Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: April 3, 2021

G. Fairhurst
T. Jones
University of Aberdeen

September 30, 2020

Datagram PLPMTUD for UDP Options
draft-fairhurst-tsvwg-udp-options-dplpmtud-03

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 is a robust method for Path MTU Discovery (PMTUD) that uses the UDP Options Packetization Layer (PL). It allows a datagram application that uses this PL, to discover the largest size of datagram that can be sent across a network path.

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] offer calls for applications to receive ICMP Packet Too Big (PTB) messages and to control the maxmium 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. Applications and upper layer protocols implement mechanisms for path MTU discovery above the UDP API.

Packetization Layer PMTUD (PLPMTUD) [RFC4821] describes a method for a Packetization Layer (PL) (such as UDP with 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 use a probing mechanism that does not solely rely on ICMP PTB messages and works in the presence of lost probes.

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 this additional functionality. Implementing DPLPMTUD using UDP Options avoids the need for each upper layer protocol or application to implement the DPLPMTUD method. This provides a standard method for applications to discover the current maximum packet size for a path and to detect when this changes.

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The structure of the present document follows the structure used to describe DPLPMTUD for other transports [RFC8899].

3. DPLPMTUD for UDP Options

The DPLPMTUD PL endpoint implements the method specified in [RFC8899].

3.1. Confirmation of Connectivity across a Path

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

UDP Options can provide this required functionality. A UDP Options sender implementing this specification SHOULD elicit a positive confirmation of connectivity of the path, using a suitable confirmed UDP Option (i.e., Timestamps, ECHO Request/Response) to .

3.2. Sending UDP-Options Probe Packets

DPLPMTUD relies upon the ability of a sender PL to generate probe packets with a specific size, and to confirm when these are delivered across the path. Therefore, a UDP options sender needs to be able to send probes up to the maximum for the size the local interface supports, which MUST NOT be further constrained by the maximum PMTU set by network layer mechanisms (such as PMTUD [RFC1063][RFC8201]).

DPLPMTUD needs to be able to generate probe packets that are not delivered to the upper layer protocol as a part of the end-to-end transport data (i.e. ensure any added padding data is not delivered to the upper layer protocol at the receiver). UDP Options provide the necessary additional support required to do this within the transport layer.

There are various designs described in DPLPMTUD to send a Packet Probe to test the size of packet supported by a path (see Section 4.1 of [RFC8899]). This prevents "Probing using padding data" or "Probing using application data and padding data" (see Section 4.1 of [RFC8899]).

A PL needs to determine whether the current path supports datagrams used as Probe Packets. DPLPMTUD SHOULD send (or add) a UDP Option (e.g., Timestamps, ECHO Request/Response) to a Packet Probe to elicit a positive confirmation that the path has delivered the Probe Packet of the corresponding size. From time to time, such probes can also be used to determine whether the current path can support a larger size of datagram that the current PLPMTU.

A PL also needs to determine that the current path supports the size of datagram that the application is currently sending when in the DPLPMTUD search_done state i.e., to detect black-holing of data (see Section 4.2 of [RFC8899]). UDP Options can provide this by eliciting a positive confirmation that the path has delivered a Datagram of the corresponding size.

3.2.1. Sending Packet Probes using the Echo Request Option Request Option

The RECOMMENDED method sends a Probe Packet with the Echo Request Option (RES) together with any padding needed to inflate the required size. The reception of this option generates an Echo Response Option that confirms reception of each received Probe Packet.

Probe Packets consume network capacity and incur endpoint processing (see Section 4.1 of [RFC8899]). Implementations ought to send a Probe Packet with a Request Probe Option only when required by their local DPLPMTUD state machine, i.e., when probing to grow the PLPMTU or to confirm the current PLPMTU.

Implementations MAY track multiple requests and respond acknowledging them with a single packet.

The UDP Options used in this method are described in section 6 of [I-D.ietf-tsvwg-udp-options]:

- The Echo Request Option (RES) is set by a sending PL to solicit a response from a remote endpoint. A four-byte token identifies each request.

- The Echo Response Option (REQ) is generated by the UDP Options receiver in response to reception of a previously received Echo Request Option. Each Echo Response Option echoes a previously received four-byte token.

The token value allows implementations 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 time). This needs each Probe Packet needs to be uniquely identifiable by the UDP Options sender within the Maximum Segment Lifetime (MSL). The UDP Options sender therefore MUST NOT recycle token values until they have expired or have been acknowledged. A four byte value for the token field provides sufficient space for multiple unique probes to be made within the MSL.

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

The procedure to handle the loss of a datagram is the responsibility of the sender of the request. Implementations MAY track multiple requests and respond to them with a single packet carrying the Echo Response Option (REQ).

3.2.2. Sending Packet Probes that include Application Data

The RECOMMENDED approach to generating a Probe Packet is to send a probe formed of a UDP Options datagram contains only control information, padded to the size required for the probe. This allows "Probing using padding data", and avoids having to retransmit application data when a probe fails.

