Network Working Group
Internet-Draft
Intended status: Informational
Expires: March 21, 2015

N.P. Bouthors
Qosmos

September 19, 2014

Metadata transport in SFC
draft-bouthors-sfc-md-00.txt

Abstract

This draft describes the transport of metadata in Service Function Chains.

It precises how metadata MAY be shared reliably by network devices and/or network services via Metadata Messages that can be exchanged offline or inline.

It proposes IPFIX as a representation mechanism for SFC Metadata and SCTP as an optional transport mechanism to enable reliable transmission of Metadata in SFC

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 March 21, 2015.

Copyright Notice

Table of Contents

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. Introduction

This draft reviews the notions of metadata transport and representation in the context of Service Function Chaining.

It builds upon the work done on Metadata in [RIJSMAN] and use the concepts and terminology defined in [SFC_ARCH]

This document is illustrated with the example of the Network Service Header [QUINN_NSH] as well as the Service Chain Header in [ZHANG_SCH] and [NIU_SFC] as they all define implictily a metadata transport service in their corresponding proposals.

As described in [SFC_ARCH] a Service Function Chain contains a set of abstract Service Functions which may be applied to a set of packets of flows. In practice, Service Functions are processing and forwarding data packets along a Rendered Service Path (RSP).

As per [RIJSMAN], in the context of Service Function Chaining, metadata may be shared between Service Functions as a way to provide contextual information about the data packets which traverse the Rendered Service Path.

[SFC_ARCH] states that metadata can be thought of as providing and sharing the result of classification. But we can see other sources for metadata, such as the SF or SFF along the Rendered Service Path.

For [RIJSMAN] Metadata can be used for:

1. Sharing Contextual information which is not locally available,

2. Avoiding repeated execution of expensive operations ,

3. Enabling support Fine-grained policies over a reduced number of chains,

4. And ensuring a single format of classification throughout the whole chain.

Metadata can be exchanged directly in-band, indirectly out-of-band, or via some hybrid mode possibly with some support from the SFC header. These different models are described in in [RIJSMAN].

Section 3 will review the representation requirements for Metadata and see that they can be addressed based on existing standards.

Section 4 will review the Metadata transport function of SFC from a reliability point of view.

2.1. Definition of Terms

Metadata: In the context of Service Function Chaining, metadata provides contextual information about the data packets which traverse a Service Function Chain.

Dataplane Metadata: These are information elements included in SFC header. They can represent values for certain contextual attributes, or keys to be used by Service Functions to access the expected contextual information via some offline collection procedures.

Mandatory Dataplane Metadata: These are contextual information elements always present in SFC headers. The Service Path field in the Base Header of NSH can be considered as such. It is shared by all packets in a chain; it remains constant and acts as an identifier for the chain instance.

Optional Dataplane Metadata: These are contextual information elements included in some SFC headers.

Offline Metadata: Metadata transported out of band but tied to a chain, a flow in a chain or to a specific packet.

Chain: For convenience in the text we will sometime refer to the Rendered Service Path as the chain.

2.2. Problem Space

Two aspects of Metadata in SFC need to be addressed thoroughly because of their potential impact on target use cases. First how Metadata can be represented, Second how it can reliably transported between SFF and delivered to the target SF where needed.

2.2.1. Metadata representation

Metadata can be extremely varied in term of usage and content. It can represent the result of Deep Packet Inspection (DPI) performed on the traffic for example by the Classifier Service Function at the ingress of the Rendered Service Path. It can contain information collected about the end user such as a policy identifier, a category or even represent an event related to the end-user session.

Service Function and Service Forwarding Function can be as well source of Metadata

This information can be used by Service Functions down the chain, as well as by monitoring entities responsible to track usage for example in the Network Function Virtualization environment to feed a VNF manager or an Orchestrator.

Metadata being information shared by many network entities needs some means to be represented it in all of its dimensions. To this effect, relying the IETF IPFIX standard is proposed in Section 3

2.2.2. Metadata transport service

As expected from service chain proposals, NSH, SCH and NIU proposals define some means to carry metadata between Service Functions in a Chain. They can be classified as follow:

Dataplane Metadata: They are defined in the Service Function Chaining Problem Statement document. They are not considered to be part of the forwarding information of the SFC header. However they are expected to carry the result of antecedent classification, allowing Service Functions to take local policy decisions based on their values.

As such, they could also be interpreted directly by Service Function Forwarder to steer traffic to various Service Functions.

This is indeed confirmed by the SFC architecture document as a not in SFP forwarding function of the SFF.

Offline Metadata: Beyond Dataplane Metadata, Offline Metadata can be shared between Service Functions in a chain, using out of bound, congruent or not, or hybrid models described in [RIJSMAN]

The hybrid model for example, defined in [RIJSMAN], utilizes the SFC header to transport some key values for correlation purposes. These Correlation Ids can be used by the Service Functions to recover the full associated contextual information.

Metadata, directly or indirectly, are transported hop by hop along a chain, in association to end-user traffic, the data payload of the SFC packets.

How these metadata are transported over a chain matters. Passing metadata directly or indirectly along packets is a service that must be analyzed from a reliability point of view.

Reliability requirement may vary depending on the nature of the metadata transported. Past experience for example in Mobile Network and data center with AAA Radius have shown that contextual information replication to various service function was indeed sensitive to packet loss events, and that ad hoc solutions had to be implemented to detect them.

