MPLS Working Group
Internet-Draft
Intended status: Informational
Expires: October 6, 2022

M. Bocci
Nokia
S. Bryant
University of Surrey 5GIC

April 04, 2022

Requirements for MPLS Label Stack Indicators and Ancillary Data
draft-bocci-mpls-miad-adi-requirements-03

Abstract

This draft specifies requirements for indicators in the MPLS label stack to support ancillary data in the packet and high level requirements on that ancillary data. This work is the product of the IETF MPLS Open Design Team. Requirements are derived from a number of new proposals for additions to the MPLS label stack to allow forwarding or other processing decisions to be made, either by a transit or terminating LSR (i.e. the LER), based on application data that may be in or below the bottom of the label stack.

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 https://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 October 6, 2022.

Table of Contents

1. Introduction

There is significant interest in developing the MPLS data plane to address the requirements of new applications [I-D.saad-mpls-miad-usecases]. These applications typically require the inclusion of ancillary data in the MPLS packet. This data may be encoded either in the label stack or below the bottom of the label stack. This data is then either intercepted and processed, or some other forwarding decision is taken by routers processing the packet. The ancillary data is added by the ingress LSR, and is then processed using mechanisms implemented by intermediate and/or egress LSRs that comply with the MPLS base architecture and potentially its extensions, including (but not limited to) [RFC3031], [RFC3032], [RFC6790].

This draft specifies requirements for indicators in the MPLS label stack to support these applications, as well as the encoding and use of the ancillary data.

Note: This terminology and any terms used in the naming of this draft may change as a result of working group consensus. Other content of the draft may also change, particularly as a result of [I-D.andersson-mpls-miad-fwk]. The editors will reflect this change as soon as consensus is achieved.

1.1. Terminology

- Ancillary Data: Data relating to the MPLS packet that may be used to affect the forwarding or other processing of that packet, either at an LER [RFC4221] or LSR. This data may be encoded within the label stack (in-stack data), and/or after the bottom of the label stack (post-stack data).

- In-Stack Data: Any data within the MPLS label stack including the outer LSE and the bottom of stack (the LSE with the S-bit set).

- Post-Stack Data: Any data beyond the LSE with the S-Bit set, but before the first octet of the user payload. This document does not prescribe whether post-stack data precedes or follows any other protocol structure such as a control word or associated channel header (ACH).

- Ancillary Data Indicator (ADI): A indicator in the MPLS label stack that ancillary data exists in this packet. It MAY also indicate the specific type of the ancillary data. [Note: the framework and requirements draft authors have discussed changing this to Network Action Indicators]

- End-to-End: End to end is defined as from one end of an LSP to the terminating end of the LSP. [Note: this needs to be defined in the framework].

- Hop-by-Hop: From the ingress LER or an intermediate LSR to another intermediate LSR or egress LSR. This implies processing along the LSP rather than at the endpoints, only [Note: this needs to be defined in the framework].

1.2. Background

The MPLS architecture is specified in [RFC3031] and provides a mechanism for forwarding packets through a network without requiring any analysis of the packet payload's network layer header by intermediate nodes (Label Switching Routers - LSRs). Formally, inspection may only occur at network ingress (the Label edge router - LER) where the packet is assigned to a forwarding equivalence class (FEC).

MPLS uses switching based on a label pushed on the packet to achieve efficient forwarding and traffic engineering of flows associated with the FEC. While originally used for IP traffic, MPLS has been extended to support point-to-point, point-to-multipoint and multipoint-to-multipoint layer 2 and layer 3 services. An overview

of the development of MPLS is provided in [I-D.bryant-mpls-dev-primer].

