Network Working Group
Request for Comments: 3436
Category: Standards Track

A. Jungmaier
University of Essen
E. Rescorla
RTFM Inc.
M. Tuexen
Siemens AG

December 2002

Transport Layer Security over Stream Control Transmission Protocol

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

Abstract

This document describes the usage of the Transport Layer Security (TLS) protocol, as defined in RFC 2246, over the Stream Control Transmission Protocol (SCTP), as defined in RFC 2960 and RFC 3309.

The user of TLS can take advantage of the features provided by SCTP, namely the support of multiple streams to avoid head of line blocking and the support of multi-homing to provide network level fault tolerance.

Additionally, discussions of extensions of SCTP are also supported, meaning especially the 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] and [RFC3309].

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 currently being discussed, 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:
An SCTP association.

Connection:
A TLS connection.

Session:
A TLS session.

Stream:
A unidirectional stream of an 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", in this document are to be interpreted as described in BCP 14, RFC 2119 [RFC2119].

3. SCTP Requirements

3.1. Number of Inbound and Outbound Streams

An association between the endpoints A and Z provides n streams from A to Z and m streams from Z to A.

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 min(n,m) bi-directional streams and (max(n,m) - min(n,m)) uni-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].

Please note that an SCTP implementation might need to support the partial delivery API to be able to support the transport of user messages of this size.

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

4. TLS Requirements

4.1 Supported Ciphersuites

A TLS implementation for TLS over SCTP MUST support at least the ciphersuite TLS_RSA_WITH_AES_128_CBC_SHA as defined in [RFC3268].

5. Connections and Bi-directional Streams

TLS makes use of a bi-directional stream by establishing a connection over it. 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.

6. Usage of bi-directional streams

It is not required that all bi-directional streams are used for TLS-based user data transmission. If TLS is not used, it is called SCTP-based user data transmission.

6.1. SCTP-based user data transmission

If a bi-directional stream is not used for TLS-based communication there are no restrictions on the features provided by SCTP for SCTP-based user data transmission.

6.2. TLS-based user data transmission

In general, the bi-directional stream will be used for TLS-based user data transmission and it SHOULD NOT be used for SCTP-based user data transmission. The exception to this rule is for protocols which contain upgrade-to-TLS mechanisms, such as those of HTTP upgrade [RFC2817] and SMTP over TLS [RFC3207].

TLS requires that the underlying transport delivers TLS records in strict sequence. Thus, the 'unordered delivery' feature of SCTP MUST NOT be used on streams which are used for TLS based user data transmission. For the same reason, TLS records delivered to SCTP for transmission MUST NOT have limited lifetimes.

7. Usage of uni-directional streams

The uni-directional streams can not be used for TLS-based user data transmission. Nevertheless, they can be used without any restrictions for SCTP-based communication.

8. Examples

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

8.1. 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 on that stream. 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.

8.2. 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 this is a limiting resource. If a large number of connections need to be established, it may be advantageous to use the TLS session resumption feature. On the other hand, before an abbreviated handshake can take place, a full handshake needs to have been 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.

8.3. 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.

8.4. Two Bi-directional Streams without Full Handshakes

This example is like the second and 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.

9. Security Considerations

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

It is possible to authenticate TLS endpoints based on IP-addresses in certificates. Unlike TCP, SCTP associations can use multiple addresses per SCTP endpoint. Therefore it is possible that TLS records will be sent from a different IP-address than that originally authenticated. This is not a problem provided that no security decisions are made based on that IP-address. This is a special case of a general rule: all decisions should be based on the peer's authenticated identity, not on its transport layer identity.

10. Acknowledgements

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

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2246] Diercks, T. and C. Allen, "The TLS Protocol Version 1.0", RFC 2246, January 1999.

[RFC2960] Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L. and V. Paxon, "Stream Control Transmission Protocol", RFC 2960, October 2000.

[RFC3268] Chown, P., "Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)", RFC 3268, June 2002.

[RFC3309] Stone, J., Stewart, R., Otis, D., "Stream Control Transmission Protocol (SCTP) Checksum Change", RFC 3309, September 2002.

11.2. Informative References

[RFC793] Postel, J. (ed.), "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[RFC2026] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

[RFC2817] Khare, R. and S. Lawrence, "Upgrading to TLS Within HTTP/1.1", RFC 2817, May 2000.

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over TLS", RFC 3207, February 2002.

12. Authors' Addresses

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

Phone: +49 201 1837667
EMail: ajung@exp-math.uni-essen.de

Eric Rescorla
RTFM, Inc.
2064 Edgewood Drive
Palo Alto, CA 94303
USA

Phone: +1 650-320-8549
EMail: ekr@rtfm.com

Michael Tuexen
Siemens AG
D-81359 Munich
Germany

Phone: +49 89 722 47210
EMail: Michael.Tuexen@siemens.com

13. Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.