances B and C is unnecessary.

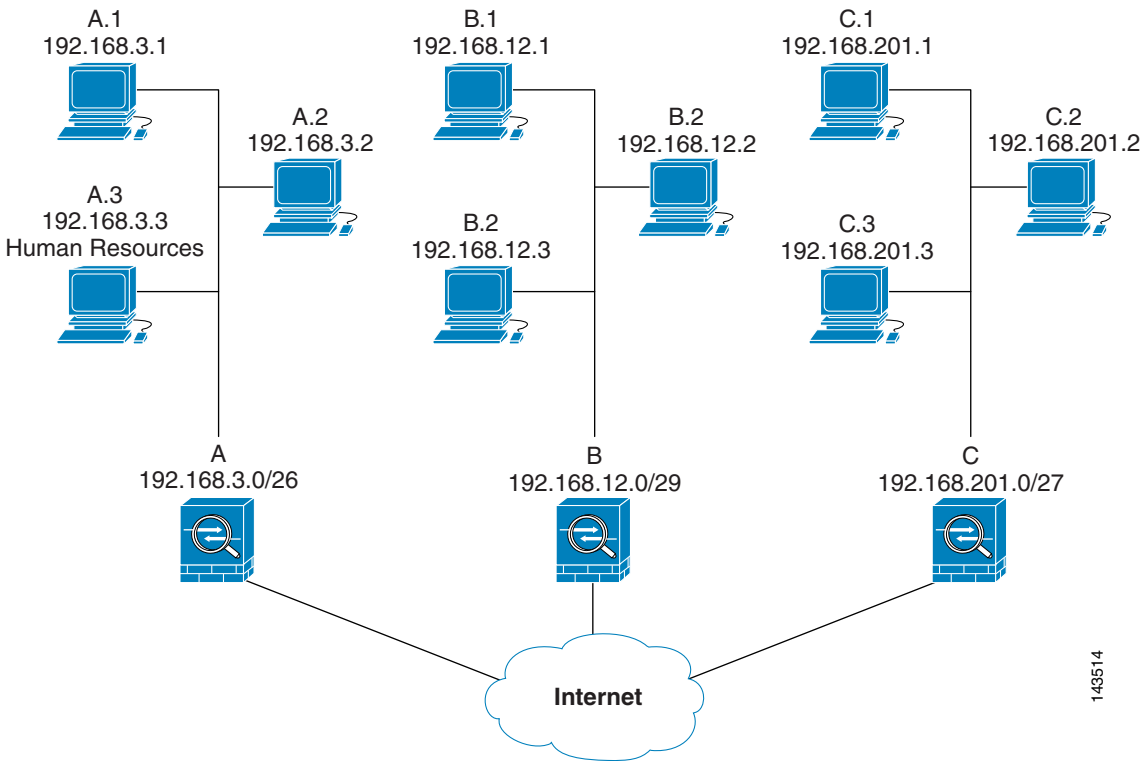
Table 1-4 shows the ACLs assigned to the crypto maps configured for all three ASAs in Figure 1-1.

Table 1-4 Example Permit and Deny Statements (Conceptual)

Security Appliance A		Security Appliance B		Security Appliance C	
Crypto Map Sequence No.	**ACE Pattern**	**Crypto Map Sequence No.**	**ACE Pattern**	**Crypto Map Sequence No.**	**ACE Pattern**
1	deny A.3 B	1	permit B A	1	permit C A
	deny A.3 C				
	permit A B				
	permit A C		permit B C		permit C B
2	permit A.3 B				
	permit A.3 C				

Figure 1-3 maps the conceptual addresses shown in Figure 1-1 to real IP addresses.

Figure 1-3 Effect of Permit and Deny ACEs on Traffic (Real Addresses)



The tables that follow combine the IP addresses shown in Figure 1-3 to the concepts shown in Table 1-4. The real ACEs shown in these tables ensure that all IPsec packets under evaluation within this network receive the proper IPsec settings.

Table 1-5 Example Permit and Deny Statements for Security Appliance A

Security Appliance	Crypto Map Sequence No.	ACE Pattern	Real ACEs
A	1	deny A.3 B	deny 192.168.3.3 255.255.255.192 192.168.12.0 255.255.255.248
		deny A.3 C	deny 192.168.3.3 255.255.255.192 192.168.201.0 255.255.255.224
		permit A B	permit 192.168.3.0 255.255.255.192 192.168.12.0 255.255.255.248
		permit A C	permit 192.168.3.0 255.255.255.192 192.168.201.0 255.255.255.224
	2	permit A.3 B	permit 192.168.3.3 255.255.255.192 192.168.12.0 255.255.255.248
		permit A.3 C	permit 192.168.3.3 255.255.255.192 192.168.201.0 255.255.255.224
B	None needed	permit B A	permit 192.168.12.0 255.255.255.248 192.168.3.0 255.255.255.192
		permit B C	permit 192.168.12.0 255.255.255.248 192.168.201.0 255.255.255.224
C	None needed	permit C A	permit 192.168.201.0 255.255.255.224 192.168.3.0 255.255.255.192
		permit C B	permit 192.168.201.0 255.255.255.224 192.168.12.0 255.255.255.248

You can apply the same reasoning shown in the example network to use cascading ACLs to assign different security settings to different hosts or subnets protected by a ASA.

Note By default, the ASA does not support IPsec traffic destined for the same interface from which it enters. Names for this type of traffic include U-turn, hub-and-spoke, and hairpinning. However, you can configure IPsec to support U-turn traffic by inserting an ACE to permit traffic to and from the network. For example, to support U-turn traffic on Security Appliance B, add a conceptual “permit B B” ACE to ACL1. The actual ACE would be as follows:

```
permit 192.168.12.0 255.255.255.248 192.168.12.0 255.255.255.248
```

Managing Public Key Infrastructure (PKI) Keys

You must set public key infrastructure (PKI) in order for an administrator to choose the Suite B ECDSA algorithms when generating or zeroing a keypair:

Prerequisites

If you are configuring a cryptography map to use an RSA or ECDSA trustpoint for authentication, you must first generate the key set. You can then create the trustpoint and reference it in the tunnel group configuration.

Restrictions

The 4096-bit RSA keys are only supported on the 5580, 5585, or later platforms.

Detailed Steps

Step 1 Choose the Suite B ECDSA algorithm when generating a keypair:

```
crypto key generate [rsa [general-keys | label <name> | modules [512 | 768 | 1024 | 2048 |
4096 ] | noconfirm | usage-keys] | ecdsa [label <name> | elliptic-curve [256 | 384 | 521]
| noconfirm] ]
```

Step 2 Choose the Suite B ECDSA algorithem when zeroizing a keypair:

```
crypto key zeroize [rsa | ecdsa] [default | label <name> | noconfirm]
```

Configuring the Pool of Cryptographic Cores

You can change the allocation of the cryptographic cores on Symmetric Multi-Processing (SMP) platforms to give you better throughput performance for AnyConnect TLS/DTLS traffic. These changes can accelerate the SSL VPN datapath and provide customer-visible performance gains in AnyConnect, smart tunnels, and port forwarding. To configure the pool of cryptographic cores, perform the following steps.

Limitations

- Cryptographic core rebalancing is available on the following platforms:
  - 5585
  - 5580
  - 5545/5555
  - ASA-SM
- The large modulus operation is only available for 5510, 5520, 5540, and 5550 platforms.

Detailed Steps

Step 1 Configure the pool of cryptographic cores specifying one of three mutually exclusive options:

- balanced—Equally distributes cryptography hardware resources (Admin/SSL and IPsec cores).
- ipsec—Allocates cryptography hardware resources to favor IPsec (includes SRTP encrypted voice traffic).
- ssl—Allocates cryptography hardware resources to favor Admin/SSL.

```
asa1(config)# crypto engine ?

configure mode commands/options:
accelerator-bias
Specify how to allocate crypto accelerator processors

asa1(config)# crypto engine accelerator-bias ?
configure mode commands/options
balanced - Equally distribute crypto hardware resources
ipsec - Allocate crypto hardware resources to favor IPsec/Encrypted Voice (SRTP)
ssl - Allocate crypto hardware resources to favor SSL

asa1(config)# crypto engine accelerator-bias ssl
```

Step 2 Perform large modulus operation in the hardware:

```
large-mode-accel
```

Applying Crypto Maps to Interfaces

You must assign a crypto map set to each interface through which IPsec traffic flows. The ASA supports IPsec on all interfaces. Assigning the crypto map set to an interface instructs the ASA to evaluate all the traffic against the crypto map set and to use the specified policy during connection or SA negotiation.

Assigning a crypto map to an interface also initializes run-time data structures, such as the SA database and the security policy database. Reassigning a modified crypto map to the interface resynchronizes the run-time data structures with the crypto map configuration. Also, adding new peers through the use of new sequence numbers and reassigning the crypto map does not tear down existing connections.

Using Interface ACLs

By default, the ASA lets IPsec packets bypass interface ACLs. If you want to apply interface ACLs to IPsec traffic, use the **no** form of the **sysopt connection permit-vpn** command.

The crypto map ACL bound to the outgoing interface either permits or denies IPsec packets through the VPN tunnel. IPsec authenticates and deciphers packets that arrive from an IPsec tunnel, and subjects them to evaluation against the ACL associated with the tunnel.

ACLs define which IP traffic to protect. For example, you can create ACLs to protect all IP traffic between two subnets or two hosts. (These ACLs are similar to ACLs used with the **access-group** command. However, with the **access-group** command, the ACL determines which traffic to forward or block at an interface.)

Before the assignment to crypto maps, the ACLs are not specific to IPsec. Each crypto map references the ACLs and determines the IPsec properties to apply to a packet if it matches a permit in one of the ACLs.

ACLs assigned to IPsec crypto maps have four primary functions:

- Select outbound traffic to be protected by IPsec (permit = protect).
- Trigger an ISAKMP negotiation for data travelling without an established SA.
- Process inbound traffic to filter out and discard traffic that should have been protected by IPsec.
- Determine whether to accept requests for IPsec SAs when processing IKE negotiation from the peer. (Negotiation applies only to **ipsec-isakmp crypto map** entries.) The peer must permit a data flow associated with an **ipsec-isakmp crypto map** command entry to ensure acceptance during negotiation.

Regardless of whether the traffic is inbound or outbound, the ASA evaluates traffic against the ACLs assigned to an interface. Follow these steps to assign IPsec to an interface:

Step 1 Create the ACLs to be used for IPsec.

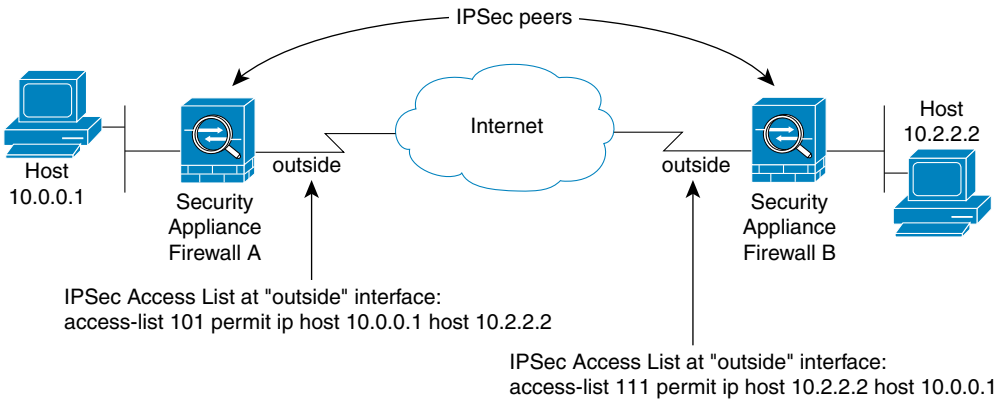
Step 2 Map the lists to one or more crypto maps, using the same crypto map name.

Step 3 Map the IKEv1 transform sets or IKEv2 proposals to the crypto maps to apply IPsec to the data flows.

Step 4 Apply the crypto maps collectively as a crypto map set by assigning the crypto map name they share to the interface.

In Figure 1-4, IPsec protection applies to traffic between Host 10.0.0.1 and Host 10.2.2.2 as the data exits the outside interface on Security Appliance A toward Host 10.2.2.2.

Figure 1-4 How Crypto ACLs Apply to IPsec



Security Appliance A evaluates traffic from Host 10.0.0.1 to Host 10.2.2.2, as follows:

- source = host 10.0.0.1
- dest = host 10.2.2.2

Security Appliance A also evaluates traffic from Host 10.2.2.2 to Host 10.0.0.1, as follows:

- source = host 10.2.2.2
- dest = host 10.0.0.1

The first permit statement that matches the packet under evaluation determines the scope of the IPsec SA.

Note If you delete the only element in an ACL, the ASA also removes the associated crypto map.

If you modify an ACL currently referenced by one or more crypto maps, use the **crypto map interface** command to reinitialize the run-time SA database. See the **crypto map** command for more information.

We recommend that for every crypto ACL specified for a static crypto map that you define at the local peer, you define a “mirror image” crypto ACL at the remote peer. The crypto maps should also support common transforms and refer to the other system as a peer. This ensures correct processing of IPsec by both peers.

Note Every static crypto map must define an ACL and an IPsec peer. If either is missing, the crypto map is incomplete and the ASA drops any traffic that it has not already matched to an earlier, complete crypto map. Use the **show conf** command to ensure that every crypto map is complete. To fix an incomplete crypto map, remove the crypto map, add the missing entries, and reapply it.

We discourage the use of the **any** keyword to specify source or destination addresses in crypto ACLs because they cause problems. We strongly discourage the **permit any any** command statement because it does the following:

- Protects all outbound traffic, including all protected traffic sent to the peer specified in the corresponding crypto map.
- Requires protection for all inbound traffic.

  In this scenario, the ASA silently drops all inbound packets that lack IPsec protection.

Be sure that you define which packets to protect. If you use the **any** keyword in a **permit** statement, preface it with a series of **deny** statements to filter out traffic that would otherwise fall within that **permit** statement that you do not want to protect.

Note Decrypted through traffic is permitted from the client despite having an access group on the outside interface, which calls a deny ip any any access-list, while **no sysopt connection permit-vpn** is configured.

Users who want to control access to the protected network via site-to-site or remote access VPN using the **no sysopt permit** command in conjunction with an access control list (ACL) on the outside interface are not successful.

In this situation, when management-access inside is enabled, the ACL is not applied, and users can still connect using SSH to the security appliance. Traffic to hosts on the inside network are blocked correctly by the ACL, but cannot block decrypted through traffic to the inside interface.

The **ssh** and **http** commands are of a higher priority than the ACLs. In other words, to deny SSH, Telnet, or ICMP traffic to the device from the VPN session, use **ssh**, **telnet** and **icmp** commands, which deny the IP local pool should be added.

Changing IPsec SA Lifetimes

You can change the global lifetime values that the ASA uses when negotiating new IPsec SAs. You can override these global lifetime values for a particular crypto map.

IPsec SAs use a derived, shared, secret key. The key is an integral part of the SA; the keys time out together to require the key to refresh. Each SA has two lifetimes: timed and traffic-volume. An SA expires after the respective lifetime and negotiations begin for a new one. The default lifetimes are 28,800 seconds (eight hours) and 4,608,000 kilobytes (10 megabytes per second for one hour).

If you change a global lifetime, the ASA drops the tunnel. It uses the new value in the negotiation of subsequently established SAs.

When a crypto map does not have configured lifetime values and the ASA requests a new SA, it inserts the global lifetime values used in the existing SA into the request sent to the peer. When a peer receives a negotiation request, it uses the smaller of either the lifetime value the peer proposes or the locally configured lifetime value as the lifetime of the new SA.

The peers negotiate a new SA before crossing the lifetime threshold of the existing SA to ensure that a new SA is ready when the existing one expires. The peers negotiate a new SA when about 5 to 15 percent of the lifetime of the existing SA remains.

Creating a Basic IPsec Configuration

You can create basic IPsec configurations with static or dynamic crypto maps.

To create a basic IPsec configuration using a static crypto map, perform the following steps:

Step 1 To create an ACL to define the traffic to protect, enter the following command:

access-list *access-list-name* **{deny | permit} ip** *source source-netmask destination destination-netmask*

For example:

```
hostname(config)# access-list 101 permit ip 10.0.0.0 255.255.255.0 10.1.1.0 255.255.255.0
```

The *access-list-name* specifies the ACL ID, as a string or integer up to 241 characters in length. The *destination-netmask* and *source-netmask* specifies an IPv4 network address and subnet mask. In this example, the **permit** keyword causes all traffic that matches the specified conditions to be protected by crypto.

Step 2 To configure an IKEv1 transform set that defines how to protect the traffic, enter the following command:

crypto ipsec ikev1 transform-set *transform-set-name encryption* [*authentication*]

Encryption specifies which encryption method protects IPsec data flows:

- esp-aes—Uses AES with a 128-bit key.
- esp-aes-192—Uses AES with a 192-bit key.
- esp-aes-256—Uses AES with a 256-bit key.\
- esp-des—Uses 56-bit DES-CBC.
- esp-3des—Uses triple DES algorithm.
- esp-null—No encryption.

Authentication specifies which encryption method to protect IPsec data flows:

- esp-md5-hmac—Uses the MD5/HMAC-128 as the hash algorithm.
- esp-sha-hmac—Uses the SHA/HMAC-160 as the hash algorithm.
- esp-none—No HMAC authentication.

For example:

```
hostname(config)# crypto ipsec ikev1 transform-set myset1 esp-des esp-sha-hmac
hostname(config)# crypto ipsec ikev1 transform-set myset2 esp-3des esp-sha-hmac
hostname(config)# crypto ipsec ikev1 transform-set aes_set esp-md5-hmac esp-aes-256
```

In this example, myset1 and myset2 and aes_set are the names of the transform sets.

To configure an IKEv2 proposal that also defines how to protect the traffic, enter the **crypto ipsec ikev2 ipsec-proposal** command to create the proposal and enter the ipsec proposal configuration mode where you can specify multiple encryption and integrity types for the proposal:

crypto ipsec ikev2 ipsec-proposal [*proposal tag*]
Proposal tag is the name of the IKEv2 IPsec proposal, a string from 1 to 64 characters.

For example:

```
hostname(config)# crypto ipsec ikev2 ipsec-proposal secure
```

In this example, secure is the name of the proposal. Enter a protocol and encryption types:

```
hostname(config-ipsec-proposal)# protocol esp encryption 3des aes des
```

Conversely, the following command chooses which AES-GCM or AES-GMAC algorithm to use:

```
hostname(config-ipsec-proposal)# [no] protocol esp encryption [3des | aes | aes-192 |
aes-256 | aes-gcm | aes-gcm-192 | aes-gcm-256 | aes-gmac | aes-gmac-192 | aes-gmac-256 |
des | null]
```

If SHA-2 or null is chosen, you must choose which algorithm to use as an IPsec integrity algorithm. You must choose the null integriy algorithm if AES-GCM/GMAC is configured as the encryption algorithm:

```
hostname(config-ipsec-proposal)# [no] protocol esp integrity [md5 | sha-1 | sha-256 |
sha-384 | sha-512 | null]
```

Note You must choose the null integrity algorithm if AES-GCM/GMAC has been configured as the encryption algorithm. SHA-256 can be used for integrity and PRF to establish IKEv2 tunnels, but it can also be used for ESP integrity protection on the newer ASA platforms (and not 5505, 5510, 5520, 5540, or 5550).

Step 3 (Optional) An administrator can enable path maximum transfer unit (PMTU) aging and set the interval at which the PMTU value is reset to its original value.

```
hostname(config-ipsec-proposal)# [no] crypto ipsec security-association pmtu-aging
<reset-interval>
```

Step 4 To create a crypto map, perform the following site-to-site steps using either single or multiple context mode:

a. Assign an ACL to a crypto map:

crypto map *map-name seq-num* **match address** *access-list-name*

A crypto map set is a collection of crypto map entries, each with a different sequence number (*seq-num)* but the same *map name*. Use the *access-list-name* to specify the ACL ID, as a string or integer up to 241 characters in length. In the following example, mymap is the name of the crypto map set. The map set sequence number 10, which is used to rank multiple entries within one crypto map set. The lower the sequence number, the higher the priority.

```
crypto map mymap 10 match address 101
```

In this example, the ACL named 101 is assigned to crypto map mymap.

b. Specify the peer to which the IPsec-protected traffic can be forwarded:

crypto map *map-name seq-num* **set peer** *ip-address*

For example:

```
crypto map mymap 10 set peer 192.168.1.100
```

The ASA sets up an SA with the peer assigned the IP address 192.168.1.100. Specify multiple peers by repeating this command.

c. Specify which IKEv1 transform sets or IKEv2 proposals are allowed for this crypto map. List multiple transform sets or proposals in order of priority (highest priority first). You can specify up to 11 transform sets or proposals in a crypto map using either of these two commands:

crypto map *map-name seq-num* **set ikev1 transform-set** *transform-set-name1 [transform-set-name2, …transform-set-name11]*

crypto map *map-name seq-num* **set ikev2 ipsec-proposal** *proposal-name1*

[*proposal-name2, … proposal-name11*]

Proposal-name1 and *proposal-name11* specifies one or more names of the IPsec proposals for IKEv2. Each crypto map entry supports up to 11 proposals.

For example (for IKEv1):

```
crypto map mymap 10 set ikev1 transform-set myset1 myset2
```

In this example, when traffic matches ACL 101, the SA can use either myset1 (first priority) or myset2 (second priority) depending on which transform set matches the transform set of the peer.

d. (Optional) Specify an SA lifetime for the crypto map if you want to override the global lifetime.

crypto map *map-name seq-num* **set security-association lifetime** {**seconds** seconds | **kilobytes** kilobytes}

Map-name specifies the name of the crypto map set. *Seq-num* specifies the number you assign to the crypto map entry.

For example:

```
crypto map mymap 10 set security-association lifetime seconds 2700
```

This example shortens the timed lifetime for the crypto map mymap 10 to 2700 seconds (45 minutes). The traffic volume lifetime is not changed.

e. (Optional) Specify that IPsec require perfect forward secrecy when requesting new SA for this crypto map, or require PFS in requests received from the peer:

crypto map map-name *seq-num* **set pfs** [**group1** | **group2** | **group5**]

For example:

```
crypto map mymap 10 set pfs group2
```

This example requires PFS when negotiating a new SA for the crypto map mymap 10. The ASA uses the 1024-bit Diffie-Hellman prime modulus group in the new SA.

Step 5 Apply a crypto map set to an interface for evaluating IPsec traffic:

crypto map *map-name* **interface** *interface-name*

Map-name specifies the name of the crypto map set. *Interface-name* specifies the name of the interface on which to enable or disable ISAKMP IKEv1 negotiation.

For example:

```
crypto map mymap interface outside
```

In this example, the ASA evaluates the traffic going through the outside interface against the crypto map mymap to determine whether it needs to be protected.

Using Dynamic Crypto Maps

A dynamic crypto map is a crypto map without all of the parameters configured. It acts as a policy template where the missing parameters are later dynamically learned, as the result of an IPsec negotiation, to match the peer requirements. The ASA applies a dynamic crypto map to let a peer negotiate a tunnel if its IP address is not already identified in a static crypto map. This occurs with the following types of peers:

- Peers with dynamically assigned public IP addresses.

  Both LAN-to-LAN and remote access peers can use DHCP to obtain a public IP address. The ASA uses this address only to initiate the tunnel.

- Peers with dynamically assigned private IP addresses.

  Peers requesting remote access tunnels typically have private IP addresses assigned by the headend. Generally, LAN-to-LAN tunnels have a predetermined set of private networks that are used to configure static maps and therefore used to establish IPsec SAs.

As an administrator configuring static crypto maps, you might not know the IP addresses that are dynamically assigned (via DHCP or some other method), and you might not know the private IP addresses of other clients, regardless of how they were assigned. VPN clients typically do not have static IP addresses; they require a dynamic crypto map to allow IPsec negotiation to occur. For example, the headend assigns the IP address to a Cisco VPN client during IKE negotiation, which the client then uses to negotiate IPsec SAs.

Note A dynamic crypto map requires only the **transform-set** parameter.

Dynamic crypto maps can ease IPsec configuration, and we recommend them for use in networks where the peers are not always predetermined. Use dynamic crypto maps for Cisco VPN clients (such as mobile users) and routers that obtain dynamically assigned IP addresses.

Tip Use care when using the **any** keyword in **permit** entries in dynamic crypto maps. If the traffic covered by such a **permit** entry could include multicast or broadcast traffic, insert **deny** entries for the appropriate address range into the ACL. Remember to insert **deny** entries for network and subnet broadcast traffic, and for any other traffic that IPsec should not protect.

Dynamic crypto maps work only to negotiate SAs with remote peers that initiate the connection. The ASA cannot use dynamic crypto maps to initiate connections to a remote peer. With a dynamic crypto map, if outbound traffic matches a permit entry in an ACL and the corresponding SA does not yet exist, the ASA drops the traffic.

A crypto map set may include a dynamic crypto map. Dynamic crypto map sets should be the lowest priority crypto maps in the crypto map set (that is, they should have the highest sequence numbers) so that the ASA evaluates other crypto maps first. It examines the dynamic crypto map set only when the other (static) map entries do not match.

Similar to static crypto map sets, a dynamic crypto map set consists of all of the dynamic crypto maps with the same dynamic-map-name. The dynamic-seq-num differentiates the dynamic crypto maps in a set. If you configure a dynamic crypto map, insert a permit ACL to identify the data flow of the IPsec peer for the crypto ACL. Otherwise the ASA accepts any data flow identity the peer proposes.

Caution Do not assign module default routes for traffic to be tunneled to a ASA interface configured with a dynamic crypto map set. To identify the traffic that should be tunneled, add the ACLs to the dynamic crypto map. Use care to identify the proper address pools when configuring the ACLs associated with remote access tunnels. Use Reverse Route Injection to install routes only after the tunnel is up.

The procedure for using a dynamic crypto map entry is the same as the basic configuration described in "Creating a Basic IPsec Configuration," except that instead of creating a static crypto map, you create a dynamic crypto map entry. You can also combine static and dynamic map entries within a single crypto map set.

Follow these steps to create a crypto dynamic map entry using either single or multiple context mode:

Step 1 (Optional) Assign an ACL to a dynamic crypto map:

```
crypto dynamic-map dynamic-map-name dynamic-seq-num match address access-list-name
```

This determines which traffic should be protected and not protected. *Dynamic-map-name* specifies the name of the crypto map entry that refers to a pre-existing dynamic crypto map. *Dynamic-seq-num* specifies the sequence number that corresponds to the dynamic crypto map entry.

For example:

```
crypto dynamic-map dyn1 10 match address 101
```

In this example, ACL 101 is assigned to dynamic crypto map dyn1. The map sequence number is 10.

Step 2 Specify which IKEv1 transform sets or IKEv2 proposals are allowed for this dynamic crypto map. List multiple transform sets or proposals in order of priority (highest priority first) using the command for IKEv1 transform sets or IKEv2 proposals:

```
crypto dynamic-map dynamic-map-name dynamic-seq-num set ikev1 transform-set
transform-set-name1, [transform-set-name2, …transform-set-name9]

crypto dynamic-map dynamic-map-name dynamic-seq-num set ikev2 ipsec-proposal
proposal-name1
[proposal-name2, … proposal-name11]
```

Dynamic-map-name specifies the name of the crypto map entry that refers to a pre-existing dynamic crypto map. *Dynamic-seq-num* specifies the sequence number that corresponds to the dynamic crypto map entry. The *transform-set-name* is the name of the transform-set being created or modified. The *proposal-name* specifies one or more names of the IPsec proposals for IKEv2.

For example (for IKEv1):

```
crypto dynamic-map dyn 10 set ikev1 transform-set myset1 myset2
```

In this example, when traffic matches ACL 101, the SA can use either myset1 (first priority) or myset2 (second priority), depending on which transform set matches the transform sets of the peer.

Step 3 (Optional) Specify the SA lifetime for the crypto dynamic map entry if you want to override the global lifetime value:

```
crypto dynamic-map dynamic-map-name dynamic-seq-num set security-association lifetime
{seconds seconds | kilobytes kilobytes}
```

Dynamic-map-name specifies the name of the crypto map entry that refers to a pre-existing dynamic crypto map. *Dynamic-seq-num* specifies the sequence number that corresponds to the dynamic crypto map entry.

For example:

```
crypto dynamic-map dyn1 10 set security-association lifetime seconds 2700
```

This example shortens the timed lifetime for dynamic crypto map dyn1 10 to 2700 seconds (45 minutes). The time volume lifetime is not changed.

Step 4 (Optional) Specify that IPsec ask for PFS when requesting new SAs for this dynamic crypto map, or should demand PFS in requests received from the peer:

```
crypto dynamic-map dynamic-map-name dynamic-seq-num set pfs [group1 | group2 | group5 |
group7]
```

Dynamic-map-name specifies the name of the crypto map entry that refers to a pre-existing dynamic crypto map. *Dynamic-seq-num* specifies the sequence number that corresponds to the dynamic crypto map entry.

For example:

```
crypto dynamic-map dyn1 10 set pfs group5
```

Step 5 Add the dynamic crypto map set into a static crypto map set.

Be sure to set the crypto maps referencing dynamic maps to be the lowest priority entries (highest sequence numbers) in a crypto map set.

crypto map *map-name seq-num* **ipsec-isakmp dynamic** *dynamic-map-name*

Map-name specifies the name of the crypto map set. *Dynamic-map-name* specifies the name of the crypto map entry that refers to a pre-existing dynamic crypto map.

For example:

```
crypto map mymap 200 ipsec-isakmp dynamic dyn1
```

Providing Site-to-Site Redundancy

You can define multiple IKEv1 peers by using crypto maps to provide redundancy. This configuration is useful for site-to-site VPNs. This feature is not supported with IKEv2.

If one peer fails, the ASA establishes a tunnel to the next peer associated with the crypto map. It sends data to the peer that it has successfully negotiated with, and that peer becomes the active peer. The active peer is the peer that the ASA keeps trying first for follow-on negotiations until a negotiation fails. At that point the ASA goes on to the next peer. The ASA cycles back to the first peer when all peers associated with the crypto map have failed.

Viewing an IPsec Configuration

Table 1-6 lists commands that you can enter in either single or multiple context mode to view information about your IPsec configuration.

Table 1-6 Commands to View IPsec Configuration Information

Command	Purpose
show running-configuration crypto	Displays the entire crypto configuration, including IPsec, crypto maps, dynamic crypto maps, and ISAKMP.
show running-config crypto ipsec	Displays the complete IPsec configuration.
show running-config crypto isakmp	Displays the complete ISAKMP configuration.
show running-config crypto map	Displays the complete crypto map configuration.
show running-config crypto dynamic-map	Displays the dynamic crypto map configuration.
show all crypto map	Displays all of the configuration parameters, including those with default values.

Table 1-6 Commands to View IPsec Configuration Information (continued)

Command	Purpose
show crypto ikev2 sa detail	Shows the Suite B algorithm support in the Encryption statistics.
show crypto ipsec sa	Shows the Suite B algorithm support and the ESPv3 IPsec output in either single or multiple context mode.
show ipsec stats	Shows information about the IPsec subsystem in either single or multiple context mode. ESPv3 statistics are shown in TFC packets and valid and invalid ICMP errors received.

Clearing Security Associations

Certain configuration changes take effect only during the negotiation of subsequent SAs. If you want the new settings to take effect immediately, clear the existing SAs to reestablish them with the changed configuration. If the ASA is actively processing IPsec traffic, clear only the portion of the SA database that the configuration changes affect. Reserve clearing the full SA database for large-scale changes, or when the ASA is processing a small amount of IPsec traffic.

Table 1-7 lists commands you can enter to clear and reinitialize IPsec SAs in either single or multiple context mode.

Table 1-7 Commands to Clear and Reinitialize IPsec SAs

Command	Purpose
clear configure crypto	Removes an entire crypto configuration, including IPsec, crypto maps, dynamic crypto maps, and ISAKMP.
clear configure crypto ca trustpoint	Removes all trustpoints.
clear configure crypto dynamic-map	Removes all dynamic crypto maps. Includes keywords that let you remove specific dynamic crypto maps.
clear configure crypto map	Removes all crypto maps. Includes keywords that let you remove specific crypto maps.
clear configure crypto isakmp	Removes the entire ISAKMP configuration.
clear configure crypto isakmp policy	Removes all ISAKMP policies or a specific policy.
clear crypto isakmp sa	Removes the entire ISAKMP SA database.

Clearing Crypto Map Configurations

The **clear configure crypto** command includes arguments that let you remove elements of the crypto configuration, including IPsec, crypto maps, dynamic crypto maps, CA trustpoints, all certificates, certificate map configurations, and ISAKMP.

Be aware that if you enter the **clear configure crypto** command without arguments, you remove the entire crypto configuration, including all certificates.

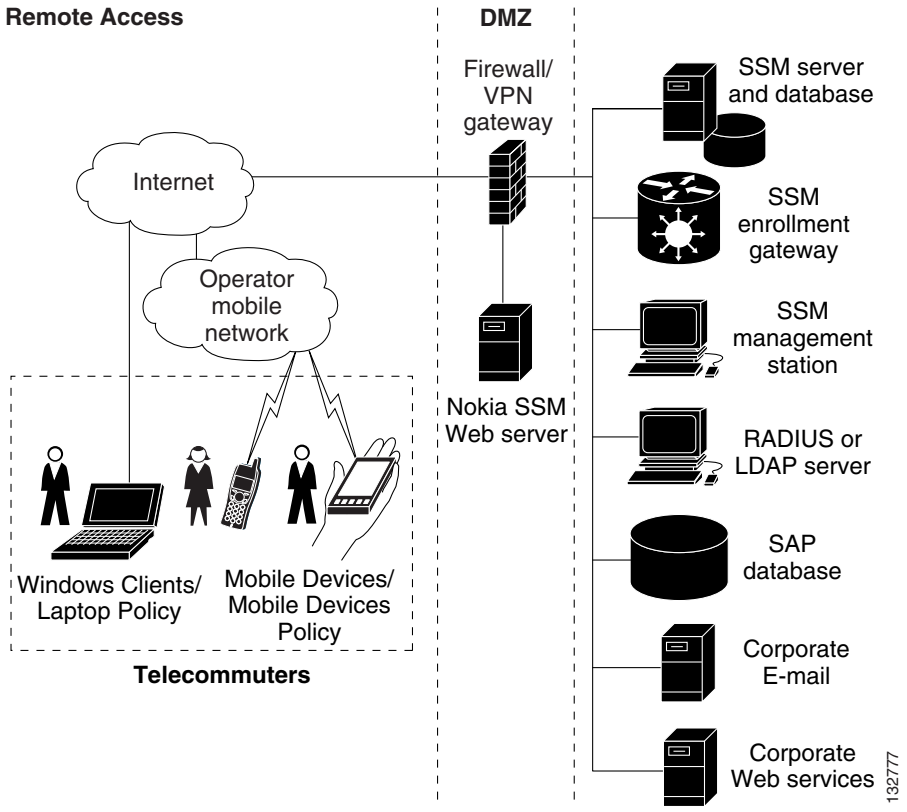
For more information, see the **clear configure crypto** command in the *Cisco ASA Series Command Reference*.

Supporting the Nokia VPN Client

The ASA supports connections from Nokia VPN clients on Nokia 92xx Communicator series phones using the Challenge/Response for Authenticated Cryptographic Keys (CRACK) protocol. CRACK is ideal for mobile IPsec-enabled clients that use legacy authentication techniques instead of digital certificates. It provides mutual authentication when the client uses a legacy-based secret-key authentication technique such as RADIUS and the gateway uses public-key authentication.

The Nokia back-end services must be in place to support both Nokia clients and the CRACK protocol. This requirement includes the Nokia Security Services Manager (NSSM) and Nokia databases as shown in Figure 1-5.

Figure 1-5 Nokia 92xx Communicator Service Requirement



To support the Nokia VPN client, perform the following step on the ASA:

- Enable CRACK authentication using the **crypto isakmp policy** *priority* **authentication** command with the **crack** keyword in global configuration mode. For example:

```
hostname(config)# crypto isakmp policy 2
hostname(config-isakmp-policy)# authentication crack
```

If you are using digital certificates for client authentication, perform the following additional steps:

Step 1 Configure the trustpoint and remove the requirement for a fully qualified domain name. The trustpoint might be NSSM or some other CA. In this example, the trustpoint is named CompanyVPNCA:

```
hostname(config)# crypto ca trustpoint CompanyVPNCA
hostname(config-ca-trustpoint)# fqdn none
```

Step 2 To configure the identity of the ISAKMP peer, perform one of the following steps:

- Use the **crypto isakmp identity** command with the **hostname** keyword. For example:

  ```
  hostname(config)# crypto isakmp identity hostname
  ```

- Use the **crypto isakmp identity** command with the **auto** keyword to configure the identity to be automatically determined from the connection type. For example:

  ```
  hostname(config)# crypto isakmp identity auto
  ```

Note If you use the **crypto isakmp identity auto** command, you must be sure that the DN attribute order in the client certificate is CN, OU, O, C, St, L.

To learn more about the Nokia services required to support the CRACK protocol on Nokia clients, and to ensure they are installed and configured properly, contact your local Nokia representative.

Configuring L2TP over IPsec

This chapter describes how to configure L2TP over IPsec/IKEv1 on the ASA. This chapter includes the following topics:

Information About L2TP over IPsec/IKEv1

Layer 2 Tunneling Protocol (L2TP) is a VPN tunneling protocol that allows remote clients to use the public IP network to securely communicate with private corporate network servers. L2TP uses PPP over UDP (port 1701) to tunnel the data.

L2TP protocol is based on the client/server model. The function is divided between the L2TP Network Server (LNS), and the L2TP Access Concentrator (LAC). The LNS typically runs on a network gateway such as a router, while the LAC can be a dial-up Network Access Server (NAS) or an endpoint device with a bundled L2TP client such as Microsoft Windows, Apple iPhone, or Android.

The primary benefit of configuring L2TP with IPsec/IKEv1 in a remote access scenario is that remote users can access a VPN over a public IP network without a gateway or a dedicated line, which enables remote access from virtually anyplace with POTS. An additional benefit is that no additional client software, such as Cisco VPN client software, is required.

Note L2TP over IPsec supports only IKEv1. IKEv2 is not supported.

The configuration of L2TP with IPsec/IKEv1 supports certificates using the preshared keys or RSA signature methods, and the use of dynamic (as opposed to static) crypto maps. This summary of tasks assumes completion of IKEv1, as well as pre-shared keys or RSA signature configuration. See Chapter 40, “Configuring Digital Certificates,” in the general operations configuration guide for the steps to configure preshared keys, RSA, and dynamic crypto maps.

Note L2TP with IPsec on the ASA allows the LNS to interoperate with native VPN clients integrated in such operating systems as Windows, MAC OS X, Android, and Cisco IOS. Only L2TP with IPsec is supported, native L2TP itself is not supported on ASA.

The minimum IPsec security association lifetime supported by the Windows client is 300 seconds. If the lifetime on the ASA is set to less than 300 seconds, the Windows client ignores it and replaces it with a 300 second lifetime.

IPsec Transport and Tunnel Modes

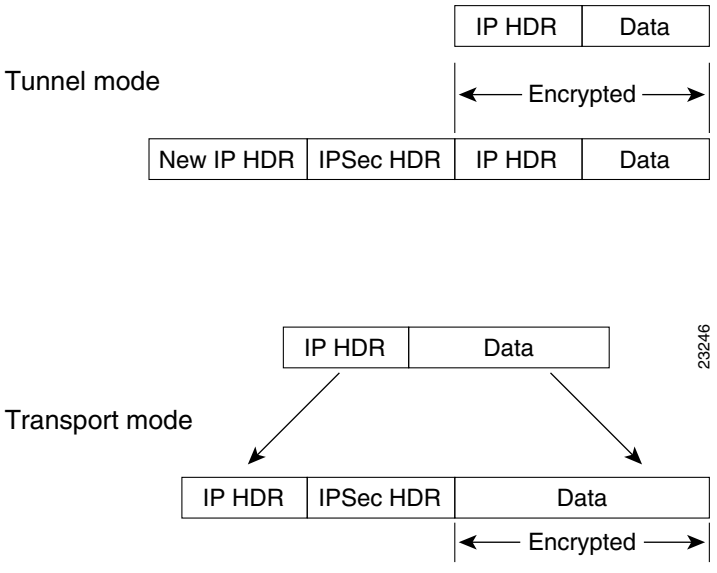
By default, the ASA uses IPsec tunnel mode—the entire original IP datagram is encrypted, and it becomes the payload in a new IP packet. This mode allows a network device, such as a router, to act as an IPsec proxy. That is, the router performs encryption on behalf of the hosts. The source router encrypts packets and forwards them along the IPsec tunnel. The destination router decrypts the original IP datagram and forwards it on to the destination system. The major advantage of tunnel mode is that the end systems do not need to be modified to receive the benefits of IPsec. Tunnel mode also protects against traffic analysis; with tunnel mode, an attacker can only determine the tunnel endpoints and not the true source and destination of the tunneled packets, even if they are the same as the tunnel endpoints.

However, the Windows L2TP/IPsec client uses IPsec transport mode—only the IP payload is encrypted, and the original IP headers are left intact. This mode has the advantages of adding only a few bytes to each packet and allowing devices on the public network to see the final source and destination of the packet. Figure 2-1 illustrates the differences between IPsec tunnel and transport modes.

In order for Windows L2TP and IPsec clients to connect to the ASA, you must configure IPsec transport mode for a transform set using the **crypto ipsec transform-set trans_name mode transport** command. This command is used in the configuration procedure**.**

With this transport capability, you can enable special processing (for example, QoS) on the intermediate network based on the information in the IP header. However, the Layer 4 header is encrypted, which limits the examination of the packet. Unfortunately, if the IP header is transmitted in clear text, transport mode allows an attacker to perform some traffic analysis.

Figure 2-1 IPsec in Tunnel and Transport Modes



Licensing Requirements for L2TP over IPsec

The following table shows the licensing requirements for this feature:

Note This feature is not available on No Payload Encryption models.

Model	License Requirement[1]
ASA 5505	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license and Security Plus license: 2 sessions. *Optional permanent or time-based licenses: 10 or 25 sessions.* *Shared licenses are not supported.*[2] – AnyConnect Essentials license[3]: 25 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: – Base license: 10 sessions. – Security Plus license: 25 sessions.
ASA 5510	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base and Security Plus license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses*[2]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license and Security Plus license: 250 sessions.
ASA 5520	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* *Optional Shared licenses*[2]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 750 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 750 sessions.

Model	License Requirement[1]
ASA 5540	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 2500 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 2500 sessions.
ASA 5550	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.
ASA 5580	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.

Model	License Requirement[1]
ASA 5512-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 250 sessions.
ASA 5515-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 250 sessions.
ASA 5525-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 750 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 750 sessions.
ASA 5545-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 2500 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 2500 sessions.

Model	License Requirement[1]
ASA 5555-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.
ASA 5585-X with SSP-10	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.

Model	License Requirement[1]
ASA 5585-X with SSP-20, -40, and -60	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.
ASA SM	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.

1. The maximum combined VPN sessions of *all* types cannot exceed the maximum sessions shown in this table. For the ASA 5505, the maximum combined sessions is 10 for the Base license, and 25 for the Security Plus license.
2. A shared license lets the security appliance act as a shared license server for multiple client security appliances. The shared license pool is large, but the maximum number of sessions used by each individual security appliance cannot exceed the maximum number listed for permanent licenses.
3. The AnyConnect Essentials license enables AnyConnect VPN client access to the security appliance. This license does not support browser-based SSL VPN access or Cisco Secure Desktop. For these features, activate an AnyConnect Premium license instead of the AnyConnect Essentials license.

   Note: With the AnyConnect Essentials license, VPN users can use a Web browser to log in, and download and start (WebLaunch) the AnyConnect client.

   The AnyConnect client software offers the same set of client features, whether it is enabled by this license or an AnyConnect Premium SSL VPN Edition license.

   The AnyConnect Essentials license cannot be active at the same time as the following licenses on a given security appliance: AnyConnect Premium license (all types) or the Advanced Endpoint Assessment license. You can, however, run AnyConnect Essentials and AnyConnect Premium licenses on different security appliances in the same network.

   By default, the security appliance uses the AnyConnect Essentials license, but you can disable it to use other licenses by using the **no anyconnect-essentials** command.

   For a detailed list of the features supported by the AnyConnect Essentials license and AnyConnect Premium license, see *AnyConnect Secure Mobility Client Features, Licenses, and OSs*:

   http://www.cisco.com/en/US/products/ps10884/products_feature_guides_list.html

Prerequisites for Configuring L2TP over IPsec

Configuring L2TP over IPsec has the following prerequisites:

- You can configure the default group policy (DfltGrpPolicy) or a user-defined group policy for L2TP/IPsec connections. In either case, the group policy must be configured to use the L2TP/IPsec tunneling protocol. If the L2TP/IPsec tunning protocol is not configured for your user-defined group policy, configure the DfltGrpPolicy for the L2TP/IPsec tunning protocol and allow your user-defined group policy to inherit this attribute.
- You need to configure the default connection proflie (tunnel group), DefaultRAGroup, if you are performing “pre-shared key” authentication. If you are performing certificate-based authentication, you can use a user-defined connection profile that can be chosen based on certificate identifiers.
- IP connectivity needs to be established between the peers. To test connectivity, try to ping the IP address of the ASA from your endpoint and try to ping the IP address of your endpoint from the ASA.
- Make sure that UDP port 1701 is not blocked anywhere along the path of the connection.
- If a Windows 7 endpoint device authenticates using a certificate that specifies a SHA signature type, the signature type must match that of the ASA, either SHA1 or SHA2.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

Supported in single context mode. Multiple context mode is not supported.

Firewall Mode Guidelines

Supported only in routed firewall mode. Transparent mode is not supported.

Failover Guidelines

L2TP over IPsec sessions are not supported by stateful failover.

IPv6 Guidelines

There is no native IPv6 tunnel setup support for L2TP over IPsec.

Authentication Guidelines

The ASA only supports the PPP authentications PAP and Microsoft CHAP, Versions 1 and 2, on the local database. EAP and CHAP are performed by proxy authentication servers. Therefore, if a remote user belongs to a tunnel group configured with the **authentication eap-proxy** or **authentication chap** commands, and the ASA is configured to use the local database, that user will not be able to connect.

Supported PPP Authentication Types

L2TP over IPsec connections on the ASA support only the PPP authentication types shown in Table 2-1.

Table 2-1 AAA Server Support and PPP Authentication Types

AAA Server Type	Supported PPP Authentication Types
LOCAL	PAP, MSCHAPv1, MSCHAPv2
RADIUS	PAP, CHAP, MSCHAPv1, MSCHAPv2, EAP-Proxy
TACACS+	PAP, CHAP, MSCHAPv1
LDAP	PAP
NT	PAP
Kerberos	PAP
SDI	SDI

Table 2-1 PPP Authentication Type Characteristics

Keyword	Authentication Type	Characteristics
chap	CHAP	In response to the server challenge, the client returns the encrypted [challenge plus password] with a cleartext username. This protocol is more secure than the PAP, but it does not encrypt data.
eap-proxy	EAP	Enables EAP which permits the security appliance to proxy the PPP authentication process to an external RADIUS authentication server.
ms-chap-v1 **ms-chap-v2**	Microsoft CHAP, Version 1 Microsoft CHAP, Version, 2	Similar to CHAP but more secure in that the server stores and compares only encrypted passwords rather than cleartext passwords as in CHAP. This protocol also generates a key for data encryption by MPPE.
pap	PAP	Passes cleartext username and password during authentication and is not secure.

Configuring L2TP over IPsec

This section provides the required ASA IKEv1 (ISAKMP) policy settings that allow native VPN clients, integrated with the operating system on an endpoint, to make a VPN connection to the ASA using L2TP over IPsec protocol.

- IKEv1 phase 1—3DES encryption with SHA1 hash method.
- IPsec phase 2—3DES or AES encryption with MD5 or SHA hash method.
- PPP Authentication—PAP, MS-CHAPv1, or MSCHAPv2 (preferred).
- Pre-shared key (only for iPhone).

Detailed CLI Configuration Steps

	Command	Purpose
Step 1	`crypto ipsec transform-set` *transform_name ESP_Encryption_Type ESP_Authentication_Type* **Example:** `hostname(config)# crypto ipsec transform-set my-transform-set esp-des esp-sha-hmac`	Creates a transform set with a specific ESP encryption type and authentication type.

Step 2	`crypto ipsec transform-set` *trans_name* `mode transport`
	
	Example:
	`hostname(config)# crypto ipsec transform-set my-transform-set mode transport`
		Instructs IPsec to use transport mode rather than tunnel mode.

Step 3	**vpn-tunnel-protocol** *tunneling_protocol*
	
	Example:
	`hostname(config)# group-policy DfltGrpPolicy attributes`
	`hostname(config-group-policy)# vpn-tunnel-protocol l2tp-ipsec`
		Specifies L2TP/IPsec as the vpn tunneling protocol.

Step 4	**dns value** [**none** | *IP_primary* [*IP_secondary*]
	
	Example:
	`hostname(config)# group-policy DfltGrpPolicy attributes`
	`hostname(config-group-policy)# dns value 209.165.201.1 209.165.201.2`
		(Optional) Instructs the adaptive security appliance to send DNS server IP addresses to the client for the group policy.

Step 5	**wins-server value** [**none** | *IP_primary* [*IP_secondary*]]
	
	Example:
	`hostname(config)# group-policy DfltGrpPolicy attributes`
	`hostname (config-group-policy)# wins-server value 209.165.201.3 209.165.201.4`
		(Optional) Instructs the adaptive security appliance to send WINS server IP addresses to the client for the group policy.

Step 6	**tunnel-group** *name* **type remote-access** **Example:** `hostname(config)# tunnel-group sales-tunnel type remote-access`	Creates a connection profile (tunnel group).

Step 7	**default-group-policy** *name*
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup general-attributes`
	`hostname(config-tunnel-general)# default-group-policy DfltGrpPolicy`
		Links the name of a group policy to the connection profile (tunnel group).

	Command	Purpose
Step 8	**ip local pool** *pool_name starting_address-ending_address* **mask** *subnet_mask* **Example:** `hostname(config)# ip local pool sales_addresses 10.4.5.10-10.4.5.20 mask 255.255.255.0`	(Optional) Creates an IP address pool.

Step 9	**address-pool** *pool_name*
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup general-attributes`
	`hostname(config-tunnel-general)# address-pool sales_addresses`
		(Optional) Associates the pool of IP addresses with the connection profile (tunnel group).

Step 10	**authentication-server-group** *server_group*
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup general-attributes`
	`hostname(config-tunnel-general)# authentication-server-group sales_server LOCAL`
		Specifies a method to authenticate users attempting L2TP over IPsec connections, for the connection profile (tunnel group). If you are not using the ASA to perform local authentication, and you want to fallback to local authentication, add LOCAL to the end of the command.

Step 11	`authentication auth_type`
	
	Example:
	`hostname(config)# tunnel-group name ppp-attributes`
	`hostname(config-ppp)# authentication ms-chap-v1`
		Specifies the PPP authentication protocol for the tunnel group. See Table 2-1 for the types of PPP authencation and their characteristics.

Step 12	**tunnel-group** *tunnel group name* **ipsec-attributes**
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup ipsec-attributes`
	`hostname(config-tunnel-ipsec)# pre-shared-key cisco123`
		Sets the pre-shared key for your connection profile (tunnel group).

Step 13	**accounting-server-group** *aaa_server_group*
	
	Example:
	`hostname(config)# tunnel-group sales_tunnel general-attributes`
	`hostname(config-tunnel-general)# accounting-server-group sales_aaa_server`
		(Optional) Generates a AAA accounting start and stop record for an L2TP session for the connection profile (tunnel group).

Step 14	**l2tp tunnel hello** *seconds* **Example:** `hostname(config)# l2tp tunnel hello 100`	Configures the interval (in seconds) between hello messages. The range is 10 through 300 seconds. The default is 60 seconds.

	Command	Purpose

Step 15	**crypto isakmp nat-traversal** *seconds*
	
	Example:
	`hostname(config)# crypto isakmp enable`
	`hostname(config)# crypto isakmp nat-traversal 1500`
		(Optional) Enables NAT traversal so that ESP packets can pass through one or more NAT devices.
		
		If you expect multiple L2TP clients behind a NAT device to attempt L2TP over IPsec connections to the adaptive security appliance, you must enable NAT traversal.
		
		To enable NAT traversal globally, check that ISAKMP is enabled (you can enable it with the **crypto isakmp enable** command) in global configuration mode, and then use the **crypto isakmp nat-traversal** command.

Step 16	`strip-group`
	`strip-realm`
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup general-attributes`
	`hostname(config-tunnel-general)# strip-group`
	`hostname(config-tunnel-general)# strip-realm`
		(Optional) Configures tunnel group switching. The goal of tunnel group switching is to give users a better chance at establishing a VPN connection when they authenticate using a proxy authentication server. Tunnel group is synonymous with connection profile.

Step 17	**username** *name* **password** *password* **mschap** **Example:** hostname`(config)# username jdoe password j!doe1 mschap`	This example shows creating a user with the username `jdoe`, the password `j!doe1`. The mschap option specifies that the password is converted to Unicode and hashed using MD4 after you enter it. This step is needed only if you are using a local user database.
Step 18	**crypto isakmp policy** *priority* **Example:** hostname(config)# crypto isakmp policy 5	The crypto isakmp policy command creates the IKE Policy for Phsase 1 and assigns it a number. There are several different configurable parameters of the IKE policy that you can configure. The isakamp policy is needed so the ASA can complete the IKE negotiation. See the "Creating IKE Policies to Respond to Windows 7 Proposals" section on page 2-12 for configuration examples for Windows 7 native VPN clients.

Creating IKE Policies to Respond to Windows 7 Proposals

Windows 7 L2TP/IPsec clients send several IKE policy proposals to establish a VPN connection with the ASA. Define one of the following IKE policies to facilitate connections from Windows 7 VPN native clients.

	Command	Purpose
Step 1	Detailed CLI Configuration Steps, page 2-10	Follow the Detailed CLI Configuration Steps procedure through step Step 18. Add the additional steps in this table to configure the IKE policy for Windows 7 native VPN clients.
Step 1	**show run crypto isakmp** **Example:** `hostname(config)# show run crypto isakmp`	Displays the attributes and the number of any existing IKE policies.

Step 2	**crypto isakmp policy** number
	
	Example:
	`hostname(config)# crypto isakmp policy number`
	`hostname(config-isakmp-policy)#`
		Allows you to configure an IKE policy. The number argument specifies the number of the IKE policy you are configuring. This number was listed in the output of the `show run crypto isakmp` command.

Step 3	**authentication** **Example:** `hostname(config-isakmp-policy)# authentication pre-share`	Sets the authentication method the ASA uses to establish the identity of each IPsec peer to use preshared keys.
Step 4	**encryption** *type* **Example:** `hostname(config-isakmp-policy)# encryption {3des|aes|aes-256}`	Choose a symmetric encryption method that protects data transmitted between two IPsec peers. For Windows 7 choose either **3des**, **aes**, for 128-bit AES, or **aes-256**.
Step 5	**hash** **Example:** `hostname(config-isakmp-policy)# hash sha`	Choose the hash algorithm that ensures data integrity. For Windows 7, specify **sha** for the SHA-1 algorithm.

Step 6	`group`
	
	Example:
	`hostname(config-isakmp-policy)# group 5`
		Choose the Diffie-Hellman group identifier. For Windows 7, specify 5 for the 1536-bit Diffie-Hellman group.

Step 7	`lifetime`
	
	Example:
	`hostname(config-isakmp-policy)# lifetime 86400`
		Specify the SA lifetime in seconds. For Windows 7, specify 86400 seconds to represent 24 hours.

Detailed CLI Configuration Steps

	Command	Purpose
Step 1	`crypto ipsec` *ike_version* `transform-set` *transform_name ESP_Encryption_Type ESP_Authentication_Type* **Example:** `crypto ipsec ikev1 transform-set my-transform-set-ikev1 esp-des esp-sha-hmac`	Creates a transform set with a specific ESP encryption type and authentication type.

Step 2	`crypto ipsec` *ike_version* `transform-set` *trans_name* `mode transport`
	
	Example:
	`crypto ipsec ikev1 transform-set my-transform-set-ikev1 mode transport`
		Instructs IPsec to use transport mode rather than tunnel mode.

Step 3	**vpn-tunnel-protocol** *tunneling_protocol*
	
	Example:
	`hostname(config)# group-policy DfltGrpPolicy attributes`
	`hostname(config-group-policy)# vpn-tunnel-protocol l2tp-ipsec`
		Specifies L2TP/IPsec as the vpn tunneling protocol.

Step 4	**dns value** [**none** | *IP_primary* [*IP_secondary*]
	
	Example:
	`hostname(config)# group-policy DfltGrpPolicy attributes`
	`hostname(config-group-policy)# dns value 209.165.201.1 209.165.201.2`
		(Optional) Instructs the adaptive security appliance to send DNS server IP addresses to the client for the group policy.

Step 5	**wins-server value** [**none** | *IP_primary* [*IP_secondary*]]
	
	Example:
	`hostname(config)# group-policy DfltGrpPolicy attributes`
	`hostname (config-group-policy)# wins-server value 209.165.201.3 209.165.201.4`
		(Optional) Instructs the adaptive security appliance to send WINS server IP addresses to the client for the group policy.

Step 6	**ip local pool** *pool_name starting_address-ending_address* **mask** *subnet_mask* **Example:** `hostname(config)# ip local pool sales_addresses 10.4.5.10-10.4.5.20 mask 255.255.255.0`	(Optional) Creates an IP address pool.

Step 7	**address-pool** *pool_name*
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup general-attributes`
	`hostname(config-tunnel-general)# address-pool sales_addresses`
		(Optional) Associates the pool of IP addresses with the connection profile (tunnel group).

	Command	Purpose
Step 8	**tunnel-group** *name* **type remote-access** **Example:** `hostname(config)# tunnel-group sales-tunnel type remote-access`	Creates a connection profile (tunnel group).

Step 9	**default-group-policy** *name*
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup general-attributes`
	`hostname(config-tunnel-general)# default-group-policy DfltGrpPolicy`
		Links the name of a group policy to the connection profile (tunnel group).

Step 10	**authentication-server-group** *server_group* [local]
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup general-attributes`
	`hostname(config-tunnel-general)# authentication-server-group sales_server LOCAL`
		Specifies a method to authenticate users attempting L2TP over IPsec connections, for the connection profile (tunnel group). If you are not using the ASA to perform local authentication, and you want to fallback to local authentication, add LOCAL to the end of the command.

Step 11	**authentication** *auth_type*
	
	Example:
	`hostname(config)# tunnel-group` *name* `ppp-attributes`
	`hostname(config-ppp)# authentication ms-chap-v1`
		Specifies the PPP authentication protocol for the tunnel group. See Table 2-1 for the types of PPP authencation and their characteristics.

Step 12	**tunnel-group** *tunnel group name* **ipsec-attributes** **Example:** `hostname(config)# tunnel-group DefaultRAGroup ipsec-attributes` `hostname(config-tunnel-ipsec)# ikev1 pre-shared-key` *cisco123*	Sets the pre-shared key for your connection profile (tunnel group).

Step 13	**accounting-server-group** *aaa_server_group*
	
	Example:
	`hostname(config)# tunnel-group sales_tunnel general-attributes`
	`hostname(config-tunnel-general)# accounting-server-group sales_aaa_server`
		(Optional) Generates a AAA accounting start and stop record for an L2TP session for the connection profile (tunnel group).

Step 14	**l2tp tunnel hello** *seconds* **Example:** `hostname(config)# l2tp tunnel hello 100`	Configures the interval (in seconds) between hello messages. The range is 10 through 300 seconds. The default interval is 60 seconds.

	Command	Purpose

Step 15	**crypto isakmp nat-traversal** *seconds*
	
	Example:
	`hostname(config)# crypto isakmp enable`
	`hostname(config)# crypto isakmp nat-traversal 1500`
		(Optional) Enables NAT traversal so that ESP packets can pass through one or more NAT devices.
		
		If you expect multiple L2TP clients behind a NAT device to attempt L2TP over IPsec connections to the adaptive security appliance, you must enable NAT traversal.
		
		To enable NAT traversal globally, check that ISAKMP is enabled (you can enable it with the **crypto isakmp enable** command) in global configuration mode, and then use the **crypto isakmp nat-traversal** command.

Step 16	`strip-group`
	`strip-realm`
	
	Example:
	`hostname(config)# tunnel-group DefaultRAGroup general-attributes`
	`hostname(config-tunnel-general)# strip-group`
	`hostname(config-tunnel-general)# strip-realm`
		(Optional) Configures tunnel group switching. The goal of tunnel group switching is to give users a better chance at establishing a VPN connection when they authenticate using a proxy authentication server. Tunnel group is synonymous with connection profile.

Step 17	**username** *name* **password** *password* **mschap** **Example:** `asa2(config)# username jdoe password j!doe1 mschap`	This example shows creating a user with the username `jdoe`, the password `j!doe1`. The mschap option specifies that the password is converted to Unicode and hashed using MD4 after you enter it. This step is needed only if you are using a local user database.
Step 18	**crypto ikev1 policy** *priority* **group** *Diffie-Hellman Group* **Example:** `hostname(config)# crypto ikev1 policy 5` `hostname`(config-ikev1-policy)# group 5	The crypto isakmp policy command creates the IKE Policy for Phase 1 and assigns it a number. There are several different configurable parameters of the IKE policy that you can configure. You can also specify a Diffie-Hellman Group for the policy. The isakamp policy is needed so the ASA can complete the IKE negotiation. See the "Creating IKE Policies to Respond to Windows 7 Proposals" section on page 2-16 for configuration examples for Windows 7 native VPN clients.

Creating IKE Policies to Respond to Windows 7 Proposals

Windows 7 L2TP/IPsec clients send several IKE policy proposals to establish a VPN connection with the ASA. Define one of the following IKE policies to facilitate connections from Windows 7 VPN native clients.

	Command	Purpose
Step 1	Detailed CLI Configuration Steps, page 2-14	Follow the Detailed CLI Configuration Steps procedure through step Step 18. Add the additional steps in this table to configure the IKE policy for Windows 7 native VPN clients.
Step 1	**show run crypto ikev1** **Example:** `hostname(config)# show run crypto ikev1`	Displays the attributes and the number of any existing IKE policies.
Step 2	**crypto ikev1 policy** number **Example:** `hostname(config)# crypto ikev1 policy` *number* `hostname(config-ikev1-policy)#`	Allows you to configure an IKE policy. The number argument specifies the number of the IKE policy you are configuring. This number was listed in the output of the `show run crypto ikev1` command.
Step 3	**authentication** **Example:** `hostname(config-ikev1-policy)# authentication pre-share`	Sets the authentication method the ASA uses to establish the identity of each IPsec peer to use preshared keys.
Step 4	**encryption** *type* **Example:** `hostname(config-ikev1-policy)# encryption {3des|aes|aes-256}`	Choose a symmetric encryption method that protects data transmitted between two IPsec peers. For Windows 7 choose either **3des**, **aes**, for 128-bit AES, or **aes-256**.
Step 5	**hash** **Example:** `hostname(config-ikev1-policy)# hash sha`	Choose the hash algorithm that ensures data integrity. For Windows 7, specify **sha** for the SHA-1 algorithm.

Step 6	`group`
	
	Example:
	`hostname(config-ikev1-policy)# group 5`
		Choose the Diffie-Hellman group identifier. You can specify 5 for aes, aes-256, or 3des encryption types. You can specify 2 only for 3des encryption types.

Step 7	`lifetime`
	
	Example:
	`hostname(config-ikev1-policy)# lifetime 86400`
		Specify the SA lifetime in seconds. For Windows 7, specify 86400 seconds to represent 24 hours.

Configuration Example for L2TP over IPsec Using ASA 8.2.5

The following example shows configuration file commands that ensure ASA compatibility with a native VPN client on any operating system:

```
ip local pool sales_addresses 209.165.202.129-209.165.202.158
group-policy sales_policy internal
group-policy sales_policy attributes
    wins-server value 209.165.201.3 209.165.201.4
    dns-server value 209.165.201.1 209.165.201.2
```

```
    vpn-tunnel-protocol l2tp-ipsec
tunnel-group DefaultRAGroup general-attributes
    default-group-policy sales_policy
    address-pool sales_addresses
tunnel-group DefaultRAGroup ipsec-attributes
    pre-shared-key *
tunnel-group DefaultRAGroup ppp-attributes
    no authentication pap
    authentication chap
    authentication ms-chap-v1
    authentication ms-chap-v2
crypto ipsec transform-set trans esp-3des esp-sha-hmac
crypto ipsec transform-set trans mode transport
crypto dynamic-map dyno 10 set transform-set set trans
crypto map vpn 20 ipsec-isakmp dynamic dyno
crypto map vpn interface outside
crypto isakmp enable outside
crypto isakmp policy 10
    authentication pre-share
    encryption 3des
    hash sha
    group 2
    lifetime 86400
```

Configuration Example for L2TP over IPsec Using ASA 8.4.1 and later

The following example shows configuration file commands that ensure ASA compatibility with a native VPN client on any operating system:

```
ip local pool sales_addresses 209.165.202.129-209.165.202.158
group-policy sales_policy internal
group-policy sales_policy attributes
    wins-server value 209.165.201.3 209.165.201.4
    dns-server value 209.165.201.1 209.165.201.2
    vpn-tunnel-protocol l2tp-ipsec
tunnel-group DefaultRAGroup general-attributes
    default-group-policy sales_policy
    address-pool sales_addresses
tunnel-group DefaultRAGroup ipsec-attributes
    pre-shared-key *
tunnel-group DefaultRAGroup ppp-attributes
    no authentication pap
    authentication chap
    authentication ms-chap-v1
    authentication ms-chap-v2
crypto ipsec ikev1 transform-set my-transform-set-ikev1 esp-des esp-sha-hmac
crypto ipsec ikev1 transform-set my-transform-set-ikev1 mode transport
crypto dynamic-map dyno 10 set ikev1 transform-set trans
crypto map vpn 20 ipsec-isakmp dynamic dyno
crypto map vpn interface outside
crypto ikev1 enable outside
crypto ikev1 policy 10
    authentication pre-share
    encryption 3des
    hash sha
    group 2
    lifetime 86400
```

Feature History for L2TP over IPsec

Table 2-2 lists the release history for this feature.

Table 2-2 Feature History for L2TP over IPsec

Feature Name	Releases	Feature Information
L2TP over IPsec	7.2(1)	L2TP over IPsec provides the capability to deploy and administer an L2TP VPN solution alongside the IPsec VPN and firewall services in a single platform. The primary benefit of configuring L2TP over IPsec in a remote access scenario is that remote users can access a VPN over a public IP network without a gateway or a dedicated line, which enables remote access from virtually anyplace with POTS. An additional benefit is that the only client requirement for VPN access is the use of Windows with Microsoft Dial-Up Networking (DUN). No additional client software, such as Cisco VPN client software, is required. The following commands were introduced or modified: **authentication eap-proxy**, **authentication ms-chap-v1**, **authentication ms-chap-v2**, **authentication pap**, **l2tp tunnel hello**, **vpn-tunnel-protocol l2tp-ipsec**.

Setting General VPN Parameters

The ASA implementation of virtual private networking includes useful features that do not fit neatly into categories. This chapter describes some of these features. It includes the following sections:

Note SSL VPN in this chapter refers to the SSL VPN client (AnyConnect 2.x or its predecessor, SVC 1.x), unless clientless (browser-based) SSL VPN is specified.

Configuring IPsec to Bypass ACLs

To permit any packets that come from an IPsec tunnel without checking ACLs for the source and destination interfaces, enter the **sysopt connection permit-vpn** command in global configuration mode.

You might want to bypass interface ACLs for IPsec traffic if you use a separate VPN concentrator behind the ASA and want to maximize the ASA performance. Typically, you create an ACL that permits IPsec packets by using the **access-list** command and apply it to the source interface. Using an ACL is more secure because you can specify the exact traffic you want to allow through the ASA.

The syntax is **sysopt connection permit-vpn**. The command has no keywords or arguments.

The following example enables IPsec traffic through the ASA without checking ACLs:

```
ciscoasa(config)# sysopt connection permit-vpn
```

Note Decrypted through-traffic is permitted from the client despite having an access group on the outside interface, which calls a **deny ip any any** ACL, while **no sysopt connection permit-vpn** is configured.

Users who want to control access to the protected network via site-to-site or remote access VPN using

the **no sysopt permit-vpn** command in conjunction with an access control list (ACL) on the outside interface are not successful.

In this situation, when management-access inside is enabled, the ACL is not applied, and users can still connect to the ASA using SSH. Traffic to hosts on the inside network is blocked correctly by the ACL, but decrypted through-traffic to the inside interface is not blocked.

The **ssh** and **http** commands are of a higher priority than the ACLs. In other words, to deny SSH, Telnet, or ICMP traffic to the box from the VPN session, use **ssh**, **telnet** and **icmp** commands.

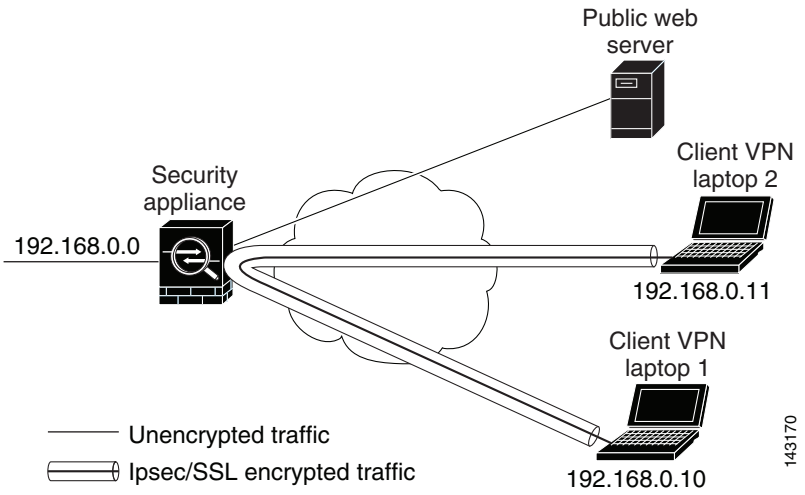
Permitting Intra-Interface Traffic (Hairpinning)

The ASA includes a feature that lets a VPN client send IPsec-protected traffic to another VPN user by allowing such traffic in and out of the same interface. Also called “hairpinning”, this feature can be thought of as VPN spokes (clients) connecting through a VPN hub (ASA).

In another application, hairpinning can redirect incoming VPN traffic back out through the same interface as unencrypted traffic. This would be useful, for example, to a VPN client that does not have split tunneling but needs to both access a VPN and browse the web.

Figure 3-1 shows VPN Client 1 sending secure IPsec traffic to VPN Client 2 while also sending unencrypted traffic to a public web server.

Figure 3-1 VPN Client Using Intra-Interface Feature for Hairpinning



To configure this feature, use the **same-security-traffic** command in global configuration mode with its **intra-interface** argument.

The command syntax is **same-security-traffic permit** {**inter-interface** | **intra-interface**}.

The following example shows how to enable intra-interface traffic:

```
ciscoasa(config)# same-security-traffic permit intra-interface
ciscoasa(config)#
```

Note You use the **same-security-traffic** command, but with the **inter-interface** argument, to permit communication between interfaces that have the same security level. This feature is not specific to IPsec connections. For more information, see the "Configuring Interface Parameters" chapter of this guide.

To use hairpinning, you must apply the proper NAT rules to the ASA interface, as discussed in the following section.

NAT Considerations for Intra-Interface Traffic

For the ASA to send unencrypted traffic back out through the interface, you must enable NAT for the interface so that publicly routable addresses replace your private IP addresses (unless you already use public IP addresses in your local IP address pool). The following example applies an interface PAT rule to traffic sourced from the client IP pool:

```
hostname(config)# ip local pool clientpool 192.168.0.10-192.168.0.100
hostname(config)# object network vpn_nat
hostname(config-network-object)# subnet 192.168.0.0 255.255.255.0
hostname(config-network-object)# nat (outside,outside) interface
```

When the ASA sends encrypted VPN traffic back out this same interface, however, NAT is optional. The VPN-to-VPN hairpinning works with or without NAT. To apply NAT to all outgoing traffic, implement only the commands above. To exempt the VPN-to-VPN traffic from NAT, add commands (to the example above) that implement NAT exemption for VPN-to-VPN traffic, such as:

```
hostname(config)# nat (outside,outside) source static vpn_nat vpn_nat destination static
vpn_nat vpn_nat
```

For more information on NAT rules, see the "Applying NAT" chapter of this guide.

Setting Maximum Active IPsec or SSL VPN Sessions

To limit VPN sessions to a lower value than the ASA allows, enter the **vpn-sessiondb** command in global configuration mode:

vpn-sessiondb {**max-anyconnect-premium-or-essentials-limit** <*number*> |
max-other-vpn-limit <*number*>}

The max-anyconnect-premium-or-essentials-limit keyword specifies the maximum number of AnyConnect sessions, from 1 to the maximum sessions allowed by the license.

The **max-other-vpn-limit** keyword specifies the maximum number of VPN sessions other than AnyConnect client sessions, from 1 to the maximum sessions allowed by the license. This includes the Cisco VPN client (IPsec IKEv1), Lan-to-Lan VPN, and clientless SSL VPN sessions.

This limit affects the calculated load percentage for VPN Load Balancing.

The following example shows how to set a maximum Anyconnect VPN session limit of 450:

```
ciscoasa(config)# vpn-sessiondb max-anyconnect-premium-or-essentials-limit 450
ciscoasa(config)#
```

Using Client Update to Ensure Acceptable IPsec Client Revision Levels

Note The information in this section applies to IPsec connections only.

The client update feature lets administrators at a central location automatically notify VPN client users that it is time to update the VPN client software and the VPN 3002 hardware client image.

Remote users might be using outdated VPN software or hardware client versions. You can use the **client-update** command at any time to enable updating client revisions; specify the types and revision numbers of clients to which the update applies; provide a URL or IP address from which to get the update; and, in the case of Windows clients, optionally notify users that they should update their VPN client version. For Windows clients, you can provide a mechanism for users to accomplish that update. For VPN 3002 hardware client users, the update occurs automatically, with no notification. This command applies only to the IPsec remote-access tunnel-group type.

To perform a client update, enter the **client-update** command in either general configuration mode or tunnel-group ipsec-attributes configuration mode. If the client is already running a software version on the list of revision numbers, it does not need to update its software. If the client is not running a software version on the list, it should update. The following procedure explains how to perform a client update:

Step 1 In global configuration mode, enable client update by entering this command:

```
ciscoasa(config)# client-update enable
ciscoasa(config)#
```

Step 2 In global configuration mode, specify the parameters for the client update that you want to apply to all clients of a particular type. That is, specify the type of client, the URL or IP address from which to get the updated image, and the acceptable revision number or numbers for that client. You can specify up to four revision numbers, separated by commas.

If the user's client revision number matches one of the specified revision numbers, there is no need to update the client. This command specifies the client update values for all clients of the specified type across the entire ASA.

Use this syntax:

```
ciscoasa(config)# client-update type type url url-string rev-nums rev-numbers
ciscoasa(config)#
```

The available client types are **win9X** (includes Windows 95, Windows 98 and Windows ME platforms), **winnt** (includes Windows NT 4.0, Windows 2000 and Windows XP platforms), **windows** (includes all Windows based platforms), and **vpn3002** (VPN 3002 hardware client).

If the client is already running a software version on the list of revision numbers, it does not need to update its software. If the client is not running a software version on the list, it should update. You can specify up to three of these client update entries. The keyword **windows** covers all of the allowable Windows platforms. If you specify **windows**, do not specify the individual Windows client types.

Note For all Windows clients, you must use the protocol http:// or https:// as the prefix for the URL. For the VPN 3002 hardware client, you must specify protocol tftp:// instead.

The following example configures client update parameters for the remote access tunnel group. It designates the revision number 4.6.1 and the URL for retrieving the update, which is https://support/updates.

```
ciscoasa(config)# client-update type windows url https://support/updates/ rev-nums 4.6.1
ciscoasa(config)#
```

Alternatively, you can configure client update just for individual tunnel groups, rather than for all clients of a particular type. (See Step 3.)

VPN 3002 clients update without user intervention and users receive no notification message. The following example applies only to VPN 3002 hardware clients. Entered in tunnel-group ipsec-attributes configuration mode the command it configures client update parameters for the IPsec remote access tunnel group **salesgrp**. This example designates the revision number, 4.7 and uses the TFTP protocol for retrieving the updated software from the site with the IP address 192.168.1.1:

```
ciscoasa(config)# tunnel-group salesgrp type ipsec-ra
ciscoasa(config)# tunnel-group salesgrp ipsec-attributes
ciscoasa(config-tunnel-ipsec)# client-update type vpn3002 url tftp:192.168.1.1 rev-nums
4.7
ciscoasa(config-tunnel-ipsec)#
```

Note You can have the browser automatically start an application by including the application name at the end of the URL; for example: **https://support/updates/vpnclient.exe.**

Step 3 Define a set of client-update parameters for a particular ipsec-ra tunnel group.

In tunnel-group ipsec-attributes mode, specify the tunnel group name and its type, the URL or IP address from which to get the updated image, and a revision number. If the user's client's revision number matches one of the specified revision numbers, there is no need to update the client, for example, for a Windows client enter this command:

```
ciscoasa(config)# tunnel-group remotegrp type ipsec-ra
ciscoasa(config)# tunnel-group remotegrp ipsec-attributes
ciscoasa(config-tunnel-ipsec)# client-update type windows url https://support/updates/
rev-nums 4.6.1
ciscoasa(config-tunnel-ipsec)#
```

Step 4 (Optional) Send a notice to active users with outdated Windows clients that their client needs updating. For these users, a pop-up window appears, offering them the opportunity to launch a browser and download the updated software from the site that you specified in the URL. The only part of this message that you can configure is the URL. (See Step 2 or 3.) Users who are not active get a notification message the next time they log on. You can send this notice to all active clients on all tunnel groups, or you can send it to clients on a particular tunnel group. For example, to notify all active clients on all tunnel groups, enter the following command in privileged EXEC mode:

```
ciscoasa# client-update all
ciscoasa#
```

If the user's client's revision number matches one of the specified revision numbers, there is no need to update the client, and no notification message is sent to the user. VPN 3002 clients update without user intervention and users receive no notification message.

Note If you specify the client-update type as **windows** (specifying all Windows-based platforms) and later want to enter a client-update type of **win9x** or **winnt** for the same entity, you must first remove the windows client type with the **no** form of the command, then use new client-update commands to specify the new client types.

Implementing NAT-Assigned IP to Public IP Connection

In rare situations, you might want to use a VPN peer's real IP address on the inside network instead of an assigned local IP address. Normally with VPN, the peer is given an assigned local IP address to access the inside network. However, you might want to translate the local IP address back to the peer-s real public address if, for example, your inside servers and network security is based on the peer's real IP address.

Cisco ASA 55xx introduced a way to translate the VPN client's assigned IP address on the internal/protected network to its public (source) IP address. This feature supports the scenario where the target servers/services on the internal network and network security policy require communication with the VPN client's public/source IP instead of the assigned IP on the internal corporate network.

You can enable this feature on one interface per tunnel group. Object NAT rules are dynamically added and deleted when the VPN session is established or disconnected.

Limitations

Because of routing issues, we do not recommend using this feature unless you know you need it.

- Only supports legacy Cisco VPN client (IKEv1) and AnyConnect clients.
- Return traffic to the public IP addresses must be routed back to the ASA so the NAT policy and VPN policy can be applied.
- Only supports IPv4 assigned and public addresses.
- Multiple peers behind a NAT/PAT device are not supported.
- Does not support load balancing (because of routing issue).
- Does not support roaming.

Detailed Steps

Step 1 In global configuration mode, enter **tunnel general**.

Step 2 Use this syntax to enable the address translation:

```
hostname(config-tunnel-general)# nat-assigned-to-public-ip <interface>
```

This command dynamically installs NAT policies of the assigned IP address to the public IP address of the source. The *interface* determines where to apply NAT.

Step 3 Use this syntax to disable the address translation:

```
hostname(config-tunnel-general)# no nat-assigned-to-public-ip
```

Displaying VPN NAT Policies

Address translation uses the underlying object NAT mechanisms; therefore, the VPN NAT policy displays just like manually configured object NAT policies. This example uses 95.1.226.4 as the assigned IP and 75.1.224.21 as the peer's public IP:

```
prompt# show nat
Auto NAT Policies (Section 2)
1 (outside) to (inside) source static _vpn_nat_95.1.226.4 75.1.224.21
    translate_hits = 315, untranslate_hits = 315

prompt# show nat detail

Auto NAT Policies (Section 2)
1 (outside) to (inside) source static _vpn_nat_95.1.226.4 75.1.224.21
    translate_hits = 315, untranslate_hits = 315
    Source - Origin: 95.1.226.4/32, Translated: 75.1.224.21/32
```

Outside is the interface to which the AnyConnect client connects and *inside* is the interface specific to the new tunnel group.

Note Since VPN NAT policies are dynamic and not added to the configuration, the VPN NAT object and NAT policy are hidden from the show run object and show run nat reports.

Understanding Load Balancing

If you have a remote-access configuration in which you are using two or more ASAs or VPN Concentrators connected on the same network, you can configure these devices to share their session load. This feature is called *load balancing*. To implement load balancing, you group together logically two or more devices on the same private LAN-to-LAN network, private subnet, and public subnet into a *virtual cluster.*

All devices in the virtual cluster carry session loads. Load balancing directs session traffic to the least-loaded device in the cluster, which distributes the load among all devices. It makes efficient use of system resources and provides increased performance and high availability.

One device in the virtual cluster, the *virtual cluster master*, directs incoming traffic to the other devices, called *backup devices*. The virtual cluster master monitors all devices in the cluster, keeps track of how busy each is, and distributes the session load accordingly. The role of virtual cluster master is not tied to a physical device; it can shift among devices. For example, if the current virtual cluster master fails, one of the backup devices in the cluster takes over that role and immediately becomes the new virtual cluster master.

The virtual cluster appears to outside clients as a single *virtual cluster IP address*. This IP address is not tied to a specific physical device. This address belongs to the current virtual cluster master, which makes it virtual. A VPN client attempting to establish a connection connects first to this virtual cluster IP address. The virtual cluster master then sends back to the client the public IP address of the least-loaded available host in the cluster. In a second transaction (transparent to the user), the client connects directly to that host. In this way, the virtual cluster master directs traffic evenly and efficiently across resources.

Note All clients other than the Cisco VPN client or the Cisco 3002 hardware client should connect directly to the ASA as usual; they do not use the virtual cluster IP address.

If a machine in the cluster fails, the terminated sessions can immediately reconnect to the virtual cluster IP address. The virtual cluster master then directs these connections to another active device in the cluster. If the virtual cluster master itself fails, a backup device in the cluster immediately and automatically takes over as the new virtual session master. Even if several devices in the cluster fail, users can continue to connect to the cluster as long as any one device in the cluster is up and available.

Comparing Load Balancing to Failover

Both load balancing and failover are high-availability features, but they function differently and have different requirements. In some circumstances you can use both load balancing and failover. The following sections describe the differences between these features.

Load Balancing

Load balancing is a mechanism for equitably distributing remote-access VPN traffic among the devices in a virtual cluster. It is based on simple distribution of traffic without taking into account throughput or other factors. A load-balancing cluster consists of two or more devices, one is the virtual master, and the other devices are the backup. These devices do not need to be of the exact same type, or have identical software versions or configurations.

All active devices in a virtual cluster carry session loads. Load balancing directs traffic to the least-loaded device in the cluster, distributing the load among all devices. It makes efficient use of system resources and provides increased performance and high availability.

Failover

A failover configuration requires two identical ASAs connected to each other through a dedicated failover link and, optionally, a stateful failover link. The health of the active interfaces and units is monitored to determine when specific failover conditions are met. If those conditions occur, failover occurs. Failover supports both VPN and firewall configurations.

The ASA supports two failover configurations: Active/Active failover and Active/Standby failover.

With Active/Active failover, both units can pass network traffic. This is not true load balancing, although it might appear to have the same effect. When failover occurs, the remaining active unit takes over passing the combined traffic, based on the configured parameters. Therefore, when configuring Active/Active failover, you must make sure that the combined traffic for both units is within the capacity of each unit.

With Active/Standby failover, only one unit passes traffic, while the other unit waits in a standby state and does not pass traffic. Active/Standby failover lets you use a second ASA to take over the functions of a failed unit. When the active unit fails, it changes to the standby state, while the standby unit changes to the active state. The unit that becomes active assumes the IP addresses (or, for transparent firewall, the management IP address) and MAC addresses of the failed unit and begins passing traffic. The unit that is now in standby state takes over the standby IP addresses of the active unit. If an active unit fails, the standby takes over without any interruption to the client VPN tunnel.

Implementing Load Balancing

Enabling load balancing involves:

- Configuring the load-balancing cluster by establishing a common virtual cluster IP address, UDP port (if necessary), and IPsec shared secret for the cluster. You configure these values identically for every device in the cluster.
- Configuring the participating device by enabling load balancing on the device and defining device-specific properties. These values vary from device to device.

Note VPN load balancing requires an active 3DES/AES license. The ASA checks for the existence of this crypto license before enabling load balancing. If it does not detect an active 3DES or AES license, the ASA prevents the enabling of load balancing and also prevents internal configuration of 3DES by the load balancing system unless the license permits this usage.

Prerequisites

Load balancing is disabled by default. You must explicitly enable load balancing.

You must have first configured the public (outside) and private (inside) interfaces and also have previously configured the interface to which the virtual cluster IP address refers. You can use the **interface** and **nameif** commands to configure different names for these interfaces. Subsequent references in this section use the names outside and inside.

All devices that participate in a cluster must share the same cluster-specific values: IP address, encryption settings, encryption key, and port.

Eligible Platforms

A load-balancing cluster can include ASA models ASA 5510 (with a Plus license) and Model 5520 and above. You can also include Cisco VPN 3000 series concentrators in the cluster. While mixed configurations are possible, administration is generally simpler if the cluster is homogeneous.

Eligible Clients

Load balancing is effective only on remote sessions initiated with the following clients:

- Cisco AnyConnect VPN client (Release 2.0 and later)
- Cisco VPN Client (Release 3.0 and later)
- Cisco ASA 5505 ASA (when acting as an Easy VPN client)
- Cisco VPN 3002 hardware client (Release 3.5 or later)
- Cisco PIX 501/506E when acting as an Easy VPN client
- Cisco IOS EZVPN client devices supporting IKE-redirect (IOS 831/871)
- Clientless SSL VPN (not a client)

Load balancing works with IPsec clients and SSL VPN client and clientless sessions. All other VPN connection types (L2TP, PPTP, L2TP/IPsec), including LAN-to-LAN, can connect to an ASA on which load balancing is enabled, but they cannot participate in load balancing.

VPN Load-Balancing Algorithm

The master device maintains a sorted list of backup cluster members in ascending IP address order. The load of each backup cluster member is computed as an integer percentage (the number of active sessions). AnyConnect inactive sessions do not count towards the SSL VPN load for load balancing. The master device redirects the IPsec and SSL VPN tunnel to the device with the lowest load until it is 1 percent higher than the rest. When all backup cluster members are 1% higher than the master, the master device redirects to itself.

For example, if you have one master and two backup cluster members, the following cycle applies:

Note All nodes start with 0%, and all percentages are rounded half-up.

1. The master device take s the connection if all members have a load at 1% higher than the master.
2. If the master does not take the connection, the session is taken by whichever backup device has the least load percentage.
3. If all members have the same percentage load, the backup device with the least number of sessions gets the session.
4. If all members have the same percentage load and the same number of sessions, the device with the least IP addresses gets the session.

VPN Load-Balancing Cluster Configurations

A load-balancing cluster can consist of ASAs of the same release, of mixed releases, as well as VPN 3000 concentrators, or a mixture of these, subject to the following restrictions:

- Load-balancing clusters that consist of same release ASAs or all VPN 3000 concentrators can run load balancing for a mixture of IPsec, AnyConnect, and clientless SSL VPN sessions.
- Load-balancing clusters that consist of both same release ASAs and VPN 3000 concentrators can run load balancing for a mixture of IPsec, AnyConnect, and clientless SSL VPN client and clientless sessions.
- Load-balancing clusters that include mixed release ASAs or same release ASAs and VPN 3000 concentrators or both can support only IPsec sessions. In such a configuration, however, the ASAs might not reach their full IPsec capacity. Scenario 1: Mixed Cluster with No SSL VPN Connections, illustrates this situation.

Since Release 7.1(1), IPsec and SSL VPN sessions count or weigh equally in determining the load that each device in the cluster carries. This is a change from the load-balancing calculation for the ASA Release 7.0(x) software and the VPN 3000 concentrator. Both platforms use a weighting algorithm that on some hardware platforms calculates the SSL VPN session load differently from the IPsec session load.

The virtual master of the cluster assigns session requests to the members of the cluster. The ASA regards all sessions, SSL VPN or IPsec, as equal and assigns them accordingly. You can configure the number of IPsec and SSL VPN sessions to allow up to the maximum allowed by your configuration and license. See Configuring VPN Session Limits for a description of how to set these limits.

We have tested up to ten nodes in a load-balancing cluster. Larger clusters might work, but we do not officially support such topologies.

Some Typical Mixed Cluster Scenarios

If you have a mixed configuration—that is, if your load-balancing cluster includes devices running a mixture of ASA software releases or at least one ASA running ASA Release 7.1(1) or later and a VPN 3000 concentrator—the difference in weighting algorithms becomes an issue if the initial cluster master fails and another device takes over as master.

The following scenarios illustrate the use of VPN load balancing in clusters consisting of a mixture of ASAs running ASA Release 7.1(1) and ASA Release 7.0(x) software, as well as VPN 3000 series concentrators.

Scenario 1: Mixed Cluster with No SSL VPN Connections

In this scenario, the cluster consists of a mixture of ASAs and VPN 3000 concentrators. Some of the ASA cluster peers are running ASA Release 7.0(x), and some are running Release 7.1(1). The pre-7.1(1) and VPN 3000 peers do not have any SSL VPN connections, and the 7.1(1) cluster peers have only the base SSL VPN license, which allows two SSL VPN sessions, but there are no SSL VPN connections. In this case, all the connections are IPsec, and load balancing works fine.

The two SSL VPN licenses have a very small effect on the user's taking advantage of the maximum IPsec session limit, and then only when a VPN 3000 concentrator is the cluster master. In general, the smaller the number of SSL VPN licenses is on a ASA in a mixed cluster, the smaller the effect on the ASA 7.1(1) device being able to reach its IPsec session limit in a scenario where there are only IPsec sessions.

Scenario 2: Mixed Cluster Handling SSL VPN Connections

Suppose, for example, an ASA running ASA Release 7.1(1) software is the initial cluster master and then that device fails. Another device in the cluster takes over automatically as master and applies its own load-balancing algorithm to determine processor loads within the cluster. A cluster master running ASA Release 7.1(1) software cannot weight session loads in any way other than what that software provides. Therefore, it cannot assign a combination of IPsec and SSL VPN session loads properly to ASA devices running earlier versions nor to VPN 3000 concentrators. Conversely, a VPN 3000 concentrator acting as the cluster master cannot assign loads properly to an ASA Release 7.1(1) ASA. The following scenario illustrates this dilemma.

This scenario is similar to the previous one, in that the cluster consists of a mixture of ASAs and VPN 3000 concentrators. Some of the ASA cluster peers are running ASA Release 7.0,(x) and some are running Release 7.1(1). In this case, however, the cluster is handling SSL VPN connections as well as IPsec connections.

