Network Working Group
Internet-Draft
Intended status: Experimental
Expires: 2 January 2026

F. L. Templin, Ed.
Boeing Research & Technology
T. Herbert
Unaffiliated

1 July 2025

# IPv6 Extended Fragment Header (EFH)
## draft-templin-6man-ipid-ext2-25

Abstract

The Internet Protocol, version 4 (IPv4) header includes a 16-bit Identification field in all packets, but this length is too small to ensure reassembly integrity even at moderate data rates in modern networks. Even for Internet Protocol, version 6 (IPv6), the 32-bit Identification field included when a Fragment Header is present may be smaller than desired for some applications. Both IPv4 and IPv6 fragmentation have further been classified as fragile to the point that their use is discouraged. This specification addresses these limitations by defining an IPv6 Extended Fragment Header (EFH) that includes a 64-bit Identification in the context of more robust, secure and efficient fragmentation and reassembly procedures.

Table of Contents



## 1. Introduction

The Internet Protocol, version 4 (IPv4) header includes a 16-bit Identification in all packets [RFC0791], but this length is too small to ensure reassembly integrity even at moderate data rates in modern networks [RFC4963][RFC6864][RFC8900]. For Internet Protocol, version 6 (IPv6), the Identification field is only present in packets that include a Fragment Header [RFC8200], but even its standard length of 32 bits may be too small for some applications. Standard IP fragmentation is also subject to numerous performance and security issues that indicate a need for a more robust service. This specification therefore defines a new fragmentation service that

addresses these issues through the introduction of an IPv6 Extended Fragment Header (EFH).

The EFH employs a fragmentation/reassembly algorithm based on an ordinal fragment index in combination with the non-final fragment payload length instead of a 13-bit integer encoding an 8-octet offset. In this arrangement, both fragmentation and reassembly are greatly simplified allowing for efficient implementations. These improvements are based on an ample minimum fragment payload length made possible by the 1280 octet IPv6 minimum MTU.

The EFH is needed for networks that engage fragmentation and reassembly at extreme data rates, or for cases when advanced packet Identification uniqueness assurance is critical. (Placement of the EFH in a Destination Options header should also make the option less prone to network filtering.) This specification further defines a messaging service for adaptive realtime response to loss and congestion related to fragmentation/reassembly. Together, these extensions support robust fragmentation and reassembly services as well as packet Identification uniqueness for IPv6.

The EFH 64-bit Identification concept is similar to the Extended Sequence Number (ESN) framework found in IPsec AH [RFC4302] and ESP [RFC4303]. In both cases, nodes can apply header compression to transmit only the least significant bits while retaining the most significant bits in cache memory.

## 2. Terminology

The terms "Maximum Transmission Unit (MTU)", "Effective MTU to Receive (EMTU_R)", "Effective MTU to Send (EMTU_S)" and "Maximum Segment Lifetime (MSL)" from standard Internetworking terminology apply [RFC1122]. The term "Maximum Receive Unit (MRU)" is equivalent to EMTU_R, and the term "maximum datagram lifetime (MDL)" (see: [RFC0791][RFC6864]) is equivalent to MSL.

The term "Packet Too Big (PTB)" refers to an ICMPv6 "Packet Too Big" message [RFC8201][RFC4443] or a new ICMPv4 "PTB" message type defined in this document (see: IANA Considerations).

The term "flow" refers to a sequence of packets sent from a particular source to a particular unicast, anycast or multicast destination that a node desires to label as a flow [RFC6437].

The term "Extended Fragment Header (EFH)" refers to a new IPv6 Destination Option defined in this document. The EFH is included in a Destination Options header as the final Per-Fragment header, while the remainder of the packet that follows the Per-Fragment headers is known as the "fragment payload".

The term "Fragmentation Report (FRAGREP)" refers to an alternate IPv6 option type encoding of the EFH option used to report reassembly conditions. Destinations may include FRAGREPs in return packets to EFH fragment sources.

