HTTP Services Configuration Guide, Cisco IOS Release 12.2SY

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

Last Updated: August 17, 2011

and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

HTTP 1.1 Web Server and Client

The HTTP 1.1 Web Server and Client feature provides a consistent interface for users and applications by implementing support for HTTP 1.1 in Cisco IOS software-based devices. When combined with the HTTPS feature, the HTTP 1.1 Web Server and Client feature provides a complete, secure solution for HTTP services between Cisco devices.

This module describes the concepts and tasks related to configuring the HTTP 1.1 Web Server and Client feature.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About the HTTP 1.1 Web Server and Client

This feature updates the Cisco implementation of HTTP from 1.0 to 1.1. The HTTP server allows features and applications, such as the Cisco web browser user interface, to be run on your routing device.

The Cisco implementation of HTTP 1.1 is backward-compatible with previous Cisco IOS releases. If you are currently using configurations that enable the HTTP server, no configuration changes are needed because all defaults remain the same.

The process of enabling and configuring the HTTP server also remains the same as in previous releases. Support for Server Side Includes (SSIs) and HTML forms has not changed. Additional configuration options, such as the **ip http timeout-policy** and **ip http max-connections** commands, have been added.

These options allow configurable resource limits for the HTTP server. If you do not use these optional commands, default policies are used.

Remote applications may require that you enable the HTTP server before using them. Applications that use the HTTP server include the following:

- The Cisco web browser user interface, which uses the Cisco IOS Homepage Server, HTTP-based EXEC Server, and HTTP IOS File System (IFS) Server.
- The VPN Device Manager (VDM) application, which uses the VDM Server and the XML Session Manager (XSM).
- The QoS Device Manager (QDM) application, which uses the QDM Server.
- IP Phone and Cisco IOS Telephony Service applications, which use the ITS Local Directory Search and IOS Telephony Server (ITS).

No Cisco applications use the HTTP Client in Cisco IOS Release 12.2(15)T.

- About HTTP Server General Access Policies, page 2

About HTTP Server General Access Policies

The **ip http timeout-policy** command allows you to specify general access characteristics for the server by configuring a value for idle time, connection life, and request maximum. By adjusting these values, you can configure a general policy; for example, if you want to maximize throughput for HTTP connections, you should configure a policy that minimizes the connection overhead. You can configure this type of policy by specifying large values for the **life** and **request** options so that each connection stays open longer and more requests are processed for each connection.

Another example would be to configure a policy that minimizes the response time for new connections. You can configure this type of policy by specifying small values for the **life** and **request** options so that the connections are quickly released to serve new clients.

A throughput policy would be better for HTTP sessions with dedicated management applications because it would allow the application to send more requests before the connection is closed, while a response time policy would be better for interactive HTTP sessions because it would allow more people to connect to the server at the same time without having to wait for connections to become available.

In general, you should configure these options as appropriate for your environment. The value for the **idle** option should be balanced; it should be large enough so as to not cause an unwanted request or response timeout on the connection and small enough so as to not hold a connection open longer than necessary.

Access security policies for the HTTP server are configured using the following commands:

- **ip http authentication**—Allows only selective users to access the server.
- **ip http access-class**—Allows only selective IP hosts to access the server.
- **ip http accounting commands**—Specifies the command accounting method for HTTP server users.

How to Configure the HTTP 1.1 Web Server and Client

- Configuring the HTTP 1.1 Web Server, page 3
- Configuring the HTTP Client, page 5

Configuring the HTTP 1.1 Web Server

Perform this task to enable the HTTP server and configure optional server characteristics. The HTTP server is disabled by default.

Note If you want to configure authentication (step 4), you must configure the authentication type before you begin configuring the HTTP 1.1 web server.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **ip http authentication** {**aaa** | **enable** | **local** | **tacacs**}
5. **ip http accounting commands** *level* {**default** | *named-accounting-method-list*}
6. **ip http port** *port-number*
7. **ip http path** *url*
8. **ip http access-class** *access-list-number*
9. **ip http max-connections** *value*
10. **ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip http server** **Example:** `Router(config)# ip http server`	Enables the HTTP 1.1 server, including the Cisco web browser user interface. **Note** If you are enabling HTTP over the Secure Socket Layer (HTTPS) server using the **ip http secure-server** command, you should disable the standard HTTP server using the **no ip http server** command. This command is required to ensure only secure connections to the server.

