Network Working Group
Internet-Draft
Intended status: Informational
Expires: 31 August 2025

T. Herbert
27 February 2025

Limits on Sending and Processing IPv6 Extension Headers
draft-ietf-6man-eh-limits-19

Abstract

This document defines various limits that may be applied to receiving, sending, and otherwise processing packets that contain IPv6 extension headers. Limits are pragmatic to facilitate interoperability amongst hosts and routers, thereby increasing the deployability of extension headers. The limits described herein establish the minimum baseline of support for use of extension headers on the Internet. If it is known that all communicating parties for a particular communication, including destination hosts and any routers in the path, are capable of supporting more than the baseline then these default limits may be freely exceeded.

Commenté [MB1]: Not sure what is meant here.

Commenté [MB2]: To clarify

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 31 August 2025.

Copyright Notice

Table of Contents

1. Introduction

Extension headers are a core component of the IPv6 protocol as specified in Section 4 of [RFC8200]. IPv6 extension headers were originally defined with few restrictions. For instance, there is no specified limit on the number of extension headers that a packet may have, nor is there a limit on the length in bytes of extension headers in a packet other than being limited by the Path MTU or 1,280 bytes for those hosts that do not discover the Path MTU [RFC7112]. Similarly, variable length extension headers typically do not have prescribed limits such as limits on the number of Hop-by-Hop or Destination options in a packet. The lack of limits essentially requires implementations to handle every conceivable usage of the protocol, including myriad use cases outside the realm of ever being realistic or useful in real world deployment. A packet that includes an excessive number or size of Hop-by-Hop ~~options~~ Options in a packet has also been raised as a security concern [RFC4942].

The lack of limits and the requirements for supporting a virtually open-ended protocol have led to a current lack of support and deployment of extension headers ([RFC7872], [Cus23b]). Instead of attempting to satisfy the protocol requirements concerning extension headers, some router and middlebox vendors have opted to invent and apply their own ad hoc limits, relegate packets with extension headers to slow path processing, or have gone so far as to summarily discard all packets with extension headers [RFC9098]. For those hosts and routers that properly attempt to process all extension headers per the specifications, the lack of limits has made them susceptible to ~~Denial~~ Denial-~~of~~ of-Service attacks. The net effect of this situation is that deployment and use of extension headers ~~is~~are

currently underwhelming.

This document describes various limits that hosts and routers may apply to the processing of extension headers. The goal of establishing limits is to narrow the requirements to better match reasonable use cases, thereby facilitating practical implementation and increased ~~deployability~~ deployment of extension headers.

1.1. Related work

Some of the problems of unlimited extension headers have been described or addressed in certain aspects.

[RFC8200] relaxed the requirement that all nodes in the path must process all Hop-by-Hop options in a packet ~~to be~~:

> NOTE: While [RFC2460] required that all nodes must examine and process the Hop-by-Hop Options header, it is now expected that nodes along a packet's delivery path only examine and process the Hop-by-Hop Options header if explicitly configured to do so.

Section 5.3 of [RFC8504] defines a number of limits that hosts may apply to processing extension headers. For instance, a limit on the maximum number of non-padding options allowed in a Destination Options header or Hop-by-Hop Options header is defined. This document expands on the requirements of [RFC8504] to allow limits to be set for routers and other intermediate nodes.

[RFC8883] defines a set of ICMP errors that may be sent if a limit concerning extension headers is exceeded and a node discards a packet as a result. [RFC8883] allows both hosts and routers to send such messages (effectively acknowledging that some routers discard packets with extension headers even though such behavior might be considered non-conformant with [RFC8200]).

Commenté [MB3]: May be cited as well; RFC9740 specified a mechanism to retrieve a limit that is typically set by hardware or software.

Section 14 of [RFC9000] (QUIC) gives an example of limits being set on extension headers per the requirements of the transport layer protocol. Note that the default limits in this document are greater than that of [RFC9000]. From [RFC9000]:

> Note: This requirement to support a UDP payload of 1200 bytes limits the space available for IPv6 extension headers to 32 bytes or IPv4 options to 52 bytes if the path only supports the IPv6 minimum MTU of 1280 bytes. This affects Initial packets and path validation.