The Automatic Extended Route Optimization (AERO) [I-D.templin-6man-aero3] and Overlay Multilink Network Interface (OMNI) [I-D.templin-6man-omni3] services employ the EFH for secure adaptation layer encapsulation and fragmentation. New packet types termed "IPv6 Parcels and Advanced Jumbos (AJs)" are specified in [I-D.templin-6man-parcels2].

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. Motivation

Upper layer protocols can achieve greater performance by configuring segment sizes that exceed the path Maximum Transmission Unit (MTU). When the segment size exceeds the path MTU, lower layer IP fragmentation is a natural consequence. However, the 4-octet (32-bit) Identification field of the IPv6 Fragment Header may be too small to ensure reassembly integrity at sufficiently high data rates, especially when the source resets the starting sequence number often to maintain an unpredictable profile [RFC7739]. This specification therefore proposes to fortify the IPv6 Identification by extending its length.

Performance increases for upper layer protocols that use larger segment sizes was historically observed for NFS over UDP, and can still be readily observed today for TCP and the Delay Tolerant Network (DTN) Licklider Transmission Protocol (LTP) - see: [I-D.templin-dtn-ltpfrag]. The performance testbed included a pair of modern high-performance workstations with 100Gbps Ethernet cards connected via a point-to-point link and running a modern public domain linux release. TCP performance using the public domain 'iperf3' tool was proven to increase when larger user buffer sizes are used even if they exceed the path MTU and invoke a service known as Generic Segment/Receive Offload (GSO/GRO). LTP performance

improvement with segment sizes that exceed the path MTU was similarly proven using the HDTN and ION-DTN LTP implementations when IP fragmentation and reassembly were invoked.

In addition to accommodating higher data rates in the presence of fragmentation and reassembly, extending the IPv6 Identification can enable other important services. For example, an extended Identification can support a duplicate packet detection service where the network remembers recent Identification values to aid detection of potential duplicates. When an encapsulation source includes an EFH, the extended Identification can also serve as a sequence number that allows each encapsulation destination to exclude any packets with values outside of the current sequence number window as potential spurious transmissions from an off-path attacker.

The standard IPv6 Fragment Header also carried forward the awkward fragmentation parameters found in IPv4 including a 13-bit quadword Fragment Offset value, no restrictions on fragment-by-fragment payload lengths and no limits on numbers of fragments produced. In contrast, the EFH service mandates same-sized fragments, forbids tiny non-final fragments, places a conservative limit on the maximum number of fragments and eliminates any possibilities for fragment overlap. These factors ensure a more secure and performance-optimized fragmentation and reassembly service.

An optimized IP fragmentation and reassembly service using an extended Identification can provide a useful tool for performance maximization and path MTU robustness in the Internet. This document therefore presents a means to extend the IPv6 Identification in a more efficient fragmentation and reassembly specification to better support these services through the introduction of an Extended Fragment Header (EFH).

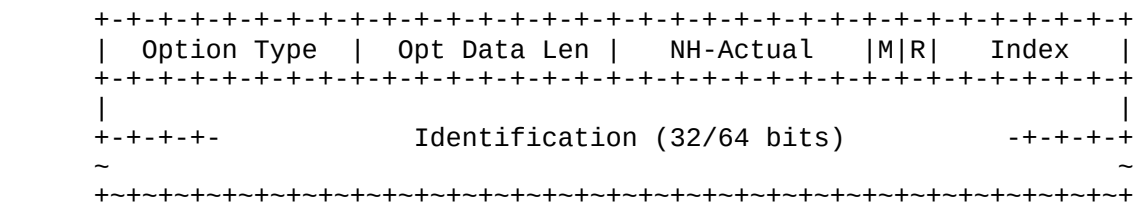
## 4. Extended Fragment Header (EFH)

For a conventional 4-octet IPv6 Identification, the source can simply include a standard IPv6 Fragment Header as specified in [RFC8200]. The source then includes a 4-octet Identification value for the packet and applies fragmentation.

