Extendable User Datagram Protocol (EUDP)
draft-yevstifeyev-eudp-08

Abstract

This document is a specification of experimental Extendable User Datagram Protocol (EUDP), which is based on User Datagram Protocol (UDP), but allows to extend its header and, therefore, its functionality with options.

Table of Contents

Table of Contents

1. Introduction

This document is a specification of experimental Extendable User Datagram Protocol (EUDP), which is based on User Datagram Protocol (UDP) [RFC0768], but allows to extend its header and, therefore, its functionality with options.

Such solution may be useful in the situations when UDP provides lack of features while other transport-layer protocols, such as Transmission Control Protocol (TCP) [RFC0793] or Datagram Congestion Control Protocol (DCCP) [RFC4340], have excessive facilities, which might not be needed in such particular case. Current transport-layer protocols are not able to cope with such situations.

Unlike them, EUDP allows to choose what features it will provide to the users via the options. Options allow to append different capabilities to EUDP's core transport functionality. Therefore, it may suit to almost any requirements.

Please note that EUDP is experimental protocol. Therefore any suggestions for improvements and comments, directed to the author of this document, are encouraged and welcome.

1.1. 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 RFC 2119 [RFC2119].

2. Protocol Description

2.1. Lower Layer Protocols Considerations

EUDP is a transport-layer protocol, per RFC 1122 [RFC1122] (Layer 4 in Open Systems Interconnection Basic Reference Model [OSI]). It is implemented on the top of Internet Protocol (IP). EUDP SHALL support as IPv4 [RFC0791], as IPv6 [RFC2460] as lower-layer protocol. The IP Protocol number to be used with EUDP is TBD1.

Moreover, EUDP SHALL be able to operate on the any protocol that provides the same functionality as IP, such as EIP (Extended Internet Protocol) [RFC1385] or IPv7 (also known as TP/IX) [RFC1475].

2.2. Packet Format

2.2.1. Header

The EUDP header is shown below, in Figure 1.

```
 0             15 16            31
+----------------+----------------+
|   Source Port  |Destination Port|
+----------------+----------------+
|   Data Offset  |    Reserved    |
+----------------+----------------+
|                                 |
:            Options              :
:                                 |
|                +----------------+
|                |     Padding    |
+=================================+
|                                 |
:              Data               :
|                                 |
+----------------+----------------+
```

Figure 1. EUDP Header

2.2.2. Fields

Source Port (16 bits) - REQUIRED field which is defined and is to be used as described in UDP specification - RFC 768 [RFC0768]. EUDP uses the same port set as UDP.

Destination Port (16 bits) - REQUIRED field which is defined and is to be used as described in UDP specification - RFC 768 [RFC0768]. EUDP uses the same port set as UDP.

Data Offset (16 bits) - REQUIRED field which is the number of 32 bit words in the EUDP header. The EUDP header (even one including options) SHALL be an integral number of 32 bits long.

Options (variable length) - OPIONAL field, that is used to carry EUDP options in. EUDP option are described in Section 2.3.

Padding (variable length) - OPTIONAL field. The EUDP header padding is used to ensure that the EUDP header ends and data begins on a 32 bit boundary. The padding MUST be composed of zeros.

NOTE: Bits 48-63 are reserved for future use and MUST be ignored by EUDP hosts until their use will be properly specified.

2.3. EUDP Options

2.3.1. Generic Description

EUDP options are placed into the 'Options' header field. Options may occupy space at the end of the EUDP header and are a multiple of 8 bits in length. An option may begin on any octet boundary. There are two cases for the format of an option:

a) a single octet of option-kind.

b) an octet of option-kind, an octet of option-length, and the actual option-data octets. The option-length counts the two octets of option-kind and option-length as well as the option-data octets.

Note that the list of options MAY be shorter than the data offset field might imply. The content of the header beyond the 'End Of Options List' option MUST be header padding (i.e., zero).

All options fall into two categories: _critical_ and _elective_ . These categories prescribe the host's behavior in the case when it receives an unsupported option, as described below. The even numbers of option-kind, including zero, (i. e. 0, 2, 4, 6...) are used to identify _critical_ options whereas odd numbers of option-kind identify _elective_ options.