[RFC7872] presents real-world data regarding the extent to which packets with IPv6 Extension Headers (EHs) are discarded in the Internet. [RFC9098] summarizes the operational implications of IPv6 extension headers, and attempts to analyze reasons why packets with IPv6 extension headers are often discarded in the public Internet.

Section 2.1.9 of [RFC4942] discusses security concerns with IPv6 extension headers. Excessive Hop-by-Hop ~~options~~ Options are one concern, and misuse of PAD1 and PADN options ~~are~~ is another. [RFC4942] also provides some foundation for the limits defined in this document.

This document sets the minimal upper bounds on the number of Hop-by-Hop Options that a node is expected to process. The lower bound is discussed in [RFC9673].

1.2. Requirements Language

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.

1.3. Terminology

This section provides definitions for some terms used in this document.

Node: a device that implements IPv6. It can be a host or a router.

Router: a node that forwards IPv6 packets not explicitly addressed to itself.

Intermediate node: a node that is addressed by an entry in a Routing Header list where the entry is not the last one in the List.

Host: any node that is not a router or intermediate node.

IPv6 header chain: the IPv6 header and the set of following IPv6 Extension Headers that precede the upper layer protocol in a Packet.

2. Overview of ~~extension~~ Extension ~~header~~ Header ~~limits~~Limits

The limits and requirements for handling extension headers defined in this document fall into the following categories:

* Limits on extension header length
* Limits on option length
* Limits on number of extension headers
* Limits on number of options
* Limits on padding for extension headers with options
* Limits on the length of the IPv6 header chain
* Limits on the ordering and types of extension headers

Limits are defined for both senders (sending hosts) and receivers (receiving hosts, intermediate nodes, or routers). A receiver limit is set to limit the amount of processing or the amount of data in received extension headers. Sender limits are set to limit the use of extension headers being sent. The purpose of sender limits is to increase the probability of successful delivery.

Commenté [MB4]: Add «non-padding »

Commenté [MB5]: See if we can barrow from RFC9740:

Extension header chain: Refers to the chain of extension headers
that are present in an IPv6 packet.

This term should not be confused with the IPv6 header chain, which
includes the IPv6 header, zero or more IPv6 extension headers, and
zero or a single Upper-Layer Header.

Commenté [MB6]: ipv6ExtensionHeadersChainLength

For receiver limits, a recommended action when a limit is exceeded is specified. The recommended action depends on the type of the node. For a host or intermediate node, the action when a limit is exceeded is generally to discard the packet. The rationale is that hosts are required to process all of the headers in a packet to process it correctly, and intermediate nodes are required to process all the extension headers up to and including the Routing Header in order to process a packet correctly. For a router, the action to take when a limit is exceeded is to stop processing the extension headers and to forward the packet.~~; s~~Specifically, if a router is processing Hop-by-Hop O~~o~~ptions and a limit is exceeded, then the router skips the option that caused the limit to be exceeded and skips any following Hop-by-Hop ~~options~~ Options per the procedures in Section 5.2 of [RFC9673]. The rationale is that the only extension header a router may process is Hop-by-Hop Options and the packet can be correctly forwarded if none or only some of the Hop-by-Hop options are processed.

This document includes limits on extension headers for their length and number of extension headers in a packet.~~, and a~~Also, it includes limits on Destination or Hop-by-Hop ~~options~~ Options for their length and number of options in a header. Limits on length are useful to nodes having hardware limitations, such as a ~~fixed~~ fixed-size parsing buffer in routers, which inherently limits the number of bytes in headers that a node can process; a node with such hardware limitations may choose to set length limits for extension headers and options accordingly. Limits on the number of extension headers or options are useful to nodes, such as end hosts, that have no inherent processing limitations.~~ f~~~~or~~For these nodes, limits on number of headers or options can be set to limit the cost of processing which is more a function of the number of items processed than the byte length of the items.

Each receiver limit described in this document has a recommended default value or minimum value when a limit is enforced. The intent of defining default limits is to establish an expected baseline of support. The default limits for senders correspond to the associated receiver default limits. The derivation for the default limits for number of options is discussed in Appendix A.1. The derivation of default length limits is discussed in Appendix A.2.