A reliable transport service for Metadata in SFC is expected. To this effect, an implementation is suggested in Section 4

3. Metadata representation

Metadata definition is that it provides contextual information about the data packets which traverse a Service Function Chain. This must be understood broadly.

Metadata can contain the result of traffic classification by Deep Packet Inspection (DPI). For example as an Application Id information which is tied to a traffic flow. There could be multiple flows with different application ids, in a chain.

Metadata can also contain the result of DPI data extraction, such as identify requested URL in HTTP. Such information can be passed to certain SF down the chain such as a URL filtering function.

Metadata can contain some punctual event information collected at the Ingress point of the chain and expected to be passed to all elements in the chain. Here this information may be triggered externally and generated only once, and be related to the tenant or the subscriber.

Metadata representation involves the definition of a set of information elements types and the encoding rules for their values.

Metadata representation can sometimes be performed by a single individual field with associated type and format. It is not always the case.

Metadata may need multiple fields transported together to represented their values.

Some addition fields may be required to describe the scope of the metadata itself. This can be any information element defining the context of the associated metadata value. For example a throughput metadata field can have a port number and a switch address as its Scope information.

The following Section 3.1 explores these two axes: encoding and scope.

The Section 3.1.3 proposes IPFIX as a preferred means to represent Metadata in Service Chain messages for Out-of-band, Congruent or not; Metadata sharing.

3.1. Metadata Representation Requirements

Mandatory Dataplane Metadata is always part of the SFC header, it is thus reasonable to consider that its representation scheme will be implicit: based on what the SFC protocol will dictate, their position in the SFC header is sufficient for the receiving end to infer their type and encoding scheme. For example, Context Header Fields in NSH are 32 bit fields.

However, it will not be the case for all metadata transported. Optional and Offline metadata, including congruent out-of-band metadata still need to be represented explicitly. This section

addresses their specific case.

3.1.1. Metadata Encoding requirements

These requirements are applicable to out-of-band metadata (Congruent or not). It could be applicable with SCH on optional in-line metadata fields.

For interoperability purposes, metadata encoding MUST allow the receiving entity to identify the type and value of the information received as metadata

Metadata encoding MUST allow for encoding techniques supporting well known types and fields as well as proprietary extensions.

A receiving entity MUST be able to identify when incoming metadata type is unknown and MUST have a defined default action to handle it.

A piece of information may need multiple attributes to be described. For example a tenant id and an IP address can be used to identify a server in a data center uniquely. Metadata encoding MUST support such structured fields.

These groups of information have to be exchanged collectively, as part of a single logical message. In this case, a sending entity MUST specify that it is sending a set of metadata in a message.

This set of transported metadata elements MUST be specified under the form of a metadata template document uniquely defined for the chain.

A receiving entity MUST be able to detect if an incoming messages contains its expected set of metadata elements.

3.1.2. Metadata Scope requirement

A piece of information may have to be qualified by some attributes identifying its particular scope. For example a throughput scope may have to describe where and when it was measured.

Scope can apply to some individual metadata elements or to a set of metadata elements. How a scope applies to a set of transported metadata elements should be defined by a specification of the transport set under the form of a metadata template document uniquely identified for the chain.

3.1.3. IPFIX Metadata representation

So far, simple Type Length Value encoding has been proposed to transport metadata. It is not clear how structured types are supported, and no distinction is done between the metadata value and the scoping value. For example, although the SCH proposal provides an optional 24-bit Organizational Unique Identifier, there is no

namespace mechanism allowing to separate type definition spaces per Tenants or per chain.

We suggest leveraging the work done by IETF on similar subject, driven by the requirement listed above, and which has been widely deployed.

A natural candidate to leverage is IPFIX [RFC7011]: IPFIX is a means for transmitting Traffic Flow information over the network. In order to transmit Traffic Flow information, it provides a common representation of flow data and a standard means of communicating them.

Metadata collected by Network Node and Service Node SHOULD be encoded in template following the principles described in IPFIX [RFC7011].

IPFIX SHOULD be used in the context of Congruent out of band for reliable metadata sharing.

IPFIX SHOULD be used in the context of offline reliable metadata sharing.

3.1.3.1. IPFIX Message Format

An IPFIX Message consisting of interleaved Template, Data, and Options Template Sets, as shown in Figure 1. Here, Template and Options Template Sets, which are optional, are shown.

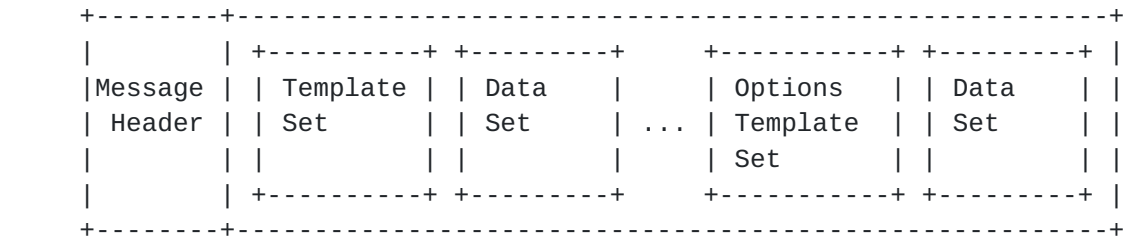


Figure 1: IPFIX Message Format

The Template Set describes the data transmitted in the following Data Set. It is an optional component of the message. The value of the metadata is encoded in the first Data Set. This Data Set contains a template Id field as a reference to its defining Template Set.

