Network Working Group
Internet-Draft
Intended status: Informational
Expires: July 26, 2021

S. Peng
Z. Li
Huawei Technologies
C. Xie
China Telecom
Z. Qin
China Unicom
G. Mishra
Verizon Inc.

January 22, 2021

Processing of the Hop-by-Hop Options Header
draft-peng-v6ops-hbh-03

Abstract

This document describes the processing of the Hop-by-Hop Options Header (HBH) in today's routers in the aspects of standards specification, common implementations, and default operations. This document outlines the reasons why the Hop-by-Hop Options Header is rarely utilized in current networks. In addition, this document describes how the HBH could be used as a powerful mechanism allowing deployment and operations of new services requiring a more optimized way to leverage network resources of an infrastructure. The Hop-by-Hop Options Header is taken into consideration by several network operators as a valuable container for carrying the information facilitating the introduction of new services. The desired, and proposed, processing behavior of the HBH and the migration strategies towards it are also suggested.

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] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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/.

Table of Contents

1. Introduction

Due to historical reasons, such as incapable ASICs, limited IPv6 deployments, and few service requirements, the most common Hop-by-Hop Options header (HBH) processing implementation is that the node sends the IPv6 packets with the Hop-by-Hop Options header to the slow path

(i.e., the control plane) of the node. The option type of each option carried within the Hop-by-Hop Options header will not even be examined before the packet is sent to the slow path. Very often, such processing behavior is the default configuration or, even worse, is the only behavior of the ipv6 implementation of the node.

Such default processing behavior of the Hop-by-Hop Options header could result in various unpleasant effects such as a risk of Denial of Service (DoS) attack on the router control plane and inconsistent packet drops due to rate limiting on the interface between the router control plane and forwarding plane, which will impact the normal end-to-end IP forwarding of the network services.

This actually introduced a circular problem:

-> An implementation problem caused HBH to become a DoS vector.

-> Because HBH is a DoS vector, network operators deployed ACLs that discard packets containing HBH.

-> Because network operators deployed ACLs that discard packets containing HBH, network designers stopped defining new HBH Options.

-> Because network designers stopped defining new HBH Options, the community was not motivated to fix the implementation problem that cause HBH to become a DoS vector.

The purpose of this draft is to break the cycle described above, fixing the problem that caused HBH not actually being utilized in operators' networks so to allow a better leverage of the HBH capability.

Driven by the wide deployments of IPv6 and ever-emerging new services, the Hop-by-Hop Options Header is taken as a valuable container for carrying the information to facilitate these new services.

This document suggests the desired processing behavior and the migration strategies towards it.

2. Modern Router Architecture

Modern router architecture design maintains a strict separation of the router control plane and its forwarding plane [RFC6192], as shown in Figure 1. Either the control plane or the forwarding plane is composed of both software and hardware, but each plane is responsible for different functions.

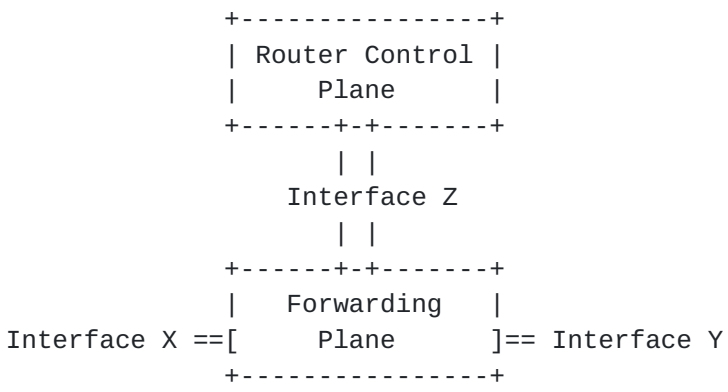

Figure 1. Modern Router Architecture

The router control plane supports routing and management functions, handling packets destined to the device as well as building and sending packets originated locally on the device, and also drives the programming of the forwarding plane. The router control plane is generally realized in software on general-purpose processors, and its hardware is usually not optimized for high-speed packet handling. Because of the wide range of functionality, it is more susceptible to security vulnerabilities and a more likely a target for a DoS attack.

