

NAME

CURLINFO_FTP_ENTRY_PATH – entry path in FTP server

SYNOPSIS

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

```
CURLcode curl_easy_getinfo(CURL *handle, CURLINFO_FTP_ENTRY_PATH, char **path);
```

DESCRIPTION

Pass a pointer to a char pointer to receive a pointer to a string holding the path of the entry path. That is the initial path libcurl ended up in when logging on to the remote FTP or SFTP server. This stores a NULL as pointer if something is wrong.

The **path** pointer is NULL or points to private memory. You **must not** free it. The memory gets freed automatically when you call *curl_easy_cleanup(3)* on the corresponding curl handle.

PROTOCOLS

This functionality affects ftp and sftp

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "ftp://example.com");

        result = curl_easy_perform(curl);

        if(result == CURLE_OK) {
            /* extract the entry path */
            char *ep = NULL;
            result = curl_easy_getinfo(curl, CURLINFO_FTP_ENTRY_PATH, &ep);
            if(!result && ep) {
                printf("Entry path was: %s\n", ep);
            }
        }
        curl_easy_cleanup(curl);
    }
}
```

HISTORY

Works for SFTP since 7.21.4

AVAILABILITY

Added in curl 7.15.4

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)**