

NAME

CURLOPT_SSLKEY_BLOB – private key for client cert from memory blob

SYNOPSIS

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

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

DESCRIPTION

Pass a pointer to a `curl_blob` structure, which contains information (pointer and size) for a private key. Compatible with OpenSSL. The format (like "PEM") must be specified with `CURLOPT_SSLKEYTYPE(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_SSLKEY(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 wolfSSL

EXAMPLE

```
extern char *certificateData; /* point to cert */  
extern size_t filesize; /* size of cert */  
  
extern char *privateKeyData; /* point to key */  
extern size_t privateKeySize; /* size of key */  
  
int main(void)  
{  
    CURL *curl = curl_easy_init();  
    if(curl) {  
        CURLcode result;  
        struct curl_blob blob;  
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/");  
        blob.data = certificateData;  
        blob.len = filesize;  
        blob.flags = CURL_BLOB_COPY;  
        curl_easy_setopt(curl, CURLOPT_SSLCERT_BLOB, &blob);  
        curl_easy_setopt(curl, CURLOPT_SSLCERTTYPE, "PEM");  
  
        blob.data = privateKeyData;  
        blob.len = privateKeySize;  
        curl_easy_setopt(curl, CURLOPT_SSLKEY_BLOB, &blob);  
        curl_easy_setopt(curl, CURLOPT_KEYPASSWD, "s3cret");  
        curl_easy_setopt(curl, CURLOPT_SSLKEYTYPE, "PEM");  
        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_SSLKEY(3), **CURLOPT_SSLKEYTYPE(3)**