The Options Template Set describes the data to be transmitted as scope information. It is an optional component of the message. The value of the scope information is encoded in the second Data Set element. If no scope information is present, then only the first Data Set is present in the message.

The Option Template Set and following Data Set are used to describe

the scope of the metadata transmitted. For example, the metadata

collected is relevant to a PDP Context or a particular line card of a particular switch.

3.1.3.2. Message Header Format

Refer to IPFIX Section 3.1 in [RFC7011].

An IPFIX Message consisting entirely of Template and Options Template Sets

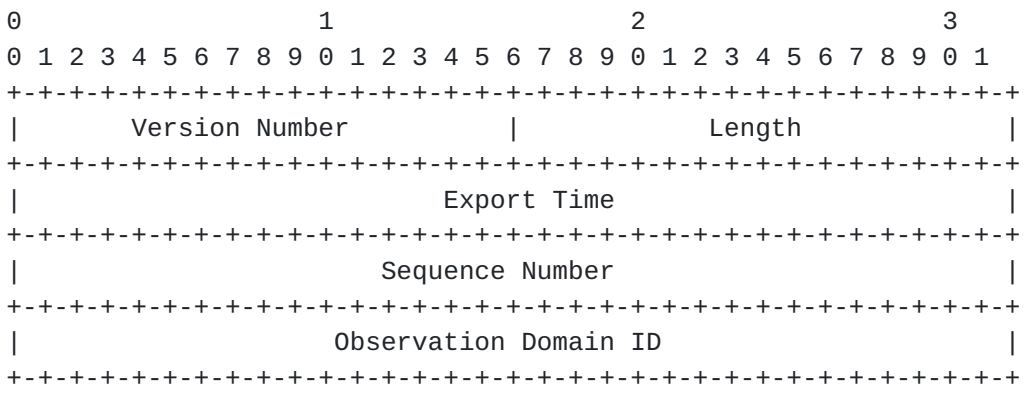


Figure 2: IPFIX Metadata Message Header

Version

Version of IPFIX to which this Message conforms. The value of this field is 0x000a for the current version

Observation Domain ID

It is RECOMMENDED that this identifier also be unique per IPFIX Device. Collecting Processes SHOULD use the Chain path, Chain index and the Observation Domain ID field to separate different export streams originating from the same Exporter. The Observation Domain ID SHOULD be 0 when no specific Observation Domain ID is relevant for the entire Metadata IPFIX Message.

The Observation Domain ID field, along with Chain path, acts as a naming space identifier. This will in particular allow for multi-tenant name space separation.

3.1.3.3. Field Specifier Format

Refer to IPFIX Section 3.2 in [RFC7011].

It defines generic Information Element identifiers and allows for enterprise specific ones. See [IANA-IPFIX] for common Information

Element identifiers definition.

Note: Additional common attributes may be defined for the purpose of SFC use cases. (E.g. PDP context identifier)

The Field Specifier format is shown in Figure 3.

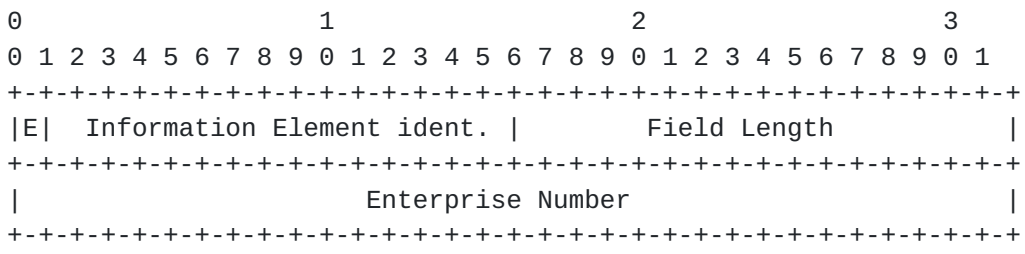


Figure 3: Field Specifier Format

Where:

E: Enterprise bit. This is the first bit of the Field Specifier. If this bit is zero, the Information Element identifier identifies an Information Element in [IANA-IPFIX], and the four-octet Enterprise Number field MUST NOT be present. If this bit is one, the Information Element identifier identifies an enterprise-specific Information Element, and the Enterprise Number field MUST be present.

3.1.3.4. Set and Set Header Format

Refer to IPFIX Section 3.3 in [RFC7011], as well in Section 4 of [RFC6759] for more details on application ID field definition, (section 6 for examples).

A Set has the format shown in Figure 4. The record types can be either Template Records, Options Template Records, or Data Records. The record types MUST NOT be mixed within a Set.

