

NAME

CURLOPT_SSLKEYTYPE – type of the private key file

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_SSLKEYTYPE, char *type);
```

DESCRIPTION

Pass a pointer to a null-terminated string as parameter. The string should be the format of your private key. Supported formats are "PEM", "DER", "ENG" and "PROV".

The format "ENG" enables you to load the private key from a crypto engine. In this case *CURLOPT_SSLKEY(3)* is used as an identifier passed to the engine. You have to set the crypto engine with *CURLOPT_SSLENGINE(3)*.

The format "PROV" enables you to load the private key from a crypto provider (Added in 8.12.0). In this case *CURLOPT_SSLKEY(3)* is used as an identifier passed to the provider.

The "DER" format does not work with OpenSSL.

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 restore to internal default.

DEFAULT

"PEM"

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

```
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_SSLCERT, "client.pem");
        curl_easy_setopt(curl, CURLOPT_SSLKEY, "key.pem");
        curl_easy_setopt(curl, CURLOPT_SSLKEYTYPE, "PEM");
        curl_easy_setopt(curl, CURLOPT_KEYPASSWD, "s3cret");
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.9.3

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_SSLKEYTYPE(3), **CURLOPT_SSLCERT**(3), **CURLOPT_SSLKEY**(3)