Padding options in Hop-by-Hop and Destination ~~options~~ Options have a particular purpose to align the next option or to pad the length of the extension header to a multiple of eight bytes. Similar to non-padding options, padding options require processing to parse over. Unlike non-padding options, padding options serve no other purpose than padding. To that end, limits on padding can be more restrictive than those on non-padding options. The justification for padding limits is discussed in Appendix A.3.

This document defines limits to optionally enforce extension header ordering and to optionally enforce that each extension header occurs at most once except for Destination options that may occur twice in a packet. The rationale for these limits is discussed in Appendix A.4.

Commenté [MB7]: Where?

Commenté [MB8]: Why discard for a receiving host?

3. Host L~~l~~imits for ~~sending~~ Sending ~~extension~~ Extension ~~h~~Headers

The requirements for limits related to a host sending packets with extension headers are:

* A source host SHOULD NOT send more than 8 non-padding options in a Destination Options header unless it has explicit knowledge that the destination host, and all intermediate nodes in a Routing Header in the case of a Destination Options header before the Routing Header, are able to process a greater number of options.

* A source host SHOULD NOT send a packet with a Destination Options header larger than 64 bytes unless it has explicit knowledge that the destination host, and all intermediate nodes in a Routing Header in the case of a Destination Options header before the Routing Header, are able to process a larger header size.

* A source host SHOULD NOT send a packet with a Destination option larger than 60 bytes unless it has explicit knowledge that the destination host, and all intermediate nodes in a Routing Header in the case of a Destination Options header before the Routing Header, are able to process options of a larger size.

* A source host SHOULD NOT send more than 8 non-padding options in a Hop-by-Hop Options header unless it has explicit knowledge that all possible routers, intermediate nodes, and the destination host in the path are able to process a greater number of options or will ignore options that exceed their limit in the case of routers.

* A source host SHOULD NOT send a packet with a Hop-by-Hop Options header larger than 64 bytes unless it has explicit knowledge that all possible routers, intermediate nodes, and the destination host in the path are able to process a larger header size.

* A source host SHOULD NOT send a packet with a Hop-by-Hop option larger than 60 bytes unless it has explicit knowledge that all possible routers, intermediate nodes, and the destination host in the path are able to process options of a larger size.

* A source host SHOULD NOT send a packet with an IPv6 header chain larger than 104 bytes unless it has explicit knowledge that all possible routers, intermediate nodes, and the destination host in the path are able to process a larger IPv6 header chain. If a packet contains an IPsec header, then this limit only applies to headers up to and including the IPsec header (the IPsec header obfuscates following headers so that they are unreadable by nodes in the path). This requirement is equivalently stated as a host SHOULD NOT send a packet with more than 64 bytes of aggregate extension headers.

* A source host SHOULD NOT set more than one consecutive pad option, either PAD1 or PADN, in a Destination Options header or Hop-by-Hop Options header.

Commenté [MB9]: Which is unlikely to know in the Global Internet

a mis en forme : Surlignage

Commenté [MB10]: That is?

* A source host SHOULD NOT send a packet with more than seven consecutive bytes of padding, using PAD1 or PADN options, in a Destination Options header or Hop-by-Hop Options header.

* A source host should follow the recommendations in Section 4.1 of [RFC8200] for extension header ordering and number of occurrences of extension headers. New extension headers should be defined following the recommendations of Section 5.2 of [RFC8504].

4. Host and ~~intermediate~~ Intermediate ~~node~~ Node ~~limits~~ Limits for ~~receiving~~ Receiving ~~extension~~ Extension ~~headers~~Headers

Per [RFC8200], a destination host that receives a packet with extension headers must process all the extension headers in the packet before accepting the packet and processing the Upper-Layer header. An intermediate node must process the Routing Header and all preceding extension headers. In the case of an intermediate node, receiver limits pertaining to Destination options are only applicable to Destination options before the Routing Header.

As described in [RFC8504] a destination host may establish limits on the processing of extension headers. This document reiterates those requirements, expands the requirements to be applicable to intermediate nodes, and allows a receiving node to send an ICMP error [RFC8883] if a limit has been exceeded.