If EUDP host receives the packet with unsupported _critical_ option, this packet SHALL be discarded. If the unknown option is _elective_, the receiver of the packet with it MAY continue its processing.

NOTE: While _critical_ and _elective_ categories are used to ensure how the options are processed, they do not describe the support criteria for them. The support of all options is OPTIONAL for EUDP hosts.

Pre-defined options are specified in Section 2.3.2.

2.3.2. Pre-Defined Options

This section specifies pre-defined EUDP options.

2.3.2.1. 'No Operation' Option

```
+--------+
| Kind=0 |
+--------+
```

Figure 2. 'No Operation' Option

This option may be used between options, for example, to align the beginning of a subsequent option on a word boundary.

The option is _critical_, per Section 2.3.1.

2.3.2.2. 'End Of Options List' Option

```
+--------+
| Kind=2 |
+--------+
```

Figure 3. 'End Of Options List' Option

This option code indicates the end of the options list. This might not coincide with the end of the EUDP header according to the 'Data Offset' header field. This option SHALL be used at the end of all options, not the end of each option, and need only to be used if the end of the options would not otherwise coincide with the end of the EUDP header.

The option is _critical_, per Section 2.3.1.

2.3.2.3. 'Echo Request' Option

```
+--------+--------+-------//-------+
| Kind=1 | Length |      Data      |
+--------+--------+-------//-------+
```

Figure 4. 'Echo Request' Option

The 'Echo Request' option is used to provide the possibility of echo debugging using the EUDP. The option-data octets of option MAY consist of arbitrary octets. The receiver of the packet with this option SHALL answer with the packet with 'Echo Response' option (see Section 2.3.2.4), if it supports echo debugging via EUDP.

The option is _elective_, per Section 2.3.1.

2.3.2.4. 'Echo Response' Option

```
+--------+--------+-------//-------+
| Kind=3 | Length |      Data      |
+--------+--------+-------//-------+
```

Figure 5. 'Echo Response' Option

The 'Echo Response' option is put in the packets that are sent in response to packets with 'Echo Request' option (see Section 2.3.2.3). The packet containing 'Echo Response' option SHALL be send by the EUDP host after receiving any EUDP packet with 'Echo Request' option, if echo debugging via EUDP is supported by it. The option-data

octets of the option MUST be the same as in 'Echo Request' option in the received packet.

The option is _elective_, per Section 2.3.1.

2.3.2.5. 'Packet Identifier' Option

```
+--------+--------+--------+-------+
| Kind=4 |Length=4|    Packet ID   |
+--------+--------+--------+-------+
```

Figure 6. 'Packet Identifier' Option

The 'Packet Identifier' option is to request the acknowledgment of the single separate packet. The 'Packet ID' field is filled by arbitrary bytes by the sender of the packet with this option. The receiver of the packet with 'Packet Identifier' option MUST answer with the packet with 'Packet Acknowledgment' option (see Section 2.3.2.6). See Section 2.5 for details.

The option is _critical_, per Section 2.3.1.

2.3.2.6. 'Packet Acknowledgment' Option

```
+--------+--------+--------+-------+
| Kind=6 |Length=4| ACK Packet ID  |
+--------+--------+--------+-------+
```

Figure 7. 'Packet Acknowledgment' Option

The 'Packet Acknowledgment' option is used in the packets that are sent in response to the packets with 'Packet Identifier' option. The 'ACK Packet ID' field in the option in the packet sent to the originating host for packet with 'Packet Identifier' option MUST be the same as in the 'Packet Identifier' option-data octets in received from this host packet. See Section 2.5 for details.

The option is _critical_, per Section 2.3.1.

2.3.2.7. 'Packet Checksum' Option

```
+--------+--------+--------+-------+
| Kind=5 |Length=4|    Checksum    |
+--------+--------+--------+-------+
```

Figure 8. 'Packet Checksum' Option

The 'Packet Checksum' option is used to carry packet checksum,

calculated using the method described in RFC 768 [RFC0768]. Packets containing this option with bad checksum SHOULD be discarded by EUDP host, if this option is supported by it.

The option is _elective_, per Section 2.3.1.

