

NAME

CURLOPT_WILDCARDMATCH – directory wildcard transfers

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_WILDCARDMATCH, long onoff);
```

DESCRIPTION

Set *onoff* to 1 if you want to transfer multiple files according to a filename pattern. The pattern can be specified as part of the *CURLOPT_URL(3)* option, using an **fnmatch**-like pattern (Shell Pattern Matching) in the last part of URL (filename).

By default, libcurl uses its internal wildcard matching implementation. You can provide your own matching function by the *CURLOPT_FNMATCH_FUNCTION(3)* option.

A brief introduction of its syntax follows:

*** - ASTERISK**

```
ftp://example.com/some/path/*.txt
```

matches all *.txt* files in the root directory. Only two asterisks are allowed within the same pattern string.

? - QUESTION MARK

Question mark matches any (exactly one) character.

```
ftp://example.com/some/path/photo?.jpg
```

[- BRACKET EXPRESSION

The left bracket opens a bracket expression. The question mark and asterisk have no special meaning in a bracket expression. Each bracket expression ends by the right bracket and matches exactly one character. Some examples follow:

[a-zA-Z0-9] or **[f-gF-G]** – character interval

[abc] – character enumeration

[^abc] or **[!abc]** – negation

[:name:] class expression. Supported classes are **alnum**, **lower**, **space**, **alpha**, **digit**, **print**, **upper**, **blank**, **graph**, **xdigit**.

[[-!^] – special case – matches only '-', '!', '^' or '~'. These characters have no special purpose.

[[]] – escape syntax. Matches '[', ']' or 'e'.

Using the rules above, a filename pattern can be constructed:

```
ftp://example.com/some/path/[a-z[:upper:]].jpg
```

PROTOCOLS

This functionality affects ftp only

EXAMPLE

```
extern long begin_cb(struct curl_fileinfo *, void *, int);
extern long end_cb(void *ptr);
```

```
int main(void)
```

```
{
    CURL *curl = curl_easy_init();
    if(curl) {
        /* turn on wildcard matching */
        curl_easy_setopt(curl, CURLOPT_WILDCARDMATCH, 1L);

        /* callback is called before download of concrete file started */
        curl_easy_setopt(curl, CURLOPT_CHUNK_BGN_FUNCTION, begin_cb);

        /* callback is called after data from the file have been transferred */
        curl_easy_setopt(curl, CURLOPT_CHUNK_END_FUNCTION, end_cb);

        /* See more on https://curl.se/libcurl/c/ftp-wildcard.html */
    }
}
```

AVAILABILITY

Added in curl 7.21.0

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_CHUNK_BGN_FUNCTION(3), CURLOPT_CHUNK_END_FUNCTION(3), CURLOPT_FNMATCH_FUNCTION(3), CURLOPT_URL(3)