### NAME

CURLOPT\_TIMEOUT - set maximum time the request is allowed to take

### **SYNOPSIS**

#include <curl/curl.h>

CURLcode curl\_easy\_setopt(CURL \*handle, CURLOPT\_TIMEOUT, long timeout);

# DESCRIPTION

Pass a long as parameter containing *timeout* - the maximum time in seconds that you allow the libcurl transfer operation to take. Normally, name lookups can take a considerable time and limiting operations to less than a few minutes risk aborting perfectly normal operations. This option may cause libcurl to use the SIGALRM signal to timeout system calls.

In unix-like systems, this might cause signals to be used unless CURLOPT\_NOSIGNAL(3) is set.

If both *CURLOPT\_TIMEOUT(3)* and *CURLOPT\_TIMEOUT\_MS(3)* are set, the value set last will be used.

Since this puts a hard limit for how long time a request is allowed to take, it has limited use in dynamic use cases with varying transfer times. You are then advised to explore *CURLOPT\_LOW\_SPEED\_LIMIT(3)*, *CURLOPT\_LOW\_SPEED\_TIME(3)* or using *CURLOPT\_PROGRESSFUNCTION(3)* to implement your own timeout logic.

# DEFAULT

Default timeout is 0 (zero) which means it never times out during transfer.

### PROTOCOLS

All

# EXAMPLE

CURL \*curl = curl\_easy\_init(); if(curl) {

curl\_easy\_setopt(curl, CURLOPT\_URL, "http://example.com");

/\* complete within 20 seconds \*/
curl\_easy\_setopt(curl, CURLOPT\_TIMEOUT, 20L);

curl\_easy\_perform(curl);

#### }

# AVAILABILITY

Always

**RETURN VALUE** 

Returns CURLE\_OK

### **SEE ALSO**

CURLOPT\_TIMEOUT\_MS(3), LOPT\_LOW\_SPEED\_LIMIT(3), CURLOPT\_CONNECTTIMEOUT(3), CUR-