

IPv6 Maintenance
Internet-Draft
Updates: [2460](#),7045 (if approved)
Intended status: Standards Track
Expires: December 8, 2015

F. Baker
Cisco Systems
June 6, 2015

IPv6 Hop-by-Hop Header Handling
draft-baker-6man-hbh-header-handling-01

Abstract

This note updates the IPv6 Specification ([RFC 2460](#)), specifically commenting on the Hop-by-Hop Options Header ([section 4.3](#)) and option format and handling ([section 4.2](#)).

It also updates [RFC 7045](#), which noted that [RFC 2460](#) is widely violated in this respect, but merely legitimized this situation with a SHOULD. The present document tries to address the issue more fundamentally.

It tries to address the issue.

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 <http://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 December 8, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

Internet-Draft

June 2015

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1.](#) Introduction [2](#)
- [1.1.](#) Requirements Language [3](#)
- [2.](#) Handling of options in extension headers [3](#)
- [2.1.](#) Hop-by_hop Options [3](#)
- [2.2.](#) Changing options in transit [4](#)
- [2.3.](#) Adding headers or options in transit [4](#)
- [2.4.](#) Interactions with the Security Extension Header [4](#)
- [3.](#) Interoperation with [RFC 2460](#) [5](#)
- [4.](#) IANA Considerations [5](#)
- [5.](#) Security Considerations [5](#)
- [6.](#) Privacy Considerations [6](#)
- [7.](#) Acknowledgements [6](#)
- [8.](#) References [6](#)
- [8.1.](#) Normative References [6](#)
- [8.2.](#) Informative References [6](#)
- [Appendix A.](#) Change Log [7](#)
- Author's Address [7](#)

[1.](#) Introduction

The IPv6 Specification [[RFC2460](#)] specifies a number of extension headers. These, and the ordering considerations given, were defined based on experience with IPv4 options. They were, however, prescient with respect to their actual use - the IETF community did not know how they would be used. In at least one case, the Hop-by-Hop option, most if not all implementations implement it by punting to a software path. In the words of [[RFC7045](#)],

The IPv6 Hop-by-Hop Options header SHOULD be processed by intermediate forwarding nodes as described in [[RFC2460](#)]. However, it is to be expected that high-performance routers will either ignore it or assign packets containing it to a slow processing path. Designers planning to use a Hop-by-Hop option need to be aware of this likely behaviour.

Fernando Gont, in his Observations on IPv6 EH Filtering in the Real World [[I-D.ietf-v6ops-ipv6-ehs-in-real-world](#)], and the operational community in IPv6 Operations, consider any punt to a software path to be an attack vector. Hence, IPv6 packets containing the Hop-by-Hop

Extension Header (and in some cases, any extension header) get dropped in transit.

The subject of this document is implementation approaches to obviate or mitigate the attack vector, and updating the Hop-by-Hop option with respect to current issues.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)].

2. Handling of options in extension headers

In short, to avoid a punt to a software path, the Hop-by-Hop option SHOULD be implemented in hardware when possible.

2.1. Hop-by_hop Options

At this writing, there are three defined Hop-by-Hop options:

PAD Options: The PAD1 and PADn options [[RFC2460](#)] define empty space.

Router Alert Option: The IPv6 Router Alert Option [[RFC2711](#)] [[RFC6398](#)] is intended to force the punting of a datagram to software, in cases in which RSVP or other protocols need that to happen.

While this is not true of older hardware, modern hardware (which is to say, microcode) is capable of parsing the Extension Header chain, and can be extended to perform at least a cursory examination of the Hop-by-Hop options. For example, such hardware should be able to identify and skip the PAD1 and PADn options, and punt the Router Alert or other options to software only if configured by software to do so.

More generally, in routers that implement a fast path, the processing of the Hop-by-Hop Extension Header (which must be performed by every router a packet transits) MUST be performed in the fast path unless there is a specific reason to punt to a slower path, including that corresponding software exists in the implementation and is configured to process the option.

[2.2.](#) Changing options in transit

[Section 4.2 of \[RFC2460\]](#) explicitly allows for options that may be updated in transit. It is likely that the original authors intended that to be very simple, such as having the originating end system provide the container, and having intermediate systems update it - perhaps performing some calculation, and in any event storing the resulting value. Examples of such a use might be in [\[XCP\]](#) or [\[RCP\]](#).

As a side comment, the Routing Header, which is an extension header rather than a list of options, is treated similarly; when a system is the destination of a packet and not the last one in the Routing Header's list, it swaps the destination address with the indicated address in the list, and updates the hop count and the list depth accordingly.

Such options must be marked appropriately (their option type is of the form XX1XXXXX), and are excluded from checksum calculations in AH and ESP.