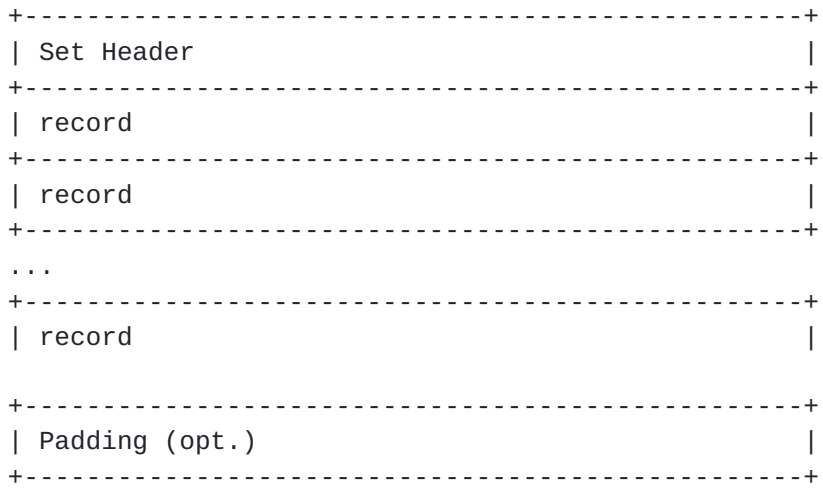


Figure 4: IPFIX Metadata Set Format

The Set Header's Set ID field value 2 is reserved for Template Sets and value 3 for Options Template Sets. Data Set value are 256 and above.

Data Set records MUST be used to transport the metadata values. The Template ID to which the Field Values belong is encoded in the Set Header field "Set ID", i.e., "Set ID" = "Template ID". This way the corresponding Template Set can be transported in the Metadata IPIX message, or can be made available to all SF in a chain as part of their configuration delivered by the chain Controller.

3.1.3.5. Record Format

3.1.3.5.1. Template Record Format

Refer to IPFIX Section 3.4.1 in [RFC7011].

The Template Record format allows applications which don't know the format of certain fields to ignore them. This is an important feature for sharing Metadata among heterogeneous Service and Network Nodes.

3.1.3.5.2. Options Record Format

Refer to IPFIX Section 3.4.2 in [RFC7011].

The Options Record format allows applications to describe the scope of the Metadata information, via a number of fields passed in the following Data Set element.

Allowing scoped Metadata provides an increased level of flexibility to the Network and Service Node when applied in the context of a particular chain.

3.2. IPFIX message scoping example:

The Metadata Exporting Process creates a Template Record with a few Information Elements:

- sourceIPv4Address (key field)
- destinationIPv4Address (key field)
- protocol (key field)
- destinationTransportPort (key field)
- octetTotalCount (non key field)

For example, a Flow Record corresponding to the above Template Record may contain:

```
{ sourceIPv4Address=192.0.2.1,
  destinationIPv4Address=192.0.2.2,
  protocol=17,
  destinationTransportPort=80,
  octetTotalCount=123456 }
```

The Options Data Record associated with the examples above would contain the Scoping inforamtion:

```
Scope:  - servicePath,
        - serviceIndex,
        - applicationId,
        - applicationName
        - applicationDescription.
```

For example:

```
{ scope=
  servicePath=0x000b,
  serviceIndex=0x000c,
  applicationId='13...10000',
  applicationName="webex",
  applicationDescription="Webex application" }
```

Scope information may be used to carry the information of the NSH header when the information is passed offline, out of band for example via controllers.

Scope information is useful when sending metadata offline, as it can contain information related to the chain and possibly the flow for which this metadata record is relevant. Here servicePath and serviceIndex are thus included in the Template.

3.3. IPFIX encoding and template provisioning

IPFIX is a quite compact encoding

For a template defined as followed and shared by the SF in a chain.

IPFIX template record shared by SF:

Note that an XML representation of IPFIX template record can be defined and used to provision Service Functions.

