NAME

CURLOPT_PATH_AS_IS − do not handle dot−dot sequences

SYNOPSIS

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

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_PATH_AS_IS, long leaveit);
```

DESCRIPTION

Set the long *leaveit* to 1, to explicitly tell libcurl to not alter the given path before passing it on to the server.

This instructs libcurl to NOT squash sequences of "/../" or "/./" that may exist in the URL's path part and that is supposed to be removed according to RFC 3986 section 5.2.4.

Some server implementations are known to (erroneously) require the dot−dot sequences to remain in the path and some clients want to pass these on in order to try out server implementations.

By default libcurl normalizes such sequences before using the path.

This is a request for the *first* request libcurl issues. When following redirects, it may no longer apply.

The corresponding flag for the *curl_url_set(3)* function is called **CURLU_PATH_AS_IS**.

DEFAULT

0

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
  CURL *curl = curl_easy_init();
  if(curl) {
    CURLcode result;
    curl_easy_setopt(curl, CURLOPT_URL,
                     "https://example.com/../../etc/password");

    curl_easy_setopt(curl, CURLOPT_PATH_AS_IS, 1L);

    result = curl_easy_perform(curl);
    curl_easy_cleanup(curl);
  }
}
```

AVAILABILITY

Added in curl 7.42.0

RETURN VALUE

curl_easy_setopt(3) returns a CURLcode indicating success or error.

CURLE_OK (0) means everything was OK, non−zero means an error occurred, see *libcurl−errors(3)*.

SEE ALSO

CURLOPT_DEBUGFUNCTION(3), **CURLOPT_STDERR**(3), **CURLOPT_URL**(3), **curl_url_set**(3)