

Transferring Files Using HTTP or HTTPS

Cisco IOS Release 12.4 provides the ability to transfer files between your Cisco IOS software-based device and a remote HTTP server using the HTTP or HTTP Secure (HTTPS) protocol. HTTP and HTTPS can now be specified as the targets and source locations in Cisco IOS command-line interface (CLI) commands that use file system prefixes such as the **copy** command.

- Prerequisites for Transferring Files Using HTTP or HTTPs, on page 1
- Restrictions for Transferring Files Using HTTP or HTTPs, on page 1
- Information About File Transfers Using HTTP or HTTPs, on page 2
- How to Transfer Files Using HTTP or HTTPs, on page 2
- Configuration Examples for the File Transfer Using HTTP or HTTPs, on page 8
- Additional References, on page 10
- Feature Information for Transferring Files Using HTTP or HTTPS, on page 12

Prerequisites for Transferring Files Using HTTP or HTTPs

To copy files to or from a remote HTTP server, your system must support the HTTP client feature, which is integrated in most Cisco IOS software images. The HTTP client is enabled by default. To determine if the HTTP client is supported on your system, issue the **show ip http client all**command. If you are able to execute the command, the HTTP client is supported.

Commands exist for the optional configuration of the embedded HTTP client and for the HTTPS client, but the default configuration is sufficient for using the File Transfer Using HTTP or HTTPS feature. For information on configuring optional HTTP or HTTPS client characteristics, see the “Related Documents” section.

Restrictions for Transferring Files Using HTTP or HTTPs

- Existing limitations to the **copy** command, such as no network-to-network copies, are in effect for the File Transfer Using HTTP or HTTPS feature.

Note The **copy** command in Cisco IOS Release 12.4T does not work in conjunction with older versions of the Apache server software. The Apache server software must be upgraded to version 2.0.49 or later in order to use the copy command.

- From Cisco Release 17.3.1, a TLS connection is only established when the hostname matches with the Subject Alternative Name (SAN) or Common Name (CN) in a certificate. If a server doesn't meet these expectations and sends invalid attributes, then the SSL handshake is denied because there a TLS connection is not established. Hence, the HTTPS copy won't be successful.
- From Cisco IOS XE 17.15.1a, you must provide the absolute file path for the copy to succeed. See section HTTP or HTTPs File Transfer Using Absolute File Path for sample configurations. Providing the relative file path will result in error "No such file or directory".



Note Only the absolute path is used in the file transfer. The file path provided in the **ip http path** <> command is not used in the file transfer.

Information About File Transfers Using HTTP or HTTPs

To transfer files using HTTP or HTTPS, you should understand the following concept:

The File Transfer Using HTTP or HTTPs feature provides the capability to copy files, such as Cisco IOS image files, core files, configuration files, log files, scripts, and so on, to and from a remote server and your local routing device using the Cisco IOS **copy** command and command-line interface. The HTTP copy operation works in the same way as copying from other remote file systems, such as FTP or TFTP.

The HTTP copy operation can use the embedded HTTPS client for HTTP Secure transfers, providing secure and authenticated file transfers within the context of a public key infrastructure (PKI).

How to Transfer Files Using HTTP or HTTPs

This section contains the following procedures:



Note To use the File Transfer Using HTTP feature, you may need to specify a username and password for the HTTP connections for those servers that require a username and password to connect. Commands are also available to specify custom connection characteristics, although default settings can be used. The feature also offers commands to monitor and maintain connections and files.

Configuring HTTP Connection Characteristics for File Transfers

Default values are provided for HTTP File transfers. The following task is used to customize the connection characteristics for your network to specify a username and password, connection preferences, a remote proxy server, and the source interface to be used.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http client connection** {**forceclose** | **idletimeout***seconds* | **timeout***seconds*}

4. **ip http client username** *username*
5. **ip http client password** *password*
6. **ip http client proxy-server** {*proxy-name* | *ip-address*} [**proxy-port***port-number*]
7. **ip http client source-interface** *interface-id*
8. **do copy running-config startup-config**
9. **end**

