

NAME

CURLINFO_NUM_CONNECTS – number of created connections

SYNOPSIS

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

```
CURLcode curl_easy_getinfo(CURL *handle, CURLINFO_NUM_CONNECTS, long *nump);
```

DESCRIPTION

Pass a pointer to a long to receive how many new connections libcurl had to create to achieve the previous transfer (only the successful connects are counted). Combined with *CURLINFO_REDIRECT_COUNT*(3) you are able to know how many times libcurl successfully reused existing connection(s) or not. See the connection options of *curl_easy_setopt*(3) to see how libcurl tries to make persistent connections to save time.

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");
        curl_easy_setopt(curl, CURLOPT_FOLLOWLOCATION, 1L);
        result = curl_easy_perform(curl);
        if(result == CURLE_OK) {
            long connects;
            result = curl_easy_getinfo(curl, CURLINFO_NUM_CONNECTS, &connects);
            if(result == CURLE_OK)
                printf("It needed %ld connects\n", connects);
        }
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.12.3

RETURN VALUE

curl_easy_getinfo(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

curl_easy_getinfo(3), *curl_easy_setopt*(3)