SFC working group   S. Vallamkonda
Internet Draft   F5 Neworks
Intended status: Standard Track   L. Dunbar
Expires: November 8, 2017   Huawei
                                                          D. Dolson
                                                           Sandvine
                                                       July 5, 2016

A Framework for SFC Metadata
draft-vallamkonda-sfc-metadata-model-01

Status of this Memo

Copyright Notice

Abstract

      Various types of metadata are applicable to Service Function
      Chaining (SFC). Metadata can be used for many purposes, such as
      conveying processing information, resource usage, and flow specific
      information, from prior nodes along the Service Function Path. It
      can also be vendor specific to leverage vendor capabilities and hint
      to downstream Service functions dynamically for improved
      performance. In contrast, metadata carried out of band introduces
      latency and overhead with inefficiency and non-synchronous to real-
      time traffic. A Service Function (SF) that needs to process the
      information carried by the metadata may need detailed information of
      the metadata structure carried by the packets and can have local
      policies based on metadata.

      The purpose of this document is to specify a framework and
      information model on how to provision information about metadata
      among classifiers and service functions on a service function chain.

Table of Contents

1. Introduction

   Service Function Chaining (SFC) is a technology for directing
   network traffic via a set of functions in a specific order. The SFC
   architecture document [RFC7665] has in-depth description of SFC,
   which will not be repeated here.

   The metadata specified by [sfc-nsh] provides a mechanism for
   additional information exchanged between nodes along the service
   function path.

   Even though many metadata exchanged among the service functions on a
   path are proprietary, there are some metadata that are expected to
   convey information from upstream service functions to downstream
   service functions by different vendors, such as time stamp and
   others. It is important that this information is not hardcoded and
   static but provisioned to a Service Node. This document will first
   describe the use cases (or the examples) of such metadata that are
   expected to be passed among service functions. It will then
   describe a framework on how to identify metadata, and specifies the
   information model and corresponding data model for those metadata.

2. Terminology

   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].

   Metadata: Information about a packet that is attached to a packet,
        specifically within the NSH header.

   SF: Service Function

3. Use Cases of Metadata exchanged by SFs (by multiple vendors)

   The SFC architecture calls for metadata to be included in packets
   sent between elements of a service chain.

   Several types of Service Functions inject packets into data streams.
   Examples include routers creating ICMP messages, or firewalls
   creating TCP reset packets. The question that naturally arises is
   what metadata should be attached to payload packets. This question
   cannot be answered without knowing what each type of metadata means.
   Further, without this there is ambiguity on limitations and
   restrictions for services offered by the service functions on the
   service function path.

```
            +----+       +----+          +----+
         ---->|SFF1|----->|SFF2|-------->|SFF |
         <----|     |<-----|      |<-------|     |
            +----+       +----+          +----+
              |             |               |
              |          +-------+          |
              |          |       |       +------+
            +----+    +----+ +----+   | SF4   |->
            |SF1 |    | SF2| |SF3 |   +------+
            +----+    +----+ +----+
             Figure 1: Metadata passed between SFs
```

   Figure 1 shows a SFC with two service functions: L3-L7 ACL with
   firewall (SF1) and second with DPI (SF2). The path can be symmetric
   or asymmetric on per pathID/flow basis.

   SF1 and SF2 are different NFVs providing different services to flow.

   Here are some examples that SF2 needs packets to carry the
   processing information done by SF1:

   SF1 may at real-time attach DoS information for the next SF in
   downstream. SF1 provides the hint and it is up to the downstream SFs
   to process it or ignore it. However if a downstream SF chooses to
   processes, it needs standardized metadata data model to understand
   the hint encoded by SF1 in packets.

SF1 may send protocol flood (DNS/HTTP/SYN) indicators in metadata.
This may be attached to packet based on local policy that can be
time orevent based. The downstream SF2 would need to be aware of a
standardized format (the proposal) to interpret the data. Then it
may process the packet per local policy.

Without standard method and framework, service functions can't pass
meaningful metadata to other SFs on a service function chain to
achieve sophisticated service functions.

[4]. Standardized Encoding of Metadata attached to packets

Metadata could be self-describing or there could be control-plane
descriptions of metadata encoding in the form of a metadata
dictionary (or a combination thereof). In either case, there needs
to be a language for describing the meaning of metadata context
vocabulary.

This document provides the analysis of various types metadata, the
framework to carry metadata across SFs or SFF on a SFC path, and the
corresponding information and data model for some well-known
metadata that can be useful for services functions.

Note: this document does not document all potential metadata used by
SFs, because there are many types of proprietary metadata exchanged
among SFs.

[5]. Framework of encoding metadata

To minimize the extra bytes added to packets in NSH, it is necessary
to have compact encoding of the metadata carried by data packets.
Achieving this goal will need control plane to inform the encoding
mechanisms to SFs via out of band control channels.

