Network Working Group
Internet-Draft
Updates: rfc8900 (if approved)
Intended status: Informational
Expires: 21 December 2025

F. L. Templin, Ed.
Boeing Research & Technology

19 June 2025

Fragmentation Revisited: For What It's Worth
draft-templin-6man-fwiw-03

Abstract

Internet Protocol (IP) fragmentation and reassembly have served as core elements of the architecture from the very earliest days but they have been subject to negative publicity by studies that have declared them "harmful" and "fragile". These warning labels have resonated deeply within the community in a way that fosters the enemies of sound engineering: fear, uncertainty and doubt. This document revisits IP fragmentation and shows that a properly engineered alternative IPv6 solution is both practical and necessary to provide a robust service for the future of Internetworking.

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 21 December 2025.

Table of Contents

1. Introduction

Internet Protocol (IP) fragmentation and reassembly have served as core elements of the architecture from the very earliest days but they have been subject to negative publicity by studies that have declared them "harmful" and "fragile". This is true for both IPv4 [RFC0791] and IPv6 [RFC8200] which employ a common set of fragmentation parameters that differ only in the length of their respective Identification fields.

Beginning in the late 1980's, "Fragmentation Considered Harmful" [KENT87][MOGUL19] inspired an investigation into path Maximum Transmission Unit (MTU) discovery that has only recently reached successful conclusions [RFC4821][RFC8899]. Still more recently, "IP Fragmentation Considered Fragile" [RFC8900] documented enough protocol and operational concerns to merit BCP status. However, these studies failed to observe that the shortcomings identified can be addressed by sound engineering solutions in support of a more robust service. The warning labels have instead inspired myths, folklore and urban legends that have caused deeply-embedded misgivings to carry forward from generation to generation in the Internet engineering community.

This document examines fragmentation and reassembly within the context of the larger multi-layer Internetworking architecture for transferring data between peer end systems. A systematic examination instead of an exclusive focus on the network layer services viewed in isolation is needed to set the appropriate context. The document concludes that an improved IPv6 fragmentation and reassembly service is both practical and necessary while citing a specification that offers a robust alternative solution.

2. Multi-Layer Internetworking Architecture

Figure 1 illustrates a common architecture where an Original Source host connects to End User Network (EUN) A and a Final Destination host connects to EUN B. An Encapsulation Source for EUN A in turn connects to an Encapsulation Destination for EUN B via a virtual link spanning any intermediate Internetworks.

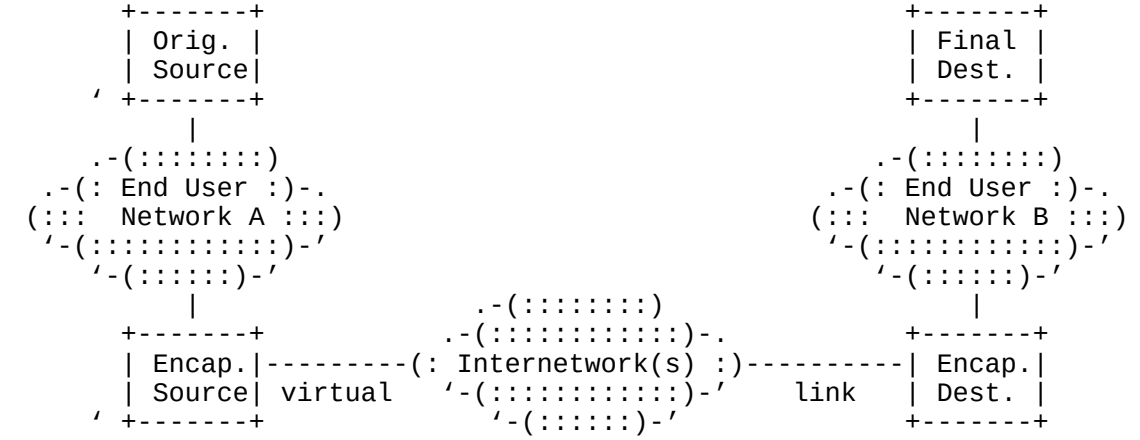

Figure 1: Multi-Layer Internetwork Architecture

Applications on the Original Source prepare user data buffers for presentation to lower layers. In environments where delay or disruption may be significant, user data buffers may be based on "bundles" according to the Delay Tolerant Networking (DTN) Bundle Protocol (BP) [RFC9171]. The user data buffers (or bundles) may range in size from very small for interactive communications such as Internet telephony to very large for transactional services such as large file transfer.

