IPPM
Internet-Draft
Intended status: Standards Track
Expires: July 8, 2021

H. Song
Futurewei
Z. Li
S. Peng
Huawei Technologies
J. Guichard
Futurewei

January 4, 2021

Approaches on Supporting IOAM in IPv6 draft-song-ippm-ioam-ipv6-support-02

Abstract

It has been proposed to encapsulate IOAM tracing option data fields in IPv6 HbH options header. However, due to size of the trace data and the extension header location in the IPv6 packets, the proposal creates practical challenges for implementation, especially when other extension headers, such as a routing header, also exist and require in-network processing. We propose several alternative approaches to address this challenge, including separating the IOAM trace data to a different extension header, using the postcard-based telemetry (e.g., IOAM DEX) instead, and applying the segment IOAM trace export scheme, based on the network scenario and application requirements. We discuss the pros and cons of each approach and hope to foster standard convergence and industry adoption.

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 RFC 2119 [RFC2119].

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 July 8, 2021.

Copyright Notice

Copyright (c) 2021 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 (https://trustee.ietf.org/license-info) in effect on the date of 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
2. IOAM Data Separate and Postpose 4
2.1. IOAM Trace Data Encapsulation 5
3. Segment IOAM Data Export 5
3.1. Independent of SRv6 5
3.2. Export at SRv6 node 6
4. Direct Export Option 7
5. Comparison 7
6. Security Considerations 8
7. Normative References 8
Authors' Addresses 10

1. Introduction

In-situ OAM (IOAM) [I-D.ietf-ippm-ioam-data] defines two tracing options, pre-allocated tracing option and incremental tracing option, which record hop-by-hop data along a packet's forwarding path. The draft [I-D.ietf-ippm-ioam-ipv6-options] proposes the method to encapsulate IOAM trace option data fields in IPv6. Because the tracing options requires per hop processing, such options can only be encapsulated in IPv6 Hop-by-Hop (HbH) options header. The draft [I-D.ioametal-ippm-6man-ioam-ipv6-deployment] further describes some deployment approaches.

[RFC8200] mandates that the HbH options header, if exists, must be the first extension header following the IPv6 header. However, the IOAM trace data can be large, which easily amount to tens to hundreds of bytes, making accessing other headers after it difficult or even impossible. There are practical limitations on how far the hardware

can reach into a packet in forwarding hardware. The IOAM tracing option cannot be applied if it makes other extension headers inaccessible. Even if the other headers can be reached, the deeper they are, the higher the cost to access and process them, and the lower the forwarding performance. A potentially more detrimental issue is that the incremental tracing option will expand the HbH header at each hop and push back all other headers, which keeps shifting the locations of the other extension headers, further complicating the hardware implementation and impeding the forwarding.

The issue becomes more severe when SRv6 and IOAM coexist. The Segment Routing Extension Header (SRH) [I-D.ietf-6man-segment-routing-header] is encapsulated in a routing header which is after the HbH options header. SRH itself can be large. It requires read and write operations at each SRv6 node. If it is deeply embedded in a packet and its location keeps shifting, either it is beyond the reach of hardware or the forwarding performance degrades.

We can avoid the problem by simply not using both at the same time, but apparently this is not ideal, because IOAM is an important OAM tool and it is even more wanted when SRv6 brings more operational complexity into IPv6 networks.

Our second recourse is to limit the IOAM to SRv6 nodes only. That is, consider SRv6 as an overlay tunnel over IPv6 and apply the IOAM pipe mode as discussed in [I-D.song-ippm-ioam-tunnel-mode], which only collects data at each SRv6 nodes. To realize this, [I-D.ali-spring-ioam-srv6] describes an approach that encapsulates the IOAM option data fields in an SRH TLV. [I-D.song-6man-srv6-pbt] and [I-D.ietf-6man-spring-srv6-oam] describe another approach to enable postcard-based telemetry for SRv6 without needing IOAM option encapsulation. In either case, the SRH is close to the packet front and its location is fixed. [I-D.song-spring-siam] proposes to support IOAM in the payload of the dedicated SRv6 probing packets only. While these approaches are useful for use cases that only need to monitor the segment end points, it fails to cover all the IPv6 nodes in a network.

