

NAME

CURLOPT_CRLFILE – Certificate Revocation List file

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_CRLFILE, char *file);
```

DESCRIPTION

Pass a char pointer to a null-terminated string naming a *file* with the concatenation of CRL (in PEM format) to use in the certificate validation that occurs during the SSL exchange.

When curl is built to use GnuTLS, there is no way to influence the use of CRL passed to help in the verification process.

When libcurl is built with OpenSSL support, X509_V_FLAG_CRL_CHECK and X509_V_FLAG_CRL_CHECK_ALL are both set, requiring CRL check against all the elements of the certificate chain if a CRL file is passed. Also note that *CURLOPT_CRLFILE(3)* implies **CURLSSLOPT_NO_PARTIALCHAIN** (see *CURLOPT_SSL_OPTIONS(3)*) since curl 7.71.0 due to an OpenSSL bug.

This option makes sense only when used in combination with the *CURLOPT_SSL_VERIFYPEER(3)* option.

A specific error code (*CURLE_SSL_CRL_BADFILE*) is defined with the option. It is returned when the SSL exchange fails because the CRL file cannot be loaded. A failure in certificate verification due to a revocation information found in the CRL does not trigger this specific error.

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.

DEFAULT

NULL

PROTOCOLS

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

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

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_CRLFILE, "/etc/certs/crl.pem");
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.19.0

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_PROXY_CRLFILE(3), CURLOPT_SSL_VERIFYHOST(3), CURLOPT_SSL_VERIFYPEER(3)