For an extended Identification, advanced fragmentation and reassembly procedures and/or for paths that drop packets including the standard IPv6 Fragment Header, this specification permits the source to instead include an EFH. The source includes the EFH in a Destination Options header positioned as the final IPv6 Per-Fragment Header. The remainder of the packet beyond the Destination Option header beginning with any Extension and Upper Layer Headers for the first fragment (or protocol data for non-first fragments) is known as the fragment payload.

The Destination Options header that includes the EFH option therefore appears in each fragment in the same position where the standard Fragment Header would normally appear while the Fragment Header itself is omitted - see Sections 4.1 and 4.5 of [RFC8200].

The EFH Destination Option is formatted as shown in Figure 1:




Option Type 8-bit value ’100[TBD1]’.

Opt Data Len 8-bit value 6 for a 32-bit Identification or 10 for a 64-bit Identification.

NH-Actual the actual value of the original Destination Options Next Header prior to fragmentation.

M/R/Index "(M)ore Fragments" and "(R)eserved" flags followed by a 6-bit "Index" that identifies the ordinal index for each fragment payload.

Identification an 8-octet (64 bit) or 4-octet (32 bit) unsigned integer Identification, in network byte order.

Figure 1: EFH Destination Option

The EFH Destination Option is therefore identified as an IPv6 Option Type with the low-order 5 bits set to TBD1 (see: IANA Considerations) and with the third-highest-order bit (i.e., "chg") set to 0. The highest-order 2 bits (i.e., "act") are set to ’10’ so that destinations that do not recognize the option will drop the packet or

fragment and (possibly) also return an ICMPv6 Parameter Problem message. When Opt Data Len is 10, the Identification field is 8 octets (64 bits) in length and the option is aligned such that the field begins on a natural 8-octet boundary as "8n+4" (see: [RFC8200]).

In a compressed form, the source sets Opt Data Len to the value 6 (i.e., instead of 10) and includes only the 4 least significant octets of the (8-octet) Identification field and the option is aligned such that the field begins on a natural 4-octet boundary as "4n" (see: [RFC8200]). The source can engage this compressed form after it has already published the 4 most significant octets to establish state in any intermediate systems and the destination end system. Intermediate and end systems that have already cached the 4 most significant octets regard the Identification as a full 8 octet value for the purpose of packet filtering and reassembly; otherwise, they regard the 4 most significant octets as 0.

For improved efficiency, sources often send packets that include full IPv6 headers (including the EFH extension) only as initial packets of a flow while including greatly compressed headers in subsequent packets. When a flow becomes stale, the source can send additional full header packets to refresh flow state until header compression can resume. The AERO/OMNI service is an example where the EFH is subject to efficient header compression.

## 5. Source Fragmentation

IPv6 fragmentation using the EFH is conducted in a manner similar to standard IPv6 fragmentation (see: Section 4.5 of [RFC8200]) with the following exceptions.

When the source performs fragmentation using the EFH, it creates fragments of the same packet based on the (Source, Destination, Identification)-tuple. The source SHOULD produce the smallest number of fragments possible within current path MTU constraints and MUST produce no more than 64 fragments per packet. The fragment payload of each non-final fragment following the Destination Options header MUST NOT be smaller than 1024 octets, allowing for up to 256 octets of Per-Fragment headers plus any lower-layer encapsulations within the 1280 octet IPv6 minimum path MTU. Each non-final fragment payload MUST be equal in length, while the final fragment payload MAY be smaller and MUST NOT be larger (a zero-length final fragment payload is therefore also permissible).

For each of the F fragments produced during fragmentation, the source writes an ordinal index number beginning with 0 in the "Index" field for the first fragment and increasing by 1 for each successive non-

first fragment while setting the "M" flag accordingly. Specifically, the source sets (Index, M) to (0, 1) for the first fragment, (1, 1) for the second, (2, 1) for the third, etc., up to and including ((F-1), 0) for the final fragment.

