NAME

SSL_in_before, SSL_in_init, SSL_is_init_finished, SSL_in_connect_init, SSL_in_accept_init, SSL_get_state - retrieve information about the handshake state machine

SYNOPSIS

```
#include <openssl/ssl.h>

int SSL_in_init(const SSL *s);
int SSL_in_before(const SSL *s);
int SSL_is_init_finished(const SSL *s);

int SSL_in_connect_init(SSL *s);
int SSL_in_accept_init(SSL *s);

OSSL_HANDSHAKE_STATE SSL_get_state(const SSL *ssl);
```

DESCRIPTION

SSL_in_init() returns 1 if the SSL/TLS state machine is currently processing or awaiting handshake messages, or 0 otherwise.

SSL_in_before() returns 1 if no SSL/TLS handshake has yet been initiated, or 0 otherwise.

SSL_is_init_finished() returns 1 if the SSL/TLS connection is in a state where fully protected application data can be transferred or 0 otherwise.

Note that in some circumstances (such as when early data is being transferred) **SSL_in_init()**, **SSL_in_before()** and **SSL_is_init_finished()** can all return 0.

SSL_in_connect_init() returns 1 if **s** is acting as a client and **SSL_in_init()** would return 1, or 0 otherwise.

SSL_in_accept_init() returns 1 if **s** is acting as a server and **SSL_in_init()** would return 1, or 0 otherwise.

SSL_in_connect_init() and **SSL_in_accept_init()** are implemented as macros.

SSL_get_state() returns a value indicating the current state of the handshake state machine. OSSL_HANDSHAKE_STATE is an enumerated type where each value indicates a discrete state machine state. Note that future versions of OpenSSL may define more states so applications should expect to receive unrecognised state values. The naming format is made up of a number of elements as

follows:

protocol_ST_**role_message**

protocol is one of TLS or DTLS. DTLS is used where a state is specific to the DTLS protocol. Otherwise TLS is used.

role is one of CR, CW, SR or SW to indicate "client reading", "client writing", "server reading" or "server writing" respectively.

message is the name of a handshake message that is being or has been sent, or is being or has been processed.

Additionally there are some special states that do not conform to the above format. These are:

TLS_ST_BEFORE
: No handshake messages have yet been been sent or received.

TLS_ST_OK
: Handshake message sending/processing has completed.

TLS_ST_EARLY_DATA
: Early data is being processed

TLS_ST_PENDING_EARLY_DATA_END
: Awaiting the end of early data processing

RETURN VALUES

SSL_in_init(), **SSL_in_before()**, **SSL_is_init_finished()**, **SSL_in_connect_init()** and **SSL_in_accept_init()** return values as indicated above.

SSL_get_state() returns the current handshake state.

SEE ALSO

ssl(7), **SSL_read_early_data**(3)

COPYRIGHT