Original source applications present user data to transport layer protocols such as TCP [RFC9293], QUIC [RFC9000] and others which divide them into segments. Traditional wisdom suggests that each

segment should be no larger than the path MTU and that each IP packet should carry exactly one segment. But, a more robust IPv6 fragmentation and reassembly service would permit segment sizes to exceed the path MTU and IP parcels [I-D.templin-6man-parcels2] would permit a single packet to carry multiple segments. The resulting services can support improved performance profiles for common operational environments.

Following segmentation, the transport layer protocol may present either a single segment or a parcel of multiple segments to the network layer for packetization. If each packet will contain a single segment, the network layer applies traditional packetization procedures while engaging a fragmentation-like service known as Generic Segment Offload (GSO) to packetize multi-segment parcels if necessary. If each packet will contain multiple segments, the network layer instead prepares the packet with a multi-segment transport layer parcel to form an IP parcel. The network layer then applies IP fragmentation if necessary and forwards the IP packets or fragments into EUN A. Standard IP forwarding within the local EUN will then deliver the packets/fragments either directly to the Final Destination or to an Encapsulation Source when the Final Destination resides in a different EUN.

Unlike the Original Source, the Encapsulation Source operates only at the network layer and below. The Encapsulation Source encapsulates any original packets or fragments bound for a remote destination according to [RFC2473] then forwards the resulting packets to an appropriate Encapsulation Destination as an adaptation layer service. As a source, the Encapsulation Source can apply adaptation layer IPv6 fragmentation following encapsulation to ensure that the resulting fragments will not be lost due to a size restriction in an intermediate Internetwork. The Encapsulation Destination is then responsible for reassembling these fragments.

After the Encapsulation Destination performs IPv6 reassembly at the adaptation layer, it decapsulates to obtain the original IP packet or fragments which it then forwards to the Final Destination. The Final Destination then reassembles at the network layer if necessary using IP reassembly and/or Generic Receive Offload (GRO) according to the manner in which the Original Source applied fragmentation. The Final Destination then delivers the resulting segment or parcel to the transport layer which delivers the resulting user data to the application layer.

3. Discussion

IP fragmentation may be applied by the Original Source, the Encapsulation Source and/or by a router in the network. IP reassembly may similarly be applied by a router in the network, the Encapsulation Destination and/or by the Final Destination. It is very important to distinguish between "source fragmentation" and "network fragmentation" to understand that the latter form should be avoided.

First, routers in the network are beyond the control of the source and may apply IP fragmentation in unpredictable ways that may not match the manner in which the source intended. Second, routers in the network are fundamentally challenged to perform reassembly at line rates when the reassembly process is relegated to the slow path. Both of these conditions can result in poor or even disrupted performance not appropriate for modern networks. For this reason, only source fragmentation and destination reassembly are considered.

To efficiently engage fragmentation, each source requires a means to determine the per-flow path MTU. From the above architecture, the contributing elements include: 1) the path MTU from the Original Source to the Encapsulation Source, 2) the path MTU from the Encapsulation Source to the Encapsulation Destination, and 3) the path MTU from the Encapsulation Destination to the Final Destination. In common use cases, however, EUNs often comprise limited domains with robust link MTUs while intervening Internetworks may be arbitrarily complex including heterogeneous links with widely varying MTUs possibly as small as the minimum IP link MTU.

More specifically, the EUN path MTUs are visible at the network layer while the path MTU between the encapsulation endpoints is visible only at the adaptation layer and presents the appearance of a single virtual link to the network layer. This suggests a multi-layer path MTU probing discipline is required, where both the Original Source probes the Final Destination at the network layer and the Encapsulation Source probes the Encapsulation Destination at the adaptation layer. The former allows the original source to determine the largest original IP packet or fragment size that can traverse the entire path to the destination while the latter allows the encapsulation source to determine the maximum adaptation layer fragment size for the encapsulation destination.

