Internet Engineering Task Force (IETF)
Request for Comments: 9673
Updates: 8200
Category: Standards Track
ISSN: 2070-1721

R. Hinden
Check Point Software
G. Fairhurst
University of Aberdeen

October 2024

IPv6 Hop-by-Hop Options Processing Procedures

Abstract

This document specifies procedures for processing IPv6 Hop-by-Hop options in IPv6 routers and hosts. It modifies the procedures specified in the IPv6 Protocol Specification (RFC 8200) to make processing of the IPv6 Hop-by-Hop Options header practical with the goal of making IPv6 Hop-by-Hop options useful to deploy and use at IPv6 routers and hosts. This document updates RFC 8200.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9673.

Table of Contents

1. Introduction
2. Requirements Language
3. Terminology
4. Background
5. Hop-by-Hop Header Processing Procedures
   5.1. Processing the Extension Header Carrying Hop-by-Hop Options
      5.1.1. Configuration Enabling Hop-by-Hop Header Processing
   5.2. Hop-by-Hop Options Processing
      5.2.1. Router Alert Option
      5.2.2. Configuration of Hop-by-Hop Options Processing
6. Defining New Hop-by-Hop Options
   6.1. Example of Robust Usage
7. IANA Considerations

8. Security Considerations
9. Normative References
10. Informative References
Acknowledgments
Authors' Addresses

1. Introduction

This document specifies procedures for processing IPv6 Hop-by-Hop options in IPv6 routers and hosts. It modifies the procedures specified in the IPv6 Protocol Specification [RFC8200] to make processing of the IPv6 Hop-by-Hop Options header practical with the goal of making IPv6 Hop-by-Hop options useful to deploy and use at IPv6 routers and hosts.

An IPv6 packet includes Hop-by-Hop options by including a Hop-by-Hop Options header. The current list of defined Hop-by-Hop options can be found at [IANA-HBH]. The focus for this document is to set the minimum requirements for router processing of Hop-by-Hop options. It also discusses how Hop-by-Hop options are used by hosts. This document does not propose a specific bound to the number or size of Hop-by-Hop options that ought to be processed.

This document updates [RFC8200].

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.

3. Terminology

This document uses the following loosely defined terms:

Forwarding Plane: IPv6 routers exchange user or applications data through the Forwarding Plane. Routers process fields contained in IPv6 packet headers. However, they do not process information contained in packet payloads.

Control Plane: IPv6 routers exchange control information through the Control Plane. The Control Plane processes the management and routing information exchanged with other routers.

Fast Path: A path through a router that is optimized for forwarding packets. The Fast Path might be supported by Application-Specific Integrated Circuits (ASICs), a Network Processor (NP), or other special purpose hardware. This is the typical processing path within a router taken by the Forwarding Plane.

Slow Path: A path through a router that is capable of general purpose processing and is not optimized for any particular function. This processing path is used for packets that require special processing or that differ from assumptions made in Fast Path heuristics or to process router control protocols used by the Control Plane.

Full Forwarding Rate: The rate at which a router can forward packets without adversely impacting the aggregate forwarding rate. For example, a router could process packets with Hop-by-Hop options at a rate that allows it to maintain the full speed on its outgoing interfaces, which is sometimes called "wire speed".

Source: The node originating the packet.

NOTE: [RFC6192] is an example of how designs can separate Control Plane and Forwarding Plane functions. The separation between hardware and software processing described in [RFC6398] does not apply to all router architectures. However, a router that performs all or most processing in software might still incur more processing cost when providing special processing for Hop-by-Hop options.

4. Background

In early versions of the IPv6 protocol specification [RFC1883] [RFC2460], Hop-by-Hop options were required to be processed by all nodes: routers and hosts. This proved to not be practical in current high speed routers, as observed in Section 2.2 of [RFC7045]: "it is to be expected that high-performance routers will either ignore it or assign packets containing it to a slow processing path". The reasons behind this include the following:

* The inability to process Hop-by-Hop options at the Full Forwarding Rate can result in issues. In some cases, Hop-by-Hop options would be sent to the control/management components that run on the Slow Path. This could degrade a router's performance and also its ability to process critical control traffic, both of which could be exploited as a Denial-of-Service (DoS) attack against the router.

