

Workgroup: Network Working Group
Internet-Draft:
draft-tuexen-tsvwg-sctp-zero-checksum-00
Published: 24 October 2022
Intended Status: Standards Track
Expires: 27 April 2023
Authors: M. Tüxen
Münster Univ. of Appl. Sciences
F. Castelli
Google
V. Boivie
Google
R. Jesup
Mozilla

Zero Checksum for the Stream Control Transmission Protocol

Abstract

The Stream Control Transmission Protocol (SCTP) uses a 32-bit checksum in the common header of each packet to provide some level of data integrity. When the lower layer used by SCTP does already provide the same or a higher level of data integrity, computing this checksum does not provide any additional protection, but does require computing resources. This document provides a simple extension to SCTP allowing to save these computing resources by using the constant 0 as the checksum.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on 27 April 2023.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

- [1. Introduction](#)
- [2. Conventions](#)
- [3. A New Chunk Parameter](#)
- [4. Procedures](#)
- [5. Socket API Considerations](#)
 - [5.1. Get or Set Accepting a Zero Checksum \(SCTP_ACCEPT_ZERO_CHECKSUM\)](#)
- [6. IANA Considerations](#)
- [7. Security Considerations](#)
- [8. References](#)
 - [8.1. Normative References](#)
 - [8.2. Informative References](#)
- [Authors' Addresses](#)

1. Introduction

SCTP as specified in [[RFC9260](#)] uses a CRC32c to provide some level of data integrity. When using, for example, Datagram Transport Layer Security (DTLS) as the lower layer for SCTP as specified in [[RFC8261](#)], using the CRC32c does not provide any additional protection over the one already provided by DTLS. However, computing the CRC32c at the sender and receiver side does require computational resources for no benefit. This is in particular important for computational limited end points using SCTP encapsulated in DTLS.

The extension described in this document allows an SCTP end point to declare that it accepts SCTP packets with a checksum of zero. This declaration happens during the setup of the SCTP association and allows end points supporting this extension to be interoperable with end points not supporting the extension described in this document.

End points using this extension still need to implement the CRC32c checksum algorithms.

2. Conventions

The key words "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 [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

3. A New Chunk Parameter

The Zero Checksum Acceptable Chunk Parameter is defined by the following figure.

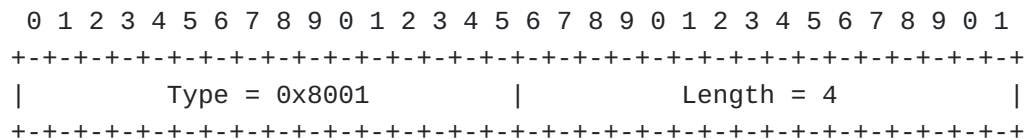


Figure 1: Zero Checksum Acceptable Chunk Parameter

Type: 16 bits (unsigned integer)

This field holds the IANA defined parameter type for the "Zero Checksum Acceptable" chunk parameter. IANA is requested to assign the value 32769 (0x8001) for this parameter type.

Length: 16 bits (unsigned integer)

This field holds the length in bytes of the chunk parameter; the value **MUST** be 4.

All transported integer numbers are in "network byte order" a.k.a., Big Endian.

The Zero Checksum Acceptable Chunk Parameter **MAY** appear in INIT and INIT ACK chunks. It **MUST NOT** appear in any other chunk.

If an end point not supporting the extension described in this document receives this parameter in an INIT or INIT ACK chunk, it skips it and continues to process further parameters in the chunk. This behaviour is **REQUIRED** by [\[RFC9260\]](#) because the highest-order 2 bits of the Type are 10.

4. Procedures

An end point willing to accept SCTP packets with a checksum of zero, **MAY** include the Zero Checksum Acceptable Chunk Parameter in the INIT or INIT ACK chunk it sends. An end point **MUST** do this consistently.

If an end point has sent the Zero Checksum Acceptable Chunk Parameter in an INIT or INIT ACK chunk, it **MUST** accept SCTP packets using a checksum value of zero for this association. It **MUST** also accept SCTP packets containing the correct CRC32c checksum field.

When an end point sends a packet containing an INIT chunk, it **MUST** include a CRC32c checksum in the packet containing the INIT chunk.

When an end point receives an "Out of the Blue" (OOTB) SCTP packet and needs to send a response, it **MUST** include a CRC32c checksum in the response.

If an end point has received an INIT or INIT ACK chunk from its peer during the association setup, it **MAY** use zero as the checksum for all packets sent in this association.

5. Socket API Considerations

This section describes how the socket API defined in [\[RFC6458\]](#) needs to be extended to provide a way for the application to control the UDP encapsulation.

Please note that this section is informational only.

A socket API implementation based on [\[RFC6458\]](#) is extended by supporting one new read/write IPPROTO_SCTP level socket option.

5.1. Get or Set Accepting a Zero Checksum (SCTP_ACCEPT_ZERO_CHECKSUM)

This socket option can be used to control the acceptance of a zero checksum. It applies only to future SCTP associations on the socket.

This option expects an integer boolean flag, where a non-zero value turns on the option, and a zero value turns off the option.

This option is off by default.

6. IANA Considerations

[NOTE to RFC-Editor: "RFCXXXX" is to be replaced by the RFC number you assign this document.]

[NOTE to RFC-Editor: The requested value for the parameter type is tentative and to be confirmed by IANA.]

This document (RFCXXXX) is the reference for the registration described in this section.

A new chunk parameter type has to be assigned by IANA. This requires an additional line in the "Chunk Parameter Types" registry for SCTP:

ID Value	Chunk Parameter Type	Reference
32769	Zero Checksum Acceptable (0x8001)	[RFCXXXX]

Table 1: New entry in "Chunk Parameter Types" registry

7. Security Considerations

This document does not change the considerations given in [\[RFC9260\]](#).

8. References

8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8261] Tuexen, M., Stewart, R., Jesup, R., and S. Loreto, "Datagram Transport Layer Security (DTLS) Encapsulation of SCTP Packets", RFC 8261, DOI 10.17487/RFC8261, November 2017, <<https://www.rfc-editor.org/info/rfc8261>>.
- [RFC9260] Stewart, R., Tüxen, M., and K. Nielsen, "Stream Control Transmission Protocol", RFC 9260, DOI 10.17487/RFC9260, June 2022, <<https://www.rfc-editor.org/info/rfc9260>>.

8.2. Informative References

- [RFC6458] Stewart, R., Tuexen, M., Poon, K., Lei, P., and V. Yasevich, "Sockets API Extensions for the Stream Control Transmission Protocol (SCTP)", RFC 6458, DOI 10.17487/RFC6458, December 2011, <<https://www.rfc-editor.org/info/rfc6458>>.

Authors' Addresses

Michael Tüxen
Münster University of Applied Sciences
Stegerwaldstrasse 39
48565 Steinfurt
Germany

Email: tuexen@fh-muenster.de

Victor Boivie
Google
Kungsbron 2
SE-11122 Stockholm
Sweden

Email: boivie@google.com

Florent Castelli
Google

Kungsbron 2
SE-11122 Stockholm
Sweden

Email: orphis@google.com

Randell Jesup
Mozilla Corporation
1835 Horse Shoe Trl
Malvern, PA 19355
United States of America

Email: randell-ietf@jesup.org