For each fragment produced during fragmentation, the source inserts a Destination Options header including the EFH option as the final Per-Fragment header. The source then caches the Destination Options header Next Header value in the NH-Actual field and (for each non-first fragment) resets the Next Header field to "No Next Header". Network middleboxes that forward non-first fragments prepared in this way should therefore ignore the fragment payload that follows by virtue of the "No Next Header" setting.

## 6. Network Fragmentation

Unlike IPv4, fragmentation in IPv6 is performed only by source nodes, not by routers along a packet’s delivery path [RFC8200]. However, the standard IPv6 Fragment Header has no protocol provisions to prevent network middleboxes (i.e., any manner of device on the path that forwards an IPv6 packet) from performing further fragmentation on individual packets processed in isolation and the destination would still reassemble correctly the same as for fragments produced by the source albeit with potentially degraded performance.

For fragment packets that include an EFH, any further fragmentation by a network middlebox would cause malformed fragments to arrive at the destination and therefore spoil any in-progress reassemblies. (Any such network middleboxes that deliberately disrupt service by violating the standards have no place in operational networks.) A network middlebox could instead perform the onerous task of (virtual) reassembly of all fragments of the same packet before re-fragmentating to a different fragment size, but this might not scale well in all environments. A network middlebox could also perform gratuitous fragmentation on an EFH packet with Index and M both set to 0, but this would serve no functional purpose while possibly impacting performance.

IPv6 routers forward EFH packets no larger than the next hop link MTU without modification while decrementing the Hop Limit; they instead drop EFH packets that are too large and return ICMPv6 PTB messages. IPv6 routers, intermediate systems and all manners of network middleboxes must not perform (further) fragmentation on EFH packets nor intentionally alter the EFH option contents in any way (see: Section 12).

## 7. Destination Reassembly

Destination reassembly using the EFH is conducted in a similar manner as for standard IPv6 reassembly (see: Section 4.5 of [RFC8200]) with the following exceptions.

When the destination receives EFH fragments with the same (Source, Destination, Identification)-tuple, it reassembles by concatenating the payloads of consecutive fragments in ascending ordinal Index numbers, i.e., ordinal 0, followed by 1, followed by 2, etc. until all fragments are concatenated. In the process, the destination discards any non-final fragments with fragment payload lengths less than 1024 octets or with fragment payload lengths that differ from the others.

When the destination receives an EFH fragment with Index 0 that contains the original packet in its entirety, it regards reassembly complete and delivers the original packet (minus the EFH itself) to upper layers.

## 8. Destination Qualification and Path MTU

IPv6 destinations that do not recognize the EFH option drop the packet and may also return a Code 2 ICMPv6 Parameter Problem message [RFC4443]. (ICMPv6 messages may be lost on the return path and/or manufactured by an adversary, however, and therefore provide only an advisory indication.)

The IPv6 source can then test whether destinations recognize the EFH option by occasionally sending "probe" packets/fragments that include the option. The source has assurance that a destination recognizes the option if it receives acknowledgments; otherwise, it may receive Code 2 ICMPv6 Parameter Problem messages as hints that a destination does not recognize the option. The source should re-probe the path occasionally in case routing redirects a flow to a different anycast destination or in case a multicast group membership changes (see: Section 11).

## 9. Packet Size Issues

When the source sends fragment payloads larger than the minimum size of 1024 octets using the EFH option, it should probe the path MTU for each flow occasionally by sending probe packets as EFH first fragments (i.e. ones with Index set to 0) per [RFC8899].

When the destination receives a probe packet for a particular flow, it returns a responsive packet that includes a Fragmentation Report (FRAGREP) Destination Option per Section 10. The responsive packet should be any authentic probe reply with the Source and Destination addresses reversed.

If the source receives a probe reply, it can then perform source fragmentation using the EFH option with the fragment payload length advanced to the size of the probe.

