

NAME

curl_multi_setopt – set options for a curl multi handle

SYNOPSIS

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

```
CURLMcode curl_multi_setopt(CURLM *multi, CURLMoption option, parameter);
```

DESCRIPTION

curl_multi_setopt(3) is used to tell a libcurl multi handle how to behave. By using the appropriate options to *curl_multi_setopt(3)*, you can change libcurl's behavior when using that multi handle. All options are set with the *option* followed by the *parameter*. That parameter can be a **long**, a **function pointer**, an **object pointer** or a **curl_off_t** type, depending on what the specific option expects. Read this manual carefully as bad input values may cause libcurl to behave badly. You can only set one option in each function call.

OPTIONS

CURLMOPT_CHUNK_LENGTH_PENALTY_SIZE

deprecated See *CURLMOPT_CHUNK_LENGTH_PENALTY_SIZE(3)*

CURLMOPT_CONTENT_LENGTH_PENALTY_SIZE

deprecated See *CURLMOPT_CONTENT_LENGTH_PENALTY_SIZE(3)*

CURLMOPT_MAXCONNECTS

Size of connection cache. See *CURLMOPT_MAXCONNECTS(3)*

CURLMOPT_MAX_CONCURRENT_STREAMS

Max concurrent streams for http2. See *CURLMOPT_MAX_CONCURRENT_STREAMS(3)*

CURLMOPT_MAX_HOST_CONNECTIONS

Max number of connections to a single host. See *CURLMOPT_MAX_HOST_CONNECTIONS(3)*

CURLMOPT_MAX_PIPELINE_LENGTH

deprecated. See *CURLMOPT_MAX_PIPELINE_LENGTH(3)*

CURLMOPT_MAX_TOTAL_CONNECTIONS

Max simultaneously open connections. See *CURLMOPT_MAX_TOTAL_CONNECTIONS(3)*

CURLMOPT_NETWORK_CHANGED

Signal that the network has changed. See *CURLMOPT_NETWORK_CHANGED(3)*

CURLMOPT_NOTIFYDATA

Custom pointer passed to the notify callback. See *CURLMOPT_NOTIFYDATA(3)*

CURLMOPT_NOTIFYFUNCTION

Callback that receives notifications. See *CURLMOPT_NOTIFYFUNCTION(3)*

CURLMOPT_PIPELINING

Enable HTTP multiplexing. See *CURLMOPT_PIPELINING(3)*

CURLMOPT_PIPELINING_SERVER_BL

deprecated. See *CURLMOPT_PIPELINING_SERVER_BL(3)*

CURLMOPT_PIPELINING_SITE_BL

deprecated. See *CURLMOPT_PIPELINING_SITE_BL(3)*

CURLMOPT_PUSHDATA

Pointer to pass to push callback. See *CURLMOPT_PUSHDATA(3)*

CURLMOPT_PUSHFUNCTION

Callback that approves or denies server pushes. See *CURLMOPT_PUSHFUNCTION(3)*

CURLMOPT_QUICK_EXIT

Enable a quicker cleanup of the multi handle. See *CURLMOPT_QUICK_EXIT(3)*

CURLMOPT_RESOLVE_THREADS_MAX

Max threads used for threaded DNS resolver. See *CURLMOPT_RESOLVE_THREADS_MAX(3)*

CURLMOPT_SOCKETDATA

Custom pointer passed to the socket callback. See *CURLMOPT_SOCKETDATA(3)*

CURLMOPT_SOCKETFUNCTION

Callback informed about what to wait for. See *CURLMOPT_SOCKETFUNCTION(3)*

CURLMOPT_TIMERDATA

Custom pointer to pass to timer callback. See *CURLMOPT_TIMERDATA(3)*

CURLMOPT_TIMERFUNCTION

Callback to receive timeout values. See *CURLMOPT_TIMERFUNCTION(3)*

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
#define MAX_PARALLEL 45

int main(void)
{
    CURLM *multi = curl_multi_init();

    /* Limit the amount of simultaneous connections curl should allow: */
    curl_multi_setopt(multi, CURLMOPT_MAXCONNECTS, (long)MAX_PARALLEL);
}
```

AVAILABILITY

Added in curl 7.15.4

RETURN VALUE

This function returns a CURLMcode indicating success or error.

CURLM_OK (0) means everything was OK, non-zero means an error occurred, see *libcurl-errors(3)*.

Note that it returns a CURLM_UNKNOWN_OPTION if you try setting an option that this version of libcurl does not know of.

SEE ALSO

curl_multi_cleanup(3), curl_multi_info_read(3), curl_multi_init(3), curl_multi_socket(3)