Stream: Independent Submission
RFC: 0000
Category: Informational
Published: January 2022
ISSN: 2070-1721
Authors: T. Mizrahi *Huawei* I.Y. Yerushalmi *Marvell* D.M. Melman *Marvell* R.B. Browne *Intel*

RFC 0000
Network Service Header (NSH) Fixed-Length Context Header Allocation: Timestamp

Abstract

The Network Service Header (NSH) specification defines two possible methods of including metadata (MD): MD Type 0x1 and MD Type 0x2. MD Type 0x1 uses a fixed-length Context Header. The allocation of this Context Header, i.e., its structure and semantics, has not been standardized. This memo presents an allocation for the fixed-length Context Header for the NSH -- the MD Type 0x1 Context Header -- which incorporates the packet's timestamp, a sequence number, and a source interface identifier.

Although the allocation presented in this document has not been standardized by the IETF, it has been implemented in silicon by several manufacturers and is published here to allow other interoperable implementations and to facilitate debugging if it is seen in the network.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not candidates for any level of Internet Standard; see 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/rfc0000.

Copyright Notice

Table of Contents

1. Introduction

The Network Service Header (NSH), defined in [RFC8300], is an encapsulation header that is used as the service encapsulation in the Service Function Chaining (SFC) architecture [RFC7665].

In order to share metadata along a service path, the NSH specification [RFC8300] supports two methods: a fixed-length Context Header (metadata (MD) Type 0x1) and a variable-length Context Header (MD Type 0x2). When using MD Type 0x1, the NSH includes 16 octets of Context Header fields.

The NSH specification [RFC8300] has not defined the semantics of the 16-octet Context Header, nor how it is used by NSH-aware service functions, Service Function Forwarders (SFFs), and proxies. Several Context Header formats are defined in [NSH-TLV]. Furthermore, some allocation schemes were proposed in the past to accommodate specific use cases, e.g., [NSH-DC-ALLOC], [NSH-BROADBAND-ALLOC], and [RFC8592].

This memo presents an allocation for the MD Type 0x1 Context Header, which incorporates the timestamp of the packet, a sequence number, and a source interface identifier. It is noted that other MD Type 0x1 allocations might be specified in the future. Although MD Type 0x1 allocations are currently not being standardized by the SFC Working Group, a consistent format (allocation) should be used in an SFC-enabled domain in order to allow interoperability.

In a nutshell, packets that enter the SFC-enabled domain are timestamped by a classifier [RFC7665]. Thus, the timestamp, sequence number, and source interface are incorporated in the NSH Context Header. As discussed in [RFC8300], if reclassification is used, it may result in an update to the NSH metadata. Specifically, when the Timestamp Context Header is used, a reclassifier may either leave it unchanged or update the three fields: Timestamp, Sequence Number, and Source Interface.

The Timestamp Context Header includes three fields that may be used for various purposes. The Timestamp field may be used for logging, troubleshooting, delay measurement, packet marking for performance monitoring, and timestamp-based policies. The source interface identifier indicates the interface through which the packet was received at the classifier. This identifier may specify a physical interface or a virtual interface. The sequence numbers can be used by Service Functions (SFs) to detect out-of-order delivery or duplicate transmissions. Note that out-of-order and duplicate packet detection is possible when packets are received by the same SF but is not necessarily possible when there are multiple instances of the same SF and multiple packets are spread across different instances of the SF. The sequence number is maintained on a per-source-interface basis.

This document presents the Timestamp Context Header but does not specify the functionality of the SFs that receive the Context Header. Although a few possible use cases are described in this document, the SF behavior and application are outside the scope of this document.

Key Performance Indicator (KPI) stamping [RFC8592] defines an NSH timestamping mechanism that uses the MD Type 0x2 format. The current memo defines a compact MD Type 0x1 Context Header that does not require the packet to be extended beyond the NSH header. Furthermore, the mechanisms described in [RFC8592] and this memo can be used in concert, as further discussed in Section 4.1.

Although the allocation presented in this document has not been standardized by the IETF, it has been implemented in silicon by several manufacturers and is published here to allow other interoperable implementations and to facilitate debugging if it is seen in the network.

2. Terminology

2.1. 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.