In addition, it is necessary for vendors to register the metadata
that their corresponding SFs can send and receive, as depicted in
the following diagram:

```
                  +-------------------------------------------------+
                  |                                                 |
                  |                 SFC  Control Plane       +-------+
          +-------|                                          |       |
          |       |                                          |       |C5
          C1      +------^-----------^-------------^----------+       |
      +-------------------|C3--------|-------------|------------+     |
      |       |           +----+     |             |            |     v
      |       |           | SF |     |C2           |C2          |
      +-----------+       +----+     |             |            | | SF
      |       |           +----+     |             |            | |
metadata|      |                     |             |            | |
      | +----V--- --+     |          |             |            | |
registration|        |          |          |          |          | |
      | |   SFC      |     +----+      +-|--+      +----+          |
      +-----------+
      | |Classifier |---->|SFF |----->|SFF |------->|SFF |         |
      | |   Node    |<----|    |<-----|    |    |<-------|    |    |
      | +-----------+     +----+      +----+      +----+          |
      |                     |           |            |            |
      |                     |C2       -------         |            |
      |                     |          |      |   +-----------+ C4 |
      |                     V      +----+ +----+   | SFC Proxy |--> |
      |                            | SF | |SF  |   +-----------+    |
      |                            +----+ +----+                    |
      |                             |C3    |C3                      |
      |   SFC Data Plane Components  V      V                       |
      +------------------------------------------------------------+
```
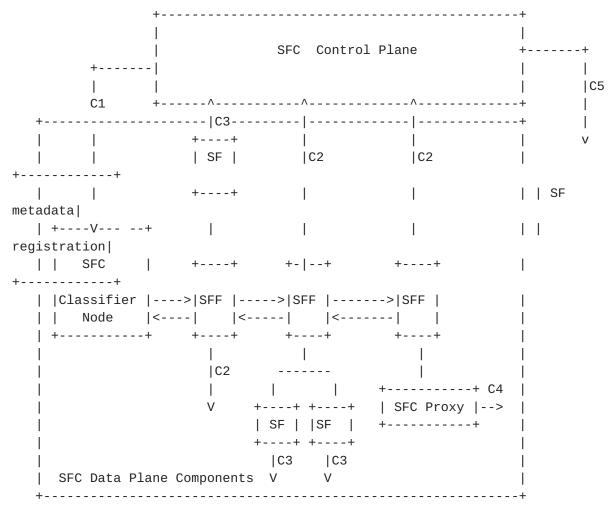
Figure 2: SFC Architecture Extension for Metadata

SF Registration Interface (C5) is for vendors of the SFs to inform
the controller on the metadata their SFs supported. The information
over this interface should include:

   SF vendor name ABC
   metadata objects
     Objects passed over the Controller - SF interface (C3)
     Objects carried by data packets, i.e. encoded in the packets'NSH
   Actions that can be performed on the SFs

6. Classes of Metadata

6.1. Metadata passed between controller and SFs

   This section describes the metadata not carried by payload packets,
   but instead communicated between controller and SFs, i.e. over the

C3 interface of the Figure 2 above.

The metadata over the C3 interface should carry the policies
associated with each metadata encoding carried by the packets
through the SFs (in NSH header).

6.1.1. IP Endpoint Property

A metadata type indicates a property of an IP endpoint of either the
source or the destination IP address in the encapsulated
conversation.

As examples, the metadata could indicate a user's class of service,
that the endpoint is flagged as the subject of an attack or may
indicate the account number to charge the user's traffic.

Injected packets may clone this type of metadata from other packets
having the same IP endpoint, for packets in the same direction.

6.2. Metadata carried by payload packets

The metadata carried by payload packets need to be encoded in the
NSH header. However, the interpretation of the encoding has to be
exchanged between controller and the SFs.

6.2.1. Routing Domain

A Routing Domain metadata type indicates the specific private
network for the packet. A policy could be "Neither traffic nor
information may cross between domains". Service functions must use
the domain to discriminate between overlapping private IPv4. When a
packet exits SFC (has the NSH header removed), a Routing Domain
metadata can indicate which routing table should be used to forward
the packet. E.g., Routing Domain metadata allows support of multiple
private networks within the same SFC cluster.

Metadata of this type is generally attached by the classifier. In
general, this type of metadata must not be removed or modified by
SFs (except in the case when the intention is to route traffic
between domains).

Injected packets must include this type of metadata, to indicate the
routing domain the packet is being injected into.

6.2.2. **Traffic Policy Indication**

   This metadata type indicates a class of treatment for customer
   traffic, which may be attached by the classifier or another SF in
   the chain.

   Class values are assigned and administered by the operator.

   This type of metadata is not required on every packet. If missing,
   a default policy can be applied.

   The most recent value can be cached for the customer IP address;
   injected packets can use the cached value.

6.2.3. **Flow Classification**

   This metadata type indicates a flow classification.

   As examples, the metadata could indicate a DPI classification result
   or whether the flow has been selected for differentiated service.

   This type may be attached by the classifier or another SF in the
   chain. It may also be overwritten by SFs along the chain.

   This type of metadata is a property of the session 5-tuple.
   Injected packets may clone this property from other packets of the
   flow, for packets in the same direction.