The Original Source is often positioned within a small number of EUN hops from the Encapsulation Source and in some cases the two functions may even be co-resident on the same physical platform. The same is true of the Encapsulation Destination and Final Destination. When the Original Source and Encapsulation Source are co-resident,

multiple layers of fragmentation may be needed for the same original packet before the resulting fragment packets are transmitted over the physical or virtual data link media. When the Encapsulation Destination and Final Destination are co-resident, multiple layers of reassembly may be needed before the reassembled segments or parcels are delivered to upper layers. This may require the operating system to perform expensive buffer linearization following initial stages of reassembly before presenting them to additional stages. The original IP and encapsulation services could alternatively be located in different physical platforms or in separate virtual machines on the same physical platform so that each operating system instance engages only a single layer of fragmentation or reassembly.

While the Encapsulation Source is fragmenting at the adaptation layer, it should probe the forward path to the Encapsulation Destination to determine the largest fragment size that can traverse the intermediate Internetworks for each flow. At the same time, the Encapsulation Source can represent an unbounded MTU to the network layer in order to accommodate all original IP packets up to 65535 octets with fragmentation-assured delivery while allowing larger original IP packets to proceed without fragmentation based on best-effort delivery.

Alternatively, the Encapsulation Source could refrain from performing fragmentation and adaptation layer path MTU probing while forwarding all encapsulated packets to the Encapsulation Destination based on best-effort delivery regardless of their size. This means that the network layer path MTU probing between the Original Source and Final Destination would also have the effect of probing the path between the Encapsulation Source and Destination over the intermediate Internetworks. The drawback of this approach is that the link MTUs in the intermediate Internetworks are often beyond the control of the source and destination endpoints and will often configure MTU sizes that are significantly smaller than those in the EUNs. Such an arrangement will often fail to benefit from the larger native MTU sizes of the EUNs.

4. Fragmentation Issues

Considerable evidence suggests that aspects of standard fragmentation for both IPv4 [RFC0791] and IPv6 [RFC8200] can result in a fragile service in some environments [RFC8900]. For example, IPv4 provides only a 16-bit Identification field which means that a source must limit its transmission rate to avoid wrapping the Identification value within the Maximum Datagram Lifetime (MDL) which may be multiple orders of magnitude too slow for modern networks [RFC4963][RFC6864]. IPv6 addresses this issue by providing a 32-bit Identification field, but reuse within the MDL is still possible when

the source resets the Identification sequence frequently to avoid predictable values [RFC7739].

Additionally, both IPv4 and IPv6 fragmentation can produce non final fragments of differing lengths as small as 8 octets (while the final fragment may be smaller still) and the offsets of later fragments may overlap with those of earlier fragments. This unconstrained (offset, length) relationship further allows intermediate systems to perform gratuitous fragmentation on individual fragments in isolation without first (virtually) reassembling all fragments of the same packet. This is true for IPv4 even when the DF bit is set and also true for IPv6 even though the standard deprecates intermediate system fragmentation. Accordingly, there is no assurance that the fragment sizes produced by the source will match the fragment sizes that arrive at the destination.

The number of fragments per packet is furthermore bounded only by the size of the original packet divided by the 8 octet minimum fragment size. For example, an 8KB packet could be fragmented into as many as 1024 fragments - far too many to support efficient reassembly procedures. Plus, the loss of a single fragment would result in retransmission of all fragments of the original packet. Fragmentation is also critical for tunnels as discussed in [I-D.ietf-intarea-tunnels] and [RFC4459] where a more robust IP fragmentation/reassembly service is needed.

These issues suggest that the original fragmentation and reassembly design common to both IP protocol versions has shortcomings that can be addressed through a well-engineered alternative solution. However, the earlier publications raised alarms that resonated both broadly and deeply throughout the Internetworking industry. This has resulted in a state of paralysis where little progress to address the issues has been made even in the modern era.

5. IPv6 Fragmentation Solution

As discussed above, standard IP fragmentation has a number of issues that have been widely known but left unaddressed for many decades. This condition can be corrected by applying the classic IETF process of problem statement leading to solution. This document in conjunction with [RFC8900] should therefore be considered as a problem statement, with an IPv6 solution found in [I-D.templin-6man-ipid-ext2].

