

**NAME**

CURLOPT\_SSLVERSION – set preferred TLS/SSL version

**SYNOPSIS**

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_SSLVERSION, long version);
```

**DESCRIPTION**

Pass a long as parameter to control which version of SSL/TLS to attempt to use.

Use one of the available defines for this purpose. The available options are:

CURL\_SSLVERSION\_DEFAULT

The default action. This will attempt to figure out the remote SSL protocol version.

CURL\_SSLVERSION\_TLSv1

TLSv1.x

CURL\_SSLVERSION\_SSLv2

SSLv2

CURL\_SSLVERSION\_SSLv3

SSLv3

CURL\_SSLVERSION\_TLSv1\_0

TLSv1.0 (Added in 7.34.0)

CURL\_SSLVERSION\_TLSv1\_1

TLSv1.1 (Added in 7.34.0)

CURL\_SSLVERSION\_TLSv1\_2

TLSv1.2 (Added in 7.34.0)

**DEFAULT**

CURL\_SSLVERSION\_DEFAULT

**PROTOCOLS**

All TLS based protocols: HTTPS, FTPS, IMAPS, POP3S, SMTPS etc.

**EXAMPLE**

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

    /* ask libcurl to use TLS version 1.0 or later */
    curl_easy_setopt(curl, CURLOPT_SSLVERSION, CURL_SSLVERSION_TLSv1);

    /* Perform the request */
    curl_easy_perform(curl);
}
```

**AVAILABILITY**

SSLv2 is disabled by default since 7.18.1. Other SSL versions availability may vary depending on which backend libcurl has been built to use.

SSLv3 is disabled by default since 7.39.0.

**RETURN VALUE**

Returns CURLE\_OK if the option is supported, and CURLE\_UNKNOWN\_OPTION if not.

**SEE ALSO****CURLOPT\_USE\_SSL(3), CURLOPT\_HTTP\_VERSION(3), CURLOPT\_IPRESOLVE(3)**