Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: 25 August 2023

G. Fairhurst
T. Jones
University of Aberdeen

21 February 2023

Datagram PLPMTUD for UDP Options
draft-ietf-tsvwg-udp-options-dplpmtud-05

Abstract

This document specifies how a UDP Options sender implements Datagram Packetization Layer Path Maximum Transmission Unit Discovery (DPLPMTUD) as a robust method for Path Maximum Transmission Unit discovery.

This method uses the UDP Options packetization layer. It allows an application to discover the largest size of datagram that can be sent across a specific network path. It also provides a way to allow the ~~the~~ application to periodically verify the current maximum packet size supported by a path and to update this when required.

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 25 August 2023.

Table of Contents

1. Introduction

The User Datagram Protocol [RFC0768] offers a minimal transport service on top of IP and is frequently used as a substrate for other protocols. Section 3.5 of UDP Guidelines [RFC8085] recommends that applications implement some form of Path MTU discovery to avoid the generation of IP fragments:

"Consequently, an application SHOULD either use the path MTU information provided by the IP layer or implement Path MTU Discovery (PMTUD)".

The UDP API [RFC8304] offers calls for applications to receive ICMP Packet Too Big (PTB) messages and to control the maximum size of datagrams that are sent, but does not offer any automated mechanisms for an application to discover the maximum packet size supported by a path. Upper layer protocols (which includes applications) implement mechanisms for Path MTU discovery above the UDP API.

Packetization Layer Path MTU Discovery (PLPMTUD) [RFC4821] describes a method for a Packetization Layer (PL) (such as UDP Options) to search for the largest Packetization Layer PMTU (PLPMTU) supported on a path. Datagram PLPMTUD (DPLPMTUD) [RFC8899] specifies this support for datagram transports. PLPMTUD and DPLPMTUD gain robustness by using a probing mechanism that does not solely rely on ICMP PTB messages and works on paths that drop ICMP PTB messages.

UDP Options [I-D.ietf-tsvwg-udp-options] supplies functionality that can be used to implement DPLPMTUD within the UDP transport service. This document specifies how DPLPMTUD is implemented (~~see~~ Section 6.1 of [RFC8899]). Implementing DPLPMTUD using UDP Options avoids the need for each upper layer protocol (or application) to implement the DPLPMTUD method. It provides a standard method for applications to discover the current maximum packet size for a path and to detect when this changes.

Commenté [BMI1]: You may cite one or two examples.

2. Terminology

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.

This document uses the terms defined for DPLPMTUD (~~see~~ Sections 2 and 5 of [RFC8899]. It also uses the terms defined in Section 3 of [I-D.ietf-tsvwg-udp-options].

3. DPLPMTUD for UDP Options

The packet formats and procedures for DPLPMTUD using UDP Options are described in this section.

There are two ways an upper layer protocol (i.e., an application or protocol layered above UDP) can perform DPLPMTUD:

* A UDP Options sender implementing DPLPMTUD uses the method specified in [RFC8899] and the upper layer protocol does not perform PMTU discovery. In this case, UDP Options processing is responsible for sending probe packets to determine a PLPMTU, as described in this document. The discovered PLPMTU can be used by UDP Options sender to either:

  - set the maximum datagram size for the current path.

  - set the maximum fragment size when a sender uses the UDP Fragmentation Option (FRAG, Section 9.4 of [I-D.ietf-tsvwg-udp-options]) to divide a datagram into multiple UDP fragments for transmission. The size of ~~Each~~ each UDP fragment is then less than the discovered size of the largest IP packet that can be received across the current path.

* An upper layer protocol using UDP can implement DPLPMTUD. It then uses probe packets ~~packetsn~~ using REQ and RES Options to determine a safe PLPMTU for the datagrams that it sends. The format and content of these probe packets ~~is~~ are determined by the upper layer protocol.

Note: If DPLPMTUD is active at more than one layer, then the values of the tokens used in REQ Options need to be coordinated with any values used for other layers' DPLPMTUD probe packets to ensure that each probe packet can be ~~idnetified~~identified by a unique token. When configurable, a design ought to avoid such performing discovery at ~~thde~~ the UDP options and upper protocol layers. Section 6.1 of [RFC8899] recommends that "An application SHOULD avoid using DPLPMTUD when the underlying transport system provides this capability".

4. Sending UDP-Options Probe Packets

DPLPMTUD relies upon a UDP Options sender sending a probe packet with a specific size, up to the maximum for the size supported by a local interface. This MUST NOT be constrained by the maximum PMTU set by network layer mechanisms (such as discovered by PMTUD [RFC1191][RFC8201] or the PMTU size held in the IP-layer cache), as noted in bullet 2 of Section 3 in [RFC8899]).

