

NAME

CURLOPT_QUOTE – (S)FTP commands to run before transfer

SYNOPSIS

```
#include <curl/curl.h>
```

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_QUOTE,
                          struct curl_slist *cmds);
```

DESCRIPTION

Pass a pointer to a linked list of FTP or SFTP commands to pass to the server prior to your request. This is done before any other commands are issued (even before the CWD command for FTP). The linked list should be a fully valid list of 'struct curl_slist' structs properly filled in with text strings. Use *curl_slist_append(3)* to append strings (commands) to the list, and clear the entire list afterwards with *curl_slist_free_all(3)*.

Using this option multiple times makes the last set list override the previous ones. Set it to NULL to disable its use again. libcurl does not copy the list, it needs to be kept around until after the transfer has completed.

When speaking to an FTP server, prefix the command with an asterisk (*) to make libcurl continue even if the command fails as by default libcurl stops at first failure.

The set of valid FTP commands depends on the server (see RFC 959 for a list of mandatory commands).

libcurl does not inspect, parse or "understand" the commands passed to the server using this option. If you change connection state, working directory or similar using quote commands, libcurl does not know about it.

The path arguments for FTP or SFTP should use double quotes to distinguish a space from being the parameter separator or being a part of the path. For example, rename with sftp using a quote command like this:

```
rename "test/_upload.txt" "test/Hello World.txt"
```

For SFTP, filenames must be provided within double quotes to embed spaces, backslashes, quotes or double quotes. Within double quotes the following escape sequences are available for that purpose: \,

SFTP commands

atime date file

The atime command sets the last access time of the file named by the file operand. The date expression can be all sorts of date strings, see the *curl_getdate(3)* man page for date expression details. (Added in 7.73.0)

chgrp group file

The chgrp command sets the group ID of the file named by the file operand to the group ID specified by the group operand. The group operand is a decimal integer group ID.

chmod mode file

The chmod command modifies the file mode bits of the specified file. The mode operand is an octal integer mode number.

chown user file

The chown command sets the owner of the file named by the file operand to the user ID specified by the user operand. The user operand is a decimal integer user ID.

ln source_file target_file

The **ln** and **symlink** commands create a symbolic link at the target_file location pointing to the source_file location.

mkdir directory_name

The **mkdir** command creates the directory named by the directory_name operand.

mtime date file

The **mtime** command sets the last modification time of the file named by the file operand. The date expression can be all sorts of date strings, see the *curl_getdate(3)* man page for date expression details. (Added in 7.73.0)

pwd The **pwd** command returns the absolute path of the current working directory.

rename source target

The **rename** command renames the file or directory named by the source operand to the destination path named by the target operand.

rm file The **rm** command removes the file specified by the file operand.

rmdir directory

The **rmdir** command removes the directory entry specified by the directory operand, provided it is empty.

statvfs file

The **statvfs** command returns statistics on the file system in which specified file resides.

symlink source_file target_file

See **ln**.

DEFAULT

NULL

PROTOCOLS

This functionality affects ftp and sftp

EXAMPLE

```
int main(void)
{
    struct curl_slist *cmdlist = NULL;
    cmdlist = curl_slist_append(cmdlist, "RNFR source-name");
    cmdlist = curl_slist_append(cmdlist, "RNT0 new-name");

    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "ftp://example.com/foo.bin");

        /* pass in the FTP commands to run before the transfer */
        curl_easy_setopt(curl, CURLOPT_QUOTE, cmdlist);

        result = curl_easy_perform(curl);

        curl_easy_cleanup(curl);
    }

    curl_slist_free_all(cmdlist);
}
```

HISTORY

SFTP support added in 7.16.3. *-prefix for SFTP added in 7.24.0

AVAILABILITY

Added in curl 7.1

RETURN VALUE

curl_easy_setopt(3) returns a CURLcode indicating success or error.

CURLE_OK (0) means everything was OK, non-zero means an error occurred, see *libcurl-errors(3)*.

SEE ALSO

**CURLOPT_CUSTOMREQUEST(3), CURLOPT_DIRLISTONLY(3), CURLOPT_POSTQUOTE(3),
CURLOPT_PREQUOTE(3)**