* If a subset of packets within a flow includes Hop-by-Hop options, it could lead to an increased number of reordered packets and greater reordering distances for packets delivered to the destination. Such reordering could occur if the Hop-by-Hop Options header is included only in some packets or if a specific Hop-by-Hop option results in different processing for some of the packets within the flow. Significant reordering of packets within a flow can negatively impact the performance of upper-layer protocols and should therefore be avoided.

* Packets could include multiple Hop-by-Hop options. Too many options could make the previous issues worse by increasing the resources required to process them. The total size of the options determines the number of header bytes that might need to be processed. Measurements [Cus23a] show that the probability of successful transmission across the public Internet is currently higher for packets that include Options that result in a short total Extension Header (EH) Chain size (i.e., less than 40 bytes).

[RFC6564] specifies a uniform format for new IPv6 Extension Headers, and this update was incorporated into Section 4.8 of [RFC8200] (note that [RFC8200] obsoleted [RFC2460]).

When the IPv6 protocol specification was updated and published in July 2017 as [RFC8200], the procedures relating to Hop-by-Hop options were specified (paragraphs 5 and 6 of Section 4 of [RFC8200]) as follows:

> The Hop-by-Hop Options header is not inserted or deleted, but may be examined or processed by any node along a packet's delivery path, until the packet reaches the node (or each of the set of nodes, in the case of multicast) identified in the Destination Address field of the IPv6 header. The Hop-by-Hop Options header, when present, must immediately follow the IPv6 header. Its presence is indicated by the value zero in the Next Header field of the IPv6 header.
>
> 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.

The changes meant that an implementation complied with the IPv6 protocol specification even if it did not process Hop-by-Hop options and that routers were expected to add configuration information to control whether they process the Hop-by-Hop Options header. In practice, routers may include configuration options to control which Hop-by-Hop options they will process.

The text regarding the processing of Hop-by-Hop options in [RFC8200] was not intended to change the processing of these options. It documented how they were being used in the Internet at the time RFC 8200 was published (see Appendix B of [RFC8200]). This was a constraint on publishing the IPv6 protocol specification as an IETF Standard.

The main issues remain:

* Routers can be configured to drop transit packets containing Hop-by-Hop Options that require processing by a processor that implements the Control Plane. This could be done to protect against a DoS attack on the router [RFC9098] [RFC9288].

* IPv6 packets that include a Hop-by-Hop Options header are dropped by some Internet paths. A survey in 2015 reported a high loss rate in transit Autonomous Systems (ASes) for packets that include Hop-by-Hop options [RFC7872]. The operational implications of IPv6 packets that include Extension Headers are discussed in [RFC9098]. Measurements taken in 2023 confirm this to still be the case for many types of network paths [Cus23b].

* Allowing multiple Hop-by-Hop options in a single packet in some cases consumes more router resources to process these packets. It also adds complexity to the number of permutations that might need to be processed/configured.

* Including larger or multiple Hop-by-Hop options in a Hop-by-Hop Options header increases the number of bytes that need to be processed in forwarding, which in some designs can impact the cost of processing a packet, and in turn could increase the probability of drop [RFC7872]. A larger Extension Header could also reduce the probability of a router locating all the header bytes required to successfully process an access control list operating on fields after the Hop-by-Hop Options header.

* Any option that can be used to force packets into the processor that implements the router’s Control Plane can be exploited as a DoS attack on a transit router by saturating the resources needed for router management protocols (routing protocols, network management protocols, etc.), which could cause adverse router operation. This is an issue for the Router Alert Option [RFC2711], which intentionally forwards packets to the Control Plane as discussed in [RFC6398]. This impact could be mitigated by limiting the use of Control Plane resources by a specific packet and/or by using per-function rate-limiters for packets processed by the Control Plane.

Section 3 of [RFC6398] includes a summary of processing the IP Router Alert Option:

| In a nutshell, the IP Router Alert Option does not provide a
| convenient universal mechanism to accurately and reliably
| distinguish between IP Router Alert packets of interest and
| unwanted IP Router Alert packets. This, in turn, creates a
| security concern when the IP Router Alert Option is used, because,
| short of appropriate router-implementation-specific mechanisms,

| the router slow path is at risk of being flooded by unwanted
| traffic.