DETAILED STEPS

Procedure

	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 connection** {**forceclose** | **idletimeout***seconds* | **timeout***seconds*} **Example:** `Router(config)# ip http client connection timeout 15`	Configures characteristics for HTTP client connections to a remote HTTP server for all file transfers: • **forceclose** --Disables the default persistent connection. • **idle timeout** *seconds* --Sets the period of time allowed for an idle connection, in a range from 1 to 60 seconds. Default timeout is 30 seconds. • **timeout** *seconds* --Sets the maximum time the HTTP client waits for a connection, in a range from 1 to 60 seconds. Default is 10 seconds.
Step 4	**ip http client username** *username* **Example:** `Router(config)# ip http client username user1`	Specifies the username to be used for HTTP client connections that require user authentication. **Note** You can also specify the username on the CLI when you issue the **copy** command, in which case the username entered overrides the username entered with this command. See the "Downloading a File from a Remote Server Using HTTP or HTTPs: Example" section for an example.
Step 5	**ip http client password** *password* **Example:** `Router(config)# ip http client password letmein`	Specifies the password to be used for HTTP client connections that require user authentication. **Note**

	Command or Action	Purpose
		You can also specify the password on the CLI when you issue the **copy** command, in which case the password entered overrides the password entered with this command. See the "Downloading a File from a Remote Server Using HTTP or HTTPs: Example" section for an example.
Step 6	**ip http client proxy-server** {*proxy-name* | *ip-address*} [**proxy-port***port-number*] **Example:** `Router(config)# ip http client proxy-server edge2 proxy-port 29`	Configures the HTTP client to connect to a remote proxy server for HTTP file system client connections. • The optional**proxy-port***port-number* keyword and argument specify the proxy port number on the remote proxy server.
Step 7	**ip http client source-interface** *interface-id* **Example:** `Router(config)# ip http client source-interface Ethernet 0/1`	Specifies the interface for the source address in all HTTP client connections.
Step 8	**do copy running-config startup-config** **Example:** `Router(config)# do copy running-config startup-config`	(Optional) Saves the running configuration as the startup configuration file. • The **do** command allows you to execute privileged EXEC mode commands from global configuration mode.

Step 9	**end**
	Example:
	`Router(config)# end`
	Example:
	`Router#`
		Ends your configuration session and returns the CLI to user EXEC mode.

Downloading a File from a Remote Server Using HTTP or HTTPs

Perform this task to download a file from a remote HTTP server using HTTP or HTTPs. The **copy** command helps you to copy any file from a source to a destination.

SUMMARY STEPS

1. **enable**
2. Do one of the following:
   - **copy** [**/erase**] [**/noverify**] **http://***remote-source-urllocal-destination-url*
   - **copy https://** *remote-source-url local-destination-url*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable` **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

Step 2	Do one of the following:
	• **copy** [**/erase**] [**/noverify**] **http://***remote-source-urllocal-destination-url*
	• **copy https://** *remote-source-url local-destination-url*
	Example:
	`Router# copy http://user1:mypassword@209.165.202.129:8080/image_files/c7200-i-mx flash:c7200-i-mx`
	Example:
	`Router# copy`
	Example:
	`copy https://user1:mypassword@209.165.202.129:8080/image_files/c7200-i-mx flash:c7200-i-mx`
		Copies a file from a remote web server to a local file system using HTTP or HTTPS.
		• **/erase** --Erases the local destination file system before copying. This option is provided on Class B file system platforms with limited memory to allow an easy way to clear local flash memory space.
		• **/noverify** --If the file being copied is an image file, this keyword disables the automatic image verification that occurs after an image is copied.
		• The *remote-source-url* argument is the location URL (or alias) from which to get the file to be copied, in standard Cisco IOS file system HTTP syntax as follows:
		http:// [[*username*:*password*]**@**] {*hostname* | *host-ip*}[*/filepath*]*/filename*
		Note The optional *username* and *password* arguments can be used to log in to an HTTP server that requires user authentication, in place of configuring the **iphttpclientusername** and **iphttpclientpassword** global configuration commands to specify these authentication strings.
		• The *local-destination-url* is the location URL (or alias) to put the copied file, in standard Cisco IOS file system syntax as follows:
		filesystem **:** [*/filepath*][*/filename*]
		Note For more information on URL syntax when you use the **copy** command, see the "Additional References" section.

Troubleshooting Tips

If file transfers from a remote web server fail, verify the following:

- Your router has an active connection to the Internet.
- The correct path and filename have been specified.
- The remote server requires a username and password.
- The remote server has a nonstandard communications port configured. (The default port for HTTP is 80; the default port for HTTPS is 443.)

