Clientless SSL VPN Troubleshooting

Note **Cisco announces the feature deprecation for Clientless SSL VPN effective with ASA version 9.17(1)**. Limited support will continue on releases prior to 9.17(1). Further guidance will be provided regarding migration options to more robust and modern solutions (for example, remote Duo Network Gateway, AnyConnect, remote browser isolation capabilities, and so on).

Recover from Hosts File Errors When Using Application Access

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.

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.

If Application Access is stopped improperly, you leave the 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.

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).

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.

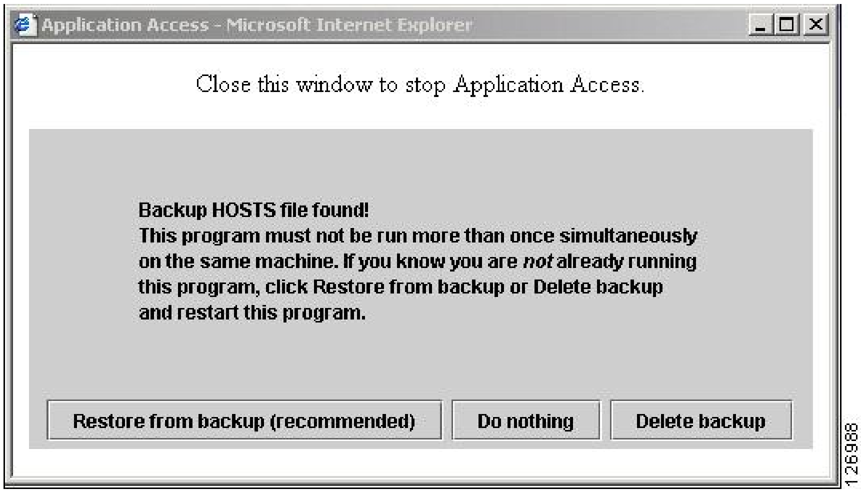
Reconfigure a Host’s File Automatically Using Clientless SSL VPN

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.

Procedure

Step 1 Start Clientless SSL VPN and log in.

Click the **Applications Access** link.


Application Access - Microsoft Internet Explorer
Restore from backup (recommended)
Do nothing
Delete backup
126988

Close this window to stop Application Access.

Backup HOSTS file found!
This program must not be run more than once simultaneously on the same machine. If you know you are *not* already running this program, click Restore from backup or Delete backup and restart this program.

Step 2 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.

Reconfigure 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.

Procedure

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.

Step 6 Click the **Application Access** link.

WebVPN Conditional Debugging

With multiple sessions running on a remote access VPN, troubleshooting can be difficult given the size of the logs. You can use the **debug webvpn condition** command to set up filters to target your debug process more precisely.

debug webvpn condition {**group** *name* | **p-ipaddress** *ip_address* [{**subnet** *subnet_mask* | **prefix** *length*}] | **reset** | **user** *name*}

Where:

- **group** *name* filters on a group policy (not a tunnel group or connection profile).
- **p-ipaddress** *ip_address* [{**subnet** *subnet_mask* | **prefix** *length*}] filters on the public IP address of the client. The subnet mask (for IPv4) or prefix (for IPv6) is optional.
- **reset** resets all filters. You can use the **no debug webvpn condition** command to turn off a specific filter.
- **user** *name* filters by username.

If you configure more than one condition, the conditions are conjoined (ANDed), so that debugs are shown only if all conditions are met.

After setting up the condition filter, use the base **debug webvpn** command to turn on the debug. Simply setting the conditions does not enable the debug. Use the **show debug** and **show webvpn debug-condition** commands to view the current state of debugging.

Troubleshooting a single user session becomes cumbersome when multiple sessions are running on ASA VPN. Conditional debugging enables verifying the logs of specific sessions based on the filter conditions set. SAML, WebVPN request/response, Anyconnect are the modules which supports conditional debugging.

Note Support for "any, any" for IPv4 and IPv6 subnets is provided.

The following shows an example of enabling a conditional debug on the user jdoe.

```
asa3(config)# debug webvpn condition user jdoe

asa3(config)# show webvpn debug-condition
INFO: Webvpn conditional debug is turned ON
INFO: User name filters:
INFO: jdoe

asa3(config)# debug webvpn
INFO: debug webvpn  enabled at level 1.

asa3(config)# show debug
debug webvpn  enabled at level 1
INFO: Webvpn conditional debug is turned ON
INFO: User name filters:
INFO: jdoe
```

Capture 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.

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.

Create a Capture File

Procedure

Step 1 Start the capture utility for Clientless SSL VPN and create a capture named hr, which captures traffic for user2 to a file.

capture capture_name type webvpn user webvpn_username

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.

Example:

```
hostname# capture hr type webvpn user user2
WebVPN capture started.
   capture name   hr
   user name      user2
hostname# no capture hr
```

Step 2 (Optional) Stop 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**.

no capture capture_name

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.

Use a Browser to Display Capture Data

Procedure

Step 1 Start the capture utility for Clientless SSL VPN.

capture capture_name type webvpn user webvpn_username

- *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) Stop the capture utility from capturing packets after a user has logged in and began a Clientless SSL VPN session.

no capture capture_name

Step 3 Open a browser and display the capture named hr in a sniffer format:

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
```

Protect Clientless SSL VPN Session Cookies

Embedded objects such as Flash applications and Java applets, as well as external applications, usually rely on an existing session cookie to work with the server. They get it from a browser using some Javascript on initialization. Adding the httponly flag to the Clientless SSL VPN session cookie makes the session cookie only visible to the browser, not the client-side scripts, and it makes session sharing impossible.

Before you begin

- Change the VPN session cookie setting only when there are no active Clientless SSL VPN sessions.
- Use the **show vpn-sessiondb webvpn** command to check the status of Clientless SSL VPN sessions.
- Use the **vpn-sessiondb logoff webvpn** command to log out of all Clientless SSL VPN sessions.
- The following Clientless SSL VPN features will not work when the **http-only-cookie** command is enabled:
  - Java plug-ins
  - Java rewriter
  - Port forwarding
  - File browser
  - Sharepoint features that require desktop applications (for example, MS Office applications)
  - AnyConnect Web launch
  - Citrix Receiver, XenDesktop, and Xenon
  - Other non-browser-based and browser plugin-based applications

To prevent a Clientless SSL VPN session cookie from being accessed by a third party through a client-side script such as Javascript, perform the following steps:

Procedure

Enable the httponly flag for a Clientless SSL VPN session cookie.

http-only-cookie

Example:

```
hostname(config)# webvpn
hostname(config-webvpn)# http-only-cookie
```

Note Use this command only if Cisco TAC advises you to do so. Enabling this command presents a security risk because the Clientless SSL VPN features listed under the Guidelines section will not work without any warning.