The forwarding plane is typically responsible for receiving a packet on an incoming interface, performing a lookup to identify the packet's next hop and determine the outgoing interface towards the destination, and forwarding the packet out through the appropriate outgoing interface. Typically, forwarding plane functionality is realized in high-performance Application Specific Integrated Circuits (ASICs) or Network Processors (NPs) that are capable of handling very high packet rates.

The router control plane interfaces with its forwarding plane through the Interface Z, as shown in the Figure 1, and the forwarding plane connects to other network devices via Interfaces such as X and Y. Since the router control plane is vulnerable to the DoS attack, usually a traffic filtering mechanism is implemented on Interface Z in order to block unwanted traffic. In order to protect the router control plane, a rate-limiting mechanism is always implemented on this interface. However, such rate limiting mechanism will always cause inconsistent packet drops, which will impact the normal IP forwarding.

Semiconductor chip technology has advanced significantly in the last decade, and as such the widely used network processing and forwarding process can now not only forward packets at line speed, but also easily support other feature processing such as QoS for DiffServ/

MPLS, Access Control List (ACL), Firewall, and Deep Packet Inspection (DPI).

A Network Processing Unit (NPU) is a non-ASIC based Integrated Circuit (IC) that is programmable through software. It performs all packet header operations between the physical layer interface and the switching fabric such as packet parsing and forwarding, modification, and forwarding. Many equipment vendors implement these functions in fixed function ASICs rather than using "off-the-shelf" NPUs, because of proprietary algorithms. Classification Co-processor is a specialized processor that can be used to lighten the processing load on an NPU by handling the parsing and classification of incoming packets such as IPv6 extended header HBH options processing. This advancement enables network processors to do the general process to handle simple control messages for traffic management, such as signaling for hardware programming, congestion state report, OAM, etc. Industry trend is for intelligent multi-core CPU fast path hardware using modern NPUs for forwarding packets at line rate while still being able to perform other complex tasks such as HBH forwarding options processing without having to punt to slow path.

Many of the fast-path packet-processing devices employed in modern switch and router designs are fixed-function ASICs to handle proprietary functions. While these devices can be very efficient for the set of functions they are designed for, they can be very inflexible. There is a tradeoff of price, performance and flexibility when vendors make a choice to use a fixed function ASIC as opposed to NPU. Due to the inflexibility of the fixed function ASIC, tasks that require additional processing such as IPv6 HBH header processing must be punted to the slow path. This problem is still a challenge today and is the reason why operators to protect against control plane DOS attack vector must drop or ignore HBH options. As industry shifts to Merchant Silicon based NPU evolution from fixed function ASIC, the gap will continue to close increasing the viability ubiquitous HBH use cases due to now processing in the fast path.

Most modern routers maintain a strict separation between forwarding plane and control plane hardware. Forwarding plane "fast path" bandwidth and resources are plentiful, while control plane "slow path" bandwidth and resources are constrained. In order to protect scarce control plane resources, routers enforce policies that restrict access from the forwarding plane to the control plane. Effective policies address packets containing the HBH Options Extension header, because HBH control options require access from the forwarding plane to the control plane. Many network operators perceive HBH Options to be a breach of the separation between the forwarding and control planes. In this case HBH control options

would be required to be punted to slow path by fixed function ASICs as well as NPUs.

The maximum length of an HBH Options header is 2,048 bytes. A source node can encode hundreds of options in 2,048 bytes. With today's technology it would be cost prohibitive to be able to process hundreds of options with either NPU or proprietary fixed function ASIC.

While [RFC8200] 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. This can be beneficial in cases where transit nodes are legacy hardware and the destination endpoint PE is newer NPU based hardware that can process HBH in the fast path.

IPv6 Extended Header limitations that need to be addressed to make HBH processing more efficient and viable in the fast path:

[RFC8504] defines the IPv6 node requirements and how to protect a node from excessive header chain and excessive header options with various limitations that can be defined on a node. [RFC8883] defines ICMPv6 Errors for discarding packets due to processing limits. Per [RFC8200] HBH options must be processed serially. However, an implementation of options processing can be made to be done with more parallelism in serial processing grouping of similar options to be processed in parallel.

The IPv6 standard does not currently limit the header chain length or number of options that can be encoded.

