

NAME

CURLINFO_TLS_SESSION – get TLS session info

SYNOPSIS

#include <curl/curl.h>

```
CURLcode curl_easy_getinfo(CURL *handle, CURLINFO_TLS_SESSION,
                           struct curl_tlssessioninfo **session);
```

DESCRIPTION

Pass a pointer to a 'struct curl_tlssessioninfo *'. The pointer will be initialized to refer to a 'struct curl_tlssessioninfo *' that will contain an enum indicating the SSL library used for the handshake and the respective internal TLS session structure of this underlying SSL library.

This may then be used to extract certificate information in a format convenient for further processing, such as manual validation. NOTE: this option may not be available for all SSL backends; unsupported SSL backends will always return NULL in the *internals* pointer to indicate that they are not supported.

```
struct curl_tlssessioninfo {
    curl_sslbackend backend;
    void *internals;
};
```

The *backend* struct member is one of the defines in the CURLSSLBACKEND_* series: CURLSSLBACKEND_NONE (when built without TLS support), CURLSSLBACKEND_OPENSSL, CURLSSLBACKEND_GNUTLS, CURLSSLBACKEND_NSS, CURLSSLBACKEND_GSKIT, CURLSSLBACKEND_POLARSSL, CURLSSLBACKEND_CYASSL, CURLSSLBACKEND_SCHANNEL, CURLSSLBACKEND_DARWINSSL or CURLSSLBACKEND_AXTLS. (Note that the OpenSSL forks are all reported as just OpenSSL here.)

The *internals* struct member will point to a TLS library specific pointer with the following underlying types:

OpenSSL	SSL_CTX *
GnuTLS	gnutls_session_t
NSS	PRFileDesc *
gskit	gsk_handle

PROTOCOLS

All TLS-based

EXAMPLE

TODO

AVAILABILITY

Added in 7.34.0

RETURN VALUE

Returns CURLE_OK if the option is supported, and CURLE_UNKNOWN_OPTION if not.

SEE ALSO

curl_easy_getinfo(3), curl_easy_setopt(3),