

**NAME**

CURLOPT\_TFTP\_NO\_OPTIONS – Do not send TFTP options requests.

**SYNOPSIS**

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_TFTP_NO_OPTIONS, long onoff);
```

**DESCRIPTION**

Set *onoff* to 1L to exclude all TFTP options defined in RFC2347, RFC2348 and RFC2349 from read and write requests (RRQs/WRQs).

This option improves interop with some legacy servers that do not acknowledge or properly implement TFTP options. When this option is used *CURLOPT\_TFTP\_BLKSIZE(3)* is ignored.

**DEFAULT**

0

**PROTOCOLS**

TFTP

**EXAMPLE**

```
size_t write_callback(char *ptr, size_t size, size_t nmemb, void *fp)
{
    return fwrite(ptr, size, nmemb, (FILE *)fp);
}

CURL *curl = curl_easy_init();
if(curl) {
    FILE *fp = fopen("foo.bin", "wb");
    if(fp) {
        curl_easy_setopt(curl, CURLOPT_WRITEDATA, (void *)fp);
        curl_easy_setopt(curl, CURLOPT_WRITEFUNCTION, write_callback);

        curl_easy_setopt(curl, CURLOPT_URL, "tftp://example.com/foo.bin");

        /* do not send TFTP options requests */
        curl_easy_setopt(curl, CURLOPT_TFTP_NO_OPTIONS, 1L);

        /* Perform the request */
        curl_easy_perform(curl);

        fclose(fp);
    }
    curl_easy_cleanup(curl);
}
```

**AVAILABILITY**

Added in 7.48.0

**RETURN VALUE**

Returns CURLE\_OK if the option is supported, and CURLE\_UNKNOWN\_OPTION if not.