

---

Stream: Internet Engineering Task Force (IETF)  
RFC: [9263](#)  
Category: Standards Track  
Published: August 2022  
ISSN: 2070-1721  
Authors: Y. Wei, Ed. U. Elzur S. Majee C. Pignataro  
ZTE Corporation Intel Individual Contributor Cisco  
D. Eastlake 3rd  
Futurewei Technologies

# RFC 9263

## Network Service Header (NSH) Metadata Type 2 Variable-Length Context Headers

---

### Abstract

Service Function Chaining (SFC) uses the Network Service Header (NSH) (RFC 8300) to steer and provide context metadata (MD) with each packet. Such metadata can be of various types, including MD Type 2, consisting of Variable-Length Context Headers. This document specifies several such Context Headers that can be used within a Service Function Path (SFP).

### 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/rfc9263>.

### Copyright Notice

Copyright (c) 2022 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

1. Introduction
2. Conventions Used in This Document
  - 2.1. Terminology
  - 2.2. Requirements Language
3. NSH MD Type 2 Format
4. NSH MD Type 2 Context Headers
  - 4.1. Forwarding Context
  - 4.2. Tenant ID
  - 4.3. Ingress Network Node Information
  - 4.4. Ingress Network Source Interface
  - 4.5. Flow ID
  - 4.6. Source and/or Destination Groups
  - 4.7. Policy ID
5. Security Considerations
  - 5.1. Forwarding Context
  - 5.2. Tenant ID
  - 5.3. Ingress Network Node Information
  - 5.4. Ingress Node Source Interface
  - 5.5. Flow ID
  - 5.6. Source and/or Destination Groups
  - 5.7. Policy ID
6. IANA Considerations
  - 6.1. MD Type 2 Context Types
  - 6.2. Forwarding Context Types
  - 6.3. Flow ID Context Types

## 7. References

### 7.1. Normative References

### 7.2. Informative References

## Acknowledgments

## Authors' Addresses

# 1. Introduction

The Network Service Header (NSH) [[RFC8300](#)] is the Service Function Chaining (SFC) encapsulation that supports the SFC architecture [[RFC7665](#)]. As such, the NSH provides the following key elements:

1. Service Function Path (SFP) identification
2. indication of location within an SFP
3. optional, per-packet metadata (fixed-length or variable-length)

[[RFC8300](#)] further defines two metadata formats (MD Types): 1 and 2. MD Type 1 defines the fixed-length, 16-octet metadata, whereas MD Type 2 defines a variable-length context format for metadata. This document defines several common metadata Context Headers for use within NSH MD Type 2. These supplement the Subscriber Identifier and Performance Policy MD Type 2 metadata Context Headers specified in [[RFC8979](#)].

This document does not address metadata usage, updating/chaining of metadata, or other SFP functions. Those topics are described in [[RFC8300](#)].

# 2. Conventions Used in This Document

## 2.1. Terminology

This document uses the terminology defined in the SFC architecture [[RFC7665](#)] and the NSH [[RFC8300](#)].

## 2.2. 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. NSH MD Type 2 Format

An NSH is composed of a 4-octet Base Header, a 4-octet Service Path Header, and optional Context Headers. The Base Header identifies the MD Type in use:

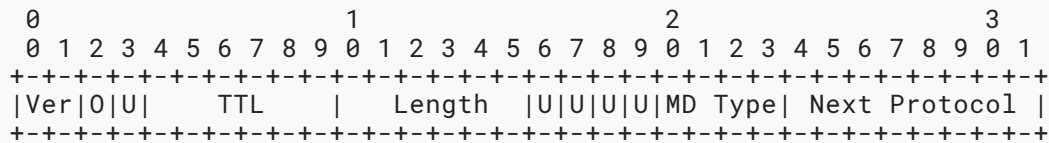


Figure 1: NSH Base Header

Please refer to the NSH [RFC8300] for a detailed header description.