7. **Metadata Information and Data Model**

7.1. **Objects over the Vendor registration interface**

7.2. **Objects over the Control Plane to SF interface**

   The purpose of the control plane interface to SF is to describe to
   the classifier and service functions both the encoding and semantics
   of each type of metadata.

   The model of each instance of metadata should include:

      - Keyword name

            - Long description
          - Data type (integer, string, enumeration of type X, timestamp,
            indirect handle to Y, etc.)
          - The class of metadata is it (see section 6)?
          - how is it transported?
              - in a MD-Type1 slot number 1, , 4
              - in a MD-Type2 TLV, with code number N and vendor type V.

   "Indirect handle" indicates that the value is a key into a table
   transferred out of band. E.g., it could be a handle for a subscriber
   identity or it could be a handle for a mobile cell sector.

   TODO: model in YANG.

7.3. Objects encoded in the NSH carried by data packets over SF path

   In the data packets, metadata items are identified by either

   (a) Position within the fixed MD-Type 1 header

   (b) Vendor/Code within the variable-length MD-Type 2 header.

   The position or vendor/code is to be conveyed in the control plane
   ahead of arrival of the information in the data packets.

8. Dictionary for Metadata

   The metadata can be defined by vendor in common published format in
   ASCII file. This file could be used by other vendors of Service
   Nodes (SF/SFFs) to recognize the metadata and its content
   dynamically. The metadata can be used by local policies on Service
   Node if needed. This common format encourages rapid deployment and
   supports interoperability on real-time traffic without restrictions
   of hardcoding or worry about dynamically changing capabilities of
   Service nodes in SFC.

   VENDOR-DEF   vendor_name   vendor_id
        VENDOR-ATTRIBUTE attribute-name   attribute_ID   syntax_type
   (DEFAULT, LENGTH, etc) flags
              ATTRIBUTE-VALUE attribute-name   value_name
   value_number_associated

   Example:

   VENDOR-DEF   ABC   100
      VENDOR-ATTRIBUTE   dns-attack 10 DEFAULT
            ATTRIBUTE-VALUE dns-attack attack_state   1 (suspect), 2
   (found).

8.1. Metadata wire format

   The above dictionary format of metadata specification could be
   translated in a common wire format for interoperability. Some data
   will be passed over the C5 (Registration) interface and others will
   be passed over the C3 interface in the Figure 2 above.

   Editor's note: details will be added after the framework is
   accepted.

   Thus the salient benefits of this metadata framework are:

     o It is independent of capabilities discovery of SF by Controller which
        is configuration and provision. It is different from Metadata which
        is real time on per flow and per packet.
     o The metadata dictionary can be uploaded to Controller which is the
        central entity which can download to SFs during provision of SFs as
        OOB initially and later as needed along with its local policy for
        flows and metadata.
     o Metadata flags can specify additional information to downstream
        Service Nodes in chain such as donot-delete, append-only, etc.
     o The proposal does not limit the semantics or content context of
        Metadata at any node. The content can be local resource such as CPU,
        Storage indicators affected by the flow and/or flow specific service
        information.
     o It eliminates hardcoding, static prototyping and type guessing by
        products and versions.
     o It is independent of any specific hardware.
     o The framework is portable across SFC technologies and SFC protocols.
     o The framework can be used to enhance definitions by extending to
        subtypes within both generic and vendor global categories.

o Scale: It supports growth of vendors, their product types and
  capabilities with versions and ease of adding attribute and types.
o The highlevel class classification would be generic and vendor where
  generic would be applicable for all NFVs/Services of same category
  (minimum subset support across all products to be compatible), and
  vendor specific are enhanced based particularly on a vendor and their
  product. Both of these can be specific as any data format as JSON,
  yang, etc.

9. Security Considerations

This draft does not introduce any new security considerations beyond
what may be present in proposed solutions.

10. IANA Considerations

This document requires no IANA actions. RFC Editor: Please remove
this section before publication.

11. References

11.1. Normative References

[RFC7365] Lasserre, M. et al., "Framework for data center (DC)
          network virtualization", October 2014.

11.2. Informative References

[RFC7348] Mahalingam, M. et al., " Virtual eXtensible Local Area
          Network (VXLAN): A Framework for Overlaying Virtualized
          Layer 2 Networks over Layer 3 Networks", August 2014.

11.3. Acknowledgments

Thanks are to.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

    Sunil Vallamkonda
    F5 Networks
    3545 N. 1st Street
    San Jose, CA 95134
    USA

    Phone: +1 408 274 4820
    Email: sunilvk@f5.com

    Linda Dunbar
    Huawei Technologies
    5340 Legacy Drive, Suite 1750
    Plano, TX 75024, USA
    Phone: (469) 277 5840
    Email: ldunbar@huawei.com

    David Dolson
    Sandvine
    408 Albert Street
    Waterloo, ON N2L 3V3
    Canada

    Phone: +1 519 880 2400
    Email: ddolson@sandvine.com