

NAME

CURLOPT_STREAM_DEPENDS – stream this transfer depends on

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_STREAM_DEPENDS,  
                          CURL *dephandle);
```

DESCRIPTION

Pass a CURL pointer in *dephandle* to identify the stream within the same connection that this stream is depending upon. This option clears the exclusive bit and is mutually exclusive to the *CURLOPT_STREAM_DEPENDS_E(3)* option.

The spec says "Including a dependency expresses a preference to allocate resources to the identified stream rather than to the dependent stream."

This option can be set during transfer.

dephandle must not be the same as *handle*, that makes this function return an error. It must be another easy handle, and it also needs to be a handle of a transfer that is about to be sent over the same HTTP/2 connection for this option to have an actual effect.

DEFAULT

NULL

PROTOCOLS

This functionality affects http only

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    CURL *curl2 = curl_easy_init(); /* a second handle */
    if(curl) {
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/one");

        /* the second depends on the first */
        curl_easy_setopt(curl2, CURLOPT_URL, "https://example.com/two");
        curl_easy_setopt(curl2, CURLOPT_STREAM_DEPENDS, curl);

        /* then add both to a multi handle and transfer them */
    }
}
```

AVAILABILITY

Added in curl 7.46.0

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

CURLMOPT_PIPELINING(3), CURLOPT_HTTP_VERSION(3), CURLOPT_STREAM_DEPENDS_E(3), CURLOPT_STREAM_WEIGHT(3)