[2.3.](#) Adding headers or options in transit

Use cases under current consideration take this a step further: a router or middleware process MAY add an extension header, MAY add an option to the header, which may extend the length of the Hop-by-Hop Extension Header, or MAY process such an option in a manner that extends both the length of the option and the Extension Header containing it. The obvious implication is that other equipment in the network may not understand or implement the new option type. As

such, the Option Type value of such an option MUST indicate that it is to be skipped by a system that does not understand it. Since, by definition, it is being updated in transit and not included in any AH or ESP integrity check if present, the Option Type MUST also indicate that it may be updated in transit, and so is excluded from AH and ESP processing. By implication, such an Option Type MUST be of the form 001XXXXX.

[2.4.](#) Interactions with the Security Extension Header

The interactions with the IP Authentication Header [[RFC4302](#)] and IP Encapsulating Security Payload (ESP) [[RFC4303](#)], as in the case of existing option uses, is minimally defined. AH and ESP call for the exclusion of mutable data in their calculations by zeroing it out prior to performing the integrity check calculation. However, in the case that network operation has changed the length of the option or the extension header, that may still cause the integrity check to fail. Specifications that define such options SHOULD consider the implications of this for AH and ESP. An option whose insertion would

affect the integrity check MUST be removed prior to the integrity check, and as a result the packet restored to its state as originally sent.

[3.](#) Interoperation with [RFC 2460](#)

There are four possible modes of interaction with routers that don't implement the Hop-By-Hop Option in the fast path:

1. Presume that they cannot handle the Hop-By-Hop option at close to wire speed, and that's OK.
2. Presume that they will drop traffic containing Hop-By-Hop options.
3. Presume that they can handle the Hop-By-Hop option at or close to wire speed, and are configured to do so.
4. Presume that they don't exist, perhaps because older routers are configured to ignore all Hop-by-Hop options.

If the first model actually works in a given network, it may be

acceptable in that domain. It is not a model that will work in the general Internet, however.

The second model (which is most probable at this writing) is a description of the general Internet in 2015.

The third and fourth models, if applicable in a given context, are what one might hope for. Vendors are in a position to either have an option to ignore the Hop-By-Hop header in older equipment, or add such an option in upgraded software.

[4.](#) IANA Considerations

This memo asks the IANA for no new parameters.

[5.](#) Security Considerations

In general, modification of a datagram in transit is considered very closely from the viewpoint of the End-to-End Principle, which in this context may be summarized as "the network should do nothing that is of concern to the communicating applications or introduces operational issues." The concept of changing the length of an Extension Header or an option contained within it ([Section 2.3](#)) is of concern in that context. The obvious concern is around the interaction with AH or ESP, and a less obvious concern relates to Path MTU, which might change if the size of an underlying header

Baker

Expires December 8, 2015

[Page 5]

Internet-Draft

June 2015

changes. [Section 2.4](#) is intended to mitigate that issue. However, some ramifications, such as with Path MTU, may not be completely solvable in the general Internet, but require use cases to be confined to a network or set of consenting networks.

[6.](#) Privacy Considerations

Data formats in this memo reveal no personally identifying information.

[7.](#) Acknowledgements

This note grew out of a discussion among the author, Ole Troan, Mark Townsley, Frank Brockners, and Shwetha Bhandari, and benefited from comments by Brian Carpenter and Joe Touch.

[8.](#) References

[8.1.](#) Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.

[8.2.](#) Informative References

- [I-D.ietf-v6ops-ipv6-ehs-in-real-world]
Gont, F., Linkova, J., Chown, T., and S. LIU,
"Observations on IPv6 EH Filtering in the Real World",
[draft-ietf-v6ops-ipv6-ehs-in-real-world-00](#) (work in progress), April 2015.
- [RCP] Dukkupati, N., "Rate Control Protocol (RCP): Congestion control to make flows complete quickly", Stanford University , 2006.
- [RFC2711] Partridge, C. and A. Jackson, "IPv6 Router Alert Option", [RFC 2711](#), October 1999.
- [RFC4302] Kent, S., "IP Authentication Header", [RFC 4302](#), December 2005.
- [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", [RFC 4303](#), December 2005.

- [RFC6398] Le Faucheur, F., "IP Router Alert Considerations and Usage", [BCP 168](#), [RFC 6398](#), October 2011.
- [RFC7045] Carpenter, B. and S. Jiang, "Transmission and Processing of IPv6 Extension Headers", [RFC 7045](#), December 2013.
- [XCP] Katabi, D., Handley, M., and C. Rohrs, "Congestion control for high bandwidth-delay product networks", SIGCOMM

Symposium proceedings on Communications architectures and protocols , 2002.

[Appendix A](#). Change Log

Initial Version: June 2015

Author's Address

Fred Baker
Cisco Systems
Santa Barbara, California 93117
USA

Email: fred@cisco.com