When the Base Header specifies MD Type = 0x2, zero or more Variable-Length Context Headers **MAY** be added, immediately following the Service Path Header. Figure 2 below depicts the format of the Context Header as defined in Section 2.5.1 of [RFC8300].

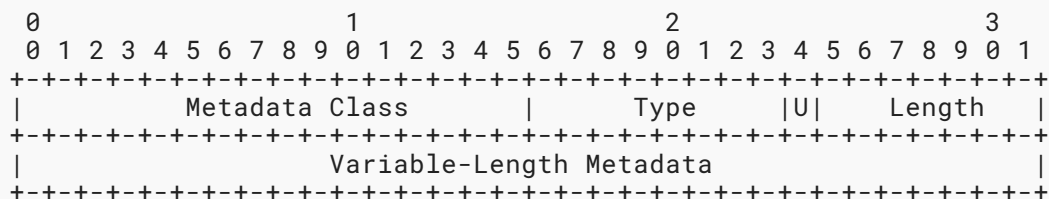


Figure 2: NSH Variable-Length Context Headers

### 4. NSH MD Type 2 Context Headers

[RFC8300] specifies Metadata Class 0x0000 as IETF Base NSH MD Class. In this document, metadata types are defined for the IETF Base NSH MD Class. The Context Headers specified in the subsections below are as follows:

1. Forwarding Context
2. Tenant ID
3. Ingress Network Node Information
4. Ingress Node Source Interface
5. Flow ID
6. Source and/or Destination Groups
7. Policy ID

## 4.1. Forwarding Context

This metadata context carries a network forwarding context, used for segregation and forwarding scope. Forwarding context can take several forms depending on the network environment, for example, Virtual eXtensible Local Area Network (VXLAN) / Generic Protocol Extension for VXLAN (VXLAN-GPE) Virtual Network Identifier (VNID), VPN Routing and Forwarding (VRF) identification, or VLAN.

```

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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Metadata Class = 0x0000 | Type = 0x04 |U| Length = 4 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|CT=0x0 | Reserved | VLAN ID |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Figure 3: VLAN Forwarding Context

```

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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Metadata Class = 0x0000 | Type = 0x04 |U| Length = 4 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|CT=0x1 |Resv | Service VLAN ID | Customer VLAN ID |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Figure 4: QinQ Forwarding Context

```

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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Metadata Class = 0x0000 | Type = 0x04 |U| Length = 4 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|CT=0x2 | Reserved | MPLS VPN Label |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Figure 5: MPLS VPN Forwarding Context

```

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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Metadata Class = 0x0000 | Type = 0x04 |U| Length = 4 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|CT=0x3 | Resv | Virtual Network Identifier |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Figure 6: VNI Forwarding Context

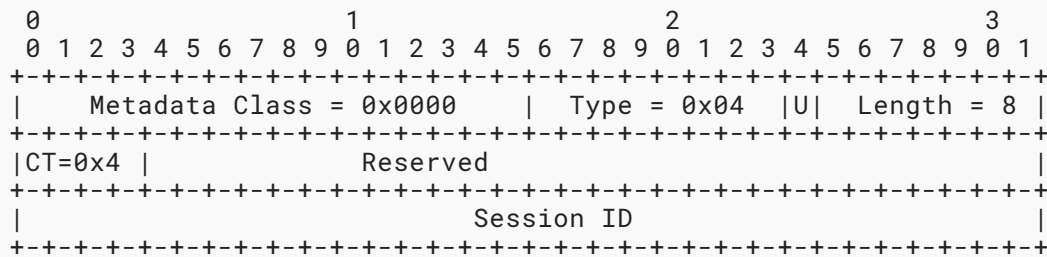


Figure 7: Session ID Forwarding Context

The fields are described as follows:

**Context Type (CT):** This 4-bit field that defines the interpretation of the Forwarding Context field. Please see the IANA considerations in [Section 6.2](#). This document defines these CT values:

- 0x0: 12-bit VLAN identifier [[IEEE.802.1Q\\_2018](#)]. See [Figure 3](#).
- 0x1: 24-bit double tagging identifiers. A service VLAN tag followed by a customer VLAN tag [[IEEE.802.1Q\\_2018](#)]. The two VLAN IDs are concatenated and appear in the same order that they appeared in the payload. See [Figure 4](#).
- 0x2: 20-bit MPLS VPN label [[RFC3032](#)] [[RFC4364](#)]. See [Figure 5](#).
- 0x3: 24-bit virtual network identifier (VNI) [[RFC8926](#)]. See [Figure 6](#).
- 0x4: 32-bit Session ID [[RFC3931](#)]. This is called Key in GRE [[RFC2890](#)]. See [Figure 7](#).

**Reserved (Resv):** These bits in the context fields **MUST** be sent as zero and ignored on receipt.

## 4.2. Tenant ID

Tenant identification is often used for segregation within a multi-tenant environment. Orchestration system-generated Tenant IDs are an example of such data. This Context Header carries the value of the Tenant ID. Virtual Tenant Network (VTN) [[OpenDaylight-VTN](#)] is an application that provides multi-tenant virtual networks on a Software-Defined Networking (SDN) controller.

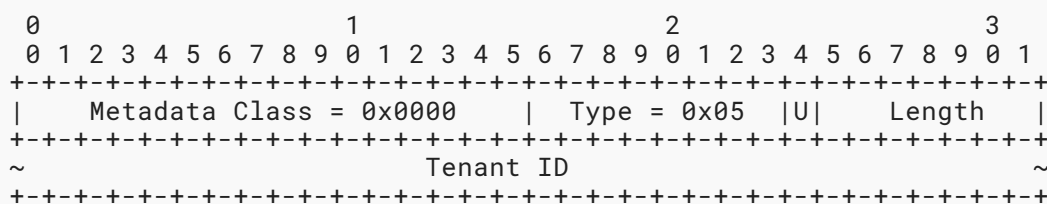


Figure 8: Tenant ID List

The fields are described as follows:

Length: Indicates the length of the Tenant ID in octets (see [Section 2.5.1](#) of [RFC8300]).

Tenant ID: Represents an opaque value pointing to orchestration system-generated Tenant ID. The structure and semantics of this field are specific to the operator's deployment across its operational domain and are specified and assigned by an orchestration function. The specifics of that orchestration-based assignment are outside the scope of this document.

### 4.3. Ingress Network Node Information

This Context Header carries a Node ID of the network node at which the packet entered the SFC-enabled domain. This node will necessarily be a classifier [RFC7665]. In cases where the Service Path Identifier (SPI) identifies the ingress node, this Context Header is superfluous.

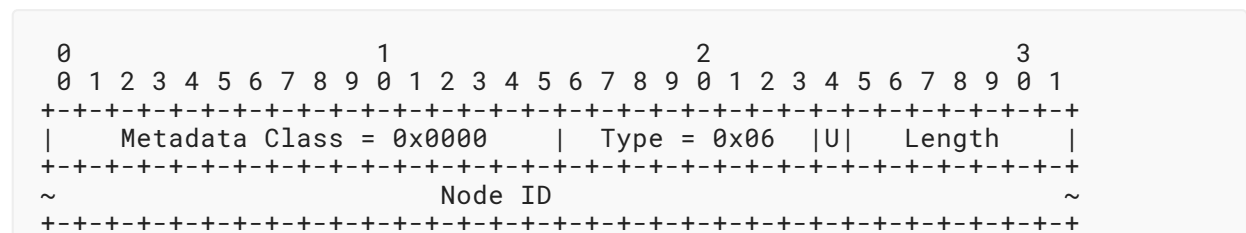


Figure 9: Ingress Network Node ID

The fields are described as follows:

Length: Indicates the length of the Node ID in octets (see [Section 2.5.1](#) of [RFC8300]).

