

NAME

CURLOPT_HSTS_CTRL – control HSTS behavior

SYNOPSIS

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

```
#define CURLHSTS_ENABLE    (1L << 0)
#define CURLHSTS_READONLYFILE (1L << 1)
```

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_HSTS_CTRL, long bitmask);
```

DESCRIPTION

HSTS (HTTP Strict Transport Security) means that an HTTPS server can instruct the client to not contact it again over clear-text HTTP for a certain period into the future. libcurl then automatically redirects HTTP attempts to such hosts to instead use HTTPS. This is done by libcurl retaining this knowledge in an in-memory cache.

Populate the long *bitmask* with the correct set of features to instruct libcurl how to handle HSTS for the transfers using this handle.

BITS

CURLHSTS_ENABLE

Enable the in-memory HSTS cache for this handle.

CURLHSTS_READONLYFILE

Make the HSTS file (if specified) read-only – makes libcurl not save the cache to the file when closing the handle.

DEFAULT

0

PROTOCOLS

This functionality affects http only

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        /* enable HSTS */
        curl_easy_setopt(curl, CURLOPT_HSTS_CTRL, CURLHSTS_ENABLE);
        /* specify where to store the HSTS cache */
        curl_easy_setopt(curl, CURLOPT_HSTS, "/home/user/.hsts-cache");
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.74.0

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_ALTSVC(3), CURLOPT_CONNECT_TO(3), CURLOPT_HSTS(3), CURLOPT_RESOLVE(3)