The requirements for limits related to hosts and intermediate nodes receiving packets with extension headers are:

* A host or intermediate node MAY set a limit on the maximum number of non-padding options allowed in a Destination Options header. If this limit is supported, then the maximum number SHOULD be configurable, the limit SHOULD be greater than or equal to 8, and the RECOMMENDED default value is 8. If a packet is received and the number of non-padding Destination options exceeds the limit, then the receiving node MUST discard the packet. If the receiving node discards a packet because the limit is exceeded, then it MAY send an ICMP Parameter Problem message with code 9 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

* A host or intermediate node MAY set a limit on the maximum number of non-padding options allowed in a Hop-by-Hop Options header. If this limit is supported then the maximum number SHOULD be configurable, the limit SHOULD be greater than or equal to 8, and the RECOMMENDED default value is 8. If a packet is received and the number of non-padding Hop-by-Hop options exceeds the limit, then the receiving node MAY either: 1) discard the packet, or 2) stop processing the Hop-by-Hop Options header and process the rest of the packet normally. If the receiving node discards a packet because the limit is exceeded, then it MAY send an ICMP Parameter Problem message with code 9 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

* A host or intermediate node MAY set a limit on the length of a Destination Options header. If this limit is supported, then the limit SHOULD be configurable and the limit SHOULD be greater than

a mis en forme : Surlignage

Commenté [MB11]: Such limit SHOULD also be exposed/retreivable.

See IPFIX as an example

Idem for other similar constructs

or equal to 64 bytes. If a packet is received and the length of the Destination Options header exceeds the limit, then the receiving node MUST discard the packet. If the receiving node discards the packet because the limit is exceeded, then it MAY send an ICMP Parameter Problem message with code 6 (Extension Header Too Big) [RFC8883] to the packet's source address.

* A host or intermediate node MAY set a limit on the length of a Hop-by-Hop Options header. If this limit is supported, then the limit SHOULD be configurable and the limit SHOULD be greater than or equal to 64 bytes. If a packet is received and the length of the Hop-by-Hop Options header exceeds the limit, then the receiving node MAY either: 1) discard the packet, 2) skip processing of Hop-by-Hop Options header and process the rest of the packet normally, or 3) process the options up to the one that causes the limit to be exceeded and then stop processing of the Hop-by-Hop Options header and process the rest of the packet normally. If the receiving node discards the packet because the limit is exceeded, then it MAY send an ICMP Parameter Problem message with code 6 (Extension Header Too Big) [RFC8883] to the packet's source address.

* A host MAY set a limit on the maximum length of the IPv6 header chain, or equivalently a host MAY set a limit on the aggregate length of extension headers in a packet. If the limit is set then it SHOULD be greater than or equal to 104 bytes, or equivalently, the limit on aggregate header extension length SHOULD be greater than or equal to 64 bytes. If a packet is received and the length of the IPv6 header chain exceeds the limit, then the receiving host MUST discard the packet and MAY send an ICMP Parameter Problem message with code 7 (Extension Header Chain Too Long) [RFC8883] to the packet's source address.

* An intermediate node MAY set a limit on the maximum length of the IPv6 header chain up to an including the Routing Header, or equivalently an intermediate node MAY set a limit on the aggregate length of extension headers in a packet up to and including the Routing Header. If the limit is set then it SHOULD be greater than or equal to 104 bytes, or equivalently, the limit on aggregate header extension length up to and including the Routing Header SHOULD be greater than or equal to 64 bytes. If a packet is received and the aggregate length of the IPv6 header chain up to and including the Routing Header exceeds the limit, then the receiving node MUST discard the packet and MAY send an ICMP Parameter Problem message with code 7 (Extension Header Chain Too Long) [RFC8883] to the packet's source address.

