MPLS Working Group A. Malis
Internet-Draft S. Bryant
Intended status: Informational Huawei Technologies
Expires: June 7, 2019 J. Halpern
Ericsson
W. Henderickx
Nokia
December 04, 2018

MPLS Encapsulation for SFC NSH
draft-ietf-mpls-sfc-encapsulation-01

Abstract

This document describes how to use a Service Function Forwarder (SFF) Label (similar to a pseudowire label or VPN label) to indicate the presence of a Service Function Chaining (SFC) Network Service Header (NSH) between an MPLS label stack and the packet payload. This allows SFC packets using the NSH to be forwarded between SFFs over an MPLS network, and the selection between multiple SFFs in the destination MPLS node.

Table of Contents

1. Introduction

As discussed in [RFC8300], a number of transport encapsulations for the Service Function Chaining (SFC) Network Service Header (NSH) already exist, such as Ethernet, UDP, GRE, and others.

This document describes an MPLS transport encapsulation for the NSH, and also describes how to use a Service Function Forwarder (SFF) [RFC7665] Label to indicate the presence of the NSH in the MPLS packet payload. This allows SFC packets using the NSH to be forwarded between SFFs in an MPLS transport network, where MPLS is used to interconnect the network nodes that contain one or more SFFs. The label is also used to select between multiple SFFs in the destination MPLS node.

SFF Labels are similar to other service labels at the bottom of an MPLS label stack that denote the contents of the MPLS payload being other than IP, such as a layer 2 pseudowire, an IP packet that is routed in a VPN context with a private address, or an Ethernet virtual private wire service.

This informational document follows well-established MPLS procedures and does not require any actions by IANA or any new protocol extensions.

2. MPLS Encapsulation Using an SFF Label

The encapsulation is a standard MPLS label stack [RFC3032] with an SFF Label at the bottom of the stack, followed by a NSH as defined by [RFC8300] and the NSH payload.

Much like a pseudowire label, an SFF Label is allocated by the downstream receiver of the NSH from its per-platform label space.

If a receiving node supports more than one SFF (i.e, more than one SFC forwarding instance), then the SFF Label can be used to select the proper SFF, by having the receiving node advertise more than one SFF Label to its upstream sending nodes as appropriate.

The method used by the downstream receiving node to advertise SFF Labels to the upstream sending node is out of scope of this document. That said, a number of methods are possible, such as via a protocol exchange, or via a controller that manages both the sender and the receiver using NETCONF/YANG, BGP, PCEP, etc. These are meant as possible examples and not to constrain the future definition of such advertisement methods.

While the SFF label will usually be at the bottom of the label stack, there may be cases where there are additional label stack entries beneath it. For example, when an ACH is carried that applies to the SFF, a GAL [RFC5586] will be in the label stack below the SFF. Similarly, an ELI/EL [RFC6790] may be carried below the SFF in the label stack. This is identical to the situation with VPN labels.

2.1. MPLS Label Stack Construction at the Sending Node

When one SFF wishes to send an SFC packet with the NSH to another SFF over an MPLS transport network, a label stack needs to be constructed by the MPLS node that contains the sending SFF in order to transport the packet to the destination MPLS node that contains the receiving SFF. The label can be constructed as follows:

1. Push on zero or more labels that are interpreted by the destination MPLS node, such as the Generic Associated Channel [RFC5586] label (see OAM Considerations below).

2. Push on the SFF Label to identify the desired SFF in the receiving MPLS node.

3. Push on zero or more additional labels such that (a) the resulting label stack will cause the packet to be transported to the destination MPLS node, and (b) when the packet arrives at the destination node, either:

* the SFF Label will be at the top of the label stack, or

* the SFF Label will rise to the top of the label stack before the packet is forwarded to another node and before the packet is dispatched to a higher layer.

2.2. SFF Label Processing at the Destination Node

The destination MPLS node performs a lookup on the SFF label to retrieve the next-hop context between the SFF and SF, e.g. to retrieve the destination MAC address in the case where native Ethernet encapsulation is used between SFF and SF. How the next-hop context is populated is out of the scope of this document.

The receiving MPLS node then pops the SFF Label (and any labels beneath it) so that the destination SFF receives the SFC packet with the NSH is at the top of the packet.

3. Equal Cost Multipath (ECMP) Considerations

As discussed in [RFC4928] and [RFC7325], there are ECMP considerations for payloads carried by MPLS.

Many existing routers use deep packet inspection to examine the payload of an MPLS packet, and if the first nibble of the payload is equal to 0x4 or 0x6, these routers (sometimes incorrectly, as discussed in [RFC4928]) assume that the payload is IPv4 or IPv6 respectively, and as a result, perform ECMP load balancing based on (presumed) information present in IP/TCP/UDP payload headers or in a combination of MPLS label stack and (presumed) IP/TCP/UDP payload headers in the packet.

For SFC, ECMP may or may not be desirable. To prevent unintended ECMP when it is not desired, the NSH Base Header was carefully constructed so that the NSH could not look like IPv4 or IPv6 based on its first nibble. See Section 2.2 of [RFC8300] for further details.

If ECMP is desired when SFC is used with an MPLS transport network, there are two possible options, Entropy [RFC6790] and Flow-Aware Transport [RFC6391] labels. A recommendation between these options, and their proper placement in the label stack, is for future study.

4. Operations, Administration, and Maintenance (OAM) Considerations

OAM at the SFC Layer is handled by SFC-defined mechanisms [RFC8300]. However, OAM may be required at the MPLS transport layer. If so, then standard MPLS-layer OAM mechanisms such as the Generic Associated Channel [RFC5586] label may be used.

5. IANA Considerations

This document does not request any actions from IANA.

Editorial note to RFC Editor: This section may be removed at your discretion.

6. Security Considerations

This document describes a method for transporting SFC packets using the NSH over an MPLS transport network. It follows well-established MPLS procedures in widespread operational use and does not define any new protocol elements or allocate any new code points, and is no more or less secure than carrying any other protocol over MPLS. To the MPLS network, the NSH and its contents is simply an opaque payload.

Discussion of the security properties of SFC networks can be found in [RFC7665]. Further security discussion regarding the NSH is contained in [RFC8300].

[RFC8300] references a number of transport encapsulations of the NSH, including Ethernet, GRE, UDP, and others. This document simply defines one additional transport encapsulation. The NSH was specially constructed to be agnostic to its transport encapsulation. As as result, in general this additional encapsulation is no more or less secure than carrying the NSH in any other encapsulation.

However, it can be argued that carrying the NSH over MPLS is more secure than using other encapsulations, as it is extremely difficult, due to the MPLS architecture, for an attempted attacker to inject unexpected MPLS packets into a network, as MPLS networks do not by design accept MPLS packets from external interfaces, and an attacker would need knowledge of the specific labels allocated by control and/or management plane protocols. Thus, an attacker attempting to spoof MPLS-encapsulated NSH packets would require insider knowledge of the network's control and management planes and a way to inject packets into internal interfaces. This is compared to, for example, NSH over UDP over IP, which could be injected into any external interface in a network that was not properly configured to filter out such packets at the ingress.

7. Acknowledgements

The authors would like to thank Jim Guichard, Eric Rosen, Med Boucadair, Sasha Vainshtein, and Jeff Tantsura for their reviews and comments.

8. References

8.1. Normative References

[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>.

[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>.

8.2. Informative References

[RFC4928] Swallow, G., Bryant, S., and L. Andersson, "Avoiding Equal Cost Multipath Treatment in MPLS Networks", BCP 128, RFC 4928, DOI 10.17487/RFC4928, June 2007, <https://www.rfc-editor.org/info/rfc4928>.

[RFC5586] Bocci, M., Ed., Vigoureux, M., Ed., and S. Bryant, Ed., "MPLS Generic Associated Channel", RFC 5586, DOI 10.17487/RFC5586, June 2009, <https://www.rfc-editor.org/info/rfc5586>.

[RFC6391] Bryant, S., Ed., Filsfils, C., Drafz, U., Kompella, V., Regan, J., and S. Amante, "Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network", RFC 6391, DOI 10.17487/RFC6391, November 2011, <https://www.rfc-editor.org/info/rfc6391>.

[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>.

[RFC7325] Villamizar, C., Ed., Kompella, K., Amante, S., Malis, A., and C. Pignataro, "MPLS Forwarding Compliance and Performance Requirements", RFC 7325, DOI 10.17487/RFC7325, August 2014, <https://www.rfc-editor.org/info/rfc7325>.

[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>.

Authors' Addresses

Andrew G. Malis
Huawei Technologies

Email: agmalis@gmail.com

Stewart Bryant
Huawei Technologies

Email: stewart.bryant@gmail.com

Joel M. Halpern
Ericsson

Email: joel.halpern@ericsson.com

Wim Henderickx
Nokia

Email: wim.henderickx@nokia.com