

October 19, 2015

Checksum option for Generic UDP Encapsulation
[draft-herbert-guecsum-01](#)

Abstract

This specification defines the Generic UDP Encapsulation (GUE) checksum and an associated header option. This checksum covers the GUE header, IP addresses, UDP ports, and optionally all or part of the encapsulated payload. It provides verification of protocol header elements, and is particularly relevant in the case where the UDP checksum is set to zero.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

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/1id-abstracts.html>

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

Copyright and License Notice

Copyright (c) 2015 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

(<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1	Introduction	3
2	Option format	3
3	Operation	4
3.1	Requirements	4
3.2	GUE pseudo header	5
3.3	Checksum computation	6
3.4	Transmitter operation	6
3.5	Receiver operation	7
4	Security Considerations	7
5	IANA Considerations	7
6	References	7
6.1	Normative References	7
6.2	Informative References	8
	Author's Address	8

1 Introduction

The UDP checksum provides a method to detect corrupted packets [RFC0768]. The covered checksum includes a pseudo header consisting of IP addresses, payload length, and protocol number (17 for UDP). The pseudo header checksum protects against misdelivery due to corrupted IP addresses, as well as some other issues occurring when the IP header is corrupted. The IPv4 header contains its own header checksum, however IPv6 does not. In the latter case there is motivation when using UDP to enable the UDP checksum to protect against misdelivery due to address corruption.

For UDP tunnels, there may be performance disadvantages in enabling the UDP checksum. This may, for instance, be an issue in switch hardware which might only have access to a limited portion of the packet for inspection. Therefore, there is motivation to use zero checksums with UDP tunneling. The requirements and applicability of using zero UDP checksums with IPv6 are in [RFC 6935](#) [[RFC6935](#)] and [RFC 6936](#) [[RFC6936](#)].

In this document we define the Generic UDP Encapsulation [[GUE](#)] checksum. This provides a checksum that covers the GUE header and a GUE pseudo header. The GUE pseudo header includes the corresponding IP addresses as well as the UDP ports of the encapsulating headers. This checksum should provide adequate protection against address corruption in IPv6 when the UDP checksum is zero. Additionally, the GUE checksum provides protection of the GUE header when the UDP checksum is set to zero with either IPv4 or IPv6. In particular, the GUE checksum can provide protection for some sensitive data, such as the virtual network identifier [[GUENVO3](#)], which when corrupted could lead to misdelivery of the packet.

The GUE header checksum may optionally cover all or part of the encapsulated payload. This is similar to the model of UDP-Lite [RFC3828] where an additional field indicates the portion of the payload that is covered in the checksum.

