

NAME

CURLOPT_REQUEST_TARGET – alternative target for this request

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_REQUEST_TARGET, string);
```

DESCRIPTION

Pass a char pointer to string which libcurl uses in the upcoming request instead of the path as extracted from the URL.

libcurl passes on the verbatim string in its request without any filter or other safe guards. That includes white space and control characters.

Using this option multiple times makes the last set string override the previous ones. Set it to NULL to disable its use again.

The application does not have to keep the string around after setting this option.

DEFAULT

NULL

PROTOCOLS

This functionality affects http only

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/*");
        curl_easy_setopt(curl, CURLOPT_CUSTOMREQUEST, "OPTIONS");

        /* issue an OPTIONS * request (no leading slash) */
        curl_easy_setopt(curl, CURLOPT_REQUEST_TARGET, "*");

        /* Perform the request */
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.55.0

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_CUSTOMREQUEST(3), CURLOPT_HTTPGET(3), CURLOPT_PATH_AS_IS(3),
CURLOPT_URL(3)**