

NAME

CURLOPT_SSL_VERIFYSTATUS – verify the certificate's status

SYNOPSIS

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

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

DESCRIPTION

Pass a long as parameter set to 1 to enable or 0 to disable.

This option determines whether libcurl verifies the status of the server cert using the "Certificate Status Request" TLS extension (aka. OCSP stapling).

Note that if this option is enabled but the server does not support the TLS extension, the verification fails.

DEFAULT

0

PROTOCOLS

This functionality affects all TLS based protocols: HTTPS, FTPS, IMAPS, POP3S, SMTPS etc.

This option works only with the following TLS backends: GnuTLS and OpenSSL

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/");
        /* ask for OCSP stapling */
        curl_easy_setopt(curl, CURLOPT_SSL_VERIFYSTATUS, 1L);
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.41.0

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_CAINFO(3), CURLOPT_SSL_VERIFYHOST(3), CURLOPT_SSL_VERIFYPEER(3)