If the destination experiences reassembly congestion, it can begin returning authentic packets with FRAGREP options to the source with MRU set to a reduced size (see: Section 10). When the source receives the packets, it should temporarily reduce the size of its future transmissions for the flow but may resume using larger sizes if the FRAGREPs subside.

If the source is an encapsulation ingress, it also returns a translated PTB message with a corresponding soft error Code to the original source per [RFC2473]. If the source regards the packet as lost, it sets the Code to "Soft Error (loss); otherwise, it sets the Code to "Soft Error (no loss)". For IPv4, the source uses a new ICMPv4 PTB message Type TBD2 and with a corresponding soft error Code (see: IANA Considerations).

## 10. Fragmentation Reports

End systems that recognize the EFH also recognize an IPv6 Fragmentation Report (FRAGREP) option that uses type TBD1 the same as for the EFH but with the "act/chg" bits set to ’000’ and formatted as shown below:

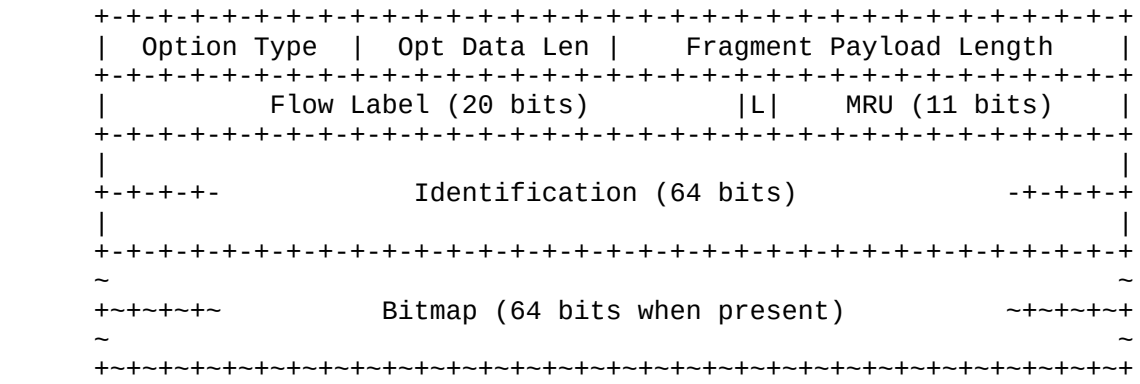



Figure 2: Fragmentation Report Option

The destination end system may include FRAGREP options in return packets to the source to report receipt of an explicit probe (see: Section 9), reassembly congestion and/or fragment loss.

The destination can use any return packet (i.e., one with the Source and Destination Addresses from the packet that triggered the FRAGREP reversed) to carry the option, especially if it includes identifying parameters and/or authentication signatures. The option is aligned such that the Identification field begins on a natural 8-octet boundary as "8n" (see: [RFC8200]).

The destination sets Flow Label and Fragment Payload Length to the values corresponding to the invoking fragment, sets (Res)erved to 0 and sets MRU to the most significant 11 bits of the recommended 16-bit maximum reassembly size under current congestion conditions. When the 11 transmitted MRU bits are all-1’s, the recipient regards the 5 untransmitted bits as all-1’s; otherwise, it regards them as all-0’s.

If no fragments of the subject packet have (yet) been lost, the destination next sets Opt Data Len to 14, sets (L)oss to 0 and omits the Bitmap field. If the destination has abandoned reassembly for the packet due to fragment loss, it instead sets L to 1.

The destination then includes the full 8 octet (64-bit) Identification even if the invoking packet includes only the 4 least significant octets. If some fragments are missing, the destination optionally sets Opt Data Len to 22 (i.e., instead of 14) and includes a 64-bit ordinal fragment index Bitmap field. The destination sets each Bitmap(i) bit (for i=0 to 63) to 1 for each ordinal fragment index received or 0 for each index not received for this reassembly.

