NAME

SSL_CTX_set_cert_verify_callback - set peer certificate verification procedure

SYNOPSIS

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

void SSL_CTX_set_cert_verify_callback(SSL_CTX *ctx, int (*callback)(X509_STORE_C
```

DESCRIPTION

SSL_CTX_set_cert_verify_callback() sets the verification callback function for *ctx*. SSL objects that are created from *ctx* inherit the setting valid at the time when *SSL_new(3)* is called.

NOTES

Whenever a certificate is verified during a SSL/TLS handshake, a verification function is called. If the application does not explicitly specify a verification callback function, the built-in verification function is used. If a verification callback *callback* is specified via *SSL_CTX_set_cert_verify_callback()*, the supplied callback function is called instead. By setting *callback* to NULL, the default behaviour is restored.

When the verification must be performed, *callback* will be called with the arguments callback(X509_STORE_CTX *x509_store_ctx, void *arg). The argument *arg* is specified by the application when setting *callback*.

callback should return 1 to indicate verification success and 0 to indicate verification failure. If SSL_VERIFY_PEER is set and *callback* returns 0, the handshake will fail. As the verification procedure may allow to continue the connection in case of failure (by always returning 1) the verification result must be set in any case using the **error** member of *x509_store_ctx* so that the calling application will be informed about the detailed result of the verification procedure!

Within *x509_store_ctx*, *callback* has access to the *verify_callback* function set using *SSL_CTX_set_verify(3)*.

WARNINGS

Do not mix the verification callback described in this function with the **verify_callback** function called during the verification process. The latter is set using the *SSL_CTX_set_verify(3)* family of functions.

Providing a complete verification procedure including certificate purpose settings etc is a complex task. The built-in procedure is quite powerful and in most cases it should be sufficient to modify its behaviour using the **verify_callback** function.

BUGS

SSL_CTX_set_cert_verify_callback() does not provide diagnostic information.

SEE ALSO

ssl(3), *SSL_CTX_set_verify(3)*, *SSL_get_verify_result(3)*, *SSL_CTX_load_verify_locations(3)*

COPYRIGHT