So the proposition of this draft is, suppose we need to apply IOAM on all nodes in an SRv6 network, how we can amend the approach in [I-D.ietf-ippm-ioam-ipv6-options] or use alternative approaches to circumvent the aforementioned issues. In this draft, we propose three such approaches: (1) separating the IOAM trace data to a different extension header, (2) using the postcard-based telemetry (e.g., IOAM DEX) instead, and (3) applying the segment IOAM trace export scheme. We discuss the pros and cons of each approach and hope to foster standard convergence and industry adoption.

2. IOAM Data Separate and Postpose

An IOAM trace type data fields contain two parts: instruction and trace data. Although by convention the trace data part immediately follow the instruction part, there is not fundamental reason why these two parts must stick together. This observation provides us an optimization opportunity to amend the original proposal in [I-D.ietf-ippm-ioam-ipv6-options].

We separate the IOAM trace type data fields into the instruction part and the trace data part. We encapsulate only the instruction part in the HbH options header, and encapsulate the trace data part in another metadata extension header after all the IPv6 extension headers and before upper layer protocol headers. This arrangement allows high performance hardware implementation. When using the incremental data trace, even if the data trace size increases at each node, all IPv6 extension headers remain intact and new data is inserted at a fixed location.

Figure 1 shows the HbH option format for IOAM trace type instruction. The field specification is identical to that in [RFC8200] and [I-D.ietf-ippm-ioam-data].

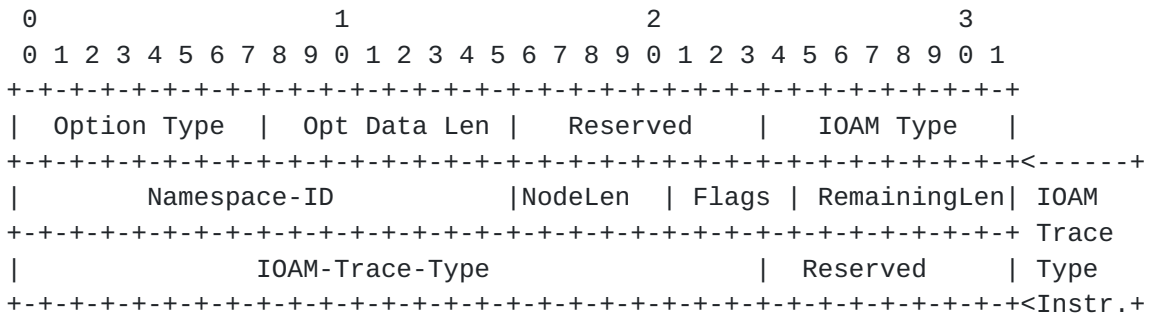

Figure 1: HbH Option Format for IOAM Trace Type Instruction

Figure 2 shows the TLV option format for IOAM trace type data. The IOAM trace type data format is compliant with [I-D.ietf-ippm-ioam-data].

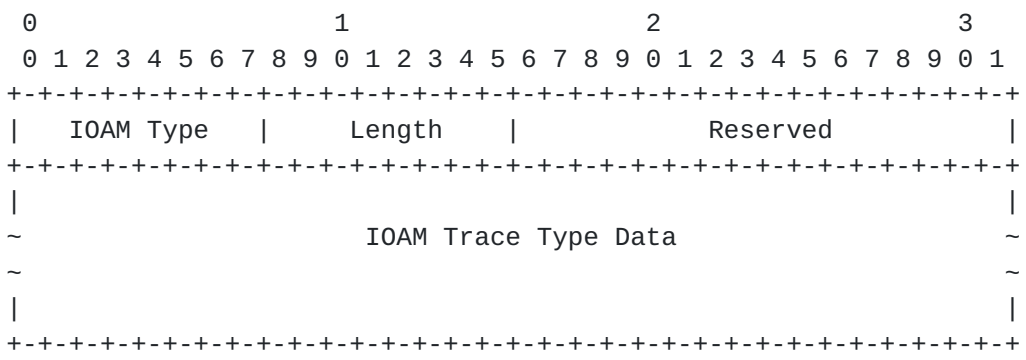

Figure 2: Option Format for IOAM Trace Type Data

2.1. IOAM Trace Data Encapsulation

We have basically two methods to encapsulate the IOAM trace data. First, we can define a new IPv6 extension header which is dedicated to metadata. Once standardized, this extension header can also be used to host potential metadata from other applications such as NSH for SFC [RFC8300]. Second, this option can be carried as a TLV option in another existing extension header such as the destination header. The only requirement is that this extension header should be the last one in the extension header chain. The first method is cleaner but it requires extra standard effort; the second method is simpler but it needs to overcome the access constraints exerted by [RFC8300].

3. Segment IOAM Data Export

If the overhead of the IOAM trace type data fields is under control, we may still manage to encapsulate both instruction and data in HbH options header as in [I-D.ietf-ippm-ioam-ipv6-options]. To this end, we introduce two sub-approaches.

3.1. Independent of SRv6

[I-D.song-ippm-segment-ioam] proposes an enhancement to IOAM trace type which can configure the allowable overhead of the IOAM trace type data fields. Once the trace data size is up to the limit at a network node (i.e., a segment or a fixed number of network nodes are traversed), the trace data will be stripped and exported so room is made to accommodate new trace data from nodes in the next segment of the forwarding path.

This approach requires some moderate updates to the IOAM trace type data fields, as described in [I-D.song-ippm-segment-ioam]. Figure 3 shows the format of the HbH Option Header containing Segment IOAM trace type data fields. A flag bit (#23) in the Flags field is used

to indicate the current header is a segment IOAM header. In this context, the last octet in the IOAM header is partitioned into two 4-bit nibbles. The first nibble (SSize) is used to save the segment size and the second nibble (RHop) is used to save the remaining hops. This limits the maximum segment size to 15.

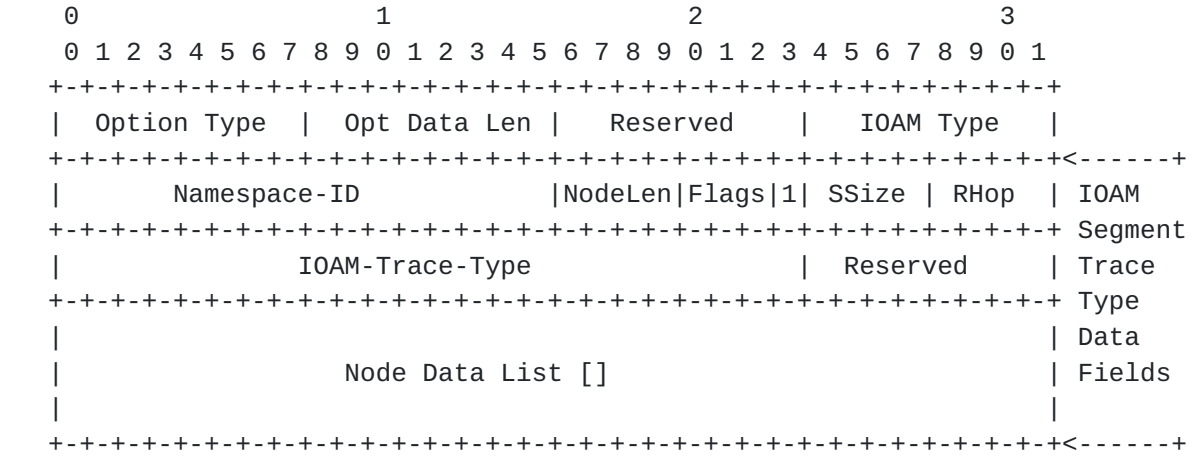

Figure 3: HbH Option Format for Segment IOAM Trace Type Data Fields

At the beginning of each segment, the segment size (SSize) and the remaining hops (RHop) are initialized: RHop is set to equal to SSize. At each hop, if RHop is not zero, the node data is added to the node data list and then RHop is decremented by 1. If RHop is equal to 0 when receiving the packet, the node needs to remove (in incremental trace option) or clear (in pre-allocated trace option) the IOAM node data list and reset RHop to SSize.

In this case, if we use the IOAM pre-allocated trace type, the size and location of each IPv6 extension header is fixed and predictable, and the hardware capability and performance can be guaranteed.

3.2. Export at SRv6 node

Whenever a packet with the IOAM option reaches a SRv6 node which needs to access the SRH, we can configure the node to export immediately the IOAM trace data accumulated so far. In this case, basically at each SRv6 node, the HbH header size is fixed and the header contains an IOAM option with only the instruction part. After the SRH processing, this node can add local IOAM trace data in the HbH option header before forwarding the packet.

The incremental trace type can be used in this approach. In an extreme case when every node is also an SRv6 node, this approach regresses to a per-hop postcard-based telemetry approach as described in [I-D.song-ippm-postcard-based-telemetry]. In this case, the HbH

option for IOAM can even be avoided altogether if we can find a way to simply mark the packet for postcard export.

4. Direct Export Option

As an embodiment of the PBT-I approach introduced in [I-D.song-ippm-postcard-based-telemetry], IOAM Direct Export (DEX) Option Type discussed in [I-D.ioamteam-ippm-ioam-direct-export] can be used to replace the IOAM trace type. IOAM DEX only needs to encapsulate a fix-size instruction header in the HbH option header.

Figure 4 shows the HbH option format for IOAM DEX type fields. The field specification is identical to that in [RFC8200] and [I-D.ioamteam-ippm-ioam-direct-export].

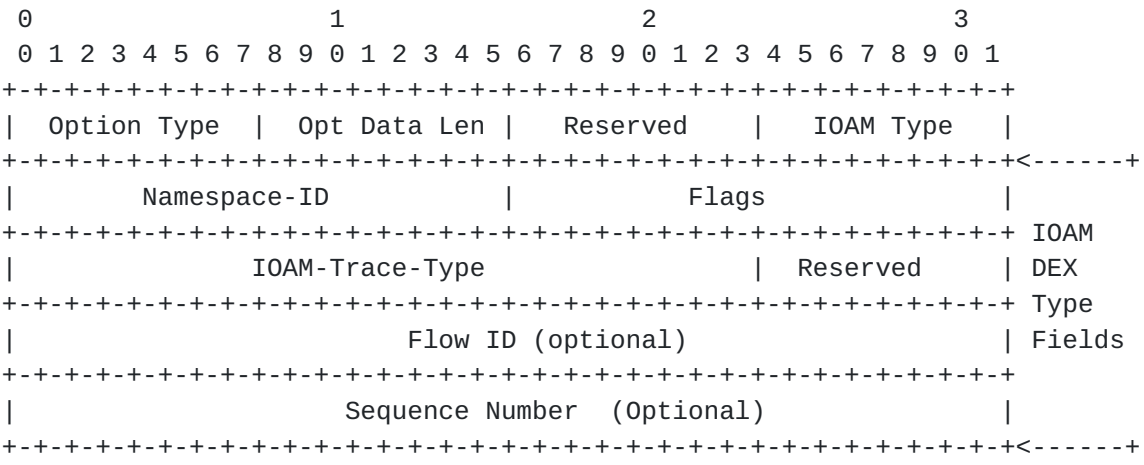

Figure 4: HbH Option Format for IOAM DEX Type Fields

5. Comparison

The following table compares the existing approach and the four other alternative approaches proposed in this draft.

