

CURLOPT\_RESUME\_FROM\_LARGE(3) curl\_easy\_setopt options CURLOPT\_RESUME\_FROM\_LARGE(3)

## NAME

CURLOPT\_RESUME\_FROM\_LARGE – set a point to resume transfer from

## SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_RESUME_FROM_LARGE,  
                           curl_off_t from);
```

## DESCRIPTION

Pass a `curl_off_t` as parameter. It contains the offset in number of bytes that you want the transfer to start from. Set this option to 0 to make the transfer start from the beginning (effectively disabling resume). For FTP, set this option to -1 to make the transfer start from the end of the target file (useful to continue an interrupted upload).

When doing uploads with FTP, the resume position is where in the local/source file libcurl should try to resume the upload from and it will then append the source file to the remote target file.

## DEFAULT

0, not used

## PROTOCOLS

HTTP, FTP, SFTP, FILE

## EXAMPLE

```
CURL *curl = curl_easy_init();  
if(curl) {  
    curl_off_t resume_position = GET_IT_SOMEHOW;  
    curl_off_t file_size = GET_IT_SOMEHOW_AS_WELL;  
  
    curl_easy_setopt(curl, CURLOPT_URL, "ftp://example.com");  
  
    /* resuming upload at this position, possibly beyond 2GB */  
    curl_easy_setopt(curl, CURLOPT_RESUME_FROM_LARGE, resume_position);  
  
    /* ask for upload */  
    curl_easy_setopt(curl, CURLOPT_UPLOAD, 1L);  
  
    /* set total data amount to expect */  
    curl_easy_setopt(curl, CURLOPT_INFILESIZE_LARGE, file_size);  
  
    /* Perform the request */  
    curl_easy_perform(curl);  
}
```

## AVAILABILITY

Added in 7.11.0

## RETURN VALUE

Returns `CURLE_OK` if the option is supported, and `CURLE_UNKNOWN_OPTION` if not.

## SEE ALSO

`CURLOPT_RESUME_FROM(3)`, `CURLOPT_RANGE(3)`, `CURLOPT_INFILESIZE_LARGE(3)`,