NAME

CURLOPT_POSTFIELDSIZE_LARGE − size of POST data pointed to

SYNOPSIS

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

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_POSTFIELDSIZE_LARGE,
                          curl_off_t size);
```

DESCRIPTION

If you want to post data to the server without having libcurl do a strlen() to measure the data size, this option must be used. When this option is used you can post fully binary data, which otherwise is likely to fail. If this size is set to -1, the library will use strlen() to get the size.

DEFAULT

-1

PROTOCOLS

HTTP(S)

EXAMPLE

```
CURL *curl = curl_easy_init();
if(curl) {
  const char *data = large_chunk;
  curl_off_t length_of_data; /* set somehow */

  curl_easy_setopt(curl, CURLOPT_URL, "http://example.com");

  /* size of the POST data */
  curl_easy_setopt(curl, CURLOPT_POSTFIELDSIZE_LARGE, length_of_data);

  curl_easy_setopt(curl, CURLOPT_POSTFIELDS, data);

  curl_easy_perform(curl);
}
```

AVAILABILITY

Along with HTTP

RETURN VALUE

Returns CURLE_OK if HTTP is supported, and CURLE_UNKNOWN_OPTION if not.

SEE ALSO

CURLOPT_POSTFIELDS(3), **CURLOPT_COPYPOSTFIELDS**(3), **CURLOPT_POSTFIELDSIZE**(3),