A number of applications have emerged which require LSRs to make forwarding or other processing decisions based on inspection of the network layer header, or some other ancillary information in the protocol stack encapsulated deeper in the packet. An early example of this was generation of a hash of the payload header to be used for load balancing over Equal Cost Multipath (ECMP) or Link Aggregation Group (LAG) next hops. This is based on an assumption that the network layer protocol is IP. MPLS was extended to avoid the need for LSRs to perform this operation if load balancing was needed based on the payload and instead use only the MPLS label stack, using the Entropy Label / Entropy Label Indicator [RFC6790] which are inserted at the LER. Other applications where the intermediate LSRs may need to inspect and process a packet on an LSP include OAM, which can make use of mechanisms such the Router Alert Label [RFC3032] or the Generic Associated Channel Label (GAL) [RFC5586] to indicate that an intercepted packet should be processed locally. See [I-D.bryant-mpls-dev-primer] for detailed list of such applications.

There have been a number of new proposals for how ancillary data is carried in MPLS and how its presence is indicated to the LSR or egress LER, for example In-situ OAM and Service Function Chaining (SFC). A summary of these proposals is contained in [I-D.bryant-mpls-dev-primer], an overview of use cases is provided in [Reference to MIAD use cases]. [I-D.song-mpls-extension-header] summarises some of the issues with existing solutions to address these new applications (note that this document draws on the requirements and issues without endorsing a specific solution from [I-D.song-mpls-extension-header]):

These solutions rely on either the built-in next-protocol indicator in the header or the knowledge of the format and size of the header to access the following packet data. The node is required to be able to parse the new header, which is unrealistic in an incremental deployment environment.

A piecemeal solution often assumes the new header is the only extra header and its location in the packet is fixed by default. It is impossible or difficult to support multiple new headers in one packet due to the conflicted assumption. An example of this is that the GAL/G-ACH mechanism assumes that if the GAL is present, only a single G-ACH header follows.

New applications therefore require the definition of extensions to the MPLS architecture and label stack operations that can be used

across these applications in order to minimise implementation complexity and promote interoperability and extensibility.

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. MPLS Ancillary Data Indicator Requirements

This document specifies requirements of MPLS Indicators for Ancillary Data (MIAD) and the Ancillary Data itself. The requirements are for the behavior of the protocol mechanisms and procedures that constitute building blocks out of which indicators for ancillary data are constructed. It does not specify the detailed processing that may be required by an application of that ancillary data by an LSR or LER. The requirements in this document do not describe what functions an implementation must support. The purpose of this document is to identify the toolkit and any new protocol work that is required. This new protocol work MUST be based on the existing MPLS architecture.

3.1. General Requirements

1. MPLS combines extensibility, flexibility and efficiency by using control plane context combined with a simple data plane mechanism to allow the network to make forwarding decisions about a packet. Any solution MUST maintain these properties of MPLS.

2. Any solutions to these requirements MUST NOT restrict the generality of MPLS architecture [RFC3031], [RFC3032].

3. If extensions to the MPLS data plane are required, they MUST NOT be inconsistent with the MPLS architecture [RFC3031], [RFC3032].

4. The design of any mechanism SHOULD be such that an LSR is able to efficiently parse the label stack.

5. Mechanisms MUST NOT add more labels to the stack than is necessary.

3.2. Requirements on ADIs

1. When ancillary data is present in the MPLS label stack, a mechanism is REQUIRED to indicate its presence. This mechanism is the ADI.

2. When ancillary data is present below the MPLS label stack, a mechanism is REQUIRED to indicate its presence. This mechanism is the ADI.

3. Any solution MUST respect the principle that Special Purpose Labels are the mechanism of last resort and therefore must minimise the number of new SPLs that are allocated.

4. Solutions for the ADI MUST be able to coexist with and MUST NOT obsolete existing MPLS mechanisms.

5. If the ADI is in the MPLS label stack, Ancillary Data Indicators (ADIs) SHOULD make use of existing MPLS data plane operations.

6. An ADI MUST NOT be delivered to a node that is not capable of processing it. That is, an ADI MUST NOT become top of stack at a node that does not understand the ADI and is thus is not able to perform a disposition operation on it. Disposition includes both processing and ignoring.

7. The ADI design MUST support end-to-end (E2E) processing of ancillary data.

8. The ADI design MUST support hop-by-hop (HBH) processing of ancillary data.

9. If a design allows both HBH and E2E ancillary data indicators to be present in the same packet, a mechanisms MUST be provided to specify the precedence.

10. A mechanism is REQUIRED to enable an LER inserting ADIs to determine if the far-end LER can accept and process a packet containing a given ADI.

11. ADIs SHOULD be supported for both P2P and P2MP paths, but any specific ADI may only be supported for one or the other.

12. Data plane mechanisms for ADIs MUST be independent of the control plane type (LDP, RSVP, BGP, static, IGP, etc).

13. A mechanism MUST be defined for control planes in use (e.g. LDP, RSVP, BGP, static, IGP, etc) to determine the ability of downstream LSRs/LERs to accept/process a given ADI.

14. A mechanism is REQUIRED to enable an LSR to determine if an ADI is present in a packet.

15. ADIs can only be inserted at LERs, but MAY be processed at LSRs and LERs. If it is required to insert an ADI at a transit router on an LSP, then a new label stack MUST be pushed.

16. It SHOULD be possible to include indicators for multiple ancillary data objects in the same packet.

17. The solution MUST allow ADI-carrying and non-ADI-carrying packets to coexist on the same LSP.

18. The solution MUST support the processing of a subset of the ADIs on a packet.

19. The design of the ADIs MUST NOT expose confidential information [RFC6973] [RFC3552] to the LSRs.

20. Any specification of a solution that inserts or modifies the ADI MUST discuss the possible ECMP consequences.

3.3. Requirements on Ancillary Data

1. Solutions for ancillary data within the label stack MUST be able to coexist with and MUST NOT obsolete existing MPLS mechanisms.

2. A common preamble for ancillary data MUST be defined so that a node receiving the ancillary data can determine whether to process, ignore, skip over or discard it according to network or local policies.

3. Any specification of a mechanism MUST describe whether it can coexist with existing post-stack data mechanisms e.g. control words and G-ACH, and if so how this coexistence operates.

4. A mechanism MUST be defined for an LER inserting ancillary data to determine that each node that needs to process the ancillary data can read the required distance into the packet at that node, for example [RFC9088].

5. Ancillary data MAY be associated with control or maintenance information for traffic carried by an LSP, and/or it MAY be associated with the user traffic itself.

6. For HBH ancillary data, a mechanism is REQUIRED to enable an LER inserting ADIs to determine if the ADI will be processed by LSRs along the path. Such a mechanism MAY need to determine if LSRs along the path can process a specific type of AD indicated by the ADI at the depth in the stack that it will be presented to the LSR.

7. Application specifications MUST specify if the ancillary data needs to be processed as a part of the immediate forwarding operation and whether packet mis-ordering is allowed to occur as a result of the time taken to process the ancillary data.

8. In order to prevent unnecessary scanning of the packet, care needs to be taken in the location of post stack ancillary data, for example it SHOULD be located as close to the bottom of the label stack as possible.

9. A solution MUST be provided to verify the authenticity of ancillary data processed to LSRs [RFC3552].

10. The design of the ancillary data MUST NOT expose confidential information [RFC6973] [RFC3552] to the LSRs.

4. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

5. Security Considerations

The mechanisms required by this document introduce new security considerations to MPLS. Individual solution specifications meeting these requirements MUST address any security considerations.

6. Acknowledgements

The authors gratefully acknowledge the contributions from Greg Mirsky, Yingzhen Qu, Haoyu Song, Tarek Saad, Loa Andersson, and Bruno Decraene.

The authors also gratefully acknowledge the input of the members of the MPLS Open Design Team.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[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>.

7.2. Informative References

[I-D.andersson-mpls-miad-fwk] Andersson, L., Bryant, S., and M. Bocci, "MPLS MIAD Framework", draft-andersson-mpls-miad-fwk-01 (work in progress), March 2022.

[I-D.bryant-mpls-dev-primer] