Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 26, 2010

M. Tuexen
R. Seggelmann
Muenster Univ. of Applied Sciences
E. Rescorla
RTFM, Inc.
October 23, 2009

Datagram Transport Layer Security for Stream Control Transmission Protocol
draft-ietf-tsvwg-dtls-for-sctp-02.txt

Status of this Memo

Copyright Notice

Abstract

This document describes the usage of the Datagram Transport Layer Security (DTLS) protocol over the Stream Control Transmission Protocol (SCTP).

The user of DTLS over SCTP can take advantage of most of the features provided by SCTP and its extensions, especially support of

- multi-homing to provide network level fault tolerance.
- dynamic reconfiguration of IPv4 and IPv6 addresses.
- multiple streams to avoid head of line blocking.
- unordered delivery.
- dynamic reconfiguration of streams.
- partially reliable data transfer.

However, the DTLS maximum user message size limit of 2^14 bytes applies also to DTLS over SCTP. Since DTLS over SCTP uses the SCTP-AUTH extension, the DTLS user can not manage the keying material, since this is done by the DTLS layer.

Table of Contents

1. Introduction

1.1. Overview

This document describes the usage of the Datagram Transport Layer Security (DTLS) protocol, as defined in [I-D.ietf-tls-rfc4347-bis], over the Stream Control Transmission Protocol (SCTP), as defined in [RFC4960].

TLS, from which DTLS was derived, 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 [RFC0793].

TLS over SCTP as described in [RFC3436] has some serious limitations:

- It does not support the unordered delivery of SCTP user messages.
- It does not support partial reliability as defined in [RFC3758].
- It only supports the usage of the same number of streams in both directions.
- It uses a TLS connection for every bidirectional stream, which requires a substantial amount of resources and message exchanges if a large number of streams is used.

DTLS over SCTP as described in this document overcomes these limitations of TLS over SCTP. The user of DTLS over SCTP can use

almost all services provided by SCTP and its partial reliability extension. However, DTLS limits the user message size to 2^14 bytes. The dynamic modification of the IP-addresses used by the SCTP end-points is also supported. The same applies to the dynamic reconfiguration of streams. The DTLS user can request SCTP chunk types to be authenticated by using SCTP-AUTH as defined in [RFC4895]. However, the DTLS user can not perform the SCTP-AUTH key management, because this is done by the DTLS layer.

The method described in this document requires that the SCTP implementation supports the optional feature of fragmentation of SCTP user messages and the SCTP authentication extension defined in [RFC4895].

1.2. Terminology

This document uses the following terms:

Association: An SCTP association.

Stream: A unidirectional stream of an SCTP association. It is uniquely identified by a stream identifier.

1.3. Abbreviations

DTLS: Datagram Transport Layer Security.

MTU: Maximum Transmission Unit.

PPID: Payload Protocol Identifier.

SCTP: Stream Control Transmission Protocol.

TCP: Transmission Control Protocol.

TLS: Transport Layer Security.

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. DTLS Considerations

3.1. Message Sizes

DTLS limits the DTLS user message size to the current Path MTU minus the header sizes. This limit SHOULD be increased to 2^14 Bytes for DTLS over SCTP.

3.2. Replay Detection

Replay detection of DTLS MUST NOT be used.

3.3. Path MTU Discovery

Path MTU discovery of DTLS MUST NOT be used.

3.4. Retransmission of Messages

DTLS procedures for retransmissions MUST NOT be used.

4. SCTP Considerations

4.1. Mapping of DTLS Records

The supported maximum length of SCTP user messages MUST be at least 2^14 + 2048 + 13 = 18445 bytes (2^14 + 2048 is the maximum length of the DTLSCiphertext.fragment and 13 is the size of the DTLS record header). In particular, the SCTP implementation MUST support fragmentation of user messages.

Every SCTP user message MUST consist of exactly one DTLS record.

4.2. Payload Protocol Identifier Usage

Application protocols running over DTLS over SCTP SHOULD register and

use a separate payload protocol identifier (PPID) and SHOULD NOT reuse the PPID which they registered for running directly over SCTP.

This means in particular that there is no specific PPID for DTLS.

4.3. Stream Usage

All DTLS messages of the ChangeCipherSpec, Alert, or Handshake protocol MUST be transported on stream 0 with unlimited reliability and with the ordered delivery feature.

All DTLS messages of the ApplicationData protocol MAY be transported over stream 0 but users SHOULD use other streams to avoid possible performance problems due to head of line blocking.

4.4. Chunk Handling

The DATA, SACK, SHUTDOWN, and FORWARD-TSN chunks of SCTP MUST be sent in an authenticated way as described in [RFC4895]. Other chunks MAY be sent in an authenticated way.

This makes sure that an attacker can not modify the stream a message is sent in or affect the ordered/unordered delivery of the message. It is also not possible for an attacker to drop messages and use forged FORWARD-TSN, SACK, and/or SHUTDOWN chunks to hide this dropping.

4.5. Handshake

A DTLS implementation discards DTLS messages from older epochs after some time as described in section 4.1 of [I-D.ietf-tls-rfc4347-bis]. This is not acceptable when the DTLS user performs a reliable data transfer. To avoid the discarding of messages, the following

procedures are required.

Before sending a ChangeCipherSpec message all outstanding SCTP user messages MUST have been acknowledged by the SCTP peer and MUST NOT be revoked anymore by the SCTP peer.

Prior to processing a received ChangeCipherSpec all other received SCTP user messages which are buffered in the SCTP layer MUST be read

and processed by DTLS.

User messages arriving between ChangeCipherSpec and Finished using the new epoch have probably passed the Finished and MUST be buffered by DTLS until the Finished is read.

4.6. Handling of Endpoint-pair Shared Secrets

The endpoint-pair shared secret for Shared Key Identifier 0 is empty. Whenever the master key changes, a 64 byte shared secret is derived from every master secret and provided as a new end-point pair shared secret by using the algorithm described in [I-D.ietf-tls-extractor].

The Shared Key Identifier MUST be incremented by 1. If it is 65535, the next value MUST be 1.

Before sending the Finished message the active SCTP-AUTH key MUST be switched to the new one.

Once the corresponding Finished message from the peer has been received the old SCTP-AUTH key SHOULD be removed.

4.7. Shutdown

To prevent DTLS from discarding DTLS user messages while shutting down, before sending a CloseNotify message all outstanding SCTP user messages MUST have been acknowledged by the SCTP peer and MUST NOT be revoked anymore by the SCTP peer.

Prior to processing a received CloseNotify all other received SCTP user messages which are buffered in the SCTP layer MUST be read and processed by DTLS.

5. IANA Considerations

IANA needs to add a value to the TLS Exporter Label registry as described in [I-D.ietf-tls-extractor]. The label suggested is EXTRACTOR_DTLS_OVER_SCTP. The reference should refer to this document.

6. Security Considerations

The security considerations given in [I-D.ietf-tls-rfc4347-bis], [RFC4895], and [RFC4960] also apply to this document.

It is possible to authenticate DTLS endpoints based on IP-addresses in certificates. SCTP associations can use multiple addresses per SCTP endpoint. Therefore it is possible that DTLS 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.

The SCTP user provides for each user message also a stream identifier, a flag whether the message is sent ordered or unordered and a payload protocol identifier. Although DTLS can be used to provide privacy for the actual user message, none of these three are protected by DTLS. They are sent as clear text, because they are part of the SCTP DATA chunk header.

If future SCTP extensions define chunk types which processing affect the handling of TSNs, these chunk types MUST be sent in an authenticated way as described in [RFC4895]. One example would be an extension providing an alternate way of acknowledging TSNs.

7. Acknowledgments

The authors wish to thank Carsten Hohendorf, Alfred Hoenes, Daniel Mentz, Ian Goldberg, Anna Brunstrom, and Stefan Lindskog for their invaluable comments.

8. References

8.1. Normative References

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

[RFC3758] Stewart, R., Ramalho, M., Xie, Q., Tuexen, M., and P. Conrad, "Stream Control Transmission Protocol (SCTP) Partial Reliability Extension", RFC 3758, May 2004.

[RFC4895] Tuexen, M., Stewart, R., Lei, P., and E. Rescorla, "Authenticated Chunks for the Stream Control Transmission Protocol (SCTP)", RFC 4895, August 2007.

[RFC4960] Stewart, R., "Stream Control Transmission Protocol", RFC 4960, September 2007.

[I-D.ietf-tls-extractor] Rescorla, E., "Keying Material Exporters for Transport Layer Security (TLS)", draft-ietf-tls-extractor-07 (work in progress), September 2009.

[I-D.ietf-tls-rfc4347-bis] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security version 1.2", draft-ietf-tls-rfc4347-bis-03 (work in progress), October 2009.

8.2. Informative References

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[RFC3436] Jungmaier, A., Rescorla, E., and M. Tuexen, "Transport Layer Security over Stream Control Transmission Protocol", RFC 3436, December 2002.

Authors' Addresses

Michael Tuexen
Muenster Univ. of Applied Sciences
Stegerwaldstr. 39
48565 Steinfurt
Germany

Email: tuexen@fh-muenster.de

Robin Seggelmann
Muenster Univ. of Applied Sciences
Stegerwaldstr. 39
48565 Steinfurt
Germany

Email: seggelmann@fh-muenster.de

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

Email: ekr@networkresonance.com