

NAME

CURLOPT_PROXYHEADER – set of HTTP headers to pass to proxy

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_PROXYHEADER,  
    struct curl_slist *headers);
```

DESCRIPTION

Pass a pointer to a linked list of HTTP headers to pass in your HTTP request sent to a proxy. The rules for this list is identical to the *CURLOPT_HTTPHEADER(3)* option's.

The headers set with this option is only ever used in requests sent to a proxy - when there is also a request sent to a host.

The first line in a request (containing the method, usually a GET or POST) is NOT a header and cannot be replaced using this option. Only the lines following the request-line are headers. Adding this method line in this list of headers causes your request to send an invalid header.

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 all supported protocols

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();

    struct curl_slist *list;

    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com");
        curl_easy_setopt(curl, CURLOPT_PROXY, "http://proxy.example.com:80");

        list = curl_slist_append(NULL, "Shoesize: 10");
        list = curl_slist_append(list, "Accept:");

        curl_easy_setopt(curl, CURLOPT_PROXYHEADER, list);

        result = curl_easy_perform(curl);

        curl_slist_free_all(list); /* free the list again */
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.37.0

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_HEADEROPT(3), CURLOPT_HTTPHEADER(3)