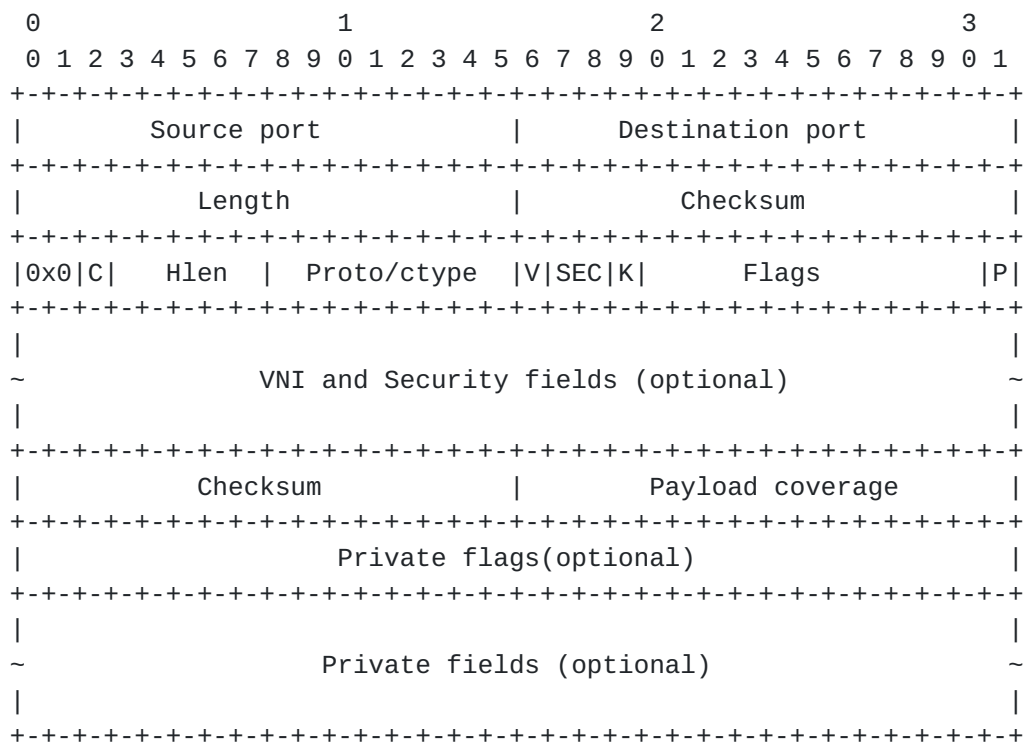
2 Option format

The GUE header checksum is sent in an optional field in the GUE header. The format of the GUE checksum option field is:

[illegible]

- o Checksum: GUE checksum. This checksum covers the GUE header, the GUE pseudo header, and optionally all or part of the payload (encapsulated packet).
- o Payload coverage: Number of bytes of payload to cover in the checksum. Zero indicates that the checksum only covers the GUE header and GUE pseudo header. If the value is greater than the encapsulated payload length, the packet must be dropped. The payload length is $\text{UDP_length} - 12 - (\text{Hlen} * 4)$.

The format of the checksum option within the GUE header is:



- o K bit - Indicates presence of the checksum and payload coverage fields.

3 Operation

3.1. Requirements

The GUE header checksum must be set on transmit when using a zero UDP checksum with IPv6.

The GUE header checksum must be set when the UDP checksum is zero for IPv4 if the GUE header includes data that when corrupted can lead to misdelivery or other serious consequences, and there is no other

mechanism that provides protection (no security field for instance). Otherwise the GUE header checksum should be used with IPv4 when the UDP checksum is zero.

The GUE header checksum should not be set when the UDP checksum is non-zero. In this case the UDP checksum provides adequate protection and this avoids convolutions when a packet traverses NAT that does address translation (in that case the UDP checksum is required).

3.2. GUE pseudo header

The GUE pseudo header checksum is included in the GUE checksum to provide protection for the IP and UDP header elements which when corrupted could lead to misdelivery of the GUE packet. The GUE pseudo header checksum is similar to the standard IP pseudo header defined in [[RFC0768](#)] and [[RFC0793](#)] for IPv4, and in [[RFC2460](#)] for IPv6.

The GUE pseudo header for IPv4 is:

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     Source Address                      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     Destination Address                  |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      Source port                    |      Destination port            |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

The GUE pseudo header for IPv6 is:

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     |
+                                     +
|                                     |
+                                     +
|                                     Source Address                      +
|                                     |
+                                     +
|                                     |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     |
+                                     +
|                                     |
+                                     +
|                                     Destination Address                  +
|                                     |
+                                     +
|                                     |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      Source port                    |      Destination port            |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```


Note that the GUE pseudo header does not include payload length or protocol as in the standard IP pseudo headers. The length field is deemed unnecessary because:

- o If the length is corrupted this will usually be detected by a checksum validation failure on the inner packet.
- o Fragmentation of packets in a tunnel should occur on the inner packet before being encapsulated. GUE packets are not expected to be fragmented when using IPv6. See [RFC6936](#) for considerations of payload length and IPv6 checksum.
- o A corrupted length field in itself should not lead to misdelivery of a packet.
- o Without the length field, the GUE pseudo header checksum is the same for all packets of flow. This is a useful property for optimizations such as TCP Segment Offload (TSO).

