

NAME

CURLMOPT_PUSHDATA – pointer to pass to push callback

SYNOPSIS

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

```
CURLMcode curl_multi_setopt(CURLM *handle, CURLMOPT_PUSHDATA, void *pointer);
```

DESCRIPTION

Set a *pointer* to pass as the last argument to the *CURLMOPT_PUSHFUNCTION(3)* callback. The pointer is not touched or used by libcurl itself, only passed on to the callback function.

DEFAULT

NULL

PROTOCOLS

This functionality affects http only

EXAMPLE

```
#include <string.h>

/* only allow pushes for filenames starting with "push-" */
int push_callback(CURL *parent,
                  CURL *easy,
                  size_t num_headers,
                  struct curl_pushheaders *headers,
                  void *clientp)
{
    char *headp;
    int *transfers = (int *)clientp;
    FILE *out;
    headp = curl_pushheader_byname(headers, ":path");
    if(headp && !strncmp(headp, "/push-", 6)) {
        fprintf(stderr, "The PATH is %s\n", headp);

        /* save the push here */
        out = fopen("pushed-stream", "wb");

        /* write to this file */
        curl_easy_setopt(easy, CURLOPT_WRITEDATA, out);

        (*transfers)++; /* one more */

        return CURL_PUSH_OK;
    }
    return CURL_PUSH_DENY;
}

int main(void)
{
    int counter;
    CURLM *multi = curl_multi_init();
    curl_multi_setopt(multi, CURLMOPT_PUSHFUNCTION, push_callback);
    curl_multi_setopt(multi, CURLMOPT_PUSHDATA, &counter);
}
```

AVAILABILITY

Added in curl 7.44.0

RETURN VALUE

curl_multi_setopt(3) returns a CURLMcode indicating success or error.

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

SEE ALSO

CURLMOPT_PIPELINING(3), CURLMOPT_PUSHFUNCTION(3), CURLOPT_PIPEWAIT(3), RFC7540