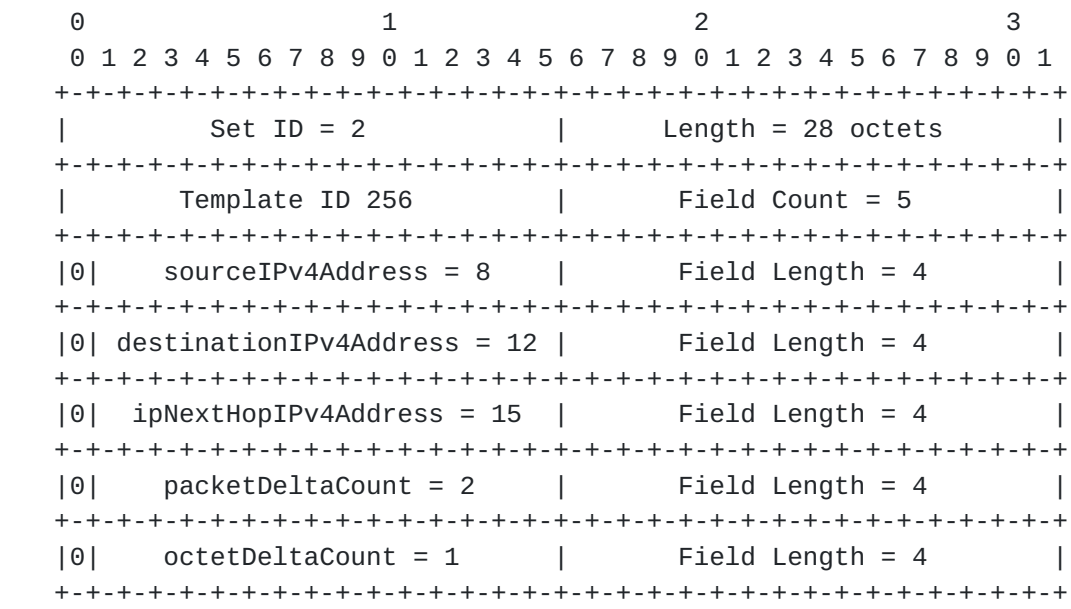
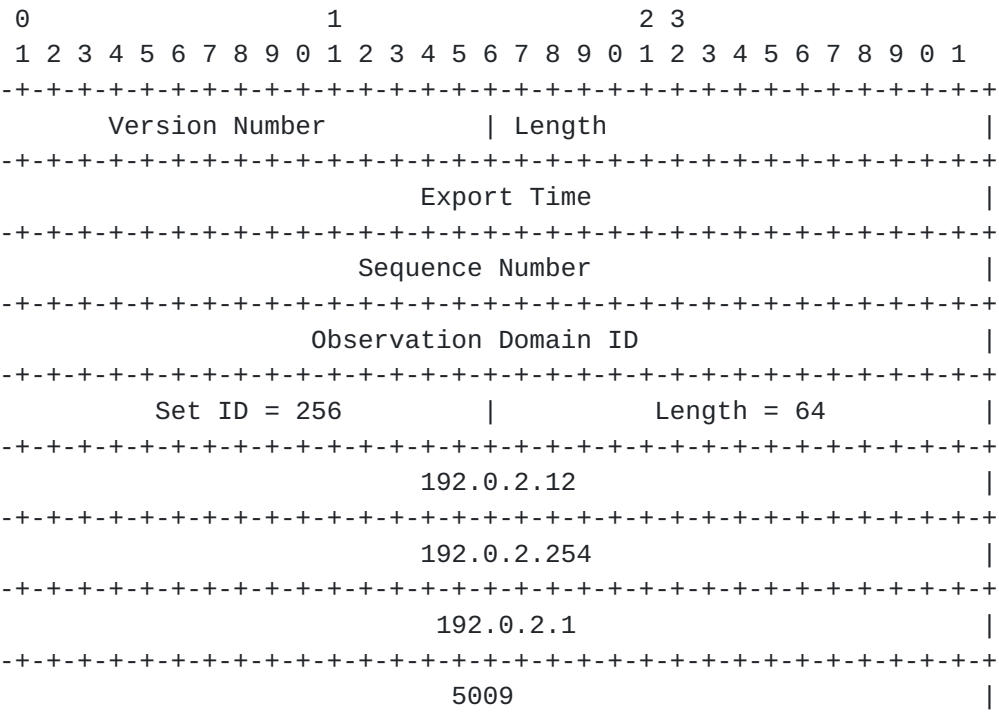


Figure 7: IPFIX Metadata template Encoding

An encoded IP fix transport message will be:



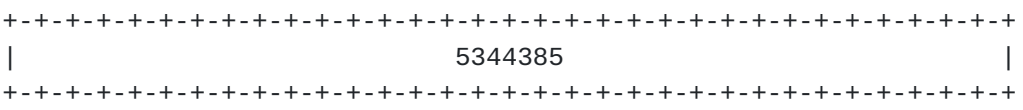


Figure 8: IPFIX Metadata Encoding

4. Reliable Metadata transport service

We saw that various uses case support the notion that Service Functions require Metadata to be transported reliably along the Rendered Service Path they belong to.

Network Orchestration for example is expected to be a significant driver for deployment of Network Services. It relies on Service Level abstractions such as Group Policies, Contracts and Services as an input for the orchestration of Service Function Chains. Specific metadata attributes such as L4-L7 fields are used as classification elements or filters to funnel packets into chains. One can expect Network Orchestration to request metadata extraction at the Classifier level, and to make it available to the Service Chaining service

Current SFC proposals shows that some Metadata element may play a role in the Service Function Chain deployment to route incoming traffic to some relevant processing resources. Other Metadata can be used by SF for their internal needs, although there is no mechanism defined yet to pass these information to the SF. In both cases, Metadata can become a critical information element, required to be transported

Indeed, Service Function Chain proposals such as NSH, SCH and NIU define a transport mechanism for sharing information along the Chains. It is thus important to understand there transport service in term of reliability. We review these separately in Section 4.1

Service Functions may have various needs to access SFC Metadata, for example Event Based Metadata tied to some subscriber related state changes. The complexity and length of these elements should not be constrained, neither should be their requirement for a reliable transport throughout a chain.

Section 4.2 and Section 4.3 propose to take advantage of Congruent Metadata Transport. It can be used, possibly reliably, to address these needs.

4.1. Metadata transport service in SFC proposals

4.1.1. Metadata transport with Network Service Header

A Network Service Header (NSH) contains metadata and service path information that is added to a packet or frame and used to create a service plane. The packets and the NSH are then encapsulated in an outer header for transport.

The service header is added by a service classification function - a

device or application - that determines which packets require servicing, and correspondingly which service path to follow to apply the appropriate service.

A NSH is composed of a 64-bit base header, and four 32-bit context headers as shown in Figure 9 below.

