

NAME

CURLOPT_PROXY – proxy to use

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_PROXY, char *proxy);
```

DESCRIPTION

Set the *proxy* to use for transfers with this easy handle. The parameter should be a char * to a null-terminated string holding the hostname or dotted numerical IP address. A numerical IPv6 address must be written within [brackets].

To specify port number in this string, append `:[port]` to the end of the host name. The proxy's port number may optionally (but discouraged) be specified with the separate option `CURLOPT_PROXYPORT(3)`. If not specified, libcurl defaults to using port 1080 for proxies.

The proxy string may be prefixed with `[scheme]://` to specify which kind of proxy is used.

Using this option multiple times makes the last set string override the previous ones. Set it to NULL to disable its use again.

The application does not have to keep the string around after setting this option.

`http://` HTTP Proxy. Default when no scheme or proxy type is specified.

`https://` HTTPS Proxy. (with OpenSSL, GnuTLS, mbedTLS, Rustls, Schannel or wolfSSL.)

This uses HTTP/1 by default. Setting `CURLOPT_PROXYTYPE(3)` to **CURLPROXY_HTTPS2** allows libcurl to negotiate using HTTP/2 with proxy.

`socks4://`

SOCKS4 Proxy.

`socks4a://`

SOCKS4a Proxy. Proxy resolves URL hostname.

`socks5://`

SOCKS5 Proxy.

`socks5h://`

SOCKS5 Proxy. Proxy resolves URL hostname.

Without a scheme prefix, `CURLOPT_PROXYTYPE(3)` can be used to specify which kind of proxy the string identifies.

When you tell the library to use an HTTP proxy, libcurl transparently converts operations to HTTP even if you specify an FTP URL etc. This may have an impact on what other features of the library you can use, such as `CURLOPT_QUOTE(3)` and similar FTP specifics that do not work unless you tunnel through the HTTP proxy. Such tunneling is activated with `CURLOPT_HTTPPROXYTUNNEL(3)`.

Setting the proxy string to "" (an empty string) explicitly disables the use of a proxy, even if there is an environment variable set for it.

Unix domain sockets are supported for SOCKS proxies. Set *localhost* for the host part and append the absolute path to the domain socket. For example: `socks5h://localhost/path/to/socket.sock`

When you set a hostname to use, do not assume that there is any particular single port number used widely for proxies. Specify it.

When a proxy is used, the active FTP mode as set with *CURLOPT_FTPPORT(3)*, cannot be used.

Doing FTP over an HTTP proxy without *CURLOPT_HTTPPROXYTUNNEL(3)* set makes libcurl do HTTP with an FTP URL over the proxy. For such transfers, common FTP specific options do not work, for example *CURLOPT_USE_SSL(3)*.

Authentication

The proxy can also be specified with its associated credentials like for ordinary URLs in the style: *scheme://username:password@hostname*

Alternatively, set them using *CURLOPT_PROXYUSERNAME(3)* and *CURLOPT_PROXYPASSWORD(3)*.

Environment variables

libcurl respects the proxy environment variables named **http_proxy**, **ftp_proxy**, **sftp_proxy** etc. If set, libcurl uses the specified proxy for that URL scheme. For an "FTP://" URL, the **ftp_proxy** is considered. **all_proxy** is used if no protocol specific proxy was set.

If **no_proxy** (or **NO_PROXY**) is set, it is the exact equivalent of setting the *CURLOPT_NOPROXY(3)* option.

The *CURLOPT_PROXY(3)* and *CURLOPT_NOPROXY(3)* options override environment variables.

DEFAULT

NULL

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/file.txt");
        curl_easy_setopt(curl, CURLOPT_PROXY, "http://proxy.example:80");
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

HISTORY

Since 7.14.1 the proxy environment variable names can include the protocol scheme.

Since 7.21.7 the proxy string supports the socks protocols as "schemes".

Since 7.50.2, unsupported schemes in proxy strings cause libcurl to return error.

Since 7.52.0, it supports HTTPS proxy for OpenSSL.

Since 7.87.0, it supports HTTPS proxy for GnuTLS, for mbedTLS, Rustls, Schannel and wolfSSL.

AVAILABILITY

Added in curl 7.1

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_HTTPPROXYTUNNEL(3), **CURLOPT_PRE_PROXY**(3), **CURLOPT_PROXY-**
PORT(3), **CURLOPT_PROXYTYPE**(3)