Probe packets consume network capacity and incur endpoint processing (~~see~~ Section 4.1 of [RFC8899]). Implementations ought to send a probe packet with a REQ Option only when required by their local DPLPMTUD state machine, i.e., when confirming the base PMTU for the path, probing to increase the PLPMTU, or to confirm the current PLPMTU.

4.1. Sending Probe Packets using the Echo Request (REQ) and Response (RES) Options

The UDP Options used in this document are described in Section 5.11 of [I-D.ietf-tsvwg-udp-options]:

* The REQ Option is set by a sending PL to solicit a response from a remote receiver. A four-byte token identifies each request.

* The RES Option is sent by a UDP Options receiver in response to a previously received REQ Option. Each RES Option echoes a received four-byte token.

* Reception of a RES Option by the sender confirms that a specific probe packet has been received by the remote UDP Options receiver.

Commenté [BMI2]: Introduce first the options

A UDP Options sender ~~sender~~ sends UDP datagrams with ~~the~~ a REQ Option and is prepared to receive~~s~~ datagrams with ~~the~~ a RES Option.

If ~~this~~ UDP Options are~~is~~ not supported by the remote receiver, DPLPMTUD will be unable to confirm the path or to discover the PLPMTU. This will result in the minimum configured PLPMTU (MIN_PLPMTU).

[RFC8899] (Section 3, item 2) requires the network interface below the PL to provide a way to transmit a probe packet that is larger than the PLPMTU without network layer endpoint fragmentation. This document adds to this: UDP datagrams used as DPLPMTUD probe packets, as described in this document, MUST NOT be fragmented at the UDP layer.

The remainder of the section describes a format of a probe packet consisting of an empty UDP datagram, the UDP Options area, and any needed ~~Padding~~padding. Each probe packet includes the UDP Options area containing a RES Option and any other required options concluded with an EOL Option (Section 9.1 of [I-D.ietf-tsvwg-udp-options]) followed by any padding needed to inflate to the required probe size.

~~The UDP Options used in this document are described in Section 5.11 of [I-D.ietf-tsvwg-udp-options]:~~

~~* The REQ Option is set by a sending PL to solicit a response from a remote receiver. A four-byte token identifies each request.~~

~~* The RES Option is sent by a UDP Options receiver in response to a previously received REQ Option. Each RES Option echoes a received four-byte token.~~

~~* Reception of a RES Option by the sender confirms that a specific probe packet has been received by the remote UDP Options receiver.~~

The token allows a UDP Options sender to distinguish between acknowledgements for initial probe packets and acknowledgements confirming receipt of subsequent probe packets (e.g., travelling along alternate paths with a larger ~~round trip~~round-trip time). Each probe packet MUST be uniquely identifiable by the UDP Options sender within the Maximum Segment Lifetime (MSL). The UDP Options sender MUST NOT ~~resuse~~reuse a token value within the MSL. A ~~four byte~~four-byte value for the token field provides sufficient space for multiple unique probe packets to be made within the MSL. Since UDP Options operates over UDP, the token values only need to be unique for the specific 5-tuple over which DPLPMTUD is operating.

The value of the ~~four byte~~four-byte token field SHOULD be initialised to a randomised value to enhance protection from off-path attacks, as described in Section 5.1 of [RFC8085]~~)~~.

Like other UDP options, each of the two option kinds MUST NOT appear more than once in each UDP datagram.

4.2. DPLPMTUD Sender Procedures for UDP Options

DPLPMTUD utilises three types of probe. These are described in the following sections:

* A probe to confirm that ~~a~~the path can support the BASE_PLPMTU (~~see~~ Section 5.1.4 of [RFC8899]).

* A probe to detect whether ~~the~~ a path can support a larger PLPMTU.

* A probe to validate ~~the~~ that a path supports the current PLPMTU.

4.2.1. Confirmation of Connectivity across a Path

The DPLPMTUD method requires a PL to confirm IP connectivity over the path (~~see~~ Section 5.1.4 of [RFC8899]), but UDP itself does not offer a mechanism for this.

UDP Options can provide this required functionality. A UDP Options sender implementing this specification MUST elicit a positive confirmation of connectivity for the path, by sending a probe packet, padded to size BASE_PLPMTU. This confirmation probe MUST include the RES UDP option to elicit a response from the remote endpoint.