If an application/transport needs protection from the loss of data in the Probe Packet payload, the application/ transport could perform transport-layer retransmission/repair of the data block (e.g., by retransmission after loss is detected or by duplicating the data block in a datagram without the padding) [RFC8085].

3.3. Validating the Path with UDP Options

A PL also needs to validate that the path continues to support the PLPMTU discovered in a previous search for a suitable PLPMTU value (see Section 6.1.4 of [RFC8899]). This confirmation MAY be provided

by an upper layer protocol confirming correct reception of data by the remote PL, but there is no generic mechanism to access this upper layer information.

This function can be implemented within UDP Options, by generating a Probe Packet of size PLPMTU to confirm the path. This Probe Packet MUST elicit a response from the remote PL and could use either the ECHO Response Option or the TimeStamp option (see Section 5.9 [I-D.ietf-tsvwg-udp-options]).

A sender MAY choose to include application data in Probe Packets (see Section 4.1 of [RFC8899] discusses the merits and demerits of this approach). For example, this might reduce the need to send an additional datagram when confirming that the current path supports datagrams of size PLPMTU.

3.3.1. Sending Packet Probes using Timestamps

Reception of a valid Timestamp Option echoed by the remote endpoint can be used to infer connectivity. It can also confirm that packets of the current size are being received by the remote PL. This can provide useful feedback, even over paths with asymmetric capacity and/or that carry UDP Option flows that have asymmetric datagram rates, because an echo of the most recent timestamp still indicates reception of at least one packet of the transmitted size. This is sufficient to confirm there is no black hole (see Section 2.1 of [RFC2923]).

When sending a probe to increase the PLPMTU, such a Timestamp might be unable to unambiguously identify that a specific Probe Packet has been received [KP87]. Timestamp mechanisms therefore cannot be used to confirm the reception of individual probe messages and cannot be used to stimulate a response from the remote peer.

Note: Probe Packets used to search for a larger PLPMTU MUST include the Echo Request Option.

3.4. PTB Message Handling for this Method

A UDP Options sender can ignore received ICMP PTB messages, and this support is OPTIONAL for use with DPLPMTUD.

A UDP Options sender that utilises ICMP PTB messages received to a Probe Packet MUST use the quoted packet to validate the UDP port information in combination with the token and/or timestamp value contained in the UDP Option, before processing the packet using the DPLPMTUD method (see Section 4.4.1 of [RFC8899]). An implementation

unable to support this validation needs to ignore received ICMP PTB messages.

As in other implementations of DPLPMTUD, a PL implementing this specification MUST suspend processing of ICMP PTB messages by the network layer (as specified in PMTUD [RFC1191] [RFC8201]) for each flow that utilises DPLPMTUD.

4. Acknowledgements

Gorry Fairhurst and Tom Jones are supported by funding provided by the University of Aberdeen.

5. IANA Considerations

This memo includes no requests to IANA.

6. 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 specification recommends that the token in the REQ/RES message is initialised to a randomised value to enhance protection from off-path attacks.

The security considerations for using DPLPMTUD are described in [RFC8899]. The proposed new method does not change the ICMP PTB message validation method described DPLPMTUD: A UDP Options sender that utilises ICMP PTB messages received to a Probe Packet MUST use the quoted packet to validate the UDP port information in combination with the token and/or timestamp value contained in the UDP Option, before processing the packet using the DPLPMTUD method.

7. References

7.1. Normative References

[I-D.ietf-tsvwg-udp-options]
Touch, J., "Transport Options for UDP", draft-ietf-tsvwg-udp-options-08 (work in progress), September 2019.

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

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

[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., Tuexen, M., Ruengeler, I., and T. Voelker, "Packetization Layer Path MTU Discovery for Datagram Transports", RFC 8899, DOI 10.17487/RFC8899, September 2020, <https://www.rfc-editor.org/info/rfc8899>.

7.2. Informative References

[KP87] Karn, P. and C. Partridge, "Improving Round-Trip Time Estimates in Reliable Transport Protocols", 1987.

[RFC1063] Mogul, J., Kent, C., Partridge, C., and K. McCloghrie, "IP MTU discovery options", RFC 1063, DOI 10.17487/RFC1063, July 1988, <https://www.rfc-editor.org/info/rfc1063>.

[RFC2923] Lahey, K., "TCP Problems with Path MTU Discovery", RFC 2923, DOI 10.17487/RFC2923, September 2000, <https://www.rfc-editor.org/info/rfc2923>.

[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.

Authors' Addresses

Godred Fairhurst
University of Aberdeen
School of Engineering
Fraser Noble Building
Aberdeen AB24 3UE
UK

Email: gorry@erg.abdn.ac.uk

Tom Jones
University of Aberdeen
School of Engineering
Fraser Noble Building
Aberdeen AB24 3UE
UK

Email: tom@erg.abdn.ac.uk