

**NAME**

SSL\_CTX\_set0\_verify\_cert\_store, SSL\_CTX\_set1\_verify\_cert\_store, SSL\_CTX\_set0\_chain\_cert\_store, SSL\_CTX\_set1\_chain\_cert\_store, SSL\_set0\_verify\_cert\_store, SSL\_set1\_verify\_cert\_store, SSL\_set0\_chain\_cert\_store, SSL\_set1\_chain\_cert\_store - set certificate verification or chain store

**SYNOPSIS**

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

int SSL_CTX_set0_verify_cert_store(SSL_CTX *ctx, X509_STORE *st);
int SSL_CTX_set1_verify_cert_store(SSL_CTX *ctx, X509_STORE *st);
int SSL_CTX_set0_chain_cert_store(SSL_CTX *ctx, X509_STORE *st);
int SSL_CTX_set1_chain_cert_store(SSL_CTX *ctx, X509_STORE *st);

int SSL_set0_verify_cert_store(SSL *ctx, X509_STORE *st);
int SSL_set1_verify_cert_store(SSL *ctx, X509_STORE *st);
int SSL_set0_chain_cert_store(SSL *ctx, X509_STORE *st);
int SSL_set1_chain_cert_store(SSL *ctx, X509_STORE *st);
```

**DESCRIPTION**

*SSL\_CTX\_set0\_verify\_cert\_store()* and *SSL\_CTX\_set1\_verify\_cert\_store()* set the certificate store used for certificate verification to **st**.

*SSL\_CTX\_set0\_chain\_cert\_store()* and *SSL\_CTX\_set1\_chain\_cert\_store()* set the certificate store used for certificate chain building to **st**.

*SSL\_set0\_verify\_cert\_store()*, *SSL\_set1\_verify\_cert\_store()*, *SSL\_set0\_chain\_cert\_store()* and *SSL\_set1\_chain\_cert\_store()* are similar except they apply to SSL structure **ssl**.

All these functions are implemented as macros. Those containing a **1** increment the reference count of the supplied store so it must be freed at some point after the operation. Those containing a **0** do not increment reference counts and the supplied store **MUST NOT** be freed after the operation.

**NOTES**

The stores pointers associated with an SSL\_CTX structure are copied to any SSL structures when *SSL\_new()* is called. As a result SSL structures will not be affected if the parent SSL\_CTX store pointer is set to a new value.

The verification store is used to verify the certificate chain sent by the peer: that is an SSL/TLS client will use the verification store to verify the server's certificate chain and a SSL/TLS server will use it to verify any client certificate chain.

The chain store is used to build the certificate chain.

If the mode **SSL\_MODE\_NO\_AUTO\_CHAIN** is set or a certificate chain is configured already (for example using the functions such as *SSL\_CTX\_add1\_chain\_cert(3)* or *SSL\_CTX\_add\_extra\_chain\_cert(3)*) then automatic chain building is disabled.

If the mode **SSL\_MODE\_NO\_AUTO\_CHAIN** is set then automatic chain building is disabled.

If the chain or the verification store is not set then the store associated with the parent SSL\_CTX is used instead to retain compatibility with previous versions of OpenSSL.

**RETURN VALUES**

All these functions return 1 for success and 0 for failure.

**SEE ALSO**

*SSL\_CTX\_add\_extra\_chain\_cert(3)*      *SSL\_CTX\_set0\_chain(3)*      *SSL\_CTX\_set1\_chain(3)*  
*SSL\_CTX\_add0\_chain\_cert(3)*   *SSL\_CTX\_add1\_chain\_cert(3)*   *SSL\_set0\_chain(3)*   *SSL\_set1\_chain(3)*  
*SSL\_add0\_chain\_cert(3)*      *SSL\_add1\_chain\_cert(3)*      *SSL\_CTX\_build\_cert\_chain(3)*  
*SSL\_build\_cert\_chain(3)*

## **HISTORY**

These functions were first added to OpenSSL 1.0.2.

## **COPYRIGHT**

Copyright 2013-2016 The OpenSSL Project Authors. All Rights Reserved.

Licensed under the OpenSSL license (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>.