

INTERNET-DRAFT
Intended Status: Standard
Expires: September 2019

T. Herbert
Quantonium

March 11, 2019

UDP Surplus Space Header
[draft-herbert-udp-space-hdr-00](#)

Abstract

This draft defines a header for surplus space in UDP. The UDP surplus space is bytes between the end of the UDP payload and the end of the IP packet. The purpose of the header is to disambiguate uses of the surplus space. The UDP surplus space header includes a type, length, and checksum field that covers the space.

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/lid-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) 2019 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

INTERNET DRAFT

[draft-herbert-udp-opt-hdr-00](#)

March 11, 2019

(<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	UDP surplus space header format	3
3	Operation	4
3.1	Sender operation	4
3.2	Receiver operation	6
4	Security Considerations	7
5	IANA Considerations	7
6	References	7
6.1	Normative References	7
6.2	Informative References	7
	Author's Address	7

INTERNET DRAFT

[draft-herbert-udp-opt-hdr-00](#)

March 11, 2019

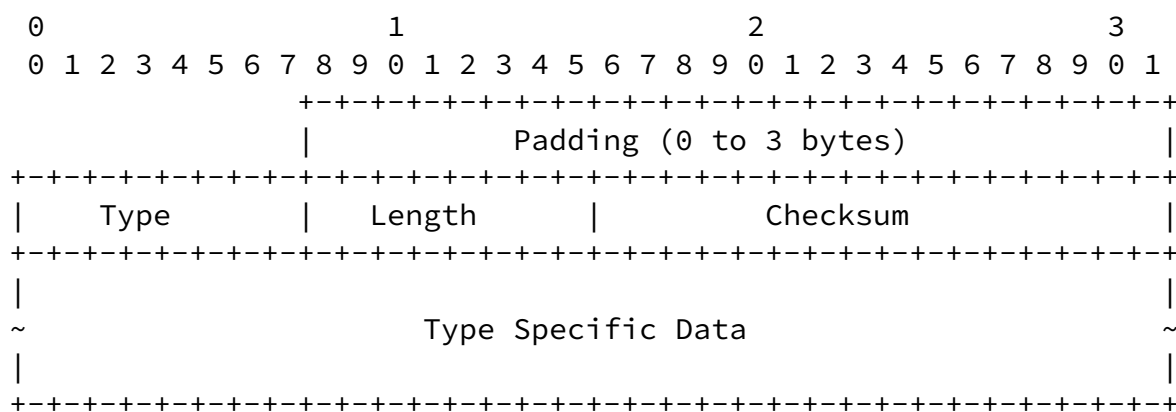
1 Introduction

This specification defines a UDP surplus header to provide an extensible and generic format of the UDP surplus space. The UDP surplus space is bytes between the end of the UDP payload and the end of the IP packet. The UDP surplus space is possible since UDP contains its own length field for the UDP datagram that is not required to coincide with the end of the IP packet.

The UDP surplus space header includes a type, length, and checksum field that covers the space. UDP options [[UDPOPT](#)] are being defined as the first standard use of the UDP surplus space. This draft permits other uses, including private uses of the space.

2 UDP surplus space header format

The format of the UDP surplus space header is shown below.



The fields are:

- o **Padding:** Aligns the header to four bytes. The number of padding bytes required is: $3 - ((udp_length - 1) \% 4)$ where the `udp_length` is the length of the UDP header and UDP payload. Padding bytes **MUST** be set to zero on transmission, and **MUST** be

verified to be zero when received.

- o Type: The type of the UDP surplus space. One type is defined for UDP options.
- o Length: Length of the Type Specific Data in units of four byte words. The length of the Type Specific Data is thus zero to 1020 bytes.
- o Checksum: The standard one's complement checksum that covers the UDP surplus space starting from the Type field through the end of the Type Specific Data as indicated in the Length field.

Preceding Padding Bytes are not included in the checksum coverage.

- o Type specific data: Data corresponding to the Type of the surplus data. For instance, in the case of UDP options, the Type specific data would contain the UDP options including any padding options to align the length of the Type Specific Data to four bytes.

Defined Type values are:

- o 0: Reserved
- o 1: UDP options
- o 2-127: Reserved
- o 128-255: Available for private use or experimentation

[3](#) Operation

[3.1](#) Sender operation

A sender MAY set a UDP surplus space header and data in the surplus space when sending a UDP packet. The UDP surplus header immediately follows the UDP payload at the offset of UDP Length from the start of the UDP header.

The sender MUST insert up to three bytes of padding to align the UDP

surplus header to four bytes.

The sender MUST set the Type field to the type of the data to be processed.

The Length field MUST be set to reflect the length of the Type Specific Data.

The Checksum MUST be set. To compute the checksum:

- 1) Set the Checksum field to zero. Compute the standard one's complement two byte checksum starting from the Type field through the end of the Type Specific Data.
- 2) Set the value of the Checksum field to the bitwise "not" of the checksum computed in the previous step.

Note that a sender MAY set multiple UDP surplus area headers in a packet. If more than one is set, then they are consecutive where one

immediately follows another.

[3.2](#) Receiver operation

The following steps are for processing a UDP packet with surplus space:

- 1) Check for minimum length to contain a surplus option header. If the surplus space length is less than $3 - ((\text{udp_length} - 1) \% 4) + 4$, then the UDP surplus space header is not present.
- 2) Check padding bytes. If the UDP Length is not a multiple of four bytes then verify that the padding bytes following the UDP payload are set to zero. Then expected number of padding bytes is $3 - ((\text{udp_length} - 1) \% 4)$.
- 3) Check the Length field. If the length determined from the

Length field plus the starting offset of the Type specific data exceeds the length of the packet then the surplus area header is considered invalid.

- 4) Verify the checksum. Compute the one's complement checksum starting from the Type field through the end of the Type specific data as indicated by the Length field. If the result of the computation ~ 0 (-0 checksum value) then the checksum is verified. If the checksum is not verified then the surplus area header is invalid.
- 5) Check the Type. If the Type is unknown to the receiver then the surplus header is considered invalid.
- 6) Process the Type Specific Data per the Type in the UDP surplus space header. In the course of processing the Type specific data, the receiver may determine that the surplus data is invalid.
- 7) If there are additional bytes beyond the surplus header, a receiver MAY attempt to process those bytes as an additional surplus header. Processing would be the same as how the first header was processed (although note that subsequent UDP surplus headers would not need Padding bytes since they would already be aligned to four bytes).

If the number surplus bytes in a UDP packet are non-zero and determined to not contain a valid UDP surplus header, the surplus bytes SHOULD be ignored following existing protocol processing convention. An implementation MAY allow configuration that would discard such packets.

[4](#) Security Considerations

The UDP surplus space header does not address nor introduce any new security considerations.

[5](#) IANA Considerations

IANA is requested to create a registry for the Type of UDP surplus

options.

6 References

6.1 Normative References

6.2 Informative References

[UDPOPT] Touch, J., "Transport Options for UDP", [draft-ietf-tsvwg-udp-options-07](#)

Author's Address

Tom Herbert
Quantonium
Santa Clara, CA
USA

Email: tom@quantonium.net