2.2. Abbreviations

The following abbreviations are used in this document:

KPI	Key Performance Indicator [RFC8592]
MD	Metadata [RFC8300]
NSH	Network Service Header [RFC8300]
SF	Service Function [RFC7665]
SFC	Service Function Chaining [RFC7665]

3. NSH Timestamp Context Header Allocation

This memo defines the following fixed-length Context Header allocation, as presented in Figure 1.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Sequence Number                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Source Interface                       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           Timestamp                           |
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: NSH Timestamp Allocation

The NSH Timestamp Allocation defined in this memo **MUST** include the following fields:

Sequence Number: A 32-bit sequence number. The sequence number is maintained on a per-source-interface basis. Sequence numbers can be used by SFs to detect out-of-order delivery or duplicate transmissions. The classifier increments the sequence number by 1 for each packet received through the source interface. This requires the classifier to maintain a per-source-interface counter. The sequence number is initialized to a random number on startup. After it reaches its maximal value (2^{32}-1), the sequence number wraps around back to zero.

Source Interface: A 32-bit source interface identifier that is assigned by the classifier. The combination of the source interface and the classifier identity is unique within the context of an SFC-enabled domain. Thus, in order for an SF to be able to use the source interface as a unique identifier, the identity of the classifier needs to be known for each packet. The source interface is unique in the context of the given classifier.

Timestamp: A 64-bit field that specifies the time at which the packet was received by the classifier. Two possible timestamp formats can be used for this field: the two 64-bit recommended formats specified in [RFC8877]. One of the formats is based on the timestamp format defined in [IEEE1588], and the other is based on the format defined in [RFC5905].

The NSH specification [RFC8300] does not specify the possible coexistence of multiple MD Type 0x1 Context Header formats in a single SFC-enabled domain. It is assumed that the Timestamp Context Header will be deployed in an SFC-enabled domain that uniquely uses this Context Header format. Thus, operators **SHOULD** ensure that either a consistent Context Header format is used in the SFC-enabled domain or there is a clear policy that allows SFs to know the Context Header format of each packet. Specifically, operators are expected to ensure the consistent use of a timestamp format across the whole SFC-enabled domain.

The two timestamp formats that can be used in the Timestamp field are as follows:

Truncated Timestamp Format [IEEE1588]: This format is specified in Section 4.3 of [RFC8877]. For the reader's convenience, this format is illustrated in Figure 2.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Seconds                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Nanoseconds                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Truncated Timestamp Format (IEEE 1588)

NTP 64-bit Timestamp Format [RFC5905]: This format is specified in Section 4.2.1 of [RFC8877]. For the reader's convenience, this format is illustrated in Figure 3.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Seconds                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Fraction                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: NTP 64-Bit Timestamp Format (RFC 5905)

Synchronization aspects of the timestamp format in the context of the NSH timestamp allocation are discussed in Section 5.

4. Timestamping Use Cases

4.1. Network Analytics

Per-packet timestamping enables coarse-grained monitoring of network delays along the Service Function Chain. Once a potential problem or bottleneck is detected (for example, when the delay exceeds a certain policy), a highly granular monitoring mechanism can be triggered (for example, using the hop-by-hop measurement data provided in [RFC8592] or [RFCYYY1]), allowing analysis and localization of the problem.

Timestamping is also useful for logging, troubleshooting, and flow analytics. It is often useful to maintain the timestamp of the first and last packet of the flow. Furthermore, traffic mirroring and sampling often require a timestamp to be attached to analyzed packets. Attaching the timestamp to the NSH provides an in-band common time reference that can be used for various network analytics applications.

4.2. Alternate Marking

A possible approach for passive performance monitoring is to use an Alternate-Marking Method [RFC8321]. This method requires data packets to carry a field that marks (colors) the traffic, and enables passive measurement of packet loss, delay, and delay variation. The value of this marking field is periodically toggled between two values.

When the timestamp is incorporated in the NSH, it can natively be used for Alternate Marking. For example, the least significant bit of the timestamp Seconds field can be used for this purpose, since the value of this bit is inherently toggled every second.

4.3. Consistent Updates

The timestamp can be used for making policy decisions, such as 'Perform action A if timestamp>=T_0'. This can be used for enforcing time-of-day policies or periodic policies in service functions. Furthermore, timestamp-based policies can be used for enforcing consistent network updates, as discussed in [DPT]. It should be noted that, as in the case of Alternate Marking, this use case alone does not require a full 64-bit timestamp but could be implemented with a significantly smaller number of bits.

5. Synchronization Considerations

Some of the applications that make use of the timestamp require the classifier and SFs to be synchronized to a common time reference -- for example, using the Network Time Protocol [RFC5905] or the Precision Time Protocol [IEEE1588]. Although it is not a requirement to use a clock synchronization mechanism, it is expected that, depending on the applications that use the timestamp, such synchronization mechanisms will be used in most deployments that use the timestamp allocation.

6. IANA Considerations

This document has no IANA actions.

7. Security Considerations

The security considerations for the NSH in general are discussed in [RFC8300]. The NSH is typically run within a confined trust domain. However, if a trust domain is not enough to provide the operator with protection against the timestamp threats as described below, then the operator **SHOULD** use transport-level protection between SFC processing nodes as described in [RFC8300].