This is an example of the need to limit the resources that can be consumed when a particular function is executed and to avoid consuming Control Plane resources where support for a function has not been configured.

There has been research that has discussed the general problem with dropping packets containing IPv6 Extension Headers, including the Hop-by-Hop Options header. For example, [Hendriks] states that "Dropping all packets that contain Extension Headers is a bad practice" and that "The share of traffic containing more than one EH however, is very small. For the design of hardware able to handle the dynamic nature of EHs, we therefore recommend to support at least one EH". Operational aspects of the topics discussed in this section are further discussed in [HBH].

"Transmission and Processing of IPv6 Extension Headers" [RFC7045] clarifies how intermediate nodes should process Extension Headers. This document is generally consistent with [RFC7045] and addresses an issue that was raised for discussion when [RFC2460] was updated and replaced by [RFC8200]. This document updates [RFC8200] as described in the next section and consequently clarifies the description in Section 2.2 of [RFC7045], using the language of BCP 14 [RFC2119] [RFC8174].

This document defines a set of procedures for the Hop-by-Hop Options header that are intended to make the processing of Hop-by-Hop options practical in modern routers. The common cases are that some Hop-by-Hop options will be processed across the Internet, while others will only be processed within a limited domain [RFC8799] (e.g., where a specific service is made available in that network segment that relies on one or more Hop-by-Hop options).

5. Hop-by-Hop Header Processing Procedures

This section describes several changes to [RFC8200]. Section 5.1 describes the processing of the Hop-by-Hop options Extension Header, and Section 5.2 describes the processing of individual Hop-by-Hop options. These sections update the text in paragraph 6 of Section 4 of [RFC8200] and, as noted in Section 5.2, modify Section 4.2 of [RFC8200].

5.1. Processing the Extension Header Carrying Hop-by-Hop Options

When a packet includes one or more Extension Headers, the Next Header field of the IPv6 Header identifies the type of Extension Header. It does not identify the transport protocol.

The Extension Header used to carry Hop-by-Hop options is defined in Section 4.3 of [RFC8200] and is identified by a Next Header value of 0 in the IPv6 header. Section 4.1 of [RFC8200] requires this Hop-by-Hop Options header to appear immediately after the IPv6 header. [RFC8200] also requires that a Hop-by-Hop Options header only appear at most once in a packet.

The Hop-by-Hop Options header as defined in [RFC8200] can contain one or more Hop-by-Hop options.

Routers that process the Hop-by-Hop Options header SHOULD do so using the method defined in this document. Exceptions to this SHOULD include routers that are configured to drop packets with a Hop-by-Hop Options header to protect downstream devices that do not comply with this specification (see [RFC9288]).

Even if a router does not process the Hop-by-Hop Options header (for example, when based on configuration), it MUST forward the packet normally based on the remaining Extension Header(s) after the Hop-by-Hop Options header. A router MUST NOT drop a packet solely because it contains an Extension Header carrying Hop-by-Hop options. A configuration could control whether normal processing skips any or all of the Hop-by-Hop options carried in the Hop-by-Hop Options header.

It is expected that the Hop-by-Hop Options header will be processed by the destination(s). Hosts SHOULD process the Hop-by-Hop Options header in received packets. A constrained host is an example of a node that does not process the Hop-by-Hop Options header. If a destination does not process the Hop-by-Hop Options header, it MUST process the remainder of the packet normally.

5.1.1. Configuration Enabling Hop-by-Hop Header Processing

Section 4 of [RFC8200] allows a router to control its processing of IPv6 Hop-by-Hop options by local configuration. The text is:

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

This document clarifies that a configuration could control whether processing skips any specific Hop-by-Hop options carried in the Hop-by-Hop Options header. A router that does not process the contents of the Hop-by-Hop Options header does not process any of the Option Types contained in the Hop-by-Hop Options header.

