

**NAME**

CURLOPT\_COOKIE – set contents of HTTP Cookie header

**SYNOPSIS**

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_COOKIE, char *cookie);
```

**DESCRIPTION**

Pass a pointer to a zero terminated string as parameter. It will be used to set a cookie in the HTTP request. The format of the string should be NAME=CONTENTS, where NAME is the cookie name and CONTENTS is what the cookie should contain.

If you need to set multiple cookies, set them all using a single option concatenated like this: "name1=content1; name2=content2;" etc.

This option sets the cookie header explicitly in the outgoing request(s). If multiple requests are done due to authentication, followed redirections or similar, they will all get this cookie passed on.

Using this option multiple times will only make the latest string override the previous ones.

This option will not enable the cookie engine. Use *CURLOPT\_COOKIEFILE(3)* or *CURLOPT\_COOKIEJAR(3)* to enable parsing and sending cookies automatically.

**DEFAULT**

NULL, no cookies

**PROTOCOLS**

HTTP

**EXAMPLE**

```
CURL *curl = curl_easy_init();
if(curl) {
    curl_easy_setopt(curl, CURLOPT_URL, "http://example.com");

    curl_easy_setopt(curl, CURLOPT_COOKIE, "tool=curl; fun=yes;");

    curl_easy_perform(curl);
}
```

**AVAILABILITY**

If HTTP is enabled

**RETURN VALUE**

Returns CURLE\_OK if HTTP is enabled, CURLE\_UNKNOWN\_OPTION if not, or CURLE\_OUT\_OF\_MEMORY if there was insufficient heap space.

**SEE ALSO**

**CURLOPT\_COOKIEFILE(3), CURLOPT\_COOKIEJAR(3), CURLOPT\_COOKIELIST(3), CURLOPT\_HTTPHEADER(3),**