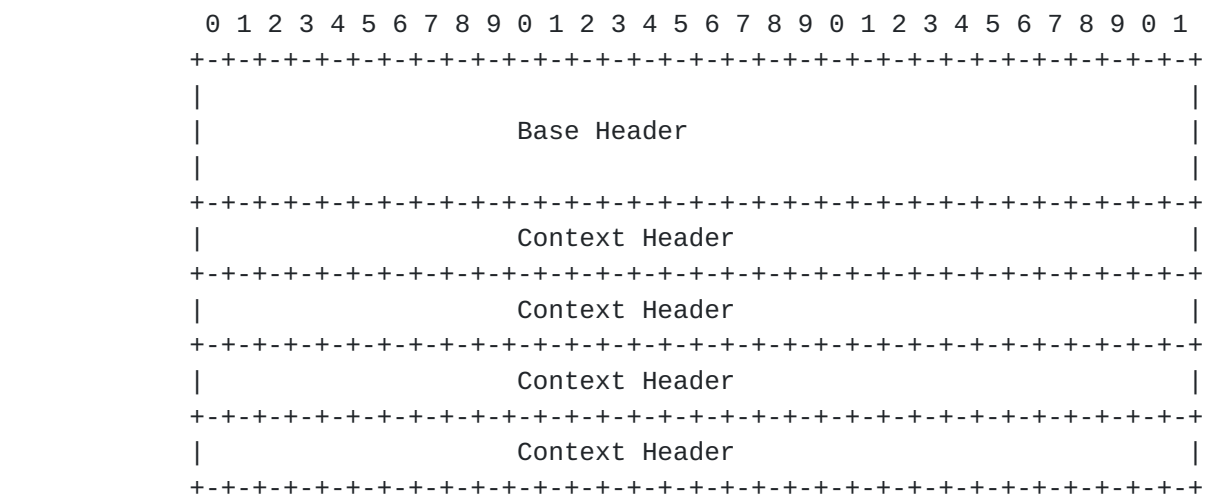


Figure 9: Network Service Header

Context headers: carry opaque metadata.

NSH provides a mechanism to carry shared metadata between network devices and service functions, and between service functions. The semantics of the shared metadata is communicated via a control plane to participating nodes. Examples of metadata include classification information used for policy enforcement and network context for forwarding post service delivery.

[QUINN_NSH] shows that metadata can be expected to identify information such as:

1. Network platform context: which is platform specific information shared between Network Nodes. Possibly platform centric, network related information.

2. Network shared context: metadata resulting for edge classification.

3. Service platform context: which is service specific information shared between Service Functions. Possibly platform-centric service related information.

4. Network shared context: metadata relevant to and shared between Service Functions.

NSH also support inline optional variable size metadata.

It contains also the notion of Critical Metadata. Critical Metadata presence is noted in the NSH Base Header so that intemediate nodes may avoid to drop such packets.

This is obviously an attempt to provide some level of reliability to the service. This is still a best effort attempt as there is not guarantee that the underlay network, unaware of NSH, drops the corresponding packets.

NSH does not state whether or not Metadata can be sent as congruent signaling messages: without carrying any payload.

4.1.2. Metadata transport with Service Chain Header

The Service Chain Header (SCH) (see Figure 10) consists of a mandatory fixed length part followed by a number of optional variable length metadata as shown in Figure 10. The mandatory fields carry "SFC path" information, which is used to steer the frames or packets through an ordered set of service function instances along the service function chain.

The optional variable length fields carry application/service/content related metadata information which can be used by any SFC entities. The optional fields are formatted as Type-Length-Value structures. If any field in the header is not in use, the value of that field MUST be set to zero.

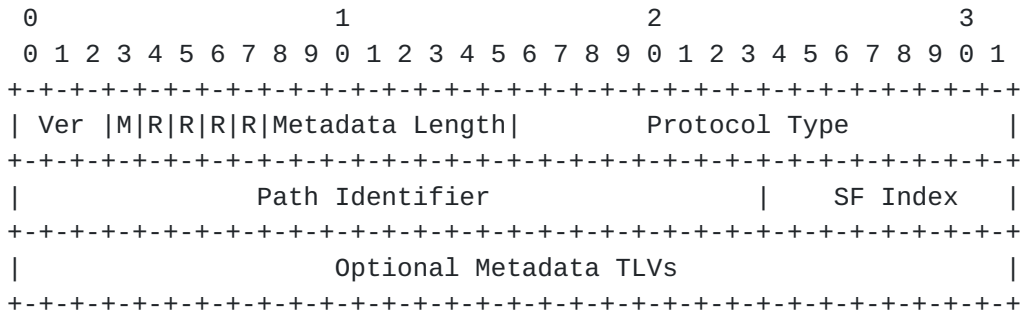


Figure 10: Service Chain Header

Optional metadata are added after the fixed part of the SCH. Each option is of variable length and has a minimum length of four octets. An optional 3-octet Organizational Unique Identifier (OUI) may be provided to differentiate multiple private number spaces for the Type field. If the OUI is not provided, the Type is assumed to be a registered globally unique type.

Figure 11. shows the format of the TLV in SCH.

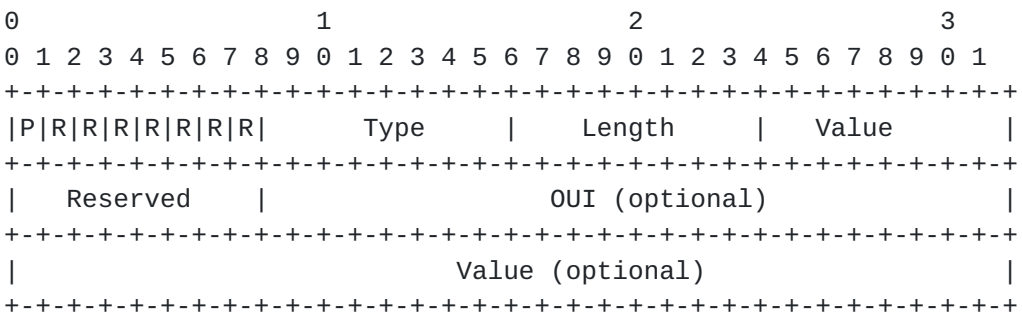


Figure 11: Service Chain Header TLV format

Metadata transport in SCH calls for the inclusion of metadata directly in the packet. The result is that long metadata can be transported inline.