Approach	Pros	Cons
IOAM Trace in HbH	Comply w/ IOAM Data Spec	Variable and long HbH header impeding access of later extension headers
IOAM Trace Data Separate and Postpose (Sec. 2)	Fix-size and short HbH header, good for later extension header access	Need extra extension header to hold trace data
Segment IOAM Data Export (Sec. 3.1)	Fix-size and controllable HbH header size	Need to enhance IOAM trace type data field spec.
Trace Export at SRv6 nodes (Sec. 3.2)	Can be done through configuration	Specific to SRv6; No better than PB & IOAM DEX in the worst case
IOAM Direct Export in HbH (Sec. 4)	Comply w/ IOAM DEX Spec; Fix-size and short HbH	Need export data correlation

Figure 5: Comparison of Different Approaches

6. Security Considerations

TBD.

7. Normative References

[I-D.ali-spring-ioam-srv6]
Ali, Z., Gandhi, R., Filsfils, C., Brockners, F., Nainar, N., Pignataro, C., Li, C., Chen, M., and G. Dawra, "Segment Routing Header encapsulation for In-situ OAM Data", draft-ali-spring-ioam-srv6-03 (work in progress), November 2020.

[I-D.ietf-6man-segment-routing-header]
Filsfils, C., Dukes, D., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-26 (work in progress), October 2019.

[I-D.ietf-6man-spring-srv6-oam]
Ali, Z., Filsfils, C., Matsushima, S., Voyer, D., and M. Chen, "Operations, Administration, and Maintenance (OAM) in Segment Routing Networks with IPv6 Data plane (SRv6)", draft-ietf-6man-spring-srv6-oam-08 (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.ioametal-ippm-6man-ioam-ipv6-deployment]
Bhandari, S., Brockners, F., Mizrahi, T., Kfir, A., Gafni, B., Spiegel, M., Krishnan, S., and M. Smith, "Deployment Considerations for In-situ OAM with IPv6 Options", draft-ioametal-ippm-6man-ioam-ipv6-deployment-03 (work in progress), March 2020.

[I-D.ioamteam-ippm-ioam-direct-export]
Song, H., Gafni, B., Zhou, T., Li, Z., Brockners, F., Bhandari, S., Sivakolundu, R., and T. Mizrahi, "In-situ OAM Direct Exporting", draft-ioamteam-ippm-ioam-direct-export-00 (work in progress), October 2019.

[I-D.song-6man-srv6-pbt]
Song, H., "Support Postcard-Based Telemetry for SRv6 OAM", draft-song-6man-srv6-pbt-01 (work in progress), October 2019.

[I-D.song-ippm-ioam-tunnel-mode]
Song, H., Li, Z., Zhou, T., and Z. Wang, "In-situ OAM Processing in Tunnels", draft-song-ippm-ioam-tunnel-mode-00 (work in progress), June 2018.

[I-D.song-ippm-postcard-based-telemetry]
Song, H., Zhou, T., Li, Z., Mirsky, G., Shin, J., and K. Lee, "Postcard-based On-Path Flow Data Telemetry using Packet Marking", draft-song-ippm-postcard-based-telemetry-08 (work in progress), October 2020.

[I-D.song-ippm-segment-ioam]
Song, H. and T. Zhou, "Control In-situ OAM Overhead with Segment IOAM", draft-song-ippm-segment-ioam-01 (work in progress), April 2018.

[I-D.song-spring-siam]
Song, H. and T. Pan, "SRv6 In-situ Active Measurement", draft-song-spring-siam-00 (work in progress), December 2020.

[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>.

[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>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

Authors' Addresses

Haoyu Song
Futurewei
USA

Email: haoyu.song@futurewei.com

Zhenbin Li
Huawei Technologies
China

Email: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
China

Email: pengshuping@huawei.com

James Guichard
Futurewei
USA

Email: james.n.guichard@futurewei.com