

NAME

CURLOPT_CONNECTTIMEOUT_MS – timeout for the connect phase

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_CONNECTTIMEOUT_MS,
                           long timeout);
```

DESCRIPTION

Pass a long. It sets the maximum time in milliseconds that you allow the connection phase to take. This timeout only limits the connection phase, it has no impact once libcurl has connected. The connection phase includes the name resolve (DNS) and all protocol handshakes and negotiations until there is an established connection with the remote side.

Set this option to zero to switch to the default built-in connection timeout – 300 seconds. See also the *CURLOPT_TIMEOUT_MS(3)* option.

CURLOPT_CONNECTTIMEOUT(3) is the same function but set in seconds.

If both *CURLOPT_CONNECTTIMEOUT(3)* and *CURLOPT_CONNECTTIMEOUT_MS(3)* are set, the value set last is used.

The connection timeout is included in the general all-covering *CURLOPT_TIMEOUT_MS(3)*:

With *CURLOPT_CONNECTTIMEOUT_MS(3)* set to 3000 and *CURLOPT_TIMEOUT_MS(3)* set to 5000, the operation can never last longer than 5000 milliseconds, and the connection phase cannot last longer than 3000 milliseconds.

With *CURLOPT_CONNECTTIMEOUT_MS(3)* set to 4000 and *CURLOPT_TIMEOUT_MS(3)* set to 2000, the operation can never last longer than 2000 milliseconds. Including the connection phase.

This option may cause libcurl to use the SIGALRM signal to timeout system calls on builds not using async DNS. In Unix-like systems, this might cause signals to be used unless *CURLOPT_NOSIGNAL(3)* is set.

DEFAULT

300000

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

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

        /* complete connection within 10000 milliseconds */
        curl_easy_setopt(curl, CURLOPT_CONNECTTIMEOUT_MS, 10000L);

        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.16.2

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

**CURLLOPT_LOW_SPEED_LIMIT(3), CURLLOPT_MAX_RECV_SPEED_LARGE(3), CUR-
LOPT_TIMEOUT_MS(3)**