When the source receives authentic packets that include the FRAGREP option it should reduce, maintain or increase the size of its continued packet transmissions for this flow according to the advertised Flow Label and MRU. If the reported Fragment Payload Length is larger than the current path MTU estimate, it can also advance the fragment size for this flow under the guidance found in [RFC8899]. This ensures that both packet and fragment sizes are adaptive for a given flow.

If the FRAGREP option further includes a Bitmap, the source can retransmit any missing ordinal fragments if it still has them in its cache provided the delay would not interfere with upper layer protocol retransmissions [RFC3819].

Note: the FRAGREP option may appear in the same Destination Options header that includes an EFH option without exceeding recommended limits (see: [I-D.ietf-6man-eh-limits]). The options should be ordered as EFH followed by FRAGREP to enable natural multi-octet word alignment with minimal padding.

## 11. Multicast and Anycast

In addition to unicast flows, similar considerations apply for flows in which the Destination Address is multicast or anycast. Unless the source and all candidate destinations are members of a limited domain network [RFC8799] for which all nodes recognize the EFH, some destinations may recognize the option while others drop packets containing the option and may return a Code 2 ICMPv6 Parameter Problem message [RFC4443].

When a source sends packets/fragments with IPv6 EFH options to a multicast group, the packets/fragments may be replicated in the network such that a single transmission may reach N destinations over as many as N different paths. Some destinations may then return packets with FRAGREP options if they experience congestion and/or loss, while other destinations may return Code 2 ICMPv6 Parameter Problem messages if they do not recognize the EFH option.

While the source receives authentic PTB messages or authentic packets with FRAGREP options, it should reduce the sizes of the packets/fragments it sends to the flow multicast group even if only one or a few paths or destinations are currently experiencing congestion. This means that transmissions to a multicast group for a given flow will converge to the performance characteristics of the lowest common denominator group member destinations and/or paths. While the source receives ICMPv6 Parameter Problem messages and/or otherwise detects that some multicast group members do not recognize the EFH option, it must determine whether the benefits for group members that recognize the option outweigh the drawbacks of service denial for those that do not.

When a source sends packets/fragments with EFH options to an anycast address, routing may direct initial fragments to a first destination while directing the remaining fragments to other destinations that configure the same address. These wayward fragments will simply result in incomplete reassemblies at each such anycast destination which will soon purge the fragments from the reassembly buffer. The source will eventually retransmit, and all resulting fragments should eventually reach a single reassembly target.

## 12. Requirements

Normative aspects of standard IPv6 fragmentation and reassembly [RFC8200] apply also to the EFH except where this document specifies differences.

Sources and Destinations MUST apply EFH fragmentation and reassembly according to the 3-tuple (Source, Destination, Identification). This means that all flows share the same Identification number space for the purpose of fragmentation and reassembly, but path MTU feedback is on a per-flow basis.

Destinations that accept flows using EFH options MUST configure an EMTU_R of 65535 octets or larger. Destinations MAY advertise "soft" temporary EMTU_R reductions in FRAGREP messages during periods of loss/congestion, but MUST continue to honor the "hard" upper limit. The source SHOULD therefore respond to FRAGREP messages from the destination by sending EFH fragments at rates that will minimize reassembly congestion.

Sources MUST NOT include more than one IPv6 Fragment Header or EFH option in each IPv6 packet/fragment, and destinations MUST silently drop packets/fragments with multiples. If the source includes an EFH option, it MUST appear in a Destination Options header that appears as the final Per-Fragment header before the fragment payload.

Sources that include an EFH option MUST perform fragmentation such that at most 64 fragments are produced and all non-final fragments include equal-length fragment payloads no smaller than 1024 octets. The final fragment MAY be smaller (or even zero-length) and MUST NOT be larger.

Sources that include the EFH option in packet transmissions MUST also recognize the FRAGREP option in return packets as specified in Section 10.

