

NAME

CURLOPT_HEADEROPT – send HTTP headers to both proxy and host or separately

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_HEADEROPT, long bitmask);
```

DESCRIPTION

Pass a long that is a bitmask of options of how to deal with headers. The two mutually exclusive options are:

CURLHEADER_UNIFIED – the headers specified in *CURLOPT_HTTPHEADER(3)* are used in requests both to servers and proxies. With this option enabled, *CURLOPT_PROXYHEADER(3)* does not have any effect.

CURLHEADER_SEPARATE – makes *CURLOPT_HTTPHEADER(3)* headers only get sent to a server and not to a proxy. Proxy headers must be set with *CURLOPT_PROXYHEADER(3)* to get used. Note that if a non-CONNECT request is sent to a proxy, libcurl sends both server headers and proxy headers. When doing CONNECT, libcurl sends *CURLOPT_PROXYHEADER(3)* headers only to the proxy and then *CURLOPT_HTTPHEADER(3)* headers only to the server.

DEFAULT

CURLHEADER_SEPARATE (changed in 7.42.1, used CURLHEADER_UNIFIED before then)

PROTOCOLS

This functionality affects http only

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        struct curl_slist *list;
        list = curl_slist_append(NULL, "Shoesize: 10");
        list = curl_slist_append(list, "Accept:");
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/");
        curl_easy_setopt(curl, CURLOPT_PROXY, "http://localhost:8080");
        curl_easy_setopt(curl, CURLOPT_HTTPHEADER, list);

        /* HTTPS over a proxy makes a separate CONNECT to the proxy, so tell
           libcurl to not send the custom headers to the proxy. Keep them
           separate. */
        curl_easy_setopt(curl, CURLOPT_HEADEROPT, CURLHEADER_SEPARATE);
        result = curl_easy_perform(curl);
        curl_slist_free_all(list);
        curl_easy_cleanup(curl);
    }
}
```

HISTORY

CURLHEADER_* macros became *long* types in 8.16.0, prior to this version a *long* cast was necessary when passed to *curl_easy_setopt(3)*.

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_HTTPHEADER(3), CURLOPT_PROXYHEADER(3)