Commenté [BMI3]: As many may be available

Reception of a datagram with the corresponding RES Option confirms the reception of a packet of the probed size has successfully traversed the path to the receiver. It also confirms that the remote ~~receiver~~ endpoint supports the RES Option.

4.2.2. Sending Probe Packets to Increase the PLPMTU

From time to time, DPLPMTUD enters the SEARCHING state (Section 5.2 of [RFC8899]) (e.g., after expiry of the PMTU_RAISE_TIMER) to detect whether the current path can support a larger PLPMTU. When the remote endpoint advertises a UDP Maximum Segment Size (MSS) option, this value MAY be used as a hint to initialise this search to increase the PLPMTU.

Probe packets seeking to increase the PLPMTU SHOULD NOT carry application data (see "Probing using padding data" in Section 4.1 of [RFC8899]), since they will be lost whenever their size exceeds the actual PMTU. A probe packet needs to elicit a positive acknowledgment that the path has delivered a datagram of the specific probed size and, therefore, MUST include the REQ Option.

At the receiver, a received probe packet that does not carry application data does not form a part of the end-to-end transport data and is not delivered to the upper layer protocol (i.e., application or protocol layered above UDP).

4.2.3. Validating the Path with UDP Options

A PL using DPLPMTUD needs to validate that a path continues to support the PLPMTU discovered in a previous search for a suitable PLPMTU value (~~see~~ Section 6.1.4 of [RFC8899]). This validation sends probe packets in the DPLPMTUD SEARCH_COMPLETE state to detect black-holing of data (~~see~~ Section 5~~4~~.2 of [RFC8899], which also defines a black~~-~~hole).

Path validation can be implemented within UDP Options, by generating a probe packet of size PLPMTU, which MUST include a REQ Option to elicit a positive confirmation that the path has delivered this probe packet. A probe packet used to validate the path MAY use either "Probing using padding data" or "Probing using application data and padding data" (~~see~~ Section 4.1 of [RFC8899]) or can construct a probe packet that does not carry any application data, as described in ~~a previous section~~Section 4.2.4.

4.2.4. Probe Packets that do not include Application Data

A simple implementation of the method might be designed to only use probe packets in a UDP datagram that ~~include~~ includes no application data. Each probe packet is padded to the required probe size including the REQ Option. This implements "Probing using padding data" (Section 4.1 of [RFC8899]), and avoids having to retransmit application data when a probe fails. In this use, the probe packets do not form a part of the end-to-end transport data and a receiver does not deliver them to the upper layer protocol.

Commenté [BMI4]: I guess this should be 5.2.

Commenté [BMI5]: Consistent with 8899

Commenté [BMI6]: Or 2 ?

4.2.5. Probe Packets that include Application Data

An implementation always uses the format in the previous section when DPLPMTUD searches to increase the PLPMTU.

An alternative format is permitted for a probe packet that confirms connectivity or that validates the path. These probe packets are permitted to carry application data. Note that ~~(~~UDP payload data is permitted because these probe packets perform black~~-~~ hole detection and will ~~therefore~~, therefore, usually have a higher probability of successful transmission~~,~~ similar to other packets sent by the upper layer protocol.~~)~~ Section 4.1 of [RFC8899] provides a discussion of the merits and demerits of including application data. For example, this reduces the need to send additional datagrams.

This type of probe MAY utilise a control message format defined by the upper layer protocol provided that the message does not need to be delivered reliably. The REQ Option MUST be included when a sending upper layer protocol performs DPLPMTUD. The DPLPMTUD method tracks the transmission of probe packets (using the RES Option) and reception of the corresponding RES Options to the upper layer protocol.

A receiver that responds to DPLPMTUD needs to ~~processes~~process the REQ Option and include the corresponding RES Option in an upper layer protocol message that it returns to the requester. DPLPMTUD can be used to manage the PL PMTU in just one direction or can be used for both directions. Probe packets that use this format form a part of the end-to-end transport data.

4.2.6. Examples with different Receiver Behaviors

A receiver that implements UDP Options ought to respond with a UDP datagram with a RES Option when requested by a sender.

The following ~~exampoles~~examples describe different receiver behaviors:

When a sender supports this ~~specificatoon~~specification, but the remote ~~receiver~~endpoint ~~that~~ does not return a RES Option, the method is unable to discover the PLPMTU and will result in using minimum configured PLPMTU (MIN_PLPMTU). Such a remote ~~receiver~~ endpoint might not process UDP options, or might not return a ~~Datagram~~ datagram with a RES Option for some other reasons (due to ~~persisent~~persistent packet loss, ~~insufficent~~insufficient space to include the option, rate-limit policy, etc.)

