

NAME

CURLOPT_REDIR_PROTOCOLS – protocols allowed to redirect to

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_REDIR_PROTOCOLS, long bitmask);
```

DESCRIPTION

This option is deprecated. We strongly recommend using *CURLOPT_REDIR_PROTOCOLS_STR(3)* instead because this option cannot control all available protocols.

Pass a long that holds a bitmask of protocol bits. If used, this bitmask limits what protocols libcurl may use in a transfer that it follows to in a redirect when *CURLOPT_FOLLOWLOCATION(3)* is enabled. This allows you to limit specific transfers to only be allowed to use a subset of protocols in redirections.

Protocols denied by *CURLOPT_PROTOCOLS(3)* are not overridden by this option.

By default libcurl allows HTTP, HTTPS, FTP and FTPS on redirect (7.65.2). *CURLPROTO_ALL* enables all protocols on redirect, including those otherwise disabled for security.

These are the available protocol defines:

```
CURLPROTO_DICT
CURLPROTO_FILE
CURLPROTO_FTP
CURLPROTO_FTPS
CURLPROTO_GOPHER
CURLPROTO_HTTP
CURLPROTO_HTTPS
CURLPROTO_IMAP
CURLPROTO_IMAPS
CURLPROTO_LDAP
CURLPROTO_LDAPS
CURLPROTO_MQTT
CURLPROTO_POP3
CURLPROTO_POP3S
CURLPROTO_RTMP
CURLPROTO_RTMPE
CURLPROTO_RTMP
CURLPROTO_RTMP
CURLPROTO_RTMPT
CURLPROTO_RTMPT
CURLPROTO_RTMPT
CURLPROTO_RTMPTS
CURLPROTO_RTSP
CURLPROTO_SCP
CURLPROTO_SFTP
CURLPROTO_SMB
CURLPROTO_SMBS
CURLPROTO_SMTP
CURLPROTO_SMTPS
CURLPROTO_TELNET
CURLPROTO_TFTP
```

DEFAULT

HTTP, HTTPS, FTP and FTPS

PROTOCOLS

This functionality affects http only

EXAMPLE

```
int main(int argc, char **argv)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        /* pass in the URL from an external source */
        curl_easy_setopt(curl, CURLOPT_URL, argv[1]);

        /* only allow redirects to HTTP and HTTPS URLs */
        curl_easy_setopt(curl, CURLOPT_REDIR_PROTOCOLS,
                        CURLPROTO_HTTP | CURLPROTO_HTTPS);

        /* Perform the request */
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

HISTORY

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

DEPRECATED

Deprecated since 7.85.0.

AVAILABILITY

Added in curl 7.19.4

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

CURLINFO_SCHEME(3), **CURLOPT_DEFAULT_PROTOCOL(3)**, **CURLOPT_PROTOCOLS(3)**, **CURLOPT_REDIR_PROTOCOLS_STR(3)**