

NAME

CURLOPT_PROXY_SSLCERT_BLOB – SSL proxy client certificate from memory blob

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_PROXY_SSLCERT_BLOB,
                          struct curl_blob *blob);
```

DESCRIPTION

Pass a pointer to a `curl_blob` structure, which contains information (pointer and size) about a memory block with binary data of the certificate used to connect to the HTTPS proxy. The format must be "P12" on Schannel. The format must be "P12" or "PEM" on OpenSSL. The string "P12" or "PEM" must be specified with `CURLOPT_PROXY_SSLCERTTYPE(3)`.

If the blob is initialized with the `flags` member of `struct curl_blob` set to `CURL_BLOB_COPY`, the application does not have to keep the buffer around after setting this.

This option is an alternative to `CURLOPT_PROXY_SSLCERT(3)` which instead expects a filename as input.

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: OpenSSL and Schannel

EXAMPLE

```
extern char *certificateData; /* point to data */
extern size_t filesize; /* size of the data */

int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        struct curl_blob blob;
        blob.data = certificateData;
        blob.len = filesize;
        blob.flags = CURL_BLOB_COPY;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/");
        curl_easy_setopt(curl, CURLOPT_PROXY, "https://proxy.example");
        curl_easy_setopt(curl, CURLOPT_PROXY_SSLKEY, "key.pem");
        curl_easy_setopt(curl, CURLOPT_PROXY_KEYPASSWD, "s3cret");
        curl_easy_setopt(curl, CURLOPT_PROXY_SSLCERT_BLOB, &blob);
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.71.0

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_SSLCERT(3), CURLOPT_PROXY_SSLCERTTYPE(3), CUR-
LOPT_PROXY_SSLKEY(3), CURLOPT_SSLCERT_BLOB(3)