Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 24 August 2025

T. Herbert

20 February 2025

IPv6 Checksum Option
draft-herbert-ipv6-checksum-option-00

Abstract

This document specifies a Checksum Option for IPv6. This is a Destination Option that allows a checksum to be computed over a variable number of bytes in the packet. The option allows the ICMPv6 checksum to be optional, and the UDPv6 checksum to be generally optional.

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 24 August 2025.

Table of Contents

1. Introduction

This document specifies a Checksum Option for IPv6 [RFC8200]. This is a Destination Option that includes a checksum coverage field and checksum field similar to UDP-Lite [RFC3828]. If the option is present then the checksum is computed starting from the first byte of the Destination Options header through the number of bytes indicated by the checksum coverage. The checksum includes coverage over a pseudo header composed of the IP addresses in the packet.

The Checksum Option allows the following ICMPv6 checksum [RFC4443] to be optional, and generalizes the following UDPv6 checksum of Section 8.1 of [RFC8200] to be optional beyond that allowed by [RFC6935].

1.1. Motivation

The motivation and goals for the IPv6 Checksum Option are:

* Provide an optional checksum, with checksum coverage, that works with any transport protocol.

* Allow the ICMPv6 checksum to be optional. This is pertinent for

routers that wish to send ICMP errors but don't have the capability to efficiently calculate checksums over hundreds of bytes.

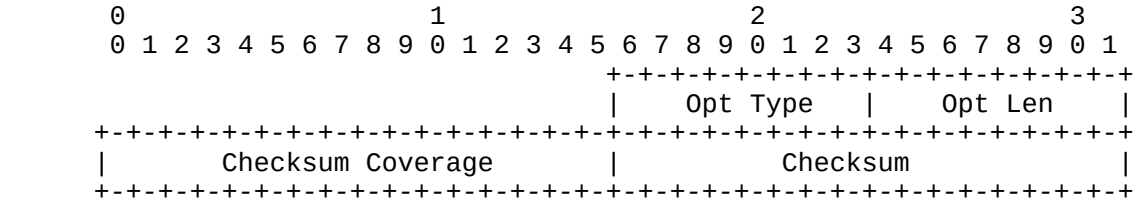
* Allow the UDPv6 checksum to be generally optional. The Checksum Option allows the UDPv6 checksum to be zero outside of use with UDP tunnels. Also, the Checksum Option allows partial coverage similar to UDP-lite.

1.2. Terminology

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].

2. Protocol format

The Checksum Option is a Destination Option with the format:



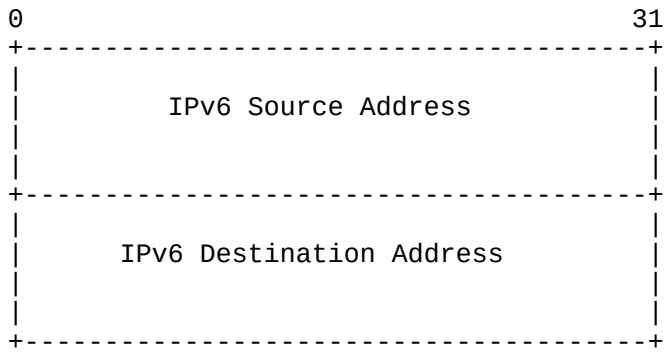
Opt Type
: The Option Type is TBD. The highest-order two bits MUST be 10, 01, or 11 to indicate that the packet MUST be dropped if the option type is unrecognized. The third-highest-order bit MUST be 0 to indicate that the option may not change en route.

Opt Len
: The Option Length is set to 4.

Checksum Coverage
: The number of bytes, counting from the first byte of the containing Destination Options header, that are covered by the checksum.

Checksum
: The 16-bit one's complement of the one's sum of a pseudo-header of information collected from the IP header and over the number of bytes specified by the Checksum Coverage (starting from the first byte of the containing Destination Options header), virtually padded with a zero byte at the end (if necessary) to make a multiple of two bytes.

The Checksum Option includes coverage for a pseudo header consisting of the IP addresses in the packet. The format of the pseudo header is:



3. Operation

3.1. Sender operation

The procedures for setting the Checksum option are:

1. A sender creates the option in a packet. The Option Type field is set with an appropriate option type. The Checksum Option SHOULD NOT be set in Destination Options before the Routing header.

2. The Checksum Coverage field is set to number of bytes covered by the checksum. The Checksum Coverage field MUST be less than or equal to the length of the IP packet starting from the first byte of the Destination Options header.

3. The Checksum field is set to zero for computing the checksum.

4. The ones' complement sum over the pseudo header is computed. This covers the source and destination IP addresses.

5. The ones' complement sum over the bytes starting from the first byte of the Destination Options header for the number of bytes set in the Checksum Coverage field is computed. Note that any other options in the Destination Options header as well as any data following the Destination Options header that is covered by the checksum MUST be set before checksum computation.

6. The results of set #4 and step #5 are ones' complement summed and a sixteen bit value is produced. The 'not' of the value is set in the Checksum field.

3.2. Receiver operation

The procedures for processing the Checksum Option are:

1. If the value in the Checksum Coverage field is greater the length of the packet starting from the first byte of the Destination Options header then that is considered an error. In this case the packet is discarded, and if the high-order two bits of the Option Type is 10 or 11 then the receiver MAY send a Parameter Problem message with code 0 ("erroneous header field encountered") back to the sender.