* A host or intermediate node MAY limit the number of consecutive bytes of padding in PAD1 or PADN options in a Destination Options header to 7. If the limit is enforced and a packet is received containing more than 7 consecutive bytes of padding in a Destination Options header, then the receiving node MUST discard the packet. If the receiving node discards the packet because the limit is exceeded, then it MAY send an ICMP Parameter Problem message with code 9 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

* A host or intermediate node MAY limit the number of consecutive

bytes of padding in PAD1 or PADN options in a Hop-by-Hop Options header to 7. If the limit is enforced and a packet is received containing more than 7 consecutive bytes of padding in a Hop-by-Hop Options header, then the receiving node MAY either: 1) discard the packet, or 2) stop processing of Hop-by-Hop Options header and process the rest of the packet normally. If the receiving node discards the packet because the limit is exceeded, then it MAY send an ICMP Parameter Problem message with code 9 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

* A host or intermediate node MAY disallow consecutive padding options, either PAD1 or PADN, to be present in a Destination Options header. If the limit is enforced and a packet is received containing consecutive padding options in a Destination Options header, then the receiving node MUST discard the packet. If the receiving node discards the packet because the limit is exceeded, then it MAY send an ICMP Parameter Problem message with code 9 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

* A host or intermediate node MAY disallow consecutive padding options, either PAD1 or PADN, to be present in a Hop-by-Hop Options header. If the limit is enforced and a packet is received containing consecutive padding options in a Hop-by-Hop Options header, then the receiving node MAY either: 1) discard the packet, or 2) stop processing of Hop-by-Hop Options header and process the rest of the packet normally. If the receiving node discards the packet because the limit is exceeded, then it MAY send an ICMP Parameter Problem message with code 9 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

* A host or intermediate MAY enforce the recommended extension header ordering and number of occurrences of extension headers described in Section 4.1 of [RFC8200]. Per the ordering recommendations, each extension header can occur at most once in a packet with the exception of Destination Options header which can occur twice. The recommended extension header ordering per Section 4 of [RFC8200] is:

  - IPv6 header
  - Hop-by-Hop Options header
  - Destination Options header
  - Routing header
  - Fragment header
  - Authentication header
  - Encapsulating Security Payload header
  - Destination Options header
  - Upper-Layer header

Commenté [MB12]: ipv6ExtensionHeaderTypeCountList

If a host or intermediate node enforces extension header ordering and a packet is received with extension headers out of order or the number of occurrences of an extension header is greater than one, or two for the Destination Options header, then the receiving node MUST discard the packet and MAY send an ICMP Parameter Problem message with code 0 (Erroneous Header Field Encountered) [RFC4443] to the packet's source address.

Note that a host may enforce extension header ordering for all extension headers in a packet, but an intermediate node may only enforce ordering for extension headers up to and including the Routing Header.

All of the above limits, except for the limit on IPv6 header chain length and the limit on extension header ordering, are updated requirements from [RFC8504]. The changes in requirements from [RFC8504] are:

Commenté [MB13]: Check if we need to update 8504

* If a limit is exceeded that pertains to Destination Options, then the packet MUST be discarded. The rationale is that Destination Options may contain information that is necessary for correct delivery of the packet and so the options cannot be ignored.

* If a limit is exceeded an ICMP error [RFC8883] MAY be sent.

5. Router L~~l~~imits for ~~receiving~~ Receiving ~~extension~~ Extension ~~headers~~Headers

A router may establish limits for processing packets with received extension headers. If a limit is exceeded, routers SHOULD still forward the packet and SHOULD NOT drop packets because a limit is exceeded.

The requirements for limits related to a router receiving packets with extension headers are:

* If a router needs to parse the upper layer protocol (as discussed in Section 7 of [RFC9098]) then it MUST be able to correctly forward packets that contain an IPv6 header chain of 104 or fewer bytes, or equivalently, a router MUST be able to process a packet with an aggregate length of extension headers less than or equal to 64 bytes.

* If a router needs to parse the upper layer protocol, for instance to deduce the transport layer port numbers, it MUST be able to correctly forward a packet containing eight or fewer extension headers that precede the transport layer header.

