



*Full credit is given to the above companies including the OS that this PDF file was generated!*

## ***Red Hat Enterprise Linux Release 9.2 Manual Pages on 'SSL\_want\_read.3ossl' command***

***\$ man SSL\_want\_read.3ossl***

SSL\_WANT(3ossl)            OpenSSL            SSL\_WANT(3ossl)

### NAME

SSL\_want, SSL\_want\_nothing, SSL\_want\_read, SSL\_want\_write,  
SSL\_want\_x509\_lookup, SSL\_want\_retry\_verify, SSL\_want\_async,  
SSL\_want\_async\_job, SSL\_want\_client\_hello\_cb - obtain state information  
TLS/SSL I/O operation

### SYNOPSIS

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

int SSL_want(const SSL *ssl);
int SSL_want_nothing(const SSL *ssl);
int SSL_want_read(const SSL *ssl);
int SSL_want_write(const SSL *ssl);
int SSL_want_x509_lookup(const SSL *ssl);
int SSL_want_retry_verify(const SSL *ssl);
int SSL_want_async(const SSL *ssl);
int SSL_want_async_job(const SSL *ssl);
int SSL_want_client_hello_cb(const SSL *ssl);
```

### DESCRIPTION

SSL\_want() returns state information for the SSL object ssl.

The other `SSL_want_*`() calls are shortcuts for the possible states returned by `SSL_want()`.

## NOTES

`SSL_want()` examines the internal state information of the SSL object. Its return values are similar to that of `SSL_get_error(3)`. Unlike `SSL_get_error(3)`, which also evaluates the error queue, the results are obtained by examining an internal state flag only. The information must therefore only be used for normal operation under nonblocking I/O. Error conditions are not handled and must be treated using `SSL_get_error(3)`.

The result returned by `SSL_want()` should always be consistent with the result of `SSL_get_error(3)`.

## RETURN VALUES

The following return values can currently occur for `SSL_want()`:

### SSL\_NOTHING

There is no data to be written or to be read.

### SSL\_WRITING

There are data in the SSL buffer that must be written to the underlying BIO layer in order to complete the actual `SSL_*`() operation. A call to `SSL_get_error(3)` should return `SSL_ERROR_WANT_WRITE`.

### SSL\_READING

More data must be read from the underlying BIO layer in order to complete the actual `SSL_*`() operation. A call to `SSL_get_error(3)` should return `SSL_ERROR_WANT_READ`.

## SSL\_X509\_LOOKUP

The operation did not complete because an application callback set by `SSL_CTX_set_client_cert_cb()` has asked to be called again. A call to `SSL_get_error(3)` should return `SSL_ERROR_WANT_X509_LOOKUP`.

## SSL\_RETRY\_VERIFY

The operation did not complete because a certificate verification callback has asked to be called again via `SSL_set_retry_verify(3)`.

A call to `SSL_get_error(3)` should return

`SSL_ERROR_WANT_RETRY_VERIFY`.

## SSL\_ASYNC\_PAUSED

An asynchronous operation partially completed and was then paused.

See `SSL_get_all_async_fds(3)`. A call to `SSL_get_error(3)` should return `SSL_ERROR_WANT_ASYNC`.

## SSL\_ASYNC\_NO\_JOBS

The asynchronous job could not be started because there were no async jobs available in the pool (see `ASYNC_init_thread(3)`). A call to `SSL_get_error(3)` should return `SSL_ERROR_WANT_ASYNC_JOB`.

## SSL\_CLIENT\_HELLO\_CB

The operation did not complete because an application callback set by `SSL_CTX_set_client_hello_cb()` has asked to be called again. A call to `SSL_get_error(3)` should return

`SSL_ERROR_WANT_CLIENT_HELLO_CB`.

`SSL_want_nothing()`, `SSL_want_read()`, `SSL_want_write()`, `SSL_want_x509_lookup()`, `SSL_want_retry_verify()`, `SSL_want_async()`, `SSL_want_async_job()`, and `SSL_want_client_hello_cb()` return 1 when the corresponding condition is true or 0 otherwise.

ssl(7), SSL\_get\_error(3)

## HISTORY

The SSL\_want\_client\_hello\_cb() function and the SSL\_CLIENT\_HELLO\_CB return value were added in OpenSSL 1.1.1.

## COPYRIGHT

Copyright 2001-2022 The OpenSSL Project Authors. All Rights Reserved.

Licensed under the Apache License 2.0 (the "License"). You may not use this file except in compliance with the License. You can obtain a copy in the file LICENSE in the source distribution or at <https://www.openssl.org/source/license.html>.

3.0.7                      2023-07-13                      SSL\_WANT(3ossl)