IPv6 routers, intermediate systems and network middleboxes MUST honor the EFH/FRAGREP Option Type "chg" bit and therefore MUST NOT perform (further) EFH fragmentation nor intentionally alter the EFH/FRAGREP Option Data in any way. Any such alterations may cause an upper layer authentication check and/or reassembly to fail resulting in denial of a legitimate service.

## 13. Implementation Status

In progress.

## 14. IANA Considerations

### 14.1. IPv6 Parameters

The IANA is requested to assign a new IPv6 Destination Option type in the "Destination Options and Hop-by-Hop Options" table of the https://www.iana.org/assignments/ipv6-parameters/ registry group (registration procedures IESG Approval, IETF Review or Standards Action). The option type should appear in 2 consecutive table entries.

The first entry sets "act" to ’00’, "chg" to ’0’, "rest" to TBD1, "Description" to "IPv6 Fragmentation Report" and "Reference" to this document [RFCXXXX].

The second entry sets "act" to ’10’, "chg" to ’0’, "rest" to TBD1, "Description" to "IPv6 Extended Fragment Header" and "Reference" to this document [RFCXXXX].

Both table entries finally set "Hex Value" to the 2-digit hexadecimal value corresponding to the 8-bit concatenation of "act/chg/rest".

### 14.2. ICMPv6 Parameters

The IANA is instructed to assign new Code values in the "ICMPv6 Code Fields: Type 2 - Packet Too Big" registry in the https://www.iana.org/assignments/icmpv6-parameters registry group (registration procedure is Standards Action or IESG Approval). The registry entries should appear as follows:

| Code | Name | Reference |
|---|---|---|
| 0 | PTB Hard Error | [RFC4443] |
| 1 (suggested) | PTB Soft Error (loss) | [RFCXXXX] |
| 2 (suggested) | PTB Soft Error (no loss) | [RFCXXXX] |

Figure 3: ICMPv6 Code Fields: Type 2 - Packet Too Big Values

### 14.3. ICMP Parameters

The IANA is instructed to assign a new Type number TBD2 in the "ICMP Type Numbers" registry in the https://www.iana.org/assignments/icmp-parameters registry group (registration procedures IESG Approval or Standards Action). The entry should set "Type" to TBD2, "Name" to "Packet Too Big (PTB)" and "Reference" to [RFCXXXX] (i.e., this document).

The IANA is further instructed to create a new table titled: "Type TBD2 - Packet Too Big (PTB)" in the "Code Fields" registry, with registration procedures IESG Approval or Standards Action. The table should have the following initial format:

| Code | Name | Reference |
|---|---|---|
| 0 | Reserved | [RFCXXXX] |
| 1 (suggested) | PTB Soft Error (loss) | [RFCXXXX] |
| 2 (suggested) | PTB Soft Error (no loss) | [RFCXXXX] |

Figure 4: Type TBD2 - Packet Too Big (PTB)

## 15. Security Considerations

All aspects of IP security apply equally to this document, which does not introduce any new vulnerabilities. Moreover, when employed correctly the mechanisms in this document robustly address known IP reassembly integrity concerns [RFC4963] and also provide an advanced degree of packet Identification uniqueness assurance.

All security aspects of [RFC7739], including the algorithms for selecting fragment identification values, apply also to the IPv6 EFH. In particular, the source should reset its starting Identification value frequently to maintain an unpredictable profile.

All normative security guidance on IPv6 fragmentation found in [RFC8200] (e.g., processing of tiny first fragments, overlapping fragments, etc.) applies also to the fragments generated under the EFH.

A performance-degrading denial of service vector is possible for EFH packets with both Index and M set to 0 when network middleboxes ignore the normative requirements and maliciously engage fragmentation. In contrast to standard IPv6 "atomic" fragments [RFC8021], however, a source is permitted to send EFH fragments with Index and M both set to 0. If the source suspects that network middlebox fragmentation may be impacting performance, it can resume sending multi-fragment EFH packets.