5.2. Hop-by-Hop Options Processing

A Source creating packets with a Hop-by-Hop Options header SHOULD use a method that is robust to network nodes selectively processing only some of the Hop-by-Hop options that are included in the packet or that forward packets without the option(s) being processed (see Section 6.1). A Source MAY, based on local configuration, allow only one Hop-by-Hop option to be included in a packet, or it could allow more than one Hop-by-Hop option but limit their size to increase the likelihood of successful transfer across a network path. Because some routers might only process one or a limited number of options in the Hop-by-Hop Options header, Sources are motivated to order the placement of Hop-by-Hop options within the Hop-by-Hop Options header in decreasing order of importance for their processing by nodes on the path.

A router configuration needs to avoid vulnerabilities that arise when it cannot process the first Hop-by-Hop option at the Full Forwarding Rate. Therefore, a router SHOULD NOT be configured to process the first Hop-by-Hop option if this adversely impacts the aggregate forwarding rate. A router SHOULD process additional Hop-by-Hop options, if configured to do so, providing that these also do not adversely impact the aggregate forwarding rate.

If a router is unable to process a specific Hop-by-Hop option (or is not configured to do so), it SHOULD behave in the same way specified for an unrecognized Option Type when the action bits are set to "00", and it SHOULD skip the remaining options using the "Hdr Ext Len" field in the Hop-by-Hop Options header. This field specifies the length of the Options Header in 8-octet units. After skipping an option, the router continues processing the remaining options in the header. Skipped options do not need to be verified.

The Router Alert Option [RFC2711] is an exception to this because it

is designed to tell a router that the packet needs additional processing, which is usually done in the Control Plane; see Section 5.2.1.

Section 4.2 of [RFC8200] defines the Option Type identifiers as internally encoded such that their highest-order 2 bits specify the action that must be taken if the processing IPv6 node does not recognize the Option Type. The text is:

- 00 - skip over this option and continue processing the header.
- 01 - discard the packet.
- 10 - discard the packet and, regardless of whether or not the packet's Destination Address was a multicast address, send an ICMP Parameter Problem, Code 2, message to the packet's Source Address, pointing to the unrecognized Option Type.
- 11 - discard the packet and, only if the packet's Destination Address was not a multicast address, send an ICMP Parameter Problem, Code 2, message to the packet's Source Address, pointing to the unrecognized Option Type.

This document modifies this behavior for the "01", "10", and "11" action bits so that if a router is unable to process a specific Hop-by-Hop option (or is not configured to do so), it SHOULD behave in the same way specified for an unrecognized Option Type when the action bits are set to "00". It also modifies the behavior for values "10" and "11" in the case where the packet is discarded and the node MAY send an ICMP Parameter Problem, Code 2 [RFC4443], message to the packet's Source Address, pointing to the unrecognized Option Type.

The modified text for values "01", "10", and "11" is:

- 01 - MAY discard the packet, if so configured. Nodes should not rely on routers dropping these unrecognized Option Types.
- 10 - MAY discard the packet, if so configured, regardless of whether or not the packet's Destination Address was a multicast address. If the packet was discarded, an ICMP Parameter Problem, Code 2, message MAY be sent to the packet's Source Address, pointing to the unrecognized Option Type.
- 11 - MAY discard the packet, if so configured. If the packet was discarded and the packet's Destination Address was not a multicast address, an ICMP Parameter Problem, Code 2, message MAY be sent to the packet's Source Address, pointing to the unrecognized Option Type.

When an ICMP Parameter Problem, Code 2, message is delivered to the Source, it indicates that at least one node on the path has failed to recognize the option [RFC4443]. Generating any ICMP message incurs additional router processing. Reception of this message is not guaranteed; routers might be unable to be configured so that they do not generate these messages, and they are not always forwarded to the Source. The motivation here is to loosen the requirement to send an ICMPv6 Parameter Problem message when a router forwards a packet without processing the list of all options.

5.2.1. Router Alert Option

The purpose of the Router Alert Option [RFC2711] is to tell a router that the packet needs additional processing in the Control Plane.

The Router Alert Option includes a two-octet Value field that describes the protocol that is carried in the packet. The current specified values can be found in the "IPv6 Router Alert Option Values" IANA registry [IANA-RA].