Metadata is optional, so no space is taken when no metadata is sent with a packet, leading to a more compact header than NSH then. However we can see that sending 4 32 metadata fields would take more header space.

The SCH may be used to carry: (1) both SFC path steering information and metadata; (2) only SFC path steering information, in which case the Metadata Length field shall be set to zero; or (3) only metadata, in which case the Path Identifier and SF Index fields shall be set to zero for transmit and ignored upon receipt.

SCH thus allows Metadata to be sent as congruent signaling messages: without carrying any payload.

From that point of view, two notions should be supported (reliably)

1. Point to point message. SF index may then be zero

2. "Down the chain" messages. SF index would not be nul then

4.1.3. Metadata transport in NIU proposal

[NIU_SFC] also includes the notion of optional metadata. It distinguishes between FORWARDING metadata element and SERVICE metadata.

FORWARDING metadata element is used to carry the SF Processing Result Metadata. Error/Success message which can be passed from a Service Function to the next one. SF Processing Result is a fixed 32 bit field. There can be only one such element in a packet.

SERVICE metadata is encoded in TLV, which format is not specified in the current version as referenced below.

4.1.4. Metadata transport analysis

Both NSH and SCH proposals support both inbound and out-of-band metadata transport.

1. In-band: the metadata can be included directly as a value in some of the NSH Context Header Fields. It is the preferred transport model for SCH.

In such case, when a particular field is always set to the same value for all packets transported by the chain instance, then the metadata transport service is in effect reliable.

Similarly, all the packet for a particular flow (defined by its 5 tupple), could receive the same metadata value. The metadata transport service is also reliable, provided that the value is understood to be attached to a flow.

The general case is when the metadata varies from packet to packet in a flow. The value is then tied to a specific packet. Here the transport service is not reliable. A retransmission of a particular packet would not necessarily lead it to carry the same metadata value.

2. Out-of-band: the metadata is sent along a packet that is relevant to the metadata, to the controller. It is the preferred model for NSH, but could also be used by SCH.

As for the in-band case, the metadata referred to indirectly can be transmitted reliably, when it remains the same for a chain or a flow in a chain.

If however, the correlation Id passed changes over time, then the correct Metadata may not be retreived by some Service Functions.

We can see that NSH and SCH do not provide a reliable transport service for metadata. Conventions can be used as particular cases when some metadata pertains to a specific chain or a flow in the chain, and when its value does not change overtime.

Such conventions however are weak. They would suppose that some mechanism exists to ensure/monitor that they are followed. And some exceptions mechanism would be required to deal with error cases.

The general case, metadata tied to a packet, has its own set of issues. What is a packet is lost? How to pass metadata to a legacy SF, preserving the connection between the packet and its metadata.

4.2. Congruent out-of-band transport service

Congruent out-of-band metadata sharing can be required for some types

of Metadata exchanges. It has the advantage of clearly tying the metadata to the chain and not to a specific packet, and to avoid payload fragmentation issues.

Up to draft 2, NSH did not allow for long inline metadata transport. Four 32 bits context fields are reserved for that purpose, and seem best suited for offline Metadata sharing, or to transport predefined policy identifiers.

NSH (since draf 3) as well as SCH could allow for metadata transport, either tied to a packet or possibly tied to the chain, when used without payload, as signaling messages.

SCH however stipulates that in case the Path Identifier and SF Index fields shall be set to zero for transmit and ignored upon receipt, when the SCH packet will contain only metadata. So congruent out-of-band metadata, transporting Metadata hop to hop to the various Service Function in the chain, does not seem to be supported.

NSH and NUI supports inline variable size metadata. They doe not mention explicitly that congruent out-of-band metadata can be used.

4.3. Reliable transport service proposal

Some metadata will need a reliable transport service to be shared inline, as well as offline.

A protocol SCTP provides such a service and has the interesting characteristic to be packet based, as opposed to stream based like TCP.

4.3.1. Using SCTP as a reliable Metadada transport service in SFC

SCTP carries a sequence number and support retransmission and congestion control.

Figure 12 illustrates how SCTP MUST used hop by hop between SFF in a chain to transport Metadata reliably.