IPsec AH [RFC4302] and ESP [RFC4303] define an Extended Sequence Number (ESN) that is analogous to the 64-bit Identification specified for the IPv6 EFH option. Nodes that employ the EFH can use the Identification value as a sequence number to improve security in the same fashion as for IPsec AH/ESP ESNs.

## 16. Acknowledgements

This work was inspired by continued DTN performance studies. Amanda Baber, Bob Hinden, Christian Huitema, Mark Smith and Eric Vyncke offered useful insights that helped improve the document.

Honoring life, liberty and the pursuit of happiness.

## 17. References

### 17.1. Normative References

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

[RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, DOI 10.17487/RFC1122, October 1989, <https://www.rfc-editor.org/info/rfc1122>.

[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>.

### 17.2. Informative References

[I-D.ietf-6man-eh-limits]
Herbert, T., "Limits on Sending and Processing IPv6 Extension Headers", Work in Progress, Internet-Draft, draft-ietf-6man-eh-limits-19, 27 February 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-6man-eh-limits-19>.

[I-D.templin-6man-aero3]
Templin, F., "Automatic Extended Route Optimization (AERO)", Work in Progress, Internet-Draft, draft-templin-6man-aero3-45, 13 June 2025, <https://datatracker.ietf.org/doc/html/draft-templin-6man-aero3-45>.

[I-D.templin-6man-omni3]
Templin, F., "Transmission of IP Packets over Overlay Multilink Network (OMNI) Interfaces", Work in Progress, Internet-Draft, draft-templin-6man-omni3-61, 13 June 2025, <https://datatracker.ietf.org/doc/html/draft-templin-6man-omni3-61>.

[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>.

[I-D.templin-dtn-ltpfrag]
Templin, F., "LTP Performance Maximization", Work in Progress, Internet-Draft, draft-templin-dtn-ltpfrag-17, 23 May 2024, <https://datatracker.ietf.org/doc/html/draft-templin-dtn-ltpfrag-17>.

[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.

[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>.

[RFC3819] Karn, P., Ed., Bormann, C., Fairhurst, G., Grossman, D., Ludwig, R., Mahdavi, J., Montenegro, G., Touch, J., and L. Wood, "Advice for Internet Subnetwork Designers", BCP 89, RFC 3819, DOI 10.17487/RFC3819, July 2004, <https://www.rfc-editor.org/info/rfc3819>.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <https://www.rfc-editor.org/info/rfc4302>.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <https://www.rfc-editor.org/info/rfc4303>.

[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>.

[RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, DOI 10.17487/RFC6437, November 2011, <https://www.rfc-editor.org/info/rfc6437>.

[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>.

[RFC8021] Gont, F., Liu, W., and T. Anderson, "Generation of IPv6 Atomic Fragments Considered Harmful", RFC 8021, DOI 10.17487/RFC8021, January 2017, <https://www.rfc-editor.org/info/rfc8021>.

[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>.

[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. Change Log

<< RFC Editor - remove prior to publication >>

Differences from draft version -24 to -25:

* Removed the (P)robe bit, since application layer data is already used for probing. An explicit (P)robe bit could also provide means for a network middlebox to give different treatment to probes and ordinary traffic.

Differences from draft version -23 to -24:

* Further clarifications on intermediate system requirements.

Differences from draft version -22 to -23:

* Concluding resolutions for EFH atomic fragments.

Differences from draft version -21 to -22:

* Clarifications on fragmentation robustness and atomic fragments.

Differences from draft version -20 to -21:

* Rate-limited explicit FRAGREPs to ensure that atomic packets are transiting the forward path.

* Include Fragment Payload Length in FRAGREP to make fragment size probing stateless.

Differences from earlier versions:

* See draft version -20.

Authors' Addresses

Fred L. Templin (editor)
Boeing Research & Technology
P.O. Box 3707
Seattle, WA 98124
United States of America
Email: fltemplin@acm.org

Tom Herbert
Unaffiliated
San Jose, CA
United States of America
Email: tom@herbertland.com