

NAME

CURLOPT_SSH_KNOWNHOSTS – filename holding the SSH known hosts

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_SSH_KNOWNHOSTS, char *fname);
```

DESCRIPTION

Pass a pointer to a null-terminated string holding the filename of the known_host file to use. The known_hosts file should use the OpenSSH file format. If this file is specified, libcurl only accepts connections with hosts that are known and present in that file, with a matching public key. Use *CURLOPT_SSH_KEYFUNCTION(3)* to alter the default behavior on host and key matches and mismatches.

We strongly suggest users doing SCP or SFTP transfers to set this option to make sure that the network communication is done with the intended server and not an impostor.

The application does not have to keep the string around after setting this option.

Using this option multiple times makes the last set string override the previous ones. Set it to NULL to disable its use again.

This option is only consulted when libcurl establishes a new connection. Once a connection has been created and its host key verified against the known hosts file, it is deemed vetted and may be reused by libcurl without re-running the known hosts check, even if you later change SSH host verification options (including setting this option to NULL or using *CURLOPT_SSH_HOST_PUBLIC_KEY_SHA256(3)* or *CURLOPT_SSH_HOST_PUBLIC_KEY_MD5(3)*). Such changes only affect subsequently created connections; existing cached connections can continue to be reused with the verification that was in effect when they were first established. If you need to force re-verification with the new settings, use *CURLOPT_FRESH_CONNECT(3)* or *CURLOPT_FORBID_REUSE(3)* to avoid reusing the old connection.

DEFAULT

NULL

PROTOCOLS

This functionality affects scp and sftp

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "sftp://example.com/file");
        curl_easy_setopt(curl, CURLOPT_SSH_KNOWNHOSTS,
                        "/home/clarkkent/.ssh/known_hosts");
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.19.6

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_SSH_AUTH_TYPES(3), **CURLOPT_SSH_HOST_PUBLIC_KEY_MD5**(3), **CURLOPT_SSH_HOST_PUBLIC_KEY_SHA256**(3)