* A router MAY limit the number of non-padding Hop-by-Hop options that it processes. If a packet is received with a Hop-by-Hop Options header having a number of non-padding options that exceeds the limit, then the router SHOULD stop processing the Hop-by-Hop Option header and ignore any Hop-by-Hop options beyond the limit following procedures in Section 5.2 of [RFC9673]. It is NOT RECOMMENDED that a router discards the packet because the limit is exceeded; however, if it does then the router MAY send an ICMP Parameter Problem message with code 9 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

* A router MAY set a limit on the maximum length of a Hop-by-Hop Options header. If a packet is received with a Hop-by-Hop Options header having a length that exceeds the limit, then the router SHOULD either: 1) ignore the Hop-by-Hop Options extension header and forward the packet normally, or 2) process Hop-by-Hop options that are contained within the extent of length limit, ignore any Hop-by-Hop options beyond the limit, and forward the packet normally. It is NOT RECOMMENDED that a router discards the packet because the limit is exceeded.~~;~~ ~~h~~ ~~H~~owever, if it does then the router MAY send an ICMP Parameter Problem message with code 6 (Extension Header Too Big) [RFC8883] to the packet's source address.

Commenté [MB14]: Shouldn't be this SHOULD?

* A router MAY limit the number of consecutive bytes of padding in PAD1 or PADN options in a Hop-by-Hop Options header to 7. If the limit is enforced and a packet is received containing more than 7 consecutive bytes of padding in Hop-by-Hop Options, then the router SHOULD stop processing the Hop-by-Hop Option header and ignore any Hop-by-Hop options beyond the limit. It is NOT RECOMMENDED that a router discards the packet because the limit is Exceeded.~~;~~ ~~hH~~owever, if it does then the router MAY send an ICMP Parameter Problem message with code 9 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

Commenté [MB15]: Shouldn't this be SHOULD?

* A router MAY disallow consecutive padding options, either PAD1 or PADN, to be present in the Hop-by-Hop Options header. If the limit is enforced and a packet is received containing consecutive padding options in Hop-by-Hop Options, then the router SHOULD stop processing the Hop-by-Hop Option header and ignore any Hop-by-Hop options beyond the limit. It is NOT RECOMMENDED that a router discards the packet because the limit is exceeded.~~;~~ ~~H~~~~h~~owever, if it does then the router MAY send an ICMP Parameter Problem message with code 7 (Too Many Options in Extension Header) [RFC8883] to the packet's source address.

Commenté [MB16]: Should?

6. IANA Considerations

There are no actions required for IANA defined in this document.

7. Security Considerations

Security issues with IPv6 extension headers are well known and have been documented in several places including [RFC6398], [RFC6192], [RFC7045], [RFC4942], [RFC9098], and [RFC9673].

Of particular concern is a Denial-of-Service attack (~~DOS~~DoS). For instance, since there is no hard limit on the number of options in an extension header, it is conceivable that an attacker could craft MTU-sized packets with hundreds of small Hop-by-Hop or Destination options where the option type is chosen to be one that will be unknown to receivers and the higher order type bits are set to 00 to indicate that an unknown option is ignored. A receiver that attempts to process all the options in such packet would incur significant processing cost (TLV processing is difficult to efficiently implement). A similar attack against hosts and intermediate nodes could be orchestrated by sending an MTU sized packet filled with nothing but minimal-sized Destination Options headers that only

contain padding options.

The potential for ~~DOS~~ DoS attack exists in routers, hosts, and intermediate nodes. Routers are susceptible to the attack using Hop-by-Hop options, hosts are susceptible using Hop-by-Hop options or Destination options, and intermediate nodes are susceptible using Hop-by-Hop options or Destination options before the Routing Header. Also note, the threat exists for both software and hardware implementations.

This document addresses the ~~DOS~~ DoS concern of extension headers and options in extension headers by allowing receivers to configure limits on the length or number of extension headers or options that they process. Such limits cap the amount of processing needed for extension headers and hence mitigate the DOS concerns of extension headers. These limits may be independently set for hosts, routers, and intermediate nodes.

This document does not introduce any new security concerns.

8. Acknowledgments

The author would like to thank Brian Carpenter, Bob Hinden, Nick Hilliard, Gorry Fairhurst, Darren Dukes, Jen Linkova, Ole Troan, Vasilenko Eduard, Suresh Krishnan, Erik Kline, Jon Geater, Peter Yee, and John Levine for their comments and suggestions that improved this document.

9. References

9.1. Normative References

[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>.

