Configuring AnyConnect VPN Client Connections

The Cisco AnyConnect SSL VPN Client provides secure SSL connections to the security appliance for remote users. Without a previously-installed client, remote users enter the IP address in their browser of an interface configured to accept SSL VPN connections. Unless the security appliance is configured to redirect http:// requests to http**s**://, 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 security appliance 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 connection and either remains or uninstalls itself (depending on the security appliance configuration) when the connection terminates.

In the case of a previously installed client, when the user authenticates, the security appliance examines the revision of the client, and upgrades the client as necessary.

When the client negotiates an SSL VPN connection with the security appliance, 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 security appliance, 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 security appliance downloads the client based on the group policy or username attributes of the user establishing the connection. You can configure the security appliance 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 security appliance to either download the client after a timeout period or present the login page.

This section covers the following topics:

- Feature History for AnyConnect Connections, page 40-17

Installing the AnyConnect SSL VPN Client

This section presents the platform requirements and the procedure for installing the AnyConnect client on the security appliance and preparing it for download to remote users.

Remote PC System Requirements

The AnyConnect client supports the following operating systems on the remote PC:

- Microsoft Vista
- Microsoft Windows 2000
- Microsoft Windows XP
- MAC Intel
- MAC Power PC
- Linux

The legacy SSL VPN Client (SVC) supports the following operating systems on the remote PC:

- Microsoft Windows 2000
- Microsoft Windows XP

Remote HTTPS Certificates Limitation

The security appliance does not verify remote HTTPS certificates.

Installing the AnyConnect Client

Installing the client on the security appliance consists of copying a client image to the security appliance and identifying the file as a client image. With multiple clients, you must also assign the order that the security appliance downloads the clients to the remote PC. Perform the following steps to install the client:

Step 1 Copy the client image package to the security appliance using the **copy** command from privileged EXEC mode, or using another method. In this example, the images are copied from a tftp server using the **copy tftp** command:

```
hostname# copy tftp flash
Address or name of remote host []? 209.165.200.226
Source filename []? anyconnect-win-2.2.0128-k9.pkg
Destination filename []? sslclient-win-2.2.0128.pkg
Accessingtftp://209.165.200.226/anyconnect-win-2.2.0128-k9.pkg...!!!!!!!!!!!!!!!!!!!!!!!!
Writing file
disk0:/cdisk71...!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
319662 bytes copied in 3.695 secs (86511 bytes/sec)
```

Step 2   Identify a file on flash as an SSL VPN client package file using the **svc image** command from webvpn configuration mode:

svc image *filename order*

The security appliance 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 security appliance 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. For example:

```
hostname(config-webvpn)# svc image anyconnect-win-2.2.0128-k9.pkg 1
hostname(config-webvpn)# svc image anyconnect-macosx-i386-2.2.0128-k9.pkg 2
hostname(config-webvpn)# svc image anyconnect-linux-2.2.0128-k9.pkg 3
```

Note   The security appliance expands SSL VPN client and the CSD images in cache memory. If you receive the error message *ERROR: Unable to load SVC image - increase disk space via the 'cache-fs' command*, use the **cache-fs limit** command to adjust the size of cache memory:

Step 3   Enter the **svc enable** command. Without issuing this command, AnyConnect does not function as expected, and **show webvpn svc** states that the "SSL VPN client is not enabled," instead of listing the installed AnyConnect packages.

