

NAME

CURLOPT_CLOSESOCKETFUNCTION – callback to socket close replacement

SYNOPSIS

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

```
int closesocket_callback(void *clientp, curl_socket_t item);
```

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_CLOSESOCKETFUNCTION,
                          closesocket_callback);
```

DESCRIPTION

Pass a pointer to your callback function, which should match the prototype shown above.

This callback function gets called by libcurl instead of the *close(3)* or *closesocket(3)* call when sockets are closed (not for any other file descriptors). This is pretty much the reverse to the *CURLOPT_OPENSOCKETFUNCTION(3)* option. Return 0 to signal success and 1 if there was an error.

The *clientp* pointer is set with *CURLOPT_CLOSESOCKETDATA(3)*. *item* is the socket libcurl wants to be closed.

Note that when using multi/share handles, your callback may get invoked even after the easy handle has been cleaned up. The callback and data is inherited by a new connection and that connection may live longer than the transfer itself in the multi/share handle's connection cache.

DEFAULT

Use the standard socket close function.

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
struct priv {
    void *custom;
};

static int closesocket(void *clientp, curl_socket_t item)
{
    struct priv *my = clientp;
    printf("our ptr: %p\n", my->custom);

    printf("libcurl wants to close %d now\n", (int)item);
    return 0;
}

int main(void)
{
    struct priv myown;
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        /* call this function to close sockets */
        curl_easy_setopt(curl, CURLOPT_CLOSESOCKETFUNCTION, closesocket);
        curl_easy_setopt(curl, CURLOPT_CLOSESOCKETDATA, &myown);

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

```
    }  
}
```

AVAILABILITY

Added in curl 7.21.7

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_CLOSESOCKETDATA(3), CURLOPT_OPENSOCKETFUNCTION(3)