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_CTX_set_msg_callback.3ossl' command

$ man SSL_CTX_set_msg_callback.3ossl

SSL_CTX_SET_MSG_CALLBACK(3ossl) OpenSSL SSL_CTX_SET_MSG_CALLBACK(3ossl)

NAME

SSL_CTX_set_msg_callback, SSL_CTX_set_msg_callback_arg,

SSL_set_msg_callback, SSL_set_msg_callback_arg - install callback for

observing protocol messages

SYNOPSIS

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

void SSL_CTX_set_msg_callback(SSL_CTX *ctx,
                              void (*cb)(int write_p, int version,
                                         int content_type, const void *buf,
                                         size_t len, SSL *ssl, void *arg));
void SSL_CTX_set_msg_callback_arg(SSL_CTX *ctx, void *arg);

void SSL_set_msg_callback(SSL *ssl,
                          void (*cb)(int write_p, int version,
                                     int content_type, const void *buf,
                                     size_t len, SSL *ssl, void *arg));
void SSL_set_msg_callback_arg(SSL *ssl, void *arg);
```

DESCRIPTION

SSL_CTX_set_msg_callback() or SSL_set_msg_callback() can be used to define a message callback function cb for observing all SSL/TLS protocol messages (such as handshake messages) that are received or sent, as well as other events that occur during processing. SSL_CTX_set_msg_callback_arg() and SSL_set_msg_callback_arg() can be used to set argument arg to the callback function, which is available for arbitrary application use.

SSL_CTX_set_msg_callback() and SSL_CTX_set_msg_callback_arg() specify default settings that will be copied to new SSL objects by SSL_new(3). SSL_set_msg_callback() and SSL_set_msg_callback_arg() modify the actual settings of an SSL object. Using a NULL pointer for cb disables the message callback.

When cb is called by the SSL/TLS library the function arguments have the following meaning:

write_p

This flag is 0 when a protocol message has been received and 1 when a protocol message has been sent.

version

The protocol version according to which the protocol message is interpreted by the library such as TLS1_3_VERSION, TLS1_2_VERSION etc. This is set to 0 for the SSL3_RT_HEADER pseudo content type (see NOTES below).

content_type

This is one of the content type values defined in the protocol specification (SSL3_RT_CHANGE_CIPHER_SPEC, SSL3_RT_ALERT, SSL3_RT_HANDSHAKE; but never SSL3_RT_APPLICATION_DATA because the callback will only be called for protocol messages). Alternatively it may be a "pseudo" content type. These pseudo content types are

used to signal some other event in the processing of data (see NOTES below).

buf, len

buf points to a buffer containing the protocol message or other data (in the case of pseudo content types), which consists of len bytes. The buffer is no longer valid after the callback function has returned.

ssl The SSL object that received or sent the message.

arg The user-defined argument optionally defined by SSL_CTX_set_msg_callback_arg() or SSL_set_msg_callback_arg().

NOTES

Protocol messages are passed to the callback function after decryption and fragment collection where applicable. (Thus record boundaries are not visible.)

If processing a received protocol message results in an error, the callback function may not be called. For example, the callback function will never see messages that are considered too large to be processed.

Due to automatic protocol version negotiation, version is not necessarily the protocol version used by the sender of the message: If a TLS 1.0 ClientHello message is received by an SSL 3.0-only server, version will be SSL3_VERSION.

Pseudo content type values may be sent at various points during the processing of data. The following pseudo content types are currently defined:

SSL3_RT_HEADER

Used when a record is sent or received. The buf contains the record header bytes only.

SSL3_RT_INNER_CONTENT_TYPE

Used when an encrypted TLSv1.3 record is sent or received. In encrypted TLSv1.3 records the content type in the record header is always SSL3_RT_APPLICATION_DATA. The real content type for the record is contained in an "inner" content type. buf contains the encoded "inner" content type byte.

RETURN VALUES

SSL_CTX_set_msg_callback(), SSL_CTX_set_msg_callback_arg(), SSL_set_msg_callback() and SSL_set_msg_callback_arg() do not return values.

SEE ALSO

ssl(7), SSL_new(3)

HISTORY

The pseudo content type SSL3_RT_INNER_CONTENT_TYPE was added in OpenSSL 1.1.1.

COPYRIGHT