

Network Working Group
INTERNET DRAFT

Expires December 22, 2001

M. Tuexen
Siemens AG
A. Jungmaier
University of Essen
June 22, 2001

TLS over SCTP
<[draft-tuexen-tsvwg-tls-over-sctp-00.txt](#)>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of \[RFC2026\]](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/lid-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

Abstract

This document describes the usage of the Transport Layer Security (TLS) protocol, as defined in [\[RFC2246\]](#), over the Stream Control Transmission Protocol (SCTP), as defined in [\[RFC2960\]](#).

The user of TLS can take advantage of the following features provided by SCTP:

- Support of multiple streams to avoid head of line blocking.
- Support of multi-homing to provide network level fault tolerance.
- Support of dynamic reconfiguration of IP-addresses.

[1.](#) Introduction

[1.1.](#) Overview

This document describes the usage of the Transport Layer Security (TLS) protocol, as defined in [[RFC2246](#)], over the Stream Control Transmission Protocol (SCTP), as defined in [[RFC2960](#)].

TLS is designed to run on top of a byte-stream oriented transport protocol providing a reliable, in-sequence delivery. Thus, TLS is currently mainly being used on top of the Transmission Control Protocol (TCP), as defined in [[RFC793](#)].

Comparing TCP and SCTP, the latter provides additional features and this document shows how TLS should be used with SCTP to provide some of these additional features to the TLS user.

This document defines

- how to use the multiple streams feature of SCTP.
- how to handle the message oriented nature of SCTP.

It should be noted that the TLS user can take advantage of the multi-homing support of SCTP. The dynamic reconfiguration of IP-addresses as described in [[SCTPEXT](#)] can also be used with the described solution.

The method described in this document does not require any changes of TLS or SCTP. It is only required that SCTP implementations support the optional feature of fragmentation of SCTP user messages.

[1.2.](#) Terminology

This document uses the following terms:

Association:
A SCTP association.

Connection:
A TLS connection.

Session:
A TLS session.

Stream:

An unidirectional stream of a SCTP association. It is uniquely identified by a stream identifier.

[1.3.](#) Abbreviations

MTU: Maximum Transmission Unit

SCTP: Stream Control Transmission Protocol

TCP: Transmission Control Protocol

TLS: Transport Layer Security

[2.](#) Conventions

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, NOT RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be interpreted as described in [[RFC2119](#)].

[3.](#) SCTP Requirements

[3.1.](#) Number of Inbound and Outbound Streams

When establishing associations used by TLS, the SCTP user MUST request an identical number of inbound and outbound streams from the SCTP layer. This rule makes sure that the association will have the same number of streams in both directions. A pair consisting of two streams with the same stream identifier is considered and used as one bi-directional stream.

Thus an SCTP association can be considered as a set of bi-directional streams.

[3.2.](#) Fragmentation of User Messages

To avoid the knowledge and handling of the MTU inside TLS, SCTP MUST provide fragmentation of user messages, which is an optional feature of [[RFC2960](#)]. Since SCTP is a message oriented protocol, it must be able to transmit all TLS records as SCTP user messages. Thus the supported

maximum length of SCTP user messages MUST be at least $2^{14} + 2048 + 5 = 18437$ bytes, which is the maximum length of a TLSCiphertext, as defined in [RFC2246].

Therefore, SCTP takes care of fragmenting and reassembling the TLS records in order to avoid IP-fragmentation.

[4.](#) Connections and Bi-directional Streams

TLS makes use of multiple bi-directional streams by establishing a connection over each bi-directional stream. This means that the number of connections for an association is limited by the number of bi-

directional streams.

The TLS handshake protocol is used on each bi-directional stream separately. Each handshake can be

- a full handshake or
- an abbreviated handshake that resumes a TLS session with a session id from another connection (on the same or another association).

After completing the handshake for a connection, the bi-directional stream can be used for TLS-based user data transmission. It should also be noted that the handshakes for the different connections are independent and can be delayed until the bi-directional stream is used for user data transmission.

[5.](#) Examples

In these examples we consider the case of an association with two bi-directional streams.