Each Option is encoded in a TLV and so processing of a long list of TLVs is expensive. Zero data length encoded options TLVs are a valid option. A DOS vector could be easily generated by encoding 1000 HBH options (Zero data length) in a standard 1500 MTU packet. So now imagine if you have a Christmas tree long header chain to parse each with many options.

3. Specification of RFC 8200

[RFC8200] defines several IPv6 extension header types, including the Hop-by-Hop (HBH) Options header. As specified in [RFC8200], the Hop-by-Hop (HBH) Options header is used to carry optional information that will be examined and processed by every node along a packet's delivery path, and it is identified by a Next Header value of zero in the IPv6 header.

The Hop-by-Hop (HBH) Options header contains the following fields:

-- Next Header: 8-bit selector, identifies the type of header immediately following the Hop-by-Hop Options header.

-- Hdr Ext Len: 8-bit unsigned integer, the length of the Hop-by-Hop Options header in 8-octet units, not including the first 8 octets.

-- Options: Variable-length field, of length such that the complete Hop-by-Hop Options header is an integer multiple of 8 octets long.

The Hop-by-Hop (HBH) Options header carries a variable number of "options" that are encoded in the format of type-length-value (TLV).

The highest-order two bits (i.e., the ACT bits) of the Option Type specify the action that must be taken if the processing IPv6 node does not recognize the Option Type. The third-highest-order bit (i.e., the CHG bit) of the Option Type specifies whether or not the Option Data of that option can change en route to the packet's final destination.

While [RFC2460] required that all nodes must examine and process the Hop-by-Hop Options header, with [RFC8200] it is 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. It means that the HBH processing behavior in a node depends on its configuration.

However, in the current [RFC8200], there is no explicit specification of the possible configurations. Therefore, the nodes may be configured to ignore the Hop-by-Hop Options header, drop packets containing a Hop-by-Hop Options header, or assign packets containing a Hop-by-Hop Options header to a slow processing path [RFC8200]. Because of these likely uncertain processing behaviors, new hop-by-hop options are not recommended.

4. Common Implementations

In the current common implementations, once an IPv6 packet, with its Next Header field set to 0, arrives at a node, it will be directly sent to the slow path (i.e., the control plane) of the node. With such implementations, the value of the Next Header field in the IPv6 header is the only trigger for the default processing behavior. The option type of each option carried within the Hop-by-Hop Options header will not even be examined before the packet is sent to the slow path.

Very often, such processing behavior is the default configuration on the node, which is embedded in the implementation and cannot be changed or reconfigured.

Another critical component of IPv6 HBH processing which is in some cases is overlooked is the operator core network which can be designed to use the global Internet routing table for internet traffic and in other cases use an overlay MPLS VPN to carry Internet traffic. In the global Internet routing table scenario where only an underlay global routing table exists, and no VPN overlay carrying customer Internet traffic, the IPv6 HBH options can be used as a DOS attack vector for both the operator nodes, adjacent inter-as peer nodes as well as customer nodes along a path. In a case where the Internet routing table is carried in a MPLS VPN overlay payload, the HBH options header does not impact the operator underlay framework and only impacts the VPN overlay payload and thus the operator underlay topmost label global table routing FEC LSP instantiation is not impacted as the operator underlay is within the operators closed domain. However HBH options DOS attack vector in the VPN overlay can still impact the customer CE destination end nodes as well as other adjacent inter-as operators that only use underlay global Internet routing table. In an operator closed domain where MPLS VPN overlay is utilized to carry internet traffic, the operator has full control of the underlay and IPv6 Extended header chain length as well as the number of HBH options encoded. However in contrast, in the global routing table scenario for Internet traffic there is no way to control the IPv6 Extended header chain lenghth as well as the number of HBH forward or HBH control options encoded.

4.1. Historical Reasons

When IPv6 was first implemented on high-speed routers, HBH options were not yet well-understood and ASICs were not as capable as they are today. So, early IPv6 implementations dispatched all packets that contain HBH options to their slow path.

