

NAME

CURLOPT_PRIVATE – store a private pointer

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_PRIVATE, void *pointer);
```

DESCRIPTION

Pass a void * as parameter, pointing to data that should be associated with this curl handle. The pointer can subsequently be retrieved using *curl_easy_getinfo(3)* with the *CURLINFO_PRIVATE(3)* option. libcurl itself never does anything with this data.

DEFAULT

NULL

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

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

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

        /* store a pointer to our private struct */
        curl_easy_setopt(curl, CURLOPT_PRIVATE, &secrets);

        result = curl_easy_perform(curl);

        /* we can extract the private pointer again too */
        curl_easy_getinfo(curl, CURLINFO_PRIVATE, &extracted);

        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.10.3

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

CURLINFO_PRIVATE(3), *CURLOPT_STDERR(3)*, *CURLOPT_VERBOSE(3)*