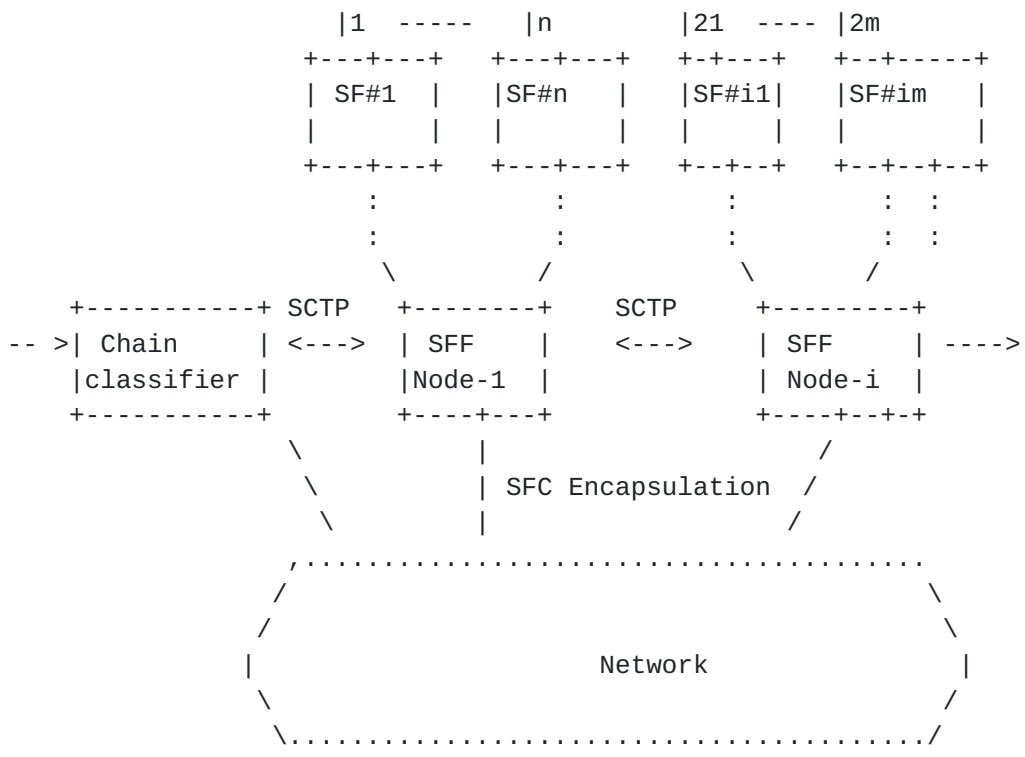


Figure 12: SCTP for Reliable Metadata transport TLV format

SCTP protocol exchanges MUST occur between SFF. The SCTP payload MUST contains the SFC header and the SFC payload.

SCTP SHOULD be used in the context of Congruent out of band for reliable metadata sharing.

SCTP SHOULD be used in the context of offline reliable metadata sharing.

The SFF along the chain MAY route the metadata received over SCTP to the next SF in the chain. For this SCTP MUST be encapsulated in the SFC header.

The SFF MUST make the received metadata available to its SF.

4.3.2. Using SCTP as a Metadada delivery service for SF

SCTP could also be used as a mechanism to deliver Metadata to SF in a chain.

Metadata is message based, so it fits well with the SCTP API and is reliable.

Defining how an SFF could delivery metadata to a SF, would facilitate

the deployment of Metadata aware SFs, allowing to distribute policies in a network for both SFC unaware application and SFC aware ones.

5. Security Considerations

As with many other protocols, SFC data can be spoofed or otherwise modified. In many deployments, SFC will be used in a controlled environment, with trusted devices (e.g. a data center) thus mitigating the risk of unauthorized header manipulation.

SFC is always encapsulated in a transport protocol and therefore, when required, existing security protocols that provide authenticity (e.g. [IPSec]) can be used.

SCTP can be run securely when transporting metadata for the chain.

Similarly if confidentiality is required, existing encryption protocols can be used in conjunction with encapsulated NSH.

6. Acknowledgments

A special thank you goes to J. Tollet and U. Elzur for their guidance and feedback.

7. IANA Considerations

An IEEE EtherType will be requested for NSH.

8. References

8.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

8.2. Informative References

[RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D. and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[RFC6071] Frankel, S. and S. Krishnan, "IP Security (IPsec) and Internet Key Exchange (IKE) Document Roadmap", RFC 6071, February 2011.

[RFC7011] Claise, B., Trammell, B. and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, September 2013.

[RFC7012] Claise, B. and B. Trammell, "Information Model for IP Flow

Information Export (IPFIX)", RFC 7012, September 2013.

[RFC7013] Trammell, B. and B. Claise, "Guidelines for Authors and Reviewers of IP Flow Information Export (IPFIX) Information Elements", BCP 184, RFC 7013, September 2013.

[RFC6759] Claise, B., Aitken, P. and N. Ben-Dvora, "Cisco Systems Export of Application Information in IP Flow Information Export (IPFIX)", RFC 6759, November 2012.

[RIJSMAN] Rijsman, B., "Metadata Considerations, draft-rijsman-sfc-metadata-considerations-00 ", .

[SFC_ARCH] Halpern, J. and C. Pignataro, "Service Function Chaining (SFC) Architecture draft-ietf-sfc-architecture-01 ", .

[QUINN_NSH] Quinn, P., "Network Service Header, draft-quinn-sfc-nsh-02.txt ", .

[ZHANG_SCH] Zang, H., "Service Chain Header draft-zhang-sfc-sch-00 ", .

[NIU_SFC] Niu, L., "A Service Function Chaining Header and Forwarding Mechanism draft-niu-sfc-mechanism-01.txt ", .

8.3. External References

[IANA-IPFIX] IANA , ""IP Flow Information Export (IPFIX) Entities", http://www.iana.org/assignments/ipfix/ ", .

Author's Address

Nicolas Bouthors
Qosmos