The offered solution addresses IPv6 fragmentation issues by establishing an Extended Fragment Header (EFH) intended for use instead of the standard IPv6 fragmentation header and procedures. The EFH is based on a 64-bit Identification value, mandates no more

than 64 fragments per packet, mandates a minimum non-final fragment size of 1024 octets, mandates an identical size for all non-final fragments (while the final fragment must not be larger) and eliminates any possibility for fragment overlap. Additionally, intermediate systems are unable to alter the size of individual fragments in isolation without first (virtually) reassembling the entire packet. The fragment sizes produced by the source will therefore match the fragment sizes that arrive at the destination unless an intermediate system performs the onerous task of (virtual) reassembly and re-fragmentation.

The EFH solution also has provisions for managing the loss unit with respect to the retransmission unit. In networks where loss is rare, this means that the source may send large packets at high data rates even if fragmentation with a non-final fragment size as small as 1024 octets is necessary. When loss becomes significant, the EFH solution provides a means for the destination to advise the source to reduce the size of its packets resulting in fewer fragments.

The EFH solution is based on an IPv6 Destination Option that appears instead of the IPv6 Fragment Header with the understanding that packets that contain IPv6 extension headers may be dropped along some paths. When (UDP)/IP encapsulation is applied, however, packets containing the EFH can often transit limited domains [RFC8799] without loss due to intermediate system filtering.

6. Conclusion

Standard IP fragmentation has well known issues that were presented in ways that caused the community to become paralyzed with uncertainty rather than move forward in confidence according to the time-proven IETF process of problem statement leads to solution. While the earlier publications succeeded in articulating the issues, their titles inspired fear, uncertainty and doubt instead of promoting a well-balanced engineering approach toward a robust solution. The time for such a solution has now arrived.

7. IANA Considerations

This document is an informational problem statement and does not in itself request any IANA actions. IANA considerations can be found in the cited solution space document.

8. Security Considerations

This document is an informational problem statement and does not in itself address security. Security considerations can be found in the cited solution space document.

9. Acknowledgements

Performance maximization efforts in the Internet engineering community have produced foundational improvements. Those who contributed are acknowledged.

The following are acknowledged for their helpful discussion and/or comments: Joel Halpern, Bob Hinden, Joe Touch.

Honoring life, liberty and the pursuit of happiness.

10. References

10.1. Normative References

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

[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>.

10.2. Informative References

[I-D.ietf-intarea-tunnels] Touch, J. D. and M. Townsley, "IP Tunnels in the Internet Architecture", Work in Progress, Internet-Draft, draft-ietf-intarea-tunnels-15, 9 May 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-intarea-tunnels-15>.

[I-D.templin-6man-ipid-ext2] Templin, F. and T. Herbert, "IPv6 Extended Fragment Header (EFH)", Work in Progress, Internet-Draft, draft-templin-6man-ipid-ext2-17, 3 June 2025, <https://datatracker.ietf.org/doc/html/draft-templin-6man-ipid-ext2-17>.

[I-D.templin-6man-parcels2] Templin, F., "IPv6 Parcels and Advanced Jumbos (AJs)", Work in Progress, Internet-Draft, draft-templin-6man-parcels2-27, 21 May 2025, <https://datatracker.ietf.org/doc/html/draft-templin-6man-parcels2-27>.

[KENT87] Kent, C. and J. Mogul, "Fragmentation Considered Harmful, SIGCOMM ’87: Proceedings of the ACM workshop on Frontiers in computer communications technology, DOI 10.1145/55482.55524, http://www.hpl.hp.com/techreports/Compaq-DEC/WRL-87-3.pdf.", August 1987.

[MOGUL19] Mogul, J. and C. Kantarjiev, "Retrospective on "Fragmentation Considered Harmful", ACM SIGCOMM Computer Communication Review, https://ccronline.sigcomm.org/wp-content/uploads/2019/10/acmdl19-328.pdf", October 2019.

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, DOI 10.17487/RFC2003, October 1996, <https://www.rfc-editor.org/info/rfc2003>.

[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>.

[RFC4459] Savola, P., "MTU and Fragmentation Issues with In-the-Network Tunneling", RFC 4459, DOI 10.17487/RFC4459, April 2006, <https://www.rfc-editor.org/info/rfc4459>.

[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>.

[RFC4963] Heffner, J., Mathis, M., and B. Chandler, "IPv4 Reassembly Errors at High Data Rates", RFC 4963, DOI 10.17487/RFC4963, July 2007, <https://www.rfc-editor.org/info/rfc4963>.