	Command or Action	Purpose
Step 4	**ip http authentication** {**aaa** | **enable** | **local** | **tacacs**} **Example:** `Router(config)# ip http authentication local`	(Optional) Specifies the authentication method to be used for login when a client connects to the HTTP server. The methods for authentication are: • **aaa**—Indicates that the authentication method used for the authentication, authorization, and accounting (AAA) login service (specified by the **aaa authentication login default** command) should be used for authentication. • **enable**—Indicates that the "enable" password should be used for authentication. (This is the default method.) • **local**—Indicates that the login username, password, and privilege-level access combination specified in the local system configuration (by the **username** global configuration command) should be used for authentication and authorization. • **tacacs**—Indicates that the TACACS (or XTACACS) server should be used for authentication.
Step 5	**ip http accounting commands** *level* {**default** | *named-accounting-method-list*} **Example:** `Router(config)# ip http accounting commands 15 default`	(Optional) Specifies a particular command accounting method for HTTP server users. • Command accounting for HTTP and HTTPS is automatically enabled when AAA is configured on the device. It is not possible to disable accounting for HTTP and HTTPS. HTTP and HTTPS will default to using the global AAA default method list for accounting. The CLI can be used to configure HTTP and HTTPS to use any predefined AAA method list. • *level*—Valid privilege level entries are integers from 0 to 15. • **default**—Indicates the **default** accounting method list configured by the aaa accounting commands. • *named-accounting-method-list*—Indicates the name of the predefined command accounting method list.
Step 6	**ip http port** *port-number* **Example:** `Router(config)# ip http port 8080`	(Optional) Specifies the server port that should be used for HTTP communication (for example, for the Cisco web browser user interface).
Step 7	**ip http path** *url* **Example:** `Router(config)# ip http path slot1:`	(Optional) Sets the base HTTP path for HTML files. The base path is used to specify the location of the HTTP server files (HTML files) on the local system. • Generally, HTML files are located in the system flash memory.
Step 8	**ip http access-class** *access-list-number* **Example:** `Router(config)# ip http access-class 20`	(Optional) Specifies the access list that should be used to allow access to the HTTP server.

	Command or Action	Purpose
Step 9	**ip http max-connections** *value* **Example:** `Router(config)# ip http max-connections 10`	(Optional) Sets the maximum number of allowed concurrent connections to the HTTP server. • The default value is 5.
Step 10	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value* **Example:** `Router(config)# ip http timeout-policy idle 30 life 120 requests 100`	(Optional) Sets the characteristics that determine how long a connection to the HTTP server should remain open. The characteristics include the following: • **idle**—The maximum number of seconds the connection will be kept open if no data is received or if response data cannot be sent out on the connection. Note that a new value may not take effect on any already existing connections. If the server is too busy or the limit on the **life** time or the number of **requests** is reached, the connection may be closed sooner. The default value is 180 seconds (3 minutes). • **life**—The maximum number of seconds the connection will be kept open, from the time the connection is established. Note that the new value may not take effect on any already existing connections. If the server is too busy or the limit on the idle time or the number of requests is reached, it may close the connection sooner. Also, since the server will not close the connection while actively processing a request, the connection may remain open longer than the specified **life** time if processing is occurring when the life maximum is reached. In this case, the connection will be closed when processing finishes. The default value is 180 seconds (3 minutes). The maximum value is 86400 seconds (24 hours). • **requests**—The maximum limit on the number of requests processed on a persistent connection before it is closed. Note that the new value may not take effect on already existing connections. If the server is too busy or the limit on the idle time or the life time is reached, the connection may be closed before the maximum number of requests are processed. The default value is 1. The maximum value is 86400.

Configuring the HTTP Client

Perform this task to enable the HTTP client and configure optional client characteristics.

The standard HTTP 1.1 client and the secure HTTP client are always enabled. No commands exist to disable the HTTP client. For information about configuring optional characteristics for the HTTPS client, see the *HTTPS-HTTP Server and Client with SSL 3.0* feature module.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http client cache** {**ager interval** *minutes* | **memory** {**file** *file-size-limit* | **pool** *pool-size-limit*}
4. **ip http client connection** {**forceclose** | **idle timeout** *seconds* | **retry** *count* | **timeout** *seconds*}
5. **ip http client password** *password*
6. **ip http client proxy-server** *proxy-name* **proxy-port** *port-number*
7. **ip http client response timeout** *seconds*
8. **ip http client source-interface** *type number*
9. **ip http client username** *username*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip http client cache** {**ager interval** *minutes*	**memory** {**file** *file-size-limit*	**pool** *pool-size-limit*} **Example:** `Router(config)# ip http client cache memory file 5`	Configures the HTTP client cache.
Step 4	**ip http client connection** {**forceclose**	**idle timeout** *seconds*	**retry** *count*	**timeout** *seconds*} **Example:** `Router(config)# ip http client connection timeout 10`	Configures an HTTP client connection.
Step 5	**ip http client password** *password* **Example:** `Router(config)# ip http client password pswd1`	Configures the default password used for connections to remote HTTP servers.

	Command or Action	Purpose
Step 6	**ip http client proxy-server** *proxy-name* **proxy-port** *port-number* **Example:** `Router(config)# ip http client proxy-server server1 proxy-port 52`	Configures an HTTP proxy server.
Step 7	**ip http client response timeout** *seconds* **Example:** `Router(config)# ip http client response timeout 60`	Specifies the timeout value, in seconds, that the HTTP client waits for a response from the server.
Step 8	**ip http client source-interface** *type number* **Example:** `Router(config)# ip http client source-interface ethernet1/0`	Configures a source interface for the HTTP client.
Step 9	**ip http client username** *username* **Example:** `Router(config)# ip http client user1`	Configures the default username used for connections to remote HTTP servers.