[RFC8504] Chown, T., Loughney, J., and T. Winters, "IPv6 Node Requirements", BCP 220, RFC 8504, DOI 10.17487/RFC8504, January 2019, <https://www.rfc-editor.org/info/rfc8504>.

[RFC8883] Herbert, T., "ICMPv6 Errors for Discarding Packets Due to Processing Limits", RFC 8883, DOI 10.17487/RFC8883,

September 2020, <https://www.rfc-editor.org/info/rfc8883>.

9.2. Informative References

[APNIC] Huston, G., "IPv6 Extension headers revisited", October 2022, <https://blog.apnic.net/2022/10/13/ipv6-extension-headers-revisited/>.

[Cus23a] Custura, A. and G. Fairhurst, "Internet Measurements: IPv6 Extension Header Edition", IEPG, IETF-116—, March 2023, <http://www.iepg.org/2023-03-26-ietf116/eh.pdf>.

[Cus23b] Custura, A., Secchi, R., Boswell, E., and G. Fairhurst, "Is it possible to extend IPv6?", Computer Communications X, October 2023, <https://www.sciencedirect.com/science/article/pii/S0140366423003705>.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <https://www.rfc-editor.org/info/rfc2460>.

[RFC4942] Davies, E., Krishnan, S., and P. Savola, "IPv6 Transition/Co-existence Security Considerations", RFC 4942, DOI 10.17487/RFC4942, September 2007, <https://www.rfc-editor.org/info/rfc4942>.

[RFC6192] Dugal, D., Pignataro, C., and R. Dunn, "Protecting the Router Control Plane", RFC 6192, DOI 10.17487/RFC6192, March 2011, <https://www.rfc-editor.org/info/rfc6192>.

[RFC6398] Le Faucheur, F., Ed., "IP Router Alert Considerations and Usage", BCP 168, RFC 6398, DOI 10.17487/RFC6398, October 2011, <https://www.rfc-editor.org/info/rfc6398>.

[RFC7045] Carpenter, B. and S. Jiang, "Transmission and Processing of IPv6 Extension Headers", RFC 7045, DOI 10.17487/RFC7045, December 2013, <https://www.rfc-editor.org/info/rfc7045>.

[RFC7112] Gont, F., Manral, V., and R. Bonica, "Implications of Oversized IPv6 Header Chains", RFC 7112, DOI 10.17487/RFC7112, January 2014, <https://www.rfc-editor.org/info/rfc7112>.

[RFC7872] Gont, F., Linkova, J., Chown, T., and W. Liu, "Observations on the Dropping of Packets with IPv6 Extension Headers in the Real World", RFC 7872, DOI 10.17487/RFC7872, June 2016, <https://www.rfc-editor.org/info/rfc7872>.

[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>.

[RFC9098] Gont, F., Hilliard, N., Doering, G., Kumari, W., Huston, G., and W. Liu, "Operational Implications of IPv6 Packets

with Extension Headers", RFC 9098, DOI 10.17487/RFC9098, September 2021, <https://www.rfc-editor.org/info/rfc9098>.

[RFC9673] Hinden, R. and G. Fairhurst, "IPv6 Hop-by-Hop Options Processing Procedures", RFC 9673, DOI 10.17487/RFC9673, October 2024, <https://www.rfc-editor.org/info/rfc9673>.

Appendix A. Deriving ~~default~~ Default ~~limits~~Limits

This appendix provides an explanation and justification for the recommended default values for limits defined in this document. The derived default values are based on current capabilities in deployment, expectations for extensibility, and an extrapolation of needs for future extensibility.

A.1. Limits on N~~n~~umber of ~~options~~Options

For the default limit for non-padding Hop-by-Hop or Destination options, consider that at the time of writing, it is observed that in the almost ~~thirty year~~thirty-year history of IPv6 there are only thirteen defined non-deprecated Destination options and Hop-by-Hop O~~o~~ptions and three temporarily assigned. Extrapolating for increased growth and new options, a default limit of 8 should be sufficient for the foreseeable future.