Step 4   Check the status of the clients using the **show webvpn svc** command:

```
hostname(config-webvpn)# show webvpn svc
1. disk0:/anyconnect-win-2.2.0128-k9.pkg 1
  CISCO STC win2k+
  2,0,0310
  Tue 03/27/2007  4:16:21.09

2. disk0:/anyconnect-macosx-i386-2.2.0128-k9.pkg 2
  CISCO STC Darwin_i386
  2,0,0
  Tue Mar 27 05:09:16 MDT 2007

3. disk0:/anyconnect-linux-2.2.0128-k9.pkg 3
  CISCO STC Linux
  2,0,0
  Tue Mar 27 04:06:53 MST 2007

3 SSL VPN Client(s) installed
```

Enabling AnyConnect Client Connections

After installing the client, enable the security appliance to allow SSL VPN client connections by performing the following steps:

Step 1   Enable clientless connections on an interface using the **enable** command from webvpn mode:

enable *interface*

For example:

```
hostname(config)# webvpn
hostname(config-webvpn)# enable outside
```

Step 2 Configure a method of address assignment. You can use DHCP, and/or user-assigned addressing. You can also create a local IP address pool using the **ip local pool** command from global configuration mode:

> **ip local pool** *poolname startaddr-endaddr* **mask** *mask*

The following example creates the local IP address pool *vpn_users*:

```
hostname(config)# ip local pool vpn_users 209.165.200.225–209.165.200.254
mask 255.255.255.224
```

Step 3 Assign IP addresses to a tunnel group. One method you can use to do this is to assign a local IP address pool with the **address-pool** command from general-attributes mode:

> **address-pool** *poolname*

To do this, first enter the **tunnel-group** *name* **general-attributes** command to enter general-attributes mode. Then specify the local IP address pool using the **address-pool** command.

In the following example, the user configures the existing tunnel group *telecommuters* to use the address pool *vpn_users created in step 3:*

```
hostname(config)# tunnel-group telecommuters general-attributes
hostname(config-tunnel-general)# address-pool vpn_users
```

Step 4 Assign a default group policy to the tunnel group with the **default-group-policy** command from tunnel group general attributes mode:

> **default-group-policy** *name*

In the following example, the user assigns the group policy *sales* to the tunnel group *telecommuters*:

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

Step 5 Create and enable a group alias that displays in the group list on the WebVPN Login page using the **group-alias** command from tunnel group webvpn attributes mode:

> **group-alias** *name* **enable**

First exit to global configuration mode, and then enter the **tunnel-group** *name* **webvpn-attributes** command to enter tunnel group webvpn attributes mode.

In the following example, the user enters webvpn attributes configuration mode for the tunnel group *telecommuters*, and creates the group alias *sales_department*:

```
hostname(config)# tunnel-group telecommuters webvpn-attributes
hostname(config-tunnel-webvpn)# group-alias sales_department enable
```

Step 6 Enable the display of the tunnel-group list on the WebVPN Login page from webvpn mode:

> **tunnel-group-list enable**

First exit to global configuration mode, and then enter webvpn mode.

In the following example, the user enters webvpn mode, and then enables the tunnel group list:

```
hostname(config)# webvpn
hostname(config-webvpn)# tunnel-group-list enable
```

Step 7 Specify SSL as a permitted VPN tunneling protocol for the group or user with the **vpn-tunnel-protocol svc** command in group-policy mode or username mode. You can also specify additional protocols. For more information, see the vpn-tunnel-protocol command in the *Cisco ASA 5500 Series Command Reference*.

> **vpn-tunnel-protocol svc**

To do this, first exit to global configuration mode, enter the **group-policy** *name* **attributes** command to enter group-policy mode, or the **username** *name* **attributes** command to enter username mode, and then enter the **webvpn** command to enter webvpn mode and change the WebVPN settings for the group or user.

The following example identifies SSL as the only permitted tunneling protocol for the group-policy *sales*:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# vpn-tunnel-protocol svc
```

For more information about assigning users to group policies, see Chapter 32, "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 **svc keep-installer** command from group-policy or username webvpn modes:

svc keep-installer installed

The default is that permanent installation of the client is disabled. The client on the remote computer uninstalls at the end of every session. The following example configures the existing group-policy *sales* to keep the client installed on the remote computer:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-policy)# svc keep-installer installed
```

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 40-13

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 **svc dtls enable** command in group policy webvpn or username webvpn configuration mode:

> [**no**] **svc dtls enable**

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 svc dtls enable
```

Prompting Remote Users

You can enable the security appliance to prompt remote SSL VPN client users to download the client with the **svc ask** command from group policy webvpn or username webvpn configuration modes:

> [**no**] **svc ask** {**none** | **enable** [**default** {**webvpn** | **svc**} **timeout** *value*]}

svc ask enable prompts the remote user to download the client or go to the clientless portal page and waits indefinitely for user response.

**svc ask enable default svc** immediately downloads the client.

svc ask enable default webvpn immediately goes to the portal page.

svc ask enable default svc 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.

svc 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 40-1 shows the prompt displayed to remote users when either **default svc timeout** *value or* **default webvpn timeout** *value is configured:*

Figure 40-1   Prompt Displayed to Remote Users for SSL VPN Client Download
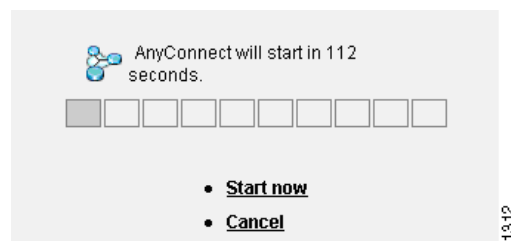
The following example configures the security appliance to prompt the user to download the client or go to the clientless portal page and wait *10 seconds for a response* before downloading the client:

```
hostname(config-group-webvpn)# svc ask enable default svc timeout 10
```

Enabling AnyConnect Client Profile Downloads

An AnyConnect client profile is a group of configuration parameters, stored in an XML file, that the client uses to configure the connection entries that appear in the client user interface. These parameters (XML tags) include the names and addresses of host computers and settings to enable additional client features.

The AnyConnect client installation includes a profile template, named *AnyConnectProfile.tmpl*, that you can edit with a text editor and use as a basis to create other profile files. You can also set advanced parameters that are not available through the user interface. The installation also includes a complete XML schema file, named *AnyConnectProfile.xsd*.

After creating a profile, you must load the file on the security appliance and configure the security appliance to download it to remote client PCs.

Follow these steps to edit a profile and enable the security appliance to download it to remote clients:

Step 1 Retrieve a copy of the profile file (AnyConnectProfile.tmpl) from a client installation. Table 40-1 shows the installation path for each operating system.

Table 40-1   Operating System and Profile File Installation Path

Operating System	Installation Path
Windows Vista	%ALLUSERSPROFILE%\Cisco\Cisco AnyConnect VPN Client\Profile[1]
Windows XP and 2000	%ALLUSERSPROFILE%/Application Data/Cisco/Cisco AnyConnect VPN Client/Profile[2]
Linux	/opt/cisco/vpn/profile
Mac OS X	/opt/cisco/vpn/profile

1. %ALLUSERSPROFILE% refers to the environmental variable by the same name for Windows Vista. In most installations, this is C:\Program Files.

2. %PROGRAMFILES% refers to the environmental variable by the same name for Windows XP and 2000. In most installations, this is C:\Program Files.

Step 2 Edit the profile file. The example below shows the contents of the profile file (AnyConnectProfile.tmpl) for Windows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
    This is a template file that can be configured to support the
    identification of secure hosts in your network.

    The file needs to be renamed to cvcprofile.xml (for now).

    There is an ASA command to import updated profiles for downloading to
    client machines.  Provide some basic instruction.....
-->
<Configuration>
    <ClientInitialization>
        <UseStartBeforeLogon>false</UseStartBeforeLogon>
    </ClientInitialization>
    <HostProfile>
        <HostName></HostName>
        <HostAddress></HostAddress>
    </HostProfile>
    <HostProfile>
```

```
          <HostName></HostName>
          <HostAddress></HostAddress>
      </HostProfile>
</Configuration>
```

The <HostProfile> tags are frequently edited so that the AnyConnect client displays the names and addresses of host computers for remote users. The following example shows the <HostName> and <HostAddress> tags, with the name and address of a host computer inserted:

```
      <HostProfile>
          <HostName>Sales_gateway</HostName>
          <HostAddress>209.165.200.225</HostAddress>
      </HostProfile>
```

Step 3   Load the profile file into flash memory on the security appliance and then use the **svc profiles** command from webvpn configuration mode to identify the file as a client profile to load into cache memory:

[no] svc profiles *name path}*

After the file is loaded into cache memory, the profile is available to group policies and username attributes of client users.

In the following example, the user previously created two new profile files (sales_hosts.xml and engineering_hosts.xml) from the AnyConnectProfile.tmpl file provided in the client installation and uploaded them to flash memory. Then the user specifies these files as profiles for use by group policies, specifying the names *sales* and *engineering*:

```
asa1(config-webvpn)# svc profiles sales disk0:/sales_hosts.xml
asa1(config-webvpn)# svc profiles engineering disk0:/engineering_hosts.xml
```

Entering the **dir cache:stc/profiles** command shows the profiles loaded into cache memory:

```
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 or username attributes webvpn configuration mode and specify a profile for the group or user with the **svc profiles** command:

[no] svc profiles *{value profile | none}*

In the following example, the user follows the **svc profiles value** command with a question mark (**?**) view the available profiles. Then the user configures the group policy to use the profile *sales*:

```
asa1(config-group-webvpn)# svc profiles value ?

config-group-webvpn mode commands/options:
Available configured profile packages:
  engineering
  sales
asa1(config-group-webvpn)# svc profiles sales
asa1(config-group-webvpn)#
```

Enabling Additional AnyConnect Client Features

To minimize download time, the client only requests downloads (from the security appliance) 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 **svc modules** command from group policy webvpn or username webvpn configuration mode:

> [**no**] **svc 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 security appliance 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 security appliance to download the GINA module for VPN connection to specific groups or users using the **svc 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)# svc modules value vpngina
```

Step 2   Retrieve a copy of the client profiles file (AnyConnectProfile.tmpl). For information on the location of the profiles file for each operating system, see Table 40-1 on page 40-7

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 security appliance using the **svc profile** command from webvpn configuration mode. For example:

```
asa1(config-webvpn)# svc profiles sales disk0:/sales_hosts.xml
```

Translating Languages for AnyConnect User Messages

The security appliance 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 security appliance to translate these user messages and includes the following sections:

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 security appliance 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\cafitz\cvc\main\Api\AgentIfc.cpp:23
#: C:\cygwin\home\cafitz\cvc\main\Api\check\AgentIfc.cpp:22
#: C:\cygwin\home\cafitz\cvc\main\Api\save\AgentIfc.cpp:23
#: C:\cygwin\home\cafitz\cvc\main\Api\save\AgentIfc.cpp~:20
#: C:\cygwin\home\cafitz\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 SSL VPN Features

The following section describes advanced features that fine-tune SSL VPN connections, and includes the following sections:

Enabling Rekey

When the security appliance and the SSL VPN client perform a rekey, 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 **svc rekey** command from group-policy and username webvpn modes.

> [no] svc rekey {**method** {**new-tunnel** | **none** | **ssl**} | **time** *minutes*}

method new-tunnel specifies that the client establishes a new tunnel during rekey.

method none disables rekey.

method ssl specifies that SSL renegotiation takes place during rekey.