Configuration Examples for the HTTP 1.1 Web Server and Client

Example Configuring the HTTP 1.1 Web Server

The following example shows a typical configuration that enables the server and sets some characteristics:

```
ip http server
ip http authentication aaa
ip http accounting commands 15 default
ip http path flash:
ip access-list standard 20
 permit 209.165.202.130 0.0.0.255
 permit 209.165.201.1 0.0.255.255
 permit 209.165.200.225 0.255.255.255
! (Note: all other access implicitly denied)
 end
ip http access-class 10
ip http max-connections 10
ip http accounting commands 1 oneacct
```

In the following example, a throughput timeout policy is applied. This configuration will allow each connection to be idle for a maximum of 30 seconds (approximately). Each connection will remain open (be

"alive") until either the HTTP server has been processing requests for approximately 2 minutes (120 seconds) or until approximately 100 requests have been processed.

```
ip http timeout-policy idle 30 life 120 requests 100
```

In the following example, a Response Time timeout policy is applied. This configuration will allow each connection to be idle for a maximum of 30 seconds (approximately). Each connection will be closed as soon as the first request has been processed.

```
ip http timeout-policy idle 30 life 30 requests 1
```

Example Verifying HTTP Connectivity

To verify remote connectivity to the HTTP server, enter the system IP address in a web browser, followed by a colon and the appropriate port number (80 is the default port number).

For example, if the system IP address is 209.165.202.129 and the port number is 8080, enter http://209.165.202.129:8080 as the URL in a web browser.

If HTTP authentication is configured, a login dialog box will appear. Enter the appropriate username and password. If the default login authentication method of "enable" is configured, you may leave the username field blank, and use the "enable" password to log in.

The system home page should appear in your browser.

Where to Go Next

For information about secure HTTP connections using Secure Sockets Layer (SSL) 3.0, refer to the *HTTPS - HTTP with SSL 3.0* feature module.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
HTTP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS HTTP Services Command Reference*
HTTPS	• HTTPS--HTTP with SSL 3.0 feature module • Firewall Support of HTTPS Authentication Proxy feature module

Standards and RFCs

Standard/RFC	Title
No specific standards are supported by this feature. Note that HTTP 1.1, as defined in RFC 2616, is currently classified as a “Standards Track” document by the IETF.	—
RFC 2616	*Hypertext Transfer Protocol -- HTTP/1.1*

The Cisco implementation of the HTTP Version 1.1 supports a subset of elements defined in RFC 2616. Following is a list of supported RFC 2616 headers:

- Allow (Only GET, HEAD, and POST methods are supported)
- Authorization, WWW-Authenticate - Basic authentication only
- Cache-control
- Chunked Transfer Encoding
- Connection close
- Content-Encoding
- Content-Language
- Content-Length
- Content-Type
- Date, Expires
- Location

MIBs

MIB	MIBs Link
• No specific MIBs are supported for this feature.	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature History and Information for the HTTP 1.1 Web Server and Client

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1 Feature History and Information for the HTTP 1.1 Web Server and Client

Feature Name	Releases	Feature Information
HTTP 1.1 Web Server and Client	12.2(15)T 12.2(33)SB 12.2(33)SRC 12.4(15)T Cisco IOS XE 3.1.0SG	The HTTP 1.1 Web Server and Client feature provides a consistent interface for users and applications by implementing support for HTTP 1.1 in Cisco IOS software-based devices. When combined with the HTTPS feature, the HTTP 1.1 Web Server and Client feature provides a complete, secure solution for HTTP services between Cisco devices. The following commands were introduced or modified by this feature: **debug ip http all**, **debug ip http client**, **ip http access-class**, **ip http authentication**, **ip http client cache**, **ip http client connection**, **ip http client password**, **ip http client proxy-server**, **ip http client response timeout**, **ip http client source-interface**, **ip http client username**, **ip http max-connections**, **ip http path**, **ip http port**, **ip http server**, **ip http timeout-policy**, **show ip http client**, **show ip http client connection**, **show ip http client history**, **show ip http client session-module**, **show ip http server**, **show ip http server secure status**.

Feature Name	Releases	Feature Information
HTTP TACACS+ Accounting Support	12.2(33)SB 12.2(33)SRC 12.2(50)SY 12.4(15)T	The HTTP TACACS+ Accounting Support feature introduces the **ip http accounting commands** command. This command is used to specify a particular command accounting method for HTTP server users. Command accounting provides information about commands, executed on a device, for a specified privilege level. Each command accounting record corresponds to one IOS command executed at its respective privilege level, as well as the date and time the command was executed, and the user who executed it. The following commands were introduced or modified by this feature: **ip http accounting commands**.