Workgroup: Network Working Group
Internet-Draft: draft-templin-6man-fragrep-02
Updates: RFC8200, RFC8201, RFC4443, RFC1191
(if approved)
Published: 17 November 2021
Intended Status: Standards Track
Expires: 21 May 2022
Authors: F. L. Templin, Ed.
Boeing Research & Technology

IPv6 Fragment Retransmission and Path MTU Discovery Soft Errors

Abstract

Internet Protocol version 6 (IPv6) provides a fragmentation and reassembly service for end systems allowing for the transmission of packets that exceed the path MTU. However, loss of just a single fragment requires retransmission of the original packet in its entirety, with potentially devastating effects on performance. This document specifies an IPv6 fragment retransmission scheme that matches the loss unit to the retransmission unit. The document further specifies an update to Path MTU Discovery that distinguishes hard link size restrictions from reassembly congestion events.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Internet Protocol version 6 (IPv6) [RFC8200] provides a fragmentation and reassembly service similar to that found in IPv4 [RFC0791], with the exception that only the source host (i.e., and not routers on the path) may perform fragmentation. When an IPv6 packet is fragmented, the loss unit (i.e., a single IPv6 fragment) becomes smaller than the retransmission unit (i.e., the entire packet) which under intermittent loss conditions could result in sustained retransmission storms with little or no forward progress [RFC8900].

The presumed drawbacks of fragmentation are tempered by the fact that greater performance can often be realized when the source sends large packets that exceed the path MTU. This is due to the fact that a single large IPv6 packet produced by upper layers results in a burst of multiple fragment packets produced by lower layers with minimal inter-packet delays. These bursts yield high network utilization for the burst duration, while modern reassembly implementations have proven capable of accommodating such bursts. If the loss unit can somehow be made to match the retransmission unit, the performance benefits of IPv6 fragmentation can be realized.

This document therefore proposes an IPv6 fragment retransmission service in which the source marks each fragment with an "Ordinal" number, and the destination may request retransmissions of any ordinal fragments that are lost. This retransmission request service is intended only for short-duration and opportunistic best-effort recovery (i.e., and not true end-to-end reliability). In this way, the service mirrors the Automatic Repeat Request (ARQ) function of common data links [RFC3366] by considering an imaginary virtual link that extends from the IPv6 source to destination. The goal therefore is for the destination to quickly obtain missing individual fragments of partial reassemblies before true end-to-end timers would cause retransmission of the entire packet.

When conditions suggest that original sources should begin sending smaller packets, the fragmentation source and/or reassembly destination can return a new type of ICMPv6 Packet Too Big or ICMPv4 Fragmentation Needed message termed a PTB "soft error" that is distinguished from classic "hard errors" by including a non-zero value in the PTB Code (ICMIPv6) or unused (ICMPv4) field. The fragmentation source can return soft errors (subject to rate limiting) suggesting a smaller packet size while fragmentation of large packets is producing excessive numbers of fragments. Similarly, the reassembly destination can return soft errors (via the fragmentation source) while reassembly of large packets is causing excessive reassembly congestion. Original sources that receive these soft errors should reduce the size of packets they send for the short term, but can again begin to increase their packet sizes without delay as long as no further soft or hard errors arrive.

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.

3. Common Use Cases

A common use case of interest is to improve the state of affairs for IPv6 encapsulation (i.e., "tunneling") [RFC2473] when the original source may be many IP hops away from the tunnel ingress, and the tunnel packet may be fragmented following encapsulation. The tunnel is seen as a "link" on the path from the original source to the final destination, and the goal is to increase the reliability of that link in order to minimize wasteful end-to-end retransmissions.

When the original source and IPv6 fragmentation source are located on the same platform (physical or virtual) the window of opportunity for successful retransmission of individual fragments may be narrow unless the link persistence timeframe is carefully coordinated with upper layer retransmission timers. (In an uncoordinated case, upper layers may retransmit the entire packet before or at roughly the same time the IPv6 fragmentation source retransmits individual fragments, leading to increased congestion and wasted retransmissions.)

4. IPv6 Fragmentation

IPv6 fragmentation is specified in Section 4.5 of [RFC8200] and is based on the IPv6 Fragment extension header formatted as shown below:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Next Header  |   Reserved    |      Fragment Offset    |Res|M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Identification                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

In this format:

*Next Header is a 1-octet IP protocol version of the next header following the Fragment Header.

*Reserved is a 1-octet reserved field set to 0 on transmission and ignored on reception.

*Fragment Offset is a 13-bit field that provides the offset (in 8-octet units) of the data portion that follows from the beginning of the packet.

*Res is a 2-bit field set to 0 on transmission and ignored on reception.