If a device that is running software earlier than ASA Release 7.1(1) is the cluster master, the master applies the protocol and logic in effect prior to Release 7.1(1). That is, sessions might be directed to load-balancing peers that have exceeded their session limit. In that case, the user is denied access.

If the cluster master is a device running ASA Release 7.0(x) software, the old session-weighting algorithm applies only to the pre-7.1(1) peers in the cluster. No one should be denied access in this case. Because the pre-7.1(1) peers use the session-weighting algorithm, they are more lightly loaded.

An issue arises, however, because you cannot guarantee that the 7.1(1) peer is always the cluster master. If the cluster master fails, another peer assumes the role of master. The new master might be any of the eligible peers. Because of the unpredictability of the results, we recommend that you avoid configuring this type of cluster.

Configuring Load Balancing

To use load balancing, configure the following elements for each device that participates in the cluster:

- Public and private interfaces
- VPN load-balancing cluster attributes

Note All participants in the cluster must have an identical cluster configuration, except for the device priority within the cluster.

Note The Local CA feature is not supported if you use Active/Active stateful failover or VPN load-balancing. The Local CA cannot be subordinate to another CA; it can act only as the Root CA.

Configuring the Public and Private Interfaces for Load Balancing

To configure the public (outside) and private (inside) interfaces for the load-balancing cluster devices, do the following steps:

Step 1 Configure the public interface on the ASA by entering the **interface** command with the **lbpublic** keyword in vpn-load-balancing configuration mode. This command specifies the name or IP address of the public interface for load balancing for this device:

```
ciscoasa(config)# vpn load-balancing
ciscoasa(config-load-balancing)# interface lbpublic outside
ciscoasa(config-load-balancing)#
```

Step 2 Configure the private interface on the ASA by entering the **interface** command with the **lbprivate** keyword in vpn-load-balancing configuration mode. This command specifies the name or IP address of the private interface for load balancing for this device:

```
ciscoasa(config-load-balancing)# interface lbprivate inside
ciscoasa(config-load-balancing)#
```

Step 3 Set the priority to assign to this device within the cluster. The range is from 1 to 10. The priority indicates the likelihood of this device becoming the virtual cluster master, either at startup or when an existing master fails. The higher you set the priority (for example, 10), the more likely it is that this device becomes the virtual cluster master.

```
ciscoasa(config-load-balancing)# priority number
ciscoasa(config-load-balancing)#
```

For example, to assign this device a priority of 6 within the cluster, enter the following command:

```
ciscoasa(config-load-balancing)# priority 6
ciscoasa(config-load-balancing)#
```

Step 4 If you want to apply network address translation for this device, enter the **nat** command with the NAT assigned address for the device. You can define an IPv4 and an IPv6 address or specify the device's hostname.

```
ciscoasa(config-load-balancing)# nat ipv4_address ipv_address
ciscoasa(config-load-balancing)#
```

For example, to assign this device a NAT address of 192.168.30.3 and 2001:DB8::1, enter the following command:

```
ciscoasa(config-load-balancing)# nat 192.168.30.3 2001:DB8::1
ciscoasa(config-load-balancing)#
```

Configuring the Load Balancing Cluster Attributes

To configure the load-balancing cluster attributes for each device in the cluster, do the following steps:

Step 1 Set up VPN load balancing by entering the **vpn load-balancing** command in global configuration mode:

```
ciscoasa(config)# vpn load-balancing
ciscoasa(config-load-balancing)#
```

This enters vpn-load-balancing configuration mode, in which you can configure the remaining load-balancing attributes.

Step 2 Configure the IP address or the fully qualified domain name of the cluster to which this device belongs. This command specifies the single IP address or FQDN that represents the entire virtual cluster. Choose an IP address that is within the public subnet address range shared by all the ASAs in the virtual cluster. You can specify an IPv4 or IPv6 address.

```
ciscoasa(config-load-balancing)# cluster ip address ip_address
ciscoasa(config-load-balancing)#
```

For example, to set the cluster IP address to IPv6 address, 2001:DB8::1, enter the following command:

```
ciscoasa(config-load-balancing)# cluster ip address 2001:DB8::1
ciscoasa(config-load-balancing)#
```

Step 3 Configure the cluster port. This command specifies the UDP port for the virtual cluster in which this device is participating. The default value is 9023. If another application is using this port, enter the UDP destination port number that you want to use for load balancing.

```
ciscoasa(config-load-balancing)# cluster port port_number
ciscoasa(config-load-balancing)#
```

For example, to set the cluster port to 4444, enter the following command:

```
ciscoasa(config-load-balancing)# cluster port 4444
ciscoasa(config-load-balancing)#
```

Step 4 (Optional) Enable IPsec encryption for the cluster. The default is no encryption. This command enables or disables IPsec encryption. If you configure this check attribute, you must first specify and verify a shared secret.The ASAs in the virtual cluster communicate via LAN-to-LAN tunnels using IPsec. To ensure that all load-balancing information communicated between the devices is encrypted, enable this attribute.

```
ciscoasa(config-load-balancing)# cluster encryption
ciscoasa(config-load-balancing)#
```

Note When using encryption, you must have previously configured the load-balancing inside interface. If that interface is not enabled on the load-balancing inside interface, you get an error message when you try to configure cluster encryption.

If the load-balancing inside interface was enabled when you configured cluster encryption, but was disabled before you configured the participation of the device in the virtual cluster, you get an error message when you enter the **participate** command (or, in ASDM, check the **Participate in Load Balancing Cluster** check box), and encryption is not enabled for the cluster.

To use cluster encryption, you must enable ISAKMP on the inside interface, using the **crypto isakmp enable** command with the inside interface specified.

Step 5 If you enable cluster encryption, you must also specify the IPsec shared secret by entering the **cluster key** command. This command specifies the shared secret between IPsec peers when you have enabled IPsec encryption. The value you enter in the box appears as consecutive asterisk characters

```
ciscoasa(config-load-balancing)# cluster key shared_secret
ciscoasa(config-load-balancing)#
```

For example, to set the shared secret to 123456789, enter the following command:

```
ciscoasa(config-load-balancing)# cluster key 123456789
ciscoasa(config-load-balancing)#
```

Step 6 Enable this device's participation in the cluster by entering the participate command:

```
ciscoasa(config-load-balancing)# participate
ciscoasa(config-load-balancing)#
```

Enabling Redirection Using a Fully Qualified Domain Name

To enable or disable redirection using a fully qualified domain name in vpn load-balancing mode, use the **redirect-fqdn enable** command in global configuration mode. This behavior is disabled by default.

By default, the ASA sends only IP addresses in load-balancing redirection to a client. If certificates are in use that are based on DNS names, the certificates will be invalid when redirected to a backup device.

As a VPN cluster master, this ASA can send a fully qualified domain name (FQDN), using reverse DNS lookup, of a cluster device (another ASA in the cluster) instead of its outside IP address when redirecting VPN client connections to that cluster device.

All of the outside and inside network interfaces on the load-balancing devices in a cluster must be on the same IP network.

To do VPN load balancing for SSL or IPsec/IKEv2 connections using FQDNs rather than IP addresses, perform the following configuration steps:

Step 1 Enable the use of FQDNs for load balancing with the **redirect-fqdn enable** command:

redirect-fqdn {enable | disable}
no redirect-fqdn {enable | disable}

```
For example:
ciscoasa(config)# vpn load-balancing
ciscoasa(config-load-balancing)# redirect-fqdn enable
```

```
ciscoasa(config-load-balancing)#
```

Step 2 Add an entry for each of your ASA outside interfaces into your DNS server if such entries are not already present. Each ASA outside IP address should have a DNS entry associated with it for lookups. These DNS entries must also be enabled for reverse lookup.

Step 3 Enable DNS lookups on your ASA with the **dns domain-lookup inside** command or whichever interface has a route to your DNS server.

Step 4 Define your DNS server IP address on the ASA; for example: **dns name-server 10.2.3.4** (IP address of your DNS server).

The following is an example of a VPN load balancing command sequence that includes an interface command that enables redirection for a fully qualified domain name, specifies the public interface of the cluster as **test** and the private interface of the cluster as **foo**"

```
ciscoasa(config)# interface GigabitEthernet 0/1
ciscoasa(config-if)# ip address 209.165.202.159 255.255.255.0
ciscoasa(config)# nameif test
ciscoasa(config)# interface GigabitEthernet 0/2
ciscoasa(config-if)# ip address 209.165.201.30 255.255.255.0
ciscoasa(config)# nameif foo
ciscoasa(config)# vpn load-balancing
ciscoasa(config-load-balancing)# nat 192.168.10.10
ciscoasa(config-load-balancing)# priority 9
ciscoasa(config-load-balancing)# interface lbpublic test
ciscoasa(config-load-balancing)# interface lbprivate foo
ciscoasa(config-load-balancing)# cluster ip address 209.165.202.224
ciscoasa(config-load-balancing)# cluster key 123456789
ciscoasa(config-load-balancing)# cluster encryption
ciscoasa(config-load-balancing)# cluster port 9023
ciscoasa(config-load-balancing)# redirect-fqdn enable
ciscoasa(config-load-balancing)# participate
```

Frequently Asked Questions About Load Balancing

IP Address Pool Exhaustion

Q: Does the ASA consider IP address pool exhaustion as part of its VPN load-balancing method?

A: No. If the remote access VPN session is directed to a device that has exhausted its IP address pools, the session does not establish. The load-balancing algorithm is based on load, and is computed as an integer percentage (number of active and maximum sessions) that each backup cluster member supplies.

Unique IP Address Pools

Q: To implement VPN load balancing, must the IP address pools for AnyConnect clients or IPsec clients on different ASAs be unique?

A: Yes. IP address pools must be unique for each device.

Using Load Balancing and Failover on the Same Device

Q: Can a single device use both load balancing and failover?

A: Yes. In this configuration, the client connects to the IP address of the cluster and is redirected to the least-loaded ASA in the cluster. If that device fails, the standby unit takes over immediately, and there is no impact to the VPN tunnel.

Load Balancing on Multiple Interfaces

Q: If we enable SSL VPN on multiple interfaces, is it possible to implement load balancing for both of the interfaces?

A: You can define only one interface to participate in the cluster as the public interface. The idea is to balance the CPU loads. Multiple interfaces converge on the same CPU, so the concept of load balancing on multiple interfaces has no meaning.

Maximum Simultaneous Sessions for Load Balancing Clusters

Q: Consider a deployment of two ASA 5520s, each with a 100-user SSL VPN license. In a load-balancing cluster, does the maximum total number of users allow 200 simultaneous sessions, or only 100? If we add a third device later with a 100-user license, can we now support 300 simultaneous sessions?

A: With VPN load balancing, all devices are active, so the maximum number of sessions that your cluster can support is the total of the number of sessions for each of the devices in the cluster, in this case 300.

Viewing Load Balancing

The load-balancing cluster master receives a periodic message from each ASA in the cluster with the number of active AnyConnect and clientless sessions, as well as the maximum allowed sessions based on the configured or license limits. If an ASA in the cluster shows 100 percent full capacity, the cluster master cannot redirect more connections to it. Although the ASA may show as full, some users may be in inactive/wait-to-resume state, wasting the licenses. As a workaround, each ASA provides the total number of sessions minus the sessions in inactive state, instead of the total number of sessions. (Refer to the **-sessiondb summary** command in the command reference. In other words, the inactive sessions are not reported to the cluster master. Even if the ASA is full (with some inactive sessions), the cluster master still redirects connections to it if necessary. When the ASA receives the new connection, the session that has been inactive the longest is logged off, allowing new connections to take its license.

The following example shows 100 SSL sessions (active only) and a 2 percent SSL load. These numbers do not include the inactive sessions. In other words, inactive sessions do not count towards the load for load balancing.

```
hostname#  load-balancing
    Status :       enabled
    Role :         Master
    Failover :     Active
    Encryption :   enabled
    Cluster IP :   192.168.1.100
    Peers :        1

                                                Load %
Sessions
    Public IP      Role    Pri     Model          IPsec   SSL     IPsec SSL
    192.168.1.9    Master  7       ASA-5540       4       2       216     100
    192.168.1.19   Backup  9       ASA-5520       0       0       0       0
```

Configuring VPN Session Limits

You can run as many IPsec and SSL VPN sessions as your platform and ASA license supports. To view the licensing information including maximum sessions for your ASA, enter the **show version** command in global configuration mode. The following example shows the command and the licensing information from the output of this command:

```
hostname(config)# show version

Cisco Adaptive Security Appliance Software Version 8.4(1)
Device Manager Version 6.4(1)

Compiled on Sun 02-Jan-11 03:45 by builders
System image file is "disk0:/cdisk.bin"
Config file at boot was "startup-config"
asa4 up 9 days 3 hours

Hardware:   ASA5510, 256 MB RAM, CPU Pentium 4 Celeron 1600 MHz
Internal ATA Compact Flash, 256MB
BIOS Flash M50FW080 @ 0xfff00000, 1024KB

Encryption hardware device : Cisco ASA-55x0 on-board accelerator (revision 0x0)
                             Boot microcode        : CN1000-MC-BOOT-2.00
                             SSL/IKE microcode     : CNLite-MC-SSLm-PLUS-2.03
                             IPsec microcode       : CNlite-MC-IPSECm-MAIN-2.06
                             Number of accelerators: 1

 0: Ext: Ethernet0/0         : address is 001e.f75e.8b84, irq 9
 1: Ext: Ethernet0/1         : address is 001e.f75e.8b85, irq 9
 2: Ext: Ethernet0/2         : address is 001e.f75e.8b86, irq 9
 3: Ext: Ethernet0/3         : address is 001e.f75e.8b87, irq 9
 4: Ext: Management0/0       : address is 001e.f75e.8b83, irq 11
 5: Int: Internal-Data0/0    : address is 0000.0001.0002, irq 11
 6: Int: Internal-Control0/0 : address is 0000.0001.0001, irq 5

Licensed features for this platform:
Maximum Physical Interfaces       : Unlimited      perpetual
Maximum VLANs                     : 100            perpetual
Inside Hosts                      : Unlimited      perpetual
Failover                          : Active/Active  perpetual
VPN-DES                           : Enabled        perpetual
VPN-3DES-AES                      : Enabled        perpetual
Security Contexts                 : 2              perpetual
GTP/GPRS                          : Disabled       perpetual
AnyConnect Premium Peers          : 250            perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 250            perpetual
Total VPN Peers                   : 250            perpetual
Shared License                    : Disabled       perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Enabled        perpetual
UC Phone Proxy Sessions           : 2              perpetual
Total UC Proxy Sessions           : 2              perpetual
Botnet Traffic Filter             : Disabled       perpetual
Intercompany Media Engine         : Disabled       perpetual

This platform has an ASA 5510 Security Plus license.

hostname#
```

To limit AnyConnect VPN sessions (either IPsec/IKEv2 or SSL) to a lower value than the ASA allows, use the **vpn-sessiondb max-anyconnect-premium-or-essentials-limit** command in global configuration mode. To remove the session limit, use the **no** version of this command.

For example, if the ASA license allows 500 SSL VPN sessions, and you want to limit the number of AnyConnect VPN sessions to 250, enter the following command:

```
ciscoasa(config)# vpn-sessiondb max-anyconnect-premium-or-essentials-limit 250
ciscoasa(config)#
```

To remove the session limit, use the **no** version of this command.:

```
ciscoasa(config)# no vpn-sessiondb max-anyconnect-premium-or-essentials-limit 250
ciscoasa(config)#
```

To limit Cisco VPN client (IPsec IKEv1), Lan-to-Lan VPN, and clientless SSL VPN sessions to a lower value than the ASA allows, enter the **vpn-sessiondb max-other-vpn-limit** command in global configuration mode:

For example, if the ASA license allows 750 IPsec sessions, and you want to limit the number of IPsec sessions to 500, enter the following command:

```
ciscoasa(config)# vpn-sessiondb max-other-vpn-limit 500
ciscoasa(config)#
```

To remove the session limit, use the **no** version of this command:

```
ciscoasa(config)# no vpn-sessiondb max-other-vpn-limit 500
ciscoasa(config)#
```

For a complete description of the features available with each license, see the document Managing Feature Licenses for Cisco ASA 5500 at this URL:

http://www.cisco.com/en/US/docs/security/asa/asa91/license/license_management/license.html

Using an Identify Certificate When Negotiating

The ASA needs to use an identity certificate when negotiating the IKEv2 tunnel with AnyConnect clients. For ikev2 remote access trustpoint configuration, use the following commands

`crypto ikev2 remote-access trustpoint <name> [line<number>]`

Using this command allows the AnyConnect client to support group selection for the end user. You can configure two trustpoints at the same time: two RSA, two ECDSA, or one of each. The ASA scans the configured trustpoint list and chooses the first one that the client supports. If ECDSA is preferred, you should configure that trustpoint before the RSA trustpoint.

The line number option specifies where in the line number you want the trustpoint inserted. Typically, this option is used to insert a trustpoint at the top without removing and re-adding the other line. If a line is not specified, the ASA adds the trustpoint at the end of the list.

If you try to add a trustpoint that already exists, you receive an error. If you use the *no crypto ikev2 remote-access trustpoint* command without specifying which trustpoint name to remove, all trustpoint configuration is removed.

Configuring the Pool of Cryptographic Cores

You can change the allocation of cryptographic cores on Symmetric Multi-Processing (SMP) platforms to give you better throughput performance for AnyConnect TLS/DTLS traffic. These changes can accelerate the SSL VPN datapath and provide customer-visible performance gains in AnyConnect, smart tunnels, and port forwarding. These steps describe configuring the pool of cryptographic cores in either single or multiple context mode:

Note Multiple context mode only applies to IKEv2 and IKEv1 site to site but does not apply to AnyConnect, clientless SSL VPN, the legacy Cisco VPN client, the Apple native VPN client, the Microsoft native VPN client, or the cTCP for IKEv1 IPsec.

Limitations

- Cryptographic core rebalancing is available on the following platforms:
  - 5585-X
  - 5580
  - 5545-X
  - 5555-X
  - ASASM
- The large modulus operation is only available for 5510, 5520, 5540, and 5550 platforms.

Detailed Steps

	Command	Purpose
Step 1	`asa1(config)# crypto engine ?` `asa1(config)# crypto engine accelerator-bias ?`	Specifies how to allocate crypto accelerator processors: • balanced - Equally distribute crypto hardware resources • ipsec - Allocate crypto hardware resources to favor IPsec/Encrypted Voice (SRTP) • ssl - Allocate crypto hardware resources to favor SSL
Step 2	`large-mode-accel`	Performs large modulus operation in the hardware.

Viewing Active VPN Sessions

Viewing Active AnyConnect Sessions by IP Address Type

To view active AnyConnect sessions using the command line interface, enter the **show vpn-sessiondb anyconnect filter p-ipversion** or **show vpn-sessiondb anyconnect filter a-ipversion** command in privileged EXEC mode.

Command	Purpose
show vpn-sessiondb anyconnect filter p-ipversion {v4 | v6}	This command shows active AnyConnect sessions filtered by the endpoint's public IPv4 or IPv6 address. The public address is the address assigned to the endpoint by the enterprise.
show vpn-sessiondb anyconnect filter a-ipversion {v4 | v6}	This command shows active AnyConnect sessions filtered by the endpoint's assigned IPv4 or IPv6 address. The assigned address is the address assigned to the AnyConnect Secure Mobility Client by the ASA.

Examples

Example 3-1 Output from show vpn-sessiondb anyconnect filter p-ipversion [v4 | v6] command

```
hostname(config)# show vpn-sessiondb anyconnect filter p-ipversion v4

Session Type: AnyConnect

Username     : user1                  Index        : 40
Assigned IP  : 192.168.17.10          Public IP    : 198.51.100.1
Protocol     : AnyConnect-Parent SSL-Tunnel
License      : AnyConnect Premium
Encryption   : AnyConnect-Parent: (1)none  SSL-Tunnel: (1)RC4
Hashing      : AnyConnect-Parent: (1)none  SSL-Tunnel: (1)SHA1
Bytes Tx     : 10570                  Bytes Rx     : 8085
Group Policy : GroupPolicy_SSLACCLIENT
Tunnel Group : SSLACCLIENT
Login Time   : 15:17:12 UTC Mon Oct 22 2012
Duration     : 0h:00m:09s
Inactivity   : 0h:00m:00s
NAC Result   : Unknown
VLAN Mapping : N/A                    VLAN         : none
```

Example 3-2 Output from show vpn-sessiondb anyconnect filter a-ipversion [v4 | v6] command

```
hostname(config)# show vpn-sessiondb anyconnect filter a-ipversion v6

Session Type: AnyConnect

Username     : user1                  Index        : 45
Assigned IP  : 192.168.17.10
Public IP    : 2001:DB8:8:1:90eb:3fe5:9eea:fb29
Assigned IPv6: 2001:DB8:9:1::24
Protocol     : AnyConnect-Parent SSL-Tunnel
License      : AnyConnect Premium
Encryption   : AnyConnect-Parent: (1)none  SSL-Tunnel: (1)RC4
Hashing      : AnyConnect-Parent: (1)none  SSL-Tunnel: (1)SHA1
Bytes Tx     : 10662                  Bytes Rx     : 17248
Group Policy : GroupPolicy_SSL_IPv6   Tunnel Group : SSL_IPv6
Login Time   : 17:42:42 UTC Mon Oct 22 2012
Duration     : 0h:00m:33s
Inactivity   : 0h:00m:00s
NAC Result   : Unknown
VLAN Mapping : N/A                    VLAN         : none
```

Viewing Active Clientless SSL VPN Sessions by IP Address Type

To view active clientless SSL VPN sessions using the command line interface, enter the **show vpn-sessiondb webvpn filter ipversion** command in privileged EXEC mode.

Command	Purpose
show vpn-sessiondb webvpn filter ipversion {v4 | v6}	This command shows active clientless SSL VPN sessions filtered by the endpoint's public IPv4 or IPv6 address. The public address is the address assigned to the endpoint by the enterprise.

Examples

Example 3-3 Output from show vpn-sessiondb webvpn filter ipversion [v4 | v6] command

```
hostname# sh vpn-sessiondb webvpn filter ipversion v4

Session Type: WebVPN

Username     : user1                  Index        : 63
Public IP    : 171.16.17.6
Protocol     : Clientless
License      : AnyConnect Premium
Encryption   : Clientless: (1)RC4     Hashing      : Clientless: (1)SHA1
Bytes Tx     : 62454                  Bytes Rx     : 13082
Group Policy : SSLv6                  Tunnel Group : SSL_IPv6
Login Time   : 18:07:48 UTC Mon Oct 22 2012
Duration     : 0h:00m:16s
Inactivity   : 0h:00m:00s
NAC Result   : Unknown
VLAN Mapping : N/A                    VLAN         : none
```

Viewing Active Lan to Lan VPN Sessions by IP Address Type

To view active clientless SSL VPN sessions using the command line interface, enter the **show vpn-sessiondb l2l filter ipversion** command in privileged EXEC mode.

Command	Purpose
show vpn-sessiondb l2l filter ipversion {v4 | v6}	This command shows active lan to lan VPN sessions filtered by the connection's public IPv4 or IPv6 address. The public address is the address assigned to the endpoint by the enterprise.

Configuring Connection Profiles, Group Policies, and Users

This chapter describes how to configure VPN connection profiles (formerly called "tunnel groups"), group policies, and users. This chapter includes the following sections.

In summary, you first configure connection profiles to set the values for the connection. Then you configure group policies. These set values for users in the aggregate. Then you configure users, which can inherit values from groups and configure certain values on an individual user basis. This chapter describes how and why to configure these entities.

Overview of Connection Profiles, Group Policies, and Users

Groups and users are core concepts in managing the security of virtual private networks (VPNs) and in configuring the ASA. They specify attributes that determine user access to and use of the VPN. A *group* is a collection of users treated as a single entity. *Users* get their attributes from *group policies*. *A connection profile* identifies the group policy for a specific connection. If you do not assign a particular group policy to a user, the default group policy for the connection applies.

Note You configure connection profiles using **tunnel-group** commands. In this chapter, the terms "connection profile" and "tunnel group" are often used interchangeably.

Connection profiles and group policies simplify system management. To streamline the configuration task, the ASA provides a default LAN-to-LAN connection profile, a default remote access connection profile, a default connection profile for SSL/IKEv2 VPN, and a default group policy (DfltGrpPolicy). The default connection profiles and group policy provide settings that are likely to be common for many users. As you add users, you can specify that they "inherit" parameters from a group policy. Thus you can quickly configure VPN access for large numbers of users.

If you decide to grant identical rights to all VPN users, then you do not need to configure specific connection profiles or group policies, but VPNs seldom work that way. For example, you might allow a finance group to access one part of a private network, a customer support group to access another part,

and an MIS group to access other parts. In addition, you might allow specific users within MIS to access systems that other MIS users cannot access. Connection profiles and group policies provide the flexibility to do so securely.

Note The ASA also includes the concept of object groups, which are a superset of network lists. Object groups let you define VPN access to ports as well as networks. Object groups relate to ACLs rather than to group policies and connection profiles. For more information about using object groups, see Chapter 20, "Configuring Objects," in the general operations configuration guide.

The security appliance can apply attribute values from a variety of sources. It applies them according to the following hierarchy:

1. Dynamic Access Policy (DAP) record
2. Username
3. Group policy
4. Group policy for the connection profile
5. Default group policy

Therefore, DAP values for an attribute have a higher priority than those configured for a user, group policy, or connection profile.

When you enable or disable an attribute for a DAP record, the ASA applies that value and enforces it. For example, when you disable HTTP proxy in dap webvpn mode, the ASA looks no further for a value. When you instead use the **no** value for the **http-proxy** command, the attribute is not present in the DAP record, so the security appliance moves down to the AAA attribute in the username, and if necessary, the group policy to find a value to apply. The ASA clientless SSL VPN configuration supports only one http-proxy and one https-proxy command each. We recommend that you use ASDM to configure DAP.

Connection Profiles

A connection profile consists of a set of records that determines tunnel connection policies. These records identify the servers to which the tunnel user is authenticated, as well as the accounting servers, if any, to which connection information is sent. They also identify a default group policy for the connection, and they contain protocol-specific connection parameters. Connection profiles include a small number of attributes that pertain to creating the tunnel itself. Connection profiles include a pointer to a group policy that defines user-oriented attributes.

The ASA provides the following default connection profiles: DefaultL2Lgroup for LAN-to-LAN connections, DefaultRAgroup for remote access connections, and DefaultWEBVPNGroup for SSL VPN (browser-based) connections. You can modify these default connection profiles, but you cannot delete them. You can also create one or more connection profiles specific to your environment. Connection profiles are local to the ASA and are not configurable on external servers.

Connection profiles specify the following attributes:

- General Connection Profile Connection Parameters, page 4-3
- IPsec Tunnel-Group Connection Parameters, page 4-4
- Connection Profile Connection Parameters for SSL VPN Sessions, page 4-5

General Connection Profile Connection Parameters

General parameters are common to all VPN connections. The general parameters include the following:

- Connection profile name—You specify a connection-profile name when you add or edit a connection profile. The following considerations apply:
  - For clients that use preshared keys to authenticate, the connection profile name is the same as the group name that a client passes to the ASA.
  - Clients that use certificates to authenticate pass this name as part of the certificate, and the ASA extracts the name from the certificate.
- Connection type—Connection types include IKEv1 remote-access, IPsec Lan-to-LAN, and Anyconnect (SSL/IKEv2). A connection profile can have only one connection type.
- Authentication, Authorization, and Accounting servers—These parameters identify the server groups or lists that the ASA uses for the following purposes:
  - Authenticating users
  - Obtaining information about services users are authorized to access
  - Storing accounting records

  A server group can consist of one or more servers.
- Default group policy for the connection—A group policy is a set of user-oriented attributes. The default group policy is the group policy whose attributes the ASA uses as defaults when authenticating or authorizing a tunnel user.
- Client address assignment method—This method includes values for one or more DHCP servers or address pools that the ASA assigns to clients.
- Override account disabled—This parameter lets you override the “account-disabled” indicator received from a AAA server.
- Password management—This parameter lets you warn a user that the current password is due to expire in a specified number of days (the default is 14 days), then offer the user the opportunity to change the password.
- Strip group and strip realm—These parameters direct the way the ASA processes the usernames it receives. They apply only to usernames received in the form user@realm.

  A realm is an administrative domain appended to a username with the @ delimiter (user@abc). If you strip the realm, the ASA uses the username and the group (if present) for authentication. If you strip the group, the ASA uses the username and the realm (if present) for authentication.

  Enter the strip-realm command to remove the realm qualifier, and enter the strip-group command to remove the group qualilfier from the username during authentication. If you remove both qualifiers, authentication is based on the *username* alone. Otherwise, authentication is based on the full *username@realm* or *username<delimiter> group* string. You must specify strip-realm if your server is unable to parse delimiters.

  In addition, for L2TP/IPsec clients only, when you specify the strip-group command the ASA selects the connection profile (tunnel group) for user connections by obtaining the group name from the username presented by the VPN client.
- Authorization required—This parameter lets you require authorization before a user can connect, or turn off that requirement.
- Authorization DN attributes—This parameter specifies which Distinguished Name attributes to use when performing authorization.

IPsec Tunnel-Group Connection Parameters

IPsec parameters include the following:

- A client authentication method: preshared keys, certificates, or both.
  - For IKE connections based on preshared keys, this is the alphanumeric key itself (up to 128 characters long), associated with the connection policy.
  - Peer-ID validation requirement—This parameter specifies whether to require validating the identity of the peer using the peer's certificate.
  - If you specify certificates or both for the authentication method, the end user must provide a valid certificate in order to authenticate.
- An extended hybrid authentication method: XAUTH and hybrid XAUTH.

  You use **isakmp ikev1-user-authentication** command to implement hybrid XAUTH authentication when you need to use digital certificates for ASA authentication and a different, legacy method for remote VPN user authentication, such as RADIUS, TACACS+ or SecurID.
- ISAKMP (IKE) keepalive settings. This feature lets the ASA monitor the continued presence of a remote peer and report its own presence to that peer. If the peer becomes unresponsive, the ASA removes the connection. Enabling IKE keepalives prevents hung connections when the IKE peer loses connectivity.

  There are various forms of IKE keepalives. For this feature to work, both the ASA and its remote peer must support a common form. This feature works with the following peers:
  - Cisco AnyConnect VPN Client
  - Cisco VPN Client (Release 3.0 and above)
  - Cisco VPN 3000 Client (Release 2.x)
  - Cisco VPN 3002 Hardware Client
  - Cisco VPN 3000 Series Concentrators
  - Cisco IOS software
  - Cisco Secure PIX Firewall

  Non-Cisco VPN clients do not support IKE keepalives.

  If you are configuring a group of mixed peers, and some of those peers support IKE keepalives and others do not, enable IKE keepalives for the entire group. The feature does not affect the peers that do not support it.

  If you disable IKE keepalives, connections with unresponsive peers remain active until they time out, so we recommend that you keep your idle timeout short. To change your idle timeout, see "Configuring Group Policies" section on page 4-39.

Note To reduce connectivity costs, disable IKE keepalives if this group includes any clients connecting via ISDN lines. ISDN connections normally disconnect if idle, but the IKE keepalive mechanism prevents connections from idling and therefore from disconnecting.

If you do disable IKE keepalives, the client disconnects only when either its IKE or IPsec keys expire. Failed traffic does not disconnect the tunnel with the Peer Timeout Profile values as it does when IKE keepalives are enabled.

Note If you have a LAN-to-LAN configuration using IKE main mode, make sure that the two peers have the same IKE keepalive configuration. Both peers must have IKE keepalives enabled or both peers must have it disabled.

- If you configure authentication using digital certificates, you can specify whether to send the entire certificate chain (which sends the peer the identity certificate and all issuing certificates) or just the issuing certificates (including the root certificate and any subordinate CA certificates).
- You can notify users who are using outdated versions of Windows client software that they need to update their client, and you can provide a mechanism for them to get the updated client version. For VPN 3002 hardware client users, you can trigger an automatic update. You can configure and change the client-update, either for all connection profiles or for particular connection profiles.
- If you configure authentication using digital certificates, you can specify the name of the trustpoint that identifies the certificate to send to the IKE peer.

Connection Profile Connection Parameters for SSL VPN Sessions

Table 4-1 provides a list of connection profile attributes that are specific to SSL VPN (AnyConnect client and clientless) connections. In addition to these attributes, you configure general connection profile attributes common to all VPN connections. For step-by-step information about configuring connection profiles, see Configuring Connection Profiles for Clientless SSL VPN Sessions, page 4-20.

Note In earlier releases, "connection profiles" were known as "tunnel groups." You configure a connection profile with tunnel-group commands. This chapter often uses these terms interchangeably.

Table 4-1 Connection Profile Attributes for SSL VPN

Command	Function
authentication	Sets the authentication method, AAA or certificate.
customization	Identifies the name of a previously defined customization to apply. Customizations determine the appearance of the windows that the user sees upon login. You configure the customization parameters as part of configuring clientless SSL VPN.
nbns-server	Identifies the name of the NetBIOS Name Service server (nbns-server) to use for CIFS name resolution.
group-alias	Specifies one or more alternate names by which the server can refer to a connection profile. At login, the user selects the group name from a dropdown menu.
group-url	Identifies one or more group URLs. If you configure this attribute, users coming in on a specified URL need not select a group at login.
dns-group	Identifies the DNS server group that specifies the DNS server name, domain name, name server, number of retries, and timeout values for a DNS server to use for a connection profile.
hic-fail-group-policy	Specifies a VPN feature policy if you use the Cisco Secure Desktop Manager to set the Group-Based Policy attribute to "Use Failure Group-Policy" or "Use Success Group-Policy, if criteria match."

Table 4-1 Connection Profile Attributes for SSL VPN (continued)

Command	Function
override-svc-download	Overrides downloading the group-policy or username attributes configured for downloading the AnyConnect VPN client to the remote user.
radius-reject-message	Enables the display of the RADIUS reject message on the login screen when authentication is rejected.

Configuring Connection Profiles

This section describes the contents and configuration of connection profiles in both single context mode or multiple-context mode:

Note Multiple-context mode applies only to IKEv2 and IKEv1 site to site and does not apply to AnyConnect, Clientless SSL VPN, legacy Cisco VPN client, the Apple native VPN client, the Microsoft native VPN client, or cTCP for IKEv1 IPsec.

You can modify the default connection profiles, and you can configure a new connection profile as any of the three tunnel-group types. If you do not explicitly configure an attribute in a connection profile, that attribute gets its value from the default connection profile. The default connection-profile type is remote access. The subsequent parameters depend upon your choice of tunnel type. To see the current configured and default configuration of all your connection profiles, including the default connection profile, enter the **show running-config all tunnel-group** command.

Maximum Connection Profiles

The maximum number of connection profiles (tunnel groups) that an ASA can support is a function of the maximum number of concurrent VPN sessions for the platform + 5. For example, an ASA 5505 can support a maximum of 25 concurrent VPN sessions allowing for 30 tunnel groups (25+5). Attempting to add an additional tunnel group beyond the limit results in the following message: “ERROR: The limit of 30 configured tunnel groups has been reached.”

Table 4-2 specifies the maximum VPN sessions and connection profiles for each ASA platform.

Table 4-2 Maximum VPN Sessions and Connection Profiles Per ASA Platform

	5505 Base/ Security Plus	5510/Base/ Security Plus	5520	5540	5550	5580-20	5580-40
Maximum VPN Sessions	10/25	250	750	5000	5000	10,000	10,000
Maximum Connection Profiles	15/30	255	755	5005	5005	10,005	10,005

Default IPsec Remote Access Connection Profile Configuration

The contents of the default remote-access connection profile are as follows:

```
tunnel-group DefaultRAGroup type remote-access
tunnel-group DefaultRAGroup general-attributes
 no address-pool
 no ipv6-address-pool
 authentication-server-group LOCAL
 accounting-server-group RADIUS
 default-group-policy DfltGrpPolicy
 no dhcp-server
 no strip-realm
 no password-management
 no override-account-disable
 no strip-group
 no authorization-required
 authorization-dn-attributes CN OU
tunnel-group DefaultRAGroup webvpn-attributes
 hic-fail-group-policy DfltGrpPolicy
 customization DfltCustomization
 authentication aaa
 no override-svc-download
 no radius-reject-message
 dns-group DefaultDNS
tunnel-group DefaultRAGroup ipsec-attributes
 no pre-shared-key
 peer-id-validate req
 no chain
 no trust-point
 isakmp keepalive threshold 1500 retry 2
 no radius-sdi-xauth
 isakmp ikev1-user-authentication xauth
tunnel-group DefaultRAGroup ppp-attributes
 no authentication pap
 authentication chap
 authentication ms-chap-v1
 no authentication ms-chap-v2
 no authentication eap-proxy
```

Configuring IPsec Tunnel-Group General Attributes

The general attributes are common across more than one tunnel-group type. IPsec remote access and clientless SSL VPN tunnels share most of the same general attributes. IPsec LAN-to-LAN tunnels use a subset. Refer to the *Cisco ASA Series Command Reference* for complete descriptions of all commands. This section describes, in order, how to configure remote-access and LAN-to-LAN connection profiles.

Configuring Remote-Access Connection Profiles

Use a remote-access connection profile when setting up a connection between the following remote clients and a central-site ASA:

- Legacy Cisco VPN Client (connecting with IPsec/IKEv1)
- AnyConnect Secure Mobility Client (connecting with SSL or IPsec/IKEv2)
- Clientless SSL VPN (browser-based connecting with SSL)
- Cisco ASA 5500 Easy VPN hardware client (connecting with IPsec/IKEv1)
- Cisco VPM 3002 hardware client (connecting with IPsec/IKEv1)

We also provide a default group policy named *DfltGrpPolicy*.

To configure an remote-access connection profile, first configure the tunnel-group general attributes, then the remote-access attributes. See the following sections:

- Specifying a Name and Type for the Remote Access Connection Profile, page 4-8.
- Configuring Remote-Access Connection Profile General Attributes, page 4-8.
- Configuring Double Authentication, page 4-12
- Configuring Remote-Access Connection Profile IPsec IKEv1 Attributes, page 4-14.
- Configuring IPsec Remote-Access Connection Profile PPP Attributes, page 4-16

Specifying a Name and Type for the Remote Access Connection Profile

Create the connection profile, specifying its name and type, by entering the **tunnel-group** command. For an remote-access tunnel, the type is **remote-access:**

```
hostname(config)# tunnel-group tunnel_group_name type remote-access
hostname(config)#
```

For example, to create an remote-access connection profile named TunnelGroup1, enter the following command:

```
hostname(config)# tunnel-group TunnelGroup1 type remote-access
hostname(config)#
```

Configuring Remote-Access Connection Profile General Attributes

To configure or change the connection profile general attributes, specify the parameters in the following steps:

Step 1 To configure the general attributes, enter the **tunnel-group general-attributes** task in either single or multiple context mode, which enters tunnel-group general-attributes configuration mode. The prompt changes to indicate the change in mode.

```
hostname(config)# tunnel-group tunnel_group_name general-attributes
hostname(config-tunnel-general)#
```

Step 2 Specify the name of the authentication-server group, if any, to use. If you want to use the LOCAL database for authentication if the specified server group fails, append the keyword **LOCAL**:

```
hostname(config-tunnel-general)# authentication-server-group [(interface_name)] groupname
[LOCAL]
hostname(config-tunnel-general)#
```

The name of the authentication server group can be up to 16 characters long.

You can optionally configure interface-specific authentication by including the name of an interface after the group name. The interface name, which specifies where the tunnel terminates, must be enclosed in parentheses. The following command configures interface-specific authentication for the interface named test using the server named servergroup1 for authentication:

```
hostname(config-tunnel-general)# authentication-server-group (test) servergroup1
hostname(config-tunnel-general)#
```

Step 3 Specify the name of the authorization-server group, if any, to use. When you configure this value, users must exist in the authorization database to connect:

```
hostname(config-tunnel-general)# authorization-server-group groupname
hostname(config-tunnel-general)#
```

The name of the authorization server group can be up to 16 characters long. For example, the following command specifies the use of the authorization-server group FinGroup:

```
hostname(config-tunnel-general)# authorization-server-group FinGroup
hostname(config-tunnel-general)#
```

Step 4 Specify the name of the accounting-server group, if any, to use:

```
hostname(config-tunnel-general)# accounting-server-group groupname
hostname(config-tunnel-general)#
```

The name of the accounting server group can be up to 16 characters long. For example, the following command specifies the use of the accounting-server group named comptroller:

```
hostname(config-tunnel-general)# accounting-server-group comptroller
hostname(config-tunnel-general)#
```

Step 5 Specify the name of the default group policy:

```
hostname(config-tunnel-general)# default-group-policy policyname
hostname(config-tunnel-general)#
```

The name of the group policy can be up to 64 characters long. The following example sets DfltGrpPolicy as the name of the default group policy:

```
ciscoasa(config-tunnel-general)# default-group-policy DfltGrpPolicy
hostname(config-tunnel-general)#
```

Step 6 Specify the names or IP addresses of the DHCP server (up to 10 servers), and the names of the DHCP address pools (up to 6 pools). The defaults are no DHCP server and no address pool. The **dhcp-server** command will allow you to configure the ASA to send additional options to the specified DHCP servers when it is trying to get IP addresses for VPN clients. See the **dhcp-server** command in the *Cisco ASA Series Command Reference* guide for more information.

```
hostname(config-tunnel-general)# dhcp-server server1 [...server10]
hostname(config-tunnel-general)# address-pool [(interface name)] address_pool1
[...address_pool6]
hostname(config-tunnel-general)#
```

Note If you specify an interface name, you must enclosed it within parentheses.

You configure address pools with the **ip local pool** command in global configuration mode.

Step 7 Specify the name of the NAC authentication server group, if you are using Network Admission Control, to identify the group of authentication servers to be used for Network Admission Control posture validation. Configure at least one Access Control Server to support NAC. Use the **aaa-server** command to name the ACS group. Then use the **nac-authentication-server-group** command, using the same name for the server group.

The following example identifies acs-group1 as the authentication server group to be used for NAC posture validation:

```
ciscoasa(config-group-policy)# nac-authentication-server-group acs-group1
ciscoasa(config-group-policy)
```

The following example inherits the authentication server group from the default remote access group:

```
ciscoasa(config-group-policy)# no nac-authentication-server-group
ciscoasa(config-group-policy)
```

Note NAC requires a Cisco Trust Agent on the remote host.

Step 8 Specify whether to strip the group or the realm from the username before passing it on to the AAA server. The default is not to strip either the group name or the realm:

```
hostname(config-tunnel-general)# strip-group
hostname(config-tunnel-general)# strip-realm
hostname(config-tunnel-general)#
```

A realm is an administrative domain. If you strip the realm, the ASA uses the username and the group (if present) authentication. If you strip the group, the ASA uses the username and the realm (if present) for authentication. Enter the **strip-realm** command to remove the realm qualifier, and use the strip-group command to remove the group qualilfier from the username during authentication. If you remove both qualifiers, authentication is based on the *username* alone. Otherwise, authentication is based on the full *username@realm* or *username<delimiter> group* string. You must specify **strip-realm** if your server is unable to parse delimiters.

Step 9 Optionally, if your server is a RADIUS, RADIUS with NT, or LDAP server, you can enable password management.

Note If you are using an LDAP directory server for authentication, password management is supported with the Sun Microsystems 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.

See the general operations configuration guide for more information.

This feature, which is disabled by default, warns a user when the current password is about to expire. The default is to begin warning the user 14 days before expiration:

```
hostname(config-tunnel-general)# password-management
hostname(config-tunnel-general)#
```

If the server is an LDAP server, you can specify the number of days (0 through 180) before expiration to begin warning the user about the pending expiration:

```
hostname(config-tunnel-general)# password-management [password-expire in days n]
hostname(config-tunnel-general)#
```

Note The **password-management** command, entered in tunnel-group general-attributes configuration mode replaces the deprecated **radius-with-expiry** command that was formerly entered in tunnel-group ipsec-attributes mode.

When you configure the **password-management** command, 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. The ASA ignores this command if RADIUS or LDAP authentication has not been configured.

Note that this 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.

If you do specify the **password-expire-in-days** keyword, you must also specify the number of days.

Specifying this command with the number of days set to 0 disables this command. The ASA does not notify the user of the pending expiration, but the user can change the password after it expires.

See Configuring Microsoft Active Directory Settings for Password Management, page 4-28 for more information.

Note The ASA Version 7.1 and later generally supports password management for the AnyConnect VPN Client, the Cisco IPsec VPN Client, the SSL VPN full-tunneling client, and Clientless connections when authenticating with LDAP or with any RADIUS connection that supports MS-CHAPv2. Password management is *not* supported for any of these connection types for Kerberos/AD (Windows password) or NT 4.0 Domain.

Some RADIUS servers that support MS-CHAP do not currently support MS-CHAPv2. The **password-management** command requires MS-CHAPv2, so please check with your vendor.

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.

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. 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.

Step 10 Optionally, configure the ability to override an account-disabled indicator from a AAA server, by entering the **override-account-disable** command:

```
hostname(config-tunnel-general)# override-account-disable
hostname(config-tunnel-general)#
```

Note Allowing override-account-disable is a potential security risk.

Step 11 Specify the attribute or attributes to use in deriving a name for an authorization query from a certificate. This attribute specifies what part of the subject DN field to use as the username for authorization:

```
ciscoasa(config-tunnel-general)# authorization-dn-attributes {primary-attribute
[secondary-attribute] | use-entire-name}
```

For example, the following command specifies the use of the CN attribute as the username for authorization:

```
ciscoasa(config-tunnel-general)# authorization-dn-attributes CN
ciscoasa(config-tunnel-general)#
```

The authorization-dn-attributes are **C** (Country), **CN** (Common Name), **DNQ** (DN qualifier), **EA** (E-mail Address), **GENQ** (Generational qualifier), **GN** (Given Name), **I** (Initials), **L** (Locality), **N** (Name), **O** (Organization), **OU** (Organizational Unit), **SER** (Serial Number), **SN** (Surname), **SP** (State/Province), **T** (Title), **UID** (User ID), and **UPN** (User Principal Name).

Step 12 Specify whether to require a successful authorization before allowing a user to connect. The default is not to require authorization.

```
ciscoasa(config-tunnel-general)# authorization-required
ciscoasa(config-tunnel-general)#
```

Configuring Double Authentication

Double authentication is an optional feature that requires a user to enter an additional authentication credential, such as a second username and password, on the login screen. Specify the following commands to configure double authentication.

Step 1 Specify the secondary authentication server group. This command specifies the AAA server group to use as the secondary AAA server.

Note This command applies only to AnyConnect client VPN connections.

The secondary server group cannot specify an SDI server group. By default, no secondary authentication is required.

```
ciscoasa(config-tunnel-general)# secondary-authentication-server-group [interface_name]
{none | LOCAL | groupname [LOCAL]} [use-primary-name]
```

If you use the none keyword, no secondary authentication is required. The *groupname* value specifies the AAA server group name. Local specifies the use of the internal server database, and when used with the groupname value, LOCAL specifies fallback. For example, to set the primary authentication server group to sdi_group and the secondary authentication server group to ldap_server, enter the following commands:

```
ciscoasa(config-tunnel-general)# authentication-server-group
ciscoasa(config-tunnel-general)# secondary-authentication-server-group
```

Note If you use the **use-primary-name** keyword, then the login dialog requests only one username. In addition, if the usernames are extracted from a digital certificate, only the primary username is used for authentication.

Step 2 If obtaining the secondary username from a certificate, enter **secondary-username-from-certificate**:

```
ciscoasa(config-tunnel-general)# secondary-username-from-certificate C | CN | ... |
use-script
```

The values for the DN fields to extract from the certificate for use as a secondary username are the same as for the primary **username-from-certificate** command. Alternatively, you can specify the use-script keyword, which directs the ASA to use a script file generated by ASDM.

For example, to specify the Common Name as the primary username field and Organizational Unit as the secondary username field, enter the following commands:

```
ciscoasa(config-tunnel-general)# tunnel-group test1 general-attributes
ciscoasa(config-tunnel-general)# username-from-certificate cn
ciscoasa(config-tunnel-general)# secondary-username-from-certificate ou
```

Step 3 Use the **secondary-pre-fill-username** command in tunnel-group webvpn-attributes mode to enable extracting a secondary username from a client certificate for use in authentication. Use the keywords to specify whether this command applies to a clientless connection or an SSL VPN (AnyConnect) client connection and whether you want to hide the extracted username from the end user. This feature is disabled by default. Clientless and SSL-client options can both exist at the same time, but you must configure them in separate commands.

```
ciscoasa(config-tunnel-general)# secondary-pre-fill-username-from-certificate {clientless
| ssl-client} [hide]
```

For example, to specify the use of pre-fill-username for both the primary and secondary authentication for a connection, enter the following commands:

```
ciscoasa(config-tunnel-general)# tunnel-group test1 general-attributes
ciscoasa(config-tunnel-general)# pre-fill-username ssl-client
ciscoasa(config-tunnel-general)# secondary-pre-fill-username ssl-client
```

Step 4 Specify which authentication server to use to obtain the authorization attributes to apply to the connection. The primary authentication server is the default selection. This command is meaningful only for double authentication.

```
ciscoasa(config-tunnel-general)# authentication-attr-from-server {primary | secondary}
```

For example, to specify the use of the secondary authentication server, enter the following commands:

```
ciscoasa(config-tunnel-general)# tunnel-group test1 general-attributes
ciscoasa(config-tunnel-general)# authentication-attr-from-server secondary
```

Step 5 Specify which authentication username, primary or secondary, to associate with the session. The default value is primary. With double authentication enabled, it is possible that two distinct usernames are authenticated for the session. The administrator must designate one of the authenticated usernames as the session username. The session username is the username provided for accounting, session database, syslogs, and debug output.

```
ciscoasa(config-tunnel-general)# authenticated-session-username {primary | secondary}
```

For example, to specify that the authentication username associated with the session must come from the secondary authentication server, enter the following commands:

```
ciscoasa(config-tunnel-general)# tunnel-group test1 general-attributes
ciscoasa(config-tunnel-general)# authenticated-session-username secondary
```

Configuring Remote-Access Connection Profile IPsec IKEv1 Attributes

To configure the IPsec IKEv1 attributes for a remote-access connection profile, perform the following steps. The following description assumes that you have already created the remote-access connection profile. Remote-access connection profiles have more attributes than LAN-to-LAN connection profiles.

Step 1 To specify the IPsec attributes of an remote-access tunnel-group, enter tunnel-group ipsec-attributes mode by entering the following command in either single or multiple context mode. The prompt changes to indicate the mode change.

```
ciscoasa(config)# tunnel-group tunnel-group-name ipsec-attributes
ciscoasa(config-tunnel-ipsec)#
```

This command enters tunnel-group ipsec-attributes configuration mode, in which you configure the remote-access tunnel-group IPsec attributes in either single or multiple context mode.

For example, the following command designates that the tunnel-group ipsec-attributes mode commands that follow pertain to the connection profile named TG1. Notice that the prompt changes to indicate that you are now in tunnel-group ipsec-attributes mode:

```
ciscoasa(config)# tunnel-group TG1 type remote-access
ciscoasa(config)# tunnel-group TG1 ipsec-attributes
ciscoasa(config-tunnel-ipsec)#
```

Step 2 Specify the preshared key to support IKEv1 connections based on preshared keys. For example, the following command specifies the preshared key xyzx to support IKEv1 connections for an IPsec IKEv1 remote access connection profile:

```
ciscoasa(config-tunnel-ipsec)# ikev1 pre-shared-key xyzx
ciscoasa(config-tunnel-ipsec)#
```

Step 3 Specify whether to validate the identity of the peer using the peer's certificate:

```
ciscoasa(config-tunnel-ipsec)# peer-id-validate option
ciscoasa(config-tunnel-ipsec)#
```

The possible *option* values are **req** (required), **cert (**if supported by certificate), and **nocheck (**do not check). The default is **req**.

For example, the following command specifies that peer-id validation is required:

```
ciscoasa(config-tunnel-ipsec)# peer-id-validate req
ciscoasa(config-tunnel-ipsec)#
```

Step 4 Specify whether to enable sending of a certificate chain. The following command includes the root certificate and any subordinate CA certificates in the transmission:

```
ciscoasa(config-tunnel-ipsec)# chain
ciscoasa(config-tunnel-ipsec)#
```

This attribute applies to all IPsec tunnel-group types.

Step 5 Specify the name of a trustpoint that identifies the certificate to be sent to the IKE peer:

```
ciscoasa(config-tunnel-ipsec)# ikev1 trust-point trust-point-name
ciscoasa(config-tunnel-ipsec)#
```

The following command specifies mytrustpoint as the name of the certificate to be sent to the IKE peer:

```
ciscoasa(config-ipsec)# ikev1 trust-point mytrustpoint
```

Step 6 Specify the ISAKMP keepalive threshold and the number of retries allowed:

```
ciscoasa(config-tunnel-ipsec)# isakmp keepalive threshold <number> retry <number>
```

```
ciscoasa(config-tunnel-ipsec)#
```

The **threshold** parameter specifies the number of seconds (10 through 3600) that the peer is allowed to idle before beginning keepalive monitoring. The **retry** parameter is the interval (2 through 10 seconds) between retries after a keepalive response has not been received. IKE keepalives are enabled by default. To disable ISAKMP keepalives, enter **isakmp keepalive disable**.

For example, the following command sets the IKE keepalive threshold value to 15 seconds and sets the retry interval to 10 seconds:

```
ciscoasa(config-tunnel-ipsec)# isakmp keepalive threshold 15 retry 10
ciscoasa(config-tunnel-ipsec)#
```

The default value for the **threshold** parameter is 300 for remote-access and 10 for LAN-to-LAN, and the default value for the retry parameter is 2.

To specify that the central site (secure gateway) should never initiate ISAKMP monitoring, enter the following command:

```
ciscoasa(config-tunnel-ipsec)# isakmp keepalive threshold infinite
hostname(config-tunnel-ipsec)#
```

Step 7 Specify the ISAKMP hybrid authentication method, XAUTH or hybrid XAUTH.

You use **isakmp ikev1-user-authentication** command to implement hybrid XAUTH authentication when you need to use digital certificates for ASA authentication and a different, legacy method for remote VPN user authentication, such as RADIUS, TACACS+ or SecurID. Hybrid XAUTH breaks phase 1 of IKE down into the following two steps, together called hybrid authentication:

a. The ASA authenticates to the remote VPN user with standard public key techniques. This establishes an IKE security association that is unidirectionally authenticated.

b. An XAUTH exchange then authenticates the remote VPN user. This extended authentication can use one of the supported legacy authentication methods.

Note Before the authentication type can be set to hybrid, you must configure the authentication server, create a preshared key, and configure a trustpoint.

You can use the **isakmp ikev1-user-authentication** command with the optional **interface** parameter to specify a particular interface. When you omit the **interface** parameter, the command applies to all the interfaces and serves as a back-up when the per-interface command is not specified. When there are two **isakmp ikev1-user-authentication** commands specified for a connection profile, and one uses the **interface** parameter and one does not, the one specifying the interface takes precedence for that particular interface.

For example, the following commands enable hybrid XAUTH on the inside interface for a connection profile called example-group:

```
ciscoasa(config)# tunnel-group example-group type remote-access
ciscoasa(config)# tunnel-group example-group ipsec-attributes
ciscoasa(config-tunnel-ipsec)# isakmp ikev1-user-authentication (inside) hybrid
ciscoasa(config-tunnel-ipsec)#
```

Configuring IPsec Remote-Access Connection Profile PPP Attributes

To configure the Point-to-Point Protocol attributes for a remote-access connection profile, perform the following steps. PPP attributes apply *only* to IPsec remote-access connection profiles. The following description assumes that you have already created the IPsec remote-access connection profile.

Step 1 Enter tunnel-group ppp-attributes configuration mode, in which you configure the remote-access tunnel-group PPP attributes, by entering the following command. The prompt changes to indicate the mode change:

```
ciscoasa(config)# tunnel-group tunnel-group-name type remote-access
ciscoasa(config)# tunnel-group tunnel-group-name ppp-attributes
ciscoasa(config-tunnel-ppp)#
```

For example, the following command designates that the tunnel-group ppp-attributes mode commands that follow pertain to the connection profile named TG1. Notice that the prompt changes to indicate that you are now in tunnel-group ppp-attributes mode:

```
ciscoasa(config)# tunnel-group TG1 type remote-access
ciscoasa(config)# tunnel-group TG1 ppp-attributes
ciscoasa(config-tunnel-ppp)#
```

Step 2 Specify whether to enable authentication using specific protocols for the PPP connection. The protocol value can be any of the following:

- pap—Enables the use of Password Authentication Protocol for the PPP connection.
- chap—Enables the use of Challenge Handshake Authentication Protocol for the PPP connection.
- ms-chap-v1 or ms-chap-v2—Enables the use of Microsoft Challenge Handshake Authentication Protocol, version 1 or version 2 for the PPP connection.
- eap—Enables the use of Extensible Authentication protocol for the PPP connection.

CHAP and MSCHAPv1 are enabled by default.

The syntax of this command is:

```
ciscoasa(config-tunnel-ppp)# authentication protocol
ciscoasa(config-tunnel-ppp)#
```

To disable authentication for a specific protocol, use the **no** form of the command:

```
ciscoasa(config-tunnel-ppp)# no authentication protocol
ciscoasa(config-tunnel-ppp)#
```

For example, the following command enables the use of the PAP protocol for a PPP connection:

```
ciscoasa(config-tunnel-ppp)# authentication pap
ciscoasa(config-tunnel-ppp)#
```

The following command enables the use of the MS-CHAP, version 2 protocol for a PPP connection:

```
ciscoasa(config-tunnel-ppp)# authentication ms-chap-v2
ciscoasa(config-tunnel-ppp)#
```

The following command enables the use of the EAP-PROXY protocol for a PPP connection:

```
ciscoasa(config-tunnel-ppp)# authentication pap
ciscoasa(config-tunnel-ppp)#
```

The following command disables the use of the MS-CHAP, version 1 protocol for a PPP connection:

```
ciscoasa(config-tunnel-ppp)# no authentication ms-chap-v1
ciscoasa(config-tunnel-ppp)#
```

Configuring LAN-to-LAN Connection Profiles

An IPsec LAN-to-LAN VPN connection profile applies only to LAN-to-LAN IPsec client connections. While many of the parameters that you configure are the same as for IPsec remote-access connection profiles, LAN-to-LAN tunnels have fewer parameters. The following sections show you how to configure a LAN-to-LAN connection profile:

Default LAN-to-LAN Connection Profile Configuration

The contents of the default LAN-to-LAN connection profile are as follows:

```
tunnel-group DefaultL2LGroup type ipsec-l2l
tunnel-group DefaultL2LGroup general-attributes
 no accounting-server-group
 default-group-policy DfltGrpPolicy
tunnel-group DefaultL2LGroup ipsec-attributes
 no ikev1 pre-shared-key
 peer-id-validate req
 no chain
 no ikev1 trust-point
 isakmp keepalive threshold 10 retry 2
```

LAN-to-LAN connection profiles have fewer parameters than remote-access connection profiles, and most of these are the same for both groups. For your convenience in configuring the connection, they are listed separately here. Any parameters that you do not explicitly configure inherit their values from the default connection profile.

Specifying a Name and Type for a LAN-to-LAN Connection Profile

To specify a name and a type for a connection profile, enter the **tunnel-group** command, as follows:

```
hostname(config)# tunnel-group tunnel_group_name type tunnel_type
```

For a LAN-to-LAN tunnel, the type is **ipsec-l2l**.; for example, to create the LAN-to-LAN connection profile named docs, enter the following command:

```
hostname(config)# tunnel-group docs type ipsec-l2l
hostname(config)#
```

Configuring LAN-to-LAN Connection Profile General Attributes

To configure the connection profile general attributes, perform the following steps:

Step 1 Enter tunnel-group general-attributes mode by specifying the general-attributes keyword in either single or multiple context mode:

```
hostname(config)# tunnel-group_tunnel-group-name general-attributes
hostname(config-tunnel-general)#
```

The prompt changes to indicate that you are now in config-general mode, in which you configure the tunnel-group general attributes.

For example, for the connection profile named docs, enter the following command:

```
hostname(config)# tunnel-group_docs general-attributes
hostname(config-tunnel-general)#
```

Step 2 Specify the name of the accounting-server group, if any, to use:

```
hostname(config-tunnel-general)# accounting-server-group groupname
hostname(config-tunnel-general)#
```

For example, the following command specifies the use of the accounting-server group acctgserv1:

```
hostname(config-tunnel-general)# accounting-server-group acctgserv1
hostname(config-tunnel-general)#
```

Step 3 Specify the name of the default group policy:

```
hostname(config-tunnel-general)# default-group-policy policyname
hostname(config-tunnel-general)#
```

For example, the following command specifies that the name of the default group policy is MyPolicy:

```
ciscoasa(config-tunnel-general)# default-group-policy MyPolicy
hostname(config-tunnel-general)#
```

Configuring LAN-to-LAN IPsec IKEv1 Attributes

To configure the IPsec IKEv1 attributes, perform the following steps:

Step 1 To configure the tunnel-group IPsec IKEv1 attributes, enter tunnel-group ipsec-attributes configuration mode by entering the tunnel-group command with the IPsec-attributes keyword in either single or multiple context mode.

```
ciscoasa(config)# tunnel-group tunnel-group-name ipsec-attributes
hostname(config-tunnel-ipsec)#
```

For example, the following command enters config-ipsec mode so that you can configure the parameters for the connection profile named TG1:

```
ciscoasa(config)# tunnel-group TG1 ipsec-attributes
hostname(config-tunnel-ipsec)#
```

The prompt changes to indicate that you are now in tunnel-group ipsec-attributes configuration mode.

Step 2 Specify the preshared key to support IKEv1 connections based on preshared keys.

```
ciscoasa(config-tunnel-ipsec)# ikev1 pre-shared-key key
hostname(config-tunnel-ipsec)#
```

For example, the following command specifies the preshared key XYZX to support IKEv1 connections for an LAN-to-LAN connection profile:

```
ciscoasa(config-tunnel-ipsec)# ikev1 pre-shared-key xyzx
hostname(config-tunnel-general)#
```

Step 3 Specify whether to validate the identity of the peer using the peer's certificate:

```
ciscoasa(config-tunnel-ipsec)# peer-id-validate option
hostname(config-tunnel-ipsec)#
```

The available options are **req** (required), **cert (**if supported by certificate), and **nocheck (**do not check). The default is **req**. For example, the following command sets the peer-id-validate option to **nocheck**:

```
ciscoasa(config-tunnel-ipsec)# peer-id-validate nocheck
hostname(config-tunnel-ipsec)#
```

Step 4 Specify whether to enable sending of a certificate chain. This action includes the root certificate and any subordinate CA certificates in the transmission:

```
ciscoasa(config-tunnel-ipsec)# chain
hostname(config-tunnel-ipsec)#
```

You can apply this attribute to all tunnel-group types.

Step 5 Specify the name of a trustpoint that identifies the certificate to be sent to the IKE peer:

```
ciscoasa(config-tunnel-ipsec)# trust-point trust-point-name
hostname(config-tunnel-ipsec)#
```

For example, the following command sets the trustpoint name to mytrustpoint:

```
ciscoasa(config-tunnel-ipsec)# trust-point mytrustpoint
hostname(config-tunnel-ipsec)#
```

You can apply this attribute to all tunnel-group types.

Step 6 Specify the ISAKMP (IKE) keepalive threshold and the number of retries allowed. The **threshold** parameter specifies the number of seconds (10 through 3600) that the peer is allowed to idle before beginning keepalive monitoring. The **retry** parameter is the interval (2 through 10 seconds) between retries after a keepalive response has not been received. IKE keepalives are enabled by default. To disable IKE keepalives, enter the **no** form of the **isakmp** command:

```
hostname(config)# isakmp keepalive threshold <number> retry <number>
hostname(config-tunnel-ipsec)#
```

For example, the following command sets the ISAKMP keepalive threshold to 15 seconds and sets the retry interval to 10 seconds:

```
ciscoasa(config-tunnel-ipsec)# isakmp keepalive threshold 15 retry 10
hostname(config-tunnel-ipsec)#
```

The default value for the **threshold** parameter for LAN-to-LAN is 10, and the default value for the retry parameter is 2.

To specify that the central site (secure gateway) should never initiate ISAKMP monitoring, enter the following command:

```
ciscoasa(config-tunnel-ipsec)# isakmp keepalive threshold infinite
hostname(config-tunnel-ipsec)#
```

Step 7 Specify the ISAKMP hybrid authentication method, XAUTH or hybrid XAUTH.

You use **isakmp ikev1-user-authentication** command to implement hybrid XAUTH authentication when you need to use digital certificates for ASA authentication and a different, legacy method for remote VPN user authentication, such as RADIUS, TACACS+ or SecurID. Hybrid XAUTH breaks phase 1 of IKE down into the following two steps, together called hybrid authentication:

a. The ASA authenticates to the remote VPN user with standard public key techniques. This establishes an IKE security association that is unidirectionally authenticated.

b. An XAUTH exchange then authenticates the remote VPN user. This extended authentication can use one of the supported legacy authentication methods.

Note Before the authentication type can be set to hybrid, you must configure the authentication server, create a preshared key, and configure a trustpoint.

For example, the following commands enable hybrid XAUTH for a connection profile called example-group:

```
ciscoasa(config)# tunnel-group example-group type remote-access
ciscoasa(config)# tunnel-group example-group ipsec-attributes
ciscoasa(config-tunnel-ipsec)# isakmp ikev1-user-authentication hybrid
ciscoasa(config-tunnel-ipsec)#
```

Configuring Connection Profiles for Clientless SSL VPN Sessions

The tunnel-group general attributes for clientless SSL VPN connection profiles are the same as those for IPsec remote-access connection profiles, except that the tunnel-group type is webvpn and the **strip-group** and **strip-realm** commands do not apply. You define the attribute specific to clientless SSL VPN separately. The following sections describe how to configure clientless SSL VPN connection profiles:

- Configuring General Tunnel-Group Attributes for Clientless SSL VPN Sessions, page 4-20
- Configuring Tunnel-Group Attributes for Clientless SSL VPN Sessions, page 4-23

Configuring General Tunnel-Group Attributes for Clientless SSL VPN Sessions

To configure or change the connection profile general attributes, specify the parameters in the following steps.

Step 1 To configure the general attributes, enter **tunnel-group general-attributes** command, which enters tunnel-group general-attributes configuration mode in either single or multiple context mode. Note that the prompt changes:

```
hostname(config)# tunnel-group tunnel_group_name general-attributes
hostname(config-tunnel-general)#
```

To configure the general attributes for TunnelGroup3, created in the previous section, enter the following command:

```
hostname(config)# tunnel-group TunnelGroup3 general-attributes
hostname(config-tunnel-general)#
```

Step 2 Specify the name of the authentication-server group, if any, to use. If you want to use the LOCAL database for authentication if the specified server group fails, append the keyword LOCAL:

```
hostname(config-tunnel-general)# authentication-server-group groupname [LOCAL]
hostname(config-tunnel-general)#
```

For example, to configure the authentication server group named test, and to provide fallback to the LOCAL server if the authentication server group fails, enter the following command:

```
hostname(config-tunnel-general)# authentication-server-group test LOCAL
hostname(config-tunnel-general)#
```

The authentication-server-group name identifies a previously configured authentication server or group of servers. Use the **aaa-server** command to configure authentication servers. The maximum length of the group tag is 16 characters.

You can also configure interface-specific authentication by including the name of an interface in parentheses before the group name. The following interfaces are available by default:

- inside—Name of interface GigabitEthernet0/1
- outside— Name of interface GigabitEthernet0/0

Note The ASA's outside interface address (for both IPv4/IPv6) cannot overlap with the private side address space.

Other interfaces you have configured (using the **interface** command) are also available. The following command configures interface-specific authentication for the interface named outside using the server servergroup1 for authentication:

```
hostname(config-tunnel-general)# authentication-server-group (outside) servergroup1
hostname(config-tunnel-general)#
```

Step 3 Optionally, specify the name of the authorization-server group, if any, to use. If you are not using authorization, go to Step 6. When you configure this value, users must exist in the authorization database to connect:

```
hostname(config-tunnel-general)# authorization-server-group groupname
hostname(config-tunnel-general)#
```

Use the **aaa-server** command to configure authorization servers. The maximum length of the group tag is 16 characters.

For example, the following command specifies the use of the authorization-server group FinGroup:

```
hostname(config-tunnel-general)# authorization-server-group FinGroup
hostname(config-tunnel-general)#
```

Step 4 Specify whether to require a successful authorization before allowing a user to connect. The default is not to require authorization.

```
ciscoasa(config-tunnel-general)# authorization-required
ciscoasa(config-tunnel-general)#
```

Step 5 Specify the attribute or attributes to use in deriving a name for an authorization query from a certificate. This attribute specifies what part of the subject DN field to use as the username for authorization:

```
ciscoasa(config-tunnel-general)# authorization-dn-attributes {primary-attribute
[secondary-attribute] | use-entire-name}
```

For example, the following command specifies the use of the CN attribute as the username for authorization:

```
ciscoasa(config-tunnel-general)# authorization-dn-attributes CN
ciscoasa(config-tunnel-general)#
```

The authorization-dn-attributes are **C** (Country), **CN** (Common Name), **DNQ** (DN qualifier), **EA** (E-mail Address), **GENQ** (Generational qualifier), **GN** (Given Name), **I** (Initials), **L** (Locality), **N** (Name), **O** (Organization), **OU** (Organizational Unit), **SER** (Serial Number), **SN** (Surname), **SP** (State/Province), **T** (Title), **UID** (User ID), and **UPN** (User Principal Name).

Step 6 Optionally, specify the name of the accounting-server group, if any, to use. If you are not using accounting, go to Step 7. Use the **aaa-server** command to configure accounting servers. The maximum length of the group tag is 16 characters.:

```
hostname(config-tunnel-general)# accounting-server-group groupname
hostname(config-tunnel-general)#
```

For example, the following command specifies the use of the accounting-server group comptroller:

```
hostname(config-tunnel-general)# accounting-server-group comptroller
hostname(config-tunnel-general)#
```

Step 7 Optionally, specify the name of the default group policy. The default value is DfltGrpPolicy:

```
hostname(config-tunnel-general)# default-group-policy policyname
hostname(config-tunnel-general)#
```

The following example sets MyDfltGrpPolicy as the name of the default group policy:

```
ciscoasa(config-tunnel-general)# default-group-policy MyDfltGrpPolicy
hostname(config-tunnel-general)#
```

Step 8 Optionally, specify the name or IP address of the DHCP server (up to 10 servers), and the names of the DHCP address pools (up to 6 pools). Separate the list items with spaces. The defaults are no DHCP server and no address pool.

```
hostname(config-tunnel-general)# dhcp-server server1 [...server10]
hostname(config-tunnel-general)# address-pool [(interface name)] address_pool1
[...address_pool6]
hostname(config-tunnel-general)#
```

Note The interface name must be enclosed in parentheses.

You configure address pools with the **ip local pool** command in global configuration mode. See Chapter 5, "Configuring IP Addresses for VPNs" for information about configuring address pools.

Step 9 Optionally, if your server is a RADIUS, RADIUS with NT, or LDAP server, you can enable password management.

Note If you are using an LDAP directory server for authentication, password management is supported with the Sun Microsystems 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.

See the general operations configuration guide for more information.

This feature, which is enabled by default, warns a user when the current password is about to expire. The default is to begin warning the user 14 days before expiration:

```
hostname(config-tunnel-general)# password-management
hostname(config-tunnel-general)#
```

If the server is an LDAP server, you can specify the number of days (0 through 180) before expiration to begin warning the user about the pending expiration:

```
hostname(config-tunnel-general)# password-management [password-expire in days n]
hostname(config-tunnel-general)#
```

Note The **password-management** command, entered in tunnel-group general-attributes configuration mode replaces the deprecated **radius-with-expiry** command that was formerly entered in tunnel-group ipsec-attributes mode.

When you configure this command, 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. The ASA ignores this command if RADIUS or LDAP authentication has not been configured.

Note that this 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.

If you do specify the **password-expire-in-days** keyword, you must also specify the number of days.

See Configuring Microsoft Active Directory Settings for Password Management, page 4-28 for more information.

Step 10 Specifying this command with the number of days set to 0 disables this command. The ASA does not notify the user of the pending expiration, but the user can change the password after it expires.Optionally, configure the ability to override an account-disabled indicator from the AAA server, by entering the **override-account-disable** command:

```
hostname(config-tunnel-general)# override-account-disable
hostname(config-tunnel-general)#
```

Note Allowing override account-disabled is a potential security risk.

Configuring Tunnel-Group Attributes for Clientless SSL VPN Sessions

To configure the parameters specific to a clientless SSL VPN connection profile, follow the steps in this section. Clientless SSL VPN was formerly known as WebVPN, and you configure these attributes in tunnel-group webvpn-attributes mode.

Step 1 To specify the attributes of a clientless SSL VPN tunnel-group, enter tunnel-group webvpn-attributes mode by entering the following command. The prompt changes to indicate the mode change:

```
ciscoasa(config)# tunnel-group tunnel-group-name webvpn-attributes
ciscoasa(config-tunnel-ipsec)#
```

For example, to specify the webvpn-attributes for the clientless SSL VPN tunnel-group named sales, enter the following command:

```
ciscoasa(config)# tunnel-group sales webvpn-attributes
ciscoasa(config-tunnel-webvpn)#
```

Step 2 To specify the authentication method to use: AAA, digital certificates, or both, enter the **authentication** command. You can specify either aaa or certificate or both, in any order.

```
ciscoasa(config-tunnel-webvpn)# authentication authentication_method
ciscoasa(config-tunnel-webvpn)#
```

For example, The following command allows both AAA and certificate authentication:

```
ciscoasa(config-tunnel-webvpn)# authentication aaa certificate
ciscoasa(config-tunnel-webvpn)#
```

Applying Customization

Customizations determine the appearance of the windows that the user sees upon login. You configure the customization parameters as part of configuring clientless SSL VPN.

To apply a previously defined web-page customization to change the look-and-feel of the web page that the user sees at login, enter the customization command in username webvpn configuration mode:

```
ciscoasa(config-username-webvpn)# customization {none | value customization_name}
ciscoasa(config-username-webvpn)#
```

For example, to use the customization named blueborder, enter the following command:

```
ciscoasa(config-username-webvpn)# customization value blueborder
ciscoasa(config-username-webvpn)#
```

You configure the customization itself by entering the **customization** command in webvpn mode.

The following example shows a command sequence that first establishes a customization named “123” that defines a password prompt. The example then defines a clientless SSL VPN tunnel-group named “test” and uses the **customization** command to specify the use of the customization named “123”:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# customization 123
ciscoasa(config-webvpn-custom)# password-prompt Enter password
ciscoasa(config-webvpn)# exit
ciscoasa(config)# tunnel-group test type webvpn
ciscoasa(config)# tunnel-group test webvpn-attributes
ciscoasa(config-tunnel-webvpn)# customization value 123
ciscoasa(config-tunnel-webvpn)#
```

Step 3 The ASA queries NetBIOS name servers to map NetBIOS names to IP addresses. Clientless SSL VPN requires NetBIOS to access or share files on remote systems. Clientless SSL VPN uses NetBIOS and the CIFS protocol to access or share files on remote systems. When you attempt a file-sharing connection to a Windows computer by using its computer name, the file server you specify corresponds to a specific NetBIOS name that identifies a resource on the network.

To make the NBNS function operational, you must configure at least one NetBIOS server (host). You can configure up to three NBNS servers for redundancy. The ASA uses the first server on the list for NetBIOS/CIFS name resolution. If the query fails, it uses the next server.

To specify the name of the NBNS (NetBIOS Name Service) server to use for CIFS name resolution, use the **nbns-server** command. You can enter up to three server entries. The first server you configure is the primary server, and the others are backups, for redundancy. You can also specify whether this is a master browser (rather than just a WINS server), the timeout interval, and the number of retries. A WINS server or a master browser is typically on the same network as the ASA, or reachable from that network. You must specify the timeout interval before the number of retries:

```
ciscoasa(config-tunnel-webvpn)# nbns-server {host-name | IP_address} [master]
[timeout seconds] [retry number]
ciscoasa(config-tunnel-webvpn)#
```

For example, to configure the server named nbnsprimary as the primary server and the server 192.168.2.2 as the secondary server, each allowing three retries and having a 5-second timeout, enter the following command:

```
ciscoasa(config)# name 192.168.2.1 nbnsprimary
ciscoasa(config-tunnel-webvpn)# nbns-server nbnsprimary master timeout 5 retry 3
ciscoasa(config-tunnel-webvpn)# nbns-server 192.168.2.2 timeout 5 retry 3
ciscoasa(config-tunnel-webvpn)#
```

The timeout interval can range from 1 through 30 seconds (default 2), and the number of retries can be in the range 0 through 10 (default 2).

The **nbns-server** command in tunnel-group webvpn-attributes configuration mode replaces the deprecated **nbns-server** command in webvpn configuration mode.

Step 4 To specify alternative names for the group, use the **group-alias** command. Specifying the group alias creates one or more alternate names by which the user can refer to a tunnel-group. The group alias that you specify here appears in the drop-down list on the user's login page. Each group can have multiple aliases or no alias, each specified in separate commands. This feature is useful when the same group is known by several common names, such as "Devtest" and "QA".

For each group alias, enter a **group-alias** command. Each alias is enabled by default. You can optionally explicitly enable or disable each alias:

```
ciscoasa(config-tunnel-webvpn)# group-alias alias [enable | disable]
ciscoasa(config-tunnel-webvpn)#
```

For example, to enable the aliases QA and Devtest for a tunnel-group named QA, enter the following commands:

```
ciscoasa(config-tunnel-webvpn)# group-alias QA enable
ciscoasa(config-tunnel-webvpn)# group-alias Devtest enable
ciscoasa(config-tunnel-webvpn)#
```

Note The webvpn tunnel-group-list must be enabled for the (dropdown) group list to appear.

Step 5 To specify incoming URLs or IP addresses for the group, use the **group-url** command. Specifying a group URL or IP address eliminates the need for the user to select a group at login. When a user logs in, the ASA looks for the user's incoming URL or address in the tunnel-group-policy table. If it finds the URL or address and if group-url is enabled in the connection profile, then the ASA automatically selects the associated connection profile and presents the user with only the username and password fields in the login window. This simplifies the user interface and has the added advantage of never exposing the list of groups to the user. The login window that the user sees uses the customizations configured for that connection profile.

If the URL or address is disabled and group-alias is configured, then the dropdown list of groups is also displayed, and the user must make a selection.

You can configure multiple URLs or addresses (or none) for a group. Each URL or address can be enabled or disabled individually. You must use a separate **group-url** command for each URL or address specified. You must specify the entire URL or address, including either the http or https protocol.

You cannot associate the same URL or address with multiple groups. The ASA verifies the uniqueness of the URL or address before accepting the URL or address for a connection profile.

For each group URL or address, enter a **group-url** command. You can optionally explicitly enable (the default) or disable each URL or alias:

```
ciscoasa(config-tunnel-webvpn)# group-url url [enable | disable]
ciscoasa(config-tunnel-webvpn)#
```

Url specifies a URL or IP address for this tunnel group.

For example, to enable the group URLs http://www.example.com and http://192.168.10.10 for the tunnel-group named RadiusServer, enter the following commands:

```
ciscoasa(config)# tunnel-group RadiusServer type webvpn
ciscoasa(config)# tunnel-group RadiusServer general-attributes
ciscoasa(config-tunnel-general)# authentication server-group RADIUS
ciscoasa(config-tunnel-general)# accounting-server-group RADIUS
ciscoasa(config-tunnel-general)# tunnel-group RadiusServer webvpn-attributes
ciscoasa(config-tunnel-webvpn)# group-alias "Cisco Remote Access" enable
ciscoasa(config-tunnel-webvpn)# group-url http://www.example.com enable
ciscoasa(config-tunnel-webvpn)# group-url http://192.168.10.10 enable
ciscoasa(config-tunnel-webvpn)#
```

For a more extensive example, see Customizing Login Windows for Users of Clientless SSL VPN Sessions, page 4-27.

Step 6 To exempt certain users from running Cisco Secure Desktop on a per connection profile basis if they enter one of the group-urls, enter the following command:

```
ciscoasa(config-tunnel-webvpn)# without-csd
ciscoasa(config-tunnel-webvpn)#
```

Note Entering this command prevents the detection of endpoint conditions for these sessions, so you may need to adjust the dynamic access policy (DAP) configuration.

Step 7 To specify the DNS server group to use for a connection profile for clientless SSL VPN sessions, use the **dns-group** command. The group you specify must be one you already configured in global configuration mode (using the **dns server-group** and **name-server** commands).

By default, the connection profile uses the DNS server group *DefaultDNS*. However, this group must be configured before the security appliance can resolve DNS requests.

The following example configures a new DNS server group named *corp_dns* and specifies that server group for the connection profile *telecommuters*:

```
hostname(config)# dns server-group corp_dns
hostname(config-dns-server-group)# domain-name cisco.com
hostname(config-dns-server-group)# name-server 209.165.200.224

hostname(config)# tunnel-group telecommuters webvpn-attributes
hostname(config-tunnel-webvpn)# dns-group corp_dns
hostname(config-tunnel-webvpn)#
```

Step 8 (Optional) To enable extracting a username from a client certificate for use in authentication and authorization, use the **pre-fill-username** command in tunnel-group webvpn-attributes mode. There is no default value.

```
ciscoasa(config)# pre-fill-username {ssl-client | clientless}
```

The **pre-fill-username** command enables the use of a username extracted from the certificate field specified in the **username-from-certificate** command (in tunnel-group general-attributes mode) as the username for username/password authentication and authorization. To use this pre-fill username from certificate feature, you must configure both commands.

Note In Version 8.0.4, the username is not pre-filled; instead, any data sent in the username field is ignored.

The following example, entered in global configuration mode, creates an IPsec remote access tunnel group named remotegrp, enables getting the username from a certificate, and specifies that the name for an authentication or authorization query for an SSL VPN client must be derived from a digital certificate:

```
ciscoasa(config)# tunnel-group remotegrp type ipsec_ra
ciscoasa(config)# tunnel-group remotegrp general-attributes
ciscoasa(config-tunnel-general)# username-from-certificate CN OU
ciscoasa(config)# tunnel-group remotegrp webvpn-attributes
ciscoasa(config-tunnel-webvpn)# pre-fill-username ssl-client
ciscoasa(config-tunnel-webvpn)#
```

Step 9 (Optional) To specify whether to override the group policy or username attributes configuration for downloading an AnyConnect or SSL VPN client, use the **override-svc-download** command. This feature is disabled by default.

The security appliance allows clientless or AnyConnect client connections for remote users based on whether clientless and/or SSL VPN is enabled in the group policy or username attributes with the **vpn-tunnel-protocol** command. The **anyconnect ask** command further modifies the client user experience by prompting the user to download the client or return to the WebVPN home page.

However, you might want clientless users logging in under specific tunnel groups to not experience delays waiting for the download prompt to expire before being presented with the clientless SSL VPN home page. You can prevent delays for these users at the connection profile level with the **override-svc-download** command. This command causes users logging through a connection profile to be immediately presented with the clientless SSL VPN home page regardless of the **vpn-tunnel-protocol** or **anyconnect ask** command settings.

In the following example, the you enter tunnel-group webvpn attributes configuration mode for the connection profile *engineering* and enable the connection profile to override the group policy and username attribute settings for client download prompts:

```
hostname(config)# tunnel-group engineering webvpn-attributes
hostname(config-tunnel-webvpn)# override-svc-download
```

Step 10 (Optional) To enable the display of a RADIUS reject message on the login screen when authentication is rejected, use the **radius-eject-message** command.

The following example enables the display of a RADIUS rejection message for the connection profile named engineering:

```
hostname(config)# tunnel-group engineering webvpn-attributes
hostname(config-tunnel-webvpn)# radius-reject-message
```

Customizing Login Windows for Users of Clientless SSL VPN Sessions

You can set up different login windows for different groups by using a combination of customization profiles and connection profiles. For example, assuming that you had created a customization profile called salesgui, you can create a connection profile for clientless SSL VPN sessions called sales that uses that customization profile, as the following example shows:

Step 1 In webvpn mode, define a customization for clientless SSL VPN access, in this case named salesgui and change the default logo to mycompanylogo.gif. You must have previously loaded mycompanylogo.gif onto the flash memory of the ASA and saved the configuration. See Chapter 14, “Introduction to Clientless SSL VPN” for details.