DISCUSSION

The function of a Router Alert Option can result in the processing that this specification is proposing to eliminate, that is, instructing a router to process the packet in the Control Plane. This processing causes concerns, which are discussed in Section 4. One approach would be to deprecate this, because current usage beyond the local network appears to be limited, and packets containing Hop-by-Hop options are frequently dropped. Deprecation would allow current implementations to continue, and its use could be phased out over time.

The Router Alert Option could potentially be used with new functions that have to be processed in the Control Plane. Keeping this as the single exception for processing in the Control Plane with the restrictions that follow is a reasonable compromise to allow future flexibility. These restrictions are compatible with Section 5 of [RFC6398].

As noted in [RFC6398], "Implementations of the IP Router Alert Option SHOULD offer the configuration option to simply ignore the presence of 'IP Router Alert' in IPv4 and IPv6 packets."

A node that is configured to process a Router Alert Option MUST protect itself from an infrastructure attack that could result from processing in the Control Plane. This might include some combination of an access control list to only permit access from trusted nodes, rate limiting of processing, or other methods [RFC6398].

As specified in [RFC2711], the top two bits of the Option Type for the Router Alert Option are always set to "00", indicating that the node should skip over this option as if it does not recognize the Option Type and continue processing the header. An implementation that does recognize the Router Alert Option SHOULD verify that the Router Alert Option contains a protocol, as indicated by the Value field in the Router Alert Option, that is configured as a protocol of interest to that router. A verified packet SHOULD be sent to the Control Plane for further processing [RFC6398]. Otherwise, the router implementation SHOULD forward this packet subject to all normal policies and forwarding rules.

5.2.2. Configuration of Hop-by-Hop Options Processing

A router can be configured to process a specific Option. The set of enabled options SHOULD be configurable by the operator of the router.

A possible approach to implementing this is to maintain a lookup table based on an Option Type of the IPv6 options that can be processed at the Full Forwarding Rate. This would allow a router to quickly determine if an option is supported and can be processed. If the option is not supported, then the router processes the option as described in Section 5.1 of this document.

The actions of the lookup table should be configurable by the operator of the router.

6. Defining New Hop-by-Hop Options

This section updates Section 4.8 of [RFC8200].

Any future new IPv6 Hop-by-Hop options should be designed to be processed at the Full Forwarding Rate and should have the following characteristics:

* New Hop-by-Hop options should be designed to ensure the router can process the options at the Full Forwarding Rate. That is, they should be simple to process.

* New Hop-by-Hop options should be defined with the Action type (highest-order 2 bits of the Option Type) set to "00", which enables skipping over this option and continuing with the processing of the header if a router does not recognize the option.

* The size of Hop-by-Hop options should not extend beyond what can be expected to be executed at the Full Forwarding Rate. A larger Hop-by-Hop Options header can increase the likelihood that a packet will be dropped [Cus23b].

* New Hop-by-Hop options should be designed with the expectation that a router might be configured to only process a subset of Hop-by-Hop options (e.g., the first option) in the Hop-by-Hop Options header.

* The design of protocols that use new Hop-by-Hop options should consider that a router may drop packets containing the new Hop-by-Hop option.

If a new Hop-by-Hop option does not meet these criteria, its specification must include a detailed explanation why that is the case and show that there is a reasonable expectation that the option can still proceed at the Full Forwarding Rate. This is consistent with [RFC6564]. This is consistent with [RFC6564].

The general issue of robust operation of packets with new Hop-by-Hop options is described in Section 6.1.

6.1. Example of Robust Usage

Recent measurement surveys (e.g., [Cus23a]) show that packets that include Extension Headers can cause the packets to be dropped by some Internet paths. In a limited domain, routers can be configured or updated to provide support for any required Hop-by-Hop options.

The primary motivation of this document is to make it more practical to use Hop-by-Hop options beyond such a limited domain, with the expectation that applications can improve the quality of or add new features to their offered service when the path successfully forwards packets with the required Hop-by-Hop options and otherwise refrains from using these options. The focus is on incremental deployability. A protocol feature (such as using Hop-by-Hop options) is incrementally deployable if early adopters gain some benefit on the paths being used, even though other paths do not support the protocol feature. A Source ought to order the Hop-by-Hop options that are carried in the Hop-by-Hop Options header in decreasing order of importance for processing by nodes on the path.