The security considerations of in-band timestamping in the context of the NSH are discussed in [RFC8592]; this section is based on that discussion.

In-band timestamping, as defined in this document and [RFC8592], can be used as a means for network reconnaissance. By passively eavesdropping on timestamped traffic, an attacker can gather information about network delays and performance bottlenecks. A man-in-the-middle attacker can maliciously modify timestamps in order to attack applications that use the timestamp values, such as performance-monitoring applications.

Since the timestamping mechanism relies on an underlying time synchronization protocol, by attacking the time protocol an attack can potentially compromise the integrity of the NSH timestamp. A detailed discussion about the threats against time protocols and how to mitigate them is presented in [RFC7384].

8. References

8.1. 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>.

[RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

[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>.

[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>.

[RFC8877] Mizrahi, T., Fabini, J., and A. Morton, "Guidelines for Defining Packet Timestamps", RFC 8877, DOI 10.17487/RFC8877, September 2020, <https://www.rfc-editor.org/info/rfc8877>.

8.2. Informative References

[DPT] Mizrahi, T. and Y. Moses, "The Case for Data Plane Timestamping in SDN", IEEE INFOCOM Workshop on Software-Driven Flexible and Agile Networking (SWFAN), 2016, <https://arxiv.org/pdf/1602.03342.pdf>.

[IEEE1588] IEEE, "IEEE 1588-2008 - IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", <https://standards.ieee.org/standard/1588-2008.html>.

[NSH-BROADBAND-ALLOC] Napper, J., Kumar, S., Muley, P., Hendericks, W., and M. Boucadair, "NSH Context Header Allocation for Broadband", Work in Progress, Internet-Draft, draft-ietf-sfc-nsh-broadband-allocation-01, 19 June 2018, <https://datatracker.ietf.org/doc/html/draft-ietf-sfc-nsh-broadband-allocation-01>.

[NSH-DC-ALLOC] Guichard, J., Ed., Smith, M., Kumar, S., Majee, S., and T. Mizrahi, "Network Service Header (NSH) MD Type 1: Context Header Allocation (Data Center)", Work in Progress, Internet-Draft, draft-ietf-sfc-nsh-dc-allocation-02, 25 September 2018, <https://datatracker.ietf.org/doc/html/draft-ietf-sfc-nsh-dc-allocation-02>.

[NSH-TLV] Wei, Y., Ed., Elzur, U., Majee, S., Pignataro, C., and D. E. Eastlake, "Network Service Header Metadata Type 2 Variable-Length Context Headers", Work in Progress, Internet-Draft, draft-ietf-sfc-nsh-tlv-10, 3 December 2021, <https://datatracker.ietf.org/doc/html/draft-ietf-sfc-nsh-tlv-10>.

[RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <https://www.rfc-editor.org/info/rfc5905>.

[RFC7384] Mizrahi, T., "Security Requirements of Time Protocols in Packet Switched Networks", RFC 7384, DOI 10.17487/RFC7384, October 2014, <https://www.rfc-editor.org/info/rfc7384>.

[RFC8321] Fioccola, G., Ed., Capello, A., Cociglio, M., Castaldelli, L., Chen, M., Zheng, L., Mirsky, G., and T. Mizrahi, "Alternate-Marking Method for Passive and Hybrid Performance Monitoring", RFC 8321, DOI 10.17487/RFC8321, January 2018, <https://www.rfc-editor.org/info/rfc8321>.

[RFC8592] Browne, R., Chilikin, A., and T. Mizrahi, "Key Performance Indicator (KPI) Stamping for the Network Service Header (NSH)", RFC 8592, DOI 10.17487/RFC8592, May 2019, <https://www.rfc-editor.org/info/rfc8592>.

[RFCYYY1] Brockners, F., Ed., Bhandari, S., Ed., and T. Mizrahi, Ed., "Data Fields for In-situ OAM", RFC YYY1, DOI 10.17487/RFCYYY1, January 2022, <https://www.rfc-editor.org/info/rfcYYY1>.

Acknowledgments

The authors thank Mohamed Boucadair and Greg Mirsky for their thorough reviews and detailed comments.

Authors' Addresses

Tal Mizrahi
Huawei
Israel
Email: tal.mizrahi.phd@gmail.com

Ilan Yerushalmi
Marvell
6 Hamada
Yokneam 2066721
Israel
Email: yilan@marvell.com

David Melman
Marvell
6 Hamada
Yokneam 2066721
Israel
Email: davidme@marvell.com

Rory Browne
Intel
Dromore House
Shannon
Co. Clare
Ireland
Email: rory.browne@intel.com