Commenté [MB17]: The causality effect is not clear

A.2. Limits on length

At the time of writing, the default ~~limit~~ 104-byte limit for the length of the IPv6 header chain is derived from an expected minimum parsing buffer size of 128 bytes. The typical sizes for parsing buffers are 64, 128, 256, or 384 bytes. [Cus23a] suggests that 128-byte parsing buffers are common and feasible on the Internet. From [APNIC]:

> The experiment used five [Destination Option] extension header lengths (8, 16, 32, 64 and 128 bytes), and in our case, the 8-, 16- and 32-byte headers had the greatest success rates, while the two larger sizes experienced greater drop rates. There is nothing obvious in the Linux source code that could explain this behaviour, unlike the PadN issues. That tends to indicate that the size-related differential response for DST Extension header handling might be due to network equipment behaviours rather than host platform behaviours.

Per [APNIC], the drop rate for Destination Options with sizes 8, 16, and 32 bytes was about 30%. The drop rates for Destination Options with size 64 was about 40%, and the drop rate with size 128 bytes was about 85%. As [APNIC] mentions, these differences are most likely due to network equipment. We can extrapolate from this data the effects of a parsing buffer. Support for ~~128~~ 128-byte extension headers implies at least a 192-byte parsing buffer, support for ~~64~~ 64-byte extension headers implies at least a 128-byte parsing buffer, and support for smaller extension headers implies a smaller parsing buffer.

Based on this analysis, assuming common support for a 128-byte parsing buffer seems reasonable. A 128-byte parsing buffer

accommodates a ~~104~~ 104-byte IPv6 header chain length including 64 bytes of extension headers. Note that 32-byte extension headers did have a bit more success than 64-byte extension headers (30% versus 40% drop rate), however requiring support for just 32 bytes of extension header would significantly limit the utility of extension headers. Therefore, 128 bytes is chosen as the expected minimum parsing buffer size on the Internet.

The 128-byte parsing buffer would be expected to at least contain:

* 16 bytes for a Layer 2 header (for instance an Ethernet header)

* 40 bytes for the IPv6 header

* 64 bytes for the extension headers

* 8 bytes for the transport layer (i.e., the first eight bytes of the transport layer header including transport layer port numbers)

A.3. Padding limits

[RFC4942] provides a rationale for limiting the number of consecutive bytes of padding:

> There is no legitimate reason for padding beyond the next eight octet boundary since the whole option header is aligned on an eight-octet boundary but cannot be guaranteed to be on a 16 (or higher power of two)-octet boundary.

This document allows a receiver to disallow consecutive padding options. The rationale is that a single PAD1 or PADN option can be used to provide 1 to 257 bytes of padding which is sufficient for any practical use case. Correspondingly, this document also recommends that a sender does not send a packet with consecutive padding options.

A.4. Limits on ~~extension~~ Extension ~~header~~ Header O~~o~~rdering and N~~n~~umber of O~~o~~ccurrences

[RFC8200] allows, but clearly does not advocate, a very flexible use of extension headers:

> IPv6 nodes must accept and attempt to process extension headers in any order and occurring any number of times in the same packet, except for the Hop-by-Hop Options header, which is restricted to appear immediately after an IPv6 header only. Nonetheless, it is strongly advised that sources of IPv6 packets adhere to the above recommended order until and unless subsequent specifications revise that recommendation.

Open-ended flexibility can be problematic in deployment. For instance, as mentioned in Section 7 filling a packet with nothing but small extension headers could be the basis of a Denial of Service attack. For this reason, allowing limits on number of extension headers and number of occurrences of extension headers in a packet is justifiable.

Similarly, allowing any extension header ordering would require nodes to process different combinations of headers which at the time of writing has no well-defined purpose. An implementation that allows any order of extension headers would be sub-optimal in performance and is potentially exposed to a ~~Denial of Service~~DoS attack.

Furthermore, the chances that someone in the foreseeable future might define a legitimate use case for out-of-order extension headers, repeated occurrences of the same extension header, or a new extension header is low. [RFC8200] strongly advises against that, a new combination of extension headers would like face issues in deployability, and historically there has never been a serious proposal for this.

Author's Address

Tom Herbert
Santa Clara, CA,
United States of America
Email: tom@herbertland.com