*M is the "More Fragments" bit telling whether additional fragments follow.

*Identification is a 32 bit numerical identification value for the entire IPv6 packet. The value is copied into each fragment of the same IPv6 packet.

The fragmentation and reassembly specification in [RFC8200] can be considered as the standard method which adheres to the details of that RFC. This document presents an enhanced method that allows for retransmissions of individual fragments.

5. IPv6 Fragment Retransmission

Fragmentation implementations that obey this specification write an "Ordinal" value beginning with 0 and monotonically incremented for each successive fragment in the (formerly) "Reserved" field of the

IPv6 Fragment Header, which is redefined as a 6-bit "Ordinal" field followed by a 1-bit R(eserved) flag followed by a 1-bit A(RQ) flag as shown below:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Next Header  |  Ordinal  |R|A|      Fragment Offset    |Res|M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Identification                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

In particular, when a source that obeys this specification fragments an IPv6 packet it sets the Ordinal value for the first fragment to '0', the Ordinal value for the second fragment to '1', the Ordinal value for the third fragment to '2', etc. up to either the final fragment or the 64th fragment (whichever comes first). The source also sets the A flag to 1 in each fragment to inform the destination that fragment retransmission is supported for this packet.

When a destination that obeys this specification receives IPv6 fragments with the A flag set to 1, it infers that the source participates in the protocol and maintains a checklist of all Ordinal numbered fragments received for a specific Identification number.

