

NAME

CURLOPT_POSTQUOTE – (S)FTP commands to run after the transfer

SYNOPSIS

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

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

DESCRIPTION

Pass a pointer to a linked list of FTP or SFTP commands to pass to the server after your FTP transfer request. The commands are only issued if no error occur. The linked list should be a fully valid list of struct curl_slist structs properly filled in as described for *CURLOPT_QUOTE(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.

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 after the transfer */
        curl_easy_setopt(curl, CURLOPT_POSTQUOTE, cmdlist);

        result = curl_easy_perform(curl);

        curl_easy_cleanup(curl);
    }
    curl_slist_free_all(cmdlist);
}
```

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_PREQUOTE(3), CURLOPT_QUOTE(3)