Node ID: Represents an opaque value of the ingress network Node ID. The structure and semantics of this field are deployment specific. For example, Node ID may be a 4-octet IPv4 address Node ID, a 16-octet IPv6 address Node ID, a 6-octet MAC address, an 8-octet MAC address (64-bit Extended Unique Identifier (EUI-64)), etc.

### 4.4. Ingress Network Source Interface

This context identifies the ingress interface of the ingress network node. The l2vlan (135), l3ipvlan (136), ipForward (142), and mpls (166) in [IANAifType] are examples of source interfaces.

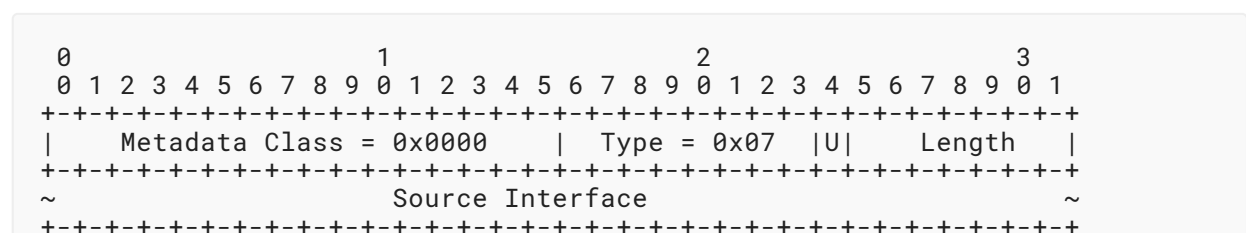


Figure 10: Ingress Network Source Interface

The fields are described as follows:

Length: Indicates the length of the Source Interface in octets (see Section 2.5.1 of [RFC8300]).

Source Interface: Represents an opaque value of the identifier of the ingress interface of the ingress network node.

### 4.5. Flow ID

Flow ID provides a field in NSH MD Type 2 to label packets belonging to the same flow. For example, [RFC8200] defines IPv6 Flow Label as Flow ID. Another example of Flow ID is how [RFC6790] defines an entropy label that is generated based on flow information in the MPLS network. Absence of this field or a value of zero denotes that packets have not been labeled with a Flow ID.

