

NAME

CURLOPT_HSTS – HSTS cache filename

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_HSTS, char *filename);
```

DESCRIPTION

Make the *filename* point to a filename to load an existing HSTS cache from, and to store the cache in when the easy handle is closed. Setting a filename with this option also enables HSTS for this handle (the equivalent of setting *CURLHSTS_ENABLE* with *CURLOPT_HSTS_CTRL(3)*).

If the given file does not exist or contains no HSTS entries at startup, the HSTS cache starts empty. Setting the filename to NULL allows HSTS without reading from or writing to any file. NULL also makes libcurl clear the list of files to read HSTS data from, if any such were previously set.

If this option is set multiple times, libcurl loads cache entries from each given file but only stores the last used name for later writing.

Since libcurl 8.20.0, each in-memory HSTS cache (per easy handle or shared cache) holds no more than the most recently added 10,000 HSTS hostnames.

FILE FORMAT

The HSTS cache is saved to and loaded from a text file with one entry per physical line. Each line in the file has the following format:

```
[host] [stamp]
```

[host] is the domain name for the entry and the name is dot-prefixed if it is an entry valid for all subdomains to the name as well or only for the exact name.

[stamp] is the time (in UTC) when the entry expires and it uses the format "YYYYMMDD HH:MM:SS".

Lines starting with "#" are treated as comments and are ignored. There is currently no length or size limit.

DEFAULT

NULL, no filename

SECURITY CONCERNS

We strongly urge users to stick to *HTTPS://* URLs, which makes this option unnecessary.

libcurl cannot fully protect against attacks where an attacker has write access to the same directory where it is directed to save files. This is particularly sensitive if you save files using elevated privileges.

libcurl creates the file to store HSTS data in using default file permissions, meaning that on *nix systems you may need to restrict your umask to prevent other users on the same system to access the file.

PROTOCOLS

This functionality affects http only

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com");
        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_HSTS_CTRL(3), CURLOPT_RESOLVE(3)