When both the sender and receiver support DPLPMTUD using ~~the~~the present ~~specication~~specification, and the receiver design only returns a RES Option with next has a UDP datagram to send to the requester. In ~~this~~ such a design, the reception of a REQ Option does not systematically trigger a response. The

design allows DPLPMTUD to operate when there is a flow of datagrams in both directions, even when one direction only provides periodic feedback (e.g., one acknowledgment packet per RTT). Use requires the PLPMTU at the receiver to be sufficiently large that it allows adding the RES option to the feedback packets that are sent. the path. This simple method helps avoid ~~opportunities~~ misusing ~~to use~~ the method as a DoS attack.

When the sender and receiver support DPLPMTUD using the ~~specication~~specification, a receiver could be designed to only return a RES
Option when it next has a UDP datagram to send to the requester, but there is a low rate of transmission (or no datagrams are sent) in the return direction. The lack of transmission ~~opprunities~~opportunities
prevents prompt delivery of the RES Option, and can result in probe packets failing to be acknowledged in time. This will result in a smaller PLPMTU than might be actually ~~suppported~~supported ~~by~~ over the path,
or using the minimum configured PLPMTU (MIN_PLPMTU).

The design of a receiver could permit it to generate a datagram (e.g., with zero payload data) solely to return a RES Option (e.g., when no other Datagrams are queued for transmission). This design allows a UDP Options endpoint to probe the set of open UDP port numbers
using DPLPMTUD probe packets. It results in ~~a~~ some additional traffic overhead, but has the advantage ~~thatb~~ that it can ensure timely
progress of DPLPMTUD. If a UDP Options endpoint creates and sends a ~~Datagram~~ datagram with a RES option solely as respond to received RES
Option, the ~~respionder~~responder MUST limit the rate of these responses
(e.g., limiting each pair of ports to send 1 per RTT or 1 per second). This rate limit mitigates the DoS vector, without significantly impacting the operation of DPLPMTUD.

4.2.7. Changes in the Path

A change in the path or the loss of a probe packet can result in DPLPMTUD updating the PLPMTU. DPLPMTUD [RFC8899] recommends that methods are robust to path changes that could have occurred since the path characteristics were last confirmed and to the possibility of inconsistent path information being received. For example, a notification that a path has changed could trigger path validation to provide black-hole protection Section 4.3 of [RFC8899]).

An upper layer protocol could trigger DPLPMTUD to validate the path when it observes a high packet loss rate (or a repeated protocol timeout).

Section 3 of [RFC8899] requires any methods designed to share the PLPMTU between PLs (such as updating the IP cache PMTU for an interface/destination) to be robust to the wide variety of underlying network forwarding behaviors. For example, an implementation could avoid sharing PMTU information that could potentially relate to packets sent with the same address over a different interface.

Commenté [BMI7]: I don't parse this.

4.3. PTB Message Handling for this Method

Support for receiving ICMP PTB messages is OPTIONAL for use with DPLPMTUD. A UDP Options sender can therefore ignore received ICMP PTB messages.

A UDP Options sender that utilises ICMP PTB messages received in response to a probe packet MUST use the ICMP quoted packet to validate the UDP port information in combination with the token contained in the UDP Option, before processing the packet using the DPLPMTUD method. Section 4.6.1 of [RFC8899] specifies this validation procedure. An implementation unable to support this validation needs to ignore received ICMP PTB messages.

5. Acknowledgements

Gorry Fairhurst and Tom Jones are supported by funding provided by the University of Aberdeen. The editors would like to thank Magnus Westerlund and Mohamed Boucadair for their detailed comments and also other people who contributed to completing this document.

6. IANA Considerations

This memo includes no requests to IANA.

7. Security Considerations

The security considerations for using UDP Options are described in [I-D.ietf-tsvwg-udp-options]. The proposed new method does not change the integrity protection offered by the UDP options method.

The security considerations for using DPLPMTUD are described in [RFC8899]. On path attackers could maliciously drop or modify probe packets to seek to decrease the PMTU, or to maliciously modify probe packets in an attempt to black-hole traffic.

The specification recommends that the token value in the REQ Option is initialised to a randomised value. This is designed to enhance protection from off-path attacks. If a subsequent probe packet uses a token value that is easily derived from the initial value, (e.g., incrementing the value) then a misbehaving on-path observer could then determine the token values used for subsequent probe packets from that sender, even if these probe ~~pakcets~~packets are not transiting via the observer. This would allow probe packets to be forged, with an impact similar to other on-path attacks against probe packets. This attack could be mitigated by using an unpredictable token value for each probe packet.