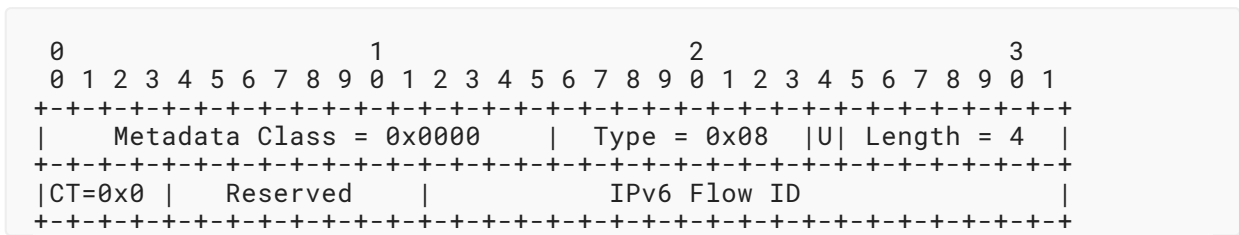


Figure 11: IPv6 Flow ID

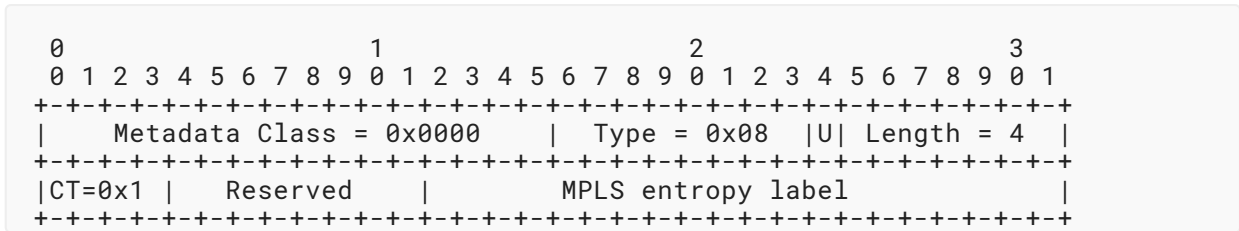


Figure 12: MPLS Entropy Label

The fields are described as follows:

Length: Indicates the length of the Flow ID in octets (see Section 2.5.1 of [RFC8300]). For example, the IPv6 Flow Label in [RFC8200] is 20 bits long. An entropy label in the MPLS network in [RFC6790] is also 20 bits long.

Context Type (CT): This 4-bit field that defines the interpretation of the Flow ID field. Please see the IANA considerations in Section 6.3. This document defines these CT values:

0x0: 20-bit IPv6 Flow Label in [RFC8200]. See Figure 11.

0x1: 20-bit entropy label in the MPLS network in [RFC6790]. See Figure 12.

Reserved: These bits in the context fields **MUST** be sent as zero and ignored on receipt.



## 4.6. Source and/or Destination Groups

Intent-based systems can use this data to express the logical grouping of source and/or destination objects. [OpenStack] and [OpenDaylight] provide examples of such a system. Each is expressed as a 32-bit opaque object.

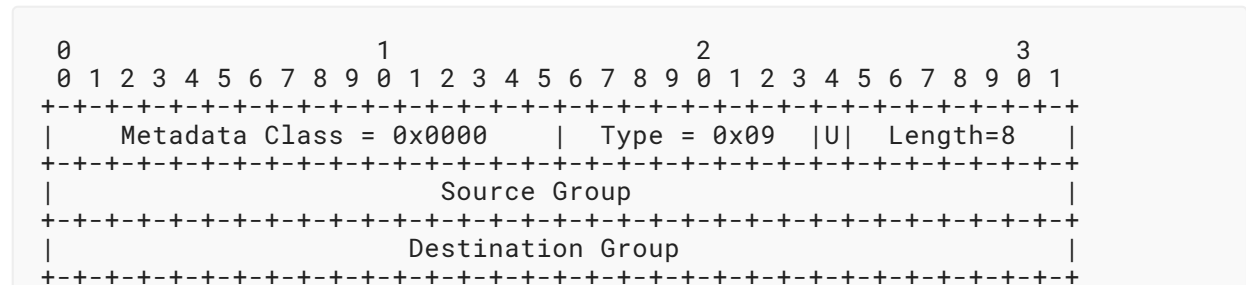


Figure 13: Source/Destination Groups

If there is no group information specified for the Source Group or Destination Group field, the field **MUST** be sent as zero and ignored on receipt.

## 4.7. Policy ID

Traffic handling policies are often referred to by a system-generated identifier, which is then used by the devices to look up the policy's content locally. For example, this identifier could be an index to an array, a lookup key, or a database ID. The identifier allows enforcement agents or services to look up the content of their part of the policy.

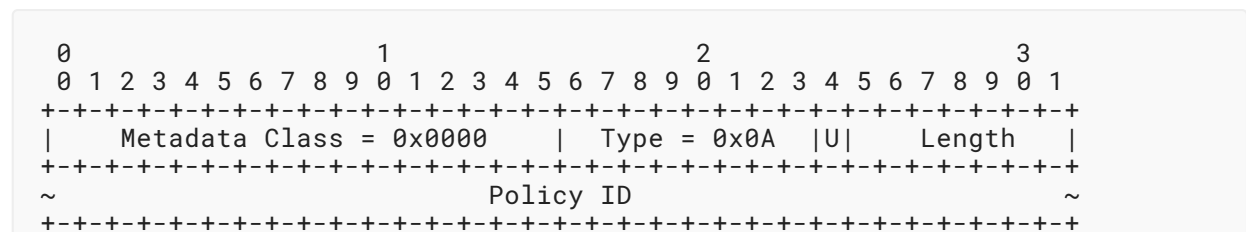


Figure 14: Policy ID

The fields are described as follows:

**Length:** Indicates the length of the Policy ID in octets (see [Section 2.5.1](#) of [RFC8300]).

**Policy ID:** Represents an opaque value of the Policy ID.

This Policy ID is a general Policy ID, essentially a key to allow Service Functions (SFs) to know which policies to apply to packets. Those policies generally will not have much to do with performance but rather with what specific treatment to apply. It may, for example, select a URL filter data set for a URL filter or select a video transcoding policy in a transcoding SF. The

Performance Policy ID in [RFC8979] is described there as having very specific use and, for example, says that fully controlled SFPs would not use it. The Policy ID in this document is for cases not covered by [RFC8979].

## 5. Security Considerations

A misbehaving node from within the SFC-enabled domain may alter the content of the Context Headers, which may lead to service disruption. Such an attack is not unique to the Context Headers defined in this document. Measures discussed in Section 8 of [RFC8300] describes the general security considerations for protecting the NSH. [RFC9145] specifies methods of protecting the integrity of the NSH metadata. If the NSH includes the Message Authentication Code (MAC) and Encrypted Metadata Context Header [RFC9145], the authentication of the packet **MUST** be verified before using any data. If the verification fails, the receiver **MUST** stop processing the Variable-Length Context Headers and notify an operator.

The security and privacy considerations for the 7 types of Context Headers specified above are discussed below. Since NSH-ignorant SFs will never see the NSH, then even if they are malign, they cannot compromise security or privacy based on the NSH or any of these Context Headers; however, they could cause compromise based on the rest of the packet. To the extent that any of these headers are included when they would be unneeded or have no effect, they provide a covert channel for the entity adding the Context Header to communicate a limited amount of arbitrary information to downstream entities within the SFC-enabled domain.

### 5.1. Forwarding Context

All of the Forwarding Context variants specified in this document (those with CT values between 0 and 4) merely repeat a field that is available in the packet encapsulated by the NSH. These variants repeat that field in the NSH for convenience. Thus, there are no special security or privacy considerations in these cases. Any future new values of CT for the Forwarding Context must specify the security and privacy considerations for those extensions.

### 5.2. Tenant ID

The Tenant ID indicates the tenant to which traffic belongs and might be used to tie together and correlate packets for a tenant that some monitoring function could not otherwise group, especially if other possible identifiers were being randomized. As such, it may reduce security by facilitating traffic analysis but only within the SFC-enabled domain where this Context Header is present in packets.

### 5.3. Ingress Network Node Information

The SFC-enabled domain manager normally operates the initial ingress/classifier node and is thus potentially aware of the information provided by this Context Header. Furthermore, in many cases, the SPI that will be present in the NSH identifies or closely constrains the ingress node. Also, in most cases, it is anticipated that many entities will be sending packets into an SFC-enabled domain through the same ingress node. Thus, under most circumstances, this Context Header is expected to weaken security and privacy to only a minor extent and only within the SFC-enabled domain.

### 5.4. Ingress Node Source Interface

This Context Header is likely to be meaningless unless the Ingress Network Node Information Context Header is also present. When that node information header is present, this source interface header provides a more fine-grained view of the source by identifying not just the initial ingress/classifier node but also the port of that node on which the data arrived. Thus, it is more likely to identify a specific source entity or at least to more tightly constrain the set of possible source entities than just the node information header. As a result, inclusion of this Context Header with the node information Context Header is potentially a greater threat to security and privacy than the node information header alone, but this threat is still constrained to the SFC-enabled domain.

### 5.5. Flow ID

The variations of this Context Header specified in this document simply repeat fields already available in the packet and thus have no special security or privacy considerations. Any future new values of CT for the Flow ID must specify the security and privacy considerations for those extensions.

### 5.6. Source and/or Destination Groups

This Context Header provides additional information that might help identify the source and/or destination of packets. Depending on the granularity of the groups, it could either (1) distinguish packets as part of flows from and/or to objects where those flows could not otherwise be easily distinguished but appear to be part of one or fewer flows or (2) group packet flows that are from and/or to an object where those flows could not otherwise be easily grouped for analysis or another purpose. Thus, the presence of this Context Header with non-zero source and/or destination groups can, within the SFC-enabled domain, erode security and privacy to an extent that depends on the details of the grouping.

### 5.7. Policy ID

This Context Header carries an identifier that nodes in the SFC-enabled domain can use to look up policy to potentially influence their actions with regard to the packet carrying this header. If there are no such decisions regarding their actions, then the header should not be included. If there are such decisions, the information on which they are to be based needs to be included

somewhere in the packet. There is no reason for inclusion in this Context Header to have any security or privacy considerations that would not apply to any other plaintext way of including such information. It may provide additional information to help identify a flow of data for analysis.

## 6. IANA Considerations

### 6.1. MD Type 2 Context Types

IANA has assigned the following types ([Table 1](#)) from the "NSH IETF-Assigned Optional Variable-Length Metadata Types" registry available at [[IANA-NSH-MD2](#)].

| Value | Description                      | Reference |
|-------|----------------------------------|-----------|
| 0x04  | Forwarding Context               | RFC 9263  |
| 0x05  | Tenant ID                        | RFC 9263  |
| 0x06  | Ingress Network Node ID          | RFC 9263  |
| 0x07  | Ingress Network Interface        | RFC 9263  |
| 0x08  | Flow ID                          | RFC 9263  |
| 0x09  | Source and/or Destination Groups | RFC 9263  |
| 0x0A  | Policy ID                        | RFC 9263  |

*Table 1: Type Values*

### 6.2. Forwarding Context Types

IANA has created a new subregistry for "Forwarding Context Types" at [[IANA-NSH-MD2](#)] as follows.

The registration policy is IETF Review.

| Value | Description                             | Reference |
|-------|---|-----------|
| 0x0   | 12-bit VLAN identifier                  | RFC 9263  |
| 0x1   | 24-bit double tagging identifiers       | RFC 9263  |
| 0x2   | 20-bit MPLS VPN label                   | RFC 9263  |
| 0x3   | 24-bit virtual network identifier (VNI) | RFC 9263  |
| 0x4   | 32-bit Session ID                       | RFC 9263  |

| Value   | Description | Reference |
|---------|-------------|-----------|
| 0x5-0xE | Unassigned  |           |
| 0xF     | Reserved    | RFC 9263  |

Table 2: Forwarding Context Types

### 6.3. Flow ID Context Types

IANA has created a new subregistry for "Flow ID Context Types" at [IANA-NSH-MD2] as follows.

The registration policy is IETF Review.

| Value   | Description                              | Reference |
|---------|--|-----------|
| 0x0     | 20-bit IPv6 Flow Label                   | RFC 9263  |
| 0x1     | 20-bit entropy label in the MPLS network | RFC 9263  |
| 0x2-0xE | Unassigned                               |           |
| 0xF     | Reserved                                 | RFC 9263  |

Table 3: Flow ID Context Types

## 7. References

### 7.1. Normative References

[IANA-NSH-MD2] IANA, "Network Service Header (NSH) Parameters", <<https://www.iana.org/assignments/nsh>>.

[IEEE.802.1Q\_2018] IEEE, "IEEE Standard for Local and Metropolitan Area Network -- Bridges and Bridged Networks", July 2018, <<https://ieeexplore.ieee.org/document/8403927>>.

[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>>.

[RFC3931] Lau, J., Ed., Townsley, M., Ed., and I. Goyret, Ed., "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, DOI 10.17487/RFC3931, March 2005, <<https://www.rfc-editor.org/info/rfc3931>>.

[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>>.
- [RFC9145] Boucadair, M., Reddy, K. T., and D. Wing, "Integrity Protection for the Network Service Header (NSH) and Encryption of Sensitive Context Headers", RFC 9145, DOI 10.17487/RFC9145, December 2021, <<https://www.rfc-editor.org/info/rfc9145>>.

## 7.2. Informative References

- [IANAifType] IANA, "IANAifType-MIB DEFINITIONS", 2021, <<https://www.iana.org/assignments/ianaiftype-mib/ianaiftype-mib>>.
- [OpenDaylight] OpenDaylight, "Group Based Policy User Guide", 2021, <<https://docs.opendaylight.org/en/stable-fluorine/user-guide/group-based-policy-user-guide.html?highlight=group%20policy#>>.
- [OpenDaylight-VTN] OpenDaylight, "OpenDaylight VTN", 2021, <[https://nexus.opendaylight.org/content/sites/site/org.opendaylight.docs/master/userguide/manuals/userguide/bk-user-guide/content/\\_vtn.html](https://nexus.opendaylight.org/content/sites/site/org.opendaylight.docs/master/userguide/manuals/userguide/bk-user-guide/content/_vtn.html)>.
- [OpenStack] OpenStack, "GroupBasedPolicy", 2021, <<https://wiki.openstack.org/wiki/GroupBasedPolicy>>.
- [RFC2890] Dommety, G., "Key and Sequence Number Extensions to GRE", RFC 2890, DOI 10.17487/RFC2890, September 2000, <<https://www.rfc-editor.org/info/rfc2890>>.
- [RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <<https://www.rfc-editor.org/info/rfc3032>>.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <<https://www.rfc-editor.org/info/rfc4364>>.
- [RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <<https://www.rfc-editor.org/info/rfc6790>>.
- [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>>.
- [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>>.
- [RFC8926] Gross, J., Ed., Ganga, I., Ed., and T. Sridhar, Ed., "Geneve: Generic Network Virtualization Encapsulation", RFC 8926, DOI 10.17487/RFC8926, November 2020, <<https://www.rfc-editor.org/info/rfc8926>>.

**[RFC8979]** Sarikaya, B., von Hugo, D., and M. Boucadair, "Subscriber and Performance Policy Identifier Context Headers in the Network Service Header (NSH)", RFC 8979, DOI 10.17487/RFC8979, February 2021, <<https://www.rfc-editor.org/info/rfc8979>>.

## Acknowledgments

The authors would like to thank Paul Quinn, Behcet Sarikaya, Dirk von Hugo, Mohamed Boucadair, Gregory Mirsky, and Joel Halpern for providing invaluable concepts and content for this document.

## Authors' Addresses

### Yuehua Wei (EDITOR)

ZTE Corporation  
No.50, Software Avenue  
Nanjing  
210012  
China  
Email: [wei.yuehua@zte.com.cn](mailto:wei.yuehua@zte.com.cn)

### Uri Elzur

Intel  
Email: [uri.elzur@intel.com](mailto:uri.elzur@intel.com)

### Sumandra Majee

Individual Contributor  
Email: [Sum.majee@gmail.com](mailto:Sum.majee@gmail.com)

### Carlos Pignataro

Cisco  
Email: [cpignata@cisco.com](mailto:cpignata@cisco.com)

### Donald E. Eastlake, 3rd

Futurewei Technologies  
2386 Panoramic Circle  
Apopka, FL 32703  
United States of America  
Phone: +1-508-333-2270  
Email: [d3e3e3@gmail.com](mailto:d3e3e3@gmail.com)