

NAME

CURLOPT_DEFAULT_PROTOCOL – default protocol to use if the URL is missing a scheme name

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_DEFAULT_PROTOCOL,  
                           char *protocol);
```

DESCRIPTION

This option tells libcurl to use *protocol* if the URL is missing a scheme name.

Use one of these protocol (scheme) names:

dict, file, ftp, ftps, gopher, http, https, imap, imaps, ldap, ldaps, pop3, pop3s, rtsp, scp, sftp, smb, smbs, smtp, smtps, telnet, tftp

An unknown or unsupported protocol causes error *CURLE_UNSUPPORTED_PROTOCOL* when libcurl parses a URL without a scheme. Parsing happens when *curl_easy_perform(3)* or *curl_multi_perform(3)* is called. The protocol set supported by libcurl vary depending on how it was built. Use *curl_version_info(3)* if you need a list of protocol names supported by the build of libcurl that you are using.

This option does not change the default proxy protocol (http).

Without this option libcurl would make a guess based on the host, see *CURLOPT_URL(3)* for details.

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

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

DEFAULT

NULL (make a guess based on the host)

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        /* set a URL without a scheme */
        curl_easy_setopt(curl, CURLOPT_URL, "example.com");

        /* set the default protocol (scheme) for schemeless URLs */
        curl_easy_setopt(curl, CURLOPT_DEFAULT_PROTOCOL, "https");

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

AVAILABILITY

Added in curl 7.45.0

RETURN VALUE

CURLE_OK if the option is supported.

CURLE_OUT_OF_MEMORY if there was insufficient heap space.

CURLE_UNKNOWN_OPTION if the option is not supported.

SEE ALSO

CURLINFO_PROTOCOL(3), CURLINFO_SCHEME(3), CURLOPT_URL(3)