[3.3. Checksum computation](#)

The GUE checksum is computed and verified following the standard process for computing the Internet checksum [[RFC1071](#)]. Checksum computation may be optimized per the mathematical properties including parallel computation and incremental updates.

[3.4. Transmitter operation](#)

The procedure for setting the GUE checksum on transmit is:

- 1) Create the GUE header including the checksum and payload coverage fields. The checksum field is initially set to zero.
- 2) Calculate the 1's complement checksum of the GUE header from the start (GUE version) through the its length as indicated in GUE Hlen.
- 3) Calculate the checksum of the GUE pseudo header for IPv4 or IPv6.
- 4) Calculate checksum of payload portion if payload coverage is enabled (payload coverage field is non-zero). If the length of the payload coverage is odd, logically append a single zero byte for the purposes of checksum calculation.
- 5) Add and fold the computed checksums for the GUE header, GUE pseudo header and payload coverage. Set the result in the GUE checksum field.

[3.5.](#) Receiver operation

If the GUE checksum option is present, the receiver must validate the checksum before processing any other fields or accepting the packet.

The procedure for verifying the checksum is:

- 1) If the payload coverage length is greater than the length of the encapsulated payload then drop the packet. The length of the encapsulated payload is: $\text{UDP_length} - 12 - (\text{Hlen} * 4)$.
- 2) Calculate the checksum of the GUE header from the start of the header to the end as indicated by Hlen.
- 3) Calculate the checksum of the appropriate GUE pseudo header.
- 4) Calculate the checksum of payload if payload coverage is enabled (payload coverage is non-zero). If the length of the payload coverage is odd logically append a single zero byte for the purposes of checksum calculation.
- 5) Sum the computed checksums for the GUE header, GUE pseudo header, and payload coverage. If the result is all 1 bits (-0 in 1's complement arithmetic), the checksum is valid and the packet is accepted; otherwise the checksum is considered invalid and the packet must be dropped.

[4](#) Security Considerations

The checksum option is only a mechanism for corruption detection, it is not a security mechanism. To provide integrity checks or authentication of the GUE header, the GUE security option should be used [[GUESEC](#)].

[5](#) IANA Considerations

There are no IANA considerations in this specification. One of the GUE reserved flag bits is allocated to indicate presence of the checksum field.

[6](#) References

[6.1](#) Normative References

[GUE] Generic UDP Encapsulation [draft-ietf-nvo3-gue-01](#)

- [RFC0791] Postel, J., "Internet Protocol", STD 5, [RFC 791](#), September 1981.
- [RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, [RFC 1122](#), October 1989.
- [RFC0768] Postel, J., "User Datagram Protocol", STD 6, [RFC 768](#), August 1980.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.
- [RFC1071] Braden, R., Borman, D., and C. Partridge, "Computing the Internet checksum", [RFC 1071](#), September 1988.

[6.2](#) Informative References

- [RFC6935] Eubanks, M. Chimento, P., and M. Westerlund, "IPv6 and UDP Checksums for Tunneled Packets", [RFC 6935](#), April 2013.
- [RFC6936] Fairhurst, G. and M. Westerlund, "Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums", [RFC 6936](#), April 2013.
- [RFC3828] Larzon, L-A., Degermark, M., Pink, S., Jonsson, L-E., Ed., and G. Fairhurst, Ed., "The Lightweight User Datagram Protocol (UDP-Lite)", [RFC 3828](#), July 2004.
- [RFC0793] Postel, J., "Transmission Control Protocol", STD 7, [RFC 793](#), September 1981.
- [GUENV03] Generic UDP Encapsulation (GUE) for Network Virtualization Overlay [draft-hy-nvo3-gue-4-nvo-00](#)
- [GUESEC] Generic UDP Encapsulation (GUE) for Secure Transport [draft-hy-gue-4-secure-transport-00](#)

Author's Address

Tom Herbert
Facebook
Menlo Park, CA
USA

Email: tom@herbertland.com

