

**NAME**

CURLOPT\_READDATA – custom pointer passed to the read callback

**SYNOPSIS**

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_READDATA, void *pointer);
```

**DESCRIPTION**

Data *pointer* to pass to the file read function. If you use the *CURLOPT\_READFUNCTION(3)* option, this is the pointer you'll get as input in the 4th argument to the callback.

If you don't specify a read callback but instead rely on the default internal read function, this data must be a valid readable FILE \* (cast to 'void \*').

If you're using libcurl as a win32 DLL, you **MUST** use a *CURLOPT\_READFUNCTION(3)* if you set this option.

**DEFAULT**

By default, this is a FILE \* to stdin.

**PROTOCOLS**

This is used for all protocols when sending data.

**EXAMPLE**

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

    /* pass pointer that gets passed in to the
       CURLOPT_READFUNCTION callback */
    curl_easy_setopt(curl, CURLOPT_READDATA, &this);

    curl_easy_perform(curl);
}
```

**AVAILABILITY**

This option was once known by the older name *CURLOPT\_INFILE*, the name *CURLOPT\_READDATA* was introduced in 7.9.7.

**RETURN VALUE**

This will return CURLE\_OK.

**SEE ALSO**

*CURLOPT\_READFUNCTION(3)*, *CURLOPT\_WRITEDATA(3)*,