```
ciscoasa# webvpn
```

```
hostname (config-webvpn)# customization value salesgui
ciscoasa(config-webvpn-custom)# logo file disk0:\mycompanylogo.gif
ciscoasa(config-webvpn-custom)#
```

Step 2 In global configuration mode, set up a username and associate with it the customization for clientless SSL VPN that you have just defined:

```
ciscoasa# username seller attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# customization value salesgui
ciscoasa(config-username-webvpn)# exit
ciscoasa(config-username)# exit
ciscoasa#
```

Step 3 In global configuration mode, create a tunnel-group for clientless SSL VPN sessions named sales:

```
ciscoasa# tunnel-group sales type webvpn
ciscoasa(config-tunnel-webvpn)#
```

Step 4 Specify that you want to use the salesgui customization for this connection profile:

```
ciscoasa# tunnel-group sales webvpn-attributes
ciscoasa(config-tunnel-webvpn)# customization salesgui
```

Step 5 Set the group URL to the address that the user enters into the browser to log in to the ASA; for example, if the ASA has the IP address 192.168.3.3, set the group URL to https://192.168.3.3:

```
ciscoasa(config-tunnel-webvpn)# group-url https://192.168.3.3.
ciscoasa(config-tunnel-webvpn)#
```

If a port number is required for a successful login, include the port number, preceded by a colon. The ASA maps this URL to the sales connection profile and applies the salesgui customization profile to the login screen that the user sees upon logging in to https://192.168.3.3.

Configuring Microsoft Active Directory Settings for Password Management

Note If you are using an LDAP directory server for authentication, password management is supported with the Sun Microsystems 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.

See the general operations configuration guide for more information.

To use password management with Microsoft Active Directory, you must set certain Active Directory parameters as well as configuring password management on the ASA. This section describes the Active Directory settings associated with various password management actions. These descriptions assume

that you have also enabled password management on the ASA and configured the corresponding password management attributes. The specific steps in this section refer to Active Directory terminology under Windows 2000 and include the following topics:

- Using Active Directory to Force the User to Change Password at Next Logon, page 4-29.
- Using Active Directory to Specify Maximum Password Age, page 4-30.
- Using Active Directory to Override an Account Disabled AAA Indicator, page 4-31
- Using Active Directory to Enforce Password Complexity, page 4-33.

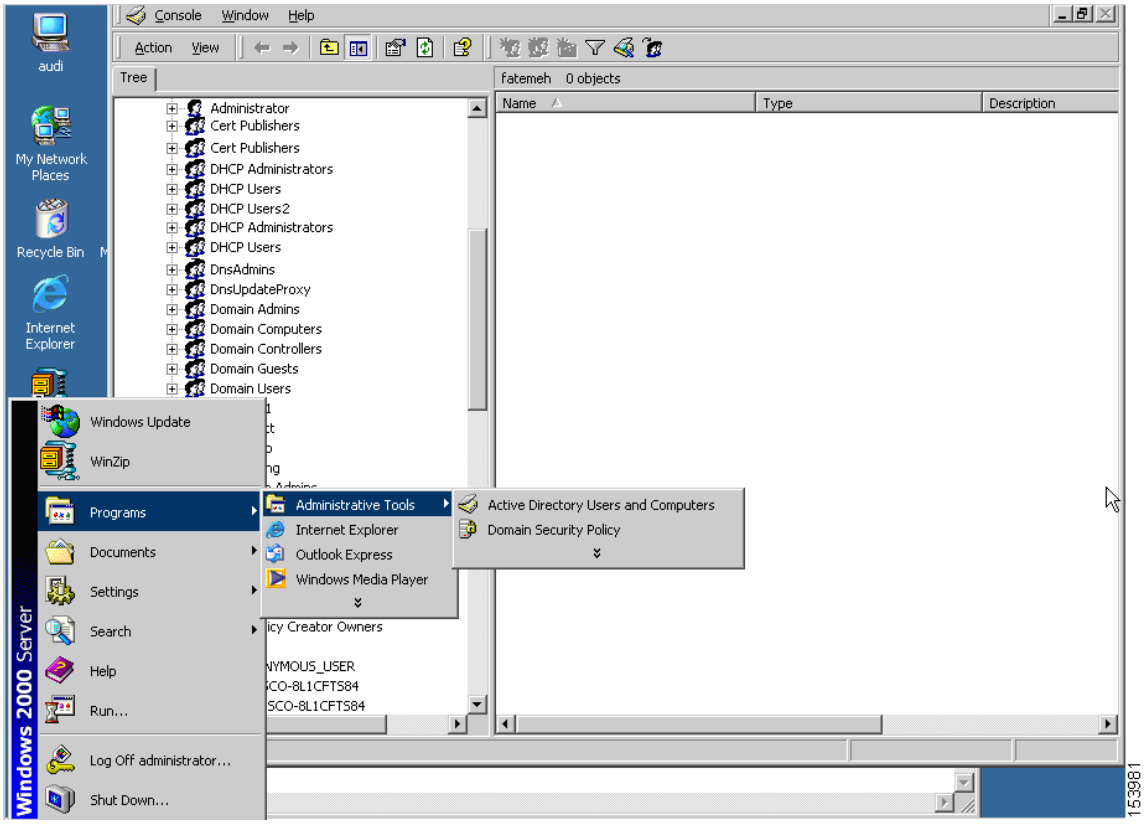
This section assumes that you are using an LDAP directory server for authentication.

Using Active Directory to Force the User to Change Password at Next Logon

To force a user to change the user password at the next logon, specify the **password-management** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory:

Step 1 Choose **Start > Programs > Administrative Tools > Active Directory Users and Computers** (Figure 4-1).

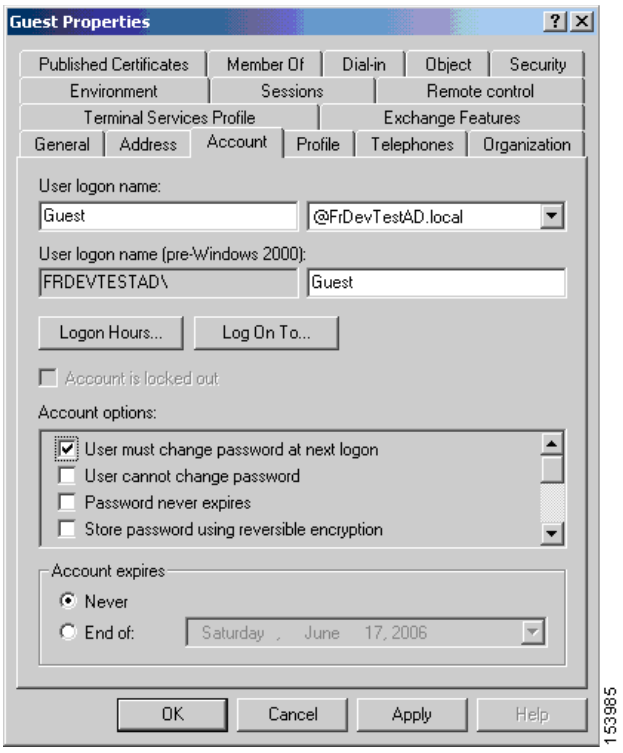
Figure 4-1 Active Directory—Administrative Tools Menu



Step 2 Right-click to choose **Username > Properties > Account**.

Step 3 Check the **User must change password at next logon** (Figure 4-2) check box.

Figure 4-2 Active Directory—User Must Change Password at Next Logon



The next time this user logs on, the ASA displays the following prompt: "New password required. Password change required. You must enter a new password with a minimum length *n* to continue." You can set the minimum required password length, *n*, as part of the Active Directory configuration at Start > Programs > Administrative Tools > Domain Security Policy > Windows Settings > Security Settings > Account Policies > Password Policy. Select **Minimum password length**.

Using Active Directory to Specify Maximum Password Age

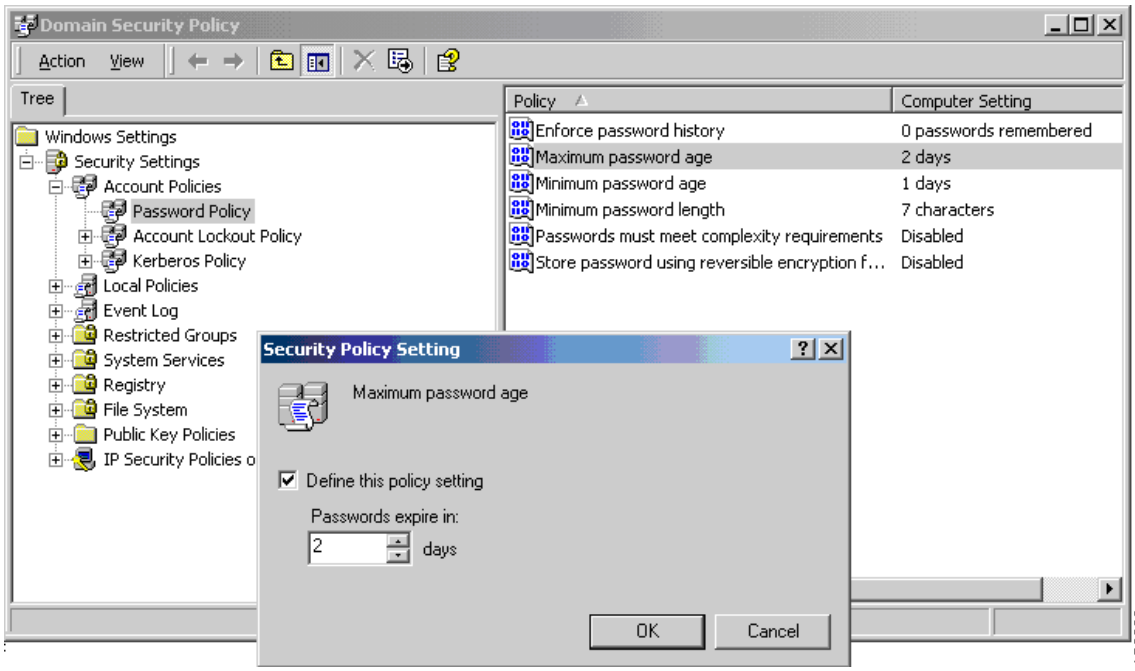
To enhance security, you can specify that passwords expire after a certain number of days. To specify a maximum password age for a user password, specify the **password-management** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory:

Step 1 Choose **Start > Programs > Administrative Tools > Domain Security Policy > Windows Settings > Security Settings > Account Policies > Password Policy**.

Step 2 Double-click Maximum password age. The Security Policy Setting dialog box appears.

Step 3 Check the **Define this policy setting** check box and specify the maximum password age, in days, that you want to allow.

Figure 4-3 Active Directory—Maximum Password Age



Note The **radius-with-expiry** command, formerly configured as part of tunnel-group remote-access configuration to perform the password age function, is deprecated. The **password-management** command, entered in tunnel-group general-attributes mode, replaces it.

Using Active Directory to Override an Account Disabled AAA Indicator

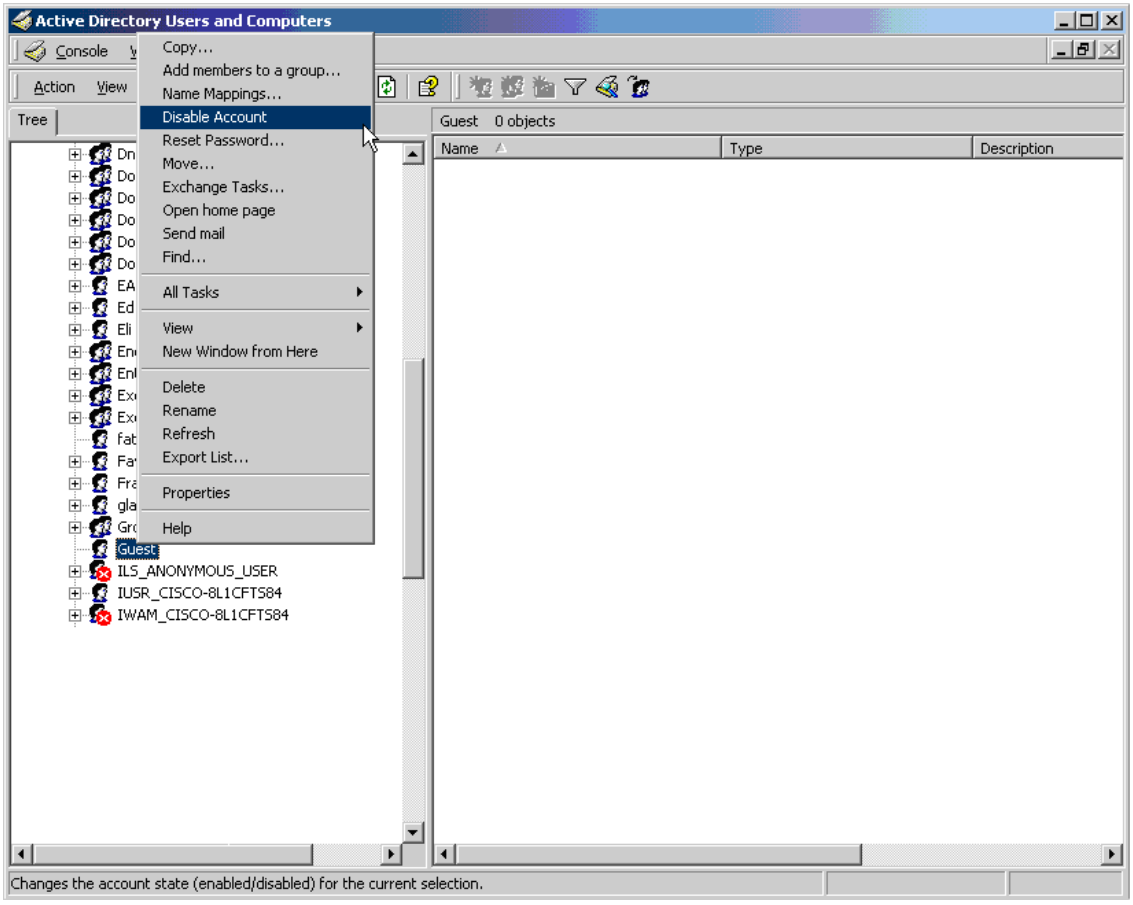
To override an account-disabled indication from a AAA server, use the **override-account-disable** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory.

Note Allowing override account-disabled is a potential security risk.

Step 1 Select Start > Programs > Administrative Tools > Active Directory Users and Computers.

Step 2 Right-click Username > Properties > Account and select Disable Account from the menu.

Figure 4-4 Active Directory—Override Account Disabled



The user should be able to log on successfully, even though a AAA server provides an account-disabled indicator.

Using Active Directory to Enforce Minimum Password Length

To enforce a minimum length for passwords, specify the **password-management** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory:

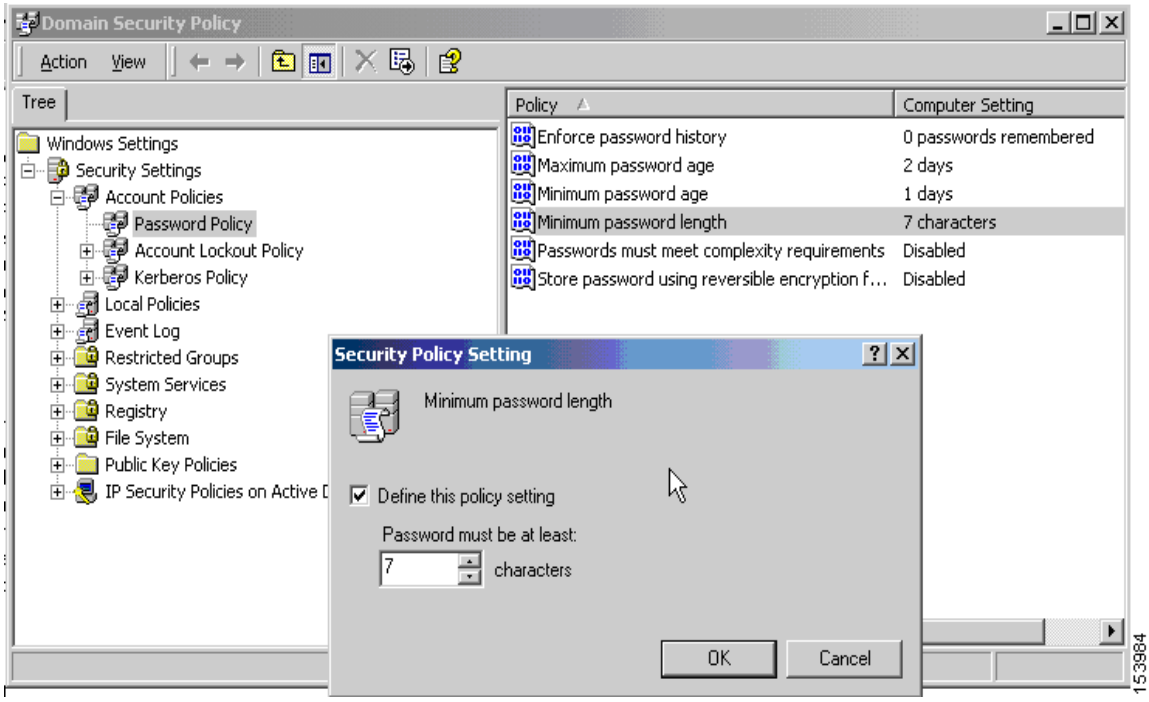
Step 1 Select Start > Programs > Administrative Tools > Domain Security Policy.

Step 2 Select Windows Settings > Security Settings > Account Policies > Password Policy.

Step 3 Double-click Minimum Password Length. The Security Policy Setting dialog box appears.

Step 4 Check the Define this policy setting check box and specify the minimum number of characters that the password must contain.

Figure 4-5 Active Directory—Minimum Password Length



Using Active Directory to Enforce Password Complexity

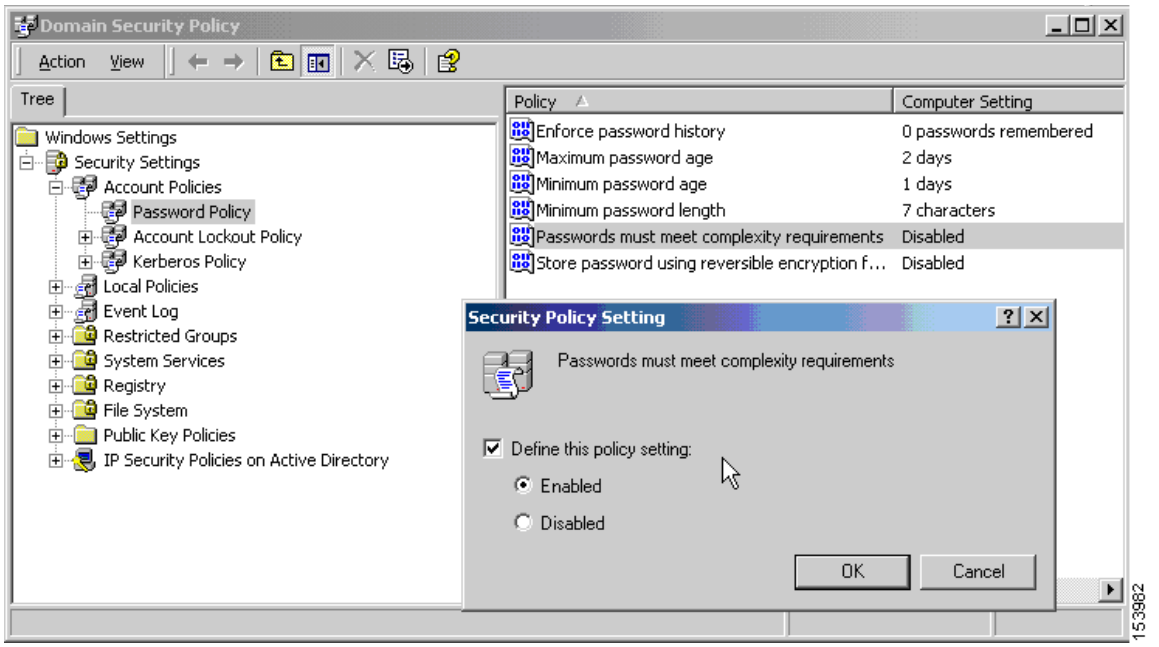
To enforce complex passwords—for example, to require that a password contain upper- and lowercase letters, numbers, and special characters—enter the **password-management** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory:

Step 1 Select Start > Programs > Administrative Tools > Domain Security Policy. Select Windows Settings > Security Settings > Account Policies > Password Policy.

Step 2 Double-click Password must meet complexity requirements to open the Security Policy Setting dialog box.

Step 3 Check the Define this policy setting check box and select **Enable**.

Figure 4-6 Active Directory—Enforce Password Complexity



Enforcing password complexity takes effect only when the user changes passwords; for example, when you have configured Enforce password change at next login or Password expires in *n* days. At login, the user receives a prompt to enter a new password, and the system will accept only a complex password.

Configuring the Connection Profile for RADIUS/SDI Message Support for the AnyConnect Client

This section describes procedures to ensure that the AnyConnect VPN client using RSA SecureID Software tokens can properly respond to user prompts delivered to the client through a RADIUS server proxying to an SDI server(s). This section contains the following topics:

- AnyConnect Client and RADIUS/SDI Server Interaction
- Configuring the Security Appliance to Support RADIUS/SDI Messages

Note If you have configured the double-authentication feature, SDI authentication is supported only on the primary authentication server.

AnyConnect Client and RADIUS/SDI Server Interaction

When a remote user connects to the ASA with the AnyConnect VPN client and attempts to authenticate using an RSA SecurID token, the ASA communicates with the RADIUS server, which in turn, communicates with the SDI server about the authentication.

During authentication, the RADIUS server presents access challenge messages to the ASA. Within these challenge messages are reply messages containing text from the SDI server. The message text is different when the ASA is communicating directly with an SDI server than when communicating through the RADIUS proxy. Therefore, in order to appear as a native SDI server to the AnyConnect client, the ASA must interpret the messages from the RADIUS server.

Also, because the SDI messages are configurable on the SDI server, the message text on the ASA must match (in whole or in part) the message text on the SDI server. Otherwise, the prompts displayed to the remote client user may not be appropriate for the action required during authentication. The AnyConnect client may fail to respond and authentication may fail.

"Configuring the Security Appliance to Support RADIUS/SDI Messages" section on page 4-35 describes how to configure the ASA to ensure successful authentication between the client and the SDI server.

Configuring the Security Appliance to Support RADIUS/SDI Messages

To configure the ASA to interpret SDI-specific RADIUS reply messages and prompt the AnyConnect user for the appropriate action, perform the following steps:

Step 1 Configure a connection profile (tunnel group) to forward RADIUS reply messages in a manner that simulates direct communication with an SDI server using the **proxy-auth sdi** command from tunnel-group webvpn configuration mode. Users authenticating to the SDI server must connect over this connection profile.

For example:

```
hostname(config)# tunnel-group sales webvpn attributes
hostname(tunnel-group-webvpn)# proxy-auth sdi
```

Step 2 Configure the RADIUS reply message text on the ASA to match (in whole or in part) the message text sent by the RADIUS server with the **proxy-auth_map sdi** command from tunnel-group webvpn configuration mode.

The default message text used by the ASA is the default message text used by Cisco Secure Access Control Server (ACS). If you are using Cisco Secure ACS, and it is using the default message text, you do not need to configure the message text on the ASA. Otherwise, use the **proxy-auth_map sdi** command to ensure the message text matches.

Table 4-3 shows the message code, the default RADIUS reply message text, and the function of each message. Because the security appliance searches for strings in the order that they appear in the table, you must ensure that the string you use for the message text is not a subset of another string.

For example, "new PIN" is a subset of the default message text for both new-pin-sup and next-ccode-and-reauth. If you configure new-pin-sup as "new PIN", when the security appliance receives "new PIN with the next card code" from the RADIUS server, it will match the text to the new-pin-sup code instead of the next-ccode-and-reauth code.

Table 4-3 SDI Op-codes, Default Message Text, and Message Function

Message Code	Default RADIUS Reply Message Text	Function
next-code	Enter Next PASSCODE	Indicates the user must enter the NEXT tokencode without the PIN.
new-pin-sup	Please remember your new PIN	Indicates the new system PIN has been supplied and displays that PIN for the user.

Message Code	Default RADIUS Reply Message Text	Function
new-pin-meth	Do you want to enter your own pin	Requests from the user which new PIN method to use to create a new PIN.
new-pin-req	Enter your new Alpha-Numerical PIN	Indicates a user-generated PIN and requests that the user enter the PIN.
new-pin-reenter	Reenter PIN:	Used internally by the ASA for user-supplied PIN confirmation. The client confirms the PIN without prompting the user.
new-pin-sys-ok	New PIN Accepted	Indicates the user-supplied PIN was accepted.
next-ccode-and-reauth	new PIN with the next card code	Follows a PIN operation and indicates the user must wait for the next tokencode and to enter both the new PIN and next tokencode to authenticate.
ready-for-sys-pin	ACCEPT A SYSTEM GENERATED PIN	Used internally by the ASA to indicate the user is ready for the system-generated PIN.

The following example enters aaa-server-host mode and changes the text for the RADIUS reply message new-pin-sup*:*

```
hostname(config)# aaa-server radius_sales host 10.10.10.1
hostname(config-aaa-server-host)# proxy-auth_map sdi new-pin-sup "This is your new PIN"
```

Group Policies

This section describes group policies and how to configure them. It includes the following topics:

- Default Group Policy, page 4-37
- Configuring Group Policies, page 4-39

A group policy is a set of user-oriented attribute/value pairs for IPsec connections that are stored either internally (locally) on the device or externally on a RADIUS server. The connection profile uses a group policy that sets terms for user connections after the tunnel is established. Group policies let you apply whole sets of attributes to a user or a group of users, rather than having to specify each attribute individually for each user.

Enter the **group-policy** commands in global configuration mode to assign a group policy to users or to modify a group policy for specific users.

The ASA includes a default group policy. In addition to the default group policy, which you can modify but not delete, you can create one or more group policies specific to your environment.

You can configure internal and external group policies. Internal groups are configured on the ASA's internal database. External groups are configured on an external authentication server, such as RADIUS. Group policies include the following attributes:

- Identity
- Server definitions
- Client firewall settings
- Tunneling protocols
- IPsec settings

- Hardware client settings
- Filters
- Client configuration settings
- Connection settings

Default Group Policy

The ASA supplies a default group policy. You can modify this default group policy, but you cannot delete it. A default group policy, named DfltGrpPolicy, always exists on the ASA, but this default group policy does not take effect unless you configure the ASA to use it. When you configure other group policies, any attribute that you do not explicitly specify takes its value from the default group policy. To view the default group policy, enter the following command:

```
ciscoasa(config)# show running-config all group-policy DfltGrpPolicy
hostname(config)#
```

To configure the default group policy, enter the following command:

```
ciscoasa(config)# group-policy DfltGrpPolicy internal
hostname(config)#
```

Note The default group policy is always internal. Despite the fact that the command syntax is ciscoasa(config)# group-policy DfltGrpPolicy {internal | external}, you cannot change its type to external.

To change any of the attributes of the default group policy, use the **group-policy attributes** command to enter attributes mode, then specify the commands to change whatever attributes that you want to modify:

```
ciscoasa(config)# group-policy DfltGrpPolicy attributes
```

Note The attributes mode applies only to internal group policies.

The default group policy, DfltGrpPolicy, that the ASA provides is as follows:

```
hostname# show run all group-policy DfltGrpPolicy
group-policy DfltGrpPolicy internal
group-policy DfltGrpPolicy attributes
 banner none
 wins-server none
 dns-server value 10.10.10.1.1
 dhcp-network-scope none
 vpn-access-hours none
 vpn-simultaneous-logins 3
 vpn-idle-timeout 30
 vpn-idle-timeout alert-interval 1
 vpn-session-timeout none
 vpn-session-timeout alert-interval 1
 vpn-filter none
 vpn-tunnel-protocol ikev1 ikev2 l2tp-ipsec ssl-client ssl-clientless
 password-storage disable
 ip-comp disable
 re-xauth disable
```

```
group-lock none
pfs disable
ipsec-udp disable
ipsec-udp-port 10000
split-tunnel-policy tunnelall
ipv6-split-tunnel-policy tunnelall
split-tunnel-network-list none
default-domain value cisco.com
split-dns none
split-tunnel-all-dns disable
intercept-dhcp 255.255.255.255 disable
secure-unit-authentication disable
user-authentication disable
user-authentication-idle-timeout 30
ip-phone-bypass disable
client-bypass-protocol disable
gateway-fqdn none
leap-bypass disable
nem disable
backup-servers keep-client-config
msie-proxy server none
msie-proxy method no-modify
msie-proxy except-list none
msie-proxy local-bypass disable
msie-proxy pac-url none
msie-proxy lockdown enable
vlan none
nac-settings none
address-pools none
ipv6-address-pools none
smartcard-removal-disconnect enable
scep-forwarding-url none
client-firewall none
client-access-rule none
webvpn
 url-list none
 filter none
 homepage none
 html-content-filter none
 port-forward name Application Access
 port-forward disable
 http-proxy disable
 sso-server none
 anyconnect ssl dtls enable
 anyconnect mtu 1406
 anyconnect firewall-rule client-interface private none
 anyconnect firewall-rule client-interface public none
 anyconnect keep-installer installed
 anyconnect ssl keepalive 20
 anyconnect ssl rekey time none
 anyconnect ssl rekey method none
 anyconnect dpd-interval client 30
 anyconnect dpd-interval gateway 30
 anyconnect ssl compression none
 anyconnect dtls compression lzs
 anyconnect modules none
 anyconnect profiles none
 anyconnect ask none
 customization none
 keep-alive-ignore 4
 http-comp gzip
 download-max-size 2147483647
 upload-max-size 2147483647
 post-max-size 2147483647
```

```
  user-storage none
  storage-objects value cookies,credentials
  storage-key none
  hidden-shares none
  smart-tunnel disable
  activex-relay enable
  unix-auth-uid 65534
  unix-auth-gid 65534
  file-entry enable
  file-browsing enable
  url-entry enable
  deny-message value Login was successful, but because certain criteria have not been met
or due to some specific group policy, you do not have permission to use any of the VPN
features. Contact your IT administrator for more information
  smart-tunnel auto-signon disable
  anyconnect ssl df-bit-ignore disable
  anyconnect routing-filtering-ignore disable
  smart-tunnel tunnel-policy tunnelall
  always-on-vpn profile-setting
```

You can modify the default group policy, and you can also create one or more group policies specific to your environment.

Configuring Group Policies

A group policy can apply to any kind of tunnel. In each case, if you do not explicitly define a parameter, the group takes the value from the default group policy.

You can perform these configuration tasks in both single context mode or multiple-context mode:

Note Multiple-context mode applies only to IKEv2 and IKEv1 site to site and does not apply to AnyConnect, Clientless SSL VPN, legacy Cisco VPN client, the Apple native VPN client, the Microsoft native VPN client, or cTCP for IKEv1 IPsec.

Configuring an External Group Policy

External group policies take their attribute values from the external server that you specify. For an external group policy, you must identify the AAA server group that the ASA can query for attributes and specify the password to use when retrieving attributes from the external AAA server group. If you are using an external authentication server, and if your external group-policy attributes exist in the same RADIUS server as the users that you plan to authenticate, you have to make sure that there is no name duplication between them.

Note External group names on the ASA refer to user names on the RADIUS server. In other words, if you configure external group X on the ASA, the RADIUS server sees the query as an authentication request for user X. So external groups are really just user accounts on the RADIUS server that have special meaning to the ASA. If your external group attributes exist in the same RADIUS server as the users that you plan to authenticate, there must be no name duplication between them.

The ASA supports user authorization on an external LDAP or RADIUS server. Before you configure the ASA to use an external server, you must configure the server with the correct ASA authorization attributes and, from a subset of these attributes, assign specific permissions to individual users. Follow

the instructions in Appendix 13, "Configuring an External Server for Authorization and Authentication" to configure your external server.

To configure an external group policy, perform the following steps specify a name and type for the group policy, along with the server-group name and a password:

```
hostname(config)# group-policy group_policy_name type server-group server_group_name
password server_password
hostname(config)#
```

Note For an external group policy, RADIUS is the only supported AAA server type.

For example, the following command creates an external group policy named ExtGroup that gets its attributes from an external RADIUS server named ExtRAD and specifies that the password to use when retrieving the attributes is newpassword:

```
hostname(config)# group-policy ExtGroup external server-group ExtRAD password newpassword
hostname(config)#
```

Note You can configure several vendor-specific attributes (VSAs), as described in Appendix 13, "Configuring an External Server for Authorization and Authentication". If a RADIUS server is configured to return the Class attribute (#25), the ASA uses that attribute to authenticate the Group Name. On the RADIUS server, the attribute must be formatted as: OU=*groupname*; where *groupname* is identical to the Group Name configured on the ASA—for example, OU=Finance.

Creating an Internal Group Policy

To configure an internal group policy, enter configuration mode, use the group-policy command, specify a name, and the **internal** type for the group policy:

```
hostname(config)# group-policy group_policy_name internal
hostname(config)#
```

For example, the following command creates the internal group policy named GroupPolicy1:

```
hostname(config)# group-policy GroupPolicy1 internal
hostname(config)#
```

Note You cannot change the name of a group policy after you create it.

You can conifgure the attributes of an internal group policy by copying the values of a preexisting group policy by appending the keyword **from** and specifying the name of the existing policy:

```
hostname(config)# group-policy group_policy_name internal from group_policy_name
ciscoasa(config-group-policy)#
```

For example, the following command creates the internal group policy named GroupPolicy2 by copying the attributes of GroupPolicy1:

```
hostname(config)# group-policy GroupPolicy2 internal from GroupPolicy1
ciscoasa(config-group-policy)#
```

Configuring General Internal Group Policy Attributes

Group Policy Name

The group policy name was chosen when the internal group policy was created. You cannot change the name of a group policy once it has been created. See Creating an Internal Group Policy, page 4-40 for more information.

Configuring the Group Policy Banner Message

Specify the banner, or welcome message, if any, that you want to display. The default is no banner. The message that you specify is displayed on remote clients when they connect. To specify a banner, enter the **banner** command in group-policy configuration mode. The banner text can be up to 510 characters long. Enter the "\n" sequence to insert a carriage return.

Note A carriage-return and line-feed included in the banner counts as two characters.

To delete a banner, enter the **no** form of this command. Be aware that using the **no** version of the command deletes all banners for the group policy.

A group policy can inherit this value from another group policy. To prevent inheriting a value, enter the **none** keyword instead of specifying a value for the banner string, as follows:

```
ciscoasa(config-group-policy)# banner {value banner_string | none}
```

The following example shows how to create a banner for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# banner value Welcome to Cisco Systems ASA 9.0.
```

Specifying Address Pools for Remote Access Connections

When remote access clients connect to the ASA, the ASA can assign the client an IPv4 or IPv6 address based on the group-policy specified for the connection.

You can specify a list of up to six local address pools to use for local address allocation. The order in which you specify the pools is significant. The ASA allocates addresses from these pools in the order in which the pools appear in this command.

Assigning an IPv4 Address Pool to an Internal Group Policy

Prerquisite

Create the IPv4 address pool. See Chapter 5, "Configuring IP Addresses for VPNs."

Detailed Steps

	Command	Purpose

Step 1	`group-policy value attributes`
	
	Example:
	`ciscoasa> en`
	`ciscoasa# config t`
	`ciscoasa(config)# group-policy FirstGroup attributes`
	`ciscoasa(config-group-policy)#`
		Enter group policy configuration mode.

Step 2	**address-pools value** *pool-name1 pool-name2 pool-name6*
	
	Example:
	`asa4(config-group-policy)# address-pools value ipv4-pool1 ipv4-pool2 ipv4-pool3`
	`asa4(config-group-policy)#`
		Assigns the address pool named ipv4-pool1, ipv4-pool2, and ipv4pool3 to the FirstGroup group policy.
		You are allowed to specify up to 6 address pools for group-policy.

Step 3	(Optional)
	no address-pools value pool-name1 pool-name2 pool-name6
	
	Example:
	`ciscoasa(config-group-policy)# no address-pools value ipv4-pool1 ipv4-pool2 ipv4-pool3`
	`ciscoasa(config-group-policy)#`
		Use the **no address-pools value pool-name** command to remove the address-pools from the goup policy configuration and returns the address pool setting to inherit the address pool information from other sources such as the DefltGroupPolicy.

Step 4	(Optional)
	address-pools none
	
	Example:
	`ciscoasa(config-group-policy)# address-pools none`
	`ciscoasa(config-group-policy)#`
		The **address-pools none** command disables this attribute from being inherited from other sources of policy, such as the DefltGrpPolicy:

Step 5	(Optional)
	no address-pools none
	
	Example:
	`ciscoasa(config-group-policy)# no address-pools none`
	`ciscoasa(config-group-policy)#`
		The **no address pools none** command removes the **address-pools none** command from the group policy, restoring the default value, which is to allow inheritance.

Assigning an IPv6 Address Pool to an Internal Group Policy

Prerquisite

Create the IPv6 address pool. See Chapter 5, “Configuring IP Addresses for VPNs.”

Detailed Steps

	Command	Purpose

Step 1	`group-policy value attributes`
	
	Example:
	`ciscoasa> en`
	`ciscoasa# config t`
	`ciscoasa(config)# group-policy FirstGroup attributes`
	`ciscoasa(config-group-policy)#`
		Enter group policy configuration mode.

Step 2	**ipv6-address-pools value** *pool-name1 pool-name2 pool-name6* **Example:** ciscoasa(config-group-policy)# **ipv6-address-pools value ipv6-pool1 ipv6-pool2 ipv6-pool3** ciscoasa(config-group-policy)#	Assigns the address pool named ipv6-pool to the FirstGroup group policy. You can assign up to six ipv6 address pools to a group policy. This example shows ipv6-pool1, ipv6-pool2, and ipv6-pool3 being assigned to the FirstGroup group policy.

Step 3	(Optional)
	no ipv6-address-pools value pool-name1 pool-name2 pool-name6
	
	Example:
	`ciscoasa(config-group-policy)# no ipv6-address-pools value ipv6-pool1 ipv6-pool2 ipv6-pool3`
	`ciscoasa(config-group-policy)#`
		Use the **no ipv6-address-pools value pool-name** command to remove the address-pools from the goup policy configuration and returns the address pool setting to inherit the address pool information from other sources such as the DfltGroupPolicy.

Step 4	(Optional)
	ipv6-address-pools none
	
	Example:
	`ciscoasa(config-group-policy)# ipv6-address-pools none`
	`ciscoasa(config-group-policy)#`
		The **ipv6-address-pools none** command disables this attribute from being inherited from other sources of policy, such as the DfltGrpPolicy:

Step 5	(Optional)
	no ipv6-address-pools none
	
	Example:
	`ciscoasa(config-group-policy)# no ipv6-address-pools none`
	`ciscoasa(config-group-policy)#`
		The **no ipv6-address pools none** command removes the **ipv6-address-pools none** command from the group policy, restoring the default value, which is to allow inheritance.

Specifying the Tunneling Protocol for the Group Policy

Specify the VPN tunnel type for this group policy by entering the **vpn-tunnel-protocol {ikev1 | ikev2 | l2tp-ipsec | ssl-client | ssl-clientless}** command from group-policy configuration mode.

The default value is to inherit the attributes of the Default Group Policy. To remove the attribute from the running configuration, enter the **no** form of this command.

The parameter values for this command follow:

- **ikev1**—Negotiates an IPsec IKEv1 tunnel between two peers (the Cisco VPN Client or another secure gateway). Creates security associations that govern authentication, encryption, encapsulation, and key management.
- **ikev2**—Negotiates an IPsec IKEv2 tunnel between two peers (the AnyConnect Secure Mobility Client or another secure gateway). Creates security associations that govern authentication, encryption, encapsulation, and key management.
- **l2tp-ipsec**—Negotiates an IPsec tunnel for an L2TP connection.
- **ssl-client**—Negotiates an SSL tunnel using TLS or DTLS with the AnyConnect Secure Mobility Client.
- **ssl-clientless**—Provides VPN services to remote users via an HTTPS-enabled web browser, and does not require a client.

Enter this command to configure one or more tunneling modes. You must configure at least one tunneling mode for users to connect over a VPN tunnel.

The following example shows how to configure the IPsec IKEv1 tunneling mode for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-tunnel-protocol ikev1
ciscoasa(config-group-policy)#
```

Specifying a VLAN for Remote Access or Applying a Unified Access Control Rule to the Group Policy

Filters consist of rules that determine whether to allow or reject tunneled data packets coming through the ASA, based on criteria such as source address, destination address, and protocol. You can specify an IPv4 or IPv6 unified access control list for your group policy or allow it to inherit the ACLs specified in the Default Group Policy.

Choose one of the following options to specify an egress VLAN (also called “VLAN mapping”) for remote access or specify an ACL to filter the traffic:

- Enter the following command in group-policy configuration mode to specify the egress VLAN for remote access VPN sessions assigned to this group policy or to a group policy that inherits this group policy:

  ```
  ciscoasa(config-group-policy)# [no] vlan {vlan_id |none}
  ```

  no vlan removes the *vlan_id* from the group policy. The group policy inherits the vlan value from the default group policy.

  none removes the *vlan_id* from the group policy and disables VLAN mapping for this group policy. The group policy does not inherit the vlan value from the default group policy.

  vlan_id is the number of the VLAN, in decimal format, to assign to remote access VPN sessions that use this group policy. The VLAN must be configured on this ASA per the instructions in the “Configuring VLAN Subinterfaces and 802.1Q Trunking” section on page 11-36 in the general operations configuration guide.

  Note The egress VLAN feature works for HTTP connections, but not for FTP and CIFS.

- Specify the name of the access control rule (ACL) to apply to VPN session, using the **vpn-filter** command in group policy mode. You can specify an IPv4 or IPv6 ACL using the vpn-filter command.

Note In previous releases, the deprecated ipv6-vpn-filter command could be used to specify an IPv6 ACL if there were no IPv6 entries specified by vpn-filter. As of ASA 9.1(4), ipv6-vpn-filter has been disabled and IPv6 ACL entries must be specified using the vpn-filter command. If ipv6-vpn-filter is set, the VPN connection will be terminated.

Note You can also configure this attribute in username mode, in which case the value configured under username supersedes the group-policy value.

```
ciscoasa(config-group-policy)# vpn-filter {value ACL name | none}
ciscoasa(config-group-policy)#
```

You configure ACLs to permit or deny various types of traffic for this group policy. You then enter the **vpn-filter** command to apply those ACLs.

To remove the ACL, including a null value created by entering the **vpn-filter none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from another group policy.

A group policy can inherit this value from another group policy. To prevent inheriting a value, enter the **none** keyword instead of specifying an ACL name. The **none** keyword indicates that there is no ACL and sets a null value, thereby disallowing an ACL.

The following example shows how to set a filter that invokes an ACL named acl_vpn for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-filter acl_vpn
ciscoasa(config-group-policy)#
```

A **vpn-filter** command is applied to post-decrypted traffic after it exits a tunnel and pre-encrypted traffic before it enters a tunnel. An ACL that is used for a vpn-filter should NOT also be used for an interface access-group. When a **vpn-filter** command is applied to a group policy that governs Remote Access VPN client connections, the ACL should be configured with the client assigned IP addresses in the **src_ip** position of the ACL and the local network in the **dest_ip** position of the ACL.

When a **vpn-filter** command is applied to a group-policy that governs a LAN to LAN VPN connection, the ACL should be configured with the remote network in the **src_ip** position of the ACL and the local network in the **dest_ip** position of the ACL.

Caution should be used when constructing the ACLs for use with the vpn-filter feature. The ACLs are constructed with the post-decrypted traffic in mind. However, ACLs are also applied to the traffic in the opposite direction. For this pre-encrypted traffic that is destined for the tunnel, the ACLs are constructed with the **src_ip** and **dest_ip** positions swapped.

In the following example, the vpn-filter is used with a Remote Access VPN client.
This example assumes that the client assigned IP address is 10.10.10.1/24 and the local network is 192.168.1.0/24.

The following ACE will allow the Remote Access VPN client to telnet to the local network:

```
ciscoasa(config-group-policy)# access-list vpnfilt-ra permit 10.10.10.1 255.255.255.255
192.168.1.0 255.255.255.0 eq 23
```

The following ACE will allow the local network to telnet to the Remote Access client:

```
ciscoasa(config-group-policy)# access-list vpnfilt-ra permit 10.10.10.1 255.255.255.255 eq
23 192.168.1.0 255.255.255.0
```

Note The ACE **access-list vpnfilt-ra permit 10.10.10.1 255.255.255.255 192.168.1.0 255.255.255.0 eq 23** allows the local network to initiate a connection to the Remote Access client on any TCP port if it uses a source port of 23. The ACE **access-list vpnfilt-ra permit 10.10.10.1 255.255.255.255 eq 23 192.168.1.0 255.255.255.0** allows the Remote Access client to initiate a connection to the local network on any TCP port if it uses a source port of 23.

In the next example, the vpn-filter is used with a LAN to LAN VPN connection. This example assumes that the remote network is 10.0.0.0/24 and the local network is 192.168.1.0/24.
The following ACE will allow remote network to telnet to the local network:

```
ciscoasa(config-group-policy)# access-list vpnfilt-l2l permit 10.0.0.0 255.255.255.0
192.168.1.0 255.255.255.0 eq 23
```

The following ACE will allow the local network to telnet to the remote network:

```
ciscoasa(config-group-policy)# access-list vpnfilt-l2l permit 10.0.0.0 255.255.255.0 eq 23
192.168.1.0 255.255.255.0
```

Note The ACE **access-list vpnfilt-l2l permit 10.0.0.0 255.255.255.0 192.168.1.0 255.255.255.0 eq 23** allows the local network to initiate a connection to the remote network on any TCP port if it uses a source port of 23. The ACE **access-list vpnfilt-l2l permit 10.0.0.0 255.255.255.0 eq 23 192.168.1.0 255.255.255.0** allows the remote network to initiate a connection to the local network on any TCP port if it uses a source port of 23.

Specifying a NAC Policy for a Group Policy

This command selects the name of a Network Admission Control policy to apply to this group policy. You can assign an optional NAC policy to each group policy. The default value is --None--.

Prerquisite

Create a NAC policy. See Configuring Network Admission Control, page 7-1.

Detailed Steps

	Command	Purpose

Step 1	`group-policy value attributes`
	
	Example:
	`ciscoasa> en`
	`ciscoasa# config t`
	`ciscoasa(config)# group-policy FirstGroup attributes`
	`ciscoasa(config-group-policy)#`
		Enter group policy configuration mode.

Step 2	**nac-settings value** *nac-policy-name* **Example:** ciscoasa(config-group-policy)# **nac-settings value nac-policy-1** ciscoasa(config-group-policy)#	Assigns the NAC policy named nac-policy-1 to the FirstGroup group policy.

Specifying VPN Access Hours for a Group Policy

Prerequisites

Create a time range. See the "Configuring Time Ranges" section on page 20-15 in the general operations configuration guide.

Detailed Steps

	Command	Purpose

Step 1	`group-policy value attributes`
	
	Example:
	`ciscoasa> en`
	`ciscoasa# config t`
	`ciscoasa(config)# group-policy FirstGroup attributes`
	`ciscoasa(config-group-policy)#`
		Enter group policy configuration mode.

Step 2	`ciscoasa(config-group-policy)# vpn-access-hours value {time-range-name | none}`
	
	Example:
	`ciscoasa(config-group-policy)# vpn-access-hours value business-hours`
	ciscoasa(config-group-policy)#
		You can set the VPN access hours by associating a configured time-range policy with a group policy using the **vpn-access-hours** command in group-policy configuration mode.
		This command assigns a VPN access time range named business-hours to the group policy named FirstGroup.
		A group policy can inherit a time-range value from a default or specified group policy. To prevent this inheritance, enter the **none** keyword instead of the name of a time-range in this command. This keyword sets VPN access hours to a null value, which allows no time-range policy.

Specifying Simultaneous VPN Logins for a Group Policy

Specify the number of simultaneous logins allowed for any user, using the **vpn-simultaneous-logins** command in group-policy configuration mode.

```
ciscoasa(config-group-policy)# vpn-simultaneous-logins integer
```

The default value is 3. The range is an integer in the range 0 through 2147483647. A group policy can inherit this value from another group policy. Enter 0 to disable login and prevent user access. The following example shows how to allow a maximum of 4 simultaneous logins for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-simultaneous-logins 4
ciscoasa(config-group-policy)#
```

Note While the maximum limit for the number of simultaneous logins is very large, allowing several simultaneous logins could compromise security and affect performance.

Stale AnyConnect, IPsec Client, or Clientless sessions (sessions that are terminated abnormally) might remain in the session database, even though a "new" session has been established with the same username.

If the value of vpn-simultaneous-logins is 1, and the same user logs in again after an abnormal termination, then the stale session is removed from the database and the new session is established. If, however, the existing session is still an active connection and the same user logs in again, perhaps from another PC, the first session is logged off and removed from the database, and the new session is established.

If the number of simultaneous logins is a value greater than 1, then, when you have reached that maximum number and try to log in again, the session with the longest idle time is logged off. If all current sessions have been idle an equally long time, then the oldest session is logged off. This action frees up a session and allows the new login.

Restricting Access to a Specific Connection Profile

Specify whether to restrict remote users to access only through the connection profile, using the **group-lock** command in group-policy configuration mode.

```
ciscoasa(config-group-policy)# group-lock {value tunnel-grp-name | none}
ciscoasa(config-group-policy)# no group-lock
ciscoasa(config-group-policy)#
```

The *tunnel-grp-name* variable specifies the name of an existing connection profile that the ASA requires for the user to connect. Group-lock restricts users by checking if the group configured in the VPN client is the same as the connection profile to which the user is assigned. If it is not, the ASA prevents the user from connecting. If you do not configure group-lock, the ASA authenticates users without regard to the assigned group. Group locking is disabled by default.

To remove the **group-lock** attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value from another group policy.

To disable group-lock, enter the **group-lock** command with the **none** keyword. The none keyword sets group-lock to a null value, thereby allowing no group-lock restriction. It also prevents inheriting a group-lock value from a default or specified group policy

Specifying the Maximum VPN Connection Time in a Group Policy

Step 1 Configure a maximum amount of time for VPN connections, using the **vpn-session-timeout** command in group-policy configuration mode or in username configuration mode.

```
ciscoasa(config-group-policy)# vpn-session-timeout {minutes | none}
ciscoasa(config-group-policy)#
```

The minimum time is 1 minute, and the maximum time is 35791394 minutes. There is no default value. At the end of this period of time, the ASA terminates the connection.

A group policy can inherit this value from another group policy. To prevent inheriting a value, enter the **none** keyword instead of specifying a number of minutes with this command. Specifying the **none** keyword permits an unlimited session timeout period and sets session timeout with a null value, which disallows a session timeout.

The following example shows how to set a VPN session timeout of 180 minutes for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-session-timeout 180
```

```
ciscoasa(config-group-policy)#
```

Step 2 Configure the the time at which a session-timeout alert message is displayed to the user using the **vpn-session-timeout alert-interval** {minutes | none} command. This alert message tells users how many minutes left they have until their VPN session is automatically disconnected.

The following example shows how to set the **vpn-session-timeout alert-interval** so that users will be notified 20 minutes before their VPN session is disconnected. You can specify a range of 1-30 minutes.

```
hostname(config-webvpn)# vpn-session-timeout alert-interval 20
```

The *none* parameter indicates that users will not receive an alert.

Use the no form of the command to indicate that the VPN session timeout alert-interval attribute will be inherited from the Default Group Policy:

```
no vpn-session-timeout alert-interval
```

Specifying a VPN Session Idle Timeout for a Group Policy

Step 1 Configure the user timeout period by entering the **vpn-idle-timeout** command in group-policy configuration mode or in username configuration mode:

```
ciscoasa(config-group-policy)# vpn-idle-timeout {minutes | none}
ciscoasa(config-group-policy)#
```

AnyConnect (SSL IPsec/IKEv2): Use the global WebVPN default-idle-timeout value (seconds) from the command: **ciscoasa(config-webvpn)# default-idle-timeout**

The range for this value in the WebVPN **default-idle-timeout** command is 60-86400 seconds; the default Global WebVPN Idle timeout in seconds -- default is 1800 seconds (30 min).

Note A non-zero idle timeout value is required by ASA for all AnyConnect connections.

For a WebVPN user, the **default-idle-timeout** value is enforced only if vpn-idle-timeout none is set in the group policy/username attribute.

Site-to-Site (IKEv1, IKEv2) and IKEv1 remote-access: Disable timeout and allow for an unlimited idle period.

The following example shows how to set a VPN idle timeout of 15 minutes for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-idle-timeout 15
ciscoasa(config-group-policy)#
```

Step 2 Configure the the time at which an idle-timeout alert message is displayed to the user using the **vpn-idle-timeout alert-interval {minutes | none}** command. This alert message tells users how many minutes left they have until their VPN session is disconnected due to inactivity.

The following example shows how to set vpn-idle-timeout alert-interval so that users will be notified 20 minutes before their VPN session is disconnected due to inactivity. You can specify a range of 1-30 minutes.

```
hostname(config-webvpn)# vpn-idle-timeout alert-interval 20
```

The *none* parameter indicates that users will not receive an alert.

Use the no form of the command to indicate that the VPN idle timeout alert-interval attribute will be inherited from the Default Group Policy:

```
no vpn-idle-timeout alert-interval
```

Configuring WINS and DNS Servers for a Group Policy

You can specify primary and secondary WINS servers and DNS servers. The default value in each case is none. To specify these servers, perform the following steps:

Step 1 Specify the primary and secondary WINS servers:

```
hostname(config-group-policy)# wins-server value {ip_address [ip_address] | none}
ciscoasa(config-group-policy)#
```

The first IP address specified is that of the primary WINS server. The second (optional) IP address is that of the secondary WINS server. Specifying the **none** keyword instead of an IP address sets WINS servers to a null value, which allows no WINS servers and prevents inheriting a value from a default or specified group policy.

Every time that you enter the **wins-server** command, you overwrite the existing setting. For example, if you configure WINS server x.x.x.x and then configure WINS server y.y.y.y, the second command overwrites the first, and y.y.y.y becomes the sole WINS server. The same is true for multiple servers. To add a WINS server rather than overwrite previously configured servers, include the IP addresses of all WINS servers when you enter this command.

The following example shows how to configure WINS servers with the IP addresses 10.10.10.15 and 10.10.10.30 for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# wins-server value 10.10.10.15 10.10.10.30
ciscoasa(config-group-policy)#
```

Step 2 Specify the primary and secondary DNS servers:

```
hostname(config-group-policy)# dns-server value {ip_address [ip_address] | none}
ciscoasa(config-group-policy)#
```

The first IP address specified is that of the primary DNS server. The second (optional) IP address is that of the secondary DNS server. Specifying the **none** keyword instead of an IP address sets DNS servers to a null value, which allows no DNS servers and prevents inheriting a value from a default or specified group policy. You can specify up to four DNS server addresses: Up to two IPv4 addresses and two IPv6 addreses.

Every time that you enter the **dns-server** command you overwrite the existing setting. For example, if you configure DNS server x.x.x.x and then configure DNS server y.y.y.y, the second command overwrites the first, and y.y.y.y becomes the sole DNS server. The same is true for multiple servers. To add a DNS server rather than overwrite previously configured servers, include the IP addresses of all DNS servers when you enter this command.

The following example shows how to configure DNS servers with the IP addresses 10.10.10.15, 10.10.10.30, 2001:DB8::1, and 2001:DB8::2 for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
```

```
ciscoasa(config-group-policy)# dns-server value 10.10.10.15 10.10.10.30 2001:DB8::1
2001:DB8::2
ciscoasa(config-group-policy)#
```

Step 3 If there is no default domain name specified in the **DeafultDNS DNS server group,** you must specify a default domain. Use the domain name and top level domain for example, **example.com.**

```
asa4(config)# group-policy FirstGroup attributes
asa4(config-group-policy)# default-domain value example.com
asa4(config-group-policy)#
```

Step 4 Configure the DHCP network scope:

```
ciscoasa(config-group-policy)# dhcp-network-scope {ip_address | none}
ciscoasa(config-group-policy)#
```

DHCP scope specifies the range of IP addresses (that is, a subnetwork) that the ASA DHCP server should use to assign addresses to users of this group policy.

The following example shows how to set an IP subnetwork of 10.10.85.0 (specifying the address range of 10.10.85.0 through 10.10.85.255) for the group policy named First Group:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# dhcp-network-scope 10.10.85.0
```

Configuring Split-Tunneling for AnyConnect Traffic

Split tunneling directs some AnyConnect network traffic through the VPN tunnel (encrypted) and other network traffic outside the VPN tunnel (unencrypted or “in the clear”).

Split tunneling is configured by creating a split tunneling policy, configuring an access control list for that policy, and adding the split tunnel policy to a group policy. When the group policy is sent to the client, that client will use the ACLs in the split tunneling policy to decide where to direct network traffic.

When you create access lists:

- You can specify both IPv4 and IPv6 addresses in an access control list.
- If you use a standard ACL, only one address or network is used.
- If you use extended ACLs, the source network is the split-tunneling network. The destination network is ignored.
- Access lists configured as any or with an address of 0.0.0.0/0.0.0.0 or ::/0 will not be sent to the client. To send all traffic over the tunnel, specify “tunnelall” when creating the split-tunnel-policy.
- Address 0.0.0.0/255.255.255.255 or ::/128 will be sent to the client only when the split-tunnel policy is **excludespecified**. This configuration tells the client not to tunnel traffic destined for any local subnets.
- AnyConnect passes traffic to all sites specified in the split tunneling policy, **and** to all sites that fall within the same subnet as the IP address assigned by the ASA. For example, if the IP address assigned by the ASA is 10.1.1.1 with a mask of 255.0.0.0, the endpoint device passes all traffic destined to 10.0.0.0/8, regardless of the split tunneling policy. Therefore, use a netmask for the assigned IP address that properly references the expected local subnet.

You can also specify a list of domains to direct split tunnel traffic. The client directs traffic to the domains in the split-dns list to the VPN, and all other traffic is in the clear.

Prerequisites

- You must create an access list with ACLs and ACEs.
- If you create a split tunnel policy for IPv4 networks and another for IPv6 networks, then the network list you specify in the split-tunnel-network-list command is used for both protocols. So, the network list should contain access control entries (ACEs) for both IPv4 and IPv6 traffic.

Set the Split-Tunneling Policy

Set the rules for tunneling traffic by specifying the split-tunneling policy for IPv4 traffic:

```
ciscoasa(config-group-policy)# split-tunnel-policy {tunnelall | tunnelspecified |
excludespecified}
ciscoasa(config-group-policy)# no split-tunnel-policy
```

Set the rules for tunneling traffic by specifying the split-tunneling policy for IPv6 traffic:

```
ciscoasa(config-group-policy)# ipv6-split-tunnel-policy {tunnelall | tunnelspecified |
excludespecified}
ciscoasa(config-group-policy)# no ipv6-split-tunnel-policy
```

The policies options are:

- **tunnelspecified**— Tunnels all traffic to or from the networks specified in the Network List through the tunnel. Data to all other addresses travels in the clear and is routed by the remote user's Internet service provider.

  For versions of ASA 9.1.4 and higher, when you specify an include list, you can also specify an exclude list for a subnet inside the include range. Addresses in the excluded subnet will not be tunneled, and the rest of the include list will be. The networks in the exclusion list will not be sent over the tunnel. The exclusion list is specified using deny entries, and the inclusion list is specified using permit entries.

Note Networks in the exclusion list that are not a subset of the include list will be ignored by the client.

- **excludespecified** — Does not tunnel traffic to or from the networks specified in the Network List. Traffic from or to all other addresses is tunneled. The VPN client profile that is active on the client must have Local LAN Access enabled.
- **tunnelall** —Specifies that all traffic goes through the tunnel. This policy disables split tunneling. Remote users have access to the corporate network, but they do not have access to local networks. This is the default option.

Note Split tunneling is a traffic management feature, not a security feature. For optimum security, we recommend that you do not enable split tunneling.

The following example shows how to set a split tunneling policy of tunneling only specified networks for the group policy named FirstGroup for IPv4 and IPv6:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-tunnel-policy tunnelspecified

ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# ipv6-split-tunnel-policy tunnelspecified
```

Specify a Network List for Split-Tunneling

In split tunneling, network lists determine what network traffic travels across the tunnel. AnyConnect makes split tunneling decisions on the basis of a network list, which is an ACL.

Procedure

```
ciscoasa(config-group-policy)# split-tunnel-network-list {value access-list_name | none}
ciscoasa(config-group-policy)# no split-tunnel-network-list value [access-list_name]
```

- **value** *access-list name* — identifies an ACL that enumerates the networks to tunnel or not tunnel. The ACL can be a unified ACL with ACEs that specify both IPv4 and IPv6 addresses.
- **none** — indicates that there is no network list for split tunneling; the ASA tunnels all traffic. Specifying the **none** keyword sets a split tunneling network list with a null value, thereby disallowing split tunneling. It also prevents inheriting a default split tunneling network list from a default or specified group policy.

To delete a network list, enter the **no** form of this command. To delete all split tunneling network lists, enter the **no split-tunnel-network-list** command without arguments. This command deletes all configured network lists, including a null list if you created one by entering the **none** keyword.

When there are no split tunneling network lists, users inherit any network lists that exist in the default or specified group policy. To prevent users from inheriting such network lists, enter the **split-tunnel-network-list none** command.

Example

The following example shows how to create a network list named FirstList, and add it to the group policy named FirstGroup. FistList is an exclusion list and an inclusion list that is a subnet of the exclusion list:

```
ciscoasa(config)# split-tunnel-policy tunnelspecified
ciscoasa(config)# access-list FirstList deny ip 10.10.10.0 255.255.255.0 any
ciscoasa(config)# access-list FirstList permit ip 10.0.0.0 255.0.0.0 any

ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-tunnel-network-list value FirstList
```

The following example shows how to create a network list named v6, and add the v6 split tunnel policy to the group policy named GroupPolicy_ipv6-ikev2. v6 is an exclusion list and an inclusion list that is a subnet of the exclusion list:

```
ciscoasa(config)# access-list v6 extended permit ip fd90:5000::/32 any6
ciscoasa(config)# access-list v6 extended deny ip fd90:5000:3000:2880::/64 any6

ciscoasa(config)# group-policy GroupPolicy_ipv6-ikev2 internal
ciscoasa(config)# group-policy GroupPolicy_ipv6-ikev2 attributes
ciscoasa(config-group-policy)# vpn-tunnel-protocol ikev2 ssl-client
ciscoasa(config-group-policy)# ipv6-split-tunnel-policy tunnelspecified
ciscoasa(config-group-policy)# split-tunnel-network-list value v6
```

Verify the Split Tunnel Configuration

Run the **show runn group-policy attributes** command to verify your configuration. This example shows that the administrator has set both an IPv4 and IPv6 network policy and used the network list (unified ACL), **FirstList** for both policies.

```
ciscoasa(config-group-policy)# show runn group-policy FirstGroup attributes
group-policy FirstGroup attributes
 split-tunnel-policy tunnelspecified
 ipv6-split-tunnel-policy tunnelspecified
 split-tunnel-network-list value FirstList
```

Configure Domain Attributes for Split Tunneling

You can specify a default domain name or a list of domains to be resolved through the split tunnel, which we refere to as split DNS.

AnyConnect 3.1 supports true split DNS functionality for Windows and Mac OS X platforms. If the group policy on the security appliance enables split-include tunneling, and if it specifies the DNS names to be tunneled, AnyConnect tunnels any DNS queries that match those names to the private DNS server. True split DNS allows tunnel access to only DNS requests that match the domains pushed to the client by the ASA. These requests are not sent in the clear. On the other hand, if the DNS requests do not match the domains pushed down by the ASA, AnyConnect lets the DNS resolver on the client operating system submit the host name in the clear for DNS resolution.

Note Split DNS supports standard and update queries (including A, AAAA, NS, TXT, MX, SOA, ANY, SRV, PTR, and CNAME). PTR queries matching any of the tunneled networks are allowed through the tunnel.

For Mac OS X, AnyConnect can use true split-DNS for a certain IP protocol only if one of the following conditions is met:

- Split-DNS is configured for one IP protocol (such as IPv4), and Client Bypass Protocol is configured for the other IP protocol (such as IPv6) in the group policy (with no address pool configured for the latter IP protocol).
- Split-DNS is configured for both IP protocols.

Define a Default Domain Name

The ASA passes the default domain name to the AnyConnect client. The client appends the domain name to DNS queries that omit the domain field. This domain name applies only to tunneled packets.When there are no default domain names, users inherit the default domain name in the default group policy.

To specify the default domain name for users of the group policy, enter the **default-domain** command in group-policy configuration mode. To delete a domain name, enter the **no** form of this command.

```
ciscoasa(config-group-policy)# default-domain {value domain-name | none}
ciscoasa(config-group-policy)# no default-domain [domain-name]
```

The **value** *domain-name* parameter identifies the default domain name for the group. To specify that there is no default domain name, enter the **none** keyword. This command sets a default domain name with a null value, which disallows a default domain name and prevents inheriting a default domain name from a default or specified group policy.

To delete all default domain names, enter the **no default-domain** command without arguments. This command deletes all configured default domain names, including a null list if you created one by entering the **default-domain** command with the **none** keyword. The **no** form allows inheriting a domain name.

The following example shows how to set a default domain name of FirstDomain for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# default-domain value FirstDomain
```

Define a List of Domains for Split Tunneling

Enter a list of domains to be resolved through the split tunnel, in addition to the default domain. Enter the **split-dns** command in group-policy configuration mode. To delete a list, enter the **no** form of this command.

When there are no split tunneling domain lists, users inherit any that exist in the default group policy. To prevent users from inheriting such split tunneling domain lists, enter the **split-dns** command with the **none** keyword.

To delete all split tunneling domain lists, enter the **no split-dns** command without arguments. This deletes all configured split tunneling domain lists, including a null list created by issuing the **split-dns** command with the **none** keyword.

The parameter **value** *domain-name* provides a domain name that the ASA resolves through the split tunnel. The **none** keyword indicates that there is no split DNS list. It also sets a split DNS list with a null value, thereby disallowing a split DNS list, and prevents inheriting a split DNS list from a default or specified group policy. The syntax of the command is as follows:

```
ciscoasa(config-group-policy)# split-dns {value domain-name1 [domain-name2...
domain-nameN] | none}
ciscoasa(config-group-policy)# no split-dns [domain-name domain-name2 domain-nameN]
```

Enter a single space to separate each entry in the list of domains. There is no limit on the number of entries, but the entire string can be no longer than 255 characters. You can use only alphanumeric characters, hyphens (-), and periods (.). If the default domain name is to be resolved through the tunnel, you must explicitly include that name in this list.

The following example shows how to configure the domains Domain1, Domain2, Domain3, and Domain4 to be resolved through split tunneling for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-dns value Domain1 Domain2 Domain3 Domain4
```

Note When configuring split DNS, ensure the private DNS servers specified do not overlap with the DNS servers configured for the client platform. If they do, name resolution does not function properly and queries may be dropped.

Configuring DHCP Intercept for Windows XP and Split Tunneling

A Microsoft XP anomaly results in the corruption of domain names if split tunnel options exceed 255 bytes. To avoid this problem, the ASA limits the number of routes it sends to 27 to 40 routes, with the number of routes dependent on the classes of the routes.

DHCP Intercept lets Microsoft Windows XP clients use split-tunneling with the ASA. The ASA replies directly to the Microsoft Windows XP client DHCP Inform message, providing that client with the subnet mask, domain name, and classless static routes for the tunnel IP address. For Windows clients prior to Windows XP, DHCP Intercept provides the domain name and subnet mask. This is useful in environments in which using a DHCP server is not advantageous.

The **intercept-dhcp** command enables or disables DHCP intercept.

```
ciscoasa(config-group-policy)# intercept-dhcp netmask {enable | disable}
ciscoasa(config-group-policy)#
```

The *netmask* variable provides the subnet mask for the tunnel IP address. The **no** form of this command removes the DHCP intercept from the configuration:

```
[no] intercept-dhcp
```

The following example shows how to set DHCP Intercepts for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# intercept-dhcp enable
```

Setting Up a Split Exclusion Policy for Web Security

Information about Cloud Web Security

The AnyConnect Web Security module is an endpoint component that routes HTTP traffic to a Cisco Cloud Web Security scanning proxy where Cisco Cloud Web Security evaluates it. Cisco Cloud Web Security deconstructs the elements of a Web page so that it can analyze each element simultaneously. It blocks potentially harmful content and allows benign content to come through.

With many Cisco Cloud Web Security scanning proxies spread around the world, users taking advantage of AnyConnect Web Security are able to route their traffic to the Cisco Cloud Web Security scanning proxy with the fastest response time to minimize latency.

When a user has established a VPN session, all network traffic is sent through the VPN tunnel. However, when AnyConnect users are using web security, the HTTP traffic originating at the endpoint needs to be excluded from the tunnel and sent directly to the Cloud Web Security scanning proxy.

To set up the split tunnel exclusions for traffic meant for the Cloud Web Security scanning proxy, use the **Set up split exclusion for Web Security** button in a group policy.

Prerequisites

- You need to have access to the ASA using ASDM. This procedure cannot be performed using the command line interface.
- Web security needs to be configured for use with the AnyConnect client. See Configuring Web Security in the *AnyConnect Secure Mobility Client Administrator Guide*.
- You have created a Group Policy and assigned it a Connection Profile for AnyConnect clients configured with Web Security.

Detailed Steps

Step 1 Start an ASDM session for the head end you want to configure and select **Remote Access VPN > Configuration > Group Policies**.

Step 2 Select the Group Policy you want to configure and click **Edit**.

Step 3 Select **Advanced > Split Tunneling.**

Step 4 Click **Set up split exclusion for Web Security.**

Step 5 Enter a new, or select an existing, ACL used for Web Security split exclusion. ASDM will set up the ACL for use in the network list.

Step 6 Click **Create Access List** for a new list or **Update Access List** for an existing list.

Step 7 Click **OK**.

What to do next

When additional scanning proxies are added, update the unified ACL you created in this procedure with new information.

Configuring Browser Proxy Settings for use with Remote Access Clients

Follow these steps to configure the proxy server parameters for a client.

Step 1 Configure a browser proxy server and port for a client device by entering the **msie-proxy server** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# msie-proxy server {value server[:port] | none}
ciscoasa(config-group-policy)#
```

The default value is **none**. To remove the attribute from the configuration, use the **no** form of the command.

```
ciscoasa(config-group-policy)# no msie-proxy server
ciscoasa(config-group-policy)#
```

The line containing the proxy server IP address or hostname and the port number must be less than 100 characters long.

The following example shows how to configure the IP address 192.168.10.1 as a browser proxy server, using port 880, for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy server value 192.168.21.1:880
ciscoasa(config-group-policy)#
```

Step 2 Configure the browser proxy actions ("methods") for a client device by entering the **msie-proxy method** command in group-policy configuration mode.

```
ciscoasa(config-group-policy)# msie-proxy method [auto-detect | no-modify | no-proxy |
use-server]
ciscoasa(config-group-policy)#
```

The default value is **use-server**. To remove the attribute from the configuration, use the **no** form of the command.

```
ciscoasa(config-group-policy)# no msie-proxy method [auto-detect | no-modify | no-proxy |
use-server]
ciscoasa(config-group-policy)#
```

The available methods are as follows:

- **auto-detect**—Enables the use of automatic proxy server detection in the browser for the client device.
- **no-modify**—Leaves the HTTP browser proxy server setting in the browser unchanged for this client device.
- **no-proxy—**Disables the HTTP proxy setting in the browser for the client device.
- **use-server—**Sets the HTTP proxy server setting in the browser to use the value configured in the **msie-proxy server** command.

The line containing the proxy server IP address or hostname and the port number must be less than 100 characters long.

The following example shows how to configure auto-detect as the browser proxy setting for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy method auto-detect
ciscoasa(config-group-policy)#
```

The following example configures the browser proxy setting for the group policy named FirstGroup to use the server QAserver, port 1001 as the server for the client device:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy server QAserver:port 1001
ciscoasa(config-group-policy)# msie-proxy method use-server
ciscoasa(config-group-policy)#
```

Step 3 Configure browser proxy exception list settings for a local bypass on the client device by entering the **msie-proxy except-list** command in group-policy configuration mode. These addresses are not accessed by a proxy server. This list corresponds to the Exceptions box in the Proxy Settings dialog box.

```
ciscoasa(config-group-policy)# msie-proxy except-list {value server[:port] | none}
ciscoasa(config-group-policy)#
```

To remove the attribute from the configuration, use the **no** form of the command:

```
ciscoasa(config-group-policy)# no msie-proxy except-list
ciscoasa(config-group-policy)#
```

- **value** *server:port*—Specifies the IP address or name of an MSIE server and port that is applied for this client device. The port number is optional.
- **none**—Indicates that there is no IP address/hostname or port and prevents inheriting an exception list.

By default, msie-proxy except-list is disabled.

The line containing the proxy server IP address or hostname and the port number must be less than 100 characters long.

The following example shows how to set a browser proxy exception list, consisting of the server at IP address 192.168.20.1, using port 880, for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy except-list value 192.168.20.1:880
ciscoasa(config-group-policy)#
```

Step 4 Enable or disable browser proxy local-bypass settings for a client device by entering the **msie-proxy local-bypass** command in group-policy configuration mode.

```
ciscoasa(config-group-policy)# msie-proxy local-bypass {enable | disable}
ciscoasa(config-group-policy)#
```

To remove the attribute from the configuration, use the **no** form of the command.

```
ciscoasa(config-group-policy)# no msie-proxy local-bypass {enable | disable}
ciscoasa(config-group-policy)#
```

By default, msie-proxy local-bypass is disabled.

The following example shows how to enable browser proxy local-bypass for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy local-bypass enable
ciscoasa(config-group-policy)#
```

Configuring Group Policy Attributes for AnyConnect Secure Mobility Client Connections

After enabling AnyConnect client connections as described in Chapter 11, "Configuring AnyConnect VPN Client Connections", you can enable or require AnyConnect features for a group policy. Follow these steps in group-policy webvpn configuration mode:

Step 1 Enter group policy webvpn configuration mode. For example:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
```

Step 2 To disable the permanent installation of the AnyConnect client on the endpoint computer, use the **anyconnect keep-installer** command with the **none** keyword. For example:

```
hostname(config-group-webvpn)# anyconnect keep-installer none
hostname(config-group-webvpn)#
```

The default is that permanent installation of the client is enabled. The client remains installed on the endpoint at the end of the AnyConnect session.

Step 3 To enable compression of HTTP data over an AnyConnect SSL connection for the group policy, enter the **anyconnect ssl compression** command. By default, compression is set to **none** (disabled). To enable compression, use the **deflate** keyword. For example:

```
hostname(config-group-webvpn)# anyconnect compression deflate
hostname(config-group-webvpn)#
```

Step 4 To enable dead peer detection (DPD) on the ASA and to set the frequency with which either the AnyConnect client or the ASA performs DPD, use the **anyconnect dpd-interval** command:

anyconnect dpd-interval {[**gateway** {*seconds* | **none**}] | [**client** {*seconds* | **none**}]}

By default, both the ASA and the AnyConnect client perform DPD every 30 seconds.

The gateway refers to the ASA. You can specify the frequency with which the ASA performs the DPD test as a range of from 30 to 3600 seconds (1 hour). Specifying **none** disables the DPD testing that the ASA performs. A value of 300 is recommended.

The client refers to the AnyConnect client. You can specify the frequency with which the client performs the DPD test as a range of from 30 to 3600 seconds (1 hour). Specifying **none** disables the DPD testing that the client performs. A value of 30 is recommended.

The following example configures the DPD frequency performed by the ASA (gateway) to 300 seconds, and the DPD frequency performed by the client to 30 seconds:

```
hostname(config-group-webvpn)# anyconnect dpd-interval gateway 300
hostname(config-group-webvpn)# anyconnect dpd-interval client 30
hostname(config-group-webvpn)#
```

Step 5 You can ensure that an AnyConnect connection through a proxy, firewall, or NAT device remains open, even if the device limits the time that the connection can be idle by adjusting the frequency of keepalive messages using the **anyconnect ssl keepalive comand:**

anyconnect ssl keepalive {none | *seconds***}**

Adjusting keepalives also ensures the AnyConnect client does not disconnect and reconnect when the remote user is not actively running a socket-based application, such as Microsoft Outlook or Microsoft Internet Explorer.

The following example configures the security appliance to enable the AnyConnect client to send keepalive messages, with a frequency of 300 seconds (5 minutes):

```
hostname(config-group-webvpn)# anyconnect ssl keepalive 300
hostname(config-group-webvpn)#
```

Step 6 To enable the AnyConnect client to perform a re-key on an SSL session, use the **anyconnect ssl rekey** command:

anyconnect ssl rekey {**method** {**ssl** | **new-tunnel**} | **time** *minutes* | **none**}}

By default, re-key is disabled.

Specifying the method as new-tunnel specifies that the AnyConnect client establishes a new tunnel during SSL re-key. Specifying the method as **none** disables re-key. Specifying the method as **ssl** specifies that SSL renegotiation takes place during re-key. Instead of specifying the method, you can specify the time; that is, the number of minutes from the start of the session until the re-key takes place, from 1 through 10080 (1 week).

The following example configures the AnyConnect client to renegotiate with SSL during re-key and configures the re-key to occur 30 minutes after the session begins:

```
hostname(config-group-webvpn)# anyconnect ssl rekey method ssl
hostname(config-group-webvpn)# anyconnect ssl rekey time 30
hostname(config-group-webvpn)#
```

Step 7 The Client Protocol Bypass feature allows you to configure how the ASA manages IPv4 traffic when it is expecting only IPv6 traffic or how it manages IPv6 traffic when it is expecting only IPv4 traffic.

When the AnyConnect client makes a VPN connection to the ASA, the ASA could assign it an IPv4, IPv6, or both an IPv4 and IPv6 address. If the ASA assigns the AnyConnect connection only an IPv4 address or only an IPv6 address, you can now configure the Client Bypass Protocol to drop network traffic for which the ASA did not assign an IP address, or allow that traffic to bypass the ASA and be sent from the client unencrypted or “in the clear”.

For example, assume that the ASA assigns only an IPv4 address to an AnyConnect connection and the endpoint is dual stacked. When the endpoint attempts to reach an IPv6 address, if Client Bypass Protocol is disabled, the IPv6 traffic is dropped; however, if Client Bypass Protocol is enabled, the IPv6 traffic is sent from the client in the clear.

Use the client-bypass-protocol command to enable or disable the client bypass protocol feature. This is the command syntax:

```
client-bypass-protocol {enable | disable}
```

The following example enables client bypass protocol:

```
hostname(config-group-policy)# client-bypass-protocol enable
hostname(config-group-policy)#
```

The following example disables client bypass protocol:

```
hostname(config-group-policy)# client-bypass-protocol disable
hostname(config-group-policy)#
```

The following example removes an enabled or disabled client bypass protocol setting:

```
hostname(config-group-policy)# no client-bypass-protocol enable
hostname(config-group-policy)#
```

Step 8 If you have configured Load Balancing between your ASAs, specify the FQDN of the ASA in order to resolve the ASA IP address used for re-establishing the VPN session. This setting is critical to support client roaming between networks of different IP protocols (such as IPv4 to IPv6).

You cannot use the ASA FQDN present in the AnyConnect profile to derive the ASA IP address after roaming. The addresses may not match the correct device (the one the tunnel was established to) in the load balancing scenario.

If the device FQDN is not pushed to the client, the client will try to reconnect to whatever IP address the tunnel had previously established. In order to support roaming between networks of different IP protocols (from IPv4 to IPv6), AnyConnect must perform name resolution of the device FQDN after roaming, so that it can determine which ASA address to use for re-establishing the tunnel. The client uses the ASA FQDN present in its profile during the initial connection. During subsequent session reconnects, it always uses the device FQDN pushed by ASA (and configured by the administrator in the group policy), when available. If the FQDN is not configured, the ASA derives the device FQDN (and sends it to the client) from whatever is set under Device Setup > Device Name/Password and Domain Name.

If the device FQDN is not pushed by the ASA, the client cannot re-establish the VPN session after roaming between networks of different IP protocols.

Use the gateway-fqdn command to configure the FQDN of the ASA. This is the command syntax:

```
gateway-fqdn value {FQDN_Name | none}
no gateway-fqdn
```

The following example defines the FQDN of the ASA as ASAName.example.cisco.com

```
hostname(config-group-policy)# gateway-fqdn value ASAName.example.cisco.com
hostname(config-group-policy)#
```

The following example removes the FQDN of the ASA from the group policy. The group policy then inherits this value from the Default Group Policy.

```
hostname(config-group-policy)# no gateway-fqdn
hostname(config-group-policy)#
```

The following example defines the FQDN as an empty value. The global FQDN configurd using hostname and domain-name commands will be used if available.

```
hostname(config-group-policy)# gateway-fqdn none
hostname(config-group-policy)#
```

Configuring Group Policy Attributes for IPsec (IKEv1) Clients

Configuring Security Attributes for IPsec (IKEv1) Clients

To specify the security settings for a group, perform these steps.

Step 1 Specify whether to let users store their login passwords on the client system, using the **password-storage** command with the **enable** keyword in group-policy configuration mode. To disable password storage, use the **password-storage** command with the **disable** keyword.

```
ciscoasa(config-group-policy)# password-storage {enable | disable}
```

```
ciscoasa(config-group-policy)#
```

For security reasons, password storage is disabled by default. Enable password storage only on systems that you know to be in secure sites.

To remove the password-storage attribute from the running configuration, enter the **no** form of this command:

```
ciscoasa(config-group-policy)# no password-storage
ciscoasa(config-group-policy)#
```

Specifying the **no** form enables inheritance of a value for password-storage from another group policy.

This command does not apply to interactive hardware client authentication or individual user authentication for hardware clients.

The following example shows how to enable password storage for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# password-storage enable
ciscoasa(config-group-policy)#
```

Step 2 Specify whether to enable IP compression, which is disabled by default.

Note IP compression is not supported for IPsec IKEv2 connections.

```
ciscoasa(config-group-policy)# ip-comp {enable | disable}
ciscoasa(config-group-policy)#
```

To enable LZS IP compression, enter the **ip-comp** command with the **enable** keyword in group-policy configuration mode. To disable IP compression, enter the **ip-comp** command with the **disable** keyword.

To remove the **ip-comp** attribute from the running configuration, enter the **no** form of this command. This enables inheritance of a value from another group policy.

```
ciscoasa(config-group-policy)# no ip-comp
ciscoasa(config-group-policy)#
```

Enabling data compression might speed up data transmission rates for remote dial-in users connecting with modems.

Caution Data compression increases the memory requirement and CPU usage for each user session and consequently decreases the overall throughput of the ASA. For this reason, we recommend that you enable data compression only for remote users connecting with a modem. Design a group policy specific to modem users, and enable compression only for them.

Step 3 Specify whether to require that users reauthenticate on IKE re-key by using the **re-xauth** command with the **enable** keyword in group-policy configuration mode.

Note IKE re-key is not supported for IKEv2 connections.

If you enable reauthentication on IKE re-key, the ASA prompts the user to enter a username and password during initial Phase 1 IKE negotiation and also prompts for user authentication whenever an IKE re-key occurs. Reauthentication provides additional security.

If the configured re-key interval is very short, users might find the repeated authorization requests inconvenient. To avoid repeated authorization requests, disable reauthentication. To check the configured re-key interval, in monitoring mode, enter the **show crypto ipsec sa** command to view the security association lifetime in seconds and lifetime in kilobytes of data. To disable user reauthentication on IKE re-key, enter the **disable** keyword. Reauthentication on IKE re-key is disabled by default.

```
ciscoasa(config-group-policy)# re-xauth {enable | disable}
ciscoasa(config-group-policy)#
```

To enable inheritance of a value for reauthentication on IKE re-key from another group policy, remove the re-xauth attribute from the running configuration by entering the **no** form of this command:

```
ciscoasa(config-group-policy)# no re-xauth
ciscoasa(config-group-policy)#
```

Note Reauthentication fails if there is no user at the other end of the connection.

Step 4 Specify whether to enable perfect forward secrecy. In IPsec negotiations, perfect forward secrecy ensures that each new cryptographic key is unrelated to any previous key. A group policy can inherit a value for perfect forward secrecy from another group policy. Perfect forward secrecy is disabled by default. To enable perfect forward secrecy, use the **pfs** command with the **enable** keyword in group-policy configuration mode.

```
ciscoasa(config-group-policy)# pfs {enable | disable}
ciscoasa(config-group-policy)#
```

To disable perfect forward secrecy, enter the **pfs** command with the **disable** keyword.

To remove the perfect forward secrecy attribute from the running configuration and prevent inheriting a value, enter the **no** form of this command.

```
ciscoasa(config-group-policy)# no pfs
ciscoasa(config-group-policy)#
```

Configuring IPsec-UDP Attributes for IKEv1 Clients

IPsec over UDP, sometimes called IPsec through NAT, lets a Cisco VPN client or hardware client connect via UDP to a ASA that is running NAT. It is disabled by default. IPsec over UDP is proprietary; it applies only to remote-access connections, and it requires mode configuration. The ASA exchanges configuration parameters with the client while negotiating SAs. Using IPsec over UDP may slightly degrade system performance.

To enable IPsec over UDP, configure the **ipsec-udp** command with the **enable** keyword in group-policy configuration mode, as follows:

```
ciscoasa(config-group-policy)# ipsec-udp {enable | disable}
ciscoasa(config-group-policy)# no ipsec-udp
```

To use IPsec over UDP, you must also configure the **ipsec-udp-port** command, as described in this section.

To disable IPsec over UDP, enter the **disable** keyword. To remove the IPsec over UDP attribute from the running configuration, enter the **no** form of this command. This enables inheritance of a value for IPsec over UDP from another group policy.

The Cisco VPN client must also be configured to use IPsec over UDP (it is configured to use it by default). The VPN 3002 requires no configuration to use IPsec over UDP.

The following example shows how to set IPsec over UDP for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# ipsec-udp enable
```

If you enabled IPsec over UDP, you must also configure the **ipsec-udp-port** command in group-policy configuration mode. This command sets a UDP port number for IPsec over UDP. In IPsec negotiations, the ASA listens on the configured port and forwards UDP traffic for that port even if other filter rules drop UDP traffic. The port numbers can range from 4001 through 49151. The default port value is 10000.

To disable the UDP port, enter the **no** form of this command. This enables inheritance of a value for the IPsec over UDP port from another group policy.

```
ciscoasa(config-group-policy)# ipsec-udp-port port
```

The following example shows how to set an IPsec UDP port to port 4025 for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# ipsec-udp-port 4025
```

Configuring Attributes for VPN Hardware Clients

This section describes how to enable or disable secure unit authentication and user authentication or set a user authentication timeout value for VPN hardware clients. They also let you allow Cisco IP phones and LEAP packets to bypass individual user authentication and allow hardware clients using Network Extension Mode to connect.

Configuring Secure Unit Authentication

Secure unit authentication provides additional security by requiring VPN hardware clients to authenticate with a username and password each time that the client initiates a tunnel. With this feature enabled, the hardware client does not have a saved username and password. Secure unit authentication is disabled by default.

Note With this feature enabled, to bring up a VPN tunnel, a user must be present to enter the username and password.

Secure unit authentication requires that you have an authentication server group configured for the connection profile the hardware client(s) use. If you require secure unit authentication on the primary ASA, be sure to configure it on any backup servers as well.

Specify whether to enable secure unit authentication by entering the **secure-unit-authentication** command with the **enable** keyword in group-policy configuration mode.

```
ciscoasa(config-group-policy)# secure-unit-authentication {enable | disable}
ciscoasa(config-group-policy)# no secure-unit-authentication
```

To disable secure unit authentication, enter the **disable** keyword. To remove the secure unit authentication attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value for secure unit authentication from another group policy.

The following example shows how to enable secure unit authentication for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# secure-unit-authentication enable
```

Configuring User Authentication

User authentication is disabled by default. When enabled, user authentication requires that individual users behind a hardware client authenticate to gain access to the network across the tunnel. Individual users authenticate according to the order of authentication servers that you configure.

Specify whether to enable user authentication by entering the **user-authentication** command with the **enable** keyword in group-policy configuration mode.

```
ciscoasa(config-group-policy)# user-authentication {enable | disable}
ciscoasa(config-group-policy)# no user-authentication
```

To disable user authentication, enter the **disable** keyword. To remove the user authentication attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value for user authentication from another group policy.

If you require user authentication on the primary ASA, be sure to configure it on any backup servers as well.

The following example shows how to enable user authentication for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# user-authentication enable
```

Configuring an Idle Timeout

Set an idle timeout for individual users behind hardware clients by entering the **user-authentication-idle-timeout** command in group-policy configuration mode. If there is no communication activity by a user behind a hardware client in the idle timeout period, the ASA terminates the client's access:

```
ciscoasa(config-group-policy)# user-authentication-idle-timeout {minutes | none}
ciscoasa(config-group-policy)# no user-authentication-idle-timeout
```

Note This timer terminates only the client's access through the VPN tunnel, not the VPN tunnel itself.

The idle timeout indicated in response to the **show uauth** command is always the idle timeout value of the user who authenticated the tunnel on the Cisco Easy VPN remote device.

The *minutes* parameter specifies the number of minutes in the idle timeout period. The minimum is 1 minute, the default is 30 minutes, and the maximum is 35791394 minutes.

To delete the idle timeout value, enter the **no** form of this command. This option allows inheritance of an idle timeout value from another group policy.

To prevent inheriting an idle timeout value, enter the **user-authentication-idle-timeout** command with the **none** keyword. This command sets the idle timeout with a null value, which disallows an idle timeout and prevents inheriting an user authentication idle timeout value from a default or specified group policy.

The following example shows how to set an idle timeout value of 45 minutes for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# user-authentication-idle-timeout 45
```

Configuring IP Phone Bypass

You can allow Cisco IP phones to bypass individual user authentication behind a hardware client. To enable IP Phone Bypass, enter the **ip-phone-bypass** command with the **enable** keyword in group-policy configuration mode. IP Phone Bypass lets IP phones behind hardware clients connect without undergoing user authentication processes. IP Phone Bypass is disabled by default. If enabled, secure unit authentication remains in effect.

To disable IP Phone Bypass, enter the **disable** keyword. To remove the IP phone Bypass attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value for IP Phone Bypass from another group policy:

```
ciscoasa(config-group-policy)# ip-phone-bypass {enable | disable}
ciscoasa(config-group-policy)# no ip-phone-bypass
```

Note You must configure mac-exempt to exempt the clients from authentication. Refer to the "Configuring Device Pass-Through" section on page 8-8 for more information.

Configuring LEAP Bypass

When LEAP Bypass is enabled, LEAP packets from wireless devices behind a VPN 3002 hardware client travel across a VPN tunnel prior to user authentication. This action lets workstations using Cisco wireless access point devices establish LEAP authentication and then authenticate again per user authentication. LEAP Bypass is disabled by default.

To allow LEAP packets from Cisco wireless access points to bypass individual users authentication, enter the **leap-bypass** command with the **enable** keyword in group-policy configuration mode. To disable LEAP Bypass, enter the **disable** keyword. To remove the LEAP Bypass attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value for LEAP Bypass from another group policy:

```
ciscoasa(config-group-policy)# leap-bypass {enable | disable}
ciscoasa(config-group-policy)# no leap-bypass
```

Note IEEE 802.1X is a standard for authentication on wired and wireless networks. It provides wireless LANs with strong mutual authentication between clients and authentication servers, which can provide dynamic per-user, per session wireless encryption privacy (WEP) keys, removing administrative burdens and security issues that are present with static WEP keys.

Cisco Systems has developed an 802.1X wireless authentication type called Cisco LEAP. LEAP (Lightweight Extensible Authentication Protocol) implements mutual authentication between a wireless client on one side of a connection and a RADIUS server on the other side. The credentials used for authentication, including a password, are always encrypted before they are transmitted over the wireless medium.

Cisco LEAP authenticates wireless clients to RADIUS servers. It does not include RADIUS accounting services.

This feature does not work as intended if you enable interactive hardware client authentication.

Caution There might be security risks to your network in allowing any unauthenticated traffic to traverse the tunnel.

The following example shows how to set LEAP Bypass for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# leap-bypass enable
```

Enabling Network Extension Mode

Network extension mode lets hardware clients present a single, routable network to the remote private network over the VPN tunnel. IPsec encapsulates all traffic from the private network behind the hardware client to networks behind the ASA. PAT does not apply. Therefore, devices behind the ASA have direct access to devices on the private network behind the hardware client over the tunnel, and only over the tunnel, and vice versa. The hardware client must initiate the tunnel, but after the tunnel is up, either side can initiate data exchange.

Enable network extension mode for hardware clients by entering the **nem** command with the **enable** keyword in group-policy configuration mode:

```
ciscoasa(config-group-policy)# nem {enable | disable}
ciscoasa(config-group-policy)# no nem
```

To disable NEM, enter the **disable** keyword. To remove the NEM attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value from another group policy.

The following example shows how to set NEM for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# nem enable
```

Configuring Backup Server Attributes

Configure backup servers if you plan on using them. IPsec backup servers let a VPN client connect to the central site when the primary ASA is unavailable.When you configure backup servers, the ASA pushes the server list to the client as the IPsec tunnel is established. Backup servers do not exist until you configure them, either on the client or on the primary ASA.

Configure backup servers either on the client or on the primary ASA. If you configure backup servers on the ASA, it pushes the backup server policy to the clients in the group, replacing the backup server list on the client if one is configured.

Note If you are using hostnames, it is wise to have backup DNS and WINS servers on a separate network from that of the primary DNS and WINS servers. Otherwise, if clients behind a hardware client obtain DNS and WINS information from the hardware client via DHCP, and the connection to the primary server is lost, and the backup servers have different DNS and WINS information, clients cannot be updated until the DHCP lease expires. In addition, if you use hostnames and the DNS server is unavailable, significant delays can occur.

To configure backup servers, enter the **backup-servers** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# backup-servers {server1 server2... server10 |
clear-client-config | keep-client-config}
```

To remove a backup server, enter the **no** form of this command with the backup server specified. To remove the backup-servers attribute from the running configuration and enable inheritance of a value for backup-servers from another group policy, enter the **no** form of this command without arguments.

```
ciscoasa(config-group-policy)# no backup-servers [server1 server2... server10 |
clear-client-config | keep-client-config]
```

The **clear-client-config** keyword specifies that the client uses no backup servers. The ASA pushes a null server list.

The **keep-client-config** keyword specifies that the ASA sends no backup server information to the client. The client uses its own backup server list, if configured. This is the default.

The *server1 server 2.... server10* parameter list is a space-delimited, priority-ordered list of servers for the VPN client to use when the primary ASA is unavailable. This list identifies servers by IP address or hostname. The list can be 500 characters long, and it can contain up to10 entries.

The following example shows how to configure backup servers with IP addresses 10.10.10.1 and 192.168.10.14, for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# backup-servers 10.10.10.1 192.168.10.14
```

Configuring Network Admission Control Parameters

The group-policy NAC commands in this section all have default values. Unless you have a good reason for changing them, accept the default values for these parameters.

The ASA uses Extensible Authentication Protocol (EAP) over UDP (EAPoUDP) messaging to validate the posture of remote hosts. Posture validation involves the checking of a remote host for compliancy with safety requirements before the assignment of a network access policy. An Access Control Server must be configured for Network Admission Control before you configure NAC on the security appliance.

The Access Control Server downloads the posture token, an informational text string configurable on the ACS, to the security appliance to aid in system monitoring, reporting, debugging, and logging. A typical posture token is Healthy, Checkup, Quarantine, Infected, or Unknown. Following posture validation or clientless authentication, the ACS downloads the access policy for the session to the security appliance.

To configure Network Admission Control settings for the default group policy or an alternative group policy, perform the following steps.

Step 1 (Optional) Configure the status query timer period. The security appliance starts the status query timer after each successful posture validation and status query response. The expiration of this timer triggers a query for changes in the host posture, referred to as a status query. Enter the number of seconds in the range 30 through 1800. The default setting is 300.

To specify the interval between each successful posture validation in a Network Admission Control session and the next query for changes in the host posture, use the **nac-sq-period** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# nac-sq-period seconds
ciscoasa(config-group-policy)#
```

To inherit the value of the status query timer from the default group policy, access the alternative group policy from which to inherit it, then use the **no** form of this command:

```
ciscoasa(config-group-policy)# no nac-sq-period [seconds]
ciscoasa(config-group-policy)#
```

The following example changes the value of the status query timer to 1800 seconds:

```
ciscoasa(config-group-policy)# nac-sq-period 1800
ciscoasa(config-group-policy)
```

The following example inherits the value of the status query timer from the default group policy:

```
ciscoasa(config-group-policy)# no nac-sq-period
ciscoasa(config-group-policy)#
```

Step 2 (*Optional*) Configure the NAC revalidation period. The security appliance starts the revalidation timer after each successful posture validation. The expiration of this timer triggers the next unconditional posture validation. The security appliance maintains posture validation during revalidation. The default group policy becomes effective if the Access Control Server is unavailable during posture validation or revalidation. Enter the interval in seconds between each successful posture validation. The range is 300 through 86400. The default setting is 36000.

To specify the interval between each successful posture validation in a Network Admission Control session, use the **nac-reval-period** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# nac-reval-period seconds
ciscoasa(config-group-policy)#
```

To inherit the value of the Revalidation Timer from the default group policy, access the alternative group policy from which to inherit it, then use the **no** form of this command:

```
ciscoasa(config-group-policy)# no nac-reval-period [seconds]
ciscoasa(config-group-policy)#
```

The following example changes the revalidation timer to 86400 seconds:

```
ciscoasa(config-group-policy)# nac-reval-period 86400
ciscoasa(config-group-policy)
```

The following example inherits the value of the revalidation timer from the default group policy:

```
ciscoasa(config-group-policy)# no nac-reval-period
ciscoasa(config-group-policy)#
```

Step 3 (*Optional*) Configure the default ACL for NAC. The security appliance applies the security policy associated with the selected ACL if posture validation fails. Specify **none** or an extended ACL. The default setting is **none**. If the setting is **none** and posture validation fails, the security appliance applies the default group policy.

To specify the ACL to be used as the default ACL for Network Admission Control sessions that fail posture validation, use the **nac-default-acl** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# nac-default-acl {acl-name | none}
ciscoasa(config-group-policy)#
```

To inherit the ACL from the default group policy, access the alternative group policy from which to inherit it, then use the **no** form of this command:

```
ciscoasa(config-group-policy)# no nac-default-acl [acl-name | none]
ciscoasa(config-group-policy)#
```

The elements of this command are as follows:

- *acl-name*—Specifies the name of the posture validation server group, as configured on the ASA using the **aaa-server host** command. The name must match the server-tag variable specified in that command.
- **none**—Disables inheritance of the ACL from the default group policy and does not apply an ACL to NAC sessions that fail posture validation.

Because NAC is disabled by default, VPN traffic traversing the ASA is not subject to the NAC Default ACL until NAC is enabled.

The following example identifies acl-1 as the ACL to be applied when posture validation fails:

```
ciscoasa(config-group-policy)# nac-default-acl acl-1
ciscoasa(config-group-policy)
```

The following example inherits the ACL from the default group policy:

```
ciscoasa(config-group-policy)# no nac-default-acl
ciscoasa(config-group-policy)
```

The following example disables inheritance of the ACL from the default group policy and does not apply an ACL to NAC sessions that fail posture validation:

```
ciscoasa(config-group-policy)# nac-default-acl none
ciscoasa(config-group-policy)#
```

Step 4 Configure NAC exemptions for VPN. By default, the exemption list is empty.The default value of the filter attribute is **none**. Enter the **vpn-nac-exempt command** once for each operating system (and ACL) to be matched to exempt remote hosts from posture validation.

To add an entry to the list of remote computer types that are exempt from posture validation, use the **vpn-nac-exempt** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# vpn-nac-exempt os "os name" [filter {acl-name | none}]
[disable]
ciscoasa(config-group-policy)#
```

To disable inheritance and specify that all hosts are subject to posture validation, use the **none** keyword immediately following **vpn-nac-exempt**:

```
ciscoasa(config-group-policy)# vpn-nac-exempt none
ciscoasa(config-group-policy)#
```

To remove an entry from the exemption list, use the **no** form of this command and name the operating system (and ACL) in the entry to be removed:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt [os "os name"] [filter {acl-name | none}]
[disable]
ciscoasa(config-group-policy)#
```

To remove all entries from the exemption list associated with this group policy and inherit the list from the default group policy, use the **no** form of this command without specifying additional keywords:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt
ciscoasa(config-group-policy)#
```

The syntax elements for these commands are as follows:

- *acl-name*—Name of the ACL present in the ASA configuration.
- **disable**—Disables the entry in the exemption list without removing it from the list.
- **filter**—(Optional) filter to apply an ACL to filter the traffic if the computer matches the os name.
- **none**—When entered immediately after **vpn-nac-exempt**, this keyword disables inheritance and specifies that all hosts will be subject to posture validation.When entered immediately after **filter**, this keyword indicates that the entry does not specify an ACL.
- **OS—**Exempts an operating system from posture validation.
- *os name*—Operating system name. Quotation marks are required only if the name includes a space (for example, “Windows XP”).

The following example adds all hosts running Windows XP to the list of computers that are exempt from posture validation:

```
ciscoasa(config-group-policy)# vpn-nac-exempt os "Windows XP"
ciscoasa(config-group-policy)
```

The following example exempts all hosts running Windows 98 that match an ACE in the ACL named acl-1:

```
ciscoasa(config-group-policy)# vpn-nac-exempt os "Windows 98" filter acl-1
ciscoasa(config-group-policy)
```

The following example adds the same entry to the exemption list, but disables it:

```
ciscoasa(config-group-policy)# vpn-nac-exempt os "Windows 98" filter acl-1 disable
ciscoasa(config-group-policy)
```

The following example removes the same entry from the exemption list, regardless of whether it is disabled:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt os "Windows 98" filter acl-1
ciscoasa(config-group-policy)
```

The following example disables inheritance and specifies that all hosts will be subject to posture validation:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt none
ciscoasa(config-group-policy)
```

The following example removes all entries from the exemption list:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt
ciscoasa(config-group-policy)
```

Step 5 Enable or disable Network Admission Control by entering the following command:

```
ciscoasa(config-group-policy)# nac {enable | disable}
ciscoasa(config-group-policy)#
```

To inherit the NAC setting from the default group policy, access the alternative group policy from which to inherit it, then use the **no** form of this command:

```
ciscoasa(config-group-policy)# no nac [enable | disable]
ciscoasa(config-group-policy)#
```

By default, NAC is disabled. Enabling NAC requires posture validation for remote access. If the remote computer passes the validation checks, the ACS server downloads the access policy for the ASA to enforce. NAC is disabled by default.

An Access Control Server must be present on the network.

The following example enables NAC for the group policy:

```
ciscoasa(config-group-policy)# nac enable
ciscoasa(config-group-policy)#
```

Configuring VPN Client Firewall Policies

A *firewall* isolates and protects a computer from the Internet by inspecting each inbound and outbound packet of data to determine whether to allow it through the firewall or to drop it. Firewalls provide extra security if remote users in a group have split tunneling configured. In this case, the firewall protects the user's computer, and thereby the corporate network, from intrusions by way of the Internet or the user's local LAN. Remote users connecting to the ASA with the VPN client can choose the appropriate firewall option.

Set personal firewall policies that the ASA pushes to the VPN client during IKE tunnel negotiation by using the **client-firewall** command in group-policy configuration mode. To delete a firewall policy, enter the **no** form of this command.

To delete all firewall policies, enter the **no client-firewall** command without arguments. This command deletes all configured firewall policies, including a null policy if you created one by entering the **client-firewall** command with the **none** keyword.

When there are no firewall policies, users inherit any that exist in the default or other group policy. To prevent users from inheriting such firewall policies, enter the **client-firewall** command with the **none** keyword.

The Add or Edit Group Policy dialog box on the Client Firewall tab, lets you configure firewall settings for VPN clients for the group policy being added or modified.

Note Only VPN clients running Microsoft Windows can use these firewall features. They are currently not available to hardware clients or other (non-Windows) software clients.

In the first scenario, a remote user has a personal firewall installed on the PC. The VPN client enforces firewall policy defined on the local firewall, and it monitors that firewall to make sure it is running. If the firewall stops running, the VPN client drops the connection to the ASA. (This firewall enforcement mechanism is called *Are You There (AYT)*, because the VPN client monitors the firewall by sending it periodic "are you there?" messages; if no reply comes, the VPN client knows the firewall is down and terminates its connection to the ASA.) The network administrator might configure these PC firewalls originally, but with this approach, each user can customize his or her own configuration.

In the second scenario, you might prefer to enforce a centralized firewall policy for personal firewalls on VPN client PCs. A common example would be to block Internet traffic to remote PCs in a group using split tunneling. This approach protects the PCs, and therefore the central site, from intrusions from the Internet while tunnels are established. This firewall scenario is called *push policy* or *Central Protection Policy (CPP)*. On the ASA, you create a set of traffic management rules to enforce on the VPN client, associate those rules with a filter, and designate that filter as the firewall policy. The ASA pushes this policy down to the VPN client. The VPN client then in turn passes the policy to the local firewall, which enforces it.

Configuring AnyConnect Client Firewall Policies

Fiewall rules for the AnyConnect client can specify IPv4 and IPv6 addresses.

Prerequisites

You have created Unified Access Rules with IPv6 addresses specified.

Table 4-1

	Command	Description
Step 1	**webvpn** **Example:** hostname(config)# **group-policy ac-client-group attributes** hostname(config-group-policy)# **webvpn** hostname(config-group-webvpn)#	Enter webvpn group policy configuration mode.
Step 2	**anyconnect firewall-rule client-interface {private | public} value** *[RuleName]* **Example:** hostname(config-group-webvpn)# **anyconnect fireall-rule client-iterface private value** *ClientFWRule*	Specifies an acces control rule for the private or public network rule. The private network rule is the rule applied to the VPN virtual dapter interface on the cient.
Step 3	**show runn group-policy** [value] **Example:** hostname(config-group-webvpn)# **show runn group-policy FirstGroup** group-policy FirstGroup internal group-policy FirstGroup attributes webvpn anyconnect firewall-rule client-interface private value ClientFWRule	Displays the group policy attributes as well as the webvpn policy attribute for the group policy.
Step 4	(optional) **no anyconnect firewall-rule client-ineterface private value [***RuleName***]** **Example:** hostname(config-group-webvpn)#**no anyconnect firewall-rule client-ineterface private value** hostname(config-group-webvpn)#	Removes the client firewall rule from the private network rule.

Supporting a Zone Labs Integrity Server

This section introduces the Zone Labs Integrity server, also called the Check Point Integrity server, and presents an example procedure for configuring the ASA to support the Zone Labs Integrity server. The Integrity server is a central management station for configuring and enforcing security policies on remote PCs. If a remote PC does not conform to the security policy dictated by the Integrity server, it is not granted access to the private network protected by the Integrity server and ASA.

This section includes the following topics:

Overview of the Integrity Server and ASA Interaction

The VPN client software and the Integrity client software are co-resident on a remote PC. The following steps summarize the actions of the remote PC, ASA, and Integrity server in the establishment of a session between the PC and the enterprise private network:

1. The VPN client software (residing on the same remote PC as the Integrity client software) connects to the ASA and tells the ASA what type of firewall client it is.
2. After the ASA approves the client firewall type, the ASA passes Integrity server address information back to the Integrity client.
3. With the ASA acting as a proxy, the Integrity client establishes a restricted connection with the Integrity server. A restricted connection is only between the Integrity client and the Integrity server.
4. The Integrity server determines if the Integrity client is in compliance with the mandated security policies. If the Integrity client is in compliance with security policies, the Integrity server instructs the ASA to open the connection and provide the Integrity client with connection details.
5. On the remote PC, the VPN client passes connection details to the Integrity client and signals that policy enforcement should begin immediately and the Integrity client can enter the private network.
6. After the VPN connection is established, the Integrity server continues to monitor the state of the Integrity client using client heartbeat messages.

Note The current release of the ASA supports one Integrity server at a time, even though the user interfaces support the configuration of up to five Integrity servers. If the active Integrity server fails, configure another one on the ASA and then reestablish the VPN client session.

Configuring Integrity Server Support

This section describes an example procedure for configuring the ASA to support the Zone Labs Integrity servers. The procedure involves configuring address, port, connection fail timeout and fail states, and SSL certificate parameters.

To configure the Integrity server, perform the following steps:

	Command	Purpose
Step 1	**zonelabs-Integrity server-address** {*hostname1* | *ip-address1*} **Example:** `ciscoasa(config)# zonelabs-Integrity server-address 10.0.0.5`	Configures an Integrity server using the IP address 10.0.0.5.
Step 2	**zonelabs-integrity port** *port-number* **Example:** `ciscoasa(config)# zonelabs-integrity port 300`	Specifies port 300 (the default port is 5054).

	Command	Purpose
Step 3	**zonelabs-integrity interface** *interface* **Example:** `ciscoasa(config)# zonelabs-integrity interface inside`	Specifies the inside interface for communications with the Integrity server.
Step 4	**zonelabs-integrity fail-timeout** *timeout* **Example:** `ciscoasa(config)# zonelabs-integrity fail-timeout 12`	Ensures that the ASA waits 12 seconds for a response from either the active or standby Integrity servers before declaring the Integrity server as failed and closing the VPN client connections. **Note** If the connection between the ASA and the Integrity server fails, the VPN client connections remain open by default so that the enterprise VPN is not disrupted by the failure of an Integrity server. However, you may want to close the VPN connections if the Zone Labs Integrity server fails.
Step 5	**zonelabs-integrity fail-close** **Example:** `ciscoasa(config)# zonelabs-integrity fail-close`	Configures the ASA so that connections to VPN clients close when the connection between the ASA and the Zone Labs Integrity server fails.
Step 6	**zonelabs-integrity fail-open** **Example:** `ciscoasa(config)# zonelabs-integrity fail-open`	Returns the configured VPN client connection fail state to the default and ensures that the client connections remain open.
Step 7	**zonelabs-integrity ssl-certificate-port** *cert-port-number* **Example:** `ciscoasa(config)# zonelabs-integrity ssl-certificate-port 300`	Specifies that the Integrity server connects to port 300 (the default is port 80) on the ASA to request the server SSL certificate.
Step 8	**zonelabs-integrity ssl-client-authentication {enable | disable}** **Example:** `ciscoasa(config)# zonelabs-integrity ssl-client-authentication enable`	While the server SSL certificate is always authenticated, also specifies that the client SSL certificate of the Integrity server be authenticated.

To set the firewall client type to the Zone Labs Integrity type, enter the following command:

Command	Purpose
client-firewall {opt | req} zonelabs-integrity **Example:** `ciscoasa(config)# client-firewall req zonelabs-integrity`	For more information, see the "Configuring VPN Client Firewall Policies" section on page 4-71. The command arguments that specify firewall policies are not used when the firewall type is **zonelabs-integrity**, because the Integrity server determines these policies.

Setting Client Firewall Parameters

Enter the following commands to set the appropriate client firewall parameters. You can configure only one instance of each command. For more information, see the "Configuring VPN Client Firewall Policies" section on page 4-71.

Cisco Integrated Firewall

```
ciscoasa(config-group-policy)# client-firewall {opt | req} cisco-integrated acl-in ACL
acl-out ACL
```

Cisco Security Agent

```
ciscoasa(config-group-policy)# client-firewall {opt | req} cisco-security-agent
```

No Firewall

```
ciscoasa(config-group-policy)# client-firewall none
```

Custom Firewall

```
ciscoasa(config-group-policy)# client-firewall {opt | req} custom vendor-id num product-id
num policy {AYT | CPP acl-in ACL acl-out ACL} [description string]
```

Zone Labs Firewalls

```
ciscoasa(config-group-policy)# client-firewall {opt | req} zonelabs-integrity
```

Note When the firewall type is **zonelabs-integrity**, do not include arguments. The Zone Labs Integrity Server determines the policies.

```
ciscoasa(config-group-policy)# client-firewall {opt | req} zonelabs-zonealarm policy {AYT
| CPP acl-in ACL acl-out ACL}

ciscoasa(config-group-policy)# client-firewall {opt | req} zonelabs-zonealarmorpro policy
{AYT | CPP acl-in ACL acl-out ACL}

client-firewall {opt | req} zonelabs-zonealarmpro policy {AYT | CPP acl-in ACL acl-out
ACL}
```

Sygate Personal Firewalls

```
ciscoasa(config-group-policy)# client-firewall {opt | req} sygate-personal

ciscoasa(config-group-policy)# client-firewall {opt | req} sygate-personal-pro

ciscoasa(config-group-policy)# client-firewall {opt | req} sygate-security-agent
```

Network Ice, Black Ice Firewall:

```
ciscoasa(config-group-policy)# client-firewall {opt | req} networkice-blackice
```

Table 4-4 client-firewall Command Keywords and Variables

Parameter	Description
acl-in *ACL*	Provides the policy the client uses for inbound traffic.
acl-out *ACL*	Provides the policy the client uses for outbound traffic.
AYT	Specifies that the client PC firewall application controls the firewall policy. The ASA checks to make sure that the firewall is running. It asks, "Are You There?" If there is no response, the ASA tears down the tunnel.
cisco-integrated	Specifies Cisco Integrated firewall type.
cisco-security-agent	Specifies Cisco Intrusion Prevention Security Agent firewall type.
CPP	Specifies Policy Pushed as source of the VPN client firewall policy.
custom	Specifies Custom firewall type.
description *string*	Describes the firewall.
networkice-blackice	Specifies Network ICE Black ICE firewall type.
none	Indicates that there is no client firewall policy. Sets a firewall policy with a null value, thereby disallowing a firewall policy. Prevents inheriting a firewall policy from a default or specified group policy.
opt	Indicates an optional firewall type.
product-id	Identifies the firewall product.
req	Indicates a required firewall type.
sygate-personal	Specifies the Sygate Personal firewall type.
sygate-personal-pro	Specifies Sygate Personal Pro firewall type.
sygate-security-agent	Specifies Sygate Security Agent firewall type.
vendor-id	Identifies the firewall vendor.
zonelabs-integrity	Specifies Zone Labs Integrity Server firewall type.
zonelabs-zonealarm	Specifies Zone Labs Zone Alarm firewall type.
zonelabs-zonealarmorpro policy	Specifies Zone Labs Zone Alarm or Pro firewall type.
zonelabs-zonealarmpro policy	Specifies Zone Labs Zone Alarm Pro firewall type.

The following example shows how to set a client firewall policy that requires Cisco Intrusion Prevention Security Agent for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# client-firewall req cisco-security-agent
ciscoasa(config-group-policy)#
```

Configuring Client Access Rules

Configure rules that limit the remote access client types and versions that can connect via IPsec through the ASA by using the **client-access-rule** command in group-policy configuration mode. Construct rules according to these guidelines:

- If you do not define any rules, the ASA permits all connection types.
- When a client matches none of the rules, the ASA denies the connection. If you define a deny rule, you must also define at least one permit rule; otherwise, the ASA denies all connections.
- For both software and hardware clients, type and version must exactly match their appearance in the **show vpn-sessiondb remote** display.
- The * character is a wildcard, which you can enter multiple times in each rule. For example, **client-access rule 3 deny type * version 3.*** creates a priority 3 client access rule that denies all client types running versions 3.x software.
- You can construct a maximum of 25 rules per group policy.
- There is a limit of 255 characters for an entire set of rules.
- You can enter n/a for clients that do not send client type and/or version.

To delete a rule, enter the **no** form of this command. This command is equivalent to the following command:

```
ciscoasa(config-group-policy)# client-access-rule 1 deny type "Cisco VPN Client" version
4.0
```

To delete all rules, enter the **no client-access-rule command** without arguments. This deletes all configured rules, including a null rule if you created one by issuing the **client-access-rule** command with the **none** keyword.

By default, there are no access rules. When there are no client access rules, users inherit any rules that exist in the default group policy.

To prevent users from inheriting client access rules, enter the **client-access-rule** command with the **none** keyword. The result of this command is that all client types and versions can connect.

```
ciscoasa(config-group-policy)# client-access rule priority {permit | deny} type type
version {version | none}

ciscoasa(config-group-policy)# no client-access rule [priority {permit | deny} type type
version version]
```

Table 4-5 explains the meaning of the keywords and parameters in these commands.

Table 4-5 client-access rule Command Keywords and Variables

Parameter	Description
deny	Denies connections for devices of a particular type and/or version.
none	Allows no client access rules. Sets client-access-rule to a null value, thereby allowing no restriction. Prevents inheriting a value from a default or specified group policy.
permit	Permits connections for devices of a particular type and/or version.
priority	Determines the priority of the rule. The rule with the lowest integer has the highest priority. Therefore, the rule with the lowest integer that matches a client type and/or version is the rule that applies. If a lower priority rule contradicts, the ASA ignores it.

Table 4-5 client-access rule Command Keywords and Variables

type *type*	Identifies device types via free-form strings, for example VPN 3002. A string must match exactly its appearance in the **show vpn-sessiondb remote** display, except that you can enter the * character as a wildcard.
version *version*	Identifies the device version via free-form strings, for example 7.0. A string must match exactly its appearance in the **show vpn-sessiondb remote** display, except that you can enter the * character as a wildcard.

The following example shows how to create client access rules for the group policy named FirstGroup. These rules permit Cisco VPN clients running software version 4.x, while denying all Windows NT clients:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# client-access-rule 1 deny type WinNT version *
ciscoasa(config-group-policy)# client-access-rule 2 permit "Cisco VPN Client" version 4.*
```

Note The "type" field is a free-form string that allows any value, but that value must match the fixed value that the client sends to the ASA at connect time.

Configuring Group Policy Attributes for Clientless SSL VPN Sessions

Clientless SSL VPN lets users establish a secure, remote-access VPN tunnel to the ASA using a web browser. There is no need for either a software or hardware client. Clientless SSL VPN provides easy access to a broad range of web resources and web-enabled applications from almost any computer that can reach HTTPS Internet sites. Clientless SSL VPN uses SSL and its successor, TLS1, to provide a secure connection between remote users and specific, supported internal resources that you configure at a central site. The ASA recognizes connections that need to be proxied, and the HTTP server interacts with the authentication subsystem to authenticate users. By default, clientless SSL VPN is disabled.

You can customize a configuration of clientless SSL VPN for specific internal group policies.

Note The webvpn mode that you enter from global configuration mode lets you configure global settings for clientless SSL VPN sessions. The webvpn mode described in this section, which you enter from group-policy configuration mode, lets you customize a configuration of group policies specifically for clientless SSL VPN sessions.

In group-policy webvpn configuration mode, you can specify whether to inherit or customize the following parameters, each of which is described in the subsequent sections:

- customizations
- html-content-filter
- homepage
- filter
- url-list
- port-forward
- port-forward-name

- sso server (single-signon server)
- auto-signon
- deny message
- AnyConnect Secure Mobility Client
- keep-alive ignore
- HTTP compression

In many instances, you define the webvpn attributes as part of configuring clientless SSL VPN, then you apply those definitions to specific groups when you configure the group-policy webvpn attributes. Enter group-policy webvpn configuration mode by using the **webvpn** command in group-policy configuration mode. Webvpn commands for group policies define access to files, URLs and TCP applications over clientless SSL VPN sessions. They also identify ACLs and types of traffic to filter. Clientless SSL VPN is disabled by default. See the description of *Chapter 14, "Introduction to Clientless SSL VPN"* for more information about configuring the attributes for clientless SSL VPN sessions.

To remove all commands entered in group-policy webvpn configuration mode, enter the **no** form of this command. These webvpn commands apply to the username or group policy from which you configure them.

```
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-policy)# no webvpn
```

The following example shows how to enter group-policy webvpn configuration mode for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)#
```

Applying Customization

Customizations determine the appearance of the windows that the user sees upon login. You configure the customization parameters as part of configuring clientless SSL VPN. To apply a previously defined web-page customization to change the look-and-feel of the web page that the user sees at login, enter the customization command in group-policy webvpn configuration mode:

```
ciscoasa(config-group-webvpn)# customization customization_name
ciscoasa(config-group-webvpn)#
```

For example, to use the customization named blueborder, enter the following command:

```
ciscoasa(config-group-webvpn)# customization blueborder
ciscoasa(config-group-webvpn)#
```

You configure the customization itself by entering the **customization** command in webvpn mode.

The following example shows a command sequence that first establishes a customization named 123 that defines a password prompt. The example then defines a group policy named testpolicy and uses the **customization** command to specify the use of the customization named 123 for clientless SSL VPN sessions:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# customization 123
ciscoasa(config-webvpn-custom)# password-prompt Enter password
ciscoasa(config-webvpn)# exit
ciscoasa(config)# group-policy testpolicy nopassword
ciscoasa(config)# group-policy testpolicy attributes
ciscoasa(config-group-policy)# webvpn
```

```
ciscoasa(config-group-webvpn)# customization value 123
ciscoasa(config-group-webvpn)#
```

Specifying a "Deny" Message

You can specify the message delivered to a remote user who logs into a clientless SSL VPN session successfully, but has no VPN privileges, by entering the **deny-message** command in group-policy webvpn configuration mode:

```
ciscoasa(config-group-webvpn)# deny-message value "message"
ciscoasa(config-group-webvpn)# no deny-message value "message"
ciscoasa(config-group-webvpn)# deny-message none
```

The **no deny-message value** command removes the message string, so that the remote user does not receive a message.

The **no deny-message none** command removes the attribute from the connection profile policy configuration. The policy inherits the attribute value.

The message can be up to 491 alphanumeric characters long, including special characters, spaces, and punctuation, but not counting the enclosing quotation marks. The text appears on the remote user's browser upon login. When typing the string in the **deny-message value** command, continue typing even if the command wraps.

The default deny message is: "Login was successful, but because certain criteria have not been met or due to some specific group policy, you do not have permission to use any of the VPN features. Contact your IT administrator for more information."

The first command in the following example creates an internal group policy named group2. The subsequent commands modify the attributes, including the webvpn deny message associated with that policy.

```
ciscoasa(config)# group-policy group2 internal
ciscoasa(config)# group-policy group2 attributes
ciscoasa(config-group)# webvpn
ciscoasa(config-group-webvpn)# deny-message value "Your login credentials are OK. However,
you have not been granted rights to use the VPN features. Contact your administrator for
more information."
ciscoasa(config-group-webvpn)
```

Configuring Group Policy Filter Attributes for Clientless SSL VPN Sessions

Specify whether to filter Java, ActiveX, images, scripts, and cookies from clientless SSL VPN sessions for this group policy by using the **html-content-filter** command in webvpn mode. HTML filtering is disabled by default.

To remove a content filter, enter the **no** form of this command. To remove all content filters, including a null value created by issuing the **html-content-filter** command with the **none** keyword, enter the **no** form of this command without arguments. The **no** option allows inheritance of a value from another group policy. To prevent inheriting an html content filter, enter the **html-content-filter** command with the **none** keyword.

Using the command a second time overrides the previous setting.

```
hostname(config-group-webvpn)# html-content-filter {java | images | scripts | cookies |
none}

hostname(config-group-webvpn)# no html-content-filter [java | images | scripts | cookies |
none]
```

Table 4-6 describes the meaning of the keywords used in this command.

Table 4-6 filter Command Keywords

Keyword	Meaning
cookies	Removes cookies from images, providing limited ad filtering and privacy.
images	Removes references to images (removes <IMG> tags).
java	Removes references to Java and ActiveX (removes <EMBED>, <APPLET>, and <OBJECT> tags).
none	Indicates that there is no filtering. Sets a null value, thereby disallowing filtering. Prevents inheriting filtering values.
scripts	Removes references to scripting (removes <SCRIPT> tags).

The following example shows how to set filtering of JAVA and ActiveX, cookies, and images for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# html-content-filter java cookies images
hostname(config-group-webvpn)#
```

Specifying the User Home Page

Specify a URL for the web page that displays when a user in this group logs in by using the **homepage** command in group-policy webvpn configuration mode. There is no default home page.

To remove a configured home page, including a null value created by issuing the **homepage none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from another group policy. To prevent inheriting a home page, enter the **homepage none** command.

The **none** keyword indicates that there is no home page for clientless SSL VPN sessions. It sets a null value, thereby disallowing a home page and prevents inheriting an home page.

The *url-string* variable following the keyword **value** provides a URL for the home page. The string must begin with either http:// or https://.

```
hostname(config-group-webvpn)# homepage {value url-string | none}
hostname(config-group-webvpn)# no homepage
hostname(config-group-webvpn)#
```

Configuring Auto-Signon

The **auto-signon** command is a single sign-on method for users of clientless SSL VPN sessions. It passes the login credentials (username and password) to internal servers for authentication using NTLM authentication, basic authentication, or both. Multiple auto-signon commands can be entered and are processed according to the input order (early commands take precedence).

You can use the auto-signon feature in three modes: webvpn configuration, webvpn group configuration, or webvpn username configuration mode. The typical precedence behavior applies where username supersedes group, and group supersedes global. The mode you choose depends upon the desired scope of authentication.

To disable auto-signon for a particular user to a particular server, use the **no** form of the command with the original specification of IP block or URI. To disable authentication to all servers, use the **no** form without arguments. The **no** option allows inheritance of a value from the group policy.

The following example, entered in group-policy webvpn configuration mode, configures auto-signon for the user named anyuser, using basic authentication, to servers with IP addresses ranging from 10.1.1.0 to 10.1.1.255:

The following example commands configure auto-signon for users of clientless SSL VPN sessions, using either basic or NTLM authentication, to servers defined by the URI mask https://*.example.com/*:

```
ciscoasa(config)# group-policy ExamplePolicy attributes
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# auto-signon allow uri https://*.example.com/* auth-type all
hostname(config-group-webvpn)#
```

The following example commands configure auto-signon for users of clientless SSL VPN sessions, using either basic or NTLM authentication, to the server with the IP address 10.1.1.0, using subnet mask 255.255.255.0:

```
ciscoasa(config)# group-policy ExamplePolicy attributes
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# auto-signon allow ip 10.1.1.0 255.255.255.0 auth-type all
hostname(config-group-webvpn)#
```

Specifying the ACL for Clientless SSL VPN Sessions

Specify the name of the ACL to use for clientless SSL VPN sessions for this group policy or username by using the **filter** command in webvpn mode. Clientless SSL VPN ACLs do not apply until you enter the **filter** command to specify them.

To remove the ACL, including a null value created by issuing the **filter none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from another group policy. To prevent inheriting filter values, enter the **filter value none** command.

ACLs for clientless SSL VPN sessions do not apply until you enter the **filter** command to specify them.

You configure ACLs to permit or deny various types of traffic for this group policy. You then enter the **filter** command to apply those ACLs for clientless SSL VPN traffic.

```
hostname(config-group-webvpn)# filter {value ACLname | none}
hostname(config-group-webvpn)# no filter
```

The **none** keyword indicates that there is no **webvpntype** ACL. It sets a null value, thereby disallowing an ACL and prevents inheriting an ACL from another group policy.

The *ACLname* string following the keyword **value** provides the name of the previously configured ACL.

Note Clientless SSL VPN sessions do not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an ACL named *acl_in* for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# filter acl_in
hostname(config-group-webvpn)#
```

Applying a URL List

You can specify a list of URLs to appear on the clientless SSL VPN home page for a group policy. First, you must create one or more named lists by entering the **url-list** command in global configuration mode. To apply a list of servers and URLs for clientless SSL VPN sessions to a particular group policy,

allowing access to the URLs in a list for a specific group policy, use the name of the list or lists you create there with the **url-list** command in group-policy webvpn configuration mode. There is no default URL list.

To remove a list, including a null value created by using the **url-list none** command**,** use the **no** form of this command. The **no** option allows inheritance of a value from another group policy. To prevent inheriting a URL list, use the **url-list none** command. Using the command a second time overrides the previous setting:

```
hostname(config-group-webvpn)# url-list {value name | none} [index]
hostname(config-group-webvpn)# no url-list
```

Table 4-7 shows the **url-list** command parameters and their meanings.

Table 4-7 url-list Command Keywords and Variables

Parameter	Meaning
index	Indicates the display priority on the home page.
none	Sets a null value for url lists. Prevents inheriting a list from a default or specified group policy.
value *name*	Specifies the name of a previously configured list of urls. To configure such a list, use the **url-list** command in global configuration mode.

The following example sets a URL list called FirstGroupURLs for the group policy named FirstGroup and specifies that this should be the first URL list displayed on the homepage:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# url-list value FirstGroupURLs 1
hostname(config-group-webvpn)#
```

Enabling ActiveX Relay for a Group Policy

ActiveX Relay lets a user who has established a Clientless SSL VPN session use the browser to launch Microsoft Office applications. The applications use the session to download and upload Microsoft Office documents. The ActiveX relay remains in force until the Clientless SSL VPN session closes.

To enable or disable ActiveX controls on Clientless SSL VPN sessions, enter the following command in group-policy webvpn configuration mode:

activex-relay {**enable** | **disable**}

To inherit the **activex-relay** command from the default group policy, enter the following command:

no activex-relay

The following commands enable ActiveX controls on clientless SSL VPN sessions associated with a given group policy:

```
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# activex-relay enable
ciscoasa(config-group-webvpn)
```

Enabling Application Access on Clientless SSL VPN Sessions for a Group Policy

To enable application access for this group policy, enter the **port-forward** command in group-policy webvpn configuration mode. Port forwarding is disabled by default.

Before you can enter the **port-forward** command in group-policy webvpn configuration mode to enable application access, you must define a list of applications that you want users to be able to use in a clientless SSL VPN session. Enter the **port-forward** command in global configuration mode to define this list.

To remove the port forwarding attribute from the group-policy configuration, including a null value created by issuing the **port-forward none** command, enter the **no** form of this command. The **no** option allows inheritance of a list from another group policy. To prevent inheriting a port forwarding list, enter the **port-forward** command with the **none** keyword. The **none** keyword indicates that there is no filtering. It sets a null value, thereby disallowing a filtering, and prevents inheriting filtering values.

The syntax of the command is as follows:

```
hostname(config-group-webvpn)# port-forward {value listname | none}
hostname(config-group-webvpn)# no port-forward
```

The *listname* string following the keyword **value** identifies the list of applications users of clientless SSL VPN sessions can access. Enter the port-forward command in webvpn configuration mode to define the list.

Using the command a second time overrides the previous setting.

The following example shows how to set a port-forwarding list called *ports1* for the internal group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup internal attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# port-forward value ports1
hostname(config-group-webvpn)#
```

Configuring the Port-Forwarding Display Name

Configure the display name that identifies TCP port forwarding to end users for a particular user or group policy by using the **port-forward-name** command in group-policy webvpn configuration mode. To delete the display name, including a null value created by using the **port-forward-name none** command**,** enter the **no** form of the command. The **no** option restores the default name, Application Access. To prevent a display name, enter the **port-forward none** command. The syntax of the command is as follows:

```
hostname(config-group-webvpn)# port-forward-name {value name | none}
hostname(config-group-webvpn)# no port-forward-name
```

The following example shows how to set the name, Remote Access TCP Applications, for the internal group policy named *FirstGroup*:

```
ciscoasa(config)# group-policy FirstGroup internal attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# port-forward-name value Remote Access TCP Applications
hostname(config-group-webvpn)#
```

Configuring the Maximum Object Size to Ignore for Updating the Session Timer

Network devices exchange short keepalive messages to ensure that the virtual circuit between them is still active. The length of these messages can vary. The **keep-alive-ignore** command lets you tell the ASA to consider all messages that are less than or equal to the specified size as keepalive messages and not as traffic when updating the session timer. The range is 0 through 900 KB. The default is 4 KB.

To specify the upper limit of the HTTP/HTTPS traffic, per transaction, to ignore, use the **keep-alive-ignore** command in group-policy attributes webvpn configuration mode:

```
ciscoasa(config-group-webvpn)# keep-alive-ignore size
ciscoasa(config-group-webvpn)#
```

```
The no form of the command removes this specification from the configuration:
ciscoasa(config-group-webvpn)# no keep-alive-ignore
ciscoasa(config-group-webvpn)#
```

The following example sets the maximum size of objects to ignore as 5 KB:

```
ciscoasa(config-group-webvpn)# keep-alive-ignore 5
ciscoasa(config-group-webvpn)#
```

Specifying HTTP Compression

Enable compression of http data over a clientless SSL VPN session for a specific group or user by entering the **http-comp** command in the group policy webvpn mode.

```
ciscoasa(config-group-webvpn)# http-comp {gzip | none}
ciscoasa(config-group-webvpn)#
```

To remove the command from the configuration and cause the value to be inherited, use the **no** form of the command:

```
ciscoasa(config-group-webvpn)# no http-comp {gzip | none}
ciscoasa(config-group-webvpn)#
```

The syntax of this command is as follows:

- **gzip—**Specifies compression is enabled for the group or user. This is the default value.
- **none—**Specifies compression is disabled for the group or user.

For clientless SSL VPN sessions, the **compression** command configured from global configuration mode overrides the **http-comp** command configured in group policy and username webvpn modes.

In the following example, compression is disabled for the group-policy sales:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# http-comp none
hostname(config-group-webvpn)#
```

Specifying the SSO Server

Single sign-on support, available only for clientless SSL VPN sessions, lets users access different secure services on different servers without reentering a username and password more than once. The **sso-server value** command, when entered in group-policy-webvpn mode, lets you assign an SSO server to a group policy.

To assign an SSO server to a group policy, use the **sso-server value** command in group-policy-webvpn configuration mode. This command requires that your configuration include CA SiteMinder command.

```
ciscoasa(config-group-webvpn)# sso-server value server_name
ciscoasa(config-group-webvpn)#
```

To remove the assignment and use the default policy, use the **no** form of this command. To prevent inheriting the default policy, use the **sso-server none** command.

```
ciscoasa(config-group-webvpn)# sso-server {value server_name | none}
ciscoasa(config-group-webvpn)# [no] sso-server value server_name
```

The default policy assigned to the SSO server is DfltGrpPolicy.

The following example creates the group policy “my-sso-grp-pol” and assigns it to the SSO server named “example”:

```
ciscoasa(config)# group-policy my-sso-grp-pol internal
ciscoasa(config)# group-policy my-sso-grp-pol attributes
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# sso-server value example
ciscoasa(config-group-webvpn)#
```

Configuring User Attributes

This section describes user attributes and how to configure them. It includes the following sections:

- Viewing the Username Configuration, page 4-87
- Configuring Attributes for Individual Users, page 4-87

By default, users inherit all user attributes from the assigned group policy. The ASA also lets you assign individual attributes at the user level, overriding values in the group policy that applies to that user. For example, you can specify a group policy giving all users access during business hours, but give a specific user 24-hour access.

Viewing the Username Configuration

To display the configuration for all usernames, including default values inherited from the group policy, enter the **all** keyword with the **show running-config username** command, as follows:

```
ciscoasa# show running-config all username
ciscoasa#
```

This displays the encrypted password and the privilege level. for all users, or, if you supply a username, for that specific user. If you omit the **all** keyword, only explicitly configured values appear in this list. The following example displays the output of this command for the user named testuser:

```
ciscoasa# show running-config all username testuser
username testuser password 12RsxXQnphyr/I9Z encrypted privilege 15
```

Configuring Attributes for Individual Users

To configure specific users, you assign a password (or no password) and attributes to a user using the **username** command, which enters username mode. Any attributes that you do not specify are inherited from the group policy.

The internal user authentication database consists of the users entered with the **username** command. The **login** command uses this database for authentication. To add a user to the ASA database, enter the **username** command in global configuration mode. To remove a user, use the **no** version of this command with the username you want to remove. To remove all usernames, use the **clear configure username** command without appending a username.

Setting a User Password and Privilege Level

Enter the **username** command to assign a password and a privilege level for a user. You can enter the **nopassword** keyword to specify that this user does not require a password. If you do specify a password, you can specify whether that password is stored in an encrypted form.

The optional **privilege** keyword lets you set a privilege level for this user. Privilege levels range from 0 (the lowest) through 15. System administrators generally have the highest privilege level. The default level is 2.

```
ciscoasa(config)# username name {nopassword | password password [encrypted]} [privilege
priv_level]}

ciscoasa(config)# no username [name]
```

Table 4-8 describes the meaning of the keywords and variables used in this command.

Table 4-8 username Command Keywords and Variables

Keyword/Variable	Meaning
encrypted	Indicates that the password is encrypted.
name	Provides the name of the user.
nopassword	Indicates that this user needs no password.
password *password*	Indicates that this user has a password, and provides the password.
privilege *priv_level*	Sets a privilege level for this user. The range is from 0 to 15, with lower numbers having less ability to use commands and administer the ASA. The default privilege level is 2. The typical privilege level for a system administrator is 15.

By default, VPN users that you add with this command have no attributes or group policy association. You must explicitly configure all values.

The following example shows how to configure a user named anyuser with an encrypted password of pw_12345678 and a privilege level of 12:

```
ciscoasa(config)# username anyuser password pw_12345678 encrypted privilege 12
ciscoasa(config)#
```

Configuring User Attributes

After configuring the user's password (if any) and privilege level, you set the other attributes. These can be in any order. To remove any attribute-value pair, enter the **no** form of the command.

Enter username mode by entering the **username** command with the **attributes** keyword:

```
ciscoasa(config)# username name attributes
ciscoasa(config-username)#
```

The prompt changes to indicate the new mode. You can now configure the attributes.

Configuring VPN User Attributes

The VPN user attributes set values specific to VPN connections, as described in the following sections.

Configuring Inheritance

You can let users inherit from the group policy the values of attributes that you have not configured at the username level. To specify the name of the group policy from which this user inherits attributes, enter the **vpn-group-policy** command. By default, VPN users have no group-policy association:

```
ciscoasa(config-username)# vpn-group-policy group-policy-name
ciscoasa(config-username)# no vpn-group-policy group-policy-name
```

For an attribute that is available in username mode, you can override the value of an attribute in a group policy for a particular user by configuring it in username mode.

The following example shows how to configure a user named anyuser to use attributes from the group policy named FirstGroup:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-group-policy FirstGroup
ciscoasa(config-username)#
```

Configuring Access Hours

Associate the hours that this user is allowed to access the system by specifying the name of a configured time-range policy:

To remove the attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a time-range value from another group policy. To prevent inheriting a value, enter the **vpn-access-hours none** command. The default is unrestricted access.

```
ciscoasa(config-username)# vpn-access-hours value {time-range | none}
ciscoasa(config-username)# vpn-access-hours value none
ciscoasa(config)#
```

The following example shows how to associate the user named anyuser with a time-range policy called 824:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-access-hours 824
ciscoasa(config-username)#
```

Configuring Maximum Simultaneous Logins

Specify the maximum number of simultaneous logins allowed for this user. The range is 0 through 2147483647. The default is 3 simultaneous logins. To remove the attribute from the running configuration, enter the **no** form of this command. Enter 0 to disable login and prevent user access.

```
ciscoasa(config-username)# vpn-simultaneous-logins integer
ciscoasa(config-username)# no vpn-simultaneous-logins
ciscoasa(config-username)# vpn-session-timeout alert-interval none
```

Note While the maximum limit for the number of simultaneous logins is very large, allowing several could compromise security and affect performance.

The following example shows how to allow a maximum of 4 simultaneous logins for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-simultaneous-logins 4
ciscoasa(config-username)#
```

Configuring the Idle Timeout

Specify the idle timeout period in minutes, or enter **none** to disable the idle timeout. If there is no communication activity on the connection in this period, the ASA terminates the connection. You can optionally set the alert interval, or leave the default of one minute.

The range is 1 through 35791394 minutes. The default is 30 minutes. To allow an unlimited timeout period, and thus prevent inheriting a timeout value, enter the **vpn-idle-timeout** command with the **none** keyword. To remove the attribute from the running configuration, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-idle-timeout {minutes | none} alert-interval {minutes}
ciscoasa(config-username)# no vpn-idle-timeout alert-interval
ciscoasa(config-username)# vpn-idle-timeout alert-interval none
```

The following example shows how to set a VPN idle timeout of 15 minutes and alert interval of 3 minutes for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-idle-timeout 30 alert-interval 3
ciscoasa(config-username)#
```

Configuring the Maximum Connect Time

Specify the maximum user connection time in minutes, or enter **none** to allow unlimited connection time and prevent inheriting a value for this attribute. At the end of this period of time, the ASA terminates the connection. You can optionally set the alert interval, or leave the default of one minute.

The range is 1 through 35791394 minutes. There is no default timeout. To allow an unlimited timeout period, and thus prevent inheriting a timeout value, enter the **vpn-session-timeout** command with the **none** keyword. To remove the attribute from the running configuration, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-session-timeout {minutes | none} alert-interval {minutes}
ciscoasa(config-username)# no vpn-session-timeout alert-interval
ciscoasa(config-username)#
```

The following example shows how to set a VPN session timeout of 180 minutes for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-session-timeout 180 alert-interval {minutes}
ciscoasa(config-username)#
```

Applying an ACL Filter

Specify the name of a previously-configured, user-specific ACL to use as a filter for VPN connections. To disallow an ACL and prevent inheriting an ACL from the group policy, enter the **vpn-filter** command with the none keyword. To remove the ACL, including a null value created by issuing the **vpn-filter none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from the group policy. There are no default behaviors or values for this command.

You configure ACLs to permit or deny various types of traffic for this user. You then use the **vpn-filter** command to apply those ACLs.

```
ciscoasa(config-username)# vpn-filter {value ACL_name | none}
ciscoasa(config-username)# no vpn-filter
```

```
ciscoasa(config-username)#
```

Note Clientless SSL VPN does not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an ACL named acl_vpn for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-filter value acl_vpn
ciscoasa(config-username)#
```

Specifying the IPv4 Address and Netmask

Specify the IP address and netmask to assign to a particular user. To remove the IP address, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-framed-ip-address {ip_address}
ciscoasa(config-username)# no vpn-framed-ip-address
ciscoasa(config-username)
```

The following example shows how to set an IP address of 10.92.166.7 for a user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-framed-ip-address 10.92.166.7
ciscoasa(config-username)
```

Specify the network mask to use with the IP address specified in the previous step. If you used the **no vpn-framed-ip-address** command, do not specify a network mask. To remove the subnet mask, enter the **no** form of this command. There is no default behavior or value.

```
ciscoasa(config-username)# vpn-framed-ip-netmask {netmask}
ciscoasa(config-username)# no vpn-framed-ip-netmask
ciscoasa(config-username)
```

The following example shows how to set a subnet mask of 255.255.255. 254 for a user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-framed-ip-netmask 255.255.255.254
ciscoasa(config-username)
```

Specifying the IPv6 Address and Netmask

Specify the IPv6 address and netmask to assign to a particular user. To remove the IP address, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-framed-ipv6-address {ip_address}
ciscoasa(config-username)# no vpn-framed-ipv6-address
ciscoasa(config-username)
```

The following example shows how to set an IP address and netmask of 2001::3000:1000:2000:1/64 for a user named anyuser. This address indicates a prefix value of 2001:0000:0000:0000 and an interface ID of 3000:1000:2000:1.

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-framed-ipv6-address 2001::3000:1000:2000:1/64
ciscoasa(config-username)
```

Specifying the Tunnel Protocol

Specify the VPN tunnel types (IPsec or clientless SSL VPN) that this user can use. The default is taken from the default group policy, the default for which is IPsec. To remove the attribute from the running configuration, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-tunnel-protocol {webvpn | IPsec}
ciscoasa(config-username)# no vpn-tunnel-protocol [webvpn | IPsec]
ciscoasa(config-username)
```

The parameter values for this command are as follows:

- **IPsec—**Negotiates an IPsec tunnel between two peers (a remote access client or another secure gateway). Creates security associations that govern authentication, encryption, encapsulation, and key management.
- **webvpn—**Provides clientless SSL VPN access to remote users via an HTTPS-enabled web browser, and does not require a client

Enter this command to configure one or more tunneling modes. You must configure at least one tunneling mode for users to connect over a VPN tunnel.

The following example shows how to configure clientless SSL VPN and IPsec tunneling modes for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-tunnel-protocol webvpn
ciscoasa(config-username)# vpn-tunnel-protocol IPsec
ciscoasa(config-username)
```

Restricting Remote User Access

Configure the **group-lock** attribute with the **value** keyword to restrict remote users to access only through the specified, preexisting connection profile. Group-lock restricts users by checking whether the group configured in the VPN client is the same as the connection profile to which the user is assigned. If it is not, the ASA prevents the user from connecting. If you do not configure group-lock, the ASA authenticates users without regard to the assigned group.

To remove the **group-lock** attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value from the group policy. To disable group-lock, and to prevent inheriting a group-lock value from a default or specified group policy, enter the **group-lock** command with the **none** keyword.

```
ciscoasa(config-username)# group-lock {value tunnel-grp-name | none}
ciscoasa(config-username)# no group-lock
ciscoasa(config-username)
```

The following example shows how to set group lock for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# group-lock value tunnel-group-name
ciscoasa(config-username)
```

Enabling Password Storage for Software Client Users

Specify whether to let users store their login passwords on the client system. Password storage is disabled by default. Enable password storage only on systems that you know to be in secure sites. To disable password storage, enter the **password-storage** command with the **disable** keyword. To remove the password-storage attribute from the running configuration, enter the **no** form of this command. This enables inheritance of a value for password-storage from the group policy.

```
ciscoasa(config-username)# password-storage {enable | disable}
ciscoasa(config-username)# no password-storage
ciscoasa(config-username)
```

This command has no bearing on interactive hardware client authentication or individual user authentication for hardware clients.

The following example shows how to enable password storage for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# password-storage enable
ciscoasa(config-username)
```

Configuring Clientless SSL VPN Access for Specific Users

The following sections describe how to customize a configuration for specific users of clientless SSL VPN sessions. Enter username webvpn configuration mode by using the **webvpn** command in username configuration mode. Clientless SSL VPN lets users establish a secure, remote-access VPN tunnel to the ASA using a web browser. There is no need for either a software or hardware client. Clientless SSL VPN provides easy access to a broad range of web resources and web-enabled applications from almost any computer that can reach HTTPS Internet sites. Clientless SSL VPN uses SSL and its successor, TLS1, to provide a secure connection between remote users and specific, supported internal resources that you configure at a central site. The ASA recognizes connections that need to be proxied, and the HTTP server interacts with the authentication subsystem to authenticate users.

The username webvpn configuration mode commands define access to files, URLs and TCP applications over clientless SSL VPN sessions. They also identify ACLs and types of traffic to filter. Clientless SSL VPN is disabled by default. These **webvpn** commands apply only to the username from which you configure them. Notice that the prompt changes, indicating that you are now in username webvpn configuration mode.

```
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)#
```

To remove all commands entered in username webvpn configuration mode, use the **no** form of this command:

```
ciscoasa(config-username)# no webvpn
ciscoasa(config-username)#
```

You do not need to configure clientless SSL VPN to use e-mail proxies.

Note The webvpn mode that you enter from global configuration mode lets you configure global settings for clientless SSL VPN sessions. The username webvpn configuration mode described in this section, which you enter from username mode, lets you customize the configuration of specific users specifically for clientless SSL VPN sessions.

In username webvpn configuration mode, you can customize the following parameters, each of which is described in the subsequent steps:

- customizations
- deny message
- html-content-filter
- homepage
- filter
- url-list
- port-forward
- port-forward-name
- sso server (single-signon server)
- auto-signon
- AnyConnect Secure Mobility Client
- keep-alive ignore
- HTTP compression

The following example shows how to enter username webvpn configuration mode for the username anyuser attributes:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)#
```

Specifying the Content/Objects to Filter from the HTML

To filter Java, ActiveX, images, scripts, and cookies for clientless SSL VPN sessions for this user, enter the **html-content-filter** command in username webvpn configuration mode. To remove a content filter, enter the **no** form of this command. To remove all content filters, including a null value created by issuing the **html-content-filter none** command, enter the **no** form of this command without arguments. The **no** option allows inheritance of a value from the group policy. To prevent inheriting an HTML content filter, enter the **html-content-filter none** command. HTML filtering is disabled by default.

Using the command a second time overrides the previous setting.

```
hostname(config-username-webvpn)# html-content-filter {java | images | scripts | cookies |
none}

hostname(config-username-webvpn)# no html-content-filter [java | images | scripts |
cookies | none]
```

The keywords used in this command are as follows:

- **cookies—**Removes cookies from images, providing limited ad filtering and privacy.
- **images—**Removes references to images (removes <IMG> tags).
- **java—**Removes references to Java and ActiveX (removes <EMBED>, <APPLET>, and <OBJECT> tags.
- **none—**Indicates that there is no filtering. Sets a null value, thereby disallowing filtering. Prevents inheriting filtering values.
- **scripts**—Removes references to scripting (removes <SCRIPT> tags).

The following example shows how to set filtering of JAVA and ActiveX, cookies, and images for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
hostname(config-username-webvpn)# html-content-filter java cookies images
ciscoasa(config-username-webvpn)#
```

Specifying the User Home Page

To specify a URL for the web page that displays when this user logs into clientless SSL VPN session, enter the **homepage** command in username webvpn configuration mode. To remove a configured home page, including a null value created by issuing the **homepage none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from the group policy. To prevent inheriting a home page, enter the **homepage none** command.

The **none** keyword indicates that there is no clientless SSL VPN home page. It sets a null value, thereby disallowing a home page and prevents inheriting a home page.

The *url-string* variable following the keyword **value** provides a URL for the home page. The string must begin with either http:// or https://.

There is no default home page.

```
hostname(config-username-webvpn)# homepage {value url-string | none}
hostname(config-username-webvpn)# no homepage
ciscoasa(config-username-webvpn)#
```

The following example shows how to specify www.example.com as the home page for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
hostname(config-username-webvpn)# homepage value www.example.com
ciscoasa(config-username-webvpn)#
```

Applying Customization

Customizations determine the appearance of the windows that the user sees upon login. You configure the customization parameters as part of configuring clientless SSL VPN. To apply a previously defined web-page customization to change the look-and-feel of the web page that the user sees at login, enter the customization command in username webvpn configuration mode:

```
ciscoasa(config-username-webvpn)# customization {none | value customization_name}
ciscoasa(config-username-webvpn)#
```

For example, to use the customization named blueborder, enter the following command:

```
ciscoasa(config-username-webvpn)# customization value blueborder
ciscoasa(config-username-webvpn)#
```

You configure the customization itself by entering the **customization** command in webvpn mode.

The following example shows a command sequence that first establishes a customization named 123 that defines a password prompt. The example then defines a tunnel-group named test and uses the **customization** command to specify the use of the customization named 123:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# customization 123
ciscoasa(config-webvpn-custom)# password-prompt Enter password
```

```
ciscoasa(config-webvpn)# exit
ciscoasa(config)# username testuser nopassword
ciscoasa(config)# username testuser attributes
ciscoasa(config-username-webvpn)# webvpn
ciscoasa(config-username-webvpn)# customization value 123
ciscoasa(config-username-webvpn)#
```

Specifying a "Deny" Message

You can specify the message delivered to a remote user who logs into clientless SSL VPN session successfully, but has no VPN privileges by entering the **deny-message** command in username webvpn configuration mode:

```
ciscoasa(config-username-webvpn)# deny-message value "message"
ciscoasa(config-username-webvpn)# no deny-message value "message"
ciscoasa(config-username-webvpn)# deny-message none
```

The **no deny-message value** command removes the message string, so that the remote user does not receive a message.

The **no deny-message none** command removes the attribute from the connection profile policy configuration. The policy inherits the attribute value.

The message can be up to 491 alphanumeric characters long, including special characters, spaces, and punctuation, but not counting the enclosing quotation marks. The text appears on the remote user's browser upon login. When typing the string in the **deny-message value** command, continue typing even if the command wraps.

The default deny message is: "Login was successful, but because certain criteria have not been met or due to some specific group policy, you do not have permission to use any of the VPN features. Contact your IT administrator for more information."

The first command in the following example enters username mode and configures the attributes for the user named anyuser. The subsequent commands enter username webvpn configuration mode and modify the deny message associated with that user.

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# deny-message value "Your login credentials are OK.
However, you have not been granted rights to use the VPN features. Contact your
administrator for more information."
ciscoasa(config-username-webvpn)
```

Specifying the ACL for Clientless SSL VPN Sessions

To specify the name of the ACL to use for clientless SSL VPN sessions for this user, enter the **filter** command in username webvpn configuration mode. To remove the ACL, including a null value created by issuing the **filter none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from the group policy. To prevent inheriting filter values, enter the **filter value none** command.

Clientless SSL VPN ACLs do not apply until you enter the **filter** command to specify them.

You configure ACLs to permit or deny various types of traffic for this user. You then enter the **filter** command to apply those ACLs for clientless SSL VPN traffic.

```
hostname(config-username-webvpn)# filter {value ACLname | none}
hostname(config-username-webvpn)# no filter
ciscoasa(config-username-webvpn)#
```

The **none** keyword indicates that there is no **webvpntype** ACL. It sets a null value, thereby disallowing an ACL and prevents inheriting an ACL from another group policy.

The *ACLname* string following the keyword **value** provides the name of the previously configured ACL.

Note Clientless SSL VPN does not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an ACL named *acl_in* for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
hostname(config-username-webvpn)# filter acl_in
ciscoasa(config-username-webvpn)#
```

Applying a URL List

You can specify a list of URLs to appear on the home page for a user who has established a clientless SSL VPN session. First, you must create one or more named lists by entering the **url-list** command in global configuration mode. To apply a list of servers and URLs to a particular user of clientless SSL VPN, enter the **url-list** command in username webvpn configuration mode.

To remove a list, including a null value created by using the **url-list none** command**,** enter the **no** form of this command. The **no** option allows inheritance of a value from the group policy. To prevent inheriting a url list, enter the **url-list none** command.

```
hostname(config-username-webvpn)# url-list {listname displayname url | none}
hostname(config-username-webvpn)# no url-list
```

The keywords and variables used in this command are as follows:

- *displayname*—Specifies a name for the URL. This name appears on the portal page in the clientless SSL VPN session.
- *listname*—Identifies a name by which to group URLs.
- **none**—Indicates that there is no list of URLs. Sets a null value, thereby disallowing a URL list. Prevents inheriting URL list values.
- *url*—Specifies a URL that users of clientless SSL VPN can access.

There is no default URL list.

Using the command a second time overrides the previous setting.

The following example shows how to set a URL list called AnyuserURLs for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
hostname(config-username-webvpn)# url-list value AnyuserURLs
ciscoasa(config-username-webvpn)#
```

Enabling ActiveX Relay for a User

ActiveX Relay lets a user who has established a Clientless SSL VPN session use the browser to launch Microsoft Office applications. The applications use the session to download and upload Microsoft Office documents. The ActiveX relay remains in force until the Clientless SSL VPN session closes.

To enable or disable ActiveX controls on Clientless SSL VPN sessions, enter the following command in username webvpn configuration mode:

activex-relay {**enable** | **disable**}

To inherit the **activex-relay** command from the group policy, enter the following command:

no activex-relay

The following commands enable ActiveX controls on Clientless SSL VPN sessions associated with a given username:

```
ciscoasa(config-username-policy)# webvpn
ciscoasa(config-username-webvpn)# activex-relay enable
ciscoasa(config-username-webvpn)
```

Enabling Application Access for Clientless SSL VPN Sessions

To enable application access for this user, enter the **port-forward** command in username webvpn configuration mode. Port forwarding is disabled by default.

To remove the port forwarding attribute from the configuration, including a null value created by issuing the **port-forward none** command, enter the **no** form of this command. The **no** option allows inheritance of a list from the group policy. To disallow filtering and prevent inheriting a port forwarding list, enter the **port-forward** command with the **none** keyword.

```
hostname(config-username-webvpn)# port-forward {value listname | none}
hostname(config-username-webvpn)# no port-forward
ciscoasa(config-username-webvpn)#
```

The *listname* string following the keyword **value** identifies the list of applications users of clientless SSL VPN can access. Enter the **port-forward** command in configuration mode to define the list.

Using the command a second time overrides the previous setting.

Before you can enter the **port-forward** command in username webvpn configuration mode to enable application access, you must define a list of applications that you want users to be able to use in a clientless SSL VPN session. Enter the **port-forward** command in global configuration mode to define this list.

The following example shows how to configure a portforwarding list called ports1:

```
ciscoasa(config-group-policy)# webvpn
hostname(config-username-webvpn)# port-forward value ports1
ciscoasa(config-username-webvpn)#
```

Configuring the Port-Forwarding Display Name

Configure the display name that identifies TCP port forwarding to end users for a particular user by using the **port-forward-name** command in username webvpn configuration mode. To delete the display name, including a null value created by using the **port-forward-name none** command**,** enter the **no** form of the command. The **no** option restores the default name, Application Access. To prevent a display name, enter the **port-forward none** command.

```
hostname(config-username-webvpn)# port-forward-name {value name | none}
hostname(config-username-webvpn)# no port-forward-name
```

The following example shows how to configure the port-forward name test:

```
ciscoasa(config-group-policy)# webvpn
hostname(config-username-webvpn)# port-forward-name value test
ciscoasa(config-username-webvpn)#
```

Configuring the Maximum Object Size to Ignore for Updating the Session Timer

Network devices exchange short keepalive messages to ensure that the virtual circuit between them is still active. The length of these messages can vary. The **keep-alive-ignore** command lets you tell the ASA to consider all messages that are less than or equal to the specified size as keepalive messages and not as traffic when updating the session timer. The range is 0 through 900 KB. The default is 4 KB.

To specify the upper limit of the HTTP/HTTPS traffic, per transaction, to ignore, use the **keep-alive-ignore** command in group-policy attributes webvpn configuration mode:

```
ciscoasa(config-group-webvpn)# keep-alive-ignore size
ciscoasa(config-group-webvpn)#

The no form of the command removes this specification from the configuration:
ciscoasa(config-group-webvpn)# no keep-alive-ignore
ciscoasa(config-group-webvpn)#
```

The following example sets the maximum size of objects to ignore as 5 KB:

```
ciscoasa(config-group-webvpn)# keep-alive-ignore 5
ciscoasa(config-group-webvpn)#
```

Configuring Auto-Signon

To automatically submit the login credentials of a particular user of clientless SSL VPN to internal servers using NTLM, basic HTTP authentication or both, use the **auto-signon** command in username webvpn configuration mode.

The **auto-signon** command is a single sign-on method for users of clientless SSL VPN sessions. It passes the login credentials (username and password) to internal servers for authentication using NTLM authentication, basic authentication, or both. Multiple auto-signon commands can be entered and are processed according to the input order (early commands take precedence).

You can use the auto-signon feature in three modes: webvpn configuration, webvpn group configuration, or webvpn username configuration mode. The typical precedence behavior applies where username supersedes group, and group supersedes global. The mode you choose will depend upon the desired scope of authentication.

To disable auto-signon for a particular user to a particular server, use the **no** form of the command with the original specification of IP block or URI. To disable authentication to all servers, use the **no** form without arguments. The **no** option allows inheritance of a value from the group policy.

The following example commands configure auto-signon for a user of clientless SSL VPN named anyuser, using either basic or NTLM authentication, to servers defined by the URI mask https://*.example.com/*:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# auto-signon allow uri https://*.example.com/* auth-type
all

The following example commands configure auto-signon for a user of clientless SSL VPN named
anyuser, using either basic or NTLM authentication, to the server with the IP address
10.1.1.0, using subnet mask 255.255.255.0:

ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# auto-signon allow ip 10.1.1.0 255.255.255.0 auth-type
all
hostname(config-username-webvpn)#
```

Specifying HTTP Compression

Enable compression of http data over a clientless SSL VPN session for a specific user by entering the **http-comp** command in the username webvpn configuration mode.

```
ciscoasa(config-username-webvpn)# http-comp {gzip | none}
ciscoasa(config-username-webvpn)#
```

To remove the command from the configuration and cause the value to be inherited, use the **no** form of the command:

```
ciscoasa(config-username-webvpn)# no http-comp {gzip | none}
ciscoasa(config-username-webvpn)#
```

The syntax of this command is as follows:

- **gzip—**Specifies compression is enabled for the group or user. This is the default value.
- **none—**Specifies compression is disabled for the group or user.

For clientless SSL VPN session, the **compression** command configured from global configuration mode overrides the **http-comp** command configured in group policy and username webvpn modes.

In the following example, compression is disabled for the username testuser:

```
hostname(config)# username testuser internal
hostname(config)# username testuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# http-comp none
hostname(config-username-webvpn)#
```

Specifying the SSO Server

Single sign-on support, available only for clientless SSL VPN sessions, lets users access different secure services on different servers without reentering a username and password more than once. The **sso-server value** command, when entered in username-webvpn mode, lets you assign an SSO server to a user.

To assign an SSO server to a user, use the **sso-server value** command in username-webvpn configuration mode. This command requires that your configuration include CA SiteMinder command.

```
ciscoasa(config-username-webvpn)# sso-server value server_name
ciscoasa(config-username-webvpn)#
```

To remove the assignment and use the default policy, use the **no** form of this command. To prevent inheriting the default policy, use the **sso-server none** command.

```
ciscoasa(config-username-webvpn)# sso-server {value server_name | none}
ciscoasa(config-username-webvpn)# [no] sso-server value server_name
```

The default policy assigned to the SSO server is DfltGrpPolicy.

The following example assigns the SSO server named example to the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# sso-server value example
ciscoasa(config-username-webvpn)#
```

Configuring IP Addresses for VPNs

This chapter describes IP address assignment methods.

IP addresses make internetwork connections possible. They are like telephone numbers: both the sender and receiver must have an assigned number to connect. But with VPNs, there are actually two sets of addresses: the first set connects client and server on the public network. Once that connection is made, the second set connects client and server through the VPN tunnel.

In ASA address management, we are dealing with the second set of IP addresses: those private IP addresses that connect a client with a resource on the private network, through the tunnel, and let the client function as if it were directly connected to the private network. Furthermore, we are dealing only with the private IP addresses that get assigned to clients. The IP addresses assigned to other resources on your private network are part of your network administration responsibilities, not part of VPN management. Therefore, when we discuss IP addresses here, we mean those IP addresses available in your private network addressing scheme that let the client function as a tunnel endpoint.

This chapter includes the following sections:

Configuring an IP Address Assignment Policy

The ASA can use one or more of the following methods for assigning IP addresses to remote access clients. If you configure more than one address assignment method, the ASA searches each of the options until it finds an IP address. By default, all methods are enabled.

- aaa — Retrieves addresses from an external authentication, authorization, and accounting server on a per-user basis. If you are using an authentication server that has IP addresses configured, we recommend using this method. This method is available for IPv4 and IPv6 assignment policies.
- dhcp — Obtains IP addresses from a DHCP server. If you want to use DHCP, you must configure a DHCP server. You must also define the range of IP addresses that the DHCP server can use. This method is available for IPv4 assignment policies.
- **local —** Internally configured address pools are the easiest method of address pool assignment to configure. If you choose local, you must also use the **ip-local-pool** command to define the range of IP addresses to use. This method is available for IPv4 and IPv6 assignment policies.

– Allow the reuse of an IP address so many minutes after it is released—Delays the reuse of an IP address after its return to the address pool. Adding a delay helps to prevent problems firewalls can experience when an IP address is reassigned quickly. By default the ASA does not impose a delay. This configurable element is available for IPv4 assignment policies.

Use one of these methods to specify a way to assign IP addresses to remote access clients.

- Configuring IPv4 Address Assignments at the Command Line
- Configuring IPv6 Address Assignments at the Command Line

Configuring IPv4 Address Assignments at the Command Line

Command	Purpose
vpn-addr-assign {aaa | dhcp | local [reuse-delay *minutes*]} **Example:** ciscoasa(config)# **vpn-addr-assign aaa** **Example:** ciscoasa(config)# **vpn-addr-assign local reuse-delay 180** **Example:** ciscoasa(config)# **no vpn-addr-assign dhcp**	Enables an address assignment method for the ASA to use when assigning IPv4 address to VPN connections. The available methods to obtain an IP address are from a AAA server, DHCP server, or a local address pool. All of these methods are enabled by default. For local IP address pools, you can configure the reuse of an IP address for between 0 and 480 minutes after the IP address has been released. Use the no form of the command to disable an address assignment method.

Configuring IPv6 Address Assignments at the Command Line

Command	Purpose
ipv6-vpn-addr-assign {aaa | local} **Example:** ciscoasa(config)# **ipv6-vpn-addr-assign aaa** **Example:** ciscoasa(config)# **no ipv6-vpn-addr-assign local**	Enables an address assignment method for the ASA to use when assigning IPv6 address to VPN connections. The available methods to obtain an IP address are from a AAA server or a local address pool. Both of these methods are enabled by default. Use the no form of the command to disable an address assignment method.

Modes

The following table shows the modes in which this feature is available:

Firewall Mode		Security Context		
			Multiple	
Routed	Transparent	Single	Context	System
•	—	•	—	—

Viewing Address Assignment Methods

Use one of these methods to view the address assignment method configured on the ASA:

Viewing IPv4 Address Assignments from the Command Line

Command	Purpose

show running-config all vpn-addr-assign
 	
Example:
`ciscoasa(config)# show running-config all vpn-addr-assign`
	Shows the configured address assignment method. Configured address methods could be aaa, dhcp, or local.
	
	`vpn-addr-assign aaa`
	`vpn-addr-assign dhcp`
	`vpn-addr-assign local`

Viewing IPv6 Address Assignments from the Command Line

Command	Purpose

show running-config all ipv6-vpn-addr-assign
 	
Example:
`hostname(config)# show running-config all ipv6-vpn-addr-assign`
	Shows the configured address assignment method. Configured address methods could be aaa or local.
	
	`ipv6-vpn-addr-assign aaa`
	`ipv6-vpn-addr-assign local reuse-delay 0`

Configuring Local IP Address Pools

To configure IPv4 address pools to use for VPN remote access tunnels, enter the **ip local pool** command in global configuration mode. To delete address pools, enter the **no** form of this command.

To configure IPv6 address pools to use for VPN remote access tunnels, enter the **ipv6 local pool** command in global configuration mode. To delete address pools, enter the **no** form of this command.

The ASA uses address pools based on the connection profile or group policy for the connection. The order in which you specify the pools is important. If you configure more than one address pool for a connection profile or group policy, the ASA uses them in the order in which you added them to the ASA.

If you assign addresses from a non-local subnet, we suggest that you add pools that fall on subnet boundaries to make adding routes for these networks easier.

Use one of these methods to configure a local IP address pool:

- Configuring Local IPv4 Address Pools Using CLI, page 5-4
- Configuring Local IPv6 Address Pools Using CLI, page 5-4

Configuring Local IPv4 Address Pools Using CLI

	Command	Purpose
Step 1	**vpn-addr-assign local** **Example:** ciscoasa(config)# **vpn-addr-assign local**	Configures IP address pools as the address assignment method, enter the **vpn-addr-assign** command with the **local** argument. See also Configuring IPv4 Address Assignments at the Command Line, page 5-2.
Step 2	**ip local pool** *poolname first_address—last_address* **mask** *mask* **Example:** ciscoasa(config)# ip local pool **firstpool 10.20.30.40-10.20.30.50** mask **255.255.255.0** **Example:** ciscoasa(config)# no ip local pool **firstpool**	Configures an address pool. The command names the pool, specifies a range of IPv4 addresses and the subnet mask. The first example configures an IP address pool named **firstpool**. The starting address is **10.20.30.40** and the ending address is **10.20.30.50**. The network mask is **255.255.255.0**. The second example deletes the IP address pool named **firstpool**.

Configuring Local IPv6 Address Pools Using CLI

	Command	Purpose
Step 1	**ipv6-vpn-addr-assign local** **Example:** ciscoasa(config)# **ipv6-vpn-addr-assign local**	Configures IP address pools as the address assignment method, enter the ipv6-**vpn-addr-assign** command with the **local** argument. See also Configuring IPv6 Address Assignments at the Command Line, page 5-2.
Step 2	**ipv6 local pool** *pool_name starting_address prefix_length number_of_addresses* **Example:** ciscoasa(config)# **ipv6 local pool** *ipv6pool 2001:DB8::1/32 100* **Example:** ciscoasa(config)# **no ipv6 local pool** *ipv6pool*	Configures an address pool. The command names the pool, identifies the starting IPv6 address, the prefix length in bits, and the number of addresses to use in the range. The first example configures an IP address pool named **ipv6pool**. The starting address is **2001:DB8::1** the prefix length is **32** bits and the number of addresses to use in the pool is **100**. The second example deletes the IP address pool named **ipv6pool**.

Assign Internal Address Pools to Group Policies in ASDM

The Add or Edit Group Policy dialog box lets you specify address pools, tunneling protocols, filters, connection settings, and servers for the internal Network (Client) Access group policy being added or modified. For each of the fields on this dialog box, checking the Inherit check box lets the corresponding setting take its value from the default group policy. Inherit is the default value for all the attributes in this dialog box.

You can configure both IPv4 and IPv6 address pools for the same group policy. If both versions of IP addresses are configured in the same group policy, clients configured for IPv4 will get an IPv4 address, clients configured for IPv6 will get an IPv6 address, and clients configured for both IPv4 and IPv6 addresses will get both an IPv4 and an IPv6 address.

Step 1 Connect to the ASA using ASDM and select **Configuration > Remote Access VPN > Network (Client) Access > Group Policies**.

Step 2 Create a new group policy or the group policy you want to configure with an internal address pool and click **Edit**.

The General attributes pane is selected by default in the group policy dialog.

Step 3 Use the Address Pools field to specify an IPv4 address pool for this group policy. Click Select to add or edit an IPv4 address pool.

Step 4 Use the IPv6 Address Pools field to specify an IPv6 address pools to use for this group policy. Click Select to add or edit a IPv6 address pool.

Step 5 Click **OK**.

Step 6 Click **Apply**.

Configuring AAA Addressing

To use a AAA server to assign addresses for VPN remote access clients, you must first configure a AAA server or server group. See the **aaa-server protocol** command in the command reference.

In addition, the user must match a connection profile configured for RADIUS authentication.

The following examples illustrate how to define a AAA server group called RAD2 for the tunnel group named firstgroup. It includes one more step than is necessary, in that previously you might have named the tunnel group and defined the tunnel group type. This step appears in the following example as a reminder that you have no access to subsequent tunnel-group commands until you set these values.

An overview of the configuration that these examples create follows:

```
ciscoasa(config)# vpn-addr-assign aaa
ciscoasa(config)# tunnel-group firstgroup type ipsec-ra
ciscoasa(config)# tunnel-group firstgroup general-attributes
ciscoasa(config-general)# authentication-server-group RAD2
```

To configure AAA for IP addressing, perform the following steps:

Step 1 To configure AAA as the address assignment method, enter the **vpn-addr-assign** command with the **aaa** argument:

```
ciscoasa(config)# vpn-addr-assign aaa
```

```
ciscoasa(config)#
```

Step 2 To establish the tunnel group called firstgroup as a remote access or LAN-to-LAN tunnel group, enter the **tunnel-group** command with the **type** keyword. The following example configures a remote access tunnel group.

```
ciscoasa(config)# tunnel-group firstgroup type ipsec-ra
ciscoasa(config)#
```

Step 3 To enter general-attributes configuration mode, which lets you define a AAA server group for the tunnel group called firstgroup, enter the **tunnel-group** command with the **general-attributes** argument.

```
ciscoasa(config)# tunnel-group firstgroup general-attributes
ciscoasa(config-general)#
```

Step 4 To specify the AAA server group to use for authentication, enter the **authentication-server-group** command.

```
ciscoasa(config-general)# authentication-server-group RAD2
ciscoasa(config-general)#
```

This command has more arguments that this example includes. For more information, see the command reference.

Configuring DHCP Addressing

To use DHCP to assign addresses for VPN clients, you must first configure a DHCP server and the range of IP addresses that the DHCP server can use. Then you define the DHCP server on a connection profile basis. Optionally, you can also define a DHCP network scope in the group policy associated with a connection profile or username. This is either an IP network number or IP Address that identifies to the DHCP server which pool of IP addresses to use.

The following examples define the DHCP server at IP address 172.33.44.19 for the connection profile named **firstgroup**. They also define a DHCP network scope of 192.86.0.0 for the group policy called **remotegroup**. (The group policy called remotegroup is associated with the connection profile called firstgroup). If you do not define a network scope, the DHCP server assigns IP addresses in the order of the address pools configured. It goes through the pools until it identifies an unassigned address.

The following configuration includes more steps than are necessary, in that previously you might have named and defined the connection profile type as remote access, and named and identified the group policy as internal or external. These steps appear in the following examples as a reminder that you have no access to subsequent tunnel-group and group-policy commands until you set these values.

Guidelines and Limitations

You can only use an IPv4 address to identify a DHCP server to assign client addresses.

Configuring DHCP Addressing Using the CLI

	Command	Purpose
Step 1	`vpn-addr-assign` **`dhcp`**	Configures IP address pools as the address assignment method. Enter the **vpn-addr-assign** command with the **dhcp** argument. See also Configuring IPv4 Address Assignments at the Command Line, page 5-2.
Step 2	`tunnel-group` **`firstgroup`** `type` **`remote-access`**	Establishes the connection profile called firstgroup as a remote access connection profile. Enter the **tunnel-group** command with the **type** keyword and **remote-access** argument.
Step 3	`tunnel-group firstgroup` **`general-attributes`**	Enters the general-attributes configuration mode for the connection profile so that you can configure a DHCP server. Enter the **tunnel-group** command with the **general-attributes** argument.
Step 4	`dhcp-server` *`IPv4_address_of_DHCP_server`* **Example:** `ciscoasa(config-general)# dhcp-server` **`172.33.44.19`** `ciscoasa(config-general)#`	Defines the DHCP server by IPv4 address. You can not define a DHCP server by an IPv6 address. You can specify more than one DHCP server address for a connection profile. Enter the **dhcp-server** command. This command will allow you to configure the ASA to send additional options to the specified DHCP servers when it is trying to get IP addresses for VPN clients. See the **dhcp-server** command in the *Cisco Security Appliance Command Reference* guide for more information. The example configures a DHCP server at IP address 172.33.44.19.
Step 5	`hostname(config-general)#` **`exit`** `ciscoasa(config)#`	Exit tunnel-group mode.
Step 6	`ciscoasa(config)# group-policy remotegroup` **`internal`**	Creates an internal group policy called **remotegroup**. Enter the **group-policy** command with the **internal** argument to make an internal group policy. The example configures an internal group.

	Command	Purpose

Step 7	`ciscoasa(config)# group-policy remotegroup attributes`
	
	Example:
	`hostname(config)# group-policy remotegroup attributes`
	`ciscoasa(config-group-policy)#`
		(Optional) Enters group-policy attributes configuration mode, which lets you configure a subnetwork of IP addresses for the DHCP server to use.
		
		Enter the **group-policy** command with the **attributes** keyword.
		
		The example enters group policy attributes configuration mode for **remotegroup** group-policy.

Step 8	`hostname(config-group-policy)# dhcp-network-scope 192.86.0.0` `ciscoasa(config-group-policy)#`	(Optional) To specify the range of IP addresses the DHCP server should use to assign addresses to users of the group policy called **remotegroup**, enter the **dhcp-network-scope** command. The example configures a network scope of 192.86.0.0. **Note** The dhcp-network-scope must be a routable IP address and not the subset of the DHCP pool. The DHCP server determines which subnet this IP address belongs to and assigns an IP address from that pool. Cisco recommends that you use an interface of the ASA as a dhcp-network-scope for routing reasons. You can use any IP address as the dhcp-network-scope, but it may require that static routes be added to the network.

Example

A summary of the configuration that these examples create follows:

```
ciscoasa(config)# vpn-addr-assign dhcp
ciscoasa(config)# tunnel-group firstgroup type remote-access
ciscoasa(config)# tunnel-group firstgroup general-attributes
ciscoasa(config-general)# dhcp-server 172.33.44.19
hostname(config-general)# exit
ciscoasa(config)# group-policy remotegroup internal
hostname(config)# group-policy remotegroup attributes
hostname(config-group-policy)# dhcp-network-scope 192.86.0.0
```

Assigning IP Addresses to Local Users

Local user accounts can be configured to use a group policy, and some AnyConnect attributes can also be configured. These user accounts provide fallback if the other sources of IP address fail, so administrators will still have access.

This section describes how to configure all the attributes of a local user.

Prerequisites

This procedure describes how to edit an existing user. To add a user select **Configuration > Remote Access VPN > AAA/Local Users > Local Users** and click **Add**. For more information see "Adding a User Account to the Local Database" in Chapter 42, Configuring AAA Servers and the Local Database in the *Cisco ASA 5500 Configuration Guide Using ASDM*.

User Edits

By default, the **Inherit** check box is checked for each setting on the Edit User Account screen, which means that the user account inherits the value of that setting from the default group policy, DfltGrpPolicy.

To override each setting, uncheck the **Inherit** check box, and enter a new value. The detailed steps that follow describe each of the settings on the Edit User Account screen.

Detailed Steps

Step 1 Start ASDM and select **Configuration > Remote Access VPN > AAA/Local Users > Local Users**.

Step 2 Select the user you want to configure and click **Edit**.

The Edit User Account screen opens.

Step 3 In the left pane, click **VPN Policy**.

Step 4 Specify a group policy for the user. The user policy will inherit the attributes of this group policy. If there are other fields on this screen that are set to **Inherit** the configuration from the Default Group Policy, the attributes specified in this group policy will take precedence over those in the Default Group Policy.

Step 5 Specify which tunneling protocols are available for the user, or whether the value is inherited from the group policy. Check the desired **Tunneling Protocols** check boxes to choose the VPN tunneling protocols that are available for use. Only the selected protocols are available for use. The choices are as follows:

- Clientless SSL VPN (VPN via SSL/TLS) uses a web browser to establish a secure remote-access tunnel to a VPN Concentrator; requires neither a software nor hardware client. Clientless SSL VPN can provide easy access to a broad range of enterprise resources, including corporate websites, web-enabled applications, NT/AD file shares (web-enabled), e-mail, and other TCP-based applications from almost any computer that can reach HTTPS Internet sites.
- The SSL VPN Client lets users connect after downloading the Cisco AnyConnect Client application. Users use a clientless SSL VPN connection to download this application the first time. Client updates then occur automatically as needed whenever the user connects.
- IPsec IKEv1—IP Security Protocol. Regarded as the most secure protocol, IPsec provides the most complete architecture for VPN tunnels. Both Site-to-Site (peer-to-peer) connections and Cisco VPN client-to-LAN connections can use IPsec IKEv1.
- IPsec IKEv2—IPsec IKEv2-Supported by the AnyConnect Secure Mobility Client. AnyConnect connections using IPsec with IKEv2 can make use of the same feature set available to SSL VPN Connections.
- L2TP over IPsec allows remote users with VPN clients provided with several common PC and mobile PC operating systems to establish secure connections over the public IP network to the ASA and private corporate networks.

Note If no protocol is selected, an error message appears.

Step 6 Specify which filter (IPv4 or IPv6) to use, or whether to inherit the value from the group policy. Filters consist of rules that determine whether to allow or reject tunneled data packets coming through the ASA, based on criteria such as source address, destination address, and protocol. To configure filters and rules, choose **Configuration > Remote Access VPN > Network (Client) Access > Group Policies > Add/Edit > General > More Options > Filter.**

Click **Manage** to display the ACL Manager pane, on which you can add, edit, and delete ACLs and ACEs.

Step 7 Specify whether to inherit the Connection Profile (tunnel group) lock or to use the selected tunnel group lock, if any. Selecting a specific lock restricts users to remote access through this group only. Tunnel Group Lock restricts users by checking if the group configured in the VPN client is the same as the users assigned group. If it is not, the ASA prevents the user from connecting. If the Inherit check box is not checked, the default value is None.

Step 8 Specify whether to inherit the Store Password on Client System setting from the group. Uncheck the **Inherit** check box to activate the Yes and No radio buttons. Click **Yes** to store the logon password on the client system (potentially a less-secure option). Click **No** (the default) to require the user to enter the password with each connection. For maximum security, we recommend that you *not allow* password storage.

Step 9 Specify an Access Hours policy to apply to this user, create a new access hours policy for the user, or leave the Inherit box checked. The default value is Inherit, or, if the Inherit check box is not checked, the default value is Unrestricted.

Click **Manage** to open the Add Time Range dialog box, in which you can specify a new set of access hours.

Step 10 Specify the number of simultaneous logons by the user. The Simultaneous logons parameter specifies the maximum number of simultaneous logons allowed for this user. The default value is 3. The minimum value is 0, which disables logon and prevents user access.

Note While there is no maximum limit, allowing several simultaneous connections could compromise security and affect performance.

Step 11 Specify the **maximum connection time** for the user connection time in minutes. At the end of this time, the system terminates the connection. The minimum is 1 minute, and the maximum is 2147483647 minutes (over 4000 years). To allow unlimited connection time, check the **Unlimited** check box (the default).

Step 12 Specify the Idle Timeout for the user in minutes. If there is no communication activity on the connection by this user in this period, the system terminates the connection. The minimum time is 1 minute, and the maximum time is 10080 minutes. This value does not apply to users of clientless SSL VPN connections.

Step 13 Configure the Session Alert Interval. If you uncheck the Inherit check box, the Default checkbox is checked automatically. This sets the session alert interval to 30 minutes. If you want to specify a new value, uncheck the Default check box and specify a session alert interval from 1 to 30 minutes in the minutes box.

Step 14 Configure the Idle Alert Interval. If you uncheck the Inherit check box, the Default checkbox is checked automatically. This sets the idle alert interval to 30 minutes. If you want to specify a new value, uncheck the Default check box and specify a session alert interval from 1 to 30 minutes in the minutes box.

Step 15 To set a dedicated IPv4 address for this user, enter an IPv4 address and subnet mask in the Dedicated IPv4 Address (Optional) area.

Step 16 To set a dedicated IPv6 address for this user, enter an IPv6 address with an IPv6 prefix in the Dedicated IPv6 Address (Optional) field. The IPv6 prefix indicates the subnet on which the IPv6 address resides.

Step 17 To configure clientless SSL settings, in the left pane, click **Clientless SSL VPN**. To override each setting, uncheck the **Inherit** check box, and enter a new value.

Step 18 Click **Apply**.

The changes are saved to the running configuration.

Configuring Remote Access IPsec VPNs

This chapter describes how to configure Remote Access IPsec VPNs and includes the following sections:

Information About Remote Access IPsec VPNs

Remote access VPNs allow users to connect to a central site through a secure connection over a TCP/IP network such as the Internet. The Internet Security Association and Key Management Protocol, also called IKE, is the negotiation protocol that lets the IPsec client on the remote PC and the ASA agree on how to build an IPsec Security Association. Each ISAKMP negotiation is divided into two sections called Phase1 and Phase2.

Phase 1 creates the first tunnel to protect later ISAKMP negotiation messages. Phase 2 creates the tunnel that protects data travelling across the secure connection.

To set the terms of the ISAKMP negotiations, you create an ISAKMP policy. It includes the following:

- An authentication method, to ensure the identity of the peers.
- An encryption method, to protect the data and ensure privacy.
- A Hashed Message Authentication Codes (HMAC) method to ensure the identity of the sender and to ensure that the message has not been modified in transit.
- A Diffie-Hellman group to set the size of the encryption key.
- A time limit for how long the ASA uses an encryption key before replacing it.

A transform set combines an encryption method and an authentication method. During the IPsec security association negotiation with ISAKMP, the peers agree to use a particular transform set to protect a particular data flow. The transform set must be the same for both peers.

A transform set protects the data flows for the ACL specified in the associated crypto map entry. You can create transform sets in the ASA configuration, and then specify a maximum of 11 of them in a crypto map or dynamic crypto map entry. For more overview information, including a table that lists valid encryption and authentication methods, see the "Creating an IKEv1 Transform Set" section on page 10-6 in Chapter 10, "Configuring LAN-to-LAN IPsec VPNs" of this guide.

You can configure the ASA to assign an IPv4 address, an IPv6 address, or both an IPv4 and an IPv6 address to an AnyConnect client by creating internal pools of addresses on the ASA or by assigning a dedicated address to a local user on the ASA.

The endpoint must have the dual-stack protocol implemented in its operating system to be assigned both types of addresses. In both scenarios, when no IPv6 address pools are left but IPv4 addresses are available or when no IPv4 address pools are left but IPv6 addresses are available, connection still occurs. The client is not notified; however, so the administrator must look through the ASA logs for the details.

Assigning an IPv6 address to the client is supported for the SSL protocol. This feature is not supported for the IKEv2/IPsec protocol.

Licensing Requirements for Remote Access IPsec VPNs

The following table shows the licensing requirements for this feature:

Note This feature is not available on No Payload Encryption models.

Model	License Requirement[1]
ASA 5505	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license and Security Plus license: 2 sessions. *Optional permanent or time-based licenses: 10 or 25 sessions.* *Shared licenses are not supported.*[2] – AnyConnect Essentials license[3]: 25 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: – Base license: 10 sessions. – Security Plus license: 25 sessions.
ASA 5510	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base and Security Plus license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses*[2]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license and Security Plus license: 250 sessions.

Model	License Requirement[1]
ASA 5520	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 750 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 750 sessions.
ASA 5540	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 2500 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 2500 sessions.
ASA 5550	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.

Model	License Requirement[1]
ASA 5580	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.
ASA 5512-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 250 sessions.
ASA 5515-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 250 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 250 sessions.
ASA 5525-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 750 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 750 sessions.

Model	License Requirement[1]
ASA 5545-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 2500 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 2500 sessions.
ASA 5555-X	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.
ASA 5585-X with SSP-10	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 5000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 5000 sessions.

Model	License Requirement[1]
ASA 5585-X with SSP-20, -40, and -60	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.
ASA SM	• IPsec remote access VPN using IKEv2 (use one of the following): – AnyConnect Premium license: Base license: 2 sessions. *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* *Optional Shared licenses[2]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* – AnyConnect Essentials license[3]: 10000 sessions. • IPsec remote access VPN using IKEv1 and IPsec site-to-site VPN using IKEv1 or IKEv2: Base license: 10000 sessions.

1. The maximum combined VPN sessions of *all* types cannot exceed the maximum sessions shown in this table. For the ASA 5505, the maximum combined sessions is 10 for the Base license, and 25 for the Security Plus license.
2. A shared license lets the security appliance act as a shared license server for multiple client security appliances. The shared license pool is large, but the maximum number of sessions used by each individual security appliance cannot exceed the maximum number listed for permanent licenses.
3. The AnyConnect Essentials license enables AnyConnect VPN client access to the security appliance. This license does not support browser-based SSL VPN access or Cisco Secure Desktop. For these features, activate an AnyConnect Premium license instead of the AnyConnect Essentials license.

   Note: With the AnyConnect Essentials license, VPN users can use a Web browser to log in, and download and start (WebLaunch) the AnyConnect client.

   The AnyConnect client software offers the same set of client features, whether it is enabled by this license or an AnyConnect Premium SSL VPN Edition license.

   The AnyConnect Essentials license cannot be active at the same time as the following licenses on a given security appliance: AnyConnect Premium license (all types) or the Advanced Endpoint Assessment license. You can, however, run AnyConnect Essentials and AnyConnect Premium licenses on different security appliances in the same network.

   By default, the security appliance uses the AnyConnect Essentials license, but you can disable it to use other licenses by using the **no anyconnect-essentials** command.

   For a detailed list of the features supported by the AnyConnect Essentials license and AnyConnect Premium license, see *AnyConnect Secure Mobility Client Features, Licenses, and OSs*:

   http://www.cisco.com/en/US/products/ps10884/products_feature_guides_list.html

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

Supported only in single context mode. Does not support multiple context mode.

Firewall Mode Guidelines

Supported only in routed firewall mode. Transparent mode is not supported.

Failover Guidelines

IPsec VPN sessions are replicated in Active/Standby failover configurations only. Active/Active failover configurations are not supported.

Configuring Remote Access IPsec VPNs

This section describes how to configure remote access VPNs and includes the following topics:

Configuring Interfaces

An ASA has at least two interfaces, referred to here as outside and inside. Typically, the outside interface is connected to the public Internet, while the inside interface is connected to a private network and is protected from public access.

To begin, configure and enable two interfaces on the ASA. Then assign a name, IP address and subnet mask. Optionally, configure its security level, speed and duplex operation on the security appliance.

To configure interfaces, perform the following steps, using the command syntax in the examples:

Detailed Steps

	Command	Purpose
Step 1	**interface** {*interface*} **Example:** hostname(config)# interface ethernet0 hostname(config-if)#	Enters interface configuration mode from global configuration mode.
Step 1	**ip address** *ip_address* [*mask*] [**standby** *ip_address*] **Example:** hostname(config)# interface ethernet0 hostname(config-if)# hostname(config-if)# ip address 10.10.4.200 255.255.0.0	Sets the IP address and subnet mask for the interface.
Step 2	**nameif** *name* **Example:** hostname(config-if)# nameif outside hostname(config-if)#	Specifies a name for the interface (maximum of 48 characters). You cannot change this name after you set it.
Step 3	shutdown **Example:** hostname(config-if)# no shutdown hostname(config-if)#	Enables the interface. By default, interfaces are disabled.

Configuring ISAKMP Policy and Enabling ISAKMP on the Outside Interface

This section describes the procedure to configure an ISAKMP policy on the outside interface and how to enable the policy.

Detailed Steps

Perform the following commands:

	Command	Purpose
Step 1	crypto ikev1 policy *priority* authentication {crack | pre-share | rsa-sig} **Example:** hostname(config)# **crypto ikev1 policy 1** **authentication pre-share** hostname(config)#	Specifies the authentication method and the set of parameters to use during IKEv1 negotiation. *Priority* uniquely identifies the Internet Key Exchange (IKE) policy and assigns a priority to the policy. Use an integer from 1 to 65,534, with 1 being the highest priority and 65,534 the lowest. In this example and the steps that follow, we set the priority to 1.
Step 2	**crypto ikev1** policy *priority* encryption {aes | aes-192 | aes-256 | des | 3des} **Example:** hostname(config)# **crypto ikev1 policy 1** **encryption 3des** hostname(config)#	Specifies the encryption method to use within an IKE policy.

	Command	Purpose
Step 3	**crypto ikev1** policy *priority* hash {md5 | sha} **Example:** hostname(config)# **crypto ikev1 policy 1 hash sha** hostname(config)#	Specifies the hash algorithm for an IKE policy (also called the HMAC variant).
Step 4	**crypto ikev1** policy *priority* group {1 | 2 | 5} **Example:** hostname(config)# **crypto ikev1 policy 1 group 2** hostname(config)#	Specifies the Diffie-Hellman group for the IKE policy—the crypto protocol that allows the IPsec client and the ASA to establish a shared secret key.
Step 5	**crypto ikev1** policy *priority* lifetime {*seconds*} **Example:** hostname(config)# **crypto ikev1 policy 1 lifetime 43200** hostname(config)#	Specifies the encryption key lifetime—the number of seconds each security association should exist before expiring. The range for a finite lifetime is 120 to 2147483647 seconds. Use 0 seconds for an infinite lifetime.
Step 6	**crypto ikev1** enable *interface-name* **Example:** hostname(config)# **crypto ikev1** enable outside hostname(config)#	Enables ISAKMP on the interface named *outside*.
Step 7	write memory **Example:** hostname(config-if)# write memory Building configuration... Cryptochecksum: 0f80bf71 1623a231 63f27ccf 8700ca6d 11679 bytes copied in 3.390 secs (3893 bytes/sec) [OK] hostname(config-if)#	Saves the changes to the configuration.

Configuring an Address Pool

The ASA requires a method for assigning IP addresses to users. This section uses address pools as an example. Use the command syntax in the following examples as a guide.

Command	Purpose
ip local pool *poolname* *first-address—last-address* [mask *mask*] **Example:** hostname(config)# **ip local pool testpool 192.168.0.10-192.168.0.15** hostname(config)#	Creates an address pool with a range of IP addresses, from which the ASA assigns addresses to the clients. The address mask is optional. However, You must supply the mask value when the IP addresses assigned to VPN clients belong to a non-standard network and the data could be routed incorrectly if you use the default mask. A typical example is when the IP local pool contains 10.10.10.0/255.255.255.0 addresses, since this is a Class A network by default. This could cause routing issues when the VPN client needs to access different subnets within the 10 network over different interfaces.

Adding a User

This section shows how to configure usernames and passwords. Use the command syntax in the following examples as a guide.

Command	Purpose
username *name* {nopassword | password *password* [mschap | encrypted | nt-encrypted]} [privilege *priv_level*] **Example:** hostname(config)# **username testuser password 12345678** hostname(config)#	Creates a user, password, and privilege level.

Creating an IKEv1 Transform Set or IKEv2 Proposal

This section shows how to configure a transform set (IKEv1) or proposal (IKEv2), which combines an encryption method and an authentication method.

Perform the following task:

Command	Purpose
To configure an IKEv1 transform set: `crypto ipsec ikev1 transform-set` *transform-set-name encryption-method* [*authentication*] **Example:** `hostname(config)#` **`crypto ipsec transform set FirstSet esp-3des esp-md5-hmac`** `hostname(config)#`	Configures an IKEv1 transform set that specifies the IPsec IKEv1 encryption and hash algorithms to be used to ensure data integrity. Use one of the following values for *encryption*: • **esp-aes** to use AES with a 128-bit key. • **esp-aes-192** to use AES with a 192-bit key. • **esp-aes-256** to use AES with a 256-bit key. • **esp-des** to use 56-bit DES-CBC. • **esp-3des** to use triple DES algorithm. • **esp-null** to not use encryption. Use one of the following values for *authentication*: • **esp-md5-hmac** to use the MD5/HMAC-128 as the hash algorithm. • **esp-sha-hmac** to use the SHA/HMAC-160 as the hash algorithm. • **esp-none** to not use HMAC authentication.
To configure an IKEv2 proposal: **`crypto ipsec ikev2 ipsec-proposal`** *proposal_name* Then: **`protocol`** {**`esp`**} {**`encryption`** {**`des`** | **`3des`** | **`aes`** | **`aes-192`** | **`aes-256`** | **`null`**} | **`integrity`** {**`md5`** | **`sha-1`**} **Example:** `hostname(config)#` **`crypto ipsec ikev2 ipsec-proposal secure_proposal`** `hostname(config-ipsec-proposal)#` **`protocol esp encryption des integrity md5`**	Configures an IKEv2 proposal set that specifies the IPsec IKEv2 protocol, encryption, and integrity algorithms to be used. **esp** specifies the Encapsulating Security Payload (ESP) IPsec protocol (currently the only supported protocol for IPsec). Use one of the following values for *encryption*: • **des** to use 56-bit DES-CBC encryption for ESP. • **3des** (default) to use the triple DES encryption algorithm for ESP. • **aes** to use AES with a 128-bit key encryption for ESP. • **aes-192** to use AES with a 192-bit key encryption for ESP. • **aes-256** to use AES with a 256-bit key encryption for ESP. • **null** to not use encryption for ESP. Use one of the following values for *integrity*: • **md5** specifies the md5 algorithm for the ESP integrity protection. • **sha-1** (default) specifies the Secure Hash Algorithm (SHA) SHA-1, defined in the U.S. Federal Information Processing Standard (FIPS), for ESP integrity protection.

Defining a Tunnel Group

This section describes how to configure a tunnel group, which is a set of records that contain tunnel connection policies. You configure a tunnel group to identify AAA servers, specify connection parameters, and define a default group policy. The ASA stores tunnel groups internally.

There are two default tunnel groups in the ASA system: DefaultRAGroup, which is the default remote-access tunnel group, and DefaultL2Lgroup, which is the default LAN-to-LAN tunnel group. You can change them but not delete them. The ASA uses these groups to configure default tunnel parameters for remote access and LAN-to-LAN tunnel groups when there is no specific tunnel group identified during tunnel negotiation.

Perform the following task:

Detailed Steps

	Command	Purpose
Step 1	`tunnel-group` *name* `type` *type* **Example:** `hostname(config)#` **`tunnel-group testgroup type ipsec-ra`** `hostname(config)#`	Creates an IPsec remote access tunnel-group (also called connection profile).

Step 2	**`tunnel-group`** *name* **`general-attributes`**
	
	Example:
	`hostname(config)# tunnel-group testgroup general-attributes`
	`hostname(config-tunnel-general)#`
		Enters tunnel group general attributes mode where you can enter an authentication method.

Step 3	**`address-pool`** [(*interface name*)] **`address_pool1`** [...address_pool6] **Example:** `hostname(config-general)# address-pool testpool`	Specifies an address pool to use for the tunnel group.

Step 4	**`tunnel-group`** *name* **`ipsec-attributes`**
	
	Example:
	`hostname(config)# tunnel-group testgroup ipsec-attributes`
	`hostname(config-tunnel-ipsec)#`
		Enters tunnel group ipsec attributes mode where you can enter IPsec-specific attributes for IKEv1 connections.

Step 5	**`ikev1 pre-shared-key`** *key*
	
	Example:
	`hostname(config-tunnel-ipsec)#`
	`pre-shared-key 44kkaol59636jnfx`
		(Optional) Configures a pre-shared key (IKEv1 only). The key can be an alphanumeric string from 1-128 characters.
		The keys for the adaptive security appliance and the client must be identical. If a Cisco VPN Client with a different preshared key size tries to connect, the client logs an error message indicating it failed to authenticate the peer.
		Note Configure AAA authentication for IKEv2 using certificates in the tunnel group webvpn-attributes.

Creating a Dynamic Crypto Map

This section describes how to configure dynamic crypto maps, which define a policy template where all the parameters do not have to be configured. These dynamic crypto maps let the ASA receive connections from peers that have unknown IP addresses. Remote access clients fall in this category.

Dynamic crypto map entries identify the transform set for the connection. You also enable reverse routing, which lets the ASA learn routing information for connected clients, and advertise it via RIP or OSPF.

Perform the following task:

Detailed Steps

	Command	Purpose

Step 1	For IKEv1, use this command:
	`crypto dynamic-map` *`dynamic-map-name seq-num`* `set ikev1 transform-set` *`transform-set-name`*
	
	Example:
	`hostname(config)# crypto dynamic-map dyn1 1 set ikev1 transform-set FirstSet`
	`hostname(config)#`
	
	For IKEv2, use this command:
	`crypto dynamic-map` *`dynamic-map-name seq-num`* `set ikev2 ipsec-proposal` *`proposal-name`*
	
	Example:
	`hostname(config)# crypto dynamic-map dyn1 1 set ikev2 ipsec-proposal FirstSet`
	`hostname(config)#`
		Creates a dynamic crypto map and specifies an IKEv1 transform set or IKEv2 proposal for the map.

Step 2	**`crypto dynamic-map`** ***`dynamic-map-name dynamic-seq-num`*** **`set reverse-route`**
	
	Example:
	`hostname(config)# crypto dynamic-map dyn1 1 set reverse route`
	`hostname(config)#`
		(Optional) Enables Reverse Route Injection for any connection based on this crypto map entry.

Creating a Crypto Map Entry to Use the Dynamic Crypto Map

This section describes how to create a crypto map entry that lets the ASA use the dynamic crypto map to set the parameters of IPsec security associations.

In the following examples for this command, the name of the crypto map is *mymap*, the sequence number is 1, and the name of the dynamic crypto map is *dyn1*, which you created in the previous section, "Creating a Dynamic Crypto Map."

Perform the following task:

Detailed Steps

	Command	Purpose

Step 1	`crypto map` *map-name seq-num* `ipsec-isakmp dynamic` *dynamic-map-name*
	
	Example:
	`hostname(config)# crypto map mymap 1 ipsec-isakmp dynamic dyn1`
	`hostname(config)#`
		Creates a crypto map entry that uses a dynamic crypto map.

Step 2	**crypto map** *map-name* **interface** *interface-name*
	
	Example:
	`hostname(config)# crypto map mymap interface outside`
	`hostname(config)#`
		Applies the crypto map to the outside interface.

Saving the Security Appliance Configuration

After performing the preceding configuration tasks, be sure to save your configuration changes as shown in this example:

Command	Purpose

`write memory`
 	
Example:
`hostname(config-if)# write memory`
`Building configuration...`
`Cryptochecksum: 0f80bf71 1623a231 63f27ccf 8700ca6d`
 	
`11679 bytes copied in 3.390 secs (3893 bytes/sec)`
`[OK]`
`hostname(config-if)#`
	Saves the changes to the configuration.

Configuration Examples for Remote Access IPsec VPNs

The following example shows how to configure a remote access IPsec/IKEv1 VPN:

```
hostname(config)# interface ethernet0
hostname(config-if)# ip address 10.10.4.200 255.255.0.0
hostname(config-if)# nameif outside
hostname(config-if)# no shutdown
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)# authentication pre-share
hostname(config-ikev1-policy)# encryption 3des
hostname(config-ikev1-policy)# hash sha
hostname(config-ikev1-policy)# group 2
hostname(config-ikev1-policy)# lifetime 43200
hostname(config)# crypto ikev1 outside
hostname(config)# ip local pool testpool 192.168.0.10-192.168.0.15
hostname(config)# username testuser password 12345678
```

```
hostname(config)# crypto ipsec ikev1 transform set FirstSet esp-3des esp-md5-hmac
hostname(config)# tunnel-group testgroup type remote-access
hostname(config)# tunnel-group testgroup general-attributes
hostname(config-general)# address-pool testpool
hostname(config)# tunnel-group testgroup ipsec-attributes
hostname(config-ipsec)# ikev1 pre-shared-key 44kkaol59636jnfx
hostname(config)# crypto dynamic-map dyn1 1 set ikev1 transform-set FirstSet
hostname(config)# crypto dynamic-map dyn1 1 set reverse-route
hostname(config)# crypto map mymap 1 ipsec-isakmp dynamic dyn1
hostname(config)# crypto map mymap interface outside
hostname(config)# write memory
```

The following example shows how to configure a remote access IPsec/IKEv2 VPN:

```
hostname(config)# interface ethernet0
hostname(config-if)# ip address 10.10.4.200 255.255.0.0
hostname(config-if)# nameif outside
hostname(config-if)# no shutdown
hostname(config)# crypto ikev2 policy 1
hostname(config-ikev2-policy)# group 2
hostname(config-ikev2-policy)# integrity sha
hostname(config-ikev2-policy)# lifetime 43200
hostname(config-ikev2-policy)# prf sha
hostname(config)# crypto ikev2 outside
hostname(config)# ip local pool testpool 192.168.0.10-192.168.0.15
hostname(config)# username testuser password 12345678
hostname(config)# crypto ipsec ikev2 ipsec-proposal FirstSet
hostname(config-ipsec-proposal)# protocol esp encryption 3des aes
hostname(config)# tunnel-group testgroup type remote-access
hostname(config)# tunnel-group testgroup general-attributes
hostname(config-general)# address-pool testpool
hostname(config)# tunnel-group testgroup webvpn-attributes
hostname(config-webvpn)# authentication aaa certificate
hostname(config)# crypto dynamic-map dyn1 1 set ikev2 ipsec-proposal FirstSet
hostname(config)# crypto dynamic-map dyn1 1 set reverse-route
hostname(config)# crypto map mymap 1 ipsec-isakmp dynamic dyn1
hostname(config)# crypto map mymap interface outside
hostname(config)# write memory
```

Feature History for Remote Access VPNs

Table 6-1 lists the release history for this feature.

Table 6-1 Feature History for Feature-1

Feature Name	Releases	Feature Information
Remote access VPNs for IPsec IKEv1 and SSL.	7.0	Remote access VPNs allow users to connect to a central site through a secure connection over a TCP/IP network such as the Internet.
Remote access VPNs for IPsec IKEv2	8.4(1)	Added IPsec IKEv2 support for the AnyConnect Secure Mobility Client.

Configuring Network Admission Control

This chapter includes the following sections:

Information about Network Admission Control

Network Admission Control protects the enterprise network from intrusion and infection from worms, viruses, and rogue applications by performing endpoint compliancy and vulnerability checks as a condition for production access to the network. We refer to these checks as *posture validation.* You can configure posture validation to ensure that the anti-virus files, personal firewall rules, or intrusion protection software on a host with an IPsec or WebVPN session are up-to-date before providing access to vulnerable hosts on the intranet. Posture validation can include the verification that the applications running on the remote hosts are updated with the latest patches. NAC occurs only after user authentication and the setup of the tunnel. NAC is especially useful for protecting the enterprise network from hosts that are not subject to automatic network policy enforcement, such as home PCs.

The establishment of a tunnel between the endpoint and the ASA triggers posture validation.

You can configure the ASA to pass the IP address of the client to an optional audit server if the client does not respond to a posture validation request. The audit server, such as a Trend server, uses the host IP address to challenge the host directly to assess its health. For example, it may challenge the host to determine whether its virus checking software is active and up-to-date. After the audit server completes its interaction with the remote host, it passes a token to the posture validation server, indicating the health of the remote host.

Following successful posture validation or the reception of a token indicating the remote host is healthy, the posture validation server sends a network access policy to the ASA for application to the traffic on the tunnel.

In a *NAC Framework* configuration involving the ASA, only a Cisco Trust Agent running on the client can fulfill the role of posture agent, and only a Cisco Access Control Server (ACS) can fulfill the role of posture validation server. The ACS uses dynamic ACLs to determine the access policy for each client.

As a RADIUS server, the ACS can authenticate the login credentials required to establish a tunnel, in addition to fulfilling its role as posture validation server.

Note Only a NAC Framework policy configured on the ASA supports the use of an audit server.

In its role as posture validation server, the ACS uses access control lists. If posture validation succeeds and the ACS specifies a redirect URL as part of the access policy it sends to the ASA, the ASA redirects all HTTP and HTTPS requests from the remote host to the redirect URL. Once the posture validation server uploads an access policy to the ASA, all of the associated traffic must pass both the Security Appliance and the ACS (or vice versa) to reach its destination.

The establishment of a tunnel between an IPsec or WebVPN client and the ASA triggers posture validation if a NAC Framework policy is assigned to the group policy. The NAC Framework policy can, however, identify operating systems that are exempt from posture validation and specify an optional ACL to filter such traffic.

Licensing Requirements

The following table shows the licensing requirements for this feature:

Note This feature is not available on No Payload Encryption models.

Model	License Requirement[1,2]
ASA 5505	AnyConnect Premium license: • Base License or Security Plus license: 2 sessions. • *Optional permanent or time-based licenses: 10 or 25 sessions.* • *Shared licenses are not supported.*[3]
ASA 5510	AnyConnect Premium license: • Base and Security Plus License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* • *Optional Shared licenses*[3]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5520	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* • *Optional Shared licenses*[3]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*

Model	License Requirement[1,2]
ASA 5540	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5550	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5580	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5512-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5515-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5525-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5545-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*

Model	License Requirement[1,2]
ASA 5555-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5585-X with SSP-10	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5585-X with SSP-20, -40, and -60	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA SM	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*

1. If you start a clientless SSL VPN session and then start an AnyConnect client session from the portal, 1 session is used in total. However, if you start the AnyConnect client first (from a standalone client, for example) and then log into the clientless SSL VPN portal, then 2 sessions are used.
2. The maximum combined VPN sessions of *all* types cannot exceed the maximum sessions shown in this table.
3. A shared license lets the security appliance act as a shared license server for multiple client security appliances. The shared license pool is large, but the maximum number of sessions used by each individual security appliance cannot exceed the maximum number listed for permanent licenses.

Prerequisites for NAC

When configured to support NAC, the ASA functions as a client of a Cisco Secure Access Control Server, requiring that you install a minimum of one Access Control Server on the network to provide NAC authentication services.

Guidelines and Limitations

Following the configuration of one or more Access Control Servers on the network, you must use the **aaa-server** command to name the Access Control Server group. Then follow the instructions in the "Configuring a NAC Policy" procedure on page 7-8.

ASA support for NAC Framework is limited to remote access IPsec and WebVPN client sessions. The NAC Framework configuration supports only single mode.

NAC on the ASA does not support Layer 3 (non-VPN) traffic and IPv6 traffic.

Viewing the NAC Policies on the Security Appliance

Before configuring the NAC policies to be assigned to group policies, we recommend that you view any that may already be set up on the ASA. Because the default configuration does not contain NAC policies, entering this command is a useful way to determine whether anyone has added any. If you, you may decide that policies already configured are suitable and disregard the section on configuring a NAC policy.

Detailed Steps.

	Command	Purpose

Step 1	**show running-config nac-policy**
	
	Example:
	`ciscoasa# show running-config nac-policy`
	`nac-policy nacframework1 nac-framework`
	`default-acl acl-1`
	`reval-period 36000`
	`sq-period 300`
	`exempt-list os "Windows XP" filter acl-2`
	`ciscoasa#`
		Views any NAC policies that are already set up on the ASA.
		
		Shows the configuration of a NAC policy named nac-framework1

Step 2	• default-acl—NAC default ACL applied before posture validation. Following posture validation, the security appliance replaces the default ACL with the one obtained from the Access Control Server for the remote host. The ASA retains the default ACL if posture validation fails. • reval-period—Number of seconds between each successful posture validation in a NAC Framework session. • sq-period—Number of seconds between each successful posture validation in a NAC Framework session and the next query for changes in the host posture. • exempt-list—Operating system names that are exempt from posture validation. Also shows an optional ACL to filter the traffic if the remote computer's operating system matches the name. • authentication-server-group—Name of the of authentication server group to be used for NAC posture validation.	Shows the nac-framework attributes.

	Command	Purpose

Step 3	`show nac-policy`
	
	Example:
	`asa2(config)# show nac-policy`
	`nac-policy framework1 nac-framework`
	`  applied session count = 0`
	`  applied group-policy count = 2`
	`  group-policy list:  GroupPolicy2  GroupPolicy1`
	`nac-policy framework2 nac-framework is not in use.`
	`asa2(config)#`
		Displays the assignment of NAC policies to group policies.
		
		Shows which NAC policies are unassigned and the usage count for each NAC policy.

Step 4	• applied session count—Cumulative number of VPN sessions to which this ASA applied the NAC policy. • applied group-policy count—Cumulative number of group polices to which this ASA applied the NAC policy. • group-policy list—List of group policies to which this NAC policy is assigned. In this case, the usage of a group policy does not determine whether it appears in this list; if the NAC policy is assigned to a group policy in the running configuration, then the group policy appears in this list.	Explains the fields in the show nac-policy command. **Note** When a policy is not assigned to any group policies, "is not in use" displays next to the policy type.

Refer to the following sections to create a NAC policy or modify one that is already present.

Adding, Accessing, or Removing a NAC Policy

Enter the following command to add or modify a NAC policy:

Detailed Steps

	Command	Purpose
Step 1	**global**	Switches to global configuration mode.

Step 2	**nac-policy** *nac-policy-name* **nac-framework**
	
	Example:
	`ciscoasa(config)# nac-policy nac-framework1 nac-framework`
	`ciscoasa(config-nac-policy-nac-framework)`
		Adds or modifies a NAC policy.
		
		nac-policy-name is the name of a new NAC policy or one that is already present. The name is a string of up to 64 characters.
		
		nac-framework specifies that a NAC Framework configuration will provide a network access policy for remote hosts. A Cisco Access Control Server must be present on the network to provide NAC Framework services for the ASA. When you specify this type, the prompt indicates you are in `nac-policy-nac-framework` configuration mode. This mode lets you configure the NAC Framework policy.
		
		Note You can create more than one NAC Framework policy, but you can assign no more than one to a group policy.
		
		Creates and accesses a NAC framework policy named nac-framework1.

Step 3	(Optional) [**no**] **nac-policy** *nac-policy-name* **nac-framework**	Removes a NAC policy from the configuration. You must specify both the name and type of the policy.
Step 4	(Optional) **clear configure nac-policy**	Removes all NAC policies fromthe configuration except for those that are assigned to group policies.
Step 5	**show running-config nac-policy**	Displays the name and configuration of each NAC policy already present on the security appliance.

Configuring a NAC Policy

After you use the **nac-policy** command to name a NAC Framework policy, use the following sections to assign values to its attributes before you assign it to a group policy.

Specifying the Access Control Server Group

You must configure at least one Cisco Access Control Server to support NAC.

Detailed Steps

	Command	Purpose
Step 1	**aaa-server host**	Names the Access Control Server group even if the group contains only one server.

Step 2	(Optional)
	show running-config aaa-server
	
	Example:
	`ciscoasa(config)# show running-config aaa-server`
	`aaa-server acs-group1 protocol radius`
	`aaa-server acs-group1 (outside) host 192.168.22.44`
	`key secret`
	`radius-common-pw secret`
	`ciscoasa(config)#`
		Displays the AAA server configuration.

Step 3	**nac-policy-nac-framework**	Switches to nac-policy-nac-framework configuration mode.

Step 4	**authentication-server-group** *server-group*
	
	Example:
	`ciscoasa(config-nac-policy-nac-framework)#`
	`authentication-server-group acs-group1`
	`ciscoasa(config-nac-policy-nac-framework)`
		Specifies the group used for NAC posture validation.
		server-group must match the server-tag variable specified in the **aaa-server host** command. It is optional if you are using the **no** version of the command.
		
		Specifies acs-group1 as the authentication server group used for NAC posture validation.

Step 5	(Optional) [**no**] **authentication-server-group** *server-group*	Removes the command from the NAC policy.

Setting the Query-for-Posture-Changes Timer

After each successful posture validation, the ASA starts a status query timer. The expiration of this timer triggers a query to the remote host for changes in posture since the last posture validation. A response indicating no change resets the status query timer. A response indicating a change in posture triggers an unconditional posture revalidation. The ASA maintains the current access policy during revalidation.

By default, the interval between each successful posture validation and the status query, and each subsequent status query, is 300 seconds (5 minutes). Follow these steps to change the status query interval:

Detailed Steps

	Command	Purpose
Step 1	**nac-policy-nac-framework**	Switches to nac-policy-nac-framework configuration mode.

Step 2	**sq-period seconds**
	
	Example:
	`ciscoasa(config-group-policy)# sq-period 1800`
	`ciscoasa(config-group-policy)`
		Changes the status query interval.
		seconds must be in the range 30 to 1800 seconds (5 to 30 minutes).
		Changes the query timer to 1800 seconds.

Step 3	(Optional) [**no**] **sq-period** *seconds*	Turns off the status query timer.
Step 4	**show running-config nac-policy**	Displays a 0 next to the sq-period attribute, meaning the timer is turned off.

Setting the Revalidation Timer

After each successful posture validation, the ASA starts a revalidation timer. The expiration of this timer triggers the next unconditional posture validation. The ASA maintains the current access policy during revalidation.

By default, the interval between each successful posture validation is 36000 seconds (10 hours). To change it, enter the following command in nac-policy-nac-framework configuration mode:

Detailed Steps

	Command	Purpose
Step 1	**nac-policy-nac-framework**	Switches to nac-policy-nac-framework.

Step 2	**reval-period** ***seconds***
	
	Example:
	`ciscoasa(config-nac-policy-nac-framework)#`
	reval-period 86400
	`ciscoasa(config-nac-policy-nac-framework)`
		Changes the interval between each successful posture validation.
		seconds must be in the range is 300 to 86400 seconds (5 minutes to 24 hours).

Step 3	(Optional) [**no**] **reval-period** *seconds*	Turns off the status query timer.
Step 4	**show running-config nac-policy**	Displays a 0 next to the sq-period attribute, which means the timer is turned off.

Configuring the Default ACL for NAC

Each group policy points to a default ACL to be applied to hosts that match the policy and are eligible for NAC. The ASA applies the NAC default ACL before posture validation. Following posture validation, the ASA replaces the default ACL with the one obtained from the Access Control Server for the remote host. The ASA retains the default ACL if posture validation fails.

The ASA also applies the NAC default ACL if clientless authentication is enabled (which is the default setting).

Detailed Steps

	Command	Purpose
Step 1	**nac-policy-nac-framework**	Switches to nac-policy-nac-framework configuration mode.
Step 2	**default-acl** ***acl-name***	Specifies which ACL to use as the default ACL for NAC sessions. *acl-name* is the name of the access control list to be applied to the session.
	Example: `ciscoasa(config-nac-policy-nac-framework)#` **default-acl acl-2** `ciscoasa(config-nac-policy-nac-framework)`	Identifies ac1-2 as which ACL to apply before posture validation succeeds.
Step 3	(Optional) [**no**] **default-acl** *acl-name*	Removes the command from the NAC framework policy. Specifying the *acl-name* is optional.

Configuring Exemptions from NAC

The ASA configuration stores a list of exemptions from NAC posture validation. You can specify the operating systems that are exempt. If you specify an ACL, the client running the operating system specified is exempt from posture validation and the client traffic is subject to the ACL.

To add an entry to the list of remote computer types that are exempt from NAC posture validation, enter the following command in nac-policy-nac-framework configuration mode:

Detailed Steps

	Command	Purpose
Step 1	`nac-policy-nac-framework`	Switches to nac-policy-nac-framework configuration mode.
Step 2	**exempt-list os** "*os-name*" [**disable** | **filter** *acl-name* [**disable**]	Adds an entry to the list of remote computer types that are exempt from NAC posture validation. • *os-name* is the operating system name. Use quotation marks if the name includes a space (for example, "Windows XP"). • **filter** applies an ACL to filter the traffic if the computer's operating system matches the *os name*. The **filter**/*acl-name* pair is optional. • **disable** performs one of two functions, as follows: – If you enter it after the "os-name," the ASA ignores the exemption, and applies NAC posture validation to the remote hosts that are running that operating system. – If you enter it after the *acl-name*, ASA exempts the operating system, but does not apply the ACL to the associated traffic. • *acl-name* is the name of the ACL present in the ASA configuration. When specified, it must follow the **filter** keyword.

	Example:
	`ciscoasa(config-group-policy)# exempt-list os "Windows XP"`
	`ciscoasa(config-group-policy)`
		Adds all hosts running Windows XP to the list of computers that are exempt from posture validation.

	`ciscoasa(config-nac-policy-nac-framework)# exempt-list os "Windows XP" filter acl-2` `ciscoasa(config-nac-policy-nac-framework)`	Exempts all hosts running Windows XP and applies the ACL acl-2 to traffic from those hosts
	`ciscoasa(config-nac-policy-nac-framework)# no exempt-list os "Windows XP" filter acl-2` `ciscoasa(config-nac-policy-nac-framework)`	Removes the same entry from the exemption list.
Step 3	(Optional) [**no**] **exempt-list os** "*os-name*" [**disable** | **filter** *acl-name* [**disable**]]	Removes all exemptions from the NAC framework policy. Specifying an entry when issuing the no form of the command removes the entry from the exemption list.

	Example:
	`ciscoasa(config-nac-policy-nac-framework)# no exempt-list`
	`ciscoasa(config-nac-policy-nac-framework)`
		Removes all entries from the exemption list.

Note When the command specifies an operating system, it does not overwrite the previously added entry to the exception list; enter the command once for each operating system and ACL you want to exempt.

Assigning a NAC Policy to a Group Policy

Upon completion of each tunnel setup, the ASA applies the NAC policy, if it is assigned to the group policy, to the session. By default, the **nac-settings** command is not present in the configuration of each group policy. The ASA automatically enables NAC for a group policy when you assign a NAC policy to it.

Detailed Steps

	Command	Purpose
Step 1	**group-policy**	Switches to group-policy configuration mode.
Step 2	**nac-settings** { **value** *nac-policy-name* | **none** } **Example:** ciscoasa(config-group-policy)# **nac-settings value framework1** ciscoasa(config-group-policy)	Assigns a NAC policy to a group policy. • **nac-settings none** removes the *nac-policy-name* from the group policy and disables the use of a NAC policy for this group policy. The group policy does not inherit the nac-settings value from the default group policy. • **nac-settings value** assigns the NAC policy you name to the group policy. Assigns the NAC policy named framework1 to the group policy.
Step 3	(Optional) [**no**] **nac-settings** { **value** *nac-policy-name* | **none** }	Removes the nac-policy-name from the group policy. The group policy inherits the nac-settings value from the default group policy.
Step 4	(Optional) **show running-config nac-policy**	Displays the name and configuration of each NAC policy

Changing Global NAC Framework Settings

The ASA provides default settings for a NAC Framework configuration. Use the instructions in this section to adjust these settings for adherence to the policies in force in your network.

Changing Clientless Authentication Settings

NAC Framework support for clientless authentication is configurable. It applies to hosts that do not have a Cisco Trust Agent to fulfill the role of posture agent. The ASA applies the default access policy, sends the EAP over UDP request for posture validation, and the request times out. If the ASA is not configured to request a policy for clientless hosts from the Access Control Server, it retains the default access policy already in use for the clientless host. If the ASA is configured to request a policy for clientless hosts from the Access Control Server, it does so and the Access Control Server downloads the access policy to be enforced by the ASA.

Enabling and Disabling Clientless Authentication

Clientless authentication is enabled by default. The default configuration contains the **eou allow clientless** configuration.

Restrictions

The **eou** commands apply *only* to NAC Framework sessions.

Detailed Steps

Follow these steps to enable clientless authentication for a NAC Framework configuration:

	Command	Purpose
Step 1	`global`	Switches to global configuration mode.
Step 2	`eou allow {audit | clientless | none}`	Enables clientless authentication for a NAC framework configuration. • **audit** uses an audit server to perform clientless authentication. • **clientless** uses a Cisco Access Control Server to perform clientless authentication. • **none** disables clientless authentication.

	Example:
	`ciscoasa(config)# eou allow audit`
	`ciscoasa(config)#`
		Shows how to configure the ASA to use an audit server to perform clientless authentication.

Step 3	`[no] eou allow {audit | clientless | none}`	Removes the command from the configuration.

	Example:
	`ciscoasa(config)# no eou allow audit`
	`ciscoasa(config)#`
		Disables the use of an audit server.

Changing the Login Credentials Used for Clientless Authentication

When clientless authentication is enabled, and the ASA fails to receive a response to a validation request from the remote host, it sends a clientless authentication request on behalf of the remote host to the Access Control Server. The request includes the login credentials that match those configured for clientless authentication on the Access Control Server. The default username and password for clientless authentication on the ASA matches the default username and password on the Access Control Server; the default username and password are both “clientless.”

Prerequisites

If you change these values on the Access Control Server, you must also do so on the ASA.

Detailed Steps

Enter the following to change the username used for clientless authentication:

	Command	Purpose
Step 1	**global**	Switches to global configuration mode.
Step 2	**eou clientless username** *username* **Example:** ciscoasa(config)# **eou clientless username sherlock** ciscoasa(config)# **eou clientless password 221B-baker** ciscoasa(config)#	Changes the username used for clientless authentication. *username* must match the username configured on the Access Control Server to support clientless hosts. Enter 1 to 64 ASCII characters, excluding leading and trailing spaces, pound signs (#), question marks (?), quotation marks ("), asterisks (*), and angle brackets (< and >). Changes the username and password for clientless authentication to sherlock and 221B-baker respectively. You can specify only the username, only the password, or both.
Step 3	**eou clientless password** *password*	Changes the password used for clientless authentication. *password* must match the password configured on the Access Control Server to support clientless hosts. Enter 4 – 32 ASCII characters.
Step 4	(Optional) **no eou clientless username** **Example:** ciscoasa(config)# **no eou clientless username** ciscoasa(config)#	Changes the username to its default value.
Step 5	(Optional) **no eou clientless password** **Example:** ciscoasa(config)# **no eou clientless password** ciscoasa(config)#	Changes the password to its default value.

Changing NAC Framework Session Attributes

The ASA provides default settings for the attributes that specify communications between the ASA and the remote host. These attributes specify the port no. to communicate with posture agents on remote hosts and the expiration counters that impose limits on the communications with the posture agents. These attributes, the default settings, and the commands you can enter to change them are as follows:

Detailed Steps

	Command	Purpose
Step 1	`global`	Switches to global configuration mode.

Step 2	**eou port** *port_number*
	
	Example:
	`ciscoasa(config)# eou port 62445`
	`ciscoasa(config)#`
		The default port number is 21862. This command changes the port number (on the client endpoint) used for EAP over UDP communication with posture agents.
		port_number must match the port number configured on the CTA. Enter a value in the range 1024 to 65535.
		Changes the port number for EAP over UDP communication to 62445.

Step 3	(Optional)
	no eou port
	
	Example:
	`ciscoasa(config)# no eou port`
	`ciscoasa(config)#`
		Changes the port number to its default value.

Step 4	**eou timeout retransmit** *seconds*
	
	Example:
	`ciscoasa(config)# eou timeout retransmit 6`
	`ciscoasa(config)#`
		Changes the retransmission retry timer. When the ASA sends an EAP over UDP message to the remote host, it waits for a response. If it fails to receive a response within *n* seconds, it resends the EAP over UDP message. By default, the retransmission timer is 3 seconds.
		seconds is a value in the range 1 to 60.
		Changes the retransmission timer to 6 seconds.

Step 5	(Optional)
	no eou timeout **retransmit**
	
	Example:
	`ciscoasa(config)# no eou timeout retransmit`
	`ciscoasa(config)#`
		Changes the retransmission retry timer to its default value.

Step 6	**eou max-retry** *retries*
	
	Example:
	`ciscoasa(config)# eou max-retry 1`
	`ciscoasa(config)#`
		Changes retransmission retries. When the ASA sends an EAP over UDP message to the remote host, it waits for a response. If it fails to receive a response, it resends the EAP over UDP message. By default, it retries up to 3 times.
		retries is a value in the range 1 to 3.
		Limits the number of EAP over UDP retransmissions to 1.

	Command	Purpose
Step 7	(Optional) **no eou max-retry** **Example:** ciscoasa(config)# **no eou max-retry** ciscoasa(config)#	Changes the maximum number of retransmission retries to its default value.
Step 8	**eou timeout hold-period** *seconds* **Example:** ciscoasa(config)# **eou timeout hold-period 120** ciscoasa(config)#	Changes the session reinitialization timer. When the retransmission retry counter matches the max-retry value, the ASA terminates the EAP over UDP session with the remote host and starts the hold timer. When the hold timer equals *n* seconds, the ASA establishes a new EAP over UDP session with the remote host. By default, the maximum number of seconds to wait before establishing a new session is 180 seconds. *seconds* is a value in the range 60 to 86400. Changes the wait period before initiating a new EAP over UDP association to 120 seconds
Step 9	**(Optional)** **no** eou timeout hold-period **Example:** ciscoasa(config)# **no eou timeout hold-period** ciscoasa(config)#	Changes the session reinitialization to its default value.

Configuring Easy VPN Services on the ASA 5505

This chapter describes how to configure the ASA 5505 as an Easy VPN hardware client. This chapter assumes you have configured the switch ports and VLAN interfaces of the ASA 5505 (see Chapter 12, "Starting Interface Configuration (ASA 5505)," in the general operations configuration guide).

Note The Easy VPN hardware client configuration specifies the IP address of its primary and secondary (backup) Easy VPN servers. Any ASA, including another ASA 5505 configured as a headend, a VPN 3000 Series Concentrator, an IOS-based router, or a firewall can act as an Easy VPN server. An ASA 5505 cannot, however function as both a client and a server simultaneously. To configure an ASA 5505 as a server, see the "Specifying the Client/Server Role of the Cisco ASA 5505" section on page 8-1. Then configure the ASA 5505 as you would any other ASA, beginning with the "Getting Started" section on page 3-1 in the general operations configuration guide.

This chapter includes the following sections:

Specifying the Client/Server Role of the Cisco ASA 5505

The Cisco ASA 5505 can function as a Cisco Easy VPN hardware client (also called "Easy VPN Remote") or as a server (also called a "headend"), but not both at the same time. It does not have a default role. Use one of the following commands in global configuration mode to specify its role:

- vpnclient enable to specify the role of the ASA 5505 as an Easy VPN Remote

- **no vpnclient enable** to specify the role of the ASA 5505 as server

The following example shows how to specify the ASA 5505 as an Easy VPN hardware client:

```
ciscoasa(config)# vpnclient enable
ciscoasa(config)#
```

The CLI responds with an error message indicating that you must remove certain data elements if you switch from server to hardware client, depending on whether the elements are present in the configuration. Table 8-1 lists the data elements that are permitted in both client and server configurations, and not permitted in client configurations.

Table 8-1 Configuration Privileges and Restrictions on the ASA 5505

Permitted in Both Client and Server Configurations	Not Permitted in Client Configurations
crypto ca trustpoints	tunnel-groups
digital certificates	isakmp policies
group-policies	crypto maps
crypto dynamic-maps	
crypto ipsec transform-sets	
crypto ipsec security-association lifetime	
crypto ipsec fragmentation before-encryption	
crypto ipsec df-bit copy-df	

An ASA 5505 configured as an Easy VPN hardware client retains the commands listed in the first column within its configuration, however, some have no function in the client role.

The following example shows how to specify the ASA 5505 as an Easy VPN server:

```
ciscoasa(config)# no vpnclient enable
ciscoasa(config)#
```

After entering the no version of this command, configure the ASA 5505 as you would any other ASA, beginning with "Getting Started" section on page 3-1 in the general operations configuration guide.

Specifying the Primary and Secondary Servers

Before establishing a connection with an Easy VPN hardware client, you must specify the IP address of an Easy VPN server to which it will connect. Any ASA can act as an Easy VPN server, including another ASA 5505 configured as a headend, a VPN 3000 Series Concentrator, an IOS-based router, or a firewall.

The ASA 5505 Client always tries to set up the tunnel to the headend primary VPN server. If unable to set up the tunnel to the primary server, it tries the connection to the secondary_1 VPN server, and then sequentially down the list of VPN servers at 8 second intervals. If the setup tunnel to the secondary_1 server fails, the primary comes online during this time, and the ASA proceeds to set up the tunnel to the secondary_2 VPN server.

Use the **vpnclient server** command in global configuration mode, as follows:

[**no**] **vpnclient server** *ip_primary* [*ip_secondary_1…ip_secondary_10*]

no removes the command from the running configuration.

ip_primary_address is the IP address or DNS name of the primary Easy VPN server.

ip_secondary_address_n (Optional) is a list of the IP addresses or DNS names of up to ten backup Easy VPN servers. Use a space to separate the items in the list.

For example, enter the following command to configure a VPN client to use Easy VPN Server 10.10.10.15 as the primary server, and 10.10.10.30 and 192.168.10.45 as alternate servers:

```
ciscoasa(config)# vpnclient server 10.10.10.15 10.10.10.30 192.168.10.10
ciscoasa(config)#
```

Specifying the Mode

The Easy VPN Client supports one of two modes of operation: Client Mode or Network Extension Mode (NEM). The mode of operation determines whether the inside hosts relative to the Easy VPN Client are accessible from the Enterprise network over the tunnel. Specifying a mode of operation is mandatory before making a connection because Easy VPN Client does not have a default mode.

Client mode, also called Port Address Translation (PAT) mode, isolates the IP addresses of all devices on the Easy VPN Client private network from those on the enterprise network. The Easy VPN Client performs PAT for all VPN traffic for its inside hosts. IP address management is neither required for the Easy VPN Client inside interface or the inside hosts.

NEM makes the inside interface and all inside hosts routeable across the enterprise network over the tunnel. Hosts on the inside network obtain their IP addresses from an accessible subnet (statically or via DHCP) pre-configured with static IP addresses. PAT does not apply to VPN traffic in NEM. This mode does not require a VPN configuration for each client. The Cisco ASA 5505 configured for NEM mode supports automatic tunnel initiation. The configuration must store the group name, user name, and password. Automatic tunnel initiation is disabled if secure unit authentication is enabled.

Note If the Easy VPN hardware client is using NEM and has connections to secondary servers, use the **crypto map set reverse-route** command on each headend device to configure dynamic announcements of the remote network using Reverse Route Injection (RRI).

To specify the mode for Easy VPN Clients, enter the following command in configuration mode:

[**no**] **vpnclient mode** {**client-mode** | **network-extension-mode**}

no removes the command from the running configuration.

NEM with Multiple Interfaces

If you have an ASA 5505 security appliance (version 7.2 (3) and higher) configured as an Easy VPN Client in Network Extension Mode with multiple interfaces configured, the security appliance builds a tunnel for locally encrypted traffic only from the interface with the highest security level.

For example, consider the following configuration:

```
vlan1 security level 100 nameif inside
vlan2 security level 0 nameif outside
vlan12 security level 75 nameif work
```

In this scenario, the security appliance builds the tunnel only for vlan1, the interface with the highest security level. If you want to encrypt traffic from vlan12, you must change the security level of interface vlan1 to a lower value than that of vlan 12.

Configuring Automatic Xauth Authentication

The ASA 5505 configured as an Easy VPN hardware client automatically authenticates when it connects to the Easy VPN server if all of the following conditions are true:

- Secure unit authentication is disabled on the server.
- The server requests IKE Extended Authenticate (Xauth) credentials.

  Xauth provides the capability of authenticating a user within IKE using TACACS+ or RADIUS. Xauth authenticates a user (in this case, the Easy VPN hardware client) using RADIUS or any of the other supported user authentication protocols.
- The client configuration contains an Xauth username and password.

Enter the following command in global configuration mode to configure the Xauth username and password:

vpnclient username *xauth_username* **password** *xauth password*

You can use up to 64 characters for each.

For example, enter the following command to configure the Easy VPN hardware client to use the XAUTH username testuser and password ppurkm1:

```
ciscoasa(config)# vpnclient username testuser password ppurkm1
ciscoasa(config)#
```

To remove the username and password from the running configuration, enter the following command:

no vpnclient username

For example:

```
ciscoasa(config)# no vpnclient username
ciscoasa(config)#
```

Configuring IPsec Over TCP

By default, the Easy VPN hardware client and server encapsulate IPsec in User Datagram Protocol (UDP) packets. Some environments, such as those with certain firewall rules, or NAT and PAT devices, prohibit UDP. To use standard Encapsulating Security Protocol (ESP, Protocol 50) or Internet Key Exchange (IKE, UDP 500) in such environments, you must configure the client and the server to encapsulate IPsec within TCP packets to enable secure tunneling. If your environment allows UDP, however, configuring IPsec over TCP adds unnecessary overhead.

To configure the Easy VPN hardware client to use TCP-encapsulated IPsec, enter the following command in global configuration mode:

vpnclient ipsec-over-tcp [**port** *tcp_port*]

The Easy VPN hardware client uses port 10000 if the command does not specify a port number.

If you configure an ASA 5505 to use TCP-encapsulated IPsec, enter the following command to let it send large packets over the outside interface:

```
ciscoasa(config)# crypto ipsec df-bit clear-df outside
ciscoasa(config)#
```

This command clears the Don't Fragment (DF) bit from the encapsulated header. A DF bit is a bit within the IP header that determines whether the packet can be fragmented. This command lets the Easy VPN hardware client send packets that are larger than the MTU size.

The following example shows how to configure the Easy VPN hardware client to use TCP-encapsulated IPsec, using the default port 10000, and to let it send large packets over the outside interface:

```
ciscoasa(config)# vpnclient ipsec-over-tcp
ciscoasa(config)# crypto ipsec df-bit clear-df outside
ciscoasa(config)#
```

The next example shows how to configure the Easy VPN hardware client to use TCP-encapsulated IPsec, using the port 10501, and to let it send large packets over the outside interface:

```
ciscoasa(config)# vpnclient ipsec-over-tcp port 10501
ciscoasa(config)# crypto ipsec df-bit clear-df outside
ciscoasa(config)#
```

To remove the attribute from the running configuration, use the **no** form of this command, as follows:

no vpnclient ipsec-over-tcp

For example:

```
ciscoasa(config)# no vpnclient ipsec-over-tcp
ciscoasa(config)#
```

Comparing Tunneling Options

The tunnel types the Cisco ASA 5505 configured as an Easy VPN hardware client sets up depends on a combination of the following factors:

- Use of the **split-tunnel-network-list** and the **split-tunnel-policy** commands on the headend to permit, restrict, or prohibit split tunneling. (See the Specify a Network List for Split-Tunneling, page 4-53 and "Set the Split-Tunneling Policy" section on page 4-52, respectively.)

  Split tunneling determines the networks for which the remote-access client encrypts and sends data through the secured VPN tunnel, and determines which traffic it sends to the Internet in the clear.

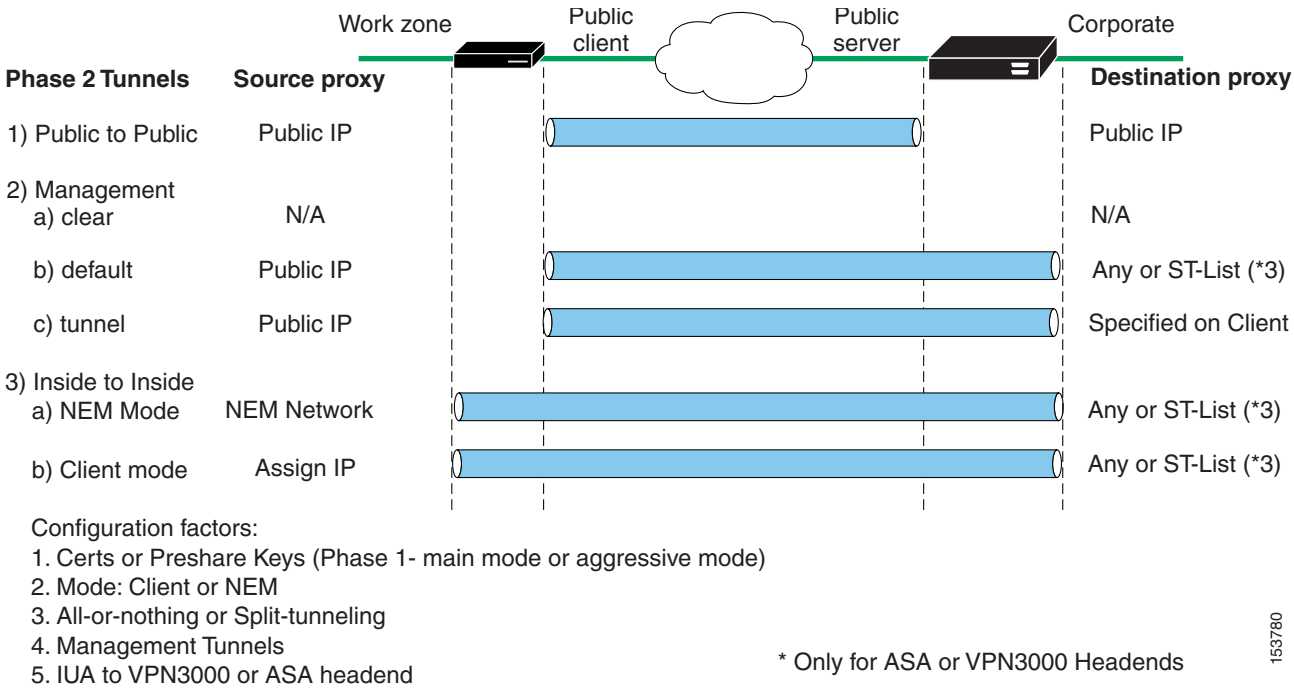
- Use of the **vpnclient management** command to specify one of the following automatic tunnel initiation options:
  - **tunnel** to limit administrative access to the client side by specific hosts or networks on the corporate side and use IPsec to add a layer of encryption to the management sessions over the HTTPS or SSH encryption that is already present.
  - **clear** to permit administrative access using the HTTPS or SSH encryption used by the management session.
  - **no** to prohibit management access

Caution Cisco does not support the use of the vpnclient management command if a NAT device is present between the client and the Internet.

- Use of the **vpnclient mode** command to specify one of the following modes of operation:
  - **client** to use Port Address Translation (PAT) mode to isolate the addresses of the inside hosts, relative to the client, from the enterprise network.
  - **network-extension-mode** to make those addresses accessible from the enterprise network.

Figure 8-1 shows the types of tunnels that the Easy VPN client initiates, based on the combination of the commands you enter.

Figure 8-1 Easy VPN Hardware Client Tunneling Options for the Cisco ASA 5505



The term "All-Or-Nothing" refers to the presence or absence of an ACL for split tunneling. The ACL ("ST-list") distinguishes networks that require tunneling from those that do not.

Specifying the Tunnel Group or Trustpoint

When configuring the Cisco ASA 5505 as an Easy VPN hardware client, you can specify a tunnel group or trustpoint configured on the Easy VPN server, depending on the Easy VPN server configuration. See the section that names the option you want to use:

- Specifying the Tunnel Group
- Specifying the Trustpoint

Specifying the Tunnel Group

Enter the following command in global configuration mode to specify the name of the VPN tunnel group and password for the Easy VPN client connection to the server:

vpnclient vpngroup *group_name* **password** *preshared_key*

group_name is the name of the VPN tunnel group configured on the Easy VPN server. You must configure this tunnel group on the server before establishing a connection.

preshared_key is the IKE pre-shared key used for authentication on the Easy VPN server.

For example, enter the following command to identify the VPN tunnel group named TestGroup1 and the IKE preshared key my_key123.

```
ciscoasa(config)# vpnclient vpngroup TestGroup1 password my_key123
ciscoasa(config)#
```

To remove the attribute from the running configuration, enter the following command:

no vpnclient vpngroup

If the configuration of the ASA 5505 running as an Easy VPN client does not specify a tunnel group, the client attempts to use an RSA certificate.

For example:

```
ciscoasa(config)# no vpnclient vpngroup
ciscoasa(config)#
```

Specifying the Trustpoint

A trustpoint represents a CA identity, and possibly a device identity, based on a certificate the CA issues. These parameters specify how the ASA obtains its certificate from the CA and define the authentication policies for user certificates issued by the CA.

First define the trustpoint using the **crypto ca trustpoint** command, as described in the general operations configuration guide. Then enter the following command in global configuration mode to name the trustpoint identifying the RSA certificate to use for authentication:

vpnclient trustpoint *trustpoint_name* [**chain**]

trustpoint_name names the trustpoint identifying the RSA certificate to use for authentication.

(Optional) **chain** sends the entire certificate chain.

For example, enter the following command to specify the identity certificate named central and send the entire certificate chain:

```
ciscoasa(config)# crypto ca trustpoint central
ciscoasa(config)# vpnclient trustpoint central chain
ciscoasa(config)#
```

To remove the attribute from the running configuration, enter the following command:

no vpnclient trustpoint

For example:

```
ciscoasa(config)# no vpnclient trustpoint
ciscoasa(config)#
```

Configuring Split Tunneling

Split tunneling lets a remote-access IPsec client conditionally direct packets over an IPsec tunnel in encrypted form or to a network interface in clear text form.

The Easy VPN server pushes the split tunneling attributes from the group policy to the Easy VPN Client for use only in the work zone. See Configuring Split-Tunneling for AnyConnect Traffic, page 4-51 to configure split tunneling on the Cisco ASA 5505.

Enter the following command in global configuration mode to enable the automatic initiation of IPsec tunnels when NEM and split tunneling are configured:

[**no**] **vpnclient nem-st-autoconnect**

no removes the command from the running configuration.

For example:

```
ciscoasa(config)# vpnclient nem-st-autoconnect
ciscoasa(config)#
```

Configuring Device Pass-Through

Devices such as Cisco IP phones, wireless access points, and printers are incapable of performing authentication. Enter the following command in global configuration mode to exempt such devices from authentication, thereby providing network access to them, if individual user authentication is enabled:

[**no**] **vpnclient mac-exempt** *mac_addr_1 mac_mask_1* [*mac_addr_2 mac_mask_2...mac_addr_n mac_mask_n*]

no removes the command from the running configuration.

mac_addr is the MAC address, in dotted hexadecimal notation, of the device to bypass individual user authentication.

mac_mask is the network mask for the corresponding MAC address. A MAC mask of ffff.ff00.0000 matches all devices made by the same manufacturer. A MAC mask of ffff.ffff.ffff matches a single device.

Note The mac-exempt list cannot exceed 15.

Only the first six characters of the specific MAC address are required if you use the MAC mask ffff.ff00.0000 to specify all devices by the same manufacturer. For example, Cisco IP phones have the Manufacturer ID 00036b, so the following command exempts any Cisco IP phone, including Cisco IP phones, you might add in the future:

```
ciscoasa(config)# vpnclient mac-exempt 0003.6b00.0000 ffff.ff00.0000
ciscoasa(config)#
```

The next example provides greater security but less flexibility because it exempts one specific Cisco IP phone:

```
ciscoasa(config)# vpnclient mac-exempt 0003.6b54.b213 ffff.ffff.ffff
ciscoasa(config)#
```

Note Make sure you have Individual User Authentiction and User Bypass configured on the headend device. For example, if you have the ASA as the headend, configure the following under group policy:

```
hostname(config-group-policy)#user-authentication enable
hostname(config-group-policy)#ip-phone-bypass enable
```

Configuring Remote Management

The Cisco ASA 5505, operating as an Easy VPN hardware client, supports management access using SSH or HTTPS, with or without a second layer of additional encryption. You can configure the Cisco ASA 5505 to require IPsec encryption within the SSH or HTTPS encryption.

Use the **vpnclient management clear** command in global configuration mode to use normal routing to provide management access from the corporate network to the outside interface of the ASA 5505 (no tunneling management packets).

Caution Do not configure a management tunnel on a Cisco ASA 5505 configured as an Easy VPN hardware client if a NAT device is operating between the Easy VPN hardware client and the Internet. In that configuration, use the **vpnclient management clear** command.

Use the **vpnclient management tunnel** command in global configuration mode if you want to automate the creation of IPsec tunnels to provide management access from the corporate network to the outside interface of the ASA 5505. The Easy VPN hardware client and server create the tunnels automatically after the execution of the **vpnclient server** command. The syntax of the vpnclient management tunnel command follows:

vpnclient management tunnel *ip_addr_1 ip_mask_1* [*ip_addr_2 ip_mask_2...ip_addr_n ip_mask_n*]

Note Regardless of your configuration, DHCP requests (including renew messages) should not flow over IPsec tunnels. Even with a vpnclient management tunnel, DHCP traffic is prohibited.

For example, enter the following command to automate the creation of an IPsec tunnel to provide management access to the host with IP address 192.168.10.10:

```
ciscoasa(config)# vpnclient management tunnel 192.198.10.10 255.255.255.0
ciscoasa(config)#
```

The **no** form of this command sets up IPsec for management tunnels in accordance with the **split-tunnel-policy** and **split-tunnel-network-list** commands.

no vpnclient management

For example:

```
ciscoasa(config)# no vpnclient management
ciscoasa(config)#
```

Guidelines for Configuring the Easy VPN Server

The following sections address the Easy VPN hardware client considerations that apply to the Easy VPN server:

- Group Policy and User Attributes Pushed to the Client
- Authentication Options

Group Policy and User Attributes Pushed to the Client

Upon tunnel establishment, the Easy VPN server pushes the values of the group policy or user attributes stored in its configuration to the Easy VPN hardware client. Therefore, to change certain attributes pushed to the Easy VPN hardware client, you must modify them on the ASAs configured as the primary and secondary Easy VPN servers. This section identifies the group policy and user attributes pushed to the Easy VPN hardware client.

Note This section serves only as a reference. For complete instructions on configuring group policies and users, see Configuring Connection Profiles, Group Policies, and Users, page 4-1.

Use Table 8-2 as a guide for determining which commands to enter to modify the group policy or user attributes.

Table 8-2 Group Policy and User Attributes Pushed to the Cisco ASA 5505 Configured as an EasyVPN Hardware Client

Command	Description
backup-servers	Sets up backup servers on the client in case the primary server fails to respond.
banner	Sends a banner to the client after establishing a tunnel.
client-access-rule	Applies access rules.
client-firewall	Sets up the firewall parameters on the VPN client.
default-domain	Sends a domain name to the client.
dns-server	Specifies the IP address of the primary and secondary DNS servers, or prohibits the use of DNS servers.
dhcp-network-scope	Specifies the IP subnetwork to which the DHCP server assigns address to users within this group.
group-lock	Specifies a tunnel group to ensure that users connect to that group.
ipsec-udp	Uses UDP encapsulation for the IPsec tunnels.
ipsec-udp-port	Specifies the port number for IPsec over UDP.
nem	Enables or disables network extension mode.
password-storage	Lets the VPN user save a password in the user profile.

Table 8-2 Group Policy and User Attributes Pushed to the Cisco ASA 5505 Configured as an EasyVPN Hardware Client (continued)

Command	Description
pfs	Commands the VPN client to use perfect forward secrecy.
re-xauth	Requires XAUTH authentication when IKE rekeys. **Note**: Disable re-xauth if secure unit authentication is enabled.
secure-unit-authentication	Enables interactive authentication for VPN hardware clients.
split-dns	Pushes a list of domains for name resolution.
split-tunnel-network-list	Specifies one of the following: • No ACL exists for split tunneling. All traffic travels across the tunnel. • Identifies the ACL the security appliance uses to distinguish networks that require tunneling and those that do not. Split tunneling lets a remote-access IPsec client conditionally direct packets over an IPsec tunnel in encrypted form, or to a network interface in cleartext form. With split-tunneling enabled, packets not bound for destinations on the other side of the IPsec tunnel do not have to be encrypted, sent across the tunnel, decrypted, and then routed to a final destination.
split-tunnel-policy	Lets a remote-access IPsec client conditionally direct packets over an IPsec tunnel in encrypted form, or to a network interface in cleartext form. Options include the following: • split-tunnel-policy—Indicates that you are setting rules for tunneling traffic. • excludespecified—Defines a list of networks to which traffic goes in the clear. • tunnelall—Specifies that no traffic goes in the clear or to any other destination than the Easy VPN server. Remote users reach Internet networks through the corporate network and do not have access to local networks. • tunnelspecified—Tunnels all traffic from or to the specified networks. This option enables split tunneling. It lets you create a network list of addresses to tunnel. Data to all other addresses travels in the clear, and is routed by the remote user's internet service provider.
user-authentication	Enables individual user authentication for hardware-based VPN clients.
vpn-access-hours	Restricts VPN access hours.
vpn-filter	Applies a filter to VPN traffic.
vpn-idle-timeout	Specifies the number of minutes a session can be idle before it times out.
vpn-session-timeout	Specifies the maximum number of minutes for VPN connections.
vpn-simultaneous-logins	Specifies the maximum number of simultaneous logins.
vpn-tunnel-protocol	Specifies the permitted tunneling protocols.
wins-server	Specifies the IP address of the primary and secondary WINS servers, or prohibits the use of WINS servers.

Note IPsec NAT-T connections are the only IPsec connection types supported on the home VLAN of a Cisco ASA 5505. IPsec over TCP and native IPsec connections are not supported.

Authentication Options

The ASA 5505 supports the following authentication mechanisms, which it obtains from the group policy stored on the Easy VPN Server. The following list identifies the authentication options supported by the Easy VPN hardware client, however, you must configure them on the Easy VPN server:

- Secure unit authentication (SUA, also called Interactive unit authentication)

  Ignores the **vpnclient username** Xauth command (described in "Configuring Automatic Xauth Authentication" section on page 8-4) and requires the user to authenticate the ASA 5505 by entering a password. By default, SUA is disabled. You can use the **secure-unit-authentication enable** command in group-policy configuration mode to enable SUA. See Configuring Secure Unit Authentication, page 4-64.

- Individual user authentication

  Requires users behind the ASA 5505 to authenticate before granting them access to the enterprise VPN network. By default, IUA is disabled. To enable the IUA, use the **user-authentication enable** command in group-policy configuration mode. See Configuring User Authentication, page 4-65.

  The security appliance works correctly from behind a NAT device, and if the ASA5505 is configured in NAT mode, the provisioned IP (to which the clients all PAT) is injected into the routing table on the central-site device.

Caution Do not configure IUA on a Cisco ASA 5505 configured as an Easy VPN server if a NAT device is operating between the server and the Easy VPN hardware client.

  Use the **user-authentication-idle-timeout** command to set or remove the idle timeout period after which the Easy VPN Server terminates the client's access. See Configuring an Idle Timeout, page 4-65.

- Authentication by HTTP redirection

  The Cisco Easy VPN server intercepts HTTP traffic and redirects the user to a login page if one of the following is true:

  – SUA or the username and password are not configured on the Easy VPN hardware client.

  – IAU is enabled.

  HTTP redirection is automatic and does not require configuration on the Easy VPN Server.

- Preshared keys, digital certificates, tokens and no authentication

  The ASA 5505 supports preshared keys, token-based (e.g., SDI one-time passwords), and "no user authentication" for user authentication. **NOTE**: The Cisco Easy VPN server can use the digital certificate as part of user authorization. See Chapter 1, "Configuring IPsec and ISAKMP" for instructions.

Configuring the PPPoE Client

This section describes how to configure the PPPoE client provided with the ASA. It includes the following topics:

PPPoE Client Overview

PPPoE combines two widely accepted standards, Ethernet and PPP, to provide an authenticated method of assigning IP addresses to client systems. PPPoE clients are typically personal computers connected to an ISP over a remote broadband connection, such as DSL or cable service. ISPs deploy PPPoE because it supports high-speed broadband access using their existing remote access infrastructure and because it is easier for customers to use.

PPPoE provides a standard method of employing the authentication methods of the Point-to-Point Protocol (PPP) over an Ethernet network. When used by ISPs, PPPoE allows authenticated assignment of IP addresses. In this type of implementation, the PPPoE client and server are interconnected by Layer 2 bridging protocols running over a DSL or other broadband connection.

PPPoE is composed of two main phases:

- Active Discovery Phase—In this phase, the PPPoE client locates a PPPoE server, called an access concentrator. During this phase, a Session ID is assigned and the PPPoE layer is established.
- PPP Session Phase—In this phase, PPP options are negotiated and authentication is performed. Once the link setup is completed, PPPoE functions as a Layer 2 encapsulation method, allowing data to be transferred over the PPP link within PPPoE headers.

At system initialization, the PPPoE client establishes a session with the access concentrator by exchanging a series of packets. Once the session is established, a PPP link is set up, which includes authentication using Password Authentication protocol (PAP). Once the PPP session is established, each packet is encapsulated in the PPPoE and PPP headers.

Note PPPoE is not supported when failover is configured on the ASA, or in multiple context or transparent mode. PPPoE is only supported in single, routed mode, without failover.

Configuring the PPPoE Client Username and Password

To configure the username and password used to authenticate the ASA to the access concentrator, use the **vpdn** command. To use the **vpdn** command, you first define a VPDN group and then create individual users within the group.

To configure a PPPoE username and password, perform the following steps:

Step 1 Define the VPDN group to be used for PPPoE using the following command:

```
ciscoasa(config)# vpdn group group_name request dialout pppoe
```

In this command, replace *group_name* with a descriptive name for the group, such as "pppoe-sbc."

Step 2 If your ISP requires authentication, select an authentication protocol by entering the following command:

```
ciscoasa(config)# vpdn group group_name ppp authentication {chap | mschap | pap}
```

Replace *group_name* with the same group name you defined in the previous step. Enter the appropriate keyword for the type of authentication used by your ISP:

- CHAP—Challenge Handshake Authentication Protocol
- MS-CHAP—Microsoft Challenge Handshake Authentication Protocol Version 1
- PAP—Password Authentication Protocol

Note When using CHAP or MS-CHAP, the username may be referred to as the remote system name, while the password may be referred to as the CHAP secret.

Step 3 Associate the username assigned by your ISP to the VPDN group by entering the following command:

```
ciscoasa(config)# vpdn group group_name localname username
```

Replace *group_name* with the VPDN group name and *username* with the username assigned by your ISP.

Step 4 Create a username and password pair for the PPPoE connection by entering the following command:

```
ciscoasa(config)# vpdn username username password password [store-local]
```

Replace *username* with the username and *password* with the password assigned by your ISP.

Note The **store-local** option stores the username and password in a special location of NVRAM on the ASA. If an Auto Update Server sends a **clear config** command to the ASA and the connection is then interrupted, the ASA can read the username and password from NVRAM and re-authenticate to the Access Concentrator.

Enabling PPPoE

Note You must complete the configuration using the **vpdn** command, described in "Configuring the PPPoE Client Username and Password," before enabling PPPoE.

The PPPoE client functionality is turned off by default. To enable PPPoE, perform the following steps:

Step 1 Enable the PPPoE client by entering the following command from interface configuration mode:

```
ciscoasa(config-if)# ip address pppoe [setroute]
```

The **setroute** option sets the default routes when the PPPoE client has not yet established a connection. When using the **setroute** option, you cannot have a statically defined route in the configuration.

PPPoE is not supported in conjunction with DHCP because with PPPoE the IP address is assigned by PPP. The **setroute** option causes a default route to be created if no default route exists. The default router is the address of the access concentrator. The maximum transmission unit (MTU) size is automatically set to 1492 bytes, which is the correct value to allow PPPoE transmission within an Ethernet frame.

Reenter this command to reset the DHCP lease and request a new lease.

Note If PPPoE is enabled on two interfaces (such as a primary and backup interface), and you do not configure dual ISP support (see the "Monitoring a Static or Default Route" section on page 25-6 in the general operations configuration guide), then the ASA can only send traffic through the first interface to acquire an IP address.

For example:

```
ciscoasa(config)# interface gigabitethernet 0/0
ciscoasa(config-if)# ip address pppoe
```

Step 2 Specify a VPDN group for the PPPoE client to use with the following command from interface configuration mode (optional):

```
ciscoasa(config-if)# pppoe client vpdn group grpname
```

grpname is the name of a VPDN *group.*

Note If you have multiple VPDN groups configured, and you do not specify a group with the **pppoe client vpdn group** command, the ASA may randomly choose a VPDN group. To avoid this, specify a VPDN group.

Using PPPoE with a Fixed IP Address

You can also enable PPPoE by manually entering the IP address, using the ip address command from interface configuration mode in the following format:

```
ciscoasa(config-if)# ip address ipaddress mask pppoe
```

This command causes the ASA to use the specified address instead of negotiating with the PPPoE server to assign an address dynamically. Replace *ipaddress* and *mask* with the IP address and subnet mask assigned to your ASA.

For example:

```
ciscoasa(config-if)# ip address outside 201.n.n.n 255.255.255.0 pppoe
```

Note The **setroute** option is an option of the **ip address** command that you can use to allow the access concentrator to set the default routes when the PPPoE client has not yet established a connection. When using the **setroute** option, you cannot have a statically defined route in the configuration.

Monitoring and Debugging the PPPoE Client

Use the following command to display the current PPPoE client configuration information:

```
ciscoasa# show ip address outside pppoe
```

Use the following command to enable or disable debugging for the PPPoE client:

```
ciscoasa# [no] debug pppoe {event | error | packet}
```

The following summarizes the function of each keyword:

- **event**—Displays protocol event information
- **error**—Displays error messages
- **packet**—Displays packet information

Use the following command to view the status of PPPoE sessions:

```
ciscoasa# show vpdn session [l2tp | pppoe] [id sess_id | packets | state | window]
```

The following example shows a sample of information provided by this command:

```
ciscoasa# show vpdn

Tunnel id 0, 1 active sessions
     time since change 65862 secs
     Remote Internet Address 10.0.0.1
    Local Internet Address 199.99.99.3
     6 packets sent, 6 received, 84 bytes sent, 0 received
Remote Internet Address is 10.0.0.1
     Session state is SESSION_UP
       Time since event change 65865 secs, interface outside
       PPP interface id is 1
       6 packets sent, 6 received, 84 bytes sent, 0 received
ciscoasa#
ciscoasa# show vpdn session
PPPoE Session Information (Total tunnels=1 sessions=1)
Remote Internet Address is 10.0.0.1
  Session state is SESSION_UP
    Time since event change 65887 secs, interface outside
    PPP interface id is 1
    6 packets sent, 6 received, 84 bytes sent, 0 received
ciscoasa#
ciscoasa# show vpdn tunnel
PPPoE Tunnel Information (Total tunnels=1 sessions=1)
Tunnel id 0, 1 active sessions
   time since change 65901 secs
   Remote Internet Address 10.0.0.1
```

```
   Local Internet Address 199.99.99.3
   6 packets sent, 6 received, 84 bytes sent, 0 received
ciscoasa#
```

Clearing the Configuration

To remove all **vpdn group** commands from the configuration, use the **clear configure vpdn group** command in global configuration mode:

```
ciscoasa(config)# clear configure vpdn group
```

To remove all **vpdn username** commands, use the **clear configure vpdn username** command:

```
ciscoasa(config)# clear configure vpdn username
```

Entering either of these commands has no affect upon active PPPoE connections.

Using Related Commands

Use the following command to cause the DHCP server to use the WINS and DNS addresses provided by the access concentrator as part of the PPP/IPCP negotiations:

```
ciscoasa(config)# dhcpd auto_config [client_ifx_name]
```

This command is only required if the service provider provides this information as described in RFC 1877. The *client_ifx_name* parameter identifies the interface supported by the DHCP **auto_config** option. At this time, this keyword is not required because the PPPoE client is only supported on a single outside interface.

CHAPTER 10

Configuring LAN-to-LAN IPsec VPNs

A LAN-to-LAN VPN connects networks in different geographic locations.

The ASA supports LAN-to-LAN VPN connections to Cisco or third-party peers when the two peers have IPv4 inside and outside networks (IPv4 addresses on the inside and outside interfaces).

For LAN-to-LAN connections using mixed IPv4 and IPv6 addressing, or all IPv6 addressing, the security appliance supports VPN tunnels if both peers are ASA 5500 series adaptive security appliances, and if both inside networks have matching addressing schemes (both IPv4 or both IPv6).

Specifically, the following topologies are supported when both peers are ASA 5500 series:

- The ASAs have IPv4 inside networks and the outside network is IPv6 (IPv4 addresses on the inside interfaces and IPv6 addresses on the outside interfaces).
- The ASAs have IPv6 inside networks and the outside network is IPv4 (IPv6 addresses on the inside interface and IPv4 addresses on the outside interfaces).
- The ASAs have IPv6 inside networks and the outside network is IPv6 (IPv6 addresses on the inside and outside interfaces).

Note The ASA supports LAN-to-LAN IPsec connections with Cisco peers, and with third-party peers that comply with all relevant standards.

This chapter describes how to build a LAN-to-LAN VPN connection. It includes the following sections:

Summary of the Configuration

This section provides a summary of the example LAN-to-LAN configuration this chapter describes. Later sections provide step-by-step instructions.

```
hostname(config)# interface ethernet0/0
hostname(config-if)# ip address 10.10.4.100 255.255.0.0
hostname(config-if)# nameif outside
hostname(config-if)# no shutdown
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)# authentication pre-share
hostname(config-ikev1-policy)# encryption 3des
hostname(config-ikev1-policy)# hash sha
hostname(config-ikev1-policy)# group 2
hostname(config-ikev1-policy)# lifetime 43200
hostname(config)# crypto ikev1 enable outside
hostname(config)# crypto ikev2 policy 1
hostname(config-ikev2-policy)# encryption 3des
hostname(config-ikev2-policy)# group 2
hostname(config-ikev12-policy)# prf sha
hostname(config-ikev2-policy)# lifetime 43200
hostname(config)# crypto ikev2 enable outside
hostname(config)# crypto ipsec ikev1 transform-set FirstSet esp-3des esp-md5-hmac
hostname(config)# crypto ipsec ikev2 ipsec-proposal secure
hostname(config-ipsec-proposal)# protocol esp encryption 3des aes des
hostname(config-ipsec-proposal)# protocol esp integrity sha-1
hostname(config)# access-list l2l_list extended permit ip 192.168.0.0 255.255.0.0
150.150.0.0 255.255.0.0
hostname(config)# tunnel-group 10.10.4.108 type ipsec-l2l
hostname(config)# tunnel-group 10.10.4.108 ipsec-attributes
hostname(config-tunnel-ipsec)# ikev1 pre-shared-key 44kkaol59636jnfx
hostname(config)# crypto map abcmap 1 match address l2l_list
hostname(config)# crypto map abcmap 1 set peer 10.10.4.108
hostname(config)# crypto map abcmap 1 set ikev1 transform-set FirstSet
hostname(config)# crypto map abcmap 1 set ikev2 ipsec-proposal secure
hostname(config)# crypto map abcmap interface outside
hostname(config)# write memory
```

Configuring Site-to-Site VPN in Multi-Context Mode

Follow these steps to allow site-to-site support in multi-mode for all platforms except the 5505. By performing these steps, you can see how resource allocation breaks down.

Step 1 To configure the VPN in multi-mode, configure a resource class and choose VPN licenses as part of the allowed resource. The "Configuring a Class for Resource Management" section on page 8-17 provides these configuration steps. The following is an example configuration:

```
class ctx1
    limit-resource VPN Burst Other 100
    limit-resource VPN Other 1000
```

Step 2 Configure a context and make it a member of the configured class that allows VPN licenses. The "Configuring a Security Context" section on page 8-20 provides these configuration steps. The following is an example configuration:

```
context context1
    member ctx1
    allocate-interface GigabitEthernet3/0.2
    allocate-interface GigabitEthernet3/1.2
    allocate-interface Management0/0
```

```
config-url disk0:/sm_s2s_ik1_ip4_no_webvpn.txt
join-failover-group 1
```

Step 3 Configure connection profiles, policies, crypto maps, and so on, just as would with single context VPN configuration of site-to-site VPN.

Configuring Interfaces

An ASA has at least two interfaces, referred to here as outside and inside. Typically, the outside interface is connected to the public Internet, while the inside interface is connected to a private network and is protected from public access.

To begin, configure and enable two interfaces on the ASA. Then, assign a name, IP address and subnet mask. Optionally, configure its security level, speed, and duplex operation on the security appliance.

Note The ASA's outside interface address (for both IPv4/IPv6) cannot overlap with the private side address space.

To configure interfaces, perform the following steps, using the command syntax in the examples:

Step 1 To enter Interface configuration mode, in global configuration mode enter the **interface** command with the default name of the interface to configure. In the following example the interface is ethernet0.

```
hostname(config)# interface ethernet0/0
hostname(config-if)#
```

Step 2 To set the IP address and subnet mask for the interface, enter the **ip address** command. In the following example the IP address is 10.10.4.100 and the subnet mask is 255.255.0.0.

```
hostname(config-if)# ip address 10.10.4.100 255.255.0.0
hostname(config-if)#
```

Step 3 To name the interface, enter the **nameif** command, maximum of 48 characters. You cannot change this name after you set it. In the following example the name of the ethernet0 interface is outside.

```
hostname(config-if)# nameif outside
hostname(config-if)##
```

Step 4 To enable the interface, enter the **no** version of the **shutdown** command. By default, interfaces are disabled.

```
hostname(config-if)# no shutdown
hostname(config-if)#
```

Step 5 To save your changes, enter the **write memory** command:

```
hostname(config-if)# write memory
hostname(config-if)#
```

Step 6 To configure a second interface, use the same procedure.

Configuring ISAKMP Policy and Enabling ISAKMP on the Outside Interface

ISAKMP is the negotiation protocol that lets two hosts agree on how to build an IPsec security association (SA). It provides a common framework for agreeing on the format of SA attributes. This includes negotiating with the peer about the SA, and modifying or deleting the SA. ISAKMP separates negotiation into two phases: Phase 1 and Phase 2. Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages. Phase 2 creates the tunnel that protects data.

IKE uses ISAKMP to setup the SA for IPsec to use. IKE creates the cryptographic keys used to authenticate peers.

The ASA supports IKEv1 for connections from the legacy Cisco VPN client, and IKEv2 for the AnyConnect VPN client.

To set the terms of the ISAKMP negotiations, you create an IKE policy, which includes the following:

- The authentication type required of the IKEv1 peer, either RSA signature using certificates or preshared key (PSK).
- An encryption method, to protect the data and ensure privacy.
- A Hashed Message Authentication Codes (HMAC) method to ensure the identity of the sender, and to ensure that the message has not been modified in transit.
- A Diffie-Hellman group to determine the strength of the encryption-key-determination algorithm. The ASA uses this algorithm to derive the encryption and hash keys.
- For IKEv2, a separate pseudo-random function (PRF) used as the algorithm to derive keying material and hashing operations required for the IKEv2 tunnel encryption, etc.
- A limit to the time the ASA uses an encryption key before replacing it.

With IKEv1 policies, for each parameter, you set one value. For IKEv2, you can configure multiple encryption and authentication types, and multiple integrity algorithms for a single policy. The ASA orders the settings from the most secure to the least secure and negotiates with the peer using that order. This allows you to potentially send a single proposal to convey all the allowed transforms instead of the need to send each allowed combination as with IKEv1.

The following sections provide procedures for creating IKEv1 and IKEv2 policies and enabling them on an interface:

- Configuring ISAKMP Policies for IKEv1 Connections, page 10-4
- Configuring ISAKMP Policies for IKEv2 Connections, page 10-5

Configuring ISAKMP Policies for IKEv1 Connections

To configure ISAKMP policies for IKEv1 connections, use the **crypto ikev1 policy** *priority* command to enter IKEv1 policy configuration mode where you can configure the IKEv1 parameters.

Perform the following steps and use the command syntax in the following examples as a guide.

Step 1 Enter IPsec IKEv1 policy configuration mode. For example:

```
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)#
```

Step 2 Set the authentication method. The following example configures a preshared key:

```
hostname(config-ikev1-policy)# authentication pre-share
hostname(config-ikev1-policy)#
```

Step 3 Set the encryption method. The following example configures 3DES:

```
hostname(config-ikev1-policy)# encryption 3des
hostname(config-ikev1-policy)#
```

Step 4 Set the HMAC method. The following example configures SHA-1:

```
hostname(config-ikev1-policy)# hash sha
hostname(config-ikev1-policy)#
```

Step 5 Set the Diffie-Hellman group. The following example configures Group 2:

```
hostname(config-ikev1-policy)# group 2
hostname(config-ikev1-policy)#
```

Step 6 Set the encryption key lifetime. The following example configures 43,200 seconds (12 hours):

```
hostname(config-ikev1-policy)# lifetime 43200
hostname(config-ikev1-policy)#
```

Step 7 Enable IKEv1 on the interface named outside in either single or multiple context mode:

```
hostname(config)# crypto ikev1 enable outside
hostname(config)#
```

Step 8 To save your changes, enter the **write memory** command:

```
hostname(config)# write memory
hostname(config)#
```

Configuring ISAKMP Policies for IKEv2 Connections

To configure ISAKMP policies for IKEv2 connections, use the **crypto ikev2 policy** *priority* command to enter IKEv2 policy configuration mode where you can configure the IKEv2 parameters.

Perform the following steps:

Step 1 Enter IPsec IKEv2 policy configuration mode. For example:

```
hostname(config)# crypto ikev2 policy 1
hostname(config-ikev2-policy)#
```

Step 2 Set the encryption method. The following example configures 3DES:

```
hostname(config-ikev2-policy)# encryption 3des
hostname(config-ikev2-policy)#
```

Step 3 Set the Diffie-Hellman group. The following example configures Group 2:

```
hostname(config-ikev2-policy)# group 2
hostname(config-ikev2-policy)#
```

Step 4 Set the pseudo-random function (PRF) used as the algorithm to derive keying material and hashing operations required for the IKEv2 tunnel encryption. The following example configures SHA-1 (an HMAC variant):

```
hostname(config-ikev12-policy)# prf sha
hostname(config-ikev2-policy)#
```

Step 5 Set the encryption key lifetime. The following example configures 43,200 seconds (12 hours):

```
hostname(config-ikev2-policy)# lifetime seconds 43200
hostname(config-ikev2-policy)#
```

Step 6 Enable IKEv2 on the interface named outside:

```
hostname(config)# crypto ikev2 enable outside
hostname(config)#
```

Step 7 To save your changes, enter the **write memory** command:

```
hostname(config)# write memory
hostname(config)#
```

Creating an IKEv1 Transform Set

An IKEv1 transform set combines an encryption method and an authentication method. During the IPsec security association negotiation with ISAKMP, the peers agree to use a particular transform set to protect a particular data flow. The transform set must be the same for both peers.

A transform set protects the data flows for the ACL specified in the associated crypto map entry. You can create transform sets in the ASA configuration, and then specify a maximum of 11 of them in a crypto map or dynamic crypto map entry.

Table 10-1 lists valid encryption and authentication methods.

Table 10-1 Valid Encryption and Authentication Methods

Valid Encryption Methods	Valid Authentication Methods
esp-des	esp-md5-hmac
esp-3des (default)	esp-sha-hmac (default)
esp-aes (128-bit encryption)	
esp-aes-192	
esp-aes-256	
esp-null	

Tunnel Mode is the usual way to implement IPsec between two ASAs that are connected over an untrusted network, such as the public Internet. Tunnel mode is the default and requires no configuration.

To configure a transform set, perform the following site-to-site tasks in either single or multiple context mode:

Step 1 In global configuration mode enter the **crypto ipsec ikev1 transform-set** command. The following example configures a transform set with the name FirstSet, esp-3des encryption, and esp-md5-hmac authentication. The syntax is as follows:

crypto ipsec ikev1 transform-set *transform-set-name* ***encryption-method authentication-method***

```
hostname(config)# crypto ipsec transform-set FirstSet esp-3des esp-md5-hmac
hostname(config)#
```

Step 2 Save your changes.

```
hostname(config)# write memory
hostname(config)#
```

Creating an IKEv2 Proposal

For IKEv2, you can configure multiple encryption and authentication types, and multiple integrity algorithms for a single policy. The ASA orders the settings from the most secure to the least secure and negotiates with the peer using that order. This allows you to potentially send a single proposal to convey all the allowed transforms instead of the need to send each allowed combination as with IKEv1.

Table 10-1 lists valid IKEv2 encryption and authentication methods.

Table 10-2 Valid IKEv2 Encryption and Integrity Methods

Valid Encryption Methods	Valid Integrity Methods
des	sha (default)
3des (default)	md5
aes	
aes-192	
aes-256	

To configure an IKEv2 proposal, perform the following tasks in either single or multiple context mode:

Step 1 In global configuration mode, use the **crypto ipsec ikev2 ipsec-proposal** command to enter ipsec proposal configuration mode where you can specify multiple encryption and integrity types for the proposal. In this example, *secure* is the name of the proposal:

```
hostname(config)# crypto ipsec ikev2 ipsec-proposal secure
hostname(config-ipsec-proposal)#
```

Step 2 Then enter a protocol and encryption types. ESP is the only supported protocol. For example:

```
hostname(config-ipsec-proposal)# protocol esp encryption 3des aes des
hostname(config-ipsec-proposal)#
```

Step 3 Enter an integrity type. For example:

```
hostname(config-ipsec-proposal)# protocol esp integrity sha-1
hostname(config-ipsec-proposal)#
```

Step 4 Save your changes.

Configuring an ACL

The ASA uses access control lists to control network access. By default, the adaptive security appliance denies all traffic. You need to configure an ACL that permits traffic. For more information, see Chapter 18, "Information About Access Control Lists," in the general operations configuration guide.

The ACLs that you configure for this LAN-to-LAN VPN control connections are based on the source and translated destination IP addresses. Configure ACLs that mirror each other on both sides of the connection.

An ACL for VPN traffic uses the translated address. For more information, see the "IP Addresses Used for ACLs When You Use NAT" section on page 18-3 in the general operations configuration guide.

To configure an ACL, perform the following steps:

Step 1 Enter the **access-list extended** command. The following example configures an ACL named l2l_list that lets traffic from IP addresses in the 192.168.0.0 network travel to the 150.150.0.0 network. The syntax is **access-list** *listname* **extended permit ip** *source-ipaddress source-netmask destination-ipaddress destination-netmask.*

```
hostname(config)# access-list l2l_list extended permit ip 192.168.0.0 255.255.0.0
150.150.0.0 255.255.0.0
hostname(config)#
```

Step 2 Configure an ACL for the ASA on the other side of the connection that mirrors the ACL. Subnets that are defined in two different crypto ACLs and are attached to the same crypto map should not overlap. In the following example, the prompt for the peer is hostname2.

```
hostname2(config)# access-list l2l_list extended permit ip 150.150.0.0 255.255.0.0
192.168.0.0 255.255.0.0
hostname(config)#
```

Note For more information on configuring an ACL with a vpn-filter, see the "Specifying a VLAN for Remote Access or Applying a Unified Access Control Rule to the Group Policy" section on page 4-44.

Defining a Tunnel Group

A tunnel group is a set of records that contain tunnel connection policies. You configure a tunnel group to identify AAA servers, specify connection parameters, and define a default group policy. The ASA stores tunnel groups internally.

There are two default tunnel groups in the ASA: DefaultRAGroup, which is the default IPsec remote-access tunnel group, and DefaultL2Lgroup, which is the default IPsec LAN-to-LAN tunnel group. You can modify them but not delete them.

The main difference between IKE versions 1 and 2 lies in terms of the authentication method they allow. IKEv1 allows only one type of authentication at both VPN ends (that is, either pre-shared key or certificate). However, IKEv2 allows assymetric authentication methods to be configured (that is, pre-shared key authentication for the originator but certificate authentication for the responder) using separate local and remote authentication CLIs. Therefore, with IKEv2 you have assymmetric authentication where one side authenticates with one credential whereas the other side uses another credential (either pre-shared key or certificate).

You can also create one or more new tunnel groups to suit your environment. The ASA uses these groups to configure default tunnel parameters for remote access and LAN-to-LAN tunnel groups when there is no specific tunnel group identified during tunnel negotiation.

To establish a basic LAN-to-LAN connection, you must set two attributes for a tunnel group:

- Set the connection type to IPsec LAN-to-LAN.
- Configure an authentication method for the IP, in the following example, preshared key for IKEv1 and IKEv2.

Note To use VPNs, including tunnel groups, the ASA must be in single-routed mode. The commands to configure tunnel-group parameters do not appear in any other mode.

Step 1 To set the connection type to IPsec LAN-to-LAN, enter the **tunnel-group** command. The syntax is **tunnel-group** *name* **type** *type,* where *name* is the name you assign to the tunnel group, and *type* is the type of tunnel. The tunnel types as you enter them in the CLI are:

- **remote-access** (IPsec, SSL, and clientless SSL remote access)
- **ipsec-l2l** (IPsec LAN to LAN)

In the following example the name of the tunnel group is the IP address of the LAN-to-LAN peer, 10.10.4.108.

```
hostname(config)# tunnel-group 10.10.4.108 type ipsec-l2l
hostname(config)#
```

Note LAN-to-LAN tunnel groups that have names that are not an IP address can be used only if the tunnel authentication method is Digital Certificates and/or the peer is configured to use Aggressive Mode.

Step 2 To set the authentication method to preshared key, enter the ipsec-attributes mode and then enter the **ikev1 pre-shared-key** command to create the preshared key. You need to use the same preshared key on both ASAs for this LAN-to-LAN connection.

The key is an alphanumeric string of 1-128 characters.

In the following example the IKEv1 preshared key is 44kkaol59636jnfx:

```
hostname(config)# tunnel-group 10.10.4.108 ipsec-attributes
hostname(config-tunnel-ipsec)# pre-shared-key 44kkaol59636jnfx
```

In the next example, the IKEv2 preshared key is configured also as 44kkaol59636jnfx:

```
hostname(config-tunnel-ipsec)# ikev2 local-authentication pre-shared-key 44kkaol59636jnfx
```

Note You must configure ikev2 remote-authentication pre-shared-key or a certificate to complete the authentication.

Step 3 Save your changes.

```
hostname(config)# write memory
hostname(config)#
```

To verify that the tunnel is up and running, use the **show vpn-sessiondb summary**, **show vpn-sessiondb detail l2l**, or **show cry ipsec sa** command.

Creating a Crypto Map and Applying It To an Interface

Crypto map entries pull together the various elements of IPsec security associations, including the following:

- Which traffic IPsec should protect, which you define in an ACL.

- Where to send IPsec-protected traffic, by identifying the peer.
- What IPsec security applies to this traffic, which a transform set specifies.
- The local address for IPsec traffic, which you identify by applying the crypto map to an interface.

For IPsec to succeed, both peers must have crypto map entries with compatible configurations. For two crypto map entries to be compatible, they must, at a minimum, meet the following criteria:

- The crypto map entries must contain compatible crypto ACLs (for example, mirror image ACLs). If the responding peer uses dynamic crypto maps, the entries in the ASA crypto ACL must be "permitted" by the peer's crypto ACL.
- The crypto map entries each must identify the other peer (unless the responding peer is using a dynamic crypto map).
- The crypto map entries must have at least one transform set in common.

If you create more than one crypto map entry for a given interface, use the sequence number (seq-num) of each entry to rank it: the lower the seq-num, the higher the priority. At the interface that has the crypto map set, the ASA evaluates traffic against the entries of higher priority maps first.

Create multiple crypto map entries for a given interface if either of the following conditions exist:

- Different peers handle different data flows.
- You want to apply different IPsec security to different types of traffic (to the same or separate peers), for example, if you want traffic between one set of subnets to be authenticated, and traffic between another set of subnets to be both authenticated and encrypted. In this case, define the different types of traffic in two separate ACLs, and create a separate crypto map entry for each crypto ACL.

To create a crypto map and apply it to the outside interface in global configuration mode, perform the following steps in either single or multiple context mode:

Step 1 To assign an ACL to a crypto map entry, enter the **crypto map match address** command.

The syntax is **crypto map** *map-name seq-num* **match address** *aclname*. In the following example the map name is abcmap, the sequence number is 1, and the ACL name is **l2l_list**.

```
hostname(config)# crypto map abcmap 1 match address l2l_list
hostname(config)#
```

Step 2 To identify the peer (s) for the IPsec connection, enter the **crypto map set peer** command.

The syntax is **crypto map** *map-name seq-num* **set peer** {*ip_address1* | *hostname1*}[... *ip_address10* | *hostname10*]. In the following example the peer name is 10.10.4.108.

```
hostname(config)# crypto map abcmap 1 set peer 10.10.4.108
hostname(config)#
```

Step 3 To specify an IKEv1 transform set for a crypto map entry, enter the **crypto map ikev1 set transform-set** command.

The syntax is **crypto map** *map-name seq-num* **ikev1 set transform-set** *transform-set-name*.
In the following example the transform set name is *FirstSet*.

```
hostname(config)# crypto map abcmap 1 set transform-set FirstSet
hostname(config)#
```

Step 4 To specify an IKEv2 proposal for a crypto map entry, enter the **crypto map ikev2 set ipsec-proposal** command:

The syntax is **crypto map** *map-name seq-num set* **ikev2 ipsec-proposal** ***proposal-name***.
In the following example the proposal name is *secure*.

With the crypto map command, you can specify multiple IPsec proposals for a single map index. In that case, multiple proposals are transmitted to the IKEv2 peer as part of the negotiation, and the order of the proposals is determined by the administrator upon the ordering of the crypto map entry.

Note If combined mode (AES-GCM/GMAC) and normal mode (all others) algorithms exist in the IPsec proposal, then you cannot send a single proposal to the peer. You must have at least two proposals in this case, one for combined mode and one for normal mode algorithms.

```
hostname(config)# crypto map abcmap 1 set ikev2 ipsec-proposal secure
hostname(config)#
```

Applying Crypto Maps to Interfaces

You must apply a crypto map set to each interface through which IPsec traffic travels. The ASA supports IPsec on all interfaces. Applying the crypto map set to an interface instructs the ASA to evaluate all interface traffic against the crypto map set and to use the specified policy during connection or security association negotiations.

Binding a crypto map to an interface also initializes the runtime data structures, such as the security association database and the security policy database. When you later modify a crypto map in any way, the ASA automatically applies the changes to the running configuration. It drops any existing connections and reestablishes them after applying the new crypto map.

To apply the configured crypto map to the outside interface, perform the following steps:

Step 1 Enter the **crypto map interface** command. The syntax is **crypto map** *map-name* **interface** *interface-name*.

```
hostname(config)# crypto map abcmap interface outside
hostname(config)#
```

Step 2 Save your changes.

```
hostname(config)# write memory
hostname(config)#
```

Configuring AnyConnect VPN Client Connections

This section describes how to configure AnyConnect VPN Client Connections and covers the following topics:

Information About AnyConnect VPN Client Connections

The Cisco AnyConnect Secure Mobility Client provides secure SSL and IPsec/IKEv2 connections to the ASA for remote users. Without a previously-installed client, remote users enter the IP address in their browser of an interface configured to accept SSL or IPsec/IKEv2 VPN connections. Unless the ASA is configured to redirect http:// requests to https://, users must enter the URL in the form https://<*address*>.

After entering the URL, the browser connects to that interface and displays the login screen. If the user satisfies the login and authentication, and the ASA identifies the user as requiring the client, it downloads the client that matches the operating system of the remote computer. After downloading, the client installs and configures itself, establishes a secure SSL or IPsec/IKEv2 connection and either remains or uninstalls itself (depending on the configuration) when the connection terminates.

In the case of a previously installed client, when the user authenticates, the ASA examines the revision of the client, and upgrades the client as necessary.

When the client negotiates an SSL VPN connection with the ASA, it connects using Transport Layer Security (TLS), and optionally, Datagram Transport Layer Security (DTLS). DTLS avoids latency and bandwidth problems associated with some SSL connections and improves the performance of real-time applications that are sensitive to packet delays.

The AnyConnect client can be downloaded from the ASA, or it can be installed manually on the remote PC by the system administrator. For more information about installing the client manually, see the *Cisco AnyConnect VPN Client Administrator Guide*.

The ASA downloads the client based on the group policy or username attributes of the user establishing the connection. You can configure the ASA to automatically download the client, or you can configure it to prompt the remote user about whether to download the client. In the latter case, if the user does not respond, you can configure the ASA to either download the client after a timeout period or present the login page.

Licensing Requirements for AnyConnect Connections

Note This feature is not available on No Payload Encryption models.

Model	License Requirement[1,2]
ASA 5505	Use one of the following: • AnyConnect Premium license: – Base license or Security Plus license: 2 sessions. – *Optional permanent or time-based licenses: 10 or 25 sessions.* – *Shared licenses are not supported.*[3] • AnyConnect Essentials license[4]: 25 sessions.
ASA 5510	Use one of the following: • AnyConnect Premium license: – Base and Security Plus license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* – *Optional Shared licenses*[3]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 250 sessions.
ASA 5520	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* – *Optional Shared licenses*[3]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 750 sessions.

Model	License Requirement[1,2]
ASA 5540	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 2500 sessions.
ASA 5550	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 5000 sessions.
ASA 5580	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 10000 sessions.
ASA 5512-X	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 250 sessions.
ASA 5515-X	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 250 sessions.

Model	License Requirement[1,2]
ASA 5525-X	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 750 sessions.
ASA 5545-X	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 2500 sessions.
ASA 5555-X	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 5000 sessions.
ASA 5585-X with SSP-10	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 5000 sessions.

Model	License Requirement[1,2]
ASA 5585-X with SSP-20, -40, and -60	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 10000 sessions.
ASA SM	Use one of the following: • AnyConnect Premium license: – Base license: 2 sessions. – *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* – *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.* • AnyConnect Essentials license[4]: 10000 sessions.

1. If you start a clientless SSL VPN session and then start an AnyConnect client session from the portal, 1 session is used in total. However, if you start the AnyConnect client first (from a standalone client, for example) and then log into the clientless SSL VPN portal, then 2 sessions are used.
2. The maximum combined VPN sessions of *all* types cannot exceed the maximum sessions shown in this table. For the ASA 5505, the maximum combined sessions is 10 for the Base license, and 25 for the Security Plus license.
3. A shared license lets the security appliance act as a shared license server for multiple client security appliances. The shared license pool is large, but the maximum number of sessions used by each individual security appliance cannot exceed the maximum number listed for permanent licenses.
4. The AnyConnect Essentials license enables AnyConnect VPN client access to the security appliance. This license does not support browser-based SSL VPN access or Cisco Secure Desktop. For these features, activate an AnyConnect Premium license instead of the AnyConnect Essentials license.

   Note: With the AnyConnect Essentials license, VPN users can use a Web browser to log in, and download and start (WebLaunch) the AnyConnect client.

   The AnyConnect client software offers the same set of client features, whether it is enabled by this license or an AnyConnect Premium SSL VPN Edition license.

   The AnyConnect Essentials license cannot be active at the same time as the following licenses on a given security appliance: AnyConnect Premium license (all types) or the Advanced Endpoint Assessment license. You can, however, run AnyConnect Essentials and AnyConnect Premium licenses on different security appliances in the same network.

   By default, the security appliance uses the AnyConnect Essentials license, but you can disable it to use other licenses by using the **no anyconnect-essentials** command.

   For a detailed list of the features supported by the AnyConnect Essentials license and AnyConnect Premium license, see *AnyConnect Secure Mobility Client Features, Licenses, and OSs*:

   http://www.cisco.com/en/US/products/ps10884/products_feature_guides_list.html

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Remote PC System Requirements

For the requirements of endpoint computers running the AnyConnect Secure Mobility Client, see the release notes for the AnyConnect client version you are deploying with the ASA.

Remote HTTPS Certificates Limitation

The ASA does not verify remote HTTPS certificates.

Configuring AnyConnect Connections

This section describes prerequisites, restrictions, and detailed tasks to configure the ASA to accept AnyConnect VPN client connections, and includes the following topics:

Configuring the ASA to Web-Deploy the Client

The section describes the steps to configure the ASA to web-deploy the AnyConnect client.

Prerequisites

Copy the client image package to the ASA using TFTP or another method.

Detailed Steps

	Command	Purpose

Step 1	`anyconnect image filename order`
	
	Example:
	`hostname(config-webvpn)#anyconnect image anyconnect-win-2.3.0254-k9.pkg 1`
	`hostname(config-webvpn)#anyconnect image anyconnect-macosx-i386-2.3.0254-k9.pkg 2`
	`hostname(config-webvpn)#anyconnect image anyconnect-linux-2.3.0254-k9.pkg 3`
		Identifies a file on flash as an AnyConnect client package file.
		
		The ASA expands the file in cache memory for downloading to remote PCs. If you have multiple clients, assign an order to the client images with the order argument.
		
		The ASA downloads portions of each client in the order you specify until it matches the operating system of the remote PC. Therefore, assign the lowest number to the image used by the most commonly-encountered operating system.
		
		Note You must issue the **anyconnect enable** command after configuring the AnyConnect images with the **anyconnect image xyz** command. If you do not enable the **anyconnect enable** command, AnyConnect will not operate as expected, and **show webvpn anyconnect** considers the SSL VPN client as not enabled rather than listing the installed AnyConnect packages.

Step 2	**enable** *interface*
	
	Example:
	`hostname(config)# webvpn`
	`hostname(config-webvpn)# enable outside`
		Enables SSL on an interface for clientless or AnyConnect SSL connections.

Step 3	**anyconnect enable**	Without issuing this command, AnyConnect does not function as expected, and a **show webvpn anyconnect** command returns that the "SSL VPN is not enabled," instead of listing the installed AnyConnect packages.
Step 4	**ip local pool** *poolname startaddr-endaddr* **mask** *mask* **Example:** `hostname(config)# ip local pool vpn_users 209.165.200.225-209.165.200.254 mask 255.255.255.224`	(Optional) Creates an address pool. You can use another method of address assignment, such as DHCP and/or user-assigned addressing.

Step 5	**address-pool** *poolname*
	
	Example:
	`hostname(config)# tunnel-group telecommuters general-attributes`
	`hostname(config-tunnel-general)# address-pool vpn_users`
		Assigns an address pool to a tunnel group.

Step 6	**default-group-policy** *name* **Example:** `hostname(config-tunnel-general)# default-group-policy sales`	Assigns a default group policy to the tunnel group.

Step 7	**group-alias** *name* **enable**
	
	Example:
	`hostname(config)# tunnel-group telecommuters webvpn-attributes`
	`hostname(config-tunnel-webvpn)# group-alias sales_department enable`
		Enables the display of the tunnel-group list on the clientless portal and AnyConnect GUI login page. The list of aliases is defined by the *group-alias name enable* command.

	Command	Purpose
Step 8	**tunnel-group-list enable** **Example:** hostname(config)# **webvpn** hostname(config-webvpn)# **tunnel-group-list enable**	Specifies the AnyConnect clients as a permitted VPN tunneling protocol for the group or user.
Step 9	**vpn-tunnel-protocol** **Example:** hostname(config)# **group-policy sales attributes** hostname(config-group-policy)# **webvpn** hostname(config-group-webvpn)# **vpn-tunnel-protocol**	Specifies SSL as a permitted VPN tunneling protocol for the group or user. You can also specify additional protocols. For more information, see the **vpn-tunnel-protocol** command in the *Cisco ASA 5500 Series Command Reference*. For more information about assigning users to group policies, see Chapter 6, Configuring Connection Profiles, Group Policies, and Users.

Enabling Permanent Client Installation

Enabling permanent client installation disables the automatic uninstalling feature of the client. The client remains installed on the remote computer for subsequent connections, reducing the connection time for the remote user.

To enable permanent client installation for a specific group or user, use the **anyconnect keep-installer** command from group-policy or username webvpn modes:

anyconnect keep-installer installer

The default is that permanent installation of the client is enabled. The client remains on the remote computer at the end of the session. The following example configures the existing group-policy *sales* to remove the client on the remote computer at the end of the session:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-policy)# anyconnect keep-installer installed none
```

Configuring DTLS

Datagram Transport Layer Security (DTLS) allows the AnyConnect client establishing an SSL VPN connection to use two simultaneous tunnels—an SSL tunnel and a DTLS tunnel. Using DTLS avoids latency and bandwidth problems associated with SSL connections and improves the performance of real-time applications that are sensitive to packet delays.

By default, DTLS is enabled when SSL VPN access is enabled on an interface. If you disable DTLS, SSL VPN connections connect with an SSL VPN tunnel only.

Note In order for DTLS to fall back to a TLS connection, Dead Peer Detection (DPD) must be enabled. If you do not enable DPD, and the DTLS connection experiences a problem, the connection terminates instead of falling back to TLS. For more information on enabling DPD, see Enabling and Adjusting Dead Peer Detection, page 11-16

You can disable DTLS for all AnyConnect client users with the **enable** command **tls-only** option in webvpn configuration mode:

enable <*interface*> **tls-only**

For example:

hostname(config-webvpn)# **enable outside tls-only**

By default, DTLS is enabled for specific groups or users with the **anyconnect ssl dtls** command in group policy webvpn or username webvpn configuration mode:

[**no**] **anyconnect ssl dtls {enable interface | none}**

If you need to disable DTLS, use the **no** form of the command. For example:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# no anyconnect ssl dtls none
```

Prompting Remote Users

You can enable the ASA to prompt remote SSL VPN client users to download the client with the **anyconnect ask** command from group policy webvpn or username webvpn configuration modes:

[**no**] **anyconnect ask** {**none** | **enable** [**default** {**webvpn** | } **timeout** *value*]}

anyconnect enable prompts the remote user to download the client or go to the clientless portal page and waits indefinitely for user response.

anyconnect ask enable default immediately downloads the client.

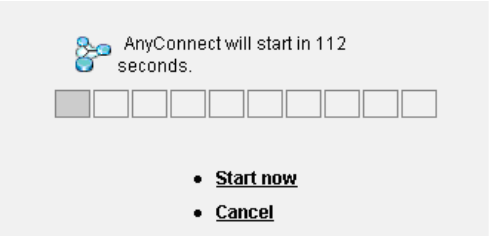
anyconnect ask enable default webvpn immediately goes to the portal page.

anyconnect ask enable default timeout *value* prompts the remote user to download the client or go to the clientless portal page and waits the duration of *value* before taking the default action—downloading the client.

anyconnect ask enable default clientless timeout *value* prompts the remote user to download the client or go to the clientless portal page, and waits the duration of *value* before taking the default action—displaying the clientless portal page.

Figure 11-1 shows the prompt displayed to remote users when either **default anyconnect timeout** *value or* **default webvpn timeout** *value is configured:*

Figure 11-1 Prompt Displayed to Remote Users for SSL VPN Client Download



The following example configures the ASA to prompt the user to download the client or go to the clientless portal page and wait *10 seconds for a response* before downloading the client:

```
hostname(config-group-webvpn)# anyconnect ask enable default anyconnect timeout 10
```

Enabling AnyConnect Client Profile Downloads

You enable Cisco AnyConnect Secure Mobility client features in the AnyConnect profiles—XML files that contain configuration settings for the core client with its VPN functionality and for the optional client modules Network Access Manager (NAM), posture, telemetry, and Web Security. The ASA deploys the profiles during AnyConnect installation and updates. Users cannot manage or modify profiles.

Profile Editor in ASDM

You can configure a profile using the AnyConnect profile editor, a convenient GUI-based configuration tool launched from ASDM. The AnyConnect software package for Windows, version 2.5 and later, includes the editor, which activates when you load the AnyConnect package on the ASA and specify it as an AnyConnect client image.

Standalone Profile Editor

We also provide a standalone version of the profile editor for Windows that you can use as an alternative to the profile editor integrated with ASDM. If you are predeploying the client, you can use the standalone profile editor to create profiles for the VPN service and other modules that you deploy to computers using your software management system. For more information about using the profile editor, see the Cisco AnyConnect Secure Mobility Client Administrator Guide.

Note The AnyConnect client protocol defaults to SSL. To enable IPsec IKEv2, you must configure the IKEv2 settings on the ASA and also configure IKEv2 as the primary protocol in the client profile. The IKEv2enabled profile must be deployed to the endpoint computer, otherwise the client attempts to connect using SSL. For more information, see the Cisco AnyConnect Secure Mobility Client Administrator Guide.

Follow these steps to edit a profile and enable the ASA to download it to remote clients:

Step 1 Use the profile editor from ASDM or the standalone profile editor to create a profile. For more information, see the Cisco AnyConnect Secure Mobility Client Administrator Guide.

Step 2 Load the profile file into flash memory on the ASA using tftp or another method.

Step 3 Use the **anyconnect profiles** command from webvpn configuration mode to identify the file as a client profile to load into cache memory.

The following example specifies the files *sales_hosts.xml* and *engineering_hosts.xml* as profiles:

```
asa1(config-webvpn)# anyconnect profiles sales disk0:/sales_hosts.xml
asa1(config-webvpn)# anyconnect profiles engineering disk0:/engineering_hosts.xml
```

The profiles are now available to group policies.

You can view the profiles loaded in cache memory using the **dir cache:stc/profiles** command:

```
hostname(config-webvpn)# dir cache:/stc/profiles

Directory of cache:stc/profiles/

0      ----  774         11:54:41 Nov 22 2006  engineering.xml
0      ----  774         11:54:29 Nov 22 2006  sales.xml

2428928 bytes total (18219008 bytes free)
hostname(config-webvpn)#
```

Step 4 Enter group policy webvpn configuration mode and specify a client profile for a group policy with the **anyconnect profiles** command:

You can enter the **anyconnect profiles value** command followed by a question mark (**?**) to view the available profiles. For example:

```
asa1(config-group-webvpn)# anyconnect profiles value ?

config-group-webvpn mode commands/options:
Available configured profile packages:
  engineering
  sales
```

The next example configures the group policy to use the profile *sales* with the client profile type *vpn*:

```
asa1(config-group-webvpn)# anyconnect profiles value sales type vpn
asa1(config-group-webvpn)#
```

Enabling AnyConnect Client Deferred Upgrade

Deferred Upgrade allows the AnyConnect user to delay download of a client upgrade. When a client update is available, AnyConnect opens a dialog asking the user if they would like to update, or to defer the upgrade.

Deferred Upgrade is enabled by adding custom attributes to the ASA, and then referencing and configuring those attributes in a group policy.

The following custom attributes support Deferred Upgrade:

Table 11-1 Custom Attributes for Deferred Upgrade

Custom Attribute	Valid Values	Default Value	Notes
DeferredUpdateAllowed	true false	false	True enables deferred update. If deferred update is disabled (false), the settings below are ignored.
DeferredUpdateMinimumVersion	x.y.z	0.0.0	Minimum version of AnyConnect that must be installed for updates to be deferrable. The minimum version check applies to all modules enabled on the headend. If any enabled module (including VPN) is not installed or does not meet the minimum version, then the connection is not eligible for deferred update. If this attribute is not specified, then a deferral prompt is displayed (or auto-dismissed) regardless of the version installed on the endpoint.

Table 11-1 Custom Attributes for Deferred Upgrade

Custom Attribute	Valid Values	Default Value	Notes
DeferredUpdateDismissTimeout	0-300 (seconds)	none (disabled)	Number of seconds that the deferred upgrade prompt is displayed before being dismissed automatically. This attribute only applies when a deferred update prompt is to be displayed (the minimum version attribute is evaluated first). If this attribute is missing, then the auto-dismiss feature is disabled, and a dialog is displayed (if required) until the user responds. Setting this attribute to zero allows automatic deferral or upgrade to be forced based on: • The installed version and the value of DeferredUpdateMinimumVersion. • The value of DeferredUpdateDismissResponse.
DeferredUpdateDismissResponse	defer update	update	Action to take when DeferredUpdateDismissTimeout occurs.

Step 1 Create the custom attributes with the **anyconnnect-custom-attr** command in webvpn configuration mode:

[no] anyconnect-custom-attr *attr-name* **[description** *description***]**

The following example shows how to add the custom attribute DeferredUpdateAllowed:

```
hostname(config)# webvpn
hostame(config-webvpn)# anyconnect-custom-attr DeferredUpdateAllowed description
"Indicates if the deferred update feature is enabled or not"
```

Step 2 Add or remove the custom attributes to a group policy, and configure values for each attribute, using the **anyconnect-custom** command:

anyconnect-custom *attr-name* **value** *value*

no anyconnect-custom *attr-nam*

The following example shows how to enable Deferred Update for the group policy named sales:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-policy)# anyconnect-custom DeferredUpdateAllowed value true
```

Enabling Additional AnyConnect Client Features

To minimize download time, the client only requests downloads (from the ASA) of the core modules that it needs. As additional features become available for the AnyConnect client, you need to update the remote clients in order for them to use the features.

To enable new features, you must specify the new module names using the **anyconnect modules** command from group policy webvpn or username webvpn configuration mode:

[no] anyconnect modules {**none** | **value** *string*}

Separate multiple strings with commas.

For a list of values to enter for each client feature, see the release notes for the Cisco AnyConnect VPN Client.

Enabling Start Before Logon

Start Before Logon (SBL) allows login scripts, password caching, drive mapping, and more, for the AnyConnect client installed on a Windows PC. For SBL, you must enable the ASA to download the module which enables graphical identification and authentication (GINA) for the AnyConnect client. The following procedure shows how to enable SBL:

Step 1 Enable the ASA to download the GINA module for VPN connection to specific groups or users using the **anyconnect modules** *vpngina* command from group policy webvpn or username webvpn configuration modes.

In the following example, the user enters group-policy attributes mode for the group policy *telecommuters*, enters webvpn configuration mode for the group policy, and specifies the string *vpngina*:

```
hostname(config)# group-policy telecommuters attributes
hostname(config-group-policy)# webvpn
hostame(config-group-webvpn)#anyconnect modules value vpngina
```

Step 2 Retrieve a copy of the client profiles file (AnyConnectProfile.tmpl).

Step 3 Edit the profiles file to specify that SBL is enabled. The example below shows the relevant portion of the profiles file (AnyConnectProfile.tmpl) for Windows:

```
<Configuration>
    <ClientInitialization>
        <UseStartBeforeLogon>false</UseStartBeforeLogon>
    </ClientInitialization>
```

The <UseStartBeforeLogon> tag determines whether the client uses SBL. To turn SBL on, replace *false* with *true*. The example below shows the tag with SBL turned on:

```
    <ClientInitialization>
        <UseStartBeforeLogon>true</UseStartBeforeLogon>
    </ClientInitialization>
```

Step 4 Save the changes to AnyConnectProfile.tmpl and update the profile file for the group or user on the ASA using the **profile** command from webvpn configuration mode. For example:

```
asa1(config-webvpn)#anyconnect profiles sales disk0:/sales_hosts.xml
```

Translating Languages for AnyConnect User Messages

The ASA provides language translation for the portal and screens displayed to users that initiate browser-based, Clientless SSL VPN connections, as well as the interface displayed to Cisco AnyConnect VPN Client users.

This section describes how to configure the ASA to translate these user messages and includes the following sections:

- Understanding Language Translation, page 11-14
- Creating Translation Tables, page 11-14

Understanding Language Translation

Functional areas and their messages that are visible to remote users are organized into translation domains. All *messages displayed on the user interface of the Cisco AnyConnect VPN Client are located in the AnyConnect domain.*

The software image package for the ASA includes a translation table template for the AnyConnect domain. You can export the template, which creates an XML file of the template at the URL you provide. The message fields in this file are empty. You can edit the messages and import the template to create a new translation table object that resides in flash memory.

You can also export an existing translation table. The XML file created displays the messages you edited previously. Reimporting this XML file with the same language name creates an new version of the translation table object, overwriting previous messages. Changes to the translation table for the AnyConnect domain are immediately visible to AnyConnect client users.

Creating Translation Tables

The following procedure describes how to create translation tables for the AnyConnect domain:

Step 1 Export a translation table template to a computer with the **export webvpn translation-table** command from privileged EXEC mode.

In the following example, the **show webvpn translation-table** command shows available translation table templates and tables.

```
hostname# show import webvpn translation-table
Translation Tables' Templates:
customization
AnyConnect
CSD
PortForwarder
url-list
webvpn
Citrix-plugin
RPC-plugin
Telnet-SSH-plugin
VNC-plugin

Translation Tables:
```

Then the user exports the translation table for the AnyConnect translation domain. The filename of the XML file created is named *client* and contains empty message fields:

```
hostname# export webvpn translation-table AnyConnect template
tftp://209.165.200.225/client
```

In the next example, the user exports a translation table named *zh*, which was previously imported from a template. zh is the abbreviation by Microsoft Internet Explorer for the Chinese language.

```
hostname# export webvpn translation-table customization language zh
tftp://209.165.200.225/chinese_client
```

Step 2 Edit the Translation Table XML file. The following example shows a portion of the AnyConnect template. The end of this output includes a message ID field (msgid) and a message string field (msgstr) for the message *Connected*, which is displayed on the AnyConnect client GUI when the client establishes a VPN connection. The complete template contains many pairs of message fields:

```
# SOME DESCRIPTIVE TITLE.
# Copyright (C) YEAR THE PACKAGE'S COPYRIGHT HOLDER
```

```
# This file is distributed under the same license as the PACKAGE package.
# FIRST AUTHOR <EMAIL@ADDRESS>, YEAR.
#
#, fuzzy
msgid ""
msgstr ""
"Project-Id-Version: PACKAGE VERSION\n"
"Report-Msgid-Bugs-To: \n"
"POT-Creation-Date: 2006-11-01 16:39-0700\n"
"PO-Revision-Date: YEAR-MO-DA HO:MI+ZONE\n"
"Last-Translator: FULL NAME <EMAIL@ADDRESS>\n"
"Language-Team: LANGUAGE <LL@li.org>\n"
"MIME-Version: 1.0\n"
"Content-Type: text/plain; charset=CHARSET\n"
"Content-Transfer-Encoding: 8bit\n"

#: C:\cygwin\home\<user>\cvc\main\Api\AgentIfc.cpp:23
#: C:\cygwin\home\<user>\cvc\main\Api\check\AgentIfc.cpp:22
#: C:\cygwin\home\<user>\cvc\main\Api\save\AgentIfc.cpp:23
#: C:\cygwin\home\<user>\cvc\main\Api\save\AgentIfc.cpp~:20
#: C:\cygwin\home\<user>\cvc\main\Api\save\older\AgentIfc.cpp:22
msgid "Connected"
msgstr ""
```

The msgid contains the default translation. The msgstr that follows msgid provides the translation. To create a translation, enter the translated text between the quotes of the msgstr string. For example, to translate the message “Connected” with a Spanish translation, insert the Spanish text between the quotes:

```
msgid "Connected"
msgstr "Conectado"
```

Be sure to save the file.

Step 3 Import the translation table using the **import webvpn translation-table** command from privileged EXEC mode. Be sure to specify the name of the new translation table with the abbreviation for the language that is compatible with the browser.

In the following example, the XML file is imported *es-us*—the abbreviation used by Microsoft Internet Explorer for Spanish spoken in the United States.

```
hostname# import webvpn translation-table AnyConnect language es-us
tftp://209.165.200.225/client
hostname# !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
hostname# show import webvpn translation-table
Translation Tables' Templates:
AnyConnect
PortForwarder
csd
customization
keepout
url-list
webvpn
Citrix-plugin
RPC-plugin
Telnet-SSH-plugin
VNC-plugin

Translation Tables:
es-us AnyConnect
```

Configuring Advanced AnyConnect SSL Features

The following section describes advanced features that fine-tune AnyConnect SSL VPN connections, and includes the following sections:

Enabling Rekey

When the ASA and the AnyConnect client client perform a rekey on an SSL VPN connection, they renegotiate the crypto keys and initialization vectors, increasing the security of the connection.

To enable the client to perform a rekey on an SSL VPN connection for a specific group or user, use the **anyconnect ssl rekey** command from group-policy or username webvpn modes.

[no]anyconnect ssl rekey {**method** {**new-tunnel** | **none** | **ssl**} | **time** *minutes*}

method new-tunnel specifies that the client establishes a new tunnel during rekey.

method ssl specifies that the client estanyablishes a new tunnel during rekey.

method none disables rekey.

Note Configuring the rekey method as **ssl** or **new-tunnel** specifies that the client establishes a new tunnel during rekey instead of the SSL renegotiation taking place during the rekey. See the Cisco ASA 5500 Series Command Reference, 8.4 for a history of the **anyconnect ssl rekey** command.

time *minutes* specifies the number of minutes from the start of the session, or from the last rekey, until the rekey takes place, from 1 to 10080 (1 week).

In the following example, the client is configured to renegotiate with SSL during rekey, which takes place 30 minutes after the session begins, for the existing group-policy *sales*:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# anyconnect ssl rekey method ssl
hostname(config-group-webvpn)# anyconnect ssl rekey time 30
```

Enabling and Adjusting Dead Peer Detection

Dead Peer Detection (DPD) ensures that the ASA (gateway) or the client can quickly detect a condition where the peer is not responding, and the connection has failed.

To enable DPD on the ASA or client for a specific group or user, and to set the frequency with which either the ASA or client performs DPD, use the **anyconnect dpd-interval** command from group-policy or username webvpn mode:

anyconnect dpd-interval {[**gateway** {*seconds* | **none***}*] | [**client** {*seconds* | **none***}*]}

Where:

gateway seconds enables DPD performed by the ASA (gateway) and specifies the frequency, from 5 to 3600 seconds, with which the ASA (gateway) performs DPD.

gateway none disables DPD performed by the ASA.

client *seconds* enable DPD performed by the client, and specifies the frequency, from 5 to 3600 seconds, with which the client performs DPD.

client none disables DPD performed by the client.

To remove the **anyconnect dpd-interval** command from the configuration, use the **no** form of the command:

no anyconnect dpd-interval {[**gateway** {*seconds* | **none***}]* | [**client** {*seconds* | **none***}]*}

Note If you enable DTLS, enable Dead Peer Detection (DPD) also. DPD enables a failed DTLS connection to fallback to TLS. Otherwise, the connection terminates.

The following example sets the frequency of DPD performed by the ASA to 30 seconds, and the frequency of DPD performed by the client set to 10 seconds for the existing group-policy *sales*:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# anyconnect dpd-interval gateway 30
hostname(config-group-webvpn)# anyconnect dpd-interval client 10
```

Enabling Keepalive

You can adjust the frequency of keepalive messages to ensure that an SSL VPN connection through a proxy, firewall, or NAT device remains open, even if the device limits the time that the connection can be idle. Adjusting the frequency also ensures that the client does not disconnect and reconnect when the remote user is not actively running a socket-based application, such as Microsoft Outlook or Microsoft Internet Explorer.

Note Keepalives are enabled by default. If you disable keepalives, in the event of a failover event, SSL VPN client sessions are not carried over to the standby device.

To set the frequency of keepalive messages, use the **keepalive** command from group-policy webvpn or username webvpn configuration mode:

[no] anyconnect ssl keepalive {none | *seconds***}**

none disables client keepalive messages.

seconds enables the client to send keepalive messages, and specifies the frequency of the messages in the range of 15 to 600 seconds.

The default is keepalive messages are enabled.

Use the **no** form of the command to remove the command from the configuration and cause the value to be inherited:

In the following example, the ASA is configured to enable the client to send keepalive messages with a frequency of 300 seconds (5 minutes), for the existing group-policy *sales*:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)#anyconnect ssl keepalive 300
```

Using Compression

Compression increases the communications performance between the ASA and the client by reducing the size of the packets being transferred for low-bandwidth connections. By default, compression for all SSL VPN connections is enabled on the ASA, both at the global level and for specific groups or users.

Note When implementing compression on broadband connections, you must carefully consider the fact that compression relies on loss-less connectivity. This is the main reason that it is not enabled by default on broadband connections.

Compression must be turned-on globally using the **anyconnect ssl compression** command from global configuration mode, and then it can be set for specific groups or users with the **anyconnect ssl compression** command in group-policy and username webvpn modes.

Changing Compression Globally

To change the global compression settings, use the **anyconnect ssl compression** command from global configuration mode:

compression

no compression

To remove the command from the configuration, use the **no** form of the command.

In the following example, compression is disabled for all SSL VPN connections globally:

```
hostname(config)# no compression
```

Changing Compression for Groups and Users

To change compression for a specific group or user, use the **anyconnect ssl compression** command in the group-policy and username webvpn modes:

anyconnect ssl compression {**deflate** | **none**}

no anyconnect ssl compression {**deflate** | **none**}

By default, for groups and users, SSL compression is set to *deflate* (enabled).

To remove the **anyconnect ssl compression** command from the configuration and cause the value to be inherited from the global setting, use the **no** form of the command:

In the following example, compression is disabled for the group-policy sales:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# no anyconnect ssl compression none
```

Adjusting MTU Size

You can adjust the MTU size (from 256 to 1406 bytes) for SSL VPN connections established by the client with the **anyconnect mtu** command from group policy webvpn or username webvpn configuration mode:

[no]anyconnect mtu *size*

This command affects only the AnyConnect client. The legacy Cisco SSL VPN Client () is not capable of adjusting to different MTU sizes.

The default for this command in the default group policy is **no anyconnect mtu**. The MTU size is adjusted automatically based on the MTU of the interface that the connection uses, minus the IP/UDP/DTLS overhead.

This command affects client connections established in SSL and those established in SSL with DTLS.

Examples The following example configures the MTU size to 1200 bytes for the group policy *telecommuters*:

```
hostname(config)# group-policy telecommuters attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)#anyconnect mtu 1200
```

Updating AnyConnect Client Images

You can update the client images on the ASA at any time using the following procedure:

Step 1 Copy the new client images to the ASA using the **copy** command from privileged EXEC mode, or using another method.

Step 2 If the new clientt image files have the same filenames as the files already loaded, reenter the **anyconnect image** command that is in the configuration. If the new filenames are different, uninstall the old files using the **noanyconnect image** command. Then use the **anyconnect image** command to assign an order to the images and cause the ASA to load the new images.

Enabling IPv6 VPN Access

If you want to configure IPv6 access, you must use the command-line interface. Release 9.0(x) of the ASA adds support for IPv6 VPN connections to its outside interface using SSL and IKEv2/IPsec protocols.

You enable IPv6 access using the **ipv6 enable** command as part of enabling SSL VPN connections. The following is an example for an IPv6 connection that enables IPv6 on the outside interface:

```
hostname(config)# interface GigabitEthernet0/0
hostname(config-if)# ipv6 enable
```

To enable IPV6 SSL VPN, do the following general actions:

1. Enable IPv6 on the outside interface.
2. Enable IPv6 and an IPv6 address on the inside interface.
3. Configure an IPv6 address local pool for client assigned IP Addresses.
4. Configure an IPv6 tunnel default gateway.

To implement this procedure, do the following steps:

Step 1 Configure Interfaces:

```
interface GigabitEthernet0/0
     nameif outside
     security-level 0
     ip address 192.168.0.1 255.255.255.0
     ipv6 enable          ; Needed for IPv6.
    !
interface GigabitEthernet0/1
```

```
nameif inside
security-level 100
ip address 10.10.0.1 255.255.0.0
ipv6 address 2001:DB8::1/32        ; Needed for IPv6.
ipv6 enable           ; Needed for IPv6.
```

Step 2 Configure an 'ipv6 local pool' (used for IPv6 address assignment):

```
ipv6 local pool ipv6pool 2001:DB8:1:1::5/32 100     ; Use your IPv6 prefix here
```

Note You can configure the ASA to assign an IPv4 address, an IPv6 address, or both an IPv4 and an IPv6 address to an AnyConnect client by creating internal pools of addresses on the ASA or by assigning a dedicated address to a local user on the ASA.

Step 3 Add the ipv6 address pool to your tunnel group policy (or group-policy):

```
tunnel-group YourTunGrp1 general-attributes  ipv6-address-pool ipv6pool
```

Note You must also configure an IPv4 address pool here as well (using the 'address-pool' command).

Step 4 Configure an IPv6 tunnel default gateway:

```
ipv6 route inside ::/0 X:X:X:X::X tunneled
```

Monitoring AnyConnect Connections

To view information about active sessions use the **show vpn-sessiondb**:

Command	Purpose
show vpn-sessiondb	Displays information about active sessions.
vpn-sessiondb logoff	Logs off VPN sessions.
show vpn-sessiondb anyconnect	Enhances the VPN session summary to show OSPFv3 session information.
show vpn-sessiondb ratio encryption	Shows the number of tunnels and percentages for the Suite B algorithms (such as AES-GCM-128, AES-GCM-192, AES-GCM-256, AES-GMAC-128, and so on).

Examples

The Inactivity field shows the elapsed time since an AnyConnect session lost connectivity. If the session is active, 00:00m:00s appears in this field.

```
hostname# show vpn-sessiondb

Session Type: SSL VPN Client

Username     : lee
Index        : 1                    IP Addr      : 209.165.200.232
```

```
Protocol     : SSL VPN Client         Encryption   : 3DES
Hashing      : SHA1                   Auth Mode    : userPassword
TCP Dst Port : 443                    TCP Src Port : 54230
Bytes Tx     : 20178                  Bytes Rx     : 8662
Pkts Tx      : 27                     Pkts Rx      : 19
Client Ver   : Cisco STC 1.1.0.117
Client Type  : Internet Explorer
Group        : DfltGrpPolicy
Login Time   : 14:32:03 UTC Wed Mar 20 2007
Duration     : 0h:00m:04s
Inactivity   : 0h:00m:04s
Filter Name  :

hostname# vpn-sessiondb logoff
INFO: Number of sessions of type "" logged off : 1

hostname# vpn-sessiondb logoff name tester
Do you want to logoff the VPN session(s)? [confirm]
INFO: Number of sessions with name "tester" logged off : 1
```

Logging Off AnyConnect VPN Sessions

To log off all VPN sessions, use the **vpn-sessiondb logoff** command in global configuration mode:

vpn-sessiondb logoff

The following example logs off all VPN sessions:

```
hostname# vpn-sessiondb logoff
INFO: Number of sessions of type "" logged off : 1
```

You can log off individual sessions using either the name argument or the index argument:

vpn-session-db logoff name *name*

vpn-session-db logoff index *index*

The sessions that have been inactive the longest time are marked as idle (and are automatically logged off) so that license capacity is not reached and new users can log in. If the session resumes at a later time, it is removed from the inactive list.

You can find both the username and the index number (established by the order of the client images) in the output of the **show vpn-sessiondb anyconnect** command. The following examples shows the username *lee* and index number *1*.

```
hostname# show vpn-sessiondb anyconnect

Session Type: AnyConnect

Username     : lee                    Index        : 1
Assigned IP  : 192.168.246.1           Public IP    : 10.139.1.2
Protocol     : AnyConnect-Parent SSL-Tunnel DTLS-Tunnel
License      : AnyConnect Premium
Encryption   : RC4 AES128             Hashing      : SHA1
Bytes Tx     : 11079                  Bytes Rx     : 4942
Group Policy : EngPolicy            Tunnel Group : EngGroup
Login Time   : 15:25:13 EST Fri Jan 28 2011
Duration     : 0h:00m:15s
Inactivity   : 0h:00m:00s
NAC Result   : Unknown
VLAN Mapping : N/A                    VLAN         : none
```

The following example terminates the session using the **name** option of the **vpn-session-db logoff** command:

```
hostname# vpn-sessiondb logoff name lee
Do you want to logoff the VPN session(s)? [confirm]
INFO: Number of sessions with name “lee” logged off : 1

hostname#
```

Configuration Examples for Enabling AnyConnect Connections

The following example shows how to configure L2TP over IPsec:

```
ip local pool sales_addresses 209.165.202.129-209.165.202.158
aaa-server sales_server protocol radius
crypto ipsec transform-set sales_l2tp_transform esp-3des esp-sha-hmac
crypto ipsec transform-set sales_l2tp_transform mode transport
crypto ipsec security-association lifetime seconds 28800
crypto ipsec security-association lifetime kilobytes 4608000
l2tp tunnel hello 100

group-policy sales_policy internal
group-policy sales_policy attributes
 wins-server value 209.165.201.3 209.165.201.4
 dns-server value 209.165.201.1 209.165.201.2
 vpn-tunnel-protocol l2tp-ipsec
tunnel-group sales_tunnel type remote-access
tunnel-group sales_tunnel general-attributes
 address-pool sales_addresses
 authentication-server-group none
 accounting-server-group sales_server
 default-group-policy sales_policy
tunnel-group sales_tunnel ppp-attributes
 authentication pap
```

Feature History for AnyConnect Connections

Table 11-2 lists the release history for this feature.

Table 11-2 Feature History for AnyConnect Connections

Feature Name	Releases	Feature Information
AnyConnect Connections	7.2(1)	The following commands were introduced or modified: **authentication eap-proxy**, **authentication ms-chap-v1**, **authentication ms-chap-v2**, **authentication pap**, **l2tp tunnel hello**, **vpn-tunnel-protocol l2tp-ipsec**.
IPsec IKEv2	8.4(1)	IKEv2 was added to support IPsec IKEv2 connections for AnyConnect and LAN-to-LAN.

Configuring AnyConnect Host Scan

Configuration > Remote Access VPN > Host Scan Image

The AnyConnect Posture Module provides the AnyConnect Secure Mobility Client the ability to identify the operating system, anti-virus, anti-spyware, and firewall software installed on the host. The Host Scan application gathers this information.

Using the secure desktop manager tool in the Adaptive Security Device Manager (ASDM), you can create a prelogin policy which evaluates the operating system, anti-virus, anti-spyware, and firewall software Host Scan identifies. Based on the result of the prelogin policy's evaluation, you can control which hosts are allowed to create a remote access connection to the security appliance.

The Host Scan support chart contains the product name and version information for the anti-virus, anti-spyware, and firewall applications you use in your prelogin policies. We deliver Host Scan and the Host Scan support chart, as well as other components, in the Host Scan package.

Starting with AnyConnect Secure Mobility Client, release 3.0, Host Scan is available separately from CSD. This means you can deploy Host Scan functionality without having to install CSD and you will be able to update your Host Scan support charts by upgrading the latest Host Scan package.

Posture assessment and the AnyConnect telemetry module require Host Scan to be installed on the host.

This chapter contains the following sections:

Host Scan Dependencies and System Requirements

Dependencies

The AnyConnect Secure Mobility Client with the posture module requires these minimum ASA components:

- ASA 8.4
- ASDM 6.4

These AnyConnect features require that you install the posture module.

- SCEP authentication
- AnyConnect Telemetry Module

System Requirements

The posture module can be installed on any of these platforms:

- Windows XP (x86 and x86 running on x64)
- Windows Vista (x86 and x86 running on x64)
- Windows 7 (x86 and x86 running on x64)
- Mac OS X 10.5,10.6 (32-bit and 32-bit running on 64-bit)
- Linux (32-bit and 32-bit running on 64-bit)
- Windows Mobile

Licensing

These are the AnyConnect licensing requirements for the posture module:

- AnyConnect Premium for basic Host Scan.
- Advanced Endpoint Assessment license is required for
  - Remediation
  - Mobile Device Management

Host Scan Packaging

You can load the Host Scan package on to the ASA in one of these ways:

- You can upload it as a standalone package: **hostscan-*version*.pkg**
- You can upload it by uploading an AnyConnect Secure Mobility package: **anyconnect-NGC-win-*version*-k9.pkg**
- You can upload it by uploading a Cisco Secure Desktop package: **csd_version-k9.pkg**

File	Description
hostscan-*version*.pkg	This file contains the Host Scan software as well as the Host Scan library and support charts.
anyconnect-NGC-win-*version*-k9.pkg	This package contains all the Cisco AnyConnect Secure Mobility Client features including the hostscan-*version*.pkg file.
csd_*version*-k9.pkg	This file contains all Cisco Secure Desktop features including Host Scan software as well as the Host Scan library and support charts. This method requires a separate license for Cisco Secure Desktop.

Installing and Enabling Host Scan on the ASA

These tasks describe installing and enabling Host Scan on the ASA:

- Installing or Upgrading Host Scan
- Enabling or Disabling a Host Scan
- Viewing the Host Scan Version Enabled on the ASA
- Uninstalling Host Scan
- Assigning AnyConnect Feature Modules to Group Policies

Installing or Upgrading Host Scan

Use this procedure to install or upgrade the Host Scan package and enable it using the command line interface for the ASA.

Prerequisites

- Log on to the ASA and enter global configuration mode. In global configuration mode, the ASA displays this prompt: hostname(config)#
- Upload the hostscan_version-k9.pkg file or anyconnect-NGC-win-version-k9.pkg file to the ASA.

Detailed Steps

	Command	Purpose
Step 1	**webvpn** **Example:** `ciscoasa(config)# webvpn`	Enter webvpn configuration mode.

Step 2	**csd hostscan image** path
	
	Example:
	`ASAName(webvpn)#csd hostscan image disk0:/hostscan-3.6.0-k9.pkg`
	`ASAName(webvpn)#csd hostscan image disk0:/anyconnect-NGC-win-3.0.0327-k9.pkg`
		Specify the path to the package you want to designate as the Host Scan image. You can specify a standalone Host Scan package or an AnyConnect Secure Mobility Client package as the Host Scan package.
		Note For all operating systems, Windows, Linux, and Mac OS X, customers need to upload the anyconnect-NGC-win-*version*-k9.pkg file in order for the endpoints to install Host Scan.

Step 3	**csd enable** **Example:** `ASAName(webvpn)#csd enable`	Enables the Host Scan image you designated in the previous step.

Step 4	`write memory`
	
	Example:
	`hostname(webvpn)# write memory`
		Saves the running configuration to flash.
		After successfully saving the new configuration to flash memory, you receive the message [OK].

Enabling or Disabling a Host Scan

These commands enable or disable an installed Host Scan image using the command line interface of the ASA.

Prerequisites

Log on to the ASA and enter global configuration mode. In global configuration mode, the ASA displays this prompt: hostname(config)#

Detailed Steps for Enabling Host Scan

	Command	Purpose
Step 1	**webvpn** **Example:** `ciscoasa(config)# webvpn`	Enter webvpn configuration mode.
Step 2	**csd enable** **Example:** ciscoasa(config)# **csd enable**	Enables the standalone Host Scan image or the Host Scan image in the AnyConnect Secure Mobility Client package if they have not been uninstalled from your ASA. If neither of those types of packages is installed and a CSD package is installed, this enables the Host Scan function in the CSD package.

Detailed Steps for Disabling Host Scan

	Command	Purpose
Step 1	**webvpn** **Example:** `ciscoasa(config)# webvpn`	Enter webvpn configuration mode.
Step 2	**no csd enable** **Example:** ciscoasa(config)# **no csd enable**	Disables Host Scan for all installed Host Scan packages. **Note** Before you uninstall the enabled Host Scan image, you must first disable Host Scan using this command.

Viewing the Host Scan Version Enabled on the ASA

Use this procedure to determine the enabled Host Scan version using ASA's command line interface.

Prerequisites

Log on to the ASA and enter privileged exec mode. In privileged exec mode, the ASA displays this prompt: **hostname#**

Command	Purpose

`show webvpn csd hostscan`
 	
Example:
`ciscoasa# show webvpn csd hostscan`
	Show the version of Host Scan enabled on the ASA.

Uninstalling Host Scan

Uninstalling Host Scan package removes it from view on the ASDM interface and prevents the ASA from deploying it even if Host Scan or CSD is enabled. Uninstalling Host Scan does not delete the Host Scan package from the flash drive.

Prerequisites

Log on to the ASA and enter global configuration mode. In global configuration mode, the ASA displays this prompt: **hostname(config)#.**

Detailed Steps

	Command	Purpose
Step 1	**`webvpn`** **Example:** `ciscoasa(config)# webvpn`	Enter webvpn configuration mode.
Step 2	**`no csd enable`** **Example:** `ASAName(webvpn)#no csd enable`	Disables the Host Scan image you want to uninstall.
Step 3	**`no csd hostscan image`** `path` **Example:** hostname(webvpn)#no csd hostscan image disk0:/hostscan-3.6.0-k9.pkg `hostname(webvpn)#no csd hostscan image disk0:/anyconnect-NGC-win-3.0.0327-k9.pkg`	Specifies the path to the Host Scan image you want to uninstall. A standalone Host Scan package or an AnyConnect Secure Mobility Client package may have been designated as the Host Scan package.

Step 4	`write memory`
	
	Example:
	`hostname(webvpn)# write memory`
		Saves the running configuration to flash.
		After successfully saving the new configuration to flash memory, you receive the message [OK].

Assigning AnyConnect Feature Modules to Group Policies

This procedure associates AnyConnect feature modules with a group policy. When VPN users connect to the ASA, the ASA downloads and installs these AnyConnect feature modules to their endpoint computer.

Prerequisites

Log on to the ASA and enter global configuration mode. In global configuration mode, the ASA displays this prompt: **hostname(config)#**

Detailed Steps

	Command	Purpose
Step 1	**group-policy** *name* **internal** **Example:** `hostname(config)# group-policy PostureModuleGroup internal`	Adds an internal group policy for Network Client Access
Step 2	**group-policy** *name* **attributes** **Example:** `hostname(config)# group-policy PostureModuleGroup attributes`	Edits the new group policy. After entering the command, you receive the prompt for group policy configuration mode, **hostname(config-group-policy)#**.

Step 3	`webvpn`
	
	Example:
	`hostname(config-group-policy)# webvpn`
		Enters group policy webvpn configuration mode. After you enter the command, the ASA returns this prompt:
		
		hostname(config-group-webvpn)#

	Command	Purpose
Step 4	`hostname(config-group-webvpn)# anyconnect modules value` AnyConnect Module *Name* **Example:** `hostname(config-group-webvpn)# anyconnect modules value websecurity,telemetry,posture`	Configures the group policy to download AnyConnect feature modules for all users in the group. The value of the anyconnect module command can contain one or more of the following values. When specifying more than one module, separate the values with a comma. **value** — **AnyConnect Module Name** **dart** — AnyConnect DART (Diagnostics and Reporting Tool) **nam** — AnyConnect Network Access Manager **vpngina** — AnyConnect SBL (Start Before Logon) **websecurity** — AnyConnect Web Security Module **telemetry** — AnyConnect Telemetry Module **posture** — AnyConnect Posture Module **none** — Used by itself to remove all AnyConnect modules from the group policy. To remove one of the modules, re-send the command specifying only the module values you want to keep. For example, this command **removes** the websecurity module: `hostname(config-group-webvpn)# anyconnect modules value telemetry,posture`

Step 5	`write memory`
	
	Example:
	`hostname(config-group-webvpn)# write memory`
		Saves the running configuration to flash.
		
		After successfully saving the new configuration to flash memory, you receive the message [OK] and the ASA returns you to this prompt:
		
		hostname(config-group-webvpn)#

Other Important Documentation Addressing Host Scan

Once Host Scan gathers the posture credentials from the endpoint computer, you will need to understand subjects like, configuring prelogin policies, configuring dynamic access policies, and using Lua expressions to make use of the information.

These topics are covered in detail in these documents:

- Cisco Secure Desktop Configuration Guides
- Cisco Adaptive Security Device Manager Configuration Guides

See also the Cisco *AnyConnect Secure Mobility Client Administrator Guide, Release 3.0* for more information about how Host Scan works with AnyConnect clients.

Configuring an External Server for Authorization and Authentication

This chapter describes how to configure an external LDAP, RADIUS, or TACACS+ server to support AAA for the ASA. Before you configure the ASA to use an external server, you must configure the AAA server with the correct ASA authorization attributes and, from a subset of these attributes, assign specific permissions to individual users.

Understanding Policy Enforcement of Authorization Attributes

The ASA supports several methods of applying user authorization attributes (also called user entitlements or permissions) to VPN connections. You can configure the ASA to obtain user attributes from any combination of:

- a Dynamic Access Policy (DAP) on the ASA
- an external RADIUS or LDAP authentication and/or authorization server
- a group policy on the ASA

If the ASA receives attributes from all sources, the attributes are evaluated, merged, and applied to the user policy. If there are conflicts between attributes, the DAP attributes take precedence.

The ASA applies attributes in the following order (see Figure 13-1).

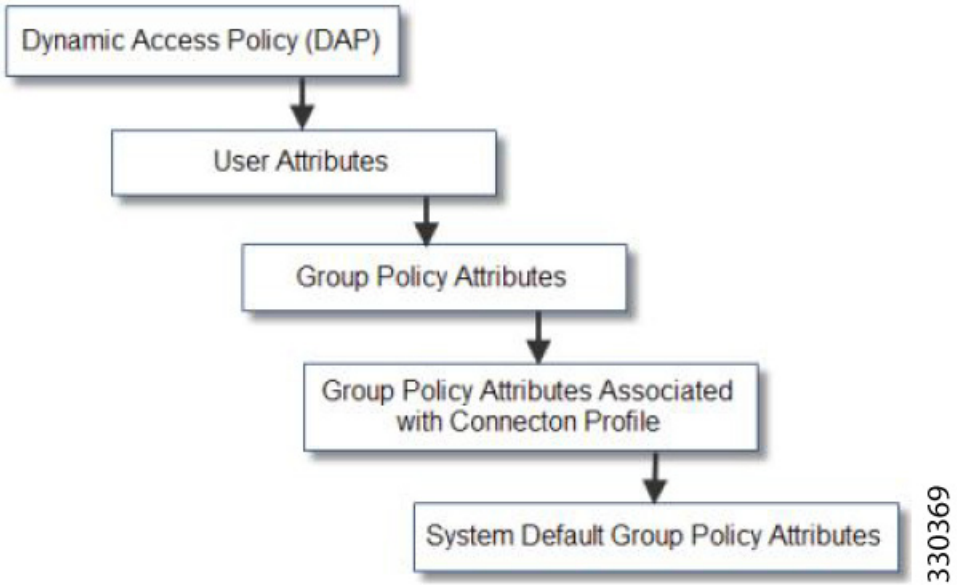
1. DAP attributes on the ASA—Introduced in Version 8.0(2), these attributes take precedence over all others. If you set a bookmark or URL list in DAP, it overrides a bookmark or URL list set in the group policy.
2. User attributes on the AAA server—The server returns these attributes after successful user authentication and/or authorization. Do not confuse these with attributes that are set for individual users in the local AAA database on the ASA (User Accounts in ASDM).
3. Group policy configured on the ASA—If a RADIUS server returns the value of the RADIUS CLASS attribute IETF-Class-25 (OU=*group-policy*) for the user, the ASA places the user in the group policy of the same name and enforces any attributes in the group policy that are not returned by the server.

   For LDAP servers, any attribute name can be used to set the group policy for the session. The LDAP attribute map that you configure on the ASA maps the LDAP attribute to the Cisco attribute IETF-Radius-Class.

4. Group policy assigned by the Connection Profile (called tunnel-group in the CLI)—The Connection Profile has the preliminary settings for the connection, and includes a default group policy applied to the user before authentication. All users connecting to the ASA initially belong to this group, which provides any attributes that are missing from the DAP, user attributes returned by the server, or the group policy assigned to the user.
5. Default group policy assigned by the ASA (DfltGrpPolicy)—System default attributes provide any values that are missing from the DAP, user attributes, group policy, or connection profile.

Figure 13-1 Policy Enforcement Flow

Defining the ASA LDAP Configuration



Authorization refers to the process of enforcing permissions or attributes. An LDAP server defined as an authentication or authorization server enforces permissions or attributes if they are configured.

Guidelines

The ASA enforces the LDAP attributes based on attribute name, not numeric ID. RADIUS attributes, are enforced by numeric ID, not by name.

For ASDM Version 7.0, LDAP attributes include the cVPN3000 prefix. For ASDM Versions 7.1 and later, this prefix was removed.

LDAP attributes are a subet of the Radius attributes, which are listed in the Radius chapter.

Active Directory/LDAP VPN Remote Access Authorization Examples

This section presents example procedures for configuring authentication and authorization on the ASA using the Microsoft Active Directory server. It includes the following topics:

- Placing LDAP Users in a Specific Group Policy, page 13-5
- Enforcing Static IP Address Assignment for AnyConnect Tunnels, page 13-7
- Enforcing Dial-in Allow or Deny Access, page 13-9
- Enforcing Logon Hours and Time-of-Day Rules, page 13-12

Other configuration examples available on Cisco.com include the following TechNotes.

- *ASA/PIX: Mapping VPN Clients to VPN Group Policies Through LDAP Configuration Example* at the following URL:

  http://www.cisco.com/en/US/products/ps6120/products_configuration_example09186a008089149d.shtml

- *PIX/ASA 8.0: Use LDAP Authentication to Assign a Group Policy at Login* at the following URL:

  http://www.cisco.com/en/US/products/ps6120/products_configuration_example09186a00808d1a7c.shtml

User-Based Attributes Policy Enforcement

You can map any standard LDAP attribute to a well-known Vendor-Specific Attribute (VSA), and you can map one or more LDAP attribute(s) to one or more Cisco LDAP attributes.

The following example shows how to configure the ASA to enforce a simple banner for a user who is configured on an AD LDAP server. On the server, use the Office field in the General tab to enter the banner text. This field uses the attribute named physicalDeliveryOfficeName. On the ASA, create an attribute map that maps physicalDeliveryOfficeName to the Cisco attribute Banner1. During authentication, the ASA retrieves the value of physicalDeliveryOfficeName from the server, maps the value to the Cisco attribute Banner1, and displays the banner to the user.

This example applies to any connection type, including the IPsec VPN client, AnyConnect SSL VPN client, or clientless SSL VPN. In the example, User1 connects through a clientless SSL VPN connection.

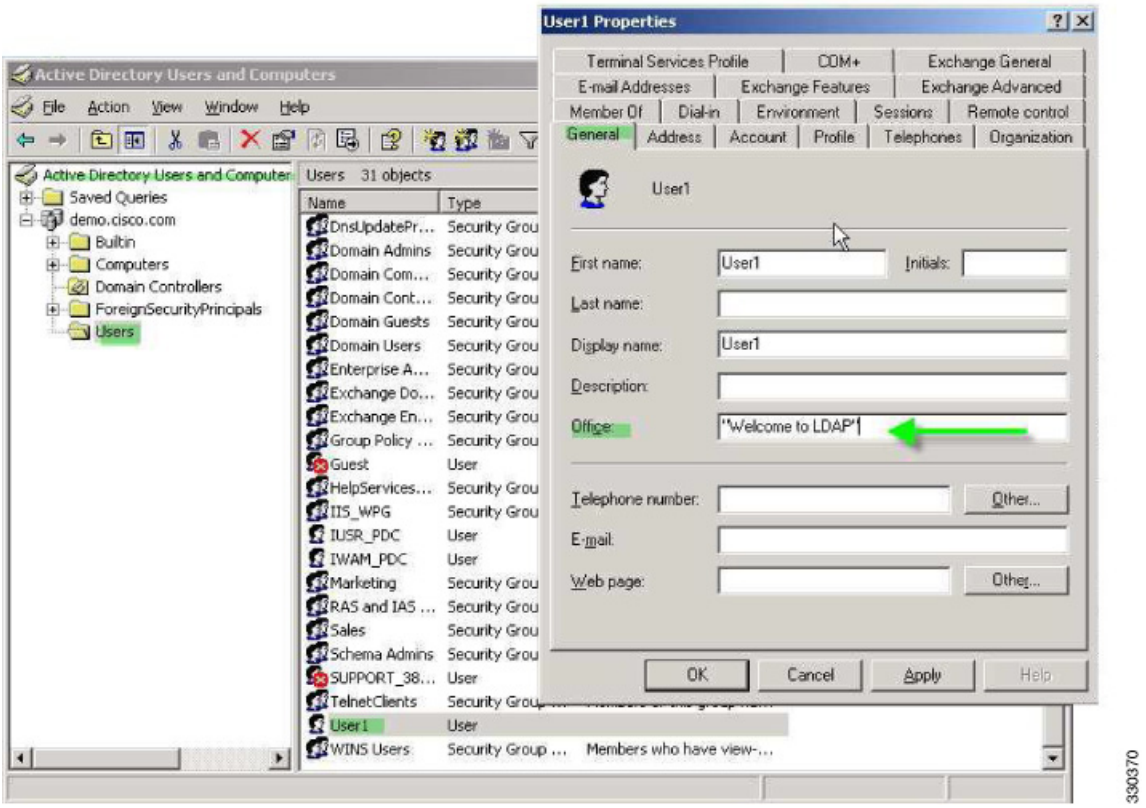
To configure the attributes for a user on the AD or LDAP Server, perform the following steps:

Step 1 Right-click a user.

The Properties dialog box appears (see Figure 13-2).

Step 2 Click the **General** tab and enter banner text in the Office field, which uses the AD/LDAP attribute physicalDeliveryOfficeName.

Figure 13-2 LDAP User Configuration



Step 3 Create an LDAP attribute map on the ASA.

The following example creates the map Banner and maps the AD/LDAP attribute physicalDeliveryOfficeName to the Cisco attribute Banner1:

```
ciscoasa(config)# ldap attribute-map Banner
ciscoasa(config-ldap-attribute-map)# map-name physicalDeliveryOfficeName Banner1
```

Step 4 Associate the LDAP attribute map to the AAA server.

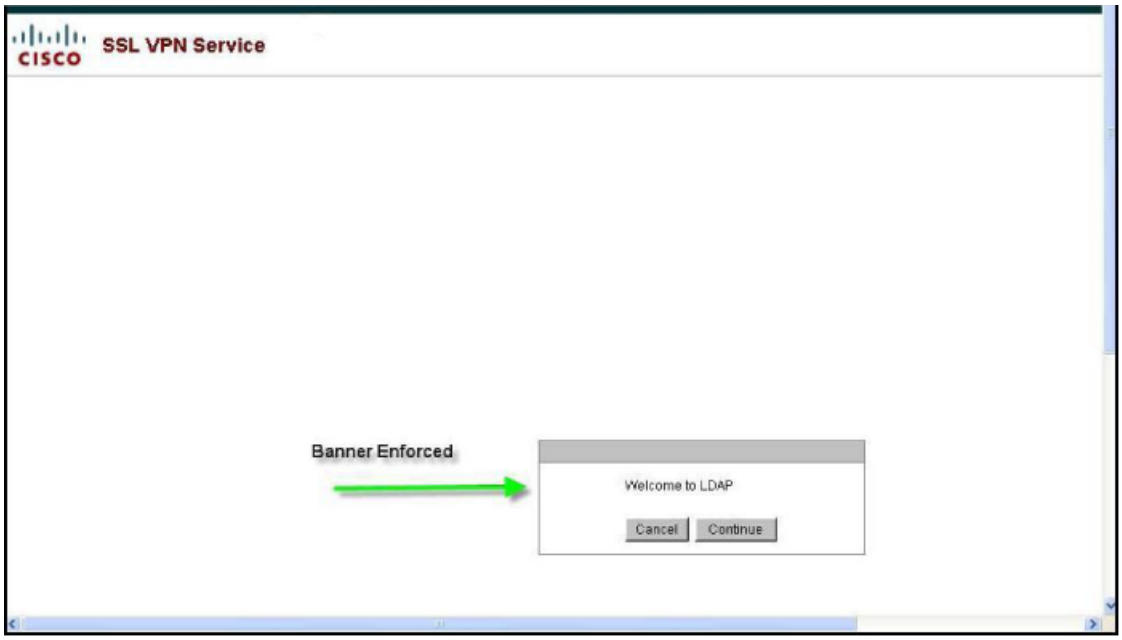
The following example enters the aaa server host configuration mode for the host 10.1.1.2 in the AAA server group MS_LDAP, and associates the attribute map Banner that you created in Step 3:

```
ciscoasa(config)# aaa-server MS_LDAP host 10.1.1.2
ciscoasa(config-aaa-server-host)# ldap-attribute-map Banner
```

Step 5 Test the banner enforcement.

The following example shows a clientless SSL connection and the banner enforced through the attribute map after the user authenticates (see Figure 13-3).

Figure 13-3 Banner Displayed



Placing LDAP Users in a Specific Group Policy

The following example shows how to authenticate User1 on the AD LDAP server to a specific group policy on the ASA. On the server, use the Department field of the Organization tab to enter the name of the group policy. Then create an attribute map, and map Department to the Cisco attribute IETF-Radius-Class. During authentication, the ASA retrieves the value of Department from the server, maps the value to the IETF-Radius-Class, and places User1 in the group policy.

This example applies to any connection type, including the IPsec VPN client, AnyConnect SSL VPN client, or clientless SSL VPN. In this example, User1 is connecting through a clientless SSL VPN connection.

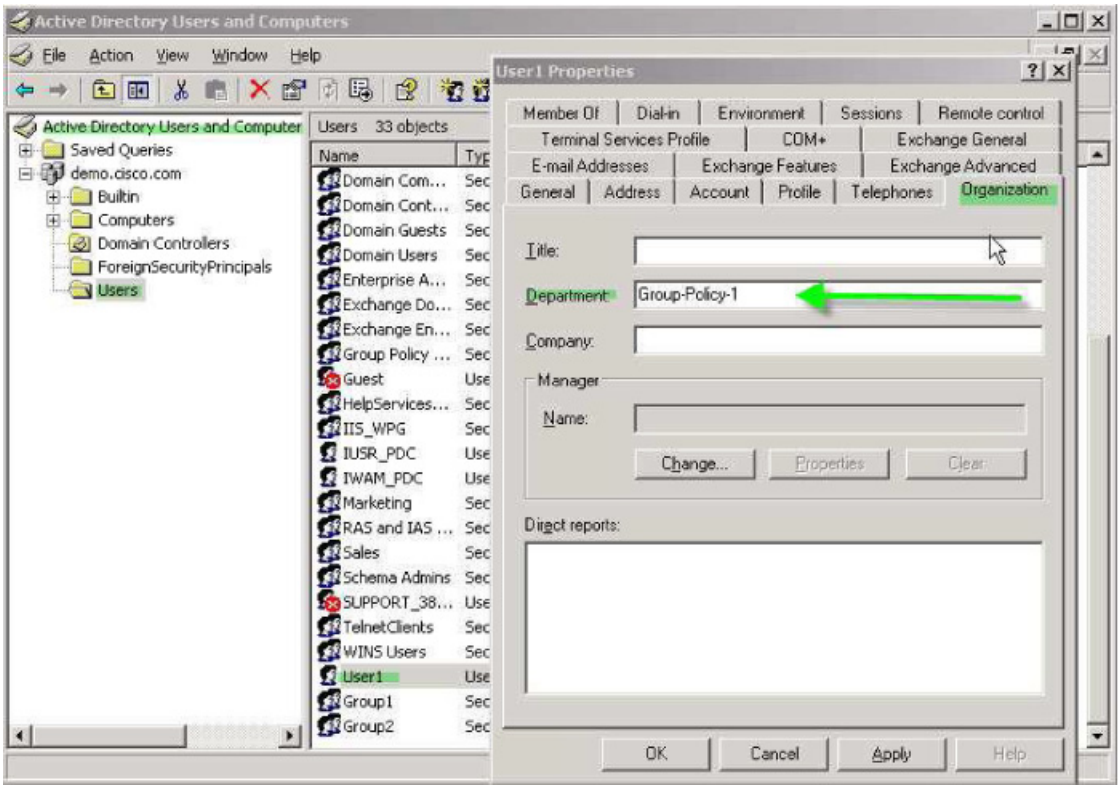
To configure the attributes for the user on the AD LDAP server, perform the following steps:

Step 1 Right-click the user.

The Properties dialog box appears (see Figure 13-4).

Step 2 Click the **Organization** tab and enter **Group-Policy-1** in the Department field.

Figure 13-4 AD/LDAP Department Attribute



Step 3 Define an attribute map for the LDAP configuration shown in Step 1.

The following example shows how to map the AD attribute Department to the Cisco attribute IETF-Radius-Class.

```
ciscoasa(config)# ldap attribute-map group_policy
ciscoasa(config-ldap-attribute-map)# map-name Department IETF-Radius-Class
```

Step 4 Associate the LDAP attribute map to the AAA server.

The following example enters the aaa server host configuration mode for the host 10.1.1.2 in the AAA server group MS_LDAP, and associates the attribute map group_policy that you created in Step 3:

```
ciscoasa(config)# aaa-server MS_LDAP host 10.1.1.2
ciscoasa(config-aaa-server-host)# ldap-attribute-map group_policy
```

Step 5 Add the new group-policy on the ASA and configure the required policy attributes that will be assigned to the user. The following example creates Group-policy-1, the name entered in the Department field on the server:

```
ciscoasa(config)# group-policy Group-policy-1 external server-group LDAP_demo
ciscoasa(config-aaa-server-group)#
```

Step 6 Establish the VPN connection as the user would, and verify that the session inherits the attributes from Group-Policy1 (and any other applicable attributes from the default group-policy).

Step 7 Monitor the communication between the ASA and the server by enabling the **debug ldap 255** command from privileged EXEC mode. The following is sample output from this command, which has been edited to provide the key messages:

```
[29] Authentication successful for user1 to 10.1.1.2
[29] Retrieving user attributes from server 10.1.1.2
```

```
[29] Retrieved Attributes:
[29] department: value = Group-Policy-1
[29] mapped to IETF-Radius-Class: value = Group-Policy-1
```

Enforcing Static IP Address Assignment for AnyConnect Tunnels

In this example, configure the AnyConnect client user Web1 to receive a static IP address. then enter the address in the Assign Static IP Address field of the Dialin tab on the AD LDAP server. This field uses the msRADIUSFramedIPAddress attribute. Create an attribute map that maps this attribute to the Cisco attribute IETF-Radius-Framed-IP-Address.

During authentication, the ASA retrieves the value of msRADIUSFramedIPAddress from the server, maps the value to the Cisco attribute IETF-Radius-Framed-IP-Address, and provides the static address to User1.

The following example applies to full-tunnel clients, including the IPsec client and the SSL VPN clients (AnyConnect client 2.x and the SSL VPN client).

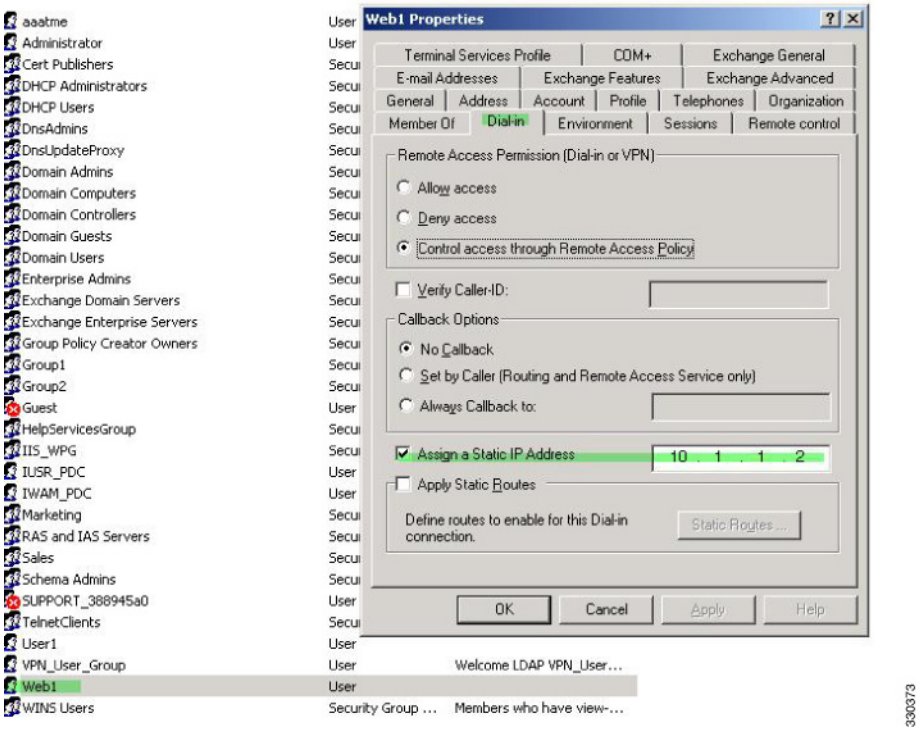
To configure the user attributes on the AD /LDAP server, perform the following steps:

Step 1 Right-click the username.

The Properties dialog box appears (see Figure 13-5).

Step 2 Click the **Dialin** tab, check the **Assign Static IP Address** check box, and enter an IP address of 10.1.1.2.

Figure 13-5 Assign Static IP Address



Step 3 Create an attribute map for the LDAP configuration shown in Step 1.

The following example shows how to map the AD attribute msRADIUSFramedIPAddress used by the Static Address field to the Cisco attribute IETF-Radius-Framed-IP-Address:

```
ciscoasa(config)# ldap attribute-map static_address
ciscoasa(config-ldap-attribute-map)# map-name msRADIUSFramedIPAddress
IETF-Radius-Framed-IP-Address
```

Step 4 Associate the LDAP attribute map to the AAA server.

The following example enters the aaa server host configuration mode for the host 10.1.1.2, in the AAA server group MS_LDAP, and associates the attribute map static_address that you created in Step 3:

```
ciscoasa(config)# aaa-server MS_LDAP host 10.1.1.2
ciscoasa(config-aaa-server-host)# ldap-attribute-map static_address
```

Step 5 Verify that the **vpn-address-assignment** command is configured to specify AAA by viewing this part of the configuration with the **show run all vpn-addr-assign** command:

```
ciscoasa(config)# show run all vpn-addr-assign
vpn-addr-assign aaa   << Make sure this is configured >>
no vpn-addr-assign dhcp
vpn-addr-assign local
ciscoasa(config)#
```

Step 6 Establish a connection to the ASA with the AnyConnect client. Observe the following:

- The banner is received in the same sequence as a clientless connection (see Figure 13-6).
- The user receives the IP address configured on the server and mapped to the ASA (see Figure 13-7).

Figure 13-6 Verify the Banner for the AnyConnect Session

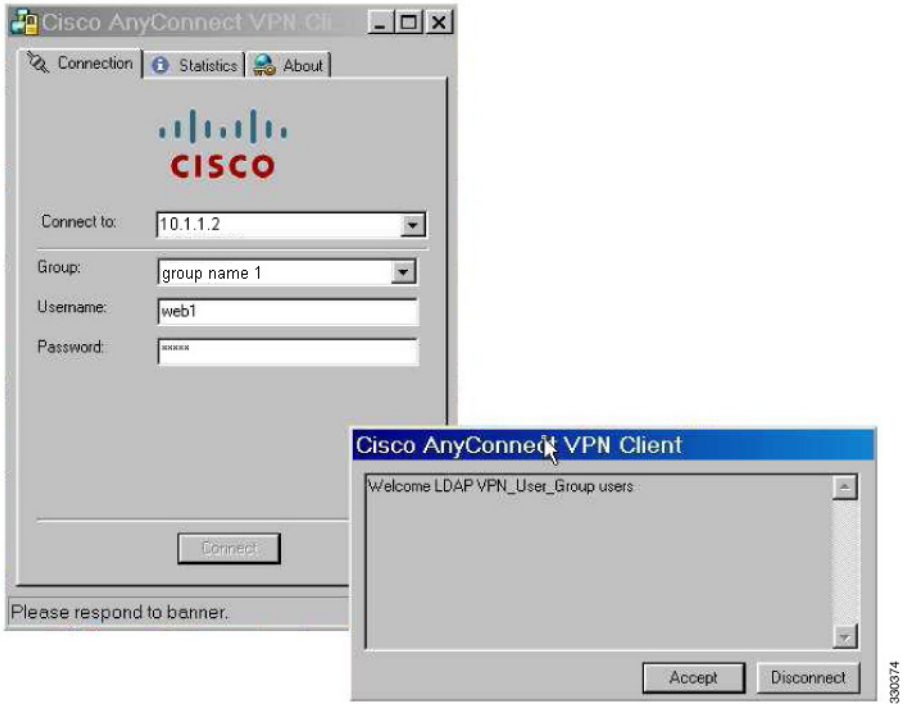


Figure 13-7 AnyConnect Session Established

Step 7 Use the **show vpn-sessiondb svc** command to view the session details and verify the address assigned:

```
ciscoasa# show vpn-sessiondb svc

Session Type: SVC
Username     : web1                   Index        : 31
Assigned IP  : 10.1.1.2              Public IP    : 10.86.181.70
Protocol     : Clientless SSL-Tunnel DTLS-Tunnel
Encryption   : RC4 AES128             Hashing      : SHA1
Bytes Tx     : 304140                 Bytes Rx     : 470506
Group Policy : VPN_User_Group         Tunnel Group : Group1_TunnelGroup
Login Time   : 11:13:05 UTC Tue Aug 28 2007
Duration     : 0h:01m:48s
NAC Result   : Unknown
VLAN Mapping : N/A                    VLAN         : none
```

Enforcing Dial-in Allow or Deny Access

The following example creates an LDAP attribute map that specifies the tunneling protocols allowed by the user. You map the allow access and deny access settings on the Dialin tab to the Cisco attribute Tunneling-Protocol, which supports the bitmap values shown in Table 13-1:

Table 13-1 Bitmap Values for Cisco Tunneling-Protocol Attribute

Value	Tunneling Protocol
1	PPTP
2	L2TP
4[1]	IPsec (IKEv1)
8[2]	L2TP/IPsec

Table 13-1 Bitmap Values for Cisco Tunneling-Protocol Attribute (continued)

Value	Tunneling Protocol
16	Clientless SSL
32	SSL client—AnyConnect or SSL VPN client
64	IPsec (IKEv2)

1. IPsec and L2TP over IPsec are not supported simultaneously. Therefore, the values 4 and 8 are mutually exclusive.
2. See note 1.

Use this attribute to create an Allow Access (TRUE) or a Deny Access (FALSE) condition for the protocols, and enforce the method for which the user is allowed access.

For this simplified example, by mapping the tunnel protocol IPsec/IKEv1 (4), you can create an allow (true) condition for the Cisco VPN client. You also map WebVPN (16) and SVC/AC (32), which are mapped as a value of 48 (16+32) and create a deny (false) condition. This allows the user to connect to the ASA using IPsec, but any attempt to connect using clientless SSL or the AnyConnect client is denied.

Another example of enforcing dial-in allow access or deny access is available in the Tech Note *ASA/PIX: Mapping VPN Clients to VPN Group Policies Through LDAP Configuration Example* at the following URL:

http://www.cisco.com/en/US/products/ps6120/products_configuration_example09186a008089149d.shtml

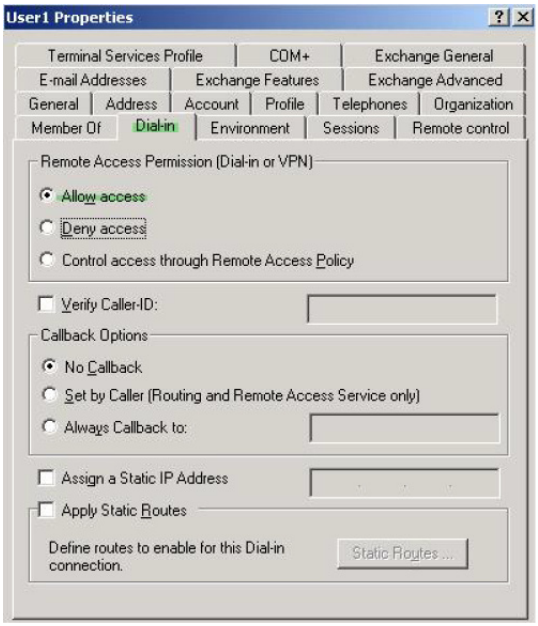
To configure the user attributes on the AD/LDAP server, perform the following steps:

Step 1 Right-click the user.

The Properties dialog box appears.

Step 2 Click the **Dial-in** tab, then click the **Allow Access** radio button (Figure 13-8).

Figure 13-8 AD/LDAP User1 - Allow Access



Note If you select the Control access through the Remote Access Policy option, then a value is not returned from the server, and the permissions that are enforced are based on the internal group policy settings of the ASA.

Step 3 Create an attribute map to allow both an IPsec and AnyConnect connection, but deny a clientless SSL connection.

The following example shows how to create the map tunneling_protocols, and map the AD attribute msNPAllowDialin used by the Allow Access setting to the Cisco attribute Tunneling-Protocols using the **map-name** command, and add map values with the **map-value** command:

```
ciscoasa(config)# ldap attribute-map tunneling_protocols
ciscoasa(config-ldap-attribute-map)# map-name msNPAllowDialin Tunneling-Protocols
ciscoasa(config-ldap-attribute-map)# map-value msNPAllowDialin FALSE 48
ciscoasa(config-ldap-attribute-map)# map-value msNPAllowDialin TRUE 4
```

Step 4 Associate the LDAP attribute map to the AAA server.

The following example enters the aaa server host configuration mode for the host 10.1.1.2, in the AAA server group MS_LDAP, and associates the attribute map tunneling_protocols that you created in Step 2:

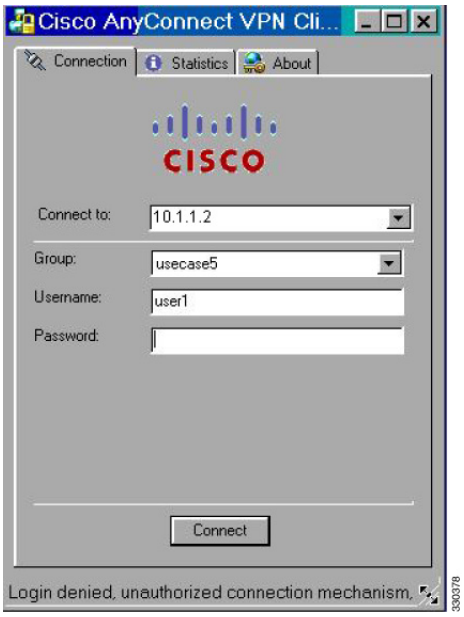
```
ciscoasa(config)# aaa-server MS_LDAP host 10.1.1.2
ciscoasa(config-aaa-server-host)# ldap-attribute-map tunneling_protocols
```

Step 5 Verify that the attribute map works as configured.

Step 6 Try connections using clientless SSL, the AnyConnect client, and the IPsec client. The clientless and AnyConnect connections should fail, and the user should be informed that an unauthorized connection mechanism was the reason for the failed connection. The IPsec client should connect because IPsec is an allowed tunneling protocol according to the attribute map (see Figure 13-9 and Figure 13-10).

Figure 13-9 Login Denied Message for Clientless User

Figure 13-10 Login Denied Message for AnyConnect Client User



Enforcing Logon Hours and Time-of-Day Rules

The following example shows how to configure and enforce the hours that a clientless SSL user (such as a business partner) is allowed to access the network.

On the AD server, use the Office field to enter the name of the partner, which uses the physicalDeliveryOfficeName attribute. Then we create an attribute map on the ASA to map that attribute to the Cisco attribute Access-Hours. During authentication, the ASA retrieves the value of physicalDeliveryOfficeName and maps it to Access-Hours.

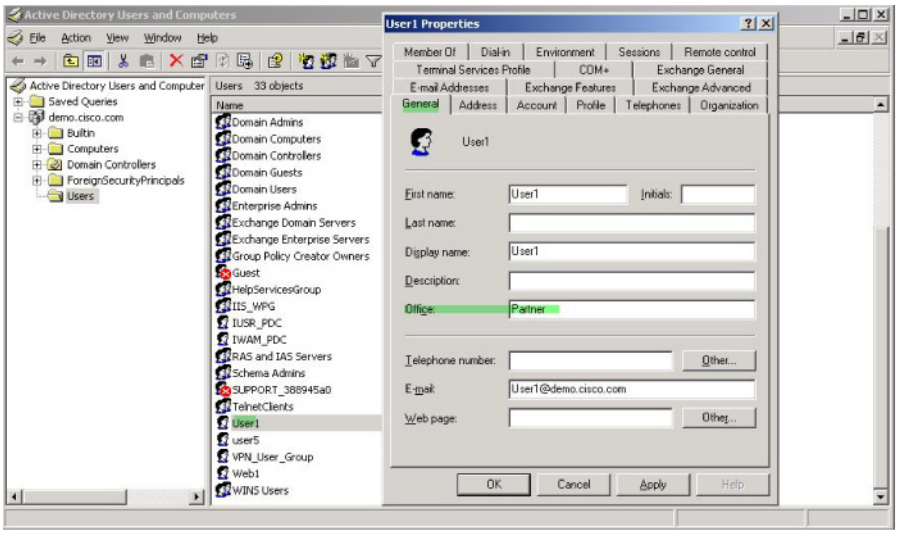
To configure the user attributes on the AD /LDAP server, perform the following steps:

Step 1 Select the user, and right-click **Properties**.

The Properties dialog box appears (see Figure 13-11).

Step 2 Click the **General** tab.

Figure 13-11 Active Directory Properties Dialog Box



Step 3 Create an attribute map.

The following example shows how to create the attribute map access_hours and map the AD attribute physicalDeliveryOfficeName used by the Office field to the Cisco attribute Access-Hours.

```
ciscoasa(config)# ldap attribute-map access_hours
ciscoasa(config-ldap-attribute-map)# map-name physicalDeliveryOfficeName Access-Hours
```

Step 4 Associate the LDAP attribute map to the AAA server.

The following example enters the aaa server host configuration mode for the host 10.1.1.2, in the AAA server group MS_LDAP, and associates the attribute map access_hours that you created in Step 3:

```
ciscoasa(config)# aaa-server MS_LDAP host 10.1.1.2
ciscoasa(config-aaa-server-host)# ldap-attribute-map access_hours
```

Step 5 Configure time ranges for each value allowed on the server.

The following example configures Partner access hours from 9am to 5pm Monday through Friday:

```
ciscoasa(config)# time-range Partner
ciscoasa(config-time-range)# periodic weekdays 09:00 to 17:00
```

Configuring Authorization with LDAP for VPN

After LDAP authentication for VPN access has succeeded, the ASA queries the LDAP server, which returns LDAP attributes. These attributes generally include authorization data that applies to the VPN session.

You may require authorization from an LDAP directory server that is separate and distinct from the authentication mechanism. For example, if you use an SDI or certificate server for authentication, no authorization information is passed back. For user authorizations in this case, you can query an LDAP directory after successful authentication, accomplishing authentication and authorization in two steps.

To set up VPN user authorization using LDAP, perform the following steps.

Detailed Steps

	Command	Purpose

Step 1	**aaa-server** *server_group* **protocol** {**kerberos** | **ldap** | **nt** | **radius** | **sdi** | **tacacs+**}
	
	Example:
	`ciscoasa(config)# aaa-server servergroup1 protocol ldap`
	`ciscoasa(config-aaa-server-group)`
		Creates a AAA server group.

Step 2	**tunnel-group** *groupname* **Example:** `ciscoasa(config)# tunnel-group remotegrp`	Creates an IPsec remote access tunnel group named remotegrp.
Step 3	**tunnel-group** *groupname* **general-attributes** **Example:** `ciscoasa(config)# tunnel-group remotegrp general-attributes`	Associates the server group and the tunnel group.
Step 4	**authorization-server-group** *group-tag* **Example:** `ciscoasa(config-general)# authorization-server-group ldap_dir_1`	Assigns a new tunnel group to a previously created AAA server group for authorization.

Examples

While there are other authorization-related commands and options available for specific requirements, the following example shows commands for enabling user authorization with LDAP. The example then creates an IPsec remote access tunnel group named remote-1, and assigns that new tunnel group to the previously created ldap_dir_1 AAA server group for authorization:

```
ciscoasa(config)# tunnel-group remote-1 type ipsec-ra
ciscoasa(config)# tunnel-group remote-1 general-attributes
ciscoasa(config-general)# authorization-server-group ldap_dir_1
ciscoasa(config-general)#
```

After you complete this configuration work, you can then configure additional LDAP authorization parameters such as a directory password, a starting point for searching a directory, and the scope of a directory search by entering the following commands:

```
ciscoasa(config)# aaa-server ldap_dir_1 protocol ldap
ciscoasa(config-aaa-server-group)# aaa-server ldap_dir_1 host 10.1.1.4
ciscoasa(config-aaa-server-host)# ldap-login-dn obscurepassword
ciscoasa(config-aaa-server-host)# ldap-base-dn starthere
ciscoasa(config-aaa-server-host)# ldap-scope subtree
ciscoasa(config-aaa-server-host)#
```

Example of Creating a Group Policy for a Local User

Prerequisites

This procedure describes how to edit an existing user.For more information see "Adding a User Account to the Local Database" section on page 33-3.

Detailed Steps

PART 2

Configuring a Clientless SSL VPN

Introduction to Clientless SSL VPN

September 13, 2013

Introduction to Clientless SSL VPN

Clientless SSL VPN enables end users to securely access resources on the corporate network from anywhere using an SSL-enabled Web browser. The user first autehnticates with a Clientless SSL VPN gateway, which then allows the user to access pre-configured network resources.

Note Security contexts (also called firewall multimode) and Active/Active stateful failover are not supported when Clientless SSL VPN is enabled.

Clientless SSL VPN creates a secure, remote-access VPN tunnel to an ASA using a Web browser without requiring a software or hardware client. It provides secure and easy access to a broad range of Web resources and both web-enabled and legacy applications from almost any device that can connect to the Internet via HTTP. They include:

- Internal websites.
- Web-enabled applications.
- NT/Active Directory file shares.
- email proxies, including POP3S, IMAP4S, and SMTPS.
- Microsoft Outlook Web Access Exchange Server 2000, 2003, and 2007.
- Microsoft Web App to Exchange Server 2010 in 8.4(2) and later.
- Application Access (smart tunnel or port forwarding access to other TCP-based applications)

Clientless SSL VPN uses Secure Sockets Layer Protocol and its successor, Transport Layer Security (SSL/TLS1) to provide the secure connection between remote users and specific, supported internal resources that you configure at an internal server. The ASA recognizes connections that must be proxied, and the HTTP server interacts with the authentication subsystem to authenticate users.

The network administrator provides access to resources by users of Clientless SSL VPN sessions on a group basis. Users have no direct access to resources on the internal network.

Prerequisites

See the *Supported VPN Platforms, Cisco ASA 5500 Series* for the platforms and browsers supported by ASA Release 9.0.

Guidelines and Limitations

- ActiveX pages require that you enable ActiveX Relay or enter **activex-relay** on the associated group policy. If you do so or assign a smart tunnel list to the policy, and the browser proxy exception list on the endpoint specifies a proxy, the user must add a "shutdown.webvpn.relay." entry to that list.
- The ASA does not support clientless access to Windows Shares (CIFS) Web Folders from Windows 7, Vista, Internet Explorer 8 to 10, Mac OS X, or Linux.
- Certificate authentication, including the DoD Common Access Card and SmartCard, works with the Safari keychain only.
- The ASA does not support DSA or RSA certificates for Clientless SSL VPN connections.
- Some domain-based security products have requirements beyond those requests that originate from the ASA.
- Configuration control inspection and other inspection features under the Modular Policy Framework are not supported.
- The *vpn-filter* command under group policy is for client-based access and is not supported. *Filter* under Clientless SSL VPN mode in group policy is for clientless-based access only.
- Neither NAT or PAT is applicable to the client.
- The ASA does not support the use of the QoS rate-limiting commands, such as **police** or **priority-queue**.
- The ASA does not support the use of connection limits, checking via the static or the Modular Policy Framework **set connection** command.
- Some components of Clientless SSL VPN require the Java Runtime Environment (JRE).With Mac OS X v10.7 and later Java is not installed by default. For details of how to install Java on Mac OS X, see http://java.com/en/download/faq/java_mac.xml.

When you have several group policies configured for the clientless portal, they are displayed in a drop-down on the logon page. When the first group policy in the list requires a certificate, then the user must have a matching certificate. If some of your group policies do not use certificates, you must configure the list to display a non-certificate policy first. Alternatively, you may want to create a dummy group policy with the name "0-Select-a-group."

Tip You can control which policy is displayed first by naming your group polices alphabetically, or prefix them with numbers. For example, 1-AAA, 2-Certificate.

Basic Clientless SSL VPN Configuration

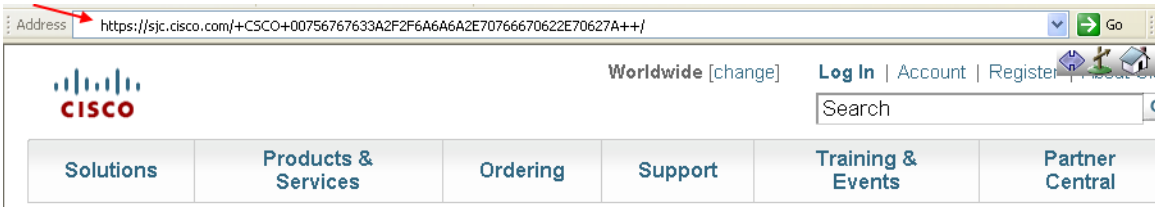
September 13, 2013

Clientless SSL VPN Security Precautions

By default, the ASA allows all portal traffic to all Web resources (for example HTTPS, CIFS, RDP, and plug-ins). Clientless SSL VPN rewrites each URL to one that is meaningful only to the ASA. The user cannot use this URL to confirm that they are connected to the website they requested. To avoid placing users at risk from phishing websites, assign a Web ACL to the policies configured for clientless access—group policies, dynamic access policies, or both—to control traffic flows from the portal. Cisco recommends switching off URL Entry on these policies to prevent user confusion over what is accessible.