The CLI returns error messages to help you determine the cause of a failed copy request. Additional information on the copy process can be displayed with the **debugiphttpclientall** command.

Uploading a File to a Remote Server Using HTTP or HTTPS

Perform this task to upload a file to a remote HTTP server using HTTP or HTTPS.

SUMMARY STEPS

1. **enable**
2. Do one of the following:
   - **copy** [**/erase**] [**/noverify**] *local-source-url***http://***remote-destination-url*
   - **copy** *local-source-url* **https://** *remote-destination-url*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	Do one of the following: • **copy** [**/erase**] [**/noverify**] *local-source-url***http://***remote-destination-url* • **copy** *local-source-url* **https://** *remote-destination-url* **Example:** `Router# http://user1:mypassword@209.165.202.129:8080/image_files/c7200-i-mx_backup` **Example:**	Copies a file from a local file system to a remote web server using HTTP or HTTPS. • **/erase** --Erases the local destination file system before copying. This option is provided on Class B file system platforms with limited memory to allow an easy way to clear local flash memory space. • **/noverify** --If the file being copied is an image file, this keyword disables the automatic image verification that occurs after an image is copied.

	Command or Action	Purpose
	`Router# copy flash:c7200-i-mx http://user1:mypassword@209.165.202.129:8080/image_files/c7200-i-mx_backup` **Example:**	• The *local-source-url* argument is the location URL (or alias) from which to get the file to be copied, in standard Cisco IOS file system syntax as follows: **http://** [[*username*:*password*]**@**] {*hostname* | *host-ip*}[*/filepath*]*/filename* **Note** The optional *username* and *password* arguments can be used to log in to an HTTP server that requires user authentication, in place of configuring the **iphttpclientusername** and **iphttpclientpassword** global configuration commands to specify these authentication strings. • The *remote-destination-url* is the URL (or alias) to put the copied file, in standard Cisco IOS file system syntax, as follows: *filesystem* **:** [*/filepath*][*/filename*] **Note** For more information on URL syntax when you use the **copy** command, see the "Additional References" section.

Troubleshooting Tips

If file transfers from a remote web server fail, verify the following:

- Your router has an active connection to the Internet.
- The correct path and filename have been specified.
- The remote server requires a username and password.
- The remote server has a nonstandard communications port configured. (The default port for HTTP is 80; the default port for HTTPS is 443.)

The CLI returns error messages to help you determine the cause of a failed copy request. Additional information on the copy process can be displayed with the **debugiphttpclientall** command.

Maintaining and Monitoring File Transfers Using HTTP

Perform this task to maintain and monitor HTTP connections. Steps 2 through 4 can be performed in any order.

SUMMARY STEPS

1. **enable**
2. **show ip http client connection**
3. **show ip http client history**

4. **show ip http client session-module**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ip http client connection** **Example:** `Router# show ip http client connection`	Displays details about active HTTP client connections.
Step 3	**show ip http client history** **Example:** `Router# show ip http client history`	Displays the last 20 URLs accessed by the HTTP client.
Step 4	**show ip http client session-module** **Example:** `Router# show ip http client session-module`	Displays details about about sessions (applications) that have registered with the HTTP client.

Configuration Examples for the File Transfer Using HTTP or HTTPs

Configuring HTTP Connection Characteristics for File Transfers Example

The following example shows how to configure the HTTP password and username for connection to a remote server that authenticates all users. The example also shows how to configure the connection for a 20-second idle connection period. The maximum time the HTTP client waits for a connection remains at the default 10 seconds.

```
Router(config)# ip http client connection idle timeout 20
Router(config)# ip http client password Secret
Router(config)# ip http client username User1
Router(config)# do show running-config | include ip http client
```

Downloading a File from a Remote Server Using HTTP or HTTPs Example

The following example shows how to configure the file c7200-i-mx is copied from a remote server to flash memory using HTTP. This example also shows how to enter a username and password from the command line for an HTTP server that authenticates users.

```
Router# copy http://user1:mypassword@209.165.202.129:8080/image_files/c7200-i-mx
flash:c7200-i-mx
```