4.2. Consequences

Such implementation introduces a risk of a DoS attack on the control plane of the node, and a large flow of IPv6 packets could congest the slow path, causing other critical functions (including routing and network management) that are executed on the control plane to fail. Rate limiting mechanisms will cause inconsistent packet drops and impact the normal end-to-end IP forwarding of the network services.

5. Operators' Typical Processing

To mitigate this DoS vulnerability, many operators deployed Access Control Lists (ACLs) that discard all packets containing HBH Options.

[RFC6564] shows the Reports from the field indicating that some IP routers deployed within the global Internet are configured either to ignore or to drop packets having a hop-by-hop header. As stated in [RFC7872], many network operators perceive HBH Options to be a breach of the separation between the forwarding and control planes. Therefore, several network operators configured their nodes so as to discard all packets containing the HBH Options Extension Header, while others configured nodes to forward the packet but to ignore the HBH Options. [RFC7045] also states that hop-by-hop options are not handled by many high-speed routers or are processed only on a slow path.

Due to such behaviors observed and described in these specifications, new hop-by-hop options are not recommended in [RFC8200] hence the usability of HBH options is severely limited.

6. New Services

As IPv6 is being rapidly and widely deployed worldwide, more and more applications and network services are migrating to or directly adopting IPv6. More and more new services that require HBH are emerging and the HBH Options header is going to be utilized by the new services in various scenarios.

In-situ OAM (IOAM) with IPv6 encapsulation [I-D.ietf-ippm-ioam-ipv6-options] is one of the examples. IOAM in IPv6 is used to enhance diagnostics of IPv6 networks and complements other mechanisms, such as the IPv6 Performance and Diagnostic Metrics Destination Option described in [RFC8250]. The IOAM data fields are encapsulated in "option data" fields of the Hop-by-Hop Options header if Pre-allocated Tracing Option, Incremental Tracing Option, or Proof of Transit Option are carried [I-D.ietf-ippm-ioam-data], that is, the IOAM performs per hop.

Alternate Marking Method can be used as the passive performance measurement tool in an IPv6 domain. The AltMark Option is defined as a new IPv6 extension header option to encode alternate marking technique and Hop-by-Hop Options Header is considered [I-D.ietf-6man-ipv6-alt-mark].

The Minimum Path MTU Hop-by-Hop Option is defined in [I-D.ietf-6man-mtu-option] to record the minimum Path MTU along the forward path between a source host to a destination host. This Hop-

by-Hop option is intended to be used in environments like Data Centers and on paths between Data Centers as well as other environments including the general Internet. It provides a useful tool for allowing to better take advantage of paths able to support a large Path MTU.

As more services start utilizing the HBH Options header, more packets containing HBH Options are going to be injected into the networks. According to the current common configuration in most network deployments, all the packets of the new services are going to be sent to the control plane of the nodes, with the possible consequence of causing a DoS on the control plane. The packets will be dropped and the normal IP forwarding may be severely impacted. The deployment of new network services involving multi-vendor interoperability will become impossible.

7. The Desired Processing Behavior

The following requirements SHOULD be met:

o The control plane SHOULD be protected from undesired traffic.

-* The HBH options header SHOULD NOT be directly sent to the control plane once the packets are received since these options may not aim for the control plane.

-* The HBH options that are not supposed to be processed by the control plane SHOULD NOT be sent to the control plane, potentially causing the DoS attack.

o Since generally the two types of HBH options (control plane (e.g., Route Alert Option [RFC2711]) and forwarding plane (e.g., AltMark Option [I-D.ietf-6man-ipv6-alt-mark])) serve different purposes and require different processing procedures on a node, they should be encoded separately and carried in different packets.

Note: More details on the two types of HBH options can be found in [I-D.li-6man-hbh-fwd-hdr].

o The packets carrying the HBH Forwarding Options are supposed to be maintained in the forwarding plane rather than being directly sent up to the control plane. While the packets carrying the HBH Control Options are supposed to be sent to the control plane.

o The source node SHOULD NOT encode the HBH Options that exceed the maximum length of an HBH Options header i.e. 2,048 bytes.

- The source node SHOULD NOT encode the number of HBH Options that exceeds the lowest processing capability of the nodes along the path.

- The source node SHOULD NOT encode the HBH Options that exceed the maximum overall length of the IPv6 extensions header chain.