2. The ones' complement sum over the pseudo header is computed. This covers the source and destination IP addresses.

3. The ones' complement sum over the bytes starting from the first byte of the Destination Options header for the number of bytes set in the Checksum Coverage field is computed.

4. The results of set #2 and step #3 are ones' complement summed and a sixteen bit value is produced. If the resultant value is equal to all ones (0xFFFF) then the checksum is valid and the packet may be accepted. If the resultant value is not all ones then the checksum is invalid. In this case, the packet MUST be discarded, and if the highest-order two bits of the Option Type is 10 or 11 then the receiver MAY send and Parameter Problem message with code 0 ("erroneous header field encountered") back to the sender.

4. Optional ICMP and UDP checksums

The ICMPv6 checksum and UDPv6 checksum are optional with the Checksum Option per the requirements in this section.

4.1. ICMP checksums

If a Destination Options header containing a Checksum Option immediately precedes an ICMP header, that is the Next Header field of the Destination Options header is equal to 1, then requirements of the ICMP checksum are modified.

4.1.1. Sender requirements

The ICMPv6 checksum MAY be set by a sender to be skipped. This is done by setting a Checksum Option in the Destination Options header immediately preceding the ICMPv6 header and setting the ICMPv6 checksum field to 0. Effectively, this makes the ICMPv6 checksum optional.

If the Checksum Option is present in the Destination Options header immediately preceding the ICMPv6 header, the ICMPv6 checksum is being computed, and the computed checksum is zero, then the ICMP checksum field is set to all ones (0xFFFF).

4.1.2. Receiver requirements

If ICMPv6 checksum is zero in a received packet and the ICMPv6 header is preceded by a Destination Options header containing the Checksum Option then the packet MAY be accepted without further consideration of the ICMP checksum.

If the ICMPv6 checksum is non-zero then the checksum processed as normal regardless of whether the Checksum Option is present.

4.1.3. Considerations

The requirements for optional ICMPv6 checksums are similar to those for a optional UDPv6 checksums described in Section 5 of [RFC6936] with a few amendments.

* Since the Checksum Option includes the checksum over the pseudo header this mitigates concerns of packet misdelivery.

* A sender MAY include the ICMP header in the checksum coverage of the Checksum Option. This provides protection of the ICMP header.

* A sender MAY include some number of bytes of the ICMP payload in the checksum coverage of the Checksum Option. This could, for instance, include coverage of the IP headers and maybe transport layer headers of the invoking packet.

* It is optional at a receiver to accept ICMPv6 packets with a zero checksum. The default behavior SHOULD be to not accept packets with a zero ICMPv6 checksum.

4.2. UDP checksums

If a Destination Options header containing a Checksum Operation immediately precedes a UDP header, that is the Next Header field of the Destination Options header is equal to 17, then the requirements of the UDP checksum are modified.

4.2.1. Sender requirements

The UDPv6 checksum MAY be set by a sender to be skipped. This is done by setting a Checksum Option in the Destination Options header immediately preceding the UDPv6 header and setting the ICMPv6 checksum field to 0. Effectively, this makes the UDPv6 checksum generally optional.

4.2.2. Receiver requirements

If a UDPv6 checksum is zero in a received packet and the UDP header is immediately preceded by a Destination Options header containing the Checksum Option then the packet MAY be accepted without further consideration of the UDP checksum.

If UDPv6 checksum is non-zero then the checksum is processed normally regardless of whether the Checksum Option is present.

4.2.3. Considerations

[RFC6936] permits a zero UDPv6 checksum to be used with UDP tunnels. This specification extends that to allow the UDPv6 checksum to be zero in non-UDP tunnel scenarios. The requirements of Section 5 of [RFC6936] are generally applicable to the protocol of this specification with the following amendments.

* Since the Checksum Option includes the checksum over the pseudo header that mitigates concerns of packet misdelivery.

* The Checksum Coverage allows alternate coverage. A sender MAY ensure that the UDP header is covered by the Checksum Options.

5. Security Considerations

This specification does not introduce any new security concerns.

6. IANA Considerations

IANA is requested to allocate an IPv6 Destination type for the Checksum Option in the "Destination Options and Hop-by-Hop Options" registry within the "Internet Protocol Version 6 (IPv6) Parameters" registry group [IANA-DESTOPT]. Variants are:

Hex Value	Binary value act chg rest	Description	Reference
TBD	01 0 xxxxx	Checksum option	This document
TBD	10 0 xxxxx	Checksum option	This document
TBD	11 0 xxxxx	Checksum option	This document

7. References

7.1. 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>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

7.2. Informative References

[IANA-DESTOPT] "Destination Options and Hop-by-Hop Options", <https://www.iana.org/assignments/ipv6-parameters>.

[RFC3828] Larzon, L., Degermark, M., Pink, S., Jonsson, L., Ed., and G. Fairhurst, Ed., "The Lightweight User Datagram Protocol (UDP-Lite)", RFC 3828, DOI 10.17487/RFC3828, July 2004, <https://www.rfc-editor.org/info/rfc3828>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[RFC6935] Eubanks, M., Chimento, P., and M. Westerlund, "IPv6 and UDP Checksums for Tunneled Packets", RFC 6935, DOI 10.17487/RFC6935, April 2013, <https://www.rfc-editor.org/info/rfc6935>.

[RFC6936] Fairhurst, G. and M. Westerlund, "Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums", RFC 6936, DOI 10.17487/RFC6936, April 2013, <https://www.rfc-editor.org/info/rfc6936>.

Author's Address

Tom Herbert
Los Gatos, CA,
United States of America
Email: tom@herbertland.com