Figure 15-1 Example URL Entered by User

Figure 15-2 Same URL Rewritten by Security Appliance and Displayed in Browser Window



Switching Off URL Entry on the Portal Page

The portal page opens when the user establishes a browser-based connection.

Prerequisites

Configure a group policy for all users who require Clientless SSL VPN access, and enable Clientless SSL VPN only for that group policy.

DETAILED STEPS

	Command	Purpose
Step 1	`webvpn`	Switches to group policy Clientless SSL VPN configuration mode.
Step 2	`url-entry`	Controls the ability of the user to enter any HTTP/HTTPS URL.
Step 3	(Optional) `url-entry disable`	Switches off URL Entry.

Verifying Clientless SSL VPN Server Certificates

When connecting to a remote SSL-enabled server through Clientless SSL VPN, it is important to know that you can trust the remote server, and that it is in fact the server you are trying to connect to. ASA 9.0 introduced support for SSL server certificate verification against a list of trusted certificate authority (CA) certificates for Clientless SSL VPN.

When connecting to a remote server with a Web browser using the HTTPS protocol, the server provides a digital certificate signed by a certificate authority (CA) to identify itself. Web browsers include a collection of CA certificates which are used to verify the validity of the server certificate. This is a form of public key infrastructure (PKI).

The ASA provides trusted pool certificate management facilities in the form of a trustpools. This can be thought of as a special case of trustpoint representing multiple known CA certificates. The ASA includes a default bundle of certificates, similar to that provided with Web browsers. It is inactive until activated by the administrator by issuing the **`crypto ca import default`** command.

Note ASA trustpools are similar but not identical to Cisco IOS trustpools.

Configuring Browser Access to Plug-ins

The following sections describe the integration of browser plug-ins for Clientless SSL VPN browser access:

- Preparing the Security Appliance for a Plug-in, page 15-4
- Installing Plug-ins Redistributed by Cisco, page 15-5
- Providing Access to a Citrix XenApp Server, page 15-6

A browser plug-in is a separate program that a Web browser invokes to perform a dedicated function, such as connect a client to a server within the browser window. The ASA lets you import plug-ins for download to remote browsers in Clientless SSL VPN sessions. Of course, Cisco tests the plug-ins it redistributes, and in some cases, tests the connectivity of plug-ins we cannot redistribute. However, we do not recommend importing plug-ins that support streaming media at this time.

The ASA does the following when you install a plug-in onto the flash device:

- (Cisco-distributed plug-ins only) Unpacks the jar file specified in the URL.
- Writes the file to the ASA file system.
- Populates the drop-down menu next to the URL attributes in ASDM.
- Enables the plug-in for all future Clientless SSL VPN sessions, and adds a main menu option and an option to the drop-down menu next to the Address field of the portal page.

Table 15-1 shows the changes to the main menu and Address field of the portal page when you add the plug-ins described in the following sections.

Table 15-1 Effects of Plug-ins on the Clientless SSL VPN Portal Page

Plug-in	Main Menu Option Added to Portal Page	Address Field Option Added to Portal Page
ica	Citrix MetaFrame Services	ica://
rdp	Terminal Servers	rdp://
rdp2*	Terminal Servers Vista	rdp2://
ssh,telnet	Secure Shell	ssh://
	Telnet Services (supporting v1 and v2)	telnet://
vnc	Virtual Network Computing services	vnc://

* Not a recommended plug-in.

When the user in a Clientless SSL VPN session clicks the associated menu option on the portal page, the portal page displays a window to the interface and displays a help pane. The user can select the protocol displayed in the drop-down menu and enter the URL in the Address field to establish a connection.

The plug-ins support single sign-on (SSO). Refer to the "Configuring SSO with the HTTP Form Protocol" section on page 19-12 for implementation details.

Prerequisites

- Clientless SSL VPN must be enabled on the ASA to provide remote access to the plug-ins.
- To configure SSO support for a plug-in, you install the plug-in, add a bookmark entry to display a link to the server, and specify SSO support when adding the bookmark.
- The minimum access rights required for remote use belong to the guest privilege mode.
- Plug-ins require ActiveX or Oracle Java Runtime Environment (JRE); see the compatibility matrix for version requirements.

Restrictions

Note The remote desktop protocol plug-in does not support load balancing with a session broker. Because of the way the protocol handles the redirect from the session broker, the connection fails. If a session broker is not used, the plug-in works.

- The plug-ins support single sign-on (SSO). They use the *same* credentials entered to open the Clientless SSL VPN session. Because the plug-ins do not support macro substitution, you do not have the options to perform SSO on different fields such as the internal domain password or on an attribute on a RADIUS or LDAP server.
- A stateful failover does not retain sessions established using plug-ins. Users must reconnect following a failover.
- If you use stateless failover instead of stateful failover, clientless features such as bookmarks, customization, and dynamic access-policies are not synchronized between the failover ASA pairs. In the event of a failover, these features do not work.

Preparing the Security Appliance for a Plug-in

Before installing a plug-in, prepare the ASA as follows:

Prerequisites

Ensure that Clientless SSL VPN is enabled on an ASA interface.

Restrictions

Do not specify an IP address as the common name (CN) for the SSL certificate. The remote user attempts to use the FQDN to communicate with the ASA. The remote PC must be able to use DNS or an entry in the System32\drivers\etc\hosts file to resolve the FQDN.

DETAILED STEPS

	Command	Purpose
Step 1	`show running-config`	Shows whether Clientless SSL VPN is enabled on the ASA.
Step 2	Install an SSL certificate onto the ASA interface	Provides a fully-qualified domain name (FQDN) for remote user connection.

Go to the section that identifies the type of plug-in to provide for Clientless SSL VPN access.

- Installing Plug-ins Redistributed by Cisco, page 15-5
- Providing Access to a Citrix XenApp Server, page 15-6

Installing Plug-ins Redistributed by Cisco

Cisco redistributes the following open-source, Java-based components to be accessed as plug-ins for Web browsers in Clientless SSL VPN sessions.

Prerequisites

Ensure Clientless SSL VPN is enabled on an interface on the ASA. To do so, enter the **show running-config** command.

Table 15-2 Plug-ins Redistributed by Cisco

Protocol	Description	Source of Redistributed Plug-in *
RDP	Accesses Microsoft Terminal Services hosted by Windows Vista and Windows 2003 R2. Supports Remote Desktop ActiveX Control. We recommend using this plug-in that supports both RDP and RDP2. Only versions up to 5.1 of the RDP and RDP2 protocols are supported. Version 5.2 and later are not supported.	http://properjavardp.sourceforge.net/
RDP2	Accesses Microsoft Terminal Services hosted by Windows Vista and Windows 2003 R2. Supports Remote Desktop ActiveX Control. **Note** This legacy plug-in supports only RDP2. We do not recommend using this plug-in; instead, use the RDP plug-in above.	http://properjavardp.sourceforge.net/
SSH	The Secure Shell-Telnet plug-in lets the remote user establish a Secure Shell (v1 or v2) or Telnet connection to a remote computer. **Note** Because keyboard-interactive authentication is not supported by JavaSSH, it cannot be supported with SSH plugin (used to implement different authentication mechanisms).	http://javassh.org/
VNC	The Virtual Network Computing plug-in lets the remote user use a monitor, keyboard, and mouse to view and control a computer with remote desktop sharing (also known as VNC server or service) turned on. This version changes the default color of the text and contains updated French and Japanese help files.	http://www.tightvnc.com/

* Consult the plug-in documentation for information on deployment configuration and restrictions.

These plug-ins are available on the Cisco Adaptive Security Appliance Software Download site.

DETAILED STEPS

Step 1 .

Note The ASA does not retain the **import webvpn plug-in protocol** command in the configuration. Instead, it loads the contents of the csco-config/97/plugin directory automatically. A secondary ASA obtains the plug-ins from the primary ASA.

	Command	Purpose

Step 1	**import webvpn plug-in** *protocol* [**rdp** | **rdp2** | [**ssh** | **telnet**] | **vnc**] *URL*
	
	Example:
	`ciscoasa# import webvpn plug-in protocol ssh,telnet tftp://local_tftp_server/plugins/ssh-plugin.jar`
	
	`Accessing tftp://local_tftp_server/plugins/ssh-plugin.jar...!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!`
	`Writing file disk0:/csco_config/97/plugin/ssh...`
	`!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!`
	`238510 bytes copied in 3.650 secs (79503 bytes/sec)`
		Installs the plug-in onto the flash device of the ASA. *protocol* is one of the following values: **ssh**, **telnet** provides plug-in access to *both* Secure Shell and Telnet services.
		
		Note Do not enter this command once for SSH and once for Telnet. When typing the ssh,telnet string, do not insert a space.
		
		URL is the remote path to the plug-in .jar file. Enter the hostname or address of the TFTP or FTP server and the path to the plug-in.

Step 2	(Optional) **revert webvpn plug-in protocol** *protocol* **Example:** `ciscoasa# revert webvpn plug-in protocol rdp`	Switches off and removes Clientless SSL VPN support for a plug-in, as well as removing it from the flash drive of the ASA.

Providing Access to a Citrix XenApp Server

As an example of how to provide Clientless SSL VPN browser access to third-party plug-ins, this section describes how to add Clientless SSL VPN support for the Citrix XenApp Server Client.

With a Citrix plug-in installed on the ASA, Clientless SSL VPN users can use a connection to the ASA to access Citrix XenApp services.

A stateful failover does not retain sessions established using the Citrix plug-in. Citrix users must reauthenticate after failover.

To provide access to the Citrix plug-in, follow the procedures in the following sections.

- Preparing the Citrix XenApp Server for Clientless SSL VPN Access
- Creating and Installing the Citrix Plug-in

Preparing the Citrix XenApp Server for Clientless SSL VPN Access

You must configure the Citrix Web Interface software to operate in a mode that does not use the (Citrix) “secure gateway.” Otherwise, the Citrix client cannot connect to the Citrix XenApp Server.

Note If you are not already providing support for a plug-in, you must follow the instructions in the “Preparing the Security Appliance for a Plug-in” section on page 15-4 before using this section.

Creating and Installing the Citrix Plug-in

DETAILED STEPS

Step 1 Download the ica-plugin.zip file from the Cisco Software Download website.

This file contains files that Cisco customized for use with the Citrix plug-in.

Step 2 Download the Citrix Java client from the Citrix site.

In the download area of the Citrix website, select **Citrix Receiver**, and **Receiver for Other Platforms**, and click **Find**. Click the **Receiver for Java** hyperlink and download the archive..

Step 3 Extract the following files from the archive, and then add them to the ica-plugin.zip file:

- JICA-configN.jar
- JICAEngN.jar

Step 4 Ensure the EULA included with the Citrix Java client grants you the rights and permissions to deploy the client on your Web servers.

Step 5 Install the plug-in by using ASDM, or entering the following CLI command in privileged EXEC mode:

import webvpn plug-in protocol ica *URL*

URL is the hostname or IP address and path to the ica-plugin.zip file.

Note Adding a bookmark is required to provide SSO support for Citrix sessions. We recommend that you use URL parameters in the bookmark the provide convenient viewing, for example:

ica://10.56.1.114/?DesiredColor=4&DesiredHRes=1024&DesiredVRes=768

Step 6 Establish an SSL VPN clientless session and click the bookmark or enter the URL for the Citrix server.

Use the Client for Java Administrator's Guide as needed.

Viewing the Plug-ins Installed on the Security Appliance

DETAILED STEPS

	Command	Purpose

Step 1	**show import webvpn plug**
	
	Example:
	`ciscoasa# show import webvpn plug`
	`ssh`
	`rdp`
	`vnc`
	`ica`
		Lists the Java-based client applications available to users of Clientless SSL VPN.

Step 2	**show import webvpn plug detail**
	
	Example:
	`hostname show import webvpn plug`
	`post GXN2BIGGOAOkBMibDQsMu2GWZ3Q= Tues, 29 Apr 2008 19:57:03 GMT`
	`rdp fHeyReIOUwDCgAL9HdTs PnjdBoo= Tues, 15 Sep 2009 23:23:56 GMT`
	`rdp2 shw8c22T2SsILLk6zyCd6H6VOz8= Wed, 11 Feb 2009 21:17:54 GMT`
		Includes hash and date of the plug-in.

Configuring Port Forwarding

The following sections describe port forwarding and how to configure it:

- Information About Port Forwarding, page 15-8
- Configuring DNS for Port Forwarding
- Making Applications Eligible for Port ForwardingAssigning a Port Forwarding List
- Automating Port Forwarding

Information About Port Forwarding

Port forwarding lets users access TCP-based applications over a Clientless SSL VPN connection. Such applications include the following:

- Lotus Notes
- Microsoft Outlook
- Microsoft Outlook Express
- Perforce
- Sametime
- Secure FTP (FTP over SSH)
- SSH

- Telnet
- Windows Terminal Service
- XDDTS

Other TCP-based applications may also work, but we have not tested them. Protocols that use UDP do not work.

Port forwarding is the legacy technology for supporting TCP-based applications over a Clientless SSL VPN connection. You may choose to use port forwarding because you have built earlier configurations that support this technology.

Consider the following alternatives to port forwarding:

- Smart tunnel access offers the following advantages to users:
  - Smart tunnel offers better performance than plug-ins.
  - Unlike port forwarding, smart tunnel simplifies the user experience by not requiring the user connection of the local application to the local port.
  - Unlike port forwarding, smart tunnel does not require users to have administrator privileges.
- Unlike port forwarding and smart tunnel access, a plug-in does not require the client application to be installed on the remote computer.

When configuring port forwarding on the ASA, you specify the port the application uses. When configuring smart tunnel access, you specify the name of the executable file or its path.

Prerequisites

- The remote host must be running a 32-bit version of one of the following:
  - Microsoft Windows Vista, Windows XP SP2 or SP3; or Windows 2000 SP4.
  - Apple Mac OS X 10.4 or 10.5 with Safari 2.0.4(419.3).
  - Fedora Core 4
- The remote host must also be running Oracle Java Runtime Environment (JRE) 5 or later.
- Browser-based users of Safari on Mac OS X 10.5.3 must identify a client certificate for use with the URL of the ASA, once with the trailing slash and once without it, because of the way Safari interprets URLs. For example,
  - https://example.com/
  - https://example.com

  For details, go to the Safari, Mac OS X 10.5.3: Changes in client certificate authentication.
- Users of Microsoft Windows Vista or later who use port forwarding or smart tunnels must add the URL of the ASA to the Trusted Site zone. To access the Trusted Site zone, they must start Internet Explorer and choose the **Tools > Internet Options > Security** tab. Vista (or later) users can also switch off Protected Mode to facilitate smart tunnel access; however, we recommend against this method because it increases the computer's vulnerability to attack.
- Ensure Oracle Java Runtime Environment (JRE) 1.5.x or later is installed on the remote computers to support port forwarding (application access) and digital certificates. If JRE 1.4.x is running and the user authenticates with a digital certificate, the application fails to start because JRE cannot access the Web browser certificate store.

Restrictions

- Port forwarding supports only TCP applications that use static TCP ports. Applications that use dynamic ports or multiple TCP ports are not supported. For example, SecureFTP, which uses port 22, works over Clientless SSL VPN port forwarding, but standard FTP, which uses ports 20 and 21, does not.
- Port forwarding does not support protocols that use UDP.
- Port forwarding does not support Microsoft Outlook Exchange (MAPI) proxy. However, you can configure smart tunnel support for Microsoft Office Outlook in conjunction with Microsoft Outlook Exchange Server.
- A stateful failover does not retain sessions established using Application Access (either port forwarding or smart tunnel access). Users must reconnect following a failover.
- Port forwarding does not support connections to personal digital assistants.
- Because port forwarding requires downloading the Java applet and configuring the local client, and because doing so requires administrator permissions on the local system, it is unlikely that users will be able to use applications when they connect from public remote systems.

  The Java applet displays in its own window on the end user HTML interface. It shows the contents of the list of forwarded ports available to the user, as well as which ports are active, and amount of traffic in bytes sent and received.
- The port forwarding applet displays the local port and the remote port as the same when the local IP address 127.0.0.1 is being used and cannot be updated by the Clientless SSL VPN connection from the ASA. As a result, the ASA creates new IP addresses 127.0.0.2, 127.0.0.3, and so on for local proxy IDs. Because you can modify the hosts file and use different loopbacks, the remote port is used as the local port in the applet. To connect, you can use Telnet with the hostname, without specifying the port. The correct local IP addresses are available in the local hosts file.

Configuring DNS for Port Forwarding

Port forwarding forwards the domain name of the remote server or its IP address to the ASA for resolution and connection. In other words, the port forwarding applet accepts a request from the application and forwards it to the ASA. The ASA makes the appropriate DNS queries and establishes the connection on behalf of the port forwarding applet. The port forwarding applet only makes DNS queries to the ASA. It updates the host file so that when a port forwarding application attempts a DNS query, the query redirects to a loopback address. Configure the ASA to accept the DNS requests from the port forwarding applet as follows:

	Command	Purpose
Step 1	**dns server-group**	Enters the dns server-group mode. Configures a DNS server group named example.com.

Step 2	**domain-name**
	
	Example:
	`ciscoasa(config)# dns server-group example.com`
	`ciscoasa(config-dns-server-group)# domain-name example.com`
		Specifies the domain name. The default domain-name setting is DefaultDNS.

Step 3	**name-server** **Example:** `ciscoasa(config-dns-server-group)# name-server 192.168.10.10`	Resolves the domain name to an IP address.
Step 4	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 5	**tunnel-group webvpn**	Switches to tunnel-group Clientless SSL VPN configuration mode.

Step 6	(Required only if you are using a domain name other than the default one [DefaultDNS].)
	dns-group
	
	Example:
	`asa2(config-dns-server-group)# exit`
	`asa2(config)# tunnel-group DefaultWEBVPNGroup webvpn-attributes`
	`asa2(config-tunnel-webvpn)# dns-group example.com`
		Specifies the domain name that the tunnel groups will use. By default, the security appliance assigns the default Clientless SSL VPN group as the default tunnel group for clientless connections. Follow this instruction if the ASA uses that tunnel group to assign settings to the clientless connections. Otherwise, follow this step for each tunnel configured for clientless connections.

Making Applications Eligible for Port Forwarding

The Clientless SSL VPN configuration of each ASA supports *port forwarding lists*, each of which specifies local and remote ports used by the applications for which to provide access. Because each group policy or username supports only one port forwarding list, you must group each set of ca supported into a list. To display the port forwarding list entries already present in the ASA configuration, enter the following commands:

DETAILED STEPS

	Command	Purpose
Step 1	`show run webvpn port-forward`	Displays the port forwarding list entries already present in the ASA configuration.
Step 2	`webvpn`	Switches to Clientless SSL VPN configuration mode.

	Command	Purpose

Step 3	**port-forward {<list name> <local port> <remote server> <remote port> <description>}**
	
	Example:
	`ciscoasa(config)# webvpn`
	`ciscoasa(config-webvpn)# port-forward SalesGroupPorts 20143 IMAP4Sserver 143 Get Mail`
	`ciscoasa(config-webvpn)# port-forward SalesGroupPorts 20025 SMTPSserver 25 Send Mail`
	`ciscoasa(config-webvpn)# port-forward SalesGroupPorts 20022 DDTSserver 22 DDTS over SSH`
	`ciscoasa(config-webvpn)# port-forward SalesGroupPorts 20023 Telnetserver 23 Telnet`
		Adds a port forwarding entry to a list.
		• *list_name*—Name for a set of applications (technically, a set of forwarded TCP ports) for users of Clientless SSL VPN sessions to access. The ASA creates a list using the name you enter if it does not recognize it. Otherwise, it adds the port forwarding entry to the list. Maximum 64 characters.
		• *local_port*—Port that listens for TCP traffic for an application running on the user's computer. You can use a local port number only once for each port forwarding list. Enter a port number in the range 1-65535 or port name. To avoid conflicts with existing services, use a port number greater than 1024.
		• *remote_server*—DNS name or IP address of the remote server for an application. The IP address can be in IPv4 or IPv6 format. We recommend a DNS name so that you do not have to configure the client applications for a specific IP address.
		Note The DNS name must match the one assigned to the tunnel group to establish the tunnel and resolve to an IP address, per the instructions in the previous section. The default setting for both the **domain-name group** and **dns-group** commands described in that section is DefaultDNS.
		• *remote_port*—Port to connect to for this application on the remote server. This is the actual port the application uses. Enter a port number in the range 1-65535 or port name.
		• *description*—Application name or short description that displays on the end user Port Forwarding Java applet screen. Maximum 64 characters.
		
		Shows how to create a port forwarding list called SalesGroupPorts that provides access to these applications.

Step 4	(Optional) **no port-forward <list name> <local port>**	Removes an entry from the list, specifying both the list and the local port.

Following the configuration of a port forwarding list, assign the list to group policies or usernames, as described in the next section.

Assigning a Port Forwarding List

You can add or edit a named list of TCP applications to associate with users or group policies for access over Clientless SSL VPN connections. For each group policy and username, you can configure Clientless SSL VPN to do one of the following:

- Start port forwarding access automatically upon user login.

Note These options are mutually exclusive for each group policy and username. Use only one.

Prerequisites

Before initiating the **port-forward enable <list name>** command, the user is required to start port forwarding manually, using **Application Access > Start Applications** on the Clientless SSL VPN portal page.

DETAILED STEPS

These commands are available to each group policy and username. The configuration of each group policy and username supports only one of these commands at a time, so when you enter one, the ASA replaces the one present in the configuration of the group policy or username in question with the new one, or in the case of the last command, simply removes the **port-forward** command from the group policy or username configuration.

	Command	Purpose
Step 1	`port-forward auto-start` **<list name>**	Starts port forwarding automatically upon user login.
	`port-forward enable` **<list name>**	Enables port forwarding upon user login.
	`port-forward disable`	Prevents port forwarding.
	`no port-forward` [`auto-start` **<list name>** | `enable` **<list name>** | `disable`]	Removes a **port-forward** command from the group policy or username configuration, which then inherits the [**no**] **port-forward** command from the default group policy. The keywords following the **no port-forward** command are optional; however, they restrict the removal to the named **port-forward** command.

Step 7

Automating Port Forwarding

To start port forwarding automatically upon user login, enter the following commands:

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**group-policy webvpn** **username webvpn**	Switches to group-policy Clientless SSL VPN configuration mode. Switches to username Clientless SSL VPN configuration mode.
Step 3	**port-forward auto-start <list name>** **Example:** ciscoasa(config-group-policy)# **webvpn** ciscoasa(config-group-webvpn)# **port-forward auto-start apps1**	Starts port forwarding automatically upon user login. *list_name* names the port forwarding list already present in the ASA Clientless SSL VPN configuration. You cannot assign more than one port forwarding list to a group policy or username. Assigns the port forwarding list named apps1 to the group policy.
Step 4	**show run webvpn port-forward**	Displays the port forwarding list entries present in the ASA configuration.
Step 5	(Optional) **no port-forward**	Removes the **port-forward** command from the group policy or username and reverts to the default.

Enabling and Switching off Port Forwarding

By default, port forwarding is switched off.

DETAILED STEPS

	Command	Purpose
Step 1	**port-forward [enable <list name> | disable]** **Example:** ciscoasa(config-group-policy)# **webvpn** ciscoasa(config-group-webvpn)# **port-forward enable apps1**	Enables port forwarding. You do not have to start port forwarding manually if you entered **port-forward auto-start** ***list_name*** from the previous table. *list_name* is the name of the port forwarding list already present in the ASA Clientless SSL VPN configuration. You cannot assign more than one port forwarding list to a group policy or username. Assigns the port forwarding list named apps1 to the group policy.
Step 2	**show running-config port-forward**	Displays the port forwarding list entries.

	Command	Purpose
Step 3	(Optional) `no port-forward`	Removes the **port-forward** command from the group policy or username and reverts to the default.
Step 4	(Optional) `port-forward disable`	Switches off port forwarding.

Configuring File Access

Clientless SSL VPN serves remote users with HTTPS portal pages that interface with proxy CIFS and/or FTP clients running on the ASA. Using either CIFS or FTP, Clientless SSL VPN provides users with network access to the files on the network, to the extent that the users meet user authentication requirements and the file properties do not restrict access. The CIFS and FTP clients are transparent; the portal pages delivered by Clientless SSL VPN provide the appearance of direct access to the file systems.

When a user requests a list of files, Clientless SSL VPN queries the server designated as the master browser for the IP address of the server containing the list. The ASA gets the list and delivers it to the remote user on a portal page.

Clientless SSL VPN lets the user invoke the following CIFS and FTP functions, depending on user authentication requirements and file properties:

- Navigate and list domains and workgroups, servers within a domain or workgroup, shares within a server, and files within a share or directory.
- Create directories.
- Download, upload, rename, move, and delete files.

The ASA uses a master browser, WINS server, or DNS server, typically on the same network as the ASA or reachable from that network, to query the network for a list of servers when the remote user clicks **Browse Networks** in the menu of the portal page or on the toolbar displayed during the Clientless SSL VPN session.

The master browser or DNS server provides the CIFS/FTP client on the ASA with a list of the resources on the network, which Clientless SSL VPN serves to the remote user.

Note Before configuring file access, you must configure the shares on the servers for user access.

CIFS File Access Requirement and Limitation

To access `\\server\share\subfolder\personal` *`folder`*, the user must have a minimum of read permission for all parent folders, including the share itself.

Use **Download** or **Upload** to copy and paste files to and from CIFS directories and the local desktop. The Copy and Paste buttons are intended for remote to remote actions only, not local to remote, or remote to local.

The CIFS browse server feature does not support double-byte character share names (share names exceeding 13 characters in length). This only affects the list of folders displayed, and does not affect user access to the folder. As a workaround, you can pre-configure the bookmark(s) for the CIFS folder(s) that use double-byte share names, or the user can enter the URL or bookmark of the folder in the format cifs://server/<long-folder-name>. For example:

```
cifs://server/Do you remember?
cifs://server/Do%20you%20remember%3F
```

Adding Support for File Access

Configure file access as follows:

Note The procedure describes how to specify the master browser and WINS servers. As an alternative, you can use ASDM to configure URL lists and entries that provide access to file shares.

Adding a share in ASDM does not require a master browser or a WINS server. However, it does not provide support for the Browse Networks link. You can use a hostname or an IP address to refer to ServerA when entering the **nbns-server** command. If you use a hostname, the ASA requires a DNS server to resolve it to an IP address.

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**tunnel-group webvpn**	Switches to tunnel-group Clientless SSL VPN configuration mode.
Step 3	**nbns-server** {IPaddress | hostname} **[master] [timeout** *timeout*] **[retry** *retries*] **Example:** ciscoasa(config-tunnel-webvpn)# **nbns-server 192.168.1.20 master** ciscoasa(config-tunnel-webvpn)# **nbns-server 192.168.1.41** ciscoasa(config-tunnel-webvpn)# **nbns-server 192.168.1.47**	Browses a network or domain for each NetBIOS Name Server (NBNS). • **master** is the computer designated as the master browser. The master browser maintains the list of computers and shared resources. Any NBNS server you identify with this command without entering the master portion of the command must be a Windows Internet Naming Server (WINS). Specify the master browser first, then specify the WINS servers. You can specify up to three servers, including the master browser, for a connection profile. • *timeout* is the number of seconds the ASA waits before sending the query again, to the same server if it is the only one, or another server if there are more than one. The default timeout is 2 seconds; the range is 1 to 30 seconds. • *retries* is the number of times to retry queries to the NBNS server. The ASA recycles through the list of servers this number of times before sending an error message. The default value is 2; the range is 1 through 10.
Step 4	ciscoasa# **show tunnel-group webvpn-attributes**	Displays the NBNS servers already present in the connection profile configuration.

	Command	Purpose

Step 5	(Optional)
	character-encoding *charset*
	
	Example:
	`hostname(config)# webvpn`
	`ciscoasa(config-webvpn)# character-encoding shift_jis`
	`ciscoasa(config-webvpn)# customization DfltCustomization`
	`ciscoasa(config-webvpn-custom)# page style background-color:white`
		Specifies the character set to encode in Clientless SSL VPN portal pages delivered to remote users. By default, the encoding type set on the remote browser determines the character set for Clientless SSL VPN portal pages, so you need to set the character encoding only if it is necessary to ensure proper encoding on the browser.
		
		charset is a string consisting of up to 40 characters, and is equal to one of the valid character sets identified in http://www.iana.org/assignments/character-sets. You can use either the name or the alias of a character set listed on that page. Examples include iso-8859-1, shift_jis, and ibm850.
		
		Note The character-encoding and file-encoding values do not exclude the font family to be used by the browser. You need to complement the setting of one these values with the **page style** command in webvpn customization command mode to replace the font family if you are using Japanese Shift_JIS character encoding, as shown in the following example, or enter the **no page style** command in webvpn customization command mode to remove the font family.
		
		Sets the character-encoding attribute to support Japanese Shift_JIS characters, removes the font family, and retains the default background color.

Step 6	(Optional) **file-encoding {server-name | server-ip-address} *charset*** **Example:** `ciscoasa(config-webvpn)# file-encoding 10.86.5.174 cp860`	Specifies the encoding for Clientless SSL VPN portal pages from specific CIFS servers. Thus, you can use different file-encoding values for CIFS servers that require different character encodings. Sets the file-encoding attribute of the CIFS server 10.86.5.174 to support IBM860 (alias "CP860") characters.

For a complete description of these commands, see the *Cisco Security Appliance Command Reference*.

Ensuring Clock Accuracy for SharePoint Access

The Clientless SSL VPN server on the ASA uses cookies to interact with applications such as Microsoft Word on the endpoint. The cookie expiration time set by the ASA can cause Word to malfunction when accessing documents on a SharePoint server if the time on the ASA is incorrect. To prevent this malfunction, set the ASA clock properly. We recommend configuring the ASA to dynamically synchronize the time with an NTP server. For instructions, see the section on setting the date and time in the general operations configuration guide.

Virtual Desktop Infrastructure (VDI)

The ASA supports connections to Citrix and VMWare VDI servers.

- For Citrix, the ASA allows access through clientless portal to user's running Citrix Receiver.
- VMWare is configured as a (smart tunnel) application.

VDI servers can also be accessed through bookmarks on the Clientless Portal, like other server applications.

Limitations

- Authentication using certificates or Smart Cards is not supported for auto sign-on, since these forms of authentication do not allow the ASA in the middle.
- The XML service must be installed and configured on the XenApp and XenDesktop servers.
- Client certificate verifications, double Auth, internal passwords and CSD (all of CSD, not just Vault) are not supported when standalone mobile clients are used.

Citrix Mobile Support

A mobile user running the Citrix Receiver can connect to the Citrix server by:

- Connecting to the ASA with AnyConnect, and then connecting to the Citrix server.
- Connecting to the Citrix server through the ASA, without using the AnyConnect client. Logon credentials can include:
  - A connection profile alias (also referred to as a tunnel-group alias) in the Citrix logon screen. A VDI server can have several group policies, each with different authorization and connection settings.
  - An RSA SecureID token value, when the RSA server is configured. RSA support includes next token for an invalid entry, and also for entering a new PIN for an initial or expired PIN.

Supported Mobile Devices

- iPad—Citrix Receiver version 4.x or later
- iPhone/iTouch—Citrix Receiver version 4.x or later
- Android 2.x/3.x/4.0/4.1 phone—Citrix Receiver version 2.x or later
- Android 4.0 phone—Citrix Receiver version 2.x or later

Limitations

Certificate Limitations

- Certificate/Smart Card authentication is not supported as means of auto sign-on.
- Client certificate verifications and CSD are not supported
- Md5 signature in the certificates are not working because of security issue, which is a known problem on iOS: http://support.citrix.com/article/CTX132798

- SHA2 signature is not supported except for Windows, as described on the Citrix website: http://www.citrix.com/
- A key size >1024 is not supported

Other Limitations

- HTTP redirect is not supported; the Citrix Receiver application does not work with redirects.
- XML service must be installed and configured on the XenApp and XenDesktop servers.

About Citrix Mobile Receiver User Logon

The logon for mobile users connecting to the Citrix server depends on whether the ASA has configured the Citrix server as a VDI server or a VDI proxy server.

When the Citrix server is configured as a VDI server:

1. Using the AnyConnect Secure Mobility Client, connect to ASA with VPN credentials.
2. Using Citrix Mobile Receiver, connect to Citrix server with Citrix server credentials (if single-signon is configured, the Citrix credentials are not required).

When the ASA is configured as a to a VDI proxy server:

1. Using Citrix Mobile Receiver, connect to the ASA entering credentials for both the VPN and Citrix server. After the first connection, if properly configured, subsequent connections only require VPN credentials.

Configuring the ASA to Proxy a Citrix Server

You can configure the ASA to act as a proxy for the Citrix servers, so that connections to the ASA appear to the user like connections to the Citrix servers. The AnyConnect client is not required when you enable VDI proxy in ASDM. The following high-level steps show how the end user connects to Citrix.

1. A mobile user opens Citrix Receiver and connects to ASA's URL.
2. The user provides credentials for the XenApp server and the VPN credentials on the Citrix logon screen.
3. For each subsequent connection to the Citrix server, the user only needs to enter the VPN credentials.

Using the ASA as a proxy for XenApp and XenDesktop removes the requirement for a Citrix Access Gateway. XenApp server info is logged on the ASA, and displays in ASDM.

Configure the Citrix server's address and logon credentials, and assign that VDI server to a Group Policy or username. If both username and group-policy are configured, username settings override group-policy settings.

Additional Information

http://www.youtube.com/watch?v=JMM2RzppaG8 - This video describes the advantages of using that ASA as a Citrix proxy.

Assigning a VDI Server to a Group Policy

VDI servers are configured and assigned to Group Policies by:

- Adding the VDI server on the VDI Access pane, and assigning a group policy to the server.

- Adding a VDI server to the group policy.

If both username and group policy are configured, username settings take precedence over group policy. Enter the following:

```
configure terminal
    group-policy DfltGrpPolicy attributes
        webvpn
            vdi type <citrix> url <url> domain <domain> username <username> password
            <password>
configure terminal
    username <username> attributes
        webvpn
            vdi type <citrix> url <url> domain <domain> username <username> password
            <password>]
```

The syntax options are defined as follows:

- type—Type of VDI. For a Citrix Receiver type, this value must be *citrix*.
- url—Full URL of the XenApp or XenDesktop server including http or https, hostname, and port number, as well as the path to the XML service.
- username—Username for logging into the virtualization infrastructure server. This value can be a clientless macro.
- password—Password for logging into the virtualization infrastructure server. This value can be a clientless macro.
- domain—Domain for logging into the virtualization infrastructure server. This value can be a clientless macro.

Using SSL to Access Internal Servers

	Command	Purpose
Step 1	`webvpn`	Switches to group policy Clientless SSL VPN configuration mode.
Step 2	`url-entry disable`	Switches off URL Entry.

Clientless SSL VPN uses SSL and its successor, TLS1 to provide a secure connection between remote users and specific, supported internal resources at an internal server. This section includes the following topics:

- Using HTTPS for Clientless SSL VPN Sessions, page 15-22
- Configuring Clientless SSL VPN and ASDM Ports, page 15-22
- Configuring Support for Proxy Servers, page 15-23
- Configuring SSL/TLS Encryption Protocols, page 15-25

Using HTTPS for Clientless SSL VPN Sessions

Prerequisites

In a Web browser, users enter the ASA address in the format https://*address* where *address* is the IP address or DNS hostname of the ASA interface.

Restrictions

- You must enable Clientless SSL VPN sessions on the ASA interface that users connect to.
- You must use HTTPS to access the ASA or load-balancing cluster.

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**enable** <name of interface to use for Clientless SSL VPN sessions> **Example:** hostname(config)# **webvpn** hostname(config-webvpn)# **enable outside**	Enables Clientless SSL VPN sessions on the interface called outside.

Configuring Clientless SSL VPN and ASDM Ports

From version 8.0(2), the ASA supports both Clientless SSL VPN sessions and ASDM administrative sessions simultaneously on port 443 of the outside interface. You can configure these applications on different interfaces.

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.

	Command	Purpose
Step 2	**port** *port_number* **Example:** ciscoasa(config)# **http server enable** ciscoasa(config)# **http 192.168.3.0 255.255.255.0 outside** ciscoasa(config)# **webvpn** ciscoasa(config-webvpn)# **port 444** ciscoasa(config-webvpn)# **enable outside**	Changes the SSL listening port for Clientless SSL VPN. Enables Clientless SSL VPN on port 444 of the outside interface. With this configuration, remote users initiating Clientless SSL VPN sessions enter https://<outside_ip>:444 in the browser.
Step 3	**http server enable** **Example:** ciscoasa(config)# **http server enable** ciscoasa(config)# **http 192.168.3.0 255.255.255.0 outside** ciscoasa(config)# **webvpn** ciscoasa(config-webvpn)# **enable outside**	(Privileged mode) Changes the listening port for ASDM. Specifies that HTTPS ASDM sessions use port 444 on the outside interface. Clientless SSL VPN is also enabled on the outside interface and uses the default port (443). With this configuration, remote users initiate ASDM sessions by entering https://<outside_ip>:444

Configuring Support for Proxy Servers

The ASA can terminate HTTPS connections and forward HTTP and HTTPS requests to proxy servers. These servers act as intermediaries between users and the public or private network. Requiring network access via a proxy server that the organization controls provides another opportunity for filtering, to assure secure network access and administrative control.

When configuring support for HTTP and HTTPS proxy services, you can assign preset credentials to send with each request for basic authentication. You can also specify URLs to exclude from HTTP and HTTPS requests.

Restrictions

You can specify a proxy autoconfiguration (PAC) file to download from an HTTP proxy server, however, you may not use proxy authentication when specifying the PAC file.

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**http-proxy and https-proxy**	Configures the ASA to use an external proxy server to handle HTTP and HTTPS requests. **Note** Proxy NTLM authentication is not supported in **http-proxy**. Only proxy without authentication and basic authentication are supported.
Step 3	**http-proxy** *host* [*port*] [**exclude** *url*] [**username** *username* {**password** *password*}]	Configure HTTP proxy.

	Command	Purpose
Step 4	**https-proxy** *host* [*port*] [**exclude** *url*] [**username** *username* {**password** *password*}]	Configure HTTPS proxy.
Step 5	**http-proxy pac** *url*	Set the PAC file URL.
Step 6	(Optional) exclude	Excludes URLs from those that can be sent to the proxy server.
Step 7	*host*	Provides the hostname or IP address for the external proxy server.
Step 8	pac	Proxy autoconfiguration file downloaded to the ASA that uses a JavaScript function to identify a proxy for each URL.
Step 9	(Optional, and only available if you specify a username) password	Accompanies each proxy request with a password to provide basic, proxy authentication.
Step 10	*password*	Password to send to the proxy server with each HTTP or HTTPS request.
Step 11	(Optional) *port*	Provides the port number used by the proxy server. The default HTTP port is 80. The default HTTPS port is 443. The ASA uses each of these ports if you do not specify an alternative value. The range is 1-65535.
Step 12	*url*	If you entered **exclude**, enter a URL or a comma-delimited list of several URLs to exclude from those that can be sent to the proxy server. The string does not have a character limit, but the entire command cannot exceed 512 characters. You can specify literal URLs or use the following wildcards: – ***** to match any string, including slashes (/) and periods (.). You must accompany this wildcard with an alphanumeric string. – **?** to match any single character, including slashes and periods. – [*x*-*y*] to match any single character in the range of *x* and *y*, where *x* represents one character and *y* represents another character in the ANSI character set. – [**!***x*-*y*] to match any single character that is not in the range.
Step 13	If you entered **http-proxy pac**, follow it with **http://** and type the URL of the proxy autoconfiguration file. (If you omit the **http://** portion, the CLI ignores the command.)	—
Step 14	(Optional) username	Accompanies each HTTP proxy request with a username for basic, proxy authentication. Only the **http-proxy** *host* command supports this keyword.
Step 15	*username*	Username to send to the proxy server with each HTTP or HTTPS request.

	Command	Purpose

Step 16	**Example:**
	`ciscoasa(config-webvpn)# http-proxy 209.165.201.1 user jsmith password mysecretdonttell`
	
	`ciscoasa(config-webvpn)`
		Shows how to configure use of an HTTP proxy server with an IP address of 209.165. 201.1 using the default port, sending a username and password with each HTTP request.

Step 17	**Example:**
	`ciscoasa(config-webvpn)# http-proxy 209.165.201.1 exclude www.example.com username jsmith password mysecretdonttell`
	`ciscoasa(config-webvpn)`
		Shows the same command, except when the ASA receives the specific URL www.example.com in an HTTP request, it resolves the request instead of passing it on to the proxy server.

Step 18	**Example:**
	`ciscoasa(config-webvpn)# http-proxy pac http://www.example.com/pac`
	`ciscoasa(config-webvpn)`
		Shows how to specify a URL to serve a proxy autoconfiguration file to the browser.

The ASA Clientless SSL VPN configuration supports only one **http-proxy** and one **https-proxy** command each. For example, if one instance of the **http-proxy** command is already present in the running configuration and you enter another, the CLI overwrites the previous instance.

Note Proxy NTLM authentication is not supported in **http-proxy**. Only proxy without authentication and basic authentication is supported.

Configuring SSL/TLS Encryption Protocols

Port forwarding requires the Oracle Java Runtime Environment (JRE). Port forwarding does not work when a user of Clientless SSL VPN connects with some SSL versions. Refer to the compatibility matrix for supported JRE versions.

Authenticating with Digital Certificates

SSL uses digital certificates for authentication. The ASA creates a self-signed SSL server certificate when it boots; or you can install in the ASA an SSL certificate that has been issued in a PKI context. For HTTPS, this certificate must then be installed on the client.

Restrictions

Email clients such as MS Outlook, MS Outlook Express, and Eudora lack the ability to access the certificate store.

For more information on authentication and authorization using digital certificates, see the section on using certifiactes and user login credetials in the general operations configuration guide.

Configuring Browser Access to Client-Server Plug-ins

The Client-Server Plug-in table displays the plug-ins the ASA makes available to browsers in Clientless SSL VPN sessions.

To add, change, or remove a plug-in, do one of the following:

- To add a plug-in, click **Import**. The Import Plug-ins dialog box opens.
- To remove a plug-in, choose it and click **Delete**.

The following sections describe the integration of browser plug-ins for Clientless SSL VPN browser access:

- About Installing Browser Plug-ins
- Preparing the Security Appliance for a Plug-in
- Installing Plug-ins Redistributed by Cisco

About Installing Browser Plug-ins

A browser plug-in is a separate program that a Web browser invokes to perform a dedicated function, such as connect a client to a server within the browser window. The ASA lets you import plug-ins for download to remote browsers in Clientless SSL VPN sessions. Of course, Cisco tests the plug-ins it redistributes, and in some cases, tests the connectivity of plug-ins we cannot redistribute. However, we do not recommend importing plug-ins that support streaming media at this time.

The ASA does the following when you install a plug-in onto the flash device:

- (Cisco-distributed plug-ins only) Unpacks the jar file specified in the *URL*.
- Writes the file to the csco-config/97/plugin directory on the ASA file system.
- Populates the drop-down menu next to the URL attributes in ASDM.
- Enables the plug-in for all future Clientless SSL VPN sessions, and adds a main menu option and an option to the drop-down menu next to the Address field of the portal page.

  Table 15-3 shows the changes to the main menu and address field of the portal page when you add the plug-ins described in the following sections.

Table 15-3 Effects of Plug-ins on the Clientless SSL VPN Portal Page

Plug-in	Main Menu Option Added to Portal Page	Address Field Option Added to Portal Page
ica	Citrix Client	citrix://
rdp	Terminal Servers	rdp://
rdp2	Terminal Servers Vista	rdp2://
ssh,telnet	SSH	ssh://
	Telnet	telnet://
vnc	VNC Client	vnc://

Note A secondary ASA obtains the plug-ins from the primary ASA.

When the user in a Clientless SSL VPN session clicks the associated menu option on the portal page, the portal page displays a window to the interface and displays a help pane. The user can select the protocol displayed in the drop-down menu and enter the URL in the Address field to establish a connection.

Note Some Java plug-ins may report a status of connected or online even when a session to the destination service is not set up. The open-source plug-in reports the status, not the ASA.

Before installing the first plug-in, you must follow the instructions in the next section.

Prerequisites

- The plug-ins do not work if the security appliance configures the clientless session to use a proxy server.

Note The remote desktop protocol plug-in does not support load balancing with a session broker. Because of the way the protocol handles the redirect from the session broker, the connection fails. If a session broker is not used, the plug-in works.

- The plug-ins support single sign-on (SSO). They use the *same* credentials entered to open the Clientless SSL VPN session. Because the plug-ins do not support macro substitution, you do not have the options to perform SSO on different fields such as the internal domain password or on an attribute on a RADIUS or LDAP server.
- To configure SSO support for a plug-in, you install the plug-in, add a bookmark entry to display a link to the server, and specify SSO support when adding the bookmark.
- The minimum access rights required for remote use belong to the guest privilege mode.

Requirements

- Per the GNU General Public License (GPL), Cisco redistributes plug-ins without having made any changes to them. Per the GPL, Cisco cannot directly enhance these plug-ins.
- Clientless SSL VPN must be enabled on the ASA to provide remote access to the plug-ins.
- A stateful failover does not retain sessions established using plug-ins. Users must reconnect following a failover.
- Plug-ins require that ActiveX or Oracle Java Runtime Environment (JRE) 1.4.2 (or later) is enabled on the browser. There is no ActiveX version of the RDP plug-in for 64-bit browsers.

RDP Plug-in ActiveX Debug Quick Reference

To set up and use an RDP plug-in, you must add a new environment variable.

Step 1 Right-click **My Computer** to access the System Properties, and choose the **Advanced** tab.

Step 2 On the Advanced tab, choose the environment variables button.

Step 3 In the new user variable dialog box, enter the RF_DEBUG variable.

Step 4 Verify the new Environment Variable in the user variables section.

Step 5 If you used the client computer with versions of Clientless SSL VPN before version 8.3, you must remove the old Cisco Portforwarder Control. Go to the C:/WINDOWS/Downloaded Program Files directory, right-click portforwarder control, and choose **Remove**.

Step 6 Clear all of the Internet Explorer browser cache.

Step 7 Launch your Clientless SSL VPN session and establish an RDP session with the RDP ActiveX Plug-in.

You can now observe events in the Windows Application Event viewer.

Preparing the Security Appliance for a Plug-in

Step 1 Ensure that Clientless SSL VPN is enabled on an ASA interface.

Step 2 Install an SSL certificate onto the ASA interface to which remote users use a fully-qualified domain name (FQDN) to connect.

Note Do not specify an IP address as the common name (CN) for the SSL certificate. The remote user attempts to use the FQDN to communicate with the ASA. The remote PC must be able to use DNS or an entry in the System32\drivers\etc\hosts file to resolve the FQDN.

Configuring the ASA to Use the New HTML File

DETAILED STEPS

	Command	Purpose
Step 1	**import webvpn webcontent <file> <url>** **Example:** hostname# **import webvpn webcontent /+CSCOU+/login.inc tftp://209.165.200.225/login.inc** !!!!* Web resource `+CSCOU+/login.inc' was successfully initialized hostname#	Imports the file and images as Web Content.
Step 2	**export webvpn customization <file> <URL>** **Example:** hostname2# **export webvpn customization template tftp://209.165.200.225/sales_vpn_login** !! %INFO: Customization object 'Template' was exported to tftp://10.21.50.120/sales_vpn_login	Exports a customization template.
Step 3	Change the full customization mode tag in the file to enable. **Example:** <full-customization> <mode>**enable**</mode> <url>/**+CSCOU+/login.inc**</url> </full-customization>	Supplies the URL of the login file stored in the ASA memory.

	Command	Purpose

Step 4	Import the file as a new customization object
	
	Example:
	`ciscoasa# import webvpn customization sales_vpn_login tftp://10.21.50.120/sales_vpn_login$`
	`!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!`
	`%INFO: customization object 'sales_vpn_login' was successfully imported`
		—

Step 5	Apply the customization object to a Connection Profile (tunnel group)
	
	Example:
	`hostname(config)# tunnel-group Sales webvpn-attributes`
	`hostname(config-tunnel-webvpn)#customization sales_vpn_login`
		—

Advanced Clientless SSL VPN Configuration

June 25, 2014

Microsoft Kerberos Constrained Delegation Solution

Many organizations want to authenticate their Clientless VPN users and extend their authentication credentials seamlessly to web-based resources using authentication methods beyond what the ASA SSO feature can offer today. With the growing demand to authenticate remote access users with smart cards and One-time Passwords (OTPs), the SSO feature falls short in meeting that demand, because it forwards only conventional user credentials, such as static username and password, to clientless web-based resources when authentication is required.

For example, neither certificate- nor OTP-based authentication methods encompass a conventional username and password necessary for the ASA to seamlessly perform SSO access to web-based resources. When authenticating with a certificate, a username and password are not required for the ASA to extend to web-based resources, making it an unsupported authentication method for SSO. On the other hand, OTP does include a static username; however, the password is dynamic and will subsequently change throughout the VPN session. In general, Web-based resources are configured to accept static usernames and passwords, thus also making OTP an unsupported authentication method for SSO.

Microsoft's Kerberos Constrained Delegation (KCD), a new feature introduced in software release 8.4 of the ASA, provides access to Kerberos-protected Web applications in the private network. With this benefit, you can seamlessly extend certificate- and OTP-based authentication methods to Web applications. Thus, with SSO and KCD working together although independently, many organizations can now authenticate their clientless VPN users and extend their authentication credentials seamlessly to Web applications using all authentication methods supported by the ASA.

Requirements

In order for the **kcd-server** command to function, the ASA must establish a trust relationship between the *source* domain (the domain where the ASA resides) and the *target* or *resource* domain (the domain where the Web services reside). The ASA, using its unique format, crosses the certification path from the source to the destination domain and acquires the necessary tickets on behalf of the remote access user to access the services.

This crossing of the certificate path is called cross-realm authentication. During each phase of cross-realm authentication, the ASA relies on the credentials at a particular domain and the trust relationship with the subsequent domain.

Understanding How KCD Works

Kerberos relies on a trusted third party to validate the digital identity of entities in a network. These entities (such as users, host machines, and services running on hosts) are called principals and must be present in the same domain. Instead of secret keys, Kerberos uses tickets to authenticate a client to a server. The ticket is derived from the secret key and consists of the client's identity, an encrypted session key, and flags. Each ticket is issued by the key distribution center and has a set lifetime.

The Kerberos security system is a network authentication protocol used to authenticate entities (users, computers, or applications) and protect network transmissions by scrambling the data so that only the device that the information was intended for can decrypt it. You can configure KCD to provide Clientless SSL VPN users with SSO access to Microsoft Web services protected by Kerberos. Supported Web services or applications include Outlook Web Access (OWA), Sharepoint, and Internet Information Server (IIS).

Note Web services from providers other than Microsoft are not currently supported.

Two extensions to the Kerberos protocol were implemented: *protocol transition* and *constrained delegation*. These extensions allow the Clientless SSL VPN remote access users to access Kerberos-authenticated applications in the private network.

Protocol transition provides you with increased flexibility and security by supporting different authentication mechanisms at the user authentication level and by switching to the Kerberos protocol for security features (such as mutual authentication and constrained delegation) in subsequent application layers. *Constrained delegation* provides a way for domain administrators to specify and enforce application trust boundaries by limiting where application services can act on a user's behalf. This flexibility improves application security designs by reducing the chance of compromise by an untrusted service.

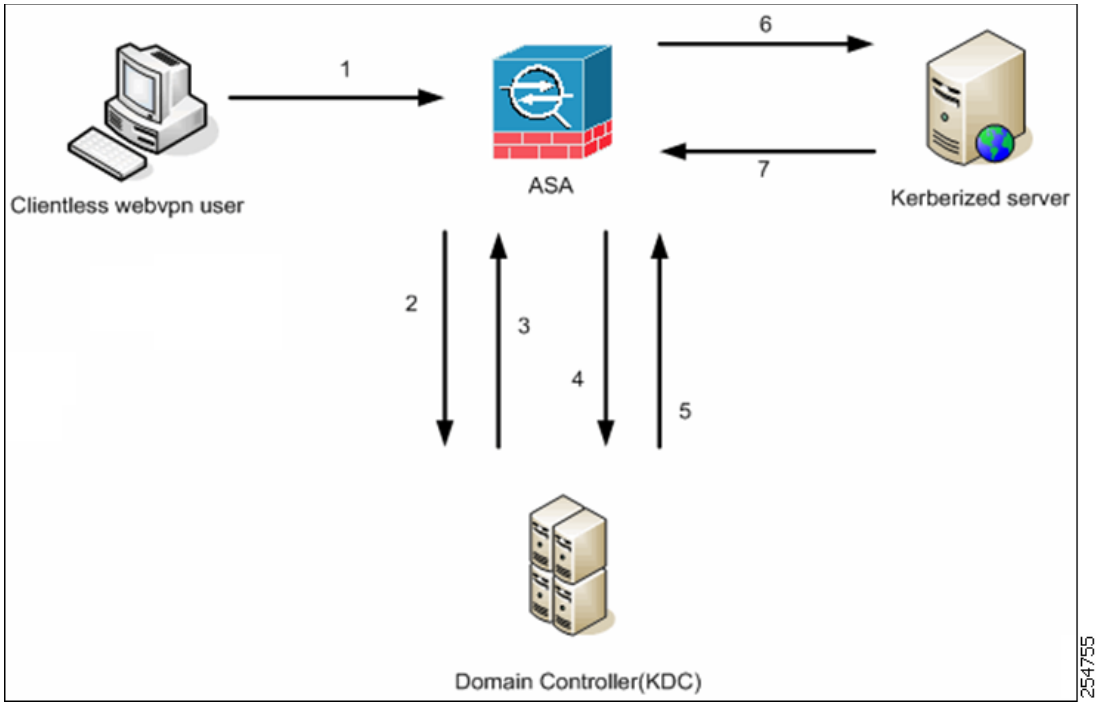
For more information on constrained delegation, see RFC 1510 via the IETF website (http://www.ietf.org).

Authentication Flow with KCD

Figure 16-1 depicts the packet and process flow a user will experience directly and indirectly when accessing resources trusted for delegation via the clientless portal. This process assumes that the following tasks have been completed:

- Configured KCD on ASA
- Joined the Windows Active Directory and ensured services are trusted for delegation
- Delegated ASA as a member of the Windows Active Directory domain

Figure 16-1 KCD Process



Note A clientless user session is authenticated by the ASA using the authentication mechanism configured for the user. (In the case of smartcard credentials, ASA performs LDAP authorization with the userPrincipalName from the digital certificate against the Windows Active Directory).

1. After successful authentication, the user logs in to the ASA clientless portal page. The user accesses a Web service by entering a URL in the portal page or by clicking on the bookmark. If the Web service requires authentication, the server challenges ASA for credentials and sends a list of authentication methods supported by the server.

   Note KCD for Clientless SSL VPN is supported for all authentication methods (RADIUS, RSA/SDI, LDAP, digital certificates, and so on). Refer to the AAA Support table at http://www.cisco.com/en/US/docs/security/asa/asa84/configuration/guide/access_aaa.html#wp1069492.

2. Based on the HTTP headers in the challenge, ASA determines whether the server requires Kerberos authentication. (This is part of the SPNEGO mechanism.) If connecting to a backend server requires Kerberos authentication, the ASA requests a service ticket for itself on behalf of the user from the key distribution center.
3. The key distribution center returns the requested tickets to the ASA. Even though these tickets are passed to the ASA, they contain the user's authorization data.ASA requests a service ticket from the KDC for the specific service that the user wants to access.

Note Steps 1 to 3 comprise protocol transition. After these steps, any user who authenticates to ASA using a non-Kerberos authentication protocol is transparently authenticated to the key distribution center using Kerberos.

4. ASA requests a service ticket from the key distribution center for the specific service that the user wants to access.
5. The key distribution center returns a service ticket for the specific service to the ASA.
6. ASA uses the service ticket to request access to the Web service.
7. The Web server authenticates the Kerberos service ticket and grants access to the service. The appropriate error message is displayed and requires acknowledgement if there is an authentication failure. If the Kerberos authentication fails, the expected behavior is to fall back to basic authentication.

Before Configuring KCD

To configure the ASA for cross-realm authentication, you must use the following commands.

	Command	Purpose
Step 1	**ntp** hostname **Example:** ciscoasa(config)# **configure terminal** #Create an alias for the Domain Controller ciscoasa(config)# **name 10.1.1.10 DC** #Configure the Name server	Joins the Active Directory domain. A 10.1.1.10 domain controller (which is reachable inside the interface).
Step 2	**dns domain-lookup** **dns server-group** **Example:** ciscoasa(config)# **ntp server DC** #Enable a DNS lookup by configuring the DNS server and Domain name ciscoasa(config)# **dns domain-lookup inside** ciscoasa(config)# **dns server-group DefaultDNS** ciscoasa(config-dns-server-group)# **name-server DC** ciscoasa(config-dns-server-group)# **domain-name private.net** #Configure the AAA server group with Server and Realm ciscoasa(config)# **aaa-server KerberosGroup protocol Kerberos** ciscoasa(config-asa-server-group)# **aaa-server KerberosGroup (inside) host DC** ciscoasa(config-asa-server-group)# **Kerberos-realm PRIVATE.NET** #Configure the Domain Join ciscoasa(config)# **webvpn** ciscoasa(config-webvpn)# **kcd-server KerberosGroup username dcuser password dcuser123**! ciscoasa(config)#	Performs a lookup. A domain name of private.net and a service account on the domain controller using dcuser as the username and dcuser123! as the password.

Configuring KCD

To have the ASA join a Windows Active Directory domain and return a success or failure status, perform these steps.

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**kcd-server**	Configure the KCD.

Step 3	**kcd-server** ***aaa-server-group***
	
	Example:
	`ASA(config)# aaa-server KG protocol kerberos`
	`ASA(config)# aaa-server KG (inside) host DC`
	`ASA(config-aaa-server-host)# kerberos-realm test.edu`
	`ASA(webvpn-config)# kcd-server KG username user1 password abc123`
	`ASA(webvpn-config)# no kcd-server`
		Specifies the domain controller name and realm. The AAA server group must be a Kerberos type.

Step 4	(Optional) **no kcd-server**	Removes the specified behavior for the ASA.
Step 5	(Optional) **kcd-server reset**	Resets to the internal state.
Step 6	**kcd domain-join username** *<user>* **password** *<pass>* user—Does not correspond to a specific administrative user but simply a user with service-level privileges to add a device on the Windows domain controller. pass—The password does not correspond to a specific password but simply a user with service-level password privileges to add a device on the Windows domain controller.	Checks for the presence of a KCD server and starts the domain join process. The Active Directory username and password are used only in EXEC mode and are not saved in the configuration. **Note** Administrative privileges are required for initial join. A user with service-level privileges on the domain controller will not get access.
Step 7	`kcd domain-leave`	Verifies whether the KCD server command has a valid domain join status and then initiates a domain leave.

Showing KCD Status Information

To display the domain controller information and the domain join status, perform this step.

	Command	Purpose

Step 8	**show webvpn kcd**
	
	Example:
	`ASA# show webvpn kcd`
	`KCD-Server Name: DC`
	`User : user1`
	`Password : ****`
	`KCD State : Joined`
		Displays the domain controller information and the domain join status.

Showing Cached Kerberos Tickets

To display all Kerberos tickets cached on the ASA, enter the following commands:

	Command	Purpose
Step 9	**show aaa kerberos**	Displays all Kerberos tickets cached on the ASA.
Step 10	**show aaa kerberos [username *user* | host *ip* | *hostname*]** **Example:** ASA# **show aaa kerberos** Default Principal Valid Starting Expires Service Principal asa@example.COM 06/29/10 18:33:00 06/30/10 18:33:00 krbtgt/example.COM@example.COM kcduser@example.COM06/29/10 17:33:00 06/30/10 17:33:00 asa$/example.COM@example.COM kcduser@example.COM06/29/10 17:33:00 06/30/10 17:33:00 http/owa.example.com@example.COM ASA# show aaa kerberos username kcduser Default Principal Valid Starting Expires Service Principal kcduser@example.COM06/29/10 17:33:00 06/30/10 17:33:00 asa$/example.COM@example.COM kcduser@example.COM06/29/10 17:33:00 06/30/10 17:33:00 http/owa.example.com@example.COM ASA# show aaa kerberos host owa.example.com Default Principal Valid Starting Expires Service Principal kcduser@example.COM06/29/1006/30/10 17:33:00 http/owa.example.com@example.COM ASA# show aaa kerberos username kcduser Default Principal Valid Starting Expires Service Principal kcduser@example.COM06/29/10 17:33:00 06/30/10 17:33:00 asa$/example.COM@example.COM kcduser@example.COM06/29/10 17:33:00 06/30/10 17:33:00 http/owa.example.com@example.COM ASA# show aaa kerberos host owa.example.com Default Principal Valid Starting Expires Service Principal kcduser@example.COM06/29/10 06/30/10 17:33:00 http/owa.example.com@example.COM	• user—Used to view the Kerberos tickets of a specific user • hostname—Used to view the Kerberos tickets issued for a specific host

Clearing Cached Kerberos Tickets

To clear all Kerberos ticket information on the ASA, perform these steps.

	Command	Purpose
Step 11	**clear aaa kerberos**	Clears all Kerberos ticket information on the ASA.
Step 12	**clear aaa kerberos [username** *user* **| host** *ip* **|** *hostname***]**	• *user*—Used to clear the Kerberos tickets of a specific user • *hostname*—Used to clear the Kerberos tickets of a specific host

Note

Restrictions

When creating a bookmark to an application that uses Kerberos constrained delegation (KCD), do not check Enable Smart Tunnel.

DETAILED STEPS

Configuring Application Profile Customization Framework

Clientless SSL VPN includes an Application Profile Customization Framework (APCF) option that lets the ASA handle non-standard applications and Web resources so they display correctly over a Clientless SSL VPN connection. An APCF profile contains a script that specifies when (pre, post), where (header, body, request, response), and what (data) to transform for a particular application. The script is in XML and uses sed (stream editor) syntax to transform strings/text.

You can configure and run multiple APCF profiles in parallel on an ASA. Within an APCF profile script, multiple APCF rules can apply. The ASA processes the oldest rule first, based on configuration history, the next oldest rule next.

You can store APCF profiles on the ASA flash memory, or on an HTTP, HTTPS, or TFTP server.

Restrictions

We recommend that you configure an APCF profile only with the assistance of Cisco personnel.

Managing APCF Packets

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**apcf** **Example:** ciscoasa(config)# **webvpn** ciscoasa(config-webvpn)# **apcf flash:/apcf/apcf1.xml** ciscoasa(config)# **webvpn** ciscoasa(config-webvpn)# **apcf https://myserver:1440/apcf/apcf2.xml**	Identifies and locates an APCF profile to load on the ASA. Shows how to enable an APCF profile named apcf1.xml, located in flash memory. Shows how to enable an APCF profile named apcf2.xml, located on an HTTPS server called myserver, port 1440, with the path being /apcf.

APCF Syntax

APCF profiles use XML format, and sed script syntax, with the XML tags in Table 16-1.

Guidelines

Misuse of an APCF profile can result in reduced performance and undesired rendering of content. In most cases, Cisco Engineering supplies APCF profiles to solve specific application rendering issues.

Table 16-1 APCF XML Tags

Tag	Use
<APCF>...</APCF>	The mandatory root element that opens any APCF XML file.
<version>1.0</version>	The mandatory tag that specifies the APCF implementation version. Currently the only version is 1.0.
<application>...</application>	The mandatory tag that wraps the body of the XML description.
<id> text </id>	The mandatory tag that describes this particular APCF functionality.
<apcf-entities>...</apcf-entities>	The mandatory tag that wraps a single or multiple APCF entities.

Table 16-1 APCF XML Tags (continued)

Tag	Use
<js-object>…</js-object> <html-object>…</html-object> <process-request-header>...</process-request-header> <process-response-header>...</process-response-header> <preprocess-response-body>...</preprocess-response-body> <postprocess-response-body>...</postprocess-response-body>	One of these tags specifies type of content or the stage at which the APCF processing should take place.
<conditions>… </conditions>	A child element of the pre/post-process tags that specifies criteria for processing such as: • http-version (such as 1.1, 1.0, 0.9) • http-method (get, put, post, webdav) • http-scheme (“http/”, “https/”, other) • server-regexp regular expression containing ("a".."z" | "A".."Z" | "0".."9" | ".-_*[]?") • server-fnmatch (regular expression containing ("a.."z" | "A".."Z" | "0".."9" | ".-_*[]?+()\{},"), • user-agent-regexp • user-agent-fnmatch • request-uri-regexp • request-uri-fnmatch • If more than one of condition tags is present, the ASA performs a logical AND for all tags.
<action> … </action>	Wraps one or more actions to perform on the content under specified conditions; you can use the following tags to define these actions (shown below): • <do> • <sed-script> • <rewrite-header> • <add-header> • <delete-header>

Table 16-1 APCF XML Tags (continued)

Tag	Use
<do>…</do>	Child element of the action tag used to define one of the following actions: • <no-rewrite/>—Do not mangle the content received from the remote server. • <no-toolbar/>—Do not insert the toolbar. • <no-gzip/>—Do not compress the content. • <force-cache/>—Preserve the original caching instructions. • <force-no-cache/>—Make object non-cacheable. • < downgrade-http-version-on-backend>—Use HTTP/1.0 when sending the request to remote server.
<sed-script> TEXT </sed-script>	Child element of the action tag used to change the content of text-based objects. The Text must be a valid Sed script. The <sed-script> applies to the <conditions> tag defined before it.
<rewrite-header></rewrite-header>	Child element of the action tag. Changes the value of the HTTP header specified in the child element <header> tag shown below.
<add-header></add-header>	Child element of the action tag used to add a new HTTP header specified in the child element <header> tag shown below.
<delete-header></delete-header>	Child element of the action tag used to delete the specified HTTP header specified by the child element <header> tag shown below.
<header></header>	Specifies the name HTTP header to be rewritten, added, or deleted. For example, the following tag changes the value of the HTTP header named Connection: <rewrite-header> <header>Connection</header> <value>close</value> </rewrite-header>

Configuration Examples for APCF

Example:

```
<APCF>
<version>1.0</version>
<application>
  <id>Do not compress content from example.com</id>
  <apcf-entities>
      <process-request-header>
         <conditions>
           <server-fnmatch>*.example.com</server-fnmatch>
         </conditions>
           <action>
             <do><no-gzip/></do>
```

```
          </action>
      </process-request-header>
  </apcf-entities>
</application>
</APCF>
```

Example:

```
<APCF>
<version>1.0</version>
<application>
 <id>Change MIME type for all .xyz objects</id>
 <apcf-entities>
      <process-response-header>
        <conditions>
            <request-uri-fnmatch>*.xyz</request-uri-fnmatch>
        </conditions>
         <action>
           <rewrite-header>
                <header>Content-Type</header>
                <value>text/html</value>
           </rewrite-header>
         </action>
      </process-response-header>
 </apcf-entities>
</application>
</APCF>
```

Encoding

With encoding, you can view or specify the character encoding for Clientless SSL VPN portal pages.

Character encoding, also called "character coding" and "a character set," is the pairing of raw data (such as 0s and 1s) with characters to represent the data. The language determines the character encoding method to use. Some languages use a single method, while others do not. Usually, the geographic region determines the default encoding method used by the browser, but the remote user can change it. The browser can also detect the encoding specified on the page, and render the document accordingly.

The encoding attribute lets you specify the value of the character-encoding method used on the portal page to ensure that the browser renders it properly, regardless of the region in which the user is using the browser, and regardless of any changes made to the browser.

By default, the ASA applies the "Global Encoding Type" to pages from Common Internet File System servers. The mapping of CIFS servers to their appropriate character encoding, globally with the "Global Encoding Type" attribute, and individually with the file-encoding exceptions displayed in the table, provides for the accurate handling and display of CIFS pages when the proper rendering of filenames or directory paths, as well as pages, is an issue.

DETAILED STEPS

Step 1 Global Encoding Type determines the character encoding that all Clientless SSL VPN portal pages inherit except for those from the CIFS servers listed in the table. You can type the string or choose one of the options from the drop-down list, which contains the most common values, as follows:

- big5
- gb2312

- ibm-850
- iso-8859-1
- shift_jis

Note If you are using Japanese Shift_jis Character encoding, click **Do Not Specify** in the Font Family area of the associated Select Page Font pane to remove the font family.

- unicode
- windows-1252
- none

Note If you click **none** or specify a value that the browser on the Clientless SSL VPN session does not support, it uses its own default encoding.

You can type a string consisting of up to 40 characters, and equal to one of the valid character sets identified in http://www.iana.org/assignments/character-sets. You can use either the name or the alias of a character set listed on that page. The string is case-insensitive. The command interpreter converts upper-case to lower-case when you save the ASA configuration.

Step 2 Enter the name or IP address of a CIFS server for which the encoding requirement differs from the "Global Encoding Type" attribute setting. The ASA retains the case you specify, although it ignores the case when matching the name to a server.

Step 3 Choose the character encoding that the CIFS server should provide for Clientless SSL VPN portal pages. You can type the string, or choose one from the drop-down list, which contains only the most common values, as follows:

- big5
- gb2312
- ibm-850
- iso-8859-1
- shift_jis

Note If you are using Japanese Shift_jis Character encoding, click **Do Not Specify** in the Font Family area of the associated Select Page Font pane to remove the font family.

- unicode
- windows-1252
- none

If you click **none** or specify a value that the browser on the Clientless SSL VPN session does not support, it uses its own default encoding.

You can type a string consisting of up to 40 characters, and equal to one of the valid character sets identified in http://www.iana.org/assignments/character-sets. You can use either the name or the alias of a character set listed on that page. The string is case-insensitive. The command interpreter converts upper-case to lower-case when you save the ASA configuration.

Using Email over Clientless SSL VPN

Clientless SSL VPN supports several ways to access email. This section includes the following methods:

- Configuring Email Proxies
- Configuring Web email: MS Outlook Web App

Configuring Email Proxies

Clientless SSL VPN supports IMAP, POP3, and SMTP email proxies. The following attributes apply globally to email proxy users.

Restrictions

email clients such as MS Outlook, MS Outlook Express, and Eudora lack the ability to access the certificate store.

DETAILED STEPS

	Command	Purpose
Step 1	**accounting-server-group**	Specifies the previously configured accounting servers to use with email proxy.
Step 2	**authentication**	Specifies the authentication method(s) for email proxy users. The default values are as follows: • IMAP: Mailhost (required) • POP3 Mailhost (required) • SMTP: AAA
Step 3	**authentication-server-group**	Specifies the previously configured authentication servers to use with email proxy. The default is LOCAL.
Step 4	**authorization-server-group**	Specifies the previously configured authorization servers to use with Clientless SSL VPN.
Step 5	**authorization-required**	Requires users to authorize successfully to connect. The default is switched off.
Step 6	**authorization-dn-attributes**	Identifies the DN of the peer certificate to use as a username for authorization. The defaults are as follows: • Primary attribute: CN • Secondary attribute: OU
Step 7	**default-group-policy**	Specifies the name of the group policy to use. The default is DfltGrpPolicy.
Step 8	**enable**	Enables email proxy on the specified interface. The default is switched off.

	Command	Purpose
Step 9	**name-separator**	Defines the separator between the email and VPN usernames and passwords. The default is colon (:).
Step 10	**outstanding**	Configures the maximum number of outstanding non-authenticated sessions. The default is 20.
Step 11	**port**	Sets the port the email proxy listens to. The default is as follows: • IMAP:143 • POP3: 110 • SMTP: 25
Step 12	**server**	Specifies the default email server.
Step 13	**server-separator**	Defines the separator between the email and server names. The default is @.

Configuring Web email: MS Outlook Web App

The ASA supports Microsoft Outlook Web App to Exchange Server 2010 and Microsoft Outlook Web Access to Exchange Server 2007, 2003, and 2000.

DETAILED STEPS

Step 1 Enter the URL of the email service into the address field or click an associated bookmark in the Clientless SSL VPN session.

Step 2 When prompted, enter the email server username in the format *domain\username*.

Step 3 Enter the email password.

Configuring Policy Groups

September 13, 2013

Creating and Applying Clientless SSL VPN Policies for Accessing Resources

Creating and applying policies for Clientless SSL VPN that govern access to resources at an internal server includes the following task:

- Assigning Users to Group Policies

Assigning Users to Group Policies

Assigning users to group policies simplifies the configuration by letting you apply policies to many users. You can use an internal authentication server on the ASA or an external RADIUS or LDAP server to assign users to group policies. See Chapter 4, "Configuring Connection Profiles, Group Policies, and Users" for a thorough explanation of ways to simplify configuration with group policies.

Configuring Connection Profile Attributes for Clientless SSL VPN

Table 17-1 provides a list of connection profile attributes that are specific to Clientless SSL VPN. In addition to these attributes, you configure general connection profile attributes common to all VPN connections. For step-by-step information on configuring connection profiles, see Chapter 4, "Configuring Connection Profiles, Group Policies, and Users."

Note In earlier releases, "connection profiles" were known as "tunnel groups." You configure a connection profile with **tunnel-group** commands. This chapter often uses these terms interchangeably.

Table 17-1 Connection Profile Attributes for Clientless SSL VPN

Command	Function
`authentication`	Sets the authentication method.
`customization`	Identifies the name of a previously defined customization to apply.
`exit`	Exits from tunnel-group Clientless SSL VPN attribute configuration mode.
`nbns-server`	Identifies the name of the NetBIOS Name Service server (nbns-server) to use for CIFS name resolution.
`group-alias`	Specifies the alternate names by which the server can refer to a connection profile.
`group-url`	Identifies one or more group URLs. If you establish URLs with this attribute, this group is selected automatically for users when they access using these URLS.
`dns-group`	Identifies the DNS server group that specifies the DNS server name, domain name, name server, number of retries, and timeout values.
`help`	Provides help for tunnel group configuration commands.
`hic-fail-group-policy`	Specifies a VPN feature policy if you use the Cisco Secure Desktop Manager to set the Group-Based Policy attribute to "Use Failure Group-Policy" or "Use Success Group-Policy, if criteria match."
`no`	Removes an attribute value pair.
`override-svc-download`	Overrides downloading the group-policy or username attributes configured for downloading the AnyConnect VPN client to the remote user.
`pre-fill-username`	Configures username-to-certificate binding on this tunnel group.
`proxy-auth`	Identifies this tunnel-group as a specific proxy authentication tunnel group.
`radius-reject-message`	Enables the display of the RADIUS reject message on the login screen when authentication is rejected.
`secondary-pre-fill-username`	Configures the secondary username-to-certificate binding on this tunnel group.
`without-csd`	Switched off CSD for a tunnel group.

Configuring Group Policy and User Attributes for Clientless SSL VPN

Table 17-2 provides a list of group policy and user attributes for Clientless SSL VPN. For step-by-step instructions on configuring group policy and user attributes, see "Configuring Group Policies" and "Configuring Attributes for Individual Users" or in Chapter 4, "Configuring Connection Profiles, Group Policies, and Users."

Table 17-2 Group Policy and User Attributes for Clientless SSL VPN

Command	Function
activex-relay	Lets a user who has established a Clientless SSL VPN session use the browser to launch Microsoft Office applications. The applications use the session to download and upload ActiveX. The ActiveX relay remains in force until the Clientless SSL VPN session closes.
auto-sign-on	Sets values for auto sign-on, which requires that the user enter username and password credentials only once for a Clientless SSL VPN connection.
customization	Assigns a customization object to a group policy or user.
deny-message	Specifies the message delivered to a remote user who logs into Clientless SSL VPN successfully, but has no VPN privileges.
file-browsing	Enables CIFS file browsing for file servers and shares. Browsing requires NBNS (Master Browser or WINS).
file-entry	Allows users to enter file server names to access.
filter	Sets the name of the webtype access list.
hidden-shares	Controls the visibility of hidden shares for CIFS files.
homepage	Sets the URL of the Web page that displays upon login.
html-content-filter	Configures the content and objects to filter from the HTML for this group policy.
http-comp	Configures compression.
http-proxy	Configures the ASA to use an external proxy server to handle HTTP requests. **Note** Proxy NTLM authentication is not supported in **http-proxy**. Only proxy without authentication and basic authentication are supported.
keep-alive-ignore	Sets the maximum object size to ignore for updating the session timer.
port-forward	Applies a list of Clientless SSL VPN TCP ports to forward. The user interface displays the applications on this list.
post-max-size	Sets the maximum object size to post.
smart-tunnel	Configures a list of programs and several smart tunnel parameters to use smart tunnel.
sso-server	Sets the name of the SSO server.
storage-objects	Configures storage objects for the data stored between sessions.
svc	Configures SSL VPN Client attributes.
unix-auth-gid	Sets the UNIX group ID.
unix-auth-uid	Sets the UNIX user ID.
upload-max-size	Sets the maximum object size to upload.
url-entry	Controls the ability of the user to enter any HTTP/HTTPS URL.
url-list	Applies a list of servers and URLs that Clientless SSL VPN portal page displays for end-user access.
user-storage	Configures a location for storing user data between sessions.

Configuring Smart Tunnel Access

The following sections describe how to enable smart tunnel access with Clientless SSL VPN sessions, specify the applications to be provided with such access, and provide notes on using it.## Configuring Smart Tunnel Access

To configure smart tunnel access, you create a smart tunnel list containing one or more applications eligible for smart tunnel access, and the endpoint operating system associated with the list. Because each group policy or local user policy supports one smart tunnel list, you must group the nonbrowser-based applications to be supported into a smart tunnel list. After creating a list, you assign it to one or more group policies or local user policies.

The following sections describe smart tunnels and how to configure them:

About Smart Tunnels

A smart tunnel is a connection between a TCP-based application and a private site, using a clientless (browser-based) SSL VPN session with the security appliance as the pathway, and the ASA as a proxy server. You can identify applications for which to grant smart tunnel access, and specify the local path to each application. For applications running on Microsoft Windows, you can also require a match of the SHA-1 hash of the checksum as a condition for granting smart tunnel access.

Lotus SameTime and Microsoft Outlook are examples of applications to which you may want to grant smart tunnel access.

Configuring smart tunnels requires one of the following procedures, depending on whether the application is a client or is a web-enabled application:

- Create one or more smart tunnel lists of the client applications, then assign the list to the group policies or local user policies for whom smart tunnel access is required.
- Create one or more bookmark list entries that specify the URLs of the web-enabled applications eligible for smart tunnel access, then assign the list to the group policies or local user policies for whom smart tunnel access is required.

You can also list web-enabled applications for which to automate the submission of login credentials in smart tunnel connections over Clientless SSL VPN sessions.

Why Smart Tunnels?

Smart tunnel access lets a client TCP-based application use a browser-based VPN connection to access a service. It offers the following advantages to users, compared to plug-ins and the legacy technology, port forwarding:

- Smart tunnel offers better performance than plug-ins.
- Unlike port forwarding, smart tunnel simplifies the user experience by not requiring the user connection of the local application to the local port.
- Unlike port forwarding, smart tunnel does not require users to have administrator privileges.

The advantage of a plug-in is that it does not require the client application to be installed on the remote computer.

Prerequisites

See the *Supported VPN Platforms, Cisco ASA 5500 Series*, for the platforms and browsers supported by ASA Release 9.0 smart tunnels.

The following requirements and limitations apply to smart tunnel access on Windows:

- ActiveX or Oracle Java Runtime Environment (JRE) 4 update 15 or later (JRE 6 or later recommended) on Windows must be enabled on the browser.

  ActiveX pages require that you enter the **activex-relay** command on the associated group policy. If you do so or assign a smart tunnel list to the policy, and the browser proxy exception list on the endpoint specifies a proxy, the user must add a “shutdown.webvpn.relay.” entry to this list.
- Only Winsock 2, TCP-based applications are eligible for smart tunnel access.
- For Mac OS X only, Java Web Start must be enabled on the browser.

Restrictions

- Smart tunnel supports only proxies placed between computers running Microsoft Windows and the security appliance. Smart Tunnel uses the Internet Explorer configuration, which sets system-wide parameters in Windows. That configuration may include proxy information:
  - If a Windows computer requires a proxy to access the ASA, then there must be a static proxy entry in the client's browser, and the host to connect to must be in the client's list of proxy exceptions.
  - If a Windows computer does not require a proxy to access the ASA, but does require a proxy to access a host application, then the ASA must be in the client's list of proxy exceptions.

  Proxy systems can be defined the client’s configuration of static proxy entry or automatic configuration, or by a PAC file. Only static proxy configurations are currently supported by Smart Tunnels.
- Kerberos constrained delegation (KCD) is not supported for smart tunnels.
- With Windows, to add smart tunnel access to an application started from the command prompt, you must specify “cmd.exe” in the Process Name of one entry in the smart tunnel list, and specify the path to the application itself in another entry, because “cmd.exe” is the parent of the application.
- With HTTP-based remote access, some subnets may block user access to the VPN gateway. To fix this, place a proxy in front of the ASA to route traffic between the Web and the end user. That proxy must support the CONNECT method. For proxies that require authentication, Smart Tunnel supports only the basic digest authentication type.

- When smart tunnel starts, the ASA by default passes all browser traffic through the VPN session if the browser process is the same. The ASA only also does this if a tunnel-all policy (the default) applies. If the user starts another instance of the browser process, it passes all traffic through the VPN session. If the browser process is the same and the security appliance does not provide access to a URL, the user cannot open it. As a workaround, assign a tunnel policy that is not tunnel-all.
- A stateful failover does not retain smart tunnel connections. Users must reconnect following a failover.
- The Mac version of smart tunnel does not support POST bookmarks, form-based auto sign-on, or POST macro substitution.
- For Mac OS X users, only those applications started from the portal page can establish smart tunnel connections. This requirement includes smart tunnel support for Firefox. Using Firefox to start another instance of Firefox during the first use of a smart tunnel requires the user profile named csco_st. If this user profile is not present, the session prompts the user to create one.
- In Mac OS X, applications using TCP that are dynamically linked to the SSL library can work over a smart tunnel.
- Smart tunnel does not support the following on Mac OS X:
  - Proxy services.
  - Auto sign-on.
  - Applications that use two-level name spaces.
  - Console-based applications, such as Telnet, SSH, and cURL.
  - Applications using dlopen or dlsym to locate libsocket calls.
  - Statically linked applications to locate libsocket calls.
- Mac OS X requires the full path to the process and is case-sensitive. To avoid specifying a path for each username, insert a tilde (~) before the partial path (e.g., ~/bin/vnc).

Adding Applications to Be Eligible for Smart Tunnel Access

The Clientless SSL VPN configuration of each ASA supports *smart tunnel lists*, each of which identifies one or more applications eligible for smart tunnel access. Because each group policy or username supports only one smart tunnel list, you must group each set of applications to be supported into a smart tunnel list.

About Smart Tunnel Lists

For each group policy and username, you can configure Clientless SSL VPN to do one of the following:

- Start smart tunnel access automatically upon user login.
- Enable smart tunnel access upon user login, but require the user to start it manually, using the **Application Access > Start Smart Tunnels** button on the Clientless SSL VPN Portal Page.

Restrictions

The smart tunnel logon options are mutually exclusive for each group policy and username. Use only one.

DETAILED STEPS

The following smart tunnel commands are available to each group policy and username. The configuration of each group policy and username supports only one of these commands at a time, so when you enter one, the ASA replaces the one present in the configuration of the group policy or username in question with the new one, or in the case of the last command, simply removes the **smart-tunnel** command already present in the group policy or username.

	Command	Purpose
Step 1	**smart-tunnel auto-start** *list* OR **smart-tunnel enable** *list* OR **smart-tunnel disable** OR **no smart-tunnel [auto-start list | enable list | disable]**	Starts smart tunnel access automatically upon user login. Enables smart tunnel access upon user login, but requires the user to start smart tunnel access manually, using the **Application Access > Start Smart Tunnels** button on the Clientless SSL VPN portal page. Prevents smart tunnel access. Removes a **smart-tunnel** command from the group policy or username configuration, which then inherits the [**no**] **smart-tunnel** command from the default group-policy. The keywords following the **no smart-tunnel** command are optional, however, they restrict the removal to the named **smart-tunnel** command.
Step 2	Refer to Automating Smart Tunnel Access for the required option.	

Configuring and Applying Smart Tunnel Policy

The smart tunnel policy requires a per group policy/username configuration. Each group policy/username references a globally configured list of networks. When the smart tunnel is turned on, you can allow traffic outside of the tunnel with the use of 2 CLIs: one configures the network (a set of hosts), and the other uses the specified smart-tunnel network to enforce a policy on a user. The following commands create a list of hosts to use for configuring smart tunnel policies:

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**[no] smart-tunnel network** *network name* **ip** *ip netmask*	Creates a list of hosts to use for configuring smart tunnel policies. *network name* is the name to apply to the tunnel policy. *ip* is the IP address of the network. *netmask* is the netmask of the network.
Step 3	**[no] smart-tunnel network** *network name* **host** *host mask*	Establishes the hostname mask, such as *.cisco.com.
Step 4	**[no] smart-tunnel tunnel-policy [{excludespecified | tunnelspecified}** *network name* **| tunnelall]** OR **[no] smart-tunnel tunnel-policy {excludespecified | tunnelspecified}** *network* ***name*** **| tunnelall]**	Applies smart tunnel policies to a particular group or user policy. *network name* is a list of networks to be tunneled. *tunnelall* makes everything tunneled (encrypted). *tunnelspecified* tunnels only networks specified by network name. *excludespecified* tunnels only networks that are outside of the networks specified by network name.

Configuring and Applying a Smart Tunnel Tunnel Policy

Like the split tunnel configuration in the SSL VPN client, the smart tunnel policy is a per group-policy/username configuration. Each group policy/username references a globally configured list of networks:

Command	Purpose
[no] smart-tunnel tunnel-policy [{excludespecified | tunnelspecified} *network name* **| tunnelall]** or **[no] smart-tunnel tunnel-policy [{excludespecified | tunnelspecified}** *network name* **| tunnelall]**	References a globally configured list of networks. *network name* is a list of networks to be tunneled. *tunnelall* makes everything tunneled (encrypted). tunnelspecified tunnels only networks specified by network name. excludespecified tunnels only networks that are outside of the networks specified by network name.

Command	Purpose
ciscoasa(config-webvpn)# [no] smart-tunnel network *network name* **ip** *ip netmask* **ciscoasa(config-webvpn)# [no] smart-tunnel network** *network name* **host** *host mask*	Applies a tunnel policy to a group-policy/user policy. One command specifies host and the other specifies network IPs; use only one. *network name*—name of network to apply to tunnel policy *ip address*—IP address of a network *netmask*—netmask of a network *host mask*—hostname mask, such as *.cisco.com
Example: ciscoasa(config-webvpn)# **smart-tunnel network inventory ip 10.5.2.2** ciscoasa(config-webvpn)# **smart-tunnel network inventory host www.example.com**	Smart tunnel policy configuration is a good option when a vendor wants to provide a partner with clientless access to an internal inventory server page upon login without going through the clientless portal first. Creates a tunnel policy that contains only one host (assuming the inventory pages are hosted at www.example.com (10.5.2.2), and you want to configure both IP address and name for the hosts).
ciscoasa(config-group-webvpn)# **smart-tunnel tunnel-policy tunnelspecified inventory**	Applies the tunnel-specified tunnel policy to the partner's group policy.
(Optional) ciscoasa(config-group-webvpn)# **homepage value http://www.example.com** ciscoasa(config-group-webvpn)# h**omepage use-smart-tunnel**	Specifies the group policy home page and enables smart tunnel on it. Without writing a script or uploading anything, an administrator can specify which homepage to connect with via smart tunnel.
(Optional) ciscoasa(config-webvpn)# **smart-tunnel notification-icon**	By default, configuration of a smart tunnel application is not necessary because all processes initiated by the browser with smart tunnel enabled have access to the tunnel. However, because no portal is visible, you may want to enable the logout notification icon.

Creating a Smart Tunnel Auto Sign-On Server List

Command	Purpose
webvpn	Switches to Clientless SSL VPN configuration mode.
smart-tunnel auto-sign-on *list* [*use-domain*] [**realm** *realm-string*] [**port** *port-num*]{**ip** *ip-address* [*netmask*] | **host** *hostname-mask*}	Use for each server to add to the server list • *list*—names the list of remote servers. Use quotation marks around the name if it includes a space. The string can be up to 64 characters. The ASA creates the list if it is not already present in the configuration. Otherwise, it adds the entry to the list. Assign a name that will help you to distinguish. • use-domain (optional)—Adds the Windows domain to the username if authentication requires it. If you enter this keyword, ensure you specify the domain name when assigning the smart tunnel list to one or more group policies, or usernames. • *realm*—Configures a realm for the authentication. Realm is associated with the protected area of the website and is passed back to the browser either in the authentication prompt or in the HTTP headers during authentication. Once auto-sign is configured and a realm string is specified, users can configure the realm string on a Web application (such as Outlook Web Access) and access Web applications without signing on • *port*—Specifies which port performs auto sign-on. For Firefox, if no port number is specified, auto sign is performed on HTTP and HTTPS, accessed by the default port numbers 80 and 443 respectively. • *ip*—Specifies the server by its IP address and netmask. • *ip-address*[*netmask*]—Identifies the sub-network of hosts to auto-authenticate to. • *host*—Specifies the server by its hostname or wildcard mask. Using this option protects the configuration from dynamic changes to IP addresses. • *hostname-mask*—Specifies which hostname or wildcard mask to auto-authenticate to.
(Optional) [**no**] **smart-tunnel auto-sign-on** *list* [**use-domain**] [**realm** *realm-string*] [**port** *port-num*] {**ip** *ip-address* [*netmask*] | **host** *hostname-mask*}	Removes an entry from the list of servers, specifying both the list and IP address or hostname as it appears in the ASA configuration.

Command	Purpose
`show running-config webvpn smart-tunnel`	Displays the smart tunnel auto sign-on list entries.
`config-webvpn`	Switches to config-webvpn configuration mode.
`smart-tunnel auto-sign-on HR use-domain ip 93.184.216.119 255.255.255.0`	Adds all hosts in the subnet and adds the Windows domain to the username if authentication requires it.
(Optional) `no smart-tunnel auto-sign-on HR use-domain ip 93.184.216.119 255.255.255.0`	Removes that entry from the list and the list named HR if the entry removed is the only entry in the list.
`no smart-tunnel auto-sign-on HR`	Removes the entire list from the ASA configuration.
`smart-tunnel auto-sign-on intranet host *.example.com`	Adds all hosts in the domain to the smart tunnel auto sign-on list named intranet.
`no smart-tunnel auto-sign-on intranet host *.example.com`	Removes that entry from the list.

Following the configuration of the smart tunnel auto sign-on server list, you must assign it to a group policy or a local user policy for it to become active, as described in the next section.

The next step is to add servers to the server list.

Adding Servers to a Smart Tunnel Auto Sign-On Server List

The following steps describe how to add servers to the list of servers for which to provide auto sign-on in smart tunnel connections, and assign that list to a group policies or a local user.

Prerequisites

You must use the **smart-tunnel auto-sign-on** *list* command to create a list of servers first. You can assign only one list to a group policy or username.

Restrictions

- The smart-tunnel auto sign-on feature supports only applications communicating HTTP and HTTPS using Internet Explorer and Firefox.
- Firefox requires the administrator to specify hosts using an exact hostname or IP address (instead of a host mask with wildcards, a subnet using IP addresses, or a netmask). For example, within Firefox, you cannot enter *.cisco.com and expect auto sign-on to host email.cisco.com.