- The options aimed for the control plane are better if they do not consume the forwarding plane resources.

- A simple and efficient way to discriminate the two types of HBH options is required.

- The new deployments should be compatible with the existing deployments, since default configuration of some devices running in the networks cannot be changed or reconfigured. The update of the networks in operation will usually take time.

- If the IPv6 extension header including the HBH options header of a packet cannot be recognized by the node, or the option in the HBH header is unknown to the node, and the node is not the destination of the packet, the packet SHOULD NOT be dropped or sent to the control plane, rather this unrecognized extension header should be skipped and the rest of the packet should be processed.

8. Migration Strategies

In order to achieve the desired processing behavior of the HBH options header and facilitate the ever-emerging new services to be deployed in operators' networks across multiple vendors' devices, the migration can happen in three parts as described below:

1. The source of the HBH options header encapsulation.

The information to be carried in the HBH options header needs to be first categorized and encapsulated into either control options or forwarding options, and then encapsulated in different packets.

2. The nodes within the network.

The nodes within the network are updated to the proposed behavior introduced in the previous section.

3. The edge nodes of the network.

The edge nodes should check whether the packet contains an HBH header with control or forwarding option. Packets with a control option may

still be filtered and dropped while packets with forwarding option SHOULD be allowed by the ACL.

If it is certain that there is no harm that can be introduced by the HBH control options to the nodes and the services, they can also be allowed.

Note: During the migration stage, the nodes that are not yet updated will stay with their existing configurations.

9. Security Considerations

The same as the Security Considerations apply as in [RFC8200] for the part related with the HBH Options header.

10. IANA Considerations

This document does not include an IANA request.

11. Acknowledgements

The authors would like to acknowledge Ron Bonica, Fred Baker, Bob Hinden, Stefano Previdi, and Donald Eastlake for their valuable review and comments.

12. References

12.1. 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>.

[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>.

[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>.

[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>.

[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>.

12.2. Informative References

[I-D.ietf-6man-ipv6-alt-mark] Fioccola, G., Zhou, T., Cociglio, M., Qin, F., and R. Pang, "IPv6 Application of the Alternate Marking Method", draft-ietf-6man-ipv6-alt-mark-02 (work in progress), October 2020.

[I-D.ietf-6man-mtu-option] Hinden, R. and G. Fairhurst, "IPv6 Minimum Path MTU Hop-by-Hop Option", draft-ietf-6man-mtu-option-04 (work in progress), October 2020.

[I-D.ietf-ippm-ioam-data] Brockners, F., Bhandari, S., and T. Mizrahi, "Data Fields for In-situ OAM", draft-ietf-ippm-ioam-data-11 (work in progress), November 2020.

[I-D.ietf-ippm-ioam-ipv6-options] Bhandari, S., Brockners, F., Pignataro, C., Gredler, H., Leddy, J., Youell, S., Mizrahi, T., Kfir, A., Gafni, B., Lapukhov, P., Spiegel, M., Krishnan, S., Asati, R., and M. Smith, "In-situ OAM IPv6 Options", draft-ietf-ippm-ioam-ipv6-options-04 (work in progress), November 2020.

[I-D.li-6man-hbh-fwd-hdr] Li, Z. and S. Peng, "Hop-by-Hop Forwarding Options Header", draft-li-6man-hbh-fwd-hdr-00 (work in progress), July 2020.

[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>.

[RFC8250] Elkins, N., Hamilton, R., and M. Ackermann, "IPv6 Performance and Diagnostic Metrics (PDM) Destination Option", RFC 8250, DOI 10.17487/RFC8250, September 2017, <https://www.rfc-editor.org/info/rfc8250>.

[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>.

Authors' Addresses

Shuping Peng
Huawei Technologies
Beijing
China

Email: pengshuping@huawei.com

Zhenbin Li
Huawei Technologies
Beijing
China

Email: lizhenbin@huawei.com

Chongfeng Xie
China Telecom
China

Email: xiechf@chinatelecom.cn

Zhuangzhuang Qin
China Unicom
Beijing
China

Email: qinzhuangzhuang@chinaunicom.cn

Gyan Mishra
Verizon Inc.
USA

Email: gyan.s.mishra@verizon.com