[5.1.](#) Two Bi-directional Streams with Full Handshake

Just after the association has been established the client sends two ClientHello messages on the bi-directional streams 0 and 1. After a full handshake has been completed on each bi-directional stream, TLS-based user data transmission can take place. It is possible that on the bi-directional stream 0 the handshake has been completed, and user data

transmission is ongoing, while on the bi-directional stream 1 the handshake has not been completed, or vice versa.

[5.2.](#) Two Bi-directional Streams with an Abbreviated Handshake

After establishing the association, the client starts a full handshake on the bi-directional stream 0. The server provides a session identifier which allows session resumption. After the full handshake has been completed, the client initiates an abbreviated handshake on the bi-directional stream 1 using the session identifier from the handshake on the bi-directional stream 0. User data can be transmitted on the bi-directional stream 0, but not on the bi-directional stream stream 1 in that state. After completion of the abbreviated handshake on the bi-directional stream 1, user data can be transmitted on both streams.

Whether or not to use abbreviated handshakes during the setup phase of a TLS connection over an SCTP association depends on several factors:

- the complexity and duration of the initial handshake processing (also determined by the number of connections),
- the network performance (round-trip times, bandwidth).

Abbreviated handshakes can reduce computational complexity of the handshake considerably, in case that this is a limiting resource. If a large number of connections need to be established, it may be of advantage to use the TLS session resumption feature. On the other hand, before an abbreviated handshakes can take place, a full handshake needs to have completed. In networks with large round-trip time delays, it may be favorable to perform a number of full handshakes in parallel. Therefore, both possibilities are allowed.

[5.3.](#) Two Bi-directional Streams with a Delayed Abbreviated Handshake

This example resembles the last one, but after the completion of the full handshake on the bi-directional stream 0, the abbreviated handshake on the bi-directional stream 1 is not started immediately. The bi-directional stream 0 can be used for user data transmission. It is only when the user also wants to transmit data on the bi-directional stream 1

that the abbreviated handshake for the bi-directional stream 1 is initiated.

This allows the user of TLS to request a large number of bi-directional streams without having to provide all the resources at association start-up if not all bi-directional streams are used right from the beginning.

[5.4.](#) Two Bi-directional Streams without Full Handshakes

This example is like the second or third one, but an abbreviated handshake is used for both bi-directional streams. This requires the existence of a valid session identifier from connections handled by another association.

[6.](#) Security Considerations

Using TLS on top of SCTP does not provide any new security issues beside the ones discussed in [[RFC2246](#)] and [[RFC2960](#)].

[7.](#) Acknowledgements

The authors would like to thank P. Calhoun, E. Rescorla, J. Wood and many others for their invaluable comments and suggestions.

[8.](#) References

- [SCTPEXT] R. R. Stewart, Q. Xie, M. Tuexen, I. Rytina, "SCTP Extensions for Dynamic Reconfiguration of IP Addresses and Enforcement of Flow and Message Limits", <[draft-ietf-tsvwg-addip-sctp-01.txt](#)>, February 2001.
- [RFC793] J. Postel (ed.), "Transmission Control Protocol", STP 7, [RFC 793](#), September 1981.
- [RFC2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2026] S. Bradner, "The Internet Standards Process -- Revision 3", [RFC 2026](#), October 1996.

- [RFC2246] T. Diercks, C. Allen, "The TLS Protocol Version 1.0", [RFC 2246](#), January 1999.
- [RFC2960] R. R. Stewart et al., "Stream Control Transmission Protocol", [RFC 2960](#), November 2000.

9. Authors' Addresses

Michael Tuexen
Siemens AG
ICN WN CS SE 5
D-81359 Munich
Germany

Tel.: +49 89 722 47210
e-mail: Michael.Tuexen@icn.siemens.de

Andreas Jungmaier
University of Essen
Networking Technology Group at the IEM
Ellernstrasse 29
D-45326 Essen
Germany

Tel.: +49 201 1837636
e-mail: ajung@exp-math.uni-essen.de

This Internet Draft expires December 22, 2001.