

NAME

CURLOPT_FTPSSLAUTH – order in which to attempt TLS vs SSL

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_FTPSSLAUTH, long order);
```

DESCRIPTION

Pass a long using one of the values from below, to alter how libcurl issues "AUTH TLS" or "AUTH SSL" when FTP over SSL is activated. This is only interesting if *CURLOPT_USE_SSL(3)* is also set.

Possible *order* values:

CURLFTPAUTH_DEFAULT

Allow libcurl to decide.

CURLFTPAUTH_SSL

Try "AUTH SSL" first, and only if that fails try "AUTH TLS".

CURLFTPAUTH_TLS

Try "AUTH TLS" first, and only if that fails try "AUTH SSL".

DEFAULT

CURLFTPAUTH_DEFAULT

PROTOCOLS

This functionality affects ftp only

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "ftp://example.com/file.txt");
        curl_easy_setopt(curl, CURLOPT_USE_SSL, CURLUSESSL_TRY);
        /* funny server, ask for SSL before TLS */
        curl_easy_setopt(curl, CURLOPT_FTPSSLAUTH, CURLFTPAUTH_SSL);
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

HISTORY

CURLFTPAUTH_* enums became *long* types in 8.16.0, prior to this version a *long* cast was necessary when passed to *curl_easy_setopt(3)*.

AVAILABILITY

Added in curl 7.12.2

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_FTP_SSL_CCC(3), **CURLOPT_USE_SSL(3)**