2.4. Pseudo Header

EUDP does not use pseudo header.

2.5. Packet Delivery Acknowledgment Mechanism

EUDP provides the possibility to request the acknowledgment of the single EUDP packet. This is provided by 'Packet Identifier' and 'Packet Acknowledgment' options (see Section 2.3.2.5 and Section 2.3.2.6, respectively). In the simplest form, the delivery acknowledgment mechanism works as below.

If EUDP host (let it be A) wants to request the acknowledgment of delivery of some packet, it puts the 'Packet Identifier' option in it and sends this packet to another EUDP host (let it be B). Once B receives the A's packet, it checks the 'Packet Identifier' option to find out the packet identifier. After it finds it out, this number becomes the 'ACK Packet Identifier', that is put into 'Packet Acknowledgment' option, which is put into the EUDP packet, and sent to A. The packet identifiers MAY be reused once they are acknowledged, since EUDP does provides stateless connection.

Compared with acknowledgment mechanism of TCP [RFC0793], EUDP provides simpler and more liberal system. While TCP makes using the packet sequence numbers and acknowledgement mandatory, EUDP allows the host to decide whether the packet needs to be acknowledged by the other side or not.

2.6. Compatibility with UDP

The applications which use UDP can safely use EUDP with no options instead.

3. Security Considerations

UDP is inheritedly insecure. It provides neither reliable packet delivery nor authentication features. EUDP without any options does not cover these issues as well. However, the 'Packet Identifier' and 'Packet Acknowledgment' options (see Section 2.3.2.5 and Section 2.3.2.6, respectively) introduce the packet delivery acknowledgment mechanism, defined in Section 2.5. Further options may add more

security features, such as e. g. congestion control, to EUDP. These options are not defined in this document.

4. IANA Considerations

4.1. 'EUDP Options Numbers' Registry

IANA is asked to create and maintain the registry named 'EUDP Options Numbers Registry' following the guidelines below.

The registry consists of 4 values: Option Kind, Option Length, Name and Reference. They are described below.

Option Kind - an integer; refers to the value used in EUDP options. Values from 0 to 255 are assigned.

Option Length - an integer, 'variable' (for multi-octet options) or 'N/A' (for one-octet options).

Name - contains the name of the option.

Reference - the reference to the document, that defines the option.

The initial values are given in Table 1; new assignments are to be made following the 'RFC Required' policies. [RFC5226]

Kind	Length	Name	Reference
0	N/A	No Operation	RFC xxxx
1	var.	Echo Request	RFC xxxx
2	N/A	End Of Options List	RFC xxxx
3	var.	Echo Response	RFC xxxx
4	4	Packet Identifier	RFC xxxx
5	4	Packet Checksum	RFC xxxx
6	4	Packet Acknowledgment	RFC xxxx
7-252	--	Unassigned	RFC xxxx
253	--	Used for Experimentation	RFC xxxx
254	--	Used for Experimentation	RFC xxxx
255	--	Reserved	RFC xxxx

[RFC Editor: Please replace xxxx with assigned RFC number]

Table 1. Initial contents of the registry

4.2. EUDP Port Numbers Assignment

As EUDP uses the same port set as UDP, IANA is asked to mark the UDP port numbers registry values may be used with EUDP as well.

4.3. IP Protocol Number Assignment

IANA has assigned the IP protocol number TBD1 to be used with EUDP.

5. References

5.1. Normative References

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

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

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

5.2. Informative References

[OSI] International Organization for Standardization (ISO), "Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model," ISO/IEC Standard 7498-1:1994, November 1994.

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

[RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, October 1989.

[RFC1385] Wang, Z., "EIP: The Extended Internet Protocol", RFC 1385, November 1992.

[RFC1475] Ullmann, R., "TP/IX: The Next Internet", RFC 1475, June 1993.

[RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, March 2006.

Appendix A. Acknowledgments

The portions of RFC 793 [RFC0793], whose author - Jon Postel - is acknowledged, are adopted in this document for describing options.

Author's Addresses

Mykyta Yevstifeyev
8 Kuzovkov St., flat 25
Kotovsk
Ukraine

EMail: evnikita2@gmail.com