If the destination notices one or more Ordinals missing after most other Ordinals for the same Identification have arrived, it can prepare an ICMPv6 Fragmentation Report (FRAGREP) message [RFC4443] to send back to the source. The message is formatted as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |     Code      |          Checksum             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Identification (0)                     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Ordinal Bitmap (0) (0-31)                  |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Ordinal Bitmap (0) (32-63)                 |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Identification (1)                     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Ordinal Bitmap (1) (0-31)                  |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Ordinal Bitmap (1) (32-63)                 |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                ...                            |
   |                                ...                            |
```

In this format, the destination prepares the FRAGREP message as a list of 12-octet (Identification(i), Bitmap(i)) pairs. The first 4 octets in each pair encode the Identification value for the IPv6 packet that is subject of the report, while the remaining 8 octets encode a 64-bit Bitmap of Ordinal fragments received for this Identification. For example, if the destination receives Ordinals 0, 1, 3, 4, 6, and 8 it sets Bitmap bits 0, 1, 3, 4, 6 and 8 to '1' and sets all other bits to '0'. The destination may include as many (Identification, Bitmap) pairs as necessary without causing the entire message to exceed the minimum IPv6 MTU of 1280 bytes. (If additional pairs are necessary, the destination may prepare and send multiple messages.)

The destination next transmits the FRAGREP message to the IPv6 fragment source. When the source receives the message, it examines each entry to determine the per-Identification Ordinal fragments that require retransmission. For example, if the source receives a Bitmap for Identification 0x12345678 with bits 0, 1, 3, 4, 6 and 8 set to '1', it would retransmit Ordinal fragments (0x12345678, 2), (0x12345678, 5) and (0x12345678, 7).

This implies that the source should maintain a cache of recently transmitted fragments for a time interval known as "link persistence" [RFC3366]. The link persistence should be at least as long as the round-trip time from the fragmentation source to the reassembly destination, plus an additional small delay to allow for reassembly processing overhead. Then, if the source receives a FRAGREP message requesting retransmission of one or more Ordinals, it can retransmit if it still holds the Ordinal in its cache. Otherwise, the Ordinal will incur a cache miss and the original source will eventually retransmit the original packet in its entirety. After processing all entries in the FRAGREP, the source discards the message.

Note that the maximum-sized IPv6 packet that a source can submit for fragmentation is 64KB, and the minimum IPv6 path MTU is 1280B. Assuming the minimum IPv6 path MTU as the nominal size for non-final fragments, the number of Ordinals for each IPv6 packet should therefore fit within the allotted 64 Bitmap bits when the fragments are transmitted over IPv6-only network paths. However, when the path may traverse one or more IPv4 networks (e.g., via tunneling) the path MTU may be significantly smaller. In that case, the number of IPv6 fragments needed may exceed the maximum number of Ordinal candidates for retransmission (i.e., 64).

When the number of IPv6 fragments exceeds 64, the source assigns an Ordinal value and sets A to 1 in the first 64 fragments, but sets both Ordinal and A to 0 in all remaining fragments then transmits all fragments. When the destination receives the fragments, it may return a FRAGREP to request retransmission of any of the first 64 fragments, but may not request retransmission of any additional fragments for which the default behavior of best-effort delivery applies. (However, all fragments are presented equally to the reassembly cache where successful reassembly is likely.)

Finally, transmission of IPv6 fragments over IPv6-only paths can safely proceed without a fragmentation-layer integrity check since IPv6 includes reassembly safeguards and a 32-bit Identification value. Conversely, transmission of IPv6 fragments over IPv4-only or mixed IPv6/IPv4 paths requires a fragmentation-layer integrity check inserted by the source before fragmentation and verified by the destination following reassembly since IPv4 provides only a 16-bit Identification and no reassembly safeguards. (In cases where the full path cannot be determined a priori, an integrity check should always be included as specified in AERO [I-D.templin-6man-aero] and OMNI [I-D.templin-6man-omni].)

6. Packet Too Big (PTB) Soft Errors

When an IPv6 fragmentation source forwards packets that produce what it considers as excessive numbers fragments (e.g., 32, 48, 64, more), the fragmentation source can also return PTB "soft errors" to the original source (subject to rate limiting). Either the fragmentation source or reassembly destination may also return PTB soft errors if the frequency of retransmissions or reassembly failures exceeds acceptable thresholds.

PTB soft errors are distinguished from ordinary "hard errors" through a non-zero value in the ICMPv6 "Code" field [RFC8201] [RFC4443] or ICMPv4 "unused" field [RFC1191]. The following values are currently defined:

*0 - "PTB hard error" - Original sources that receive these messages obey the classic Path MTU Discovery (PMTUD) specifications found in [RFC8201][RFC1191].

*1 - "PTB soft error (packet lost)" - Original sources that receive these messages should reduce their packet sizes while retransmitting the data from the lost packet, but need not wait the prescribed 10 minutes before attempting to again increase packet sizes.

*2 - "PTB soft error (packet forwarded)" - Original sources that receive these messages should reduce their packet sizes without invoking retransmission, and also need not wait the prescribed 10 minutes before attempting to again increase packet sizes.

*3-255 - reserved for future use.

PTB soft errors include as much of the invoking packet as possible without the message exceeding the minimum MTU (i.e., 1280 bytes for IPv6 or 576 bytes for IPv4). Original sources that recognize PTB soft errors should follow common logic to dynamically tune their packet sizes to obtain the best performance. In particular, an original source can gradually increase the size of packets it sends while no or few PTB soft errors are arriving then again reduce packet sizes when excessive soft errors arrive.

Original sources that do not recognize PTB soft errors (i.e., that do not examine the Code/unused field value) follow the same standards as for hard errors as described above. These sources may miss opportunities to realize improved performance.

7. Implementation Status

TBD.

8. IANA Considerations

A new ICMPv6 Message Type code for "Fragmentation Report (FRAGREP)" is requested.

The IANA is instructed to create new registries for "ICMPv6 Packet Too Big Code field" and "ICMPv4 Fragmentation Needed unused field" values. Both registries should have the following initial values:

Value	Sub-Type name	Reference
0	PTB hard error	[RFCXXXX]
1	PTB soft error (loss)	[RFCXXXX]
2	PTB soft error (no loss)	[RFCXXXX]
3-252	Unassigned	
253-254	Reserved for Experimentation	[RFCXXXX]
255	Reserved by IANA	[RFCXXXX]

Figure 1: Packet Too Big Code/unused Values

9. Security Considerations

Communications networking security is necessary to preserve confidentiality, integrity and availability.

10. Acknowledgements

This work was inspired by ongoing AERO/OMNI/DTN investigations.

.

11. References

11.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[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>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[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>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[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>.

11.2. Informative References

[I-D.templin-6man-aero] Templin, F. L., "Automatic Extended Route Optimization (AERO)", Work in Progress, Internet-Draft, draft-templin-6man-aero-36, 25 October 2021, <https://www.ietf.org/archive/id/draft-templin-6man-aero-36.txt>.

[I-D.templin-6man-omni] Templin, F. L. and T. Whyman, "Transmission of IP Packets over Overlay Multilink Network (OMNI) Interfaces", Work in Progress, Internet-Draft, draft-templin-6man-omni-49, 25 October 2021, <https://www.ietf.org/archive/id/draft-templin-6man-omni-49.txt>.

[RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", RFC 2473, DOI 10.17487/RFC2473, December 1998, <https://www.rfc-editor.org/info/rfc2473>.

[RFC8900] Bonica, R., Baker, F., Huston, G., Hinden, R., Troan, O., and F. Gont, "IP Fragmentation Considered Fragile", BCP 230, RFC 8900, DOI 10.17487/RFC8900, September 2020, <https://www.rfc-editor.org/info/rfc8900>.

Author's Address

Fred L. Templin (editor)
Boeing Research & Technology
P.O. Box 3707
Seattle, WA 98124
United States of America

Email: fltemplin@acm.org