NAME

CURLOPT_SSL_VERIFYHOST − verify the certificate's name against host

SYNOPSIS

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

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_SSL_VERIFYHOST, long verify);
```

DESCRIPTION

Pass a long as parameter specifying what to *verify*.

This option determines whether libcurl verifies that the server cert is for the server it is known as.

When negotiating TLS and SSL connections, the server sends a certificate indicating its identity.

When *CURLOPT_SSL_VERIFYHOST(3)* is 2, that certificate must indicate that the server is the server to which you meant to connect, or the connection fails. Simply put, it means it has to have the same name in the certificate as is in the URL you operate against.

Curl considers the server the intended one when the Common Name field or a Subject Alternate Name field in the certificate matches the host name in the URL to which you told Curl to connect.

When the *verify* value is 1, *curl_easy_setopt* will return an error and the option value will not be changed. It was previously (in 7.28.0 and earlier) a debug option of some sorts, but it is no longer supported due to frequently leading to programmer mistakes. Future versions will stop returning an error for 1 and just treat 1 and 2 the same.

When the *verify* value is 0, the connection succeeds regardless of the names in the certificate. Use that ability with caution!

The default value for this option is 2.

This option controls checking the server's certificate's claimed identity. The server could be lying. To control lying, see *CURLOPT_SSL_VERIFYPEER(3)*. If libcurl is built against NSS and *CURLOPT_SSL_VERIFYPEER(3)* is zero, *CURLOPT_SSL_VERIFYHOST(3)* is also set to zero and cannot be overridden.

DEFAULT

2

PROTOCOLS

All TLS based protocols: HTTPS, FTPS, IMAPS, POP3, SMTPS etc.

EXAMPLE

```
CURL *curl = curl_easy_init();
if(curl) {
  curl_easy_setopt(curl, CURLOPT_URL, "https://example.com");

  /* Set the default value: strict name check please */
  curl_easy_setopt(curl, CURLOPT_SSL_VERIFYHOST, 2L);

  curl_easy_perform(curl);
}
```

AVAILABILITY

If built TLS enabled.

RETURN VALUE

Returns CURLE_OK if TLS is supported, and CURLE_UNKNOWN_OPTION if not.

If 1 is set as argument, *CURLE_BAD_FUNCTION_ARGUMENT* is returned.

SEE ALSO

CURLOPT_SSL_VERIFYPEER(3), **CURLOPT_CAINFO**(3),