Methods can be developed that do not rely upon all routers to implement a specific Hop-by-Hop option (e.g., [RFC9268]) and that are robust when the current path drops packets that contain a Hop-by-Hop option (e.g., [RFC9098]).

For example, an application can be designed to first send a test packet that includes the required option or combination of options and then send other packets without including the option. The application does not send additional packets that include this option

(or set of options) until the test packet(s) is acknowledged. The need for potential loss recovery when a path drops these test packets can be avoided by choosing packets that do not carry application data that needs to be reliably delivered.

Since the set of nodes forming a path can change with time, this discovery process ought to be repeated from time to time. The process of sending packets both with and without a specific header to discover whether a path can support a specific header is sometimes called "racing". Transport protocol racing is explained in [TAPS-ARCH], and A/B protocol feature testing is described in [Tram17].

7. IANA Considerations

This document updates the processing of Hop-by-Hop options. IANA has added this document as an additional reference for the "Destination Options and Hop-by-Hop Options" registry in the "Internet Protocol Version 6 (IPv6) Parameters" registry group [IANA-HBH].

8. Security Considerations

Security issues caused by including IPv6 Hop-by-Hop options are well known and have been documented in several places, including [RFC6398], [RFC6192], [RFC7045], and [RFC9098]. The main issue, as noted in Section 4, is that any mechanism that can be used to force packets into the router’s Control Plane or Slow Path can be exploited as a DoS attack on a router by saturating the resources needed for router management (routing protocols, network management protocols, etc.), and this can cause the router to fail or perform suboptimally.

While Hop-by-Hop options are not required to be processed in the Control Plane, the Router Alert Option is the one exception that is designed to be processed in the Control Plane.

Some IPv6 nodes implement features that access more of the protocol information than a typical IPv6 router (e.g., [RFC9098]). Examples are nodes that provide DoS mitigation, firewall/access control, traffic engineering, or traffic normalization. These nodes could be configured to drop packets when they are unable to access and process all Extension Headers or are unable to locate and process the higher-layer packet information. This document provides guidance on the requirements concerning Hop-by-Hop options.

Finally, this document notes that Internet protocol processing needs to be robust for malformed/malicious protocol fields. For example, a packet with an excessive number of options could consume significant resources; inclusion of a large Extension Header could potentially cause an on-path router to be unable to utilize hardware optimizations to process later headers (e.g., to perform equal cost multipath forwarding or port filtering). This requirement is not specific to Hop-by-Hop options. It is important that implementations fail gracefully when a malformed or malicious Hop-by-Hop option is encountered.

This document changes how the Hop-by-Hop Options header is processed, which significantly reduces the attack surface. These changes include the following:

* A router configuration needs to avoid vulnerabilities that arise when it cannot process a Hop-by-Hop option at the Full Forwarding Rate; therefore, it SHOULD NOT be configured to process the Hop-by-Hop option if it adversely impacts the aggregate forwarding rate. Instead, it SHOULD behave in the same way specified for an unrecognized Option Type when the action bits are set to "00", as specified in Section 5.2.

* This document adds criteria for the Router Alert Option (Section 5.2.1) to allow control over how it is processed and describes how a node configured to support these options must protect itself from attacks by using the Router Alert Option.

* This document sets the expectation that if a packet includes a Hop-by-Hop Options header, the packet will be forwarded across the network path.

* A Source MAY include a single Hop-by-Hop option (based on local configuration) or MAY be configured to include more Hop-by-Hop options. The configuration of intermediate nodes determines whether a node processes any of these options, and if so, which ones and how many.

* This document adds guidance for the design of any future new Hop-by-Hop option that reduces the computational requirements and encourages a limit to their size.

The intent of this document is to highlight that these changes significantly reduce the security issues relating to processing the IPv6 Hop-by-Hop Options header and enable Hop-by-Hop options to be safely used in the Internet.