[RFC6864] Touch, J., "Updated Specification of the IPv4 ID Field", RFC 6864, DOI 10.17487/RFC6864, February 2013, <https://www.rfc-editor.org/info/rfc6864>.

[RFC7739] Gont, F., "Security Implications of Predictable Fragment Identification Values", RFC 7739, DOI 10.17487/RFC7739, February 2016, <https://www.rfc-editor.org/info/rfc7739>.

[RFC8799] Carpenter, B. and B. Liu, "Limited Domains and Internet Protocols", RFC 8799, DOI 10.17487/RFC8799, July 2020, <https://www.rfc-editor.org/info/rfc8799>.

[RFC8899] Fairhurst, G., Jones, T., T端xen, M., R端ngeler, I., and T. V旦lker, "Packetization Layer Path MTU Discovery for Datagram Transports", RFC 8899, DOI 10.17487/RFC8899, September 2020, <https://www.rfc-editor.org/info/rfc8899>.

[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>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

[RFC9171] Burleigh, S., Fall, K., and E. Birrane, III, "Bundle Protocol Version 7", RFC 9171, DOI 10.17487/RFC9171, January 2022, <https://www.rfc-editor.org/info/rfc9171>.

[RFC9293] Eddy, W., Ed., "Transmission Control Protocol (TCP)", STD 7, RFC 9293, DOI 10.17487/RFC9293, August 2022, <https://www.rfc-editor.org/info/rfc9293>.

Appendix A. Additional Considerations

The Encapsulation Source operates on already-packaged IP packets supplied by the Original Source and can therefore only adapt their sizes by applying IPv6 fragmentation following encapsulation. The Encapsulation Destination in turn applies IPv6 reassembly prior to decapsulation. Conversely, the Original Source and Final Destination apply packaging and reassembly at multiple architectural layers and can selectively apply or avoid fragmentation of the original IP packets. This section presents additional considerations for upper layer packaging.

Fragmentation and reassembly using the Extended Fragment Header (EFH) operate on IP packets that include one or more upper layer protocol (ULP) segment with the corresponding ULP headers. This means that even for very large ULP segments only a single instance of ULP headers appears in the resulting sequence of interdependent fragments even if the segment size exceeds the path MTU.

Generic Segment Offload (GSO) with its counterpart Generic Receive Offload (GRO) are widely-known services that perform fragmentation and reassembly according to the same algorithm specified for the EFH but with ULP segment sizes no larger than the path MTU. GSO produces fragment sequences (independent packets, actually) that include a separate instance of the ULP headers in each packet instead of a single instance for the entire sequence. With a nominal ULP header size of 20 octets for TCP, this means that a 64-packet sequence would need to carry 1260 redundant octets for each GSO/GRO transaction - a significant increase in overhead. When transport layer security encapsulations such as TLS/SSL are present, the ULP header overhead is greater still.

ULP use of EFH fragmentation and reassembly in contrast with GSO/GRO therefore requires an adaptive consideration of the packet loss profile for a given flow. Assuming a nominal path MTU (e.g., 1280 octets, 1500 octets, etc.) and with minimal packet loss, EFH with larger ULP segment sizes offers efficiency advantages in comparison with GSO/GRO with MTU-sized segment sizes. When packet loss levels increase, however, ULPs that use EFH should adaptively reduce their segment sizes to compensate. When loss levels become significant, ULPs that use both EFH and GSO/GRO may need to reduce their transmission rates until loss profiles improve. These adaptations are necessary to dynamically balance the flow’s loss unit in relation to the retransmission unit under the current loss profile.

For larger path MTUs (e.g., 4500 octets, 9000 octets, or larger still), the two services converge to offer similar performance profiles at segment sizes no larger than the path MTU, while EFH can advance to still larger segment sizes for improved efficiency. EFH can also transport IP parcels and Advanced Jumbos (following IPv6 encapsulation) even if the underlying path does not support them natively.

Appendix B. Change Log

<< RFC Editor - remove prior to publication >>

Differences from -02 to -03:

* Cited RFC4459 and draft-ietf-intarea-tunnels.

Differences from -00 to -02:

* Clarified that only IPv6-based solutions are in scope.

* Distinguished between source fragmentation and network fragmentation.

* De-emphasized (but did not deprecate) the use case of multiple layers of fragmentation/reassembly on the same physical platform.

* Cited [MOGUL19].

Differences from earlier versions:

* First draft publication.

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