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-policy)# svc rekey method ssl
hostname(config-group-policy)# svc rekey time 30
```

Enabling and Adjusting Dead Peer Detection

Dead Peer Detection (DPD) ensures that the security appliance (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 security appliance or client for a specific group or user, and to set the frequency with which either the security appliance or client performs DPD, use the **svc dpd-interval** command from group-policy or username webvpn mode:

> **svc dpd-interval** {[**gateway** {*seconds* | **none***}]* | [**client** {*seconds* | **none***}]*}

> **no svc dpd-interval** {[**gateway** {*seconds* | **none***}]* | [**client** {*seconds* | **none***}]*}

Where:

gateway seconds enables DPD performed by the security appliance (gateway) and specifies the frequency, from 5 to 3600 seconds, with which the security appliance (gateway) performs DPD.

gateway none disables DPD performed by the security appliance.

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 **svc dpd-interval** command from the configuration, use the **no** form of the command:

Note   If you enable DTLS, enable Dead Peer Detection (DPD) also. DPD enables a failed DTLS connection to fallback to TLS. Overwise, the connection terminates.

The following example sets the frequency of DPD performed by the security appliance 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-policy)# svc dpd-interval gateway 30
hostname(config-group-policy)# svc 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.

To set the frequency of keepalive messages, use the **svc keepalive** command from group-policy webvpn or username webvpn configuration mode:

> **[no] svc 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 disabled.

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 security appliance 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)# svc keepalive 300
```

Using Compression

Compression increases the communications performance between the security appliance 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 security appliance, 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 boradband connections.

Compression must be turned-on globally using the **compression svc** command from global configuration mode, and then it can be set for specific groups or users with the **svc compression** command in group-policy and username webvpn modes.

Changing Compression Globally

To change the global compression settings, use the **compression svc** command from global configuration mode:

compression svc

no compression svc

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 svc
```

Changing Compression for Groups and Users

To change compression for a specific group or user, use the **svc compression** command in the group-policy and username webvpn modes:

svc compression {**deflate** | **none**}

no svc compression {**deflate** | **none**}

By default, for groups and users, SSL compression is set to *deflate* (enabled).

To remove the **svc 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)# svc 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 **svc mtu** command from group policy webvpn or username webvpn configuration mode:

> [**no**] **svc mtu** *size*

This command affects only the AnyConnect client. The legacy Cisco SSL VPN Client (SVC) is not capable of adjusting to different MTU sizes.

The default for this command in the default group policy is **no svc 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)# svc mtu 1200
```

Viewing SSL VPN Sessions

You can view information about active sessions using the **show vpn-sessiondb** command in privileged EXEC mode:

> **show vpn-sessiondb svc**

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.

The following example shows the output of the **show vpn-sessiondb svc** command:

```
hostname# show vpn-sessiondb svc

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:28m:48s
Filter Name  :
```

Logging Off SVC Sessions

To log off all SSL VPN sessions, use the **vpn-sessiondb logoff svc** command in global configuration mode:

vpn-sessiondb logoff svc

The following example logs off all SSL VPN sessions:

```
hostname# vpn-sessiondb logoff svc
INFO: Number of sessions of type "svc" logged off : 1
```

You can log off individual sessions using either the **name option, or the index option**:

vpn-session-db logoff name *name*

vpn-session-db logoff index *index*

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 svc** command. The following example shows the username *lee* and index number *1*.

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 svc

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 26 2007
Duration     : 0h:28m:48s
Inactivity   : 0h:00m:04s
Filter Name  :
```

The following example terminates the session using the **name** option of the **vpn-session-db logoff command**:

```
hostname# vpn-sessiondb logoff name tester
Do you want to logoff the VPN session(s)? [confirm]
INFO: Number of sessions with name "mkrupp" logged off : 0

hostname#
```

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.

Updating SSL VPN Client Images

You can update the client images on the security appliance at any time using the following procedure:

Step 1   Copy the new client images to the security appliance using the **copy** command from privileged EXEC mode, or using another method.

Step 2   If the new client image files have the same filenames as the files already loaded, reenter the **svc image** command that is in the configuration. If the new filenames are different, uninstall the old files using the **no svc image** command. Then use the **svc image** command to assign an order to the images and cause the security appliance to load the new images.

Feature History for AnyConnect Connections

Table 40-2 lists the release history for this feature.

Table 40-2   Feature History for AnyConnect Connections

Feature Name	Releases	Feature Information
AnyConnect Connections	8.0(2)	The following commands were introduced or modified: **svc image**, **vpn-tunnel-protocol, vpn-sessiondb**.