9. Normative References

[IANA-HBH] IANA, "Destination Options and Hop-by-Hop Options", <https://www.iana.org/assignments/ipv6-parameters/>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[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. Informative References

[Cus23a] Custura, A. and G. Fairhurst, "Internet Measurements: IPv6 Extension Header Edition", IEPG Meeting: 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, vol. 214, pp. 90-99, DOI 10.1016/j.comcom.2023.10.006, January 2024, <https://www.sciencedirect.com/science/article/pii/S0140366423003705>.

[HBH] Peng, S., Li, Z., Xie, C., Qin, Z., and G. S. Mishra, "Operational Issues with Processing of the Hop-by-Hop Options Header", Work in Progress, Internet-Draft, draft-ietf-v6ops-hbh-10, 16 February 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-v6ops-hbh-10>.

[Hendriks] Hendriks, L., Velan, P., Schmidt, RO., Boer, P., and A. Aiko, "Threats and Surprises behind IPv6 Extension

Headers", 2017 Network Traffic Measurement and Analysis Conference (TMA), DOI 10.23919/TMA.2017.8002912, August 2017, <http://dl.ifip.org/db/conf/tma/tma2017/tma2017_paper22.pdf>.

[IANA-RA] IANA, "IPv6 Router Alert Option Values", <https://www.iana.org/assignments/ipv6-routeralert-values/>.

[RFC1883] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 1883, DOI 10.17487/RFC1883, December 1995, <https://www.rfc-editor.org/info/rfc1883>.

[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>.

[RFC2711] Partridge, C. and A. Jackson, "IPv6 Router Alert Option", RFC 2711, DOI 10.17487/RFC2711, October 1999, <https://www.rfc-editor.org/info/rfc2711>.

[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>.

[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>.

[RFC6564] Krishnan, S., Woodyatt, J., Kline, E., Hoagland, J., and M. Bhatia, "A Uniform Format for IPv6 Extension Headers", RFC 6564, DOI 10.17487/RFC6564, April 2012, <https://www.rfc-editor.org/info/rfc6564>.

[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>.

[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>.

[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>.

[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>.

[RFC9268] Hinden, R. and G. Fairhurst, "IPv6 Minimum Path MTU Hop-by-Hop Option", RFC 9268, DOI 10.17487/RFC9268, August 2022, <https://www.rfc-editor.org/info/rfc9268>.

[RFC9288] Gont, F. and W. Liu, "Recommendations on the Filtering of IPv6 Packets Containing IPv6 Extension Headers at Transit

Routers", RFC 9288, DOI 10.17487/RFC9288, August 2022, <https://www.rfc-editor.org/info/rfc9288>.

[TAPS-ARCH] Pauly, T., Ed., Trammell, B., Ed., Brunstrom, A., Fairhurst, G., and C. Perkins, "Architecture and Requirements for Transport Services", Work in Progress, Internet-Draft, draft-ietf-taps-arch-19, 9 November 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-taps-arch-19>.

[Tram17] Trammell, B., Khlewind, M., De Vaere, P., Learmonth, I., and G. Fairhurst, "Tracking Transport-Layer Evolution with PATHspider", ANRW ’17: Proceedings of the 2017 Applied Networking Research Workshop, DOI 10.1145/3106328.3106336, July 2017, <https://irtf.org/anrw/2017/anrw17-final16.pdf>.

Acknowledgments

Helpful comments were received from Brian Carpenter, Ron Bonica, Ole Troan, Mike Heard, Tom Herbert, Cheng Li, ric Vyncke, Greg Mirsky, Xiao Min, Fernando Gont, Darren Dukes, Peng Shuping, Dave Thaler, Ana Custura, Tim Winters, Jingrong Xie, Lorenzo Colitti, Toerless Eckert, Suresh Krishnan, Mikael Abrahamsson, Adrian Farrel, Jie Dong, Jen Linkova, Erik Kline, and other members of the 6MAN Working Group.

Authors’ Addresses

Robert M. Hinden
Check Point Software
100 Oracle Parkway, Suite 800
Redwood City, CA 94065
United States of America
Email: bob.hinden@gmail.com

Godred Fairhurst
University of Aberdeen
School of Engineering
Fraser Noble Building
Aberdeen
AB24 3UE
United Kingdom
Email: gorry@erg.abdn.ac.uk