DETAILED STEPS

To enable smart tunnel auto sign-on in clientless (browser-based) SSL VPN sessions, use the following commands:

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**group-policy webvpn** or **username webvpn**	Switches to group-policy Clientless SSL VPN configuration mode. Switches to username Clientless SSL VPN configuration mode.
Step 3	**smart-tunnel auto-sign-on enable**	Enables smart tunnel auto sign-on Clientless SSL VPN sessions.
Step 4	(Optional) [**no**] **smart-tunnel auto-sign-on enable** *list* [**domain** *domain*]	Switches off smart tunnel auto sign-on Clientless SSL VPN session, removes it from the group policy or username, and uses the default. • *list*—The name of a smart tunnel auto sign-on list already present in the ASA Clientless SSL VPN configuration. • (Optional) *domain*—The name of the domain to be added to the username during authentication. If you enter a domain, enter the **use-domain** keyword in the list entries.
Step 5	**show running-config webvpn smart-tunnel**	Views the smart tunnel auto sign-on list entries in the SSL VPN configuration.
Step 6	**smart-tunnel auto-sign-on enable HR**	Enables the smart tunnel auto sign-on list named HR.
Step 7	**smart-tunnel auto-sign-on enable HR domain CISCO**	Enables the smart tunnel auto sign-on list named HR and adds the domain named CISCO to the username during authentication.
Step 8	**(Optional)** **no smart-tunnel auto-sign-on enable HR**	Removes the smart tunnel auto sign-on list named HR from the group policy and inherits the smart tunnel auto sign-on list command from the default group policy.

•

Automating Smart Tunnel Access

To start smart tunnel access automatically upon user login, enter the following commands:

Requirements

For Mac OS X, you must click the link for the application in the portal's Application Access panel, with or without auto-start configured.

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**group-policy webvpn** or **username webvpn**	Switches to group-policy Clientless SSL VPN configuration mode. Switches to username Clientless SSL VPN configuration mode.

Step 3	**smart-tunnel auto-start** *list*
	Example:
	`ciscoasa(config-group-policy)# webvpn`
	`ciscoasa(config-group-webvpn)# smart-tunnel auto-start apps1`
		Starts smart tunnel access automatically upon user login. *list* is the name of the smart tunnel list already present.
		Assigns the smart tunnel list named apps1 to the group policy.

Step 4	**show running-config webvpn smart-tunnel**	Displays the smart tunnel list entries in the SSL VPN configuration.
Step 5	(Optional) **no smart-tunnel**	Removes the **smart-tunnel** command from the group policy or username and reverts to the default.

Enabling and Switching Off Smart Tunnel Access

By default, smart tunnels are switched off.

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**group-policy webvpn** or **username webvpn**	Switches to group-policy Clientless SSL VPN configuration mode. Switches to username Clientless SSL VPN configuration mode.

Step 3	**smart-tunnel [enable** *list* **| disable]**
	Example:
	`ciscoasa(config-group-policy)# webvpn`
	`ciscoasa(config-group-webvpn)# smart-tunnel enable apps1`
		Enables smart tunnel access. *list* is the name of the smart tunnel list already present. You do not have to start smart tunnel access manually if you entered **smart-tunnel auto-start list** from the previous table.
		Assigns the smart tunnel list named apps1 to the group policy.

	Command	Purpose
Step 4	`show running-config webvpn smart-tunnel`	Dislpays the smart tunnel list entries in the SSL VPN configuration.
Step 5	(Optional) `no smart-tunnel`	Removes the **smart-tunnel** command from the group policy or local user policy and reverts to the default group policy.
Step 6	(Optional) `smart-tunnel disable`	Switches off smart tunnel access.

Configuring Smart Tunnel Log Off

This section describes how to ensure that the smart tunnel is properly logged off. Smart tunnel can be logged off when all browser windows have been closed, or you can right click the notification icon and confirm log out.

Note We strongly recommend the use of the logout button on the portal. This method pertains to Clientless SSL VPNs and logs off regardless of whether smart tunnel is used or not. The notification icon should be used only when using standalone applications without the browser.

When Its Parent Process Terminates

This practice requires the closing of all browsers to signify log off. The smart tunnel lifetime is now tied to the starting process lifetime. For example, if you started a smart tunnel from Internet Explorer, the smart tunnel is turned off when no iexplore.exe is running. Smart tunnel can determine that the VPN session has ended even if the user closed all browsers without logging out.

Note In some cases, a lingering browser process is unintentional and is strictly a result of an error. Also, when a Secure Desktop is used, the browser process can run in another desktop even if the user closed all browsers within the secure desktop. Therefore, smart tunnel declares all browser instances gone when no more visible windows exist in the current desktop.

DETAILED STEPS

	Command	Purpose
Step 1	`[no] smart-tunnel notification-icon`	Allows administrators to turn on the notification icon on a global basis. This command configures log out properties and controls whether the user is presented with a logout icon for logging out, as opposed to having logout triggered by closing browser windows. This command also controls logging off when a parent process terminates, which is automatically turned on or off when the notification icon is turned on or off. **notification-icon** is the keyword that specifies when to use the icon for logout. **Note** The no version of this command is the default, in which case, closing all browser windows logs off the SSL VPN session. **Note** Portal logout still takes effect and is not impacted.
Step 2	`*.webvpn.`	When using a proxy and adding to the proxy list exception, ensures that smart tunnel is properly closed when you log off, regardless of icon usage or not.

With a Notification Icon

You may also choose to switch off logging off when a parent process terminates so that a session survives if you close a browser. For this practice, you use a notification icon in the system tray to log out. The icon remains until the user clicks the icon to logout. If the session has expired before the user has logged out, the icon remains until the next connection is tried. You may have to wait for the session status to update in the system tray.

Note This icon is an alternative way to log out of SSL VPN. It is not an indicator of VPN session status.

Configuring Content Transformation

By default, the ASA processes all Clientless SSL VPN traffic through a content transformation/rewriting engine that includes advanced elements such as JavaScript and Java to proxy HTTP traffic that may have different semantics and access control rules depending on whether the user is accessing an application within or independently of an SSL VPN device.

Some Web resources require highly individualized treatment. The following sections describe functionality that provides such treatment:

- Configuring a Certificate for Signing Rewritten Java Content
- Switching Off Content Rewrite

- Using Proxy Bypass

Subject to the requirements of your organization and the Web content involved, you may use one of these features.

Configuring a Certificate for Signing Rewritten Java Content

Java objects that have been transformed by Clientless SSL VPN can subsequently be signed using a PKCS12 digital certificate associated with a trustpoint.

DETAILED STEPS

	Command	Purpose
Step 1	**crypto ca import**	Imports a certificate.
Step 2	**ava-trustpoint** **Example:t** ciscoasa(config)# **crypto ca import mytrustpoint pkcs12 mypassphrase** Enter the base 64 encoded PKCS12. End with the word "quit" on a line by itself. [PKCS12 data omitted] **quit** INFO: Import PKCS12 operation completed successfully. ciscoasa(config)# **webvpn** ciscoasa(config)# **java-trustpoint mytrustpoint**	Employs a certificate. Shows the creation of a trustpoint named mytrustpoint and its assignment to signing Java objects.

Switching Off Content Rewrite

You may not want some applications and Web resources, for example, public websites, to go through the ASA. The ASA therefore lets you create rewrite rules that let users browse certain sites and applications without going through the ASA. This is similar to split-tunneling in an IPsec VPN connection.

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**rewrite**	Specifies applications and resources to access outside a clientless SSLN VPN tunnel. You can use this command multiple times.
Step 3	**disable**	Used in combination with the **rewrite** command. The rule order number is important because the security appliance searches rewrite rules by order number, starting with the lowest, and applies the first rule that matches.

Using Proxy Bypass

You can configure the ASA to use proxy bypass when applications and Web resources work better with the special content rewriting this feature provides. Proxy bypass is an alternative method of content rewriting that makes minimal changes to the original content. It is often useful with custom Web applications.

You can use the **proxy-bypass** command multiple times. The order in which you configure entries is unimportant. The interface and path mask or interface and port uniquely identify a proxy bypass rule.

If you configure proxy bypass using ports rather than path masks, depending on your network configuration, you may need to change your firewall configuration to allow these ports access to the ASA. Use path masks to avoid this restriction. Be aware, however, that path masks can change, so you may need to use multiple pathmask statements to exhaust the possibilities.

A path is everything in a URL after the .com or .org or other types of domain name. For example, in the URL www.example.com/hrbenefits, *hrbenefits* is the path. Similarly, for the URL www.example.com/hrinsurance, *hrinsurance* is the path. To use proxy bypass for all hr sites, you can avoid using the command multiple times by using the * wildcard as follows: /hr*.

DETAILED STEPS

	Command	Purpose
Step 1	`webvpn`	Switches to Clientless SSL VPN configuration mode.
Step 2	`proxy-bypass`	Configures proxy bypass.

Configuring Portal Access Rules

This enhancement allows customers to configure a global Clientless SSL VPN access policy to permit or deny Clientless SSL VPN sessions based on the data present in the HTTP header. If the ASA denies a Clientless SSL VPN session, it returns an error code to the endpoint immediately.

The ASA evaluates this access policy before the endpoint authenticates to the ASA. As a result, in the case of a denial, fewer ASA processing resources are consumed by additional connection attempts from the endpoint.

Prerequisites

Log on to the ASA and enter global configuration mode. In global configuration mode, the ASA displays this prompt:

```
hostname(config)#
```

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn** **Example:** `ciscoasa(config)# webvpn`	Enter Clientless SSL VPN configuration mode.

Step 2	**portal-access-rule** *priority* [{**permit** | **deny** [**code** *code*]} {**any** | **user-agent match** *string*}
	
	Example:
	`hostname(config-webvpn)# portal-access-rule 1 deny code 403 user-agent match *Thunderbird*`
	
	`hostname(config-webvpn)# portal-access-rule 1 deny code 403 user-agent match "*my agent*"`
		Permit or deny the creation of a Clientless SSL VPN session based on an HTTP header code or a string in the HTTP header.
		
		The second example shows the proper syntax for specifying a string with a space. Surround the string with wildcards (*) and then quotes (" ").

Optimizing Clientless SSL VPN Performance

The ASA provides several ways to optimize Clientless SSL VPN performance and functionality. Performance improvements include caching and compressing Web objects. Functionality tuning includes setting limits on content transformation and **proxy-bypass**. APCF provides an additional method of tuning content transformation. These sections explain these features:

- Configuring Caching
- Configuring Content Transformation

Configuring Caching

Caching enhances Clientless SSL VPN performance. It stores frequently reused objects in the system cache, which reduces the need to perform repeated rewriting and compressing of content. It reduces traffic between Clientless SSL VPN and the remote servers, with the result that many applications run much more efficiently.

By default, caching is enabled. You can customize the way caching works for your environment by using the caching commands in cache mode.

Clientless SSL VPN Remote Users

September 13, 2013

This section is for the system administrator who sets up Clientless (browser-based) SSL VPN for end users. It summarizes configuration requirements and tasks for the user remote system. It also specifies information to communicate to users to get them started using Clientless SSL VPN. This section includes the following topics:

- Requiring Usernames and Passwords
- Communicating Security Tips
- Configuring Remote Systems to Use Clientless SSL VPN Features
- Capturing Clientless SSL VPN Data

Note We assume you have already configured the ASA for Clientless SSL VPN.

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. Ensure users have the required access.

Table 18-1 lists the type of usernames and passwords that Clientless SSL VPN users may need to know.

Table 18-1 Usernames and Passwords to Give to Clientless SSL VPN Users

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 a Clientless SSL VPN session
File Server	Access remote file server	Using the Clientless SSL VPN file browsing feature to access a remote file server

Table 18-1 Usernames and Passwords to Give to Clientless SSL VPN Users (continued)

Login Username/ Password Type	Purpose	Entered When
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 always to log out from the session. To log out of Clientless SSL VPN, click the logout icon on the Clientless SSL VPN toolbar or close the browser.

Advise users that using Clientless SSL VPN does not ensure that communication with every site is secure. Clientless SSL VPN ensures the security of data transmission between the remote computer or workstation and the ASA on the corporate network. 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 secure.

Configuring Remote Systems to Use Clientless SSL VPN Features

Table 18-2 includes the following information about setting up remote systems to use Clientless SSL VPN:

- Starting Clientless SSL VPN
- Using the Clientless SSL VPN Floating Toolbar
- Web Browsing
- Network Browsing and File Management
- Using Applications (Port Forwarding)
- Using email via Port Forwarding, Web Access, or Email Proxy

Table 18-2 also provides information about the following:

- Clientless SSL VPN requirements, by feature
- Clientless SSL VPN supported applications
- Client application installation and configuration requirements
- Information you may need to provide end users
- Tips and use suggestions for end users

It is possible that you have configured user accounts differently, and that different features are available to each Clientless SSL VPN user. Table 18-2 organizes information by user activity, so that you can skip over the information for unavailable features.

Table 18-2 Clientless SSL VPN Remote System Configuration and End User Requirements

Task	Remote System or End User Requirements	Specifications or Use Suggestions
Starting Clientless SSL VPN	Connection to the Internet	Any Internet connection is supported, including: • Home DSL, cable, or dial-up • Public kiosks • Hotel hook-ups • Airport wireless nodes • Internet cafes
	Clientless SSL VPN-supported browser	We recommend the following browsers for Clientless SSL VPN. Other browsers may not fully support Clientless SSL VPN features. On Microsoft Windows: • Internet Explorer 8 • Firefox 8 On Linux: • Firefox 8 On Mac OS X: • Safari 5 • Firefox 8
	Cookies enabled on browser	Cookies must be enabled on the browser in order to access applications via port forwarding.
	URL for Clientless SSL VPN	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 Clientless SSL VPN is enabled. For example: https://10.89.192.163 or https://cisco.example.com.
	Clientless SSL VPN username and password	
	[Optional] Local printer	Clientless SSL VPN does not support printing from a Web browser to a network printer. Printing to a local printer is supported.

Table 18-2 Clientless SSL VPN Remote System Configuration and End User Requirements (continued)

Task	Remote System or End User Requirements	Specifications or Use Suggestions
Using the Floating Toolbar in a Clientless SSL VPN Connection		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. If you configure your browser to block popups, the floating toolbar cannot display. 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 not enabled on the Clientless SSL VPN toolbar.)
Web Browsing	Usernames and passwords for protected websites	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 Clientless SSL VPN title bar 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.

Table 18-2 Clientless SSL VPN Remote System Configuration and End User Requirements (continued)

Task	Remote System or End User Requirements	Specifications or Use Suggestions
Network Browsing and File Management	File permissions configured for shared remote access	Only shared folders and files are accessible via Clientless SSL VPN.
	Server name and passwords for protected file servers	—
	Domain, workgroup, and server names where folders and files reside	Users may not be familiar with how to locate their files through your organization network.
	—	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.

Table 18-2 Clientless SSL VPN Remote System Configuration and End User Requirements (continued)

Task	Remote System or End User Requirements	Specifications or Use Suggestions
Using Applications (called Port Forwarding or Application Access)	**Note** On Mac OS X, only the Safari browser supports this feature.	
	Note 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, it is unlikely that users will be able to use applications when they connect from public remote systems.	
	⚠ **Caution** Users should always close the Application Access window when they finish using applications by clicking the **Close** icon. Failure to close the window properly can cause Application Access or the applications themselves to be inaccessible.	
	Client applications installed	—
	Cookies enabled on browser	—
	Administrator privileges	User must have administrator access on the computer if you use DNS names to specify servers because modifying the hosts file requires it.
	Oracle Java Runtime Environment (JRE) version 1.4.x and 1.5.x installed. JavaScript must be enabled on the browser. By default, it is enabled.	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: 1. Clear the browser cache and close the browser. 2. Verify that no Java icons are in the computer task bar. Close all instances of Java. 3. Establish a Clientless SSL VPN session and launch the port forwarding Java applet.
	Client applications configured, if necessary. **Note** The Microsoft Outlook client does not require this configuration step. All non-Windows client applications require configuration. To see if configuration is necessary for a Windows application, check the value of the Remote Server. • If the Remote Server 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.	To configure the client application, use the server's locally mapped IP address and port number. To find this information: 1. Start Clientless SSL VPN on the remote system and click the Application Access link on the Clientless SSL VPN 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 Clientless SSL VPN does not open the site over Clientless SSL VPN. To open a site over Clientless SSL VPN, cut and paste the URL into the Enter (URL) Address field.	

Table 18-2 Clientless SSL VPN Remote System Configuration and End User Requirements (continued)

Task	Remote System or End User Requirements	Specifications or Use Suggestions
Using email via Application Access	Fulfill requirements for Application Access (See Using Applications)	To use mail, 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.	
	Other email clients	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	Web-based email product installed	Supported products include: • Outlook Web Access For best results, use OWA on Internet Explorer 8.x or higher, or Firefox 8.x. • Lotus Notes Other web-based email products should also work, but we have not verified them.
Using email via email Proxy	SSL-enabled mail application installed Do not set the ASA SSL version to TLSv1 Only. Outlook and Outlook Express do not support TLS.	Supported mail applications: • Microsoft Outlook • Microsoft Outlook Express versions 5.5 and 6.0 Other SSL-enabled mail clients should also work, but we have not verified them.
	Mail application configured	

Capturing Clientless SSL VPN Data

The CLI capture command lets you log information about websites that do not display properly over a Clientless SSL VPN connection. This data can help your Cisco customer support engineer troubleshoot problems. The following sections describe how to use the capture command:

- Creating a Capture File
- Using a Browser to Display Capture Data

Note Enabling Clientless SSL VPN capture affects the performance of the security appliance. Ensure you switch off the capture after you generate the capture files needed for troubleshooting.

Creating a Capture File

DETAILED STEPS

Step 1 To start the Clientless SSL VPN capture utility, use the **capture** command from privileged EXEC mode.

capture *capture-name* **type webvpn user** *csslvpn-username*

where:

- *capture-name* is a name you assign to the capture, which is also prefixed to the name of the capture files.
- *csslvpn-username* is the username to match for capture.

The capture utility starts.

Step 2 A user logs in to begin a Clientless SSL VPN session. The capture utility is capturing packets.

Stop the capture by using the **no** version of the command.

no capture *capture-name*

The capture utility creates a *capture-name*.zip file, which is encrypted with the password **koleso**

Step 3 Send the .zip file to Cisco, or attach it to a Cisco TAC service request.

Step 4 To look at the contents of the .zip file, unzip it using the password **koleso**.

The following example creates a capture named *hr*, which captures Clientless SSL VPN traffic for user2 to a file:

```
hostname# capture hr type webvpn user user2
WebVPN capture started.
   capture name   hr
   user name      user2
hostname# no capture hr
```

Using a Browser to Display Capture Data

DETAILED STEPS.

Step 1 To start the Clientless SSL VPN capture utility, use the **capture** command from privileged EXEC mode.

capture *capture-name* **type webvpn user** *csslvpn-username*

where:

- *capture-name* is a name you assign to the capture, which is also prefixed to the name of the capture files.
- *csslvpn-username* is the username to match for capture.

The capture utility starts.

Step 2 A user logs in to begin a Clientless SSL VPN session. The capture utility is capturing packets.

Stop the capture by using the **no** version of the command.

Step 3 Open a browser and in the address box enter:

https://*IP address or hostname of the ASA***/webvpn_capture.html**

The captured content displays in a sniffer format.

Step 4 When you finish examining the capture content, stop the capture by using the **no** version of the command.

Configuring Clientless SSL VPN Users

September 13, 2013

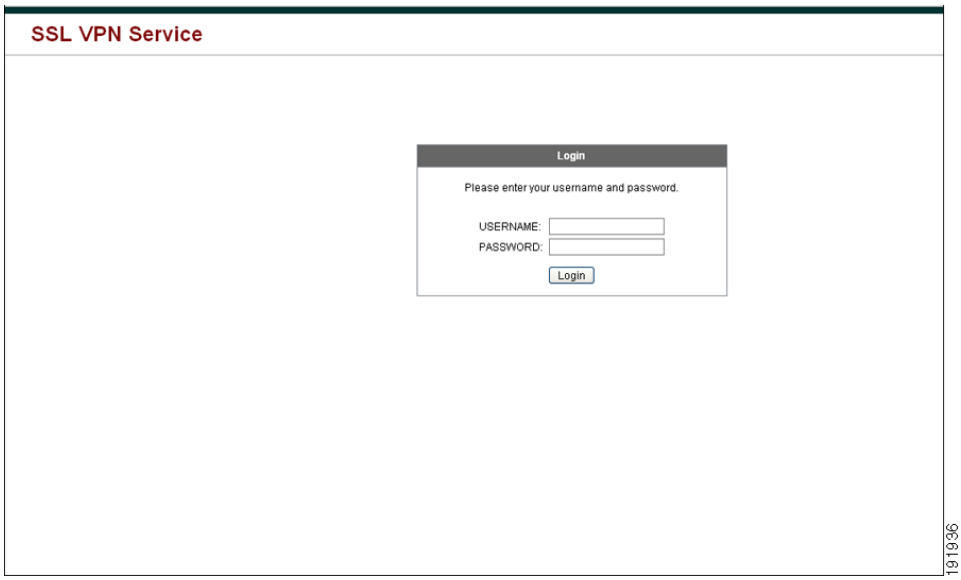
Overview

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

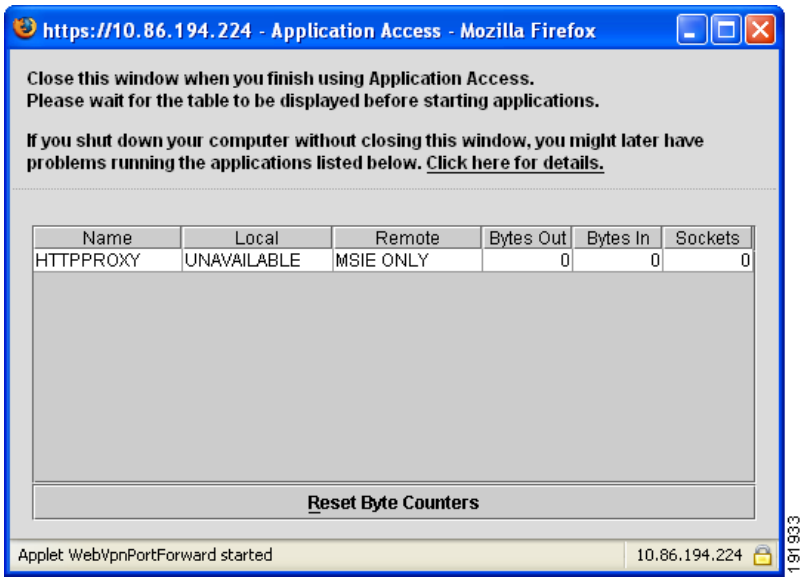
After the user logs in, the portal page opens.

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



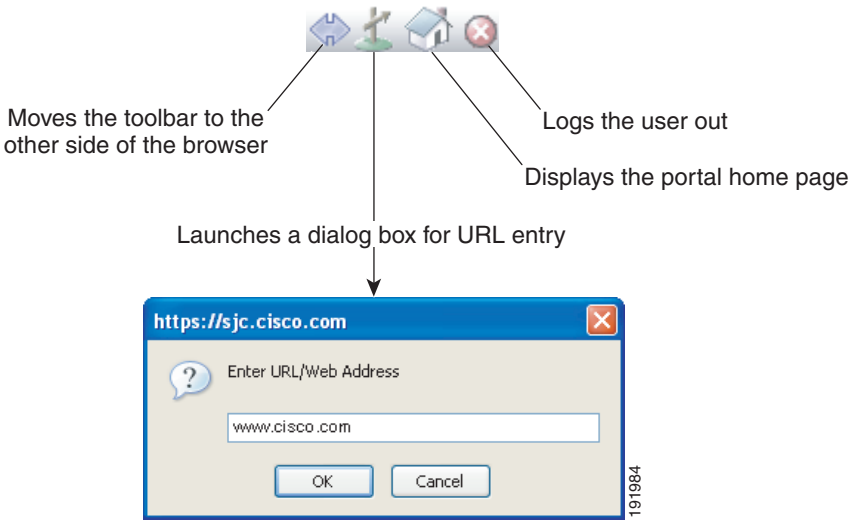
This window displays the TCP applications configured for this Clientless SSL VPN connection. To use an application with this panel open, the user starts the application in the normal way.

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.

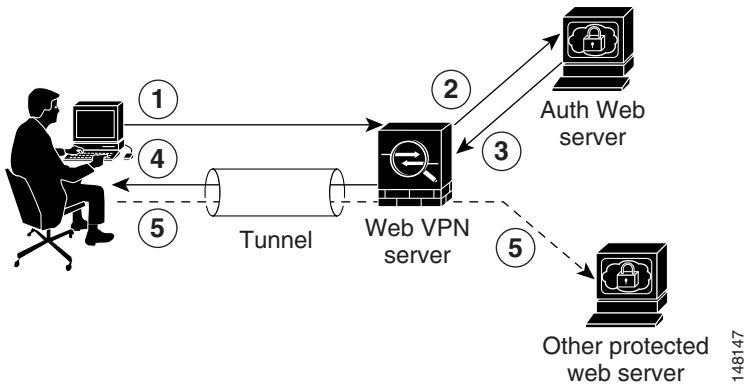
Step 2 The Clientless SSL VPN server acts as a proxy for the user and forwards the form data (username and password) to an authenticating Web server using a POST authentication request.

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*.

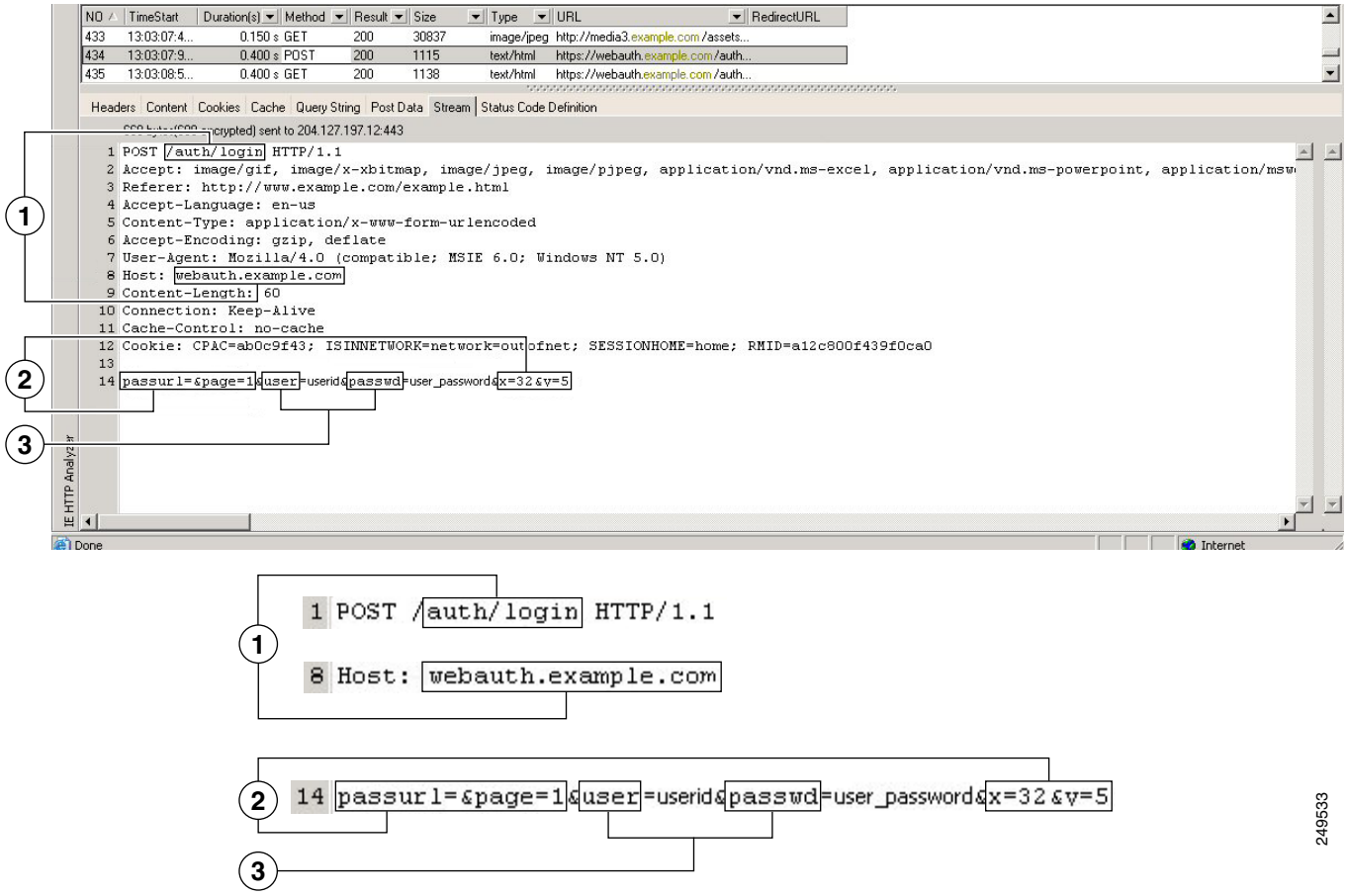
b. Password parameter. In the preceding example, this parameter is *USER_PASSWORD*.

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

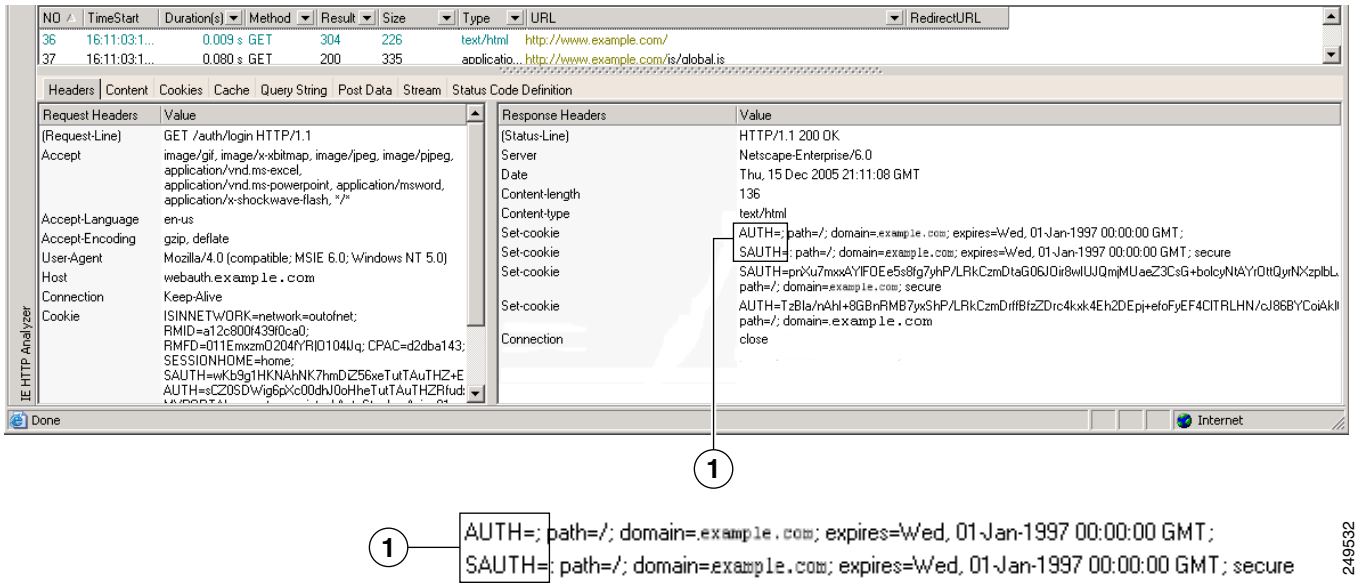
Step 6 If you successfully log on to the Web server, examine the server response with the HTTP header analyzer to locate the name of the session cookie set by the server in your browser. This is the **auth-cookie-name** parameter.

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.

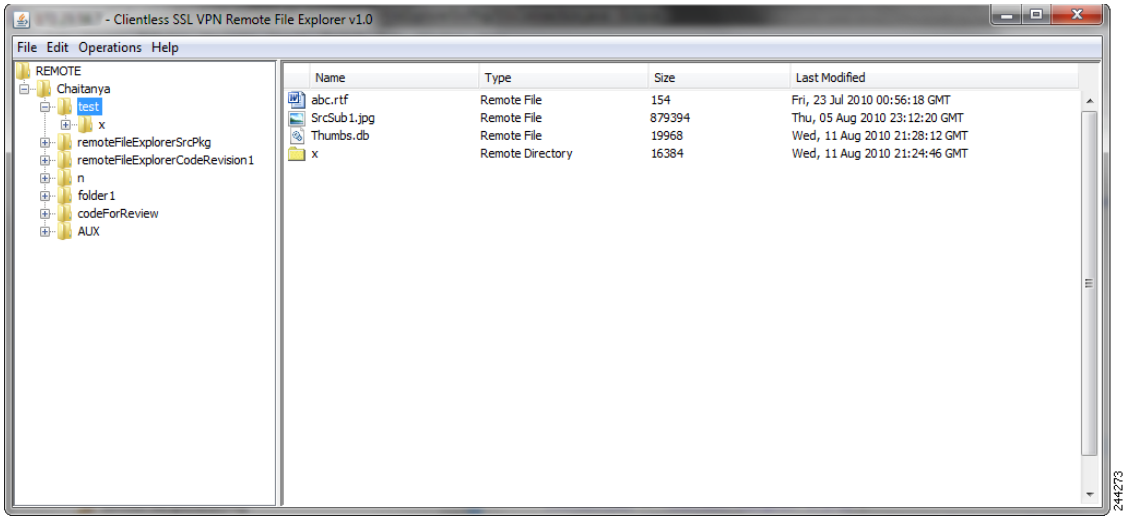
Restrictions

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.

Using Clientless SSL VPN with Mobile Devices

September 13, 2013

Using Clientless SSL VPN with Mobile Devices

You can access Clientless SSL VPN from your Pocket PC or other certified mobile device. Neither the ASA administrator nor the Clientless SSL VPN user need do anything special to use Clientless SSL VPN with a certified mobile device.

Cisco has certified the following mobile device platforms:

HP iPaq H4150

Pocket PC 2003

Windows CE 4.20.0, build 14053

Pocket Internet Explorer (PIE)

ROM version 1.10.03ENG

ROM Date: 7/16/2004

Some differences in the mobile deviceversion of Clientless SSL VPN exist:

- A banner Web page replaces the popup Clientless SSL VPN window.
- An icon bar replaces the standard Clientless SSL VPN floating toolbar. This bar displays the Go, Home and Logout buttons.
- The Show Toolbar icon is not included on the main Clientless SSL VPN portal page.
- Upon Clientless SSL VPN logout, a warning message provides instructions for closing the PIE browser properly. If you do not follow these instructions and you close the browser window in the common way, PIE does not disconnect from Clientless SSL VPN or any secure website that uses HTTPS.

Restrictions

- Clientless SSL VPN supports OWA 2000 and OWA 2003 Basic Authentication. If Basic Authentication is not configured on an OWA server and a Clientless SSL VPN user attempts to access that server, access is denied.
- Unsupported Clientless SSL VPN features:
  - Application Access and other Java-dependent features.

- HTTP proxy.
- The Citrix Metaframe feature (if the PDA does not have the corresponding Citrix ICA client software).

Customizing Clientless SSL VPN

September 13, 2013

Clientless SSL VPN End User Setup

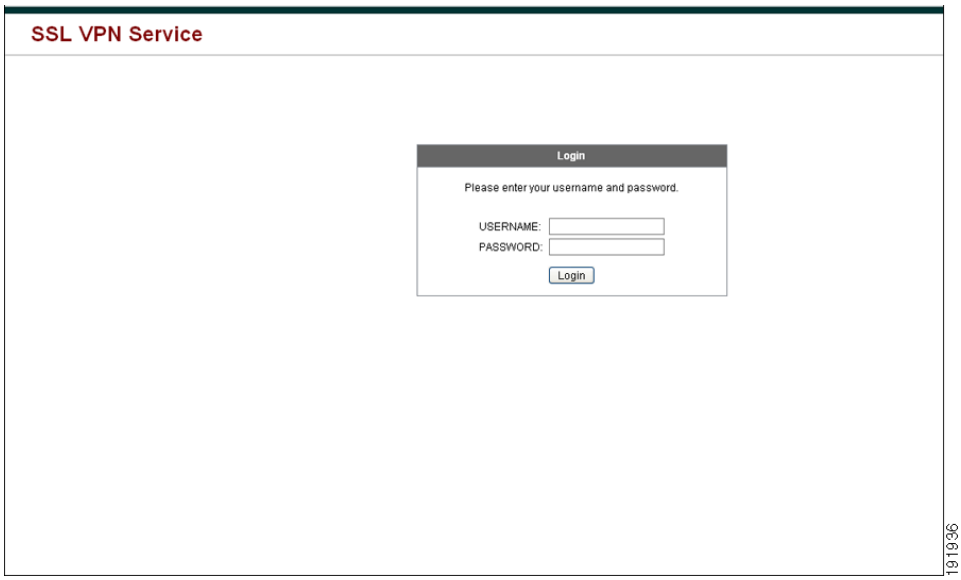
This section is for the system administrator who sets up Clientless SSL VPN for end users. It describes how to customize the end-user interface.

This section summarizes configuration requirements and tasks for a remote system. It specifies 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 21-1).

Figure 21-1 Clientless SSL VPN Login Screen



Viewing the Clientless SSL VPN Home Page

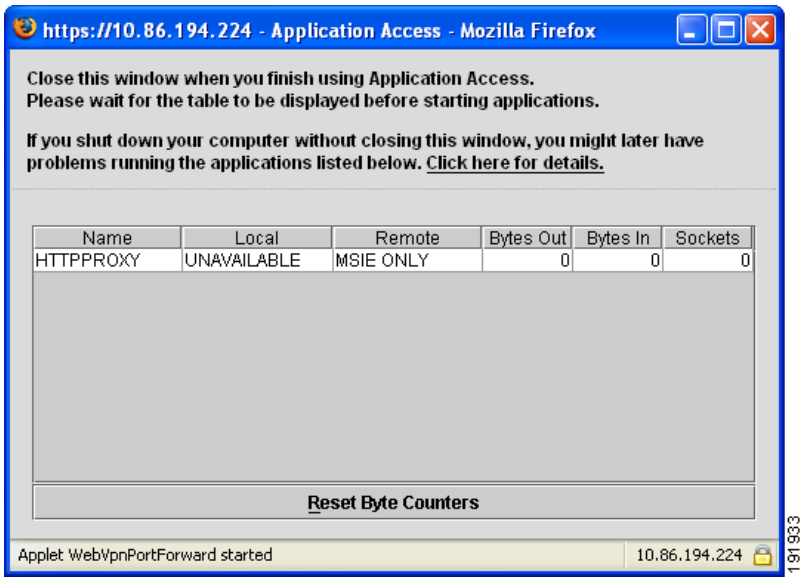
After the user logs in, the portal page opens.

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 21-2).

Figure 21-2 Clientless SSL VPN Application Access Window



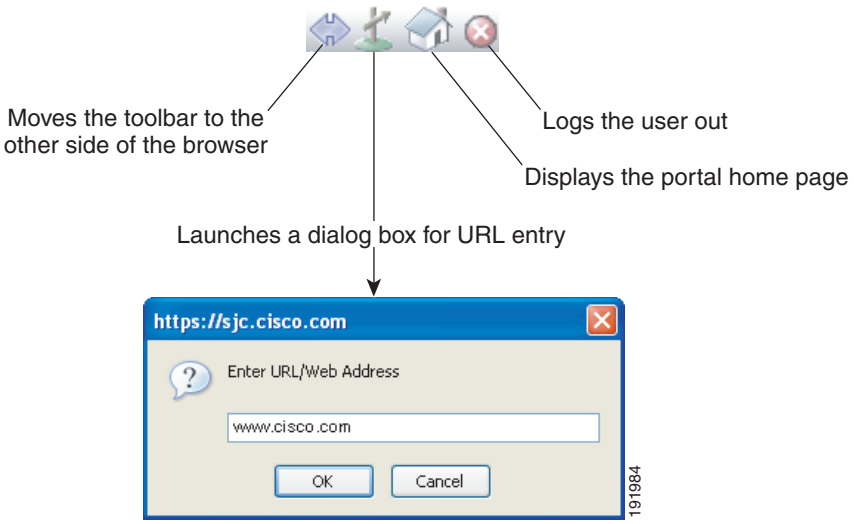
This window displays the TCP applications configured for this Clientless SSL VPN connection. To use an application with this panel open, the user starts the application in the normal way.

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 21-3 represents the current Clientless SSL VPN session.

Figure 21-3 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.

Customizing Clientless SSL VPN Pages

You can change the appearance of the portal pages displayed to Clientless SSL VPN users. This includes the Login page displayed to users when they connect to the security appliance, the Home page displayed to users after the security appliance authenticates them, the Application Access window displayed when users launch an application, and the Logout page displayed when users log out of Clientless SSL VPN sessions.

After you customize the portal pages, you can save your customization and apply it to a specific connection profile, group policy, or user. The changes do not take effect until you reload the ASA, or you switch off and then enable clientless SSL.

You can create and save many customization objects, enabling the security appliance to change the appearance of portal pages for individual users or groups of users.

This section includes the following topics:

Information About Customization

The ASA uses customization objects to define the appearance of user screens. A customization object is compiled from an XML file which contains XML tags for all the customizable screen items displayed to remote users. The ASA software contains a customization template that you can export to a remote PC. You can edit this template and import the template back into the ASA as a new customization object.

When you export a customization object, an XML file containing XML tags is created at the URL you specify. The XML file created by the customization object named *Template* contains empty XML tags, and provides the basis for creating new customization objects. This object cannot be changed or deleted from cache memory but can be exported, edited, and imported back into the ASA as a new customization object.

Customization Objects, Connection Profiles, and Group Policies

Initially, when a user first connects, the default customization object (named *DfltCustomization*) identified in the connection profile (tunnel group) determines how the logon screen appears. If the connection profile list is enabled, and the user selects a different group which has its own customization, the screen changes to reflect the customization object for that new group.

After the remote user is authenticated, the screen appearance is determined by whether a customization object that has been assigned to the group policy.

Exporting a Customization Template

When you export a customization object, an XML file is created at the URL you specify. The customization template (named *Template*) contains empty XML tags and provides the basis for creating new customization objects. This object cannot be changed or deleted from cache memory but can be exported, edited, and imported back into the ASA as a new customization object.

DETAILED STEPS

	Command	Purpose
Step 1	**export webvpn customization**	Exports a customization object and allows you to make changes to the XML tags.
Step 2	**import webvpn customization** **Example:** hostname# **export webvpn customization DfltCustomization tftp://209.165.200.225/dflt_custom** !!!!!!!!!!!!!!!!INFO: Customization object 'DfltCustomization' was exported to tftp://10.86.240.197/dflt_custom hostname#	Imports the file as a new object. Exports the default customization object (DfltCustomization) and creates the XML file named *dflt_custom*.

Editing the Customization Template

This section shows the contents of the customization template and has convenient figures to help you quickly choose the correct XML tag and make changes that affect the screens.

You can use a text editor or an XML editor to edit the XML file. The following example shows the XML tags of the customization template. Some redundant tags have been removed for easier viewing:

Example:

```
<custom>
   <localization>
      <languages>en,ja,zh,ru,ua</languages>
      <default-language>en</default-language>
   </localization>
    <auth-page>
      <window>
          <title-text l10n="yes"><![CDATA[SSL VPN Service]]></title-text>
      </window>
      <full-customization>
          <mode>disable</mode>
          <url></url>
      </full-customization>
      <language-selector>
        <mode>disable</mode>
        <title l10n="yes">Language:</title>
        <language>
```

```
            <code>en</code>
            <text>English</text>
          </language>
          <language>
            <code>zh</code>
            <text>ä¸-å›½ (Chinese)</text>
          </language>
          <language>
            <code>ja</code>
            <text>æ—¥æœ¬ (Japanese)</text>
          </language>
          <language>
            <code>ru</code>
            <text>Ð ÑƒÑÑкий (Russian)</text>
          </language>
          <language>
            <code>ua</code>
            <text>Ð£ÐºÑ?Ð°Ñ—Ð½ÑÑŒÐºÐ° (Ukrainian)</text>
          </language>
        </language-selector>
        <logon-form>
            <title-text l10n="yes"><![CDATA[Login]]></title-text>
            <title-background-color><![CDATA[#666666]]></title-background-color>
            <title-font-color><![CDATA[#ffffff]]></title-font-color>
            <message-text l10n="yes"><![CDATA[Please enter your username and
password.]]></message-text>
            <username-prompt-text l10n="yes"><![CDATA[USERNAME:]]></username-prompt-text>
            <password-prompt-text l10n="yes"><![CDATA[PASSWORD:]]></password-prompt-text>
            <internal-password-prompt-text l10n="yes">Internal
Password:</internal-password-prompt-text>
            <internal-password-first>no</internal-password-first>
            <group-prompt-text l10n="yes"><![CDATA[GROUP:]]></group-prompt-text>
            <submit-button-text l10n="yes"><![CDATA[Login]]></submit-button-text>
            <title-font-color><![CDATA[#ffffff]]></title-font-color>
            <title-background-color><![CDATA[#666666]]></title-background-color>
            <font-color>#000000</font-color>
            <background-color>#ffffff</background-color>
            <border-color>#858A91</border-color>
        </logon-form>
        <logout-form>
            <title-text l10n="yes"><![CDATA[Logout]]></title-text>
            <message-text l10n="yes"><![CDATA[Goodbye.<br>

For your own security, please:<br>

<li>Clear the browser's cache

<li>Delete any downloaded files

<li>Close the browser's window]]></message-text>
            <login-button-text l10n="yes">Logon</login-button-text>
            <hide-login-button>no</hide-login-button>
            <title-background-color><![CDATA[#666666]]></title-background-color>
            <title-font-color><![CDATA[#ffffff]]></title-font-color>
            <title-font-color><![CDATA[#ffffff]]></title-font-color>
            <title-background-color><![CDATA[#666666]]></title-background-color>
            <font-color>#000000</font-color>
            <background-color>#ffffff</background-color>
            <border-color>#858A91</border-color>
        </logout-form>
        <title-panel>
           <mode>enable</mode>
           <text l10n="yes"><![CDATA[SSL VPN Service]]></text>
           <logo-url l10n="yes">/+CSCOU+/csco_logo.gif</logo-url>
```

```
      <gradient>yes</gradient>
      <style></style>
      <background-color><![CDATA[#ffffff]]></background-color>
      <font-size><![CDATA[larger]]></font-size>
      <font-color><![CDATA[#800000]]></font-color>
      <font-weight><![CDATA[bold]]></font-weight>
    </title-panel>
    <info-panel>
      <mode>disable</mode>
      <image-url l10n="yes">/+CSCOU+/clear.gif</image-url>
      <image-position>above</image-position>
      <text l10n="yes"></text>
    </info-panel>
    <copyright-panel>
      <mode>disable</mode>
      <text l10n="yes"></text>
    </copyright-panel>
  </auth-page>
  <portal>
    <title-panel>
      <mode>enable</mode>
      <text l10n="yes"><![CDATA[SSL VPN Service]]></text>
      <logo-url l10n="yes">/+CSCOU+/csco_logo.gif</logo-url>
      <gradient>yes</gradient>
      <style></style>
      <background-color><![CDATA[#ffffff]]></background-color>
      <font-size><![CDATA[larger]]></font-size>
      <font-color><![CDATA[#800000]]></font-color>
      <font-weight><![CDATA[bold]]></font-weight>
    </title-panel>
    <browse-network-title l10n="yes">Browse Entire Network</browse-network-title>
    <access-network-title l10n="yes">Start AnyConnect</access-network-title>
    <application>
      <mode>enable</mode>
      <id>home</id>
      <tab-title l10n="yes">Home</tab-title>
      <order>1</order>
    </application>
    <application>
      <mode>enable</mode>
      <id>web-access</id>
      <tab-title l10n="yes"><![CDATA[Web Applications]]></tab-title>
      <url-list-title l10n="yes"><![CDATA[Web Bookmarks]]></url-list-title>
      <order>2</order>
    </application>
    <application>
      <mode>enable</mode>
      <id>file-access</id>
      <tab-title l10n="yes"><![CDATA[Browse Networks]]></tab-title>
      <url-list-title l10n="yes"><![CDATA[File Folder Bookmarks]]></url-list-title>
      <order>3</order>
    </application>
    <application>
      <mode>enable</mode>
      <id>app-access</id>
      <tab-title l10n="yes"><![CDATA[Application Access]]></tab-title>
      <order>4</order>
    </application>
    <application>
      <mode>enable</mode>
      <id>net-access</id>
      <tab-title l10n="yes">AnyConnect</tab-title>
      <order>4</order>
    </application>
```

```
  <application>
     <mode>enable</mode>
     <id>help</id>
     <tab-title l10n="yes">Help</tab-title>
     <order>1000000</order>
  </application>
  <toolbar>
     <mode>enable</mode>
     <logout-prompt-text l10n="yes">Logout</logout-prompt-text>
     <prompt-box-title l10n="yes">Address</prompt-box-title>
     <browse-button-text l10n="yes">Browse</browse-button-text>
  </toolbar>
  <column>
     <width>100%</width>
     <order>1</order>
  </column>
  <pane>
     <type>TEXT</type>
     <mode>disable</mode>
     <title></title>
     <text></text>
     <notitle></notitle>
     <column></column>
     <row></row>
     <height></height>
  </pane>
  <pane>
     <type>IMAGE</type>
     <mode>disable</mode>
     <title></title>
     <url l10n="yes"></url>
     <notitle></notitle>
     <column></column>
     <row></row>
     <height></height>
  </pane>
  <pane>
     <type>HTML</type>
     <mode>disable</mode>
     <title></title>
     <url l10n="yes"></url>
     <notitle></notitle>
     <column></column>
     <row></row>
     <height></height>
  </pane>
  <pane>
     <type>RSS</type>
     <mode>disable</mode>
     <title></title>
     <url l10n="yes"></url>
     <notitle></notitle>
     <column></column>
     <row></row>
     <height></height>
  </pane>
  <url-lists>
     <mode>group</mode>
  </url-lists>
  <home-page>
     <mode>standard</mode>
     <url></url>
  </home-page>
</portal>
```

```
</custom>
```

Figure 21-4 shows the Logon page and its customizing XML tags. All these tags are nested within the higher-level tag <auth-page>.

Figure 21-4 Logon Page and Associated XML Tags

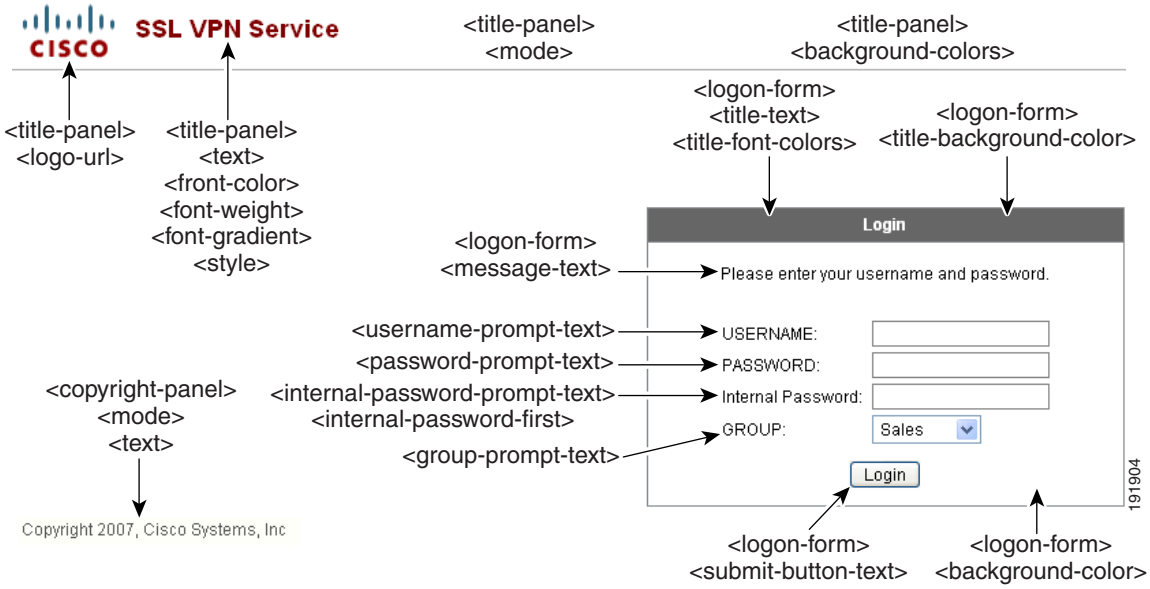


Figure 21-5 shows the Language Selector drop-down list that is available on the Logon page, and the XML tags for customizing this feature. All these tags are nested within the higher-level <auth-page> tag.

Figure 21-5 Language Selector on Logon Screen and Associated XML Tags

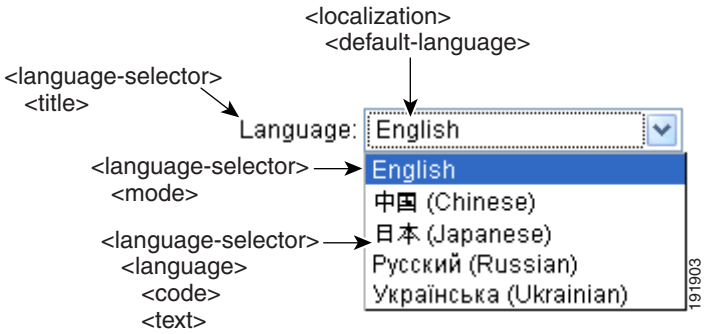


Figure 21-6 shows the Information Panel that is available on the Logon page, and the XML tags for customizing this feature. This information can appear to the left or right of the login box. These tags are nested within the higher-level <auth-page> tag.

Figure 21-6 Information Panel on Logon Screen and Associated XML Tags

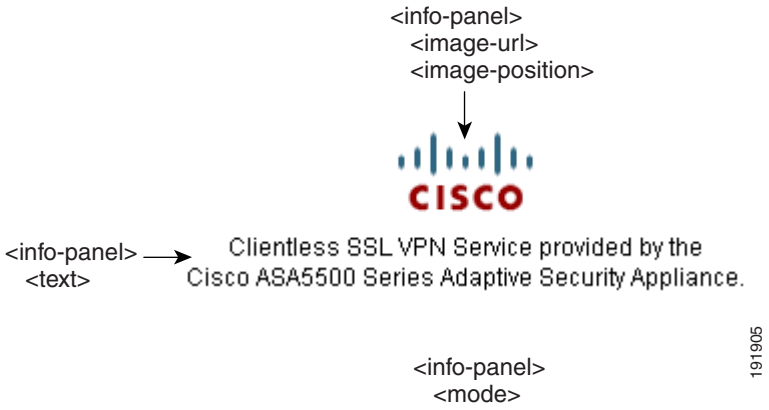
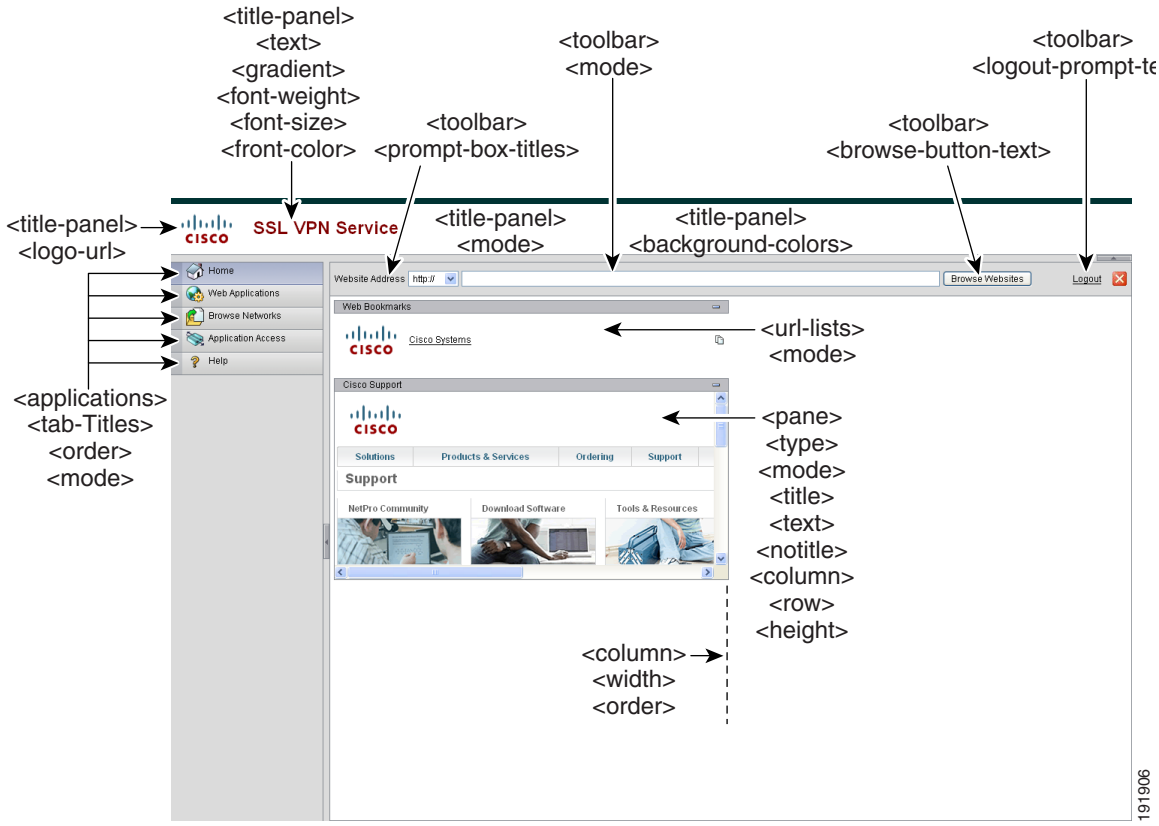


Figure 21-7 shows the Portal page and the XML tags for customizing this feature. These tags are nested within the higher-level <auth-page> tag.

Figure 21-7 Portal Page and Associated XML Tags



Importing a Customization Object

After you edit and save the XML file, import it into cache memory of the ASA using the following commands:

DETAILED STEPS

	Command	Purpose
Step 1	**`import webvpn customization`**	Imports an XML file into cache memory of the ASA. When you import the customization object, the ASA checks the XML code for validity. If the code is valid, the ASA stores the object in a hidden location in cache memory.

	Example:
	`ciscoasa#` **`import webvpn customization custom1 tftp://209.165.201.22/customization /General.xml`**
	`Accessing tftp://209.165.201.22/customization/General.xml...!!`
	`!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!`
	`!!!!!!!!!!!!!!!!!!!!!!!!!!`
	`Writing file disk0:/csco_config/97/custom1...`
	`!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!`
	`!!!!!!!!!!!!!!!!!!!!!!!!!!!!`
	`329994 bytes copied in 5.350 secs (65998 bytes/sec)`
		Imports the customization object *General.xml* from the URL 209.165.201.22/customization and names it *custom1*.

Applying Customizations to Connection Profiles, Group Policies and Users

After you create a customization, you can apply the customization to a connection profile (tunnel group), a group, or a user, with the **customization** command. The options displayed with this command are different depending on the mode you are in.

Note After you customize the portal pages, the changes do not take effect until you reload the ASA, or you disable and then enable clientless SSL.

For more information about configuring connection profiles, group policies, and users, see Chapter 4, "Configuring Connection Profiles, Group Policies, and Users.".

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**tunnel-group webvpn** OR **group-policy webvpn** OR **username webvpn**	Switches to tunnel-group Clientless SSL VPN configuration mode. Switches to group-policy Clientless SSL VPN configuration. Switches to username Clientless SSL VPN configuration.

Step 3	**customization *name***
	
	Example:
	`hostname(config)# tunnel-group cisco_telecommuters webvpn-attributes`
	`hostname(tunnel-group-webvpn)# customization cisco`
	OR
	customization {none | value *name*}
	
	Example:
	`hostname(config)# group-policy cisco_sales attributes`
	`hostname(config-group-policy)# webvpn`
	`hostname(config-username-webvpn)# customization value ?`
	`config-username-webvpn mode commands/options:`
	`Available configured customization profiles:`
	`DfltCustomization`
	`cisco`
	`hostname(config-group-webvpn)# customization value cisco`
	
	Example:
	`hostname(config)# username cisco_employee attributes`
	`hostname(config-username)# webvpn`
	`hostname(config-username-webvpn)# customization value cisco`
		Applies a customization to a connection profile. name is the name of a customization to apply to the connection profile.
		Enters tunnel-group Clientless SSL VPN configuration mode and enables the customization *cisco* for the connection profile *cisco_telecommutes*.
		Applies a customization to a group or use. The following options are included:
		• **none** disables the customization for the group or user, prevents the value from being inherited, and displays the default Clientless SSL VPN pages.
		• **value** *name* is the name of a cu
		Enters group policy Clientless SSL VPN configuration mode, queries the security appliance for a list of customizations, and enables the customization *cisco* for the group policy *cisco_sales*.
		Enters username Clientless SSL VPN configuration mode and enables the customization *cisco* for the user *cisco_employee*.

	Command	Purpose

Step 4	(Optional)
	`[no] customization name`
	OR
	`[no] customization {none | value name}`
		Removes the command from the configuration and removes a customization from the connection profile.
		Removes the command from the configuration and reverts to the default.

Step 5	**`customization`** command followed by a question mark (`?`)	Shows a list of existing customizations.

Login Screen Advanced Customization

If you prefer to use your own, custom login screen, rather than changing specific screen elements of the login screen we provide, you can perform this advanced customization using the *Full Customization* feature.

With Full Customization, you provide the HTML for your own login screen, and you insert Cisco HTML code that calls functions on the ASA that create the Login form and the Language Selector drop-down list.

This section describes the modifications you need to make to your HTML code and the tasks required to configure the ASA to use your code.

Figure 21-8 shows the standard Cisco login screen that displays to Clientless SSL VPN users. The Login form is displayed by a function called by the HTML code.

Figure 21-8 Standard Cisco Login Page

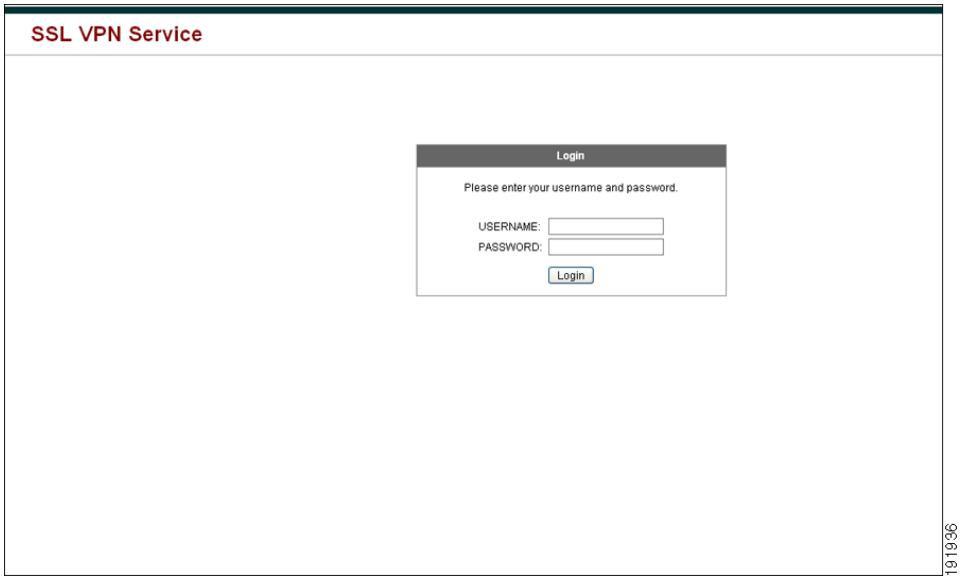


Figure 21-9 shows the Language Selector drop-down list. This feature is an option for Clientless SSL VPN users and is also called by a function in the HTML code of the login screen.

Figure 21-9 Language Selector Drop-down List

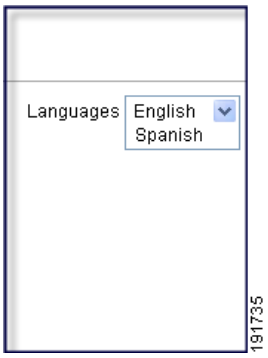
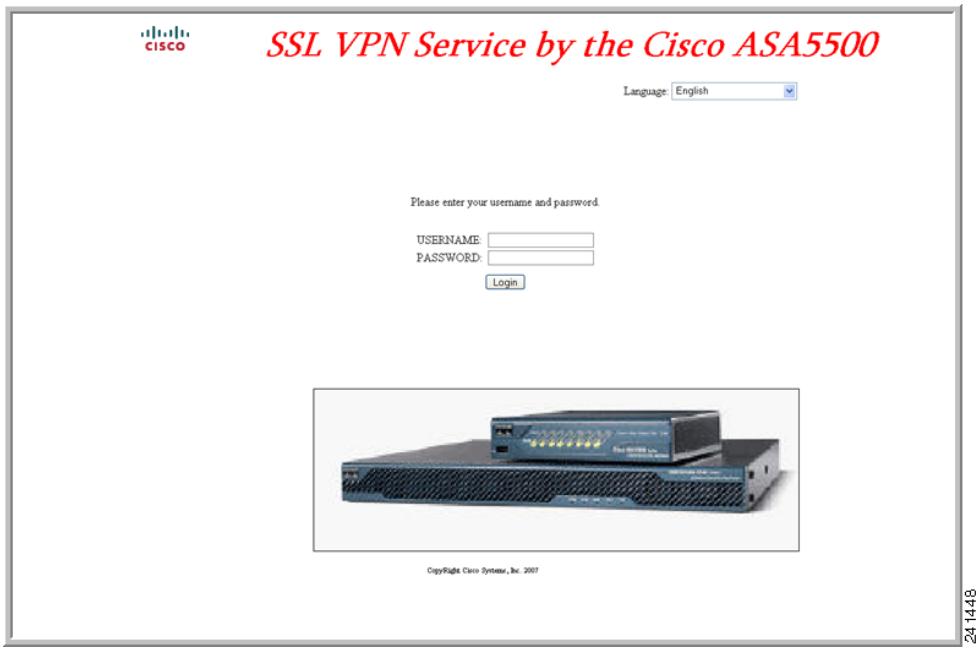


Figure 21-10 shows a simple example of a custom login screen enabled by the Full Customization feature.

Figure 21-10 Example of Full Customization of Login Screens



The following HTML code is used as an example and is the code that displays:

Example:

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>New Page 3</title>
<base target="_self">
</head>

<p align="center">
<img border="0" src="/+CSCOU+/cisco_logo.jpg" width="188" height="48"><font face="Snap
ITC" size="6" color="#FF00FF">
```

```
</font><font face="Snap ITC" color="#FF00FF" size="7"> </font><i><b><font
color="#FF0000" size="7" face="Sylfaen"> SSL VPN Service by the Cisco
ASA5500</font></b></i></p>

    <body onload="csco_ShowLoginForm('lform');csco_ShowLanguageSelector('selector')">

    <table>

    <tr><td colspan=3 height=20 align=right><div id="selector" style="width:
    300px"></div></td></tr>
    <tr><td></td><td></td><td></td></tr>
    <tr>
    <td height="379"></td>
    <td height="379"></td>
    <td align=middle valign=middle>
    <div id=lform >
    <p> </p>
    <p> </p>
    <p> </p>
    <p>Loading...</p>
    </div>
    </td>
    </tr>
    <tr>
    <td width="251"></td>
    <td width="1"></td>
    <td align=right valign=right width="800">
    <img border="1" src="/+CSCOU+/asa5500.jpg" width="660" height="220" align="middle">
    </td></tr>

    </table>
```

The indented code injects the Login form and the Language Selector on the screen. The function **csco_ShowLoginForm('lform')** injects the logon form. **csco_ShowLanguageSelector('selector')** injects the Language Selector.

Modifying Your HTML File

DETAILED STEPS

Step 1 Name your file **logon.inc**. When you import the file, the ASA recognizes this filename as the logon screen.

Step 2 Modify the paths of images used by the file to include **/+CSCOU+/**.

Files that are displayed to remote users before authentication must reside in a specific area of the ASA cache memory represented by the path /+CSCOU+/. Therefore, the source for each image in the file must include this path. For example:

```
src="/+CSCOU+/asa5520.gif"
```

Step 3 Insert the special HTML code below. This code contains the Cisco functions, described earlier, that inject the login form and language selector onto the screen.

```
<body onload="csco_ShowLoginForm('lform');csco_ShowLanguageSelector('selector')">

<table>

<tr><td colspan=3 height=20 align=right><div id="selector" style="width:
300px"></div></td></tr>
<tr><td></td><td></td><td></td></tr>
```

```
<tr>
<td height="379"></td>
<td height="379"></td>
<td align=middle valign=middle>
<div id=lform >
<p> </p>
<p> </p>
<p> </p>
<p>Loading...</p>
</div>
</td>
</tr>
<tr>
<td width="251"></td>
<td width="1"></td>
<td align=right valign=right width="800">
<img border="1" src="/+CSCOU+/asa5500.jpg" width="660" height="220" align="middle">
</td></tr>

</table>
```

-

Customizing Bookmark Help

The ASA displays help content on the application panels for each selected bookmark. You can customize those help files or create help files in other languages. You then import them to flash memory for display during subsequent sessions. You can also retrieve previously imported help content files, modify them, and reimport them to flash memory.

Each application panel displays its own help file content using a predetermined filename. The prospective location of each is in the /+CSCOE+/help/*language*/ URL within flash memory of the ASA. Table 21-1 shows the details about each of the help files you can maintain for VPN sessions.

Table 21-1 VPN Application Help Files

Application Type	Panel	URL of Help File in Flash Memory of the Security Appliance	Help File Provided By Cisco in English?
Standard	Application Access	/+CSCOE+/help/*language*/app-access-hlp.inc	Yes
Standard	Browse Networks	/+CSCOE+/help/*language*/file-access-hlp.inc	Yes
Standard	AnyConnect Client	/+CSCOE+/help/*language*/net-access-hlp.inc	Yes
Standard	Web Access	/+CSCOE+/help/*language*/web-access-hlp.inc	Yes
Plug-in	MetaFrame Access	/+CSCOE+/help/*language*/ica-hlp.inc	No
Plug-in	Terminal Servers	/+CSCOE+/help/*language*/rdp-hlp.inc	Yes
Plug-in	Telnet/SSH Servers	/+CSCOE+/help/*language*/ssh,telnet-hlp.inc	Yes
Plug-in	VNC Connections	/+CSCOE+/help/*language*/vnc-hlp.inc	Yes

language is the abbreviation of the language rendered by the browser. This field is *not* used for file translation; it indicates the language used in the file. To specify a particular language code, copy the language abbreviation from the list of languages rendered by your browser. For example, a dialog window displays the languages and associated language codes when you use one of the following procedures:

- Open Internet Explorer and choose **Tools > Internet Options > Languages > Add**.
- Open Mozilla Firefox and choose **Tools > Options > Advanced > General**, click **Choose** next to Languages, and click **Select a language to add**.

The following sections describe how to customize the help contents:

- Customizing a Help File Provided By Cisco, page 21-17
- Creating Help Files for Languages Not Provided by Cisco, page 21-18
- Importing a Help File to Flash Memory, page 21-18
- Exporting a Previously Imported Help File from Flash Memory, page 21-19

Customizing a Help File Provided By Cisco

To customize a help file provided by Cisco, you need to get a copy of the file from the flash memory card first. Get the copy and customize it as follows:

DETAILED STEPS

Step 1 Use your browser to establish a Clientless SSL VPN session with the ASA.

Step 2 Display the help file by appending the string in "URL of Help File in Flash Memory of the Security Appliance" in Table 21-1, to the address of the ASA, then press Enter.

Note Enter **en** in place of *language* to get the help file in English.

The following example address displays the English version of the Terminal Servers help:

https://*address_of_security_appliance***/+CSCOE+/help/en/rdp-hlp.inc**

Step 3 Choose **File > Save (Page) As**.

Note Do not change the contents of the File name box.

Step 4 Change the Save as type option to **Web Page, HTML only** and click **Save**.

Step 5 Use your preferred HTML editor to modify the file.

Note You can use most HTML tags, but do *not* use tags that define the document and its structure (e.g., do not use <html>, <title>, <body>, <head>, <h1>, <h2>, etc. You can use character tags, such as the <b> tag, and the <p>, <ol>, <ul>, and <li> tags to structure content.

Step 6 Save the file as HTML only, using the original filename and extension.

Step 7 Ensure the filename matches the one in Table 21-1, and that it does not have an extra filename extension.

Creating Help Files for Languages Not Provided by Cisco

Use HTML to create help files in other languages.

We recommend creating a separate folder for each language to support.

Save the file as HTML only. Use the filename following the last slash in "URL of Help File in Flash Memory of the Security Appliance" in Table 21-1.

See the next section to import the files for display during VPN sessions.

Restrictions

You can use most HTML tags, but do *not* use tags that define the document and its structure (e.g., do not use <html>, <title>, <body>, <head>, <h1>, <h2>, etc. You can use character tags, such as the <b> tag, and the <p>, <ol>, <ul>, and <li> tags to structure content.

Importing a Help File to Flash Memory

DETAILED STEPS

	Command	Purpose
Step 1	**import webvpn webcontent *destination_url source_url***	Imports a help content file to flash memory for display in Clientless SSL VPN sessions. • destination_url is the string in the URL of Help File in Flash Memory of the Security Appliance column of Table 21-1VPN Application Help Files. • *source_url* is the URL of the file to import. Valid prefixes are ftp://, http://, and tftp://.
	Example: hostname# **import webvpn webcontent /+CSCOE+/help/en/app-access-hlp.inc tftp://209.165.200.225/app-access-hlp.inc**	Copies the help file *app-access-hlp.inc* to flash memory from the TFTP server at 209.165.200.225. The URL includes the abbreviation *en* for the English language.

Exporting a Previously Imported Help File from Flash Memory

DETAILED STEPS

	Command	Purpose
Step 1	**export webvpn webcontent** *source_url destination_url*	Retrieves a previously imported help content file for subsequent edits. • *source_url* is the string in "URL of Help File in Flash Memory of the Security Appliance" in Table 21-1. • *destination_url* is **the target URL**. Valid prefixes are ftp:// and tftp://. The maximum number of characters is 255.
	Example: hostname# **export webvpn webcontent /+CSCOE+/help/en/file-access-hlp.inc tftp://209.165.200.225/file-access-hlp.inc**	Copies the English language help file *file-access-hlp.inc* displayed on the Browser Networks panel to TFTP Server 209.165.200.225.

Translating the Language of User Messages

The ASA provides language translation for the entire Clientless SSL VPN session. This includes login, logout banners, and portal pages displayed after authentication such as plugins and AnyConnect.

This section describes how to configure the ASA to translate these user messages and includes the following sections:

- Understanding Language Translation, page 21-19
- Creating Translation Tables, page 21-20
- Referencing the Language in a Customization Object, page 21-22
- Changing a Group Policy or User Attributes to Use the Customization Object, page 21-24

Understanding Language Translation

Functional areas and their messages that are visible to remote users are organized into translation domains. Table 21-2 shows the translation domains and the functional areas translated.

Table 21-2 Language Translation Domain Options

Translation Domain	Functional Areas Translated
AnyConnect	Messages displayed on the user interface of the Cisco AnyConnect VPN client.
banners	Message displayed when VPN access is denied for a clientless connection.
CSD	Messages for the Cisco Secure Desktop (CSD).

Translation Domain	Functional Areas Translated
customization	Messages on the logon and logout pages, portal page, and all the messages customizable by the user.
plugin-ica	Messages for the Citrix plug-in.
plugin-rdp	Messages for the Remote Desktop Protocol plug-in.
plugin-rdp2	Messages for the Java Remote Desktop Protocol plug-in.
plugin-telnet,ssh	Messages for the Telnet and SSH plug-in.
plugin-vnc	Messages for the VNC plug-in.
PortForwarder	Messages displayed to Port Forwarding users.
url-list	Text that user specifies for URL bookmarks on the portal page.
webvpn	All the layer 7, AAA and portal messages that are not customizable.

The ASA includes a translation table template for each domain that is part of standard functionality. The templates for plug-ins are included with the plug-ins and define their own translation domains.

You can export the template for a translation domain, which creates an XML file of the template at the URL you provide. The message fields in this file are empty. You can edit the messages and import the template to create a new translation table object that resides in flash memory.

You can also export an existing translation table. The XML file created displays the messages you edited previously. Reimporting this XML file with the same language name creates a new version of the translation table object, overwriting previous messages.

Some templates are static, but some change based on the configuration of the ASA. Because you can customize the *logon and logout pages, portal page, and URL bookmarks for clientless users,* the **ASA generates the customization** and **url-list** translation domain templates dynamically, and the template automatically reflects your changes to these functional areas.

After creating translation tables, they are available to customization objects that you create and apply to group policies or user attributes. With the exception of the AnyConnect translation domain, a translation table has no affect, and messages are not translated on user screens until you create a customization object, identify a translation table to use in that object, and specify that customization for the group policy or user. Changes to the translation table for the AnyConnect domain are immediately visible to AnyConnect client users.

Creating Translation Tables

You can create translation tables in both single context mode and multi-context mode:

DETAILED STEPS

	Command	Purpose
Step 1	**export webvpn translation-table** **Example:** hostname# **show import webvpn translation-table** Translation Tables' Templates: customization AnyConnect CSD PortForwarder url-list webvpn Citrix-plugin RPC-plugin Telnet-SSH-plugin VNC-plugin Translation Tables: **Example:** hostname# **export webvpn translation-table customization template tftp://209.165.200.225/portal**	Exports a translation table template to a computer. Shows available translation table templates and tables. Exports the translation table template for the customization domain, which affects messages displayed for users in Clientless SSL VPN sessions. The filename of the XML file created is *portal* (user-specified) and contains empty message fields.

	Command	Purpose

Step 2	Edit the translation table XML file
	
	Example:
	`# Copyright (C) 2006 by Cisco Systems, Inc.`
	`#`
	`#, fuzzy`
	`msgid ""`
	`msgstr ""`
	`"Project-Id-Version: ASA\n"`
	`"Report-Msgid-Bugs-To: vkamyshe@cisco.com\n"`
	`"POT-Creation-Date: 2007-03-12 18:57 GMT\n"`
	`"PO-Revision-Date: YEAR-MO-DA HO:MI+ZONE\n"`
	`"Last-Translator: FULL NAME <EMAIL@ADDRESS>\n"`
	`"Language-Team: LANGUAGE <LL@li.org>\n"`
	`"MIME-Version: 1.0\n"`
	`"Content-Type: text/plain; charset=UTF-8\n"`
	`"Content-Transfer-Encoding: 8bit\n"`
	
	`#: DfltCustomization:24 DfltCustomization:64`
	`msgid "Clientless SSL VPN Service"`
	`msgstr ""`
		Shows a portion of the template that was exported as *portal*. The end of this output includes a message ID field (msgid) and a message string field (msgstr) for the message which is displayed on the portal page when a user establishes a Clientless SSL VPN session. The complete template contains many pairs of message fields.

Step 3	**`import webvpn translation-table`**
	
	Example:
	`hostname# import webvpn translation-table customization language es-us tftp://209.165.200.225/portal`
	`hostname# !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!`
	`hostname# show import webvpn translation-table`
	`Translation Tables' Templates:`
	`AnyConnect`
	`PortForwarder`
	`csd`
	`customization`
	`keepout`
	`url-list`
	`webvpn`
	`Citrix-plugin`
	`RPC-plugin`
	`Telnet-SSH-plugin`
	`VNC-plugin`
	
	`Translation Tables:`
	`es-us customization`
		Imports the translation table.
		
		Import the XML file. *es-us* is the abbreviation for Spanish spoken in the United States.

If you import a translation table for the AnyConnect domain, your changes are effective immediately. If you import a translation table for any other domain, you must create a customization object, identify the translation table to use in that object, and specify that customization object for the group policy or user.

Referencing the Language in a Customization Object

This section describes how to export the customization template, edit it, and import it as a customization object so that you can refer to it.

Prerequisites

For the customization object to call these translation tables correctly, the tables must have been previously imported using the same names. These names must be compatible with language options of the browser.

DETAILED STEPS

	Command	Function
Step 1	**export webvpn customization template** **Example:** hostname# **export webvpn customization template tftp://209.165.200.225/sales**	Exports a customization template to a URL where you can edit it. Exports the template and creates the copy *sales* at the URL specified.
Step 2	Edit the customization template and reference the previously-imported translation table **Example:** <localization> <languages>en,ja,zh,ru,ua</languages> <default-language>en</default-language> </localization> **Example:** <auth-page> .... <language-selector> <mode>enable</mode> <title l10n="yes">Language:</title> <language> <code>en</code> <text>English</text> </language> <language> <code>es-us</code> <text>Spanish</text> </language> </language-selector>	Two areas of XML code in the customization template pertain to translation tables. Specifies the translation table to use. • The <languages> tag in the XML code is followed by the names of the translation tables. In this example, they are en, ja, zh, ru, and ua. • The <default-language> tag specifies the language that the remote user first encounters when connecting to the ASA. In the example code above, the language is English. Affects the display of the Language Selector and includes the <language selector> tag and the associated <language> tags that enable and customize the Language Selector: • The <language-selector> group of tags includes the <mode> tag that enables and disables the displaying of the Language Selector and the <title> tag that specifies the title of the drop-down box listing the languages. • The <language> group of tags includes the <code> and <text> tags that map the language name displayed in the Language Selector drop-down box to a specific translation table.
Step 3	Save the file after making your changes.	

	Command	Function
Step 4	**import webvpn customization** **Example:** hostname# **import webvpn customization sales tftp://209.165.200.225/sales** hostname# !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!	Imports the customization template as a new object.
Step 5	**show import webvpn customization** **Example:** hostname# **import webvpn customization sales tftp://209.165.200.225/sales** hostname# !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!	Shows the new customization object *sales*.

Changing a Group Policy or User Attributes to Use the Customization Object

This section describes how to activate your changes for specific groups or users.

DETAILED STEPS

	Command	Purpose
Step 1	**webvpn**	Switches to Clientless SSL VPN configuration mode.
Step 2	**group-policy webvpn**	Switches to group-policy Clientless SSL VPN configuration mode.
Step 3	**customization** **Example:** hostname(config)# **group-policy sales attributes** hostname(config-group-policy)# **webvpn** hostname(config-group-webvpn)# **customization value sales**	Enables the customization object. Shows the customization object *sales* enabled in the group policy *sales*.

Clientless SSL VPN Troubleshooting

September 13, 2013

Closing Application Access to Prevent hosts File Errors

To prevent hosts file errors that can interfere with Application Access, close the Application Access window properly when you finish using Application Access. To do so, click the close icon.

Recovering from Hosts File Errors When Using Application Access

The following errors can occur if you do not close the Application Access window properly:

- The next time you try to start Application Access, it may be switched off; you receive a `Backup HOSTS File Found` error message.
- The applications themselves may be switched off or malfunction, even when you are running them locally.

These errors can result from terminating the Application Access window in any improper way. For example:

- Your browser crashes while you are using Application Access.
- A power outage or system shutdown occurs while you are using Application Access.
- You minimize the Application Access window while you are working, then shut down your computer with the window active (but minimized).

This section includes the following topics:

Understanding the hosts File

The hosts file on your local system maps IP addresses to hostnames. When you start Application Access, Clientless SSL VPN modifies the hosts file, adding Clientless SSL VPN-specific entries. Stopping Application Access by properly closing the Application Access window returns the file to its original state.

Before invoking Application Access...	hosts file is in original state.
When Application Access starts....	• Clientless SSL VPN copies the hosts file to hosts.webvpn, thus creating a backup. • Clientless SSL VPN then edits the hosts file, inserting Clientless SSL VPN-specific information.
When Application Access stops...	• Clientless SSL VPN copies the backup file to the `hosts` file, thus restoring the hosts file to its original state. • Clientless SSL VPN deletes hosts.webvpn.
After finishing Application Access...	hosts file is in original state.

Note Microsoft anti-spyware software blocks changes that the port forwarding Java applet makes to the hosts file. See www.microsoft.com for information on how to allow hosts file changes when using anti-spyware software.

Stopping Application Access Improperly

When Application Access terminates abnormally, the `hosts` file remains in a Clientless SSL VPN-customized state. Clientless SSL VPN checks the state the next time you start Application Access by searching for a hosts.webvpn file. If it finds one, a `Backup HOSTS File Found` error message appears, and Application Access is temporarily switched off.

Once you shut down Application Access improperly, you leave your remote access client/server applications in limbo. If you try to start these applications without using Clientless SSL VPN, they may malfunction. You may find that hosts that you normally connect to are unavailable. This situation could commonly occur if you run applications remotely from home, fail to quit the Application Access window before shutting down the computer, then try to run the applications later from the office.

Reconfiguring a Host's File Automatically Using Clientless SSL VPN

If you are able to connect to your remote access server, follow these steps to reconfigure the host's file and re-enable both Application Access and the applications.

DETAILED STEPS

Step 1 Start Clientless SSL VPN and log in. The home page opens.

Step 2 Click the **Applications Access** link. A `Backup HOSTS File Found` message appears.

Step 3 Choose one of the following options:

- **Restore from backup**—Clientless SSL VPN forces a proper shutdown. It copies the hosts.webvpn backup file to the `hosts` file, restoring it to its original state, then deletes hosts.webvpn. You then have to restart Application Access.
- **Do nothing**—Application Access does not start. The remote access home page reappears.
- **Delete backup**—Clientless SSL VPN deletes the hosts.webvpn file, leaving the hosts file in its Clientless SSL VPN-customized state. The original `hosts` file settings are lost. Application Access then starts, using the Clientless SSL VPN-customized hosts file as the new original. Choose this option only if you are unconcerned about losing hosts file settings. If you or a program you use may have edited the hosts file after Application Access has shut down improperly, choose one of the other options, or edit the hosts file manually. (See "Reconfiguring hosts File Manually.")

Reconfiguring hosts File Manually

If you are not able to connect to your remote access server from your current location, or if you have customized the hosts file and do not want to lose your edits, follow these steps to reconfigure the hosts file and reenable both Application Access and the applications.

DETAILED STEPS

Step 1 Locate and edit your hosts file. The most common location is c:\windows\sysem32\drivers\etc\hosts.

Step 2 Check to see if any lines contain the string: `# added by WebVpnPortForward`
If any lines contain this string, your hosts file is Clientless SSL VPN-customized. If your hosts file is Clientless SSL VPN-customized, it looks similar to the following example:

```
server1 # added by WebVpnPortForward
server1.example.com invalid.cisco.com # added by WebVpnPortForward
server2 # added by WebVpnPortForward
server2.example.com invalid.cisco.com # added by WebVpnPortForward
server3 # added by WebVpnPortForward
server3.example.com invalid.cisco.com # added by WebVpnPortForward

# Copyright (c) 1993-1999 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to hostnames. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding hostname.
# The IP address and the hostname should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     cisco.example.com          # source server
#       38.25.63.10     x.example.com              # x client host

123.0.0.1       localhost
```

Step 3 Delete the lines that contain the string: `# added by WebVpnPortForward`

Step 4 Save and close the file.

Step 5 Start Clientless SSL VPN and log in.

The home page appears.

Step 6 Click the **Application Access** link.

The Application Access window appears. Application Access is now enabled.

Capturing Data

The CLI **capture** command lets you log information about websites that do not display properly over a Clientless SSL VPN session. This data can help your Cisco customer support engineer troubleshoot problems. The following sections describe how to capture and view Clientless SSL VPN session data:

- Creating a Capture File, page 22-4
- Using a Browser to Display Capture Data, page 22-5

Prerequisites

- Enabling Clientless SSL VPN capture affects the performance of the security appliance. Ensure you switch off the capture after you generate the capture files needed for troubleshooting.

Creating a Capture File

DETAILED STEPS

	Command	Purpose

Step 1	**capture *capture_name* type webvpn user *webvpn_username***
	
	Example:
	`hostname# capture hr type webvpn user user2`
	`WebVPN capture started.`
	`  capture name  hr`
	`  user name  user2`
	`hostname# no capture hr`
		Starts the capture utility for Clientless SSL VPN.
		• *capture_name* is a name you assign to the capture, which is also prepended to the name of the capture files.
		• *webvpn_user* is the username to match for capture.
		Creates a capture named hr, which captures traffic for user2 to a file.

Step 2	(Optional) **no capture capture_name**	Stops the capture utility from capturing packets after a user has logged in and began a Clientless SSL VPN session. The capture utility creates a *capture_name*.zip file, which is encrypted with the password **koleso**.
Step 3	Send the .zip file to Cisco Systems or attach it to a Cisco TAC service request.	
Step 4	Unzip the contents of the file using the *koleso* password.	

Using a Browser to Display Capture Data

DETAILED STEPS

	Command	Purpose
Step 1	**capture** ***capture_name*** **type webvpn user** ***webvpn_username***	Starts the capture utility for Clientless SSL VPN. • *capture_name* is a name you assign to the capture, which is also prepended to the name of the capture files. • *webvpn_user* is the username to match for capture.
Step 2	(Optional) **no capture capture_name**	Stops the capture utility from capturing packets after a user has logged in and began a Clientless SSL VPN session.
Step 3	Open a browser and enter the following: **https://***asdm_enabled_interface_of_the_security_appliance***:***port***/admin/capture/***capture_name***/pcap** **Example:** **https://192.0.2.1:60000/admin/capture/hr/pcap**	Displays the capture named hr in a sniffer format.
Step 4	Repeat Step 2.	

CHAPTER 23

Clientless SSL VPN Licensing

September 13, 2013

Licensing

Note This feature is not available on No Payload Encryption models.

Model	License Requirement[1,2]
ASA 5505	AnyConnect Premium license: • Base License or Security Plus license: 2 sessions. • *Optional permanent or time-based licenses: 10 or 25 sessions.* • *Shared licenses are not supported.*[3]
ASA 5510	AnyConnect Premium license: • Base and Security Plus License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* • *Optional Shared licenses*[3]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5520	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* • *Optional Shared licenses*[3]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5540	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* • *Optional Shared licenses*[3]*: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*

Model	License Requirement[1,2]
ASA 5550	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5580	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5512-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5515-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, or 250 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5525-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, or 750 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5545-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, or 2500 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5555-X	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*

Model	License Requirement[1,2]
ASA 5585-X with SSP-10	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, or 5000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA 5585-X with SSP-20, -40, and -60	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*
ASA SM	AnyConnect Premium license: • Base License: 2 sessions. • *Optional permanent or time-based licenses: 10, 25, 50, 100, 250, 500, 750, 1000, 2500, 5000, or 10000 sessions.* • *Optional Shared licenses[3]: Participant or Server. For the Server license, 500-50,000 in increments of 500 and 50,000-545,000 in increments of 1000.*

1. If you start a clientless SSL VPN session and then start an AnyConnect client session from the portal, 1 session is used in total. However, if you start the AnyConnect client first (from a standalone client, for example) and then log into the clientless SSL VPN portal, then 2 sessions are used.
2. The maximum combined VPN sessions of *all* types cannot exceed the maximum sessions shown in this table.
3. A shared license lets the security appliance act as a shared license server for multiple client security appliances. The shared license pool is large, but the maximum number of sessions used by each individual security appliance cannot exceed the maximum number listed for permanent licenses.

INDEX

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X

Z