The proposed new method does not change the ICMP PTB message validation method described by DPLPMTUD: A UDP Options sender that utilities ICMP PTB messages received to a probe packet MUST use the quoted packet to validate the UDP port information in combination with the token contained in the UDP Option, before processing the packet using the DPLPMTUD method.

8. References

8.1. Normative References

[I-D.ietf-tsvwg-udp-options]
Touch, J. D., "Transport Options for UDP", Work in Progress, Internet-Draft, draft-ietf-tsvwg-udp-options-19, 27 December 2022, <https://www.ietf.org/archive/id/draft-ietf-tsvwg-udp-options-19.txt>.

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC0768, August 1980, <https://www.rfc-editor.org/info/rfc768>.

[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>.

[RFC8899] Fairhurst, G., Jones, T., Tüxen, M., Rüngeler, I., and T. Völker, "Packetization Layer Path MTU Discovery for Datagram Transports", RFC 8899, DOI 10.17487/RFC8899, September 2020, <https://www.rfc-editor.org/info/rfc8899>.

8.2. Informative References

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, DOI 10.17487/RFC1191, November 1990, <https://www.rfc-editor.org/info/rfc1191>.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, DOI 10.17487/RFC4821, March 2007, <https://www.rfc-editor.org/info/rfc4821>.

[RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085, March 2017, <https://www.rfc-editor.org/info/rfc8085>.

[RFC8201] McCann, J., Deering, S., Mogul, J., and R. Hinden, Ed., "Path MTU Discovery for IP version 6", STD 87, RFC 8201, DOI 10.17487/RFC8201, July 2017, <https://www.rfc-editor.org/info/rfc8201>.

[RFC8304] Fairhurst, G. and T. Jones, "Transport Features of the User Datagram Protocol (UDP) and Lightweight UDP (UDP-Lite)", RFC 8304, DOI 10.17487/RFC8304, February 2018, <https://www.rfc-editor.org/info/rfc8304>.

Appendix A. Revision Notes

XXX Note to RFC-Editor: please remove this entire section prior to publication. XXX

Individual draft-00.

* This version contains a description for consideration and comment

by the TSVWG.

Individual draft-01.

* Address Nits

* Change Probe Request and Probe Reponse options to Echo to align names with draft-ietf-tsvwg-udp-options

* Remove Appendix B, Informative Description of new UDP Options

* Add additional sections around Probe Packet generation

Individual draft-02.

* Address Nits

Individual draft-03.

* Referenced DPLPMTUD RFC.

* Tidied language to clarify the method.

Individual draft-04

* Reworded text on probing with data a little

* Removed paragraph on suspending ICMP PTB suspension.

Working group draft-00

* -00 First Working Group Version

* RFC8899 call search_done SEARCH_COMPLETE, fix

Working group draft -01

* Update to reflect new fragmentation design in UDP Options.

* Add a description of uses of DPLPMTUD with UDP Options.

* Add a description on how to form probe packets with padding.

* Say that MSS options can be used to initialise the search algorithm.

* Say that the recommended approach is to not use user data for probes.

* Attempts to clarify and improve wording throughout.

* Remove text saying you can respond to multiple probes in a single packet.

* Simplified text by removing options that don't yield benefit.

Working group draft -02

* Update to reflect comments from MED.

* More consistent description of DPLPMTUD with UDP Options.

* Clarify the nonce value (token) is intended per 5-tuple, not interface.

* BASE_PLPMTU related to RFC8899.

* Probes with user data can carry application control data.

* Added that application data uses RES and REQ nonce (token) values from the app.

* QUIC was intended as an informational reference to an example of RFC8899.

Working group draft -03

* Update to reflect more comments from MED.

* Again more consistent description of DPLPMTUD with UDP Options.

* Clarify token/nonce to use "token".

* Clarify any use of application data for black-hole detection.

* Minor changes to reflect update to UDP Options base spec.

Working group draft-04.

Update for WG Last Call

Working group draft-05.

Update following WG Last Call

Authors' Addresses

Godred Fairhurst
University of Aberdeen
School of Engineering
Fraser Noble Building
Aberdeen
AB24 3UE
United Kingdom
Email: gorry@erg.abdn.ac.uk

Tom Jones
University of Aberdeen
School of Engineering
Fraser Noble Building
Aberdeen
AB24 3UE
United Kingdom
Email: tom@erg.abdn.ac.uk