Uploading a File from Flash to the Remote HTTP Server Example

The following example shows how to copy a file from flash memory to the remote HTTP server. The example shows the prompts and displays that can be expected from transferring a file using the**copy** privileged EXEC command.

```
Router# copy flash:c7200-js-mz.ELL2 http://172.19.209.190/user1/c7200-js-mz.ELL2
Address or name of remote host [172.19.209.190]?
Destination filename [user1/c7200-js-mz.ELL2]?
Storing http://172.19.209.190/user1/c7200-js-mz.ELL2 !!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
17571956 bytes copied in 57.144 secs (307503 bytes/sec)
```

Downloading a File from the Remote HTTP Server to Flash Memory Example

The following example shows how to copy a file from the remote HTTP server to the flash memory. The example shows the prompts and displays that can be expected from transferring a file using the **copy** privileged EXEC command.

```
Router# copy http://172.19.209.190/user1/c7200-i-mz.test  flash:c7200-i-mz.test
Destination filename [c7200-i-mz.test]?
Loading http://172.19.209.190/user1/c7200-i-mz.test
.
.
.
```

11272788 bytes copied in 527.104 secs (21386 bytes/sec)

Uploading a File to a Remote Server Using HTTP or HTTPs

The following example shows how to copy a file to the remote server using HTTP or HTTPs.

```
router#copy flash
: http:
Source filename []? running-config
Address or name of remote host []? 10.1.102.1 Destination filename [pilot-confg]?file1 ...
```

HTTP or HTTPs File Transfer Using Absolute File Path

From Cisco IOS XE 17.15.1a, providing the relative file path results in an error. You must provide the absolute file path for the transfer to succeed.

Example	Old Behavior (Cisco IOS XE 17.14.x and lower)	New Behavior (Cisco IOS XE 17.15.1a and higher)

Relative File Path
Configuration:
`Router(config)#ip http path <path e.g. flash:>`
Example 1:
`Router#copy http(s)://<server ip>/common.tar flash:`
	System picks the absolute file path from the configuration, transfer is successful.	System does not pick the absolute file path from the configuration.
		`Error:(No such file or directory)`

Example 2: `wget -N http://<device ip>/common.tar -t 2 --no-check-certificate --no-proxy --http-user=<username> --http-password=<password>`	System picks the absolute file path from the configuration, transfer is successful.	System does not pick the absolute file path from the configuration. `Error:(No such file or directory)`

Absolute File Path
Configuration:
`Router(config)#ip http path <path e.g. flash:>`
Example 1:
`Router#copy http(s)://<server ip>/flash/common.tar flash:`
	Transfer is successful.	Transfer is successful.

Example 2: `wget -N http://<device ip>/flash/common.tar -t 2 --no-check-certificate --no-proxy --http-user=<username> --http-password=<password>`	Transfer is successful.	Transfer is successful.

Additional References

The following sections provide information related to transferring files using HTTP or HTTPS.

Related Documents

Related Topic	Document Title
Secure HTTP communications	*HTTPS --HTTP Server and Client with SSL 3.0*
Cisco IOS embedded web server	*HTTP 1.1 Web Server and Client*
Cisco IOS embedded web client	*HTTP 1.1 Client*
Network Management Commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS Network Management Command Reference*
Configuration Fundamentals Commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new of modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2616	*Hypertext Transfer Protocol -- HTTP/1.1*, R. Fielding, et al.
RFC 2617	*HTTP Authentication: Basic and Digest Access Authentication*, J. Franks, et al.

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for Transferring Files Using HTTP or HTTPS

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 Information for Transferring Files Using HTTP or HTTPS

Feature Name	Releases	Feature Information
File Download Using HTTP	12.3(2)T	The File Download Using HTTP feature allows you to copy files from an HTTP server to a Cisco IOS software-based platform.
File Upload Using HTTP	12.3(7)T	
File Transfer Using HTTP	12.3(7)T	The File Transfer Using HTTP feature provides the capability to copy files, such as Cisco IOS image files, core files, configuration files, log files, and scripts to and from a remote server and your local routing device using the Cisco IOS **copy** command and command-line interface. The HTTP copy operation works in the same way as copying from other remote file systems, such as FTP or TFTP. This feature provides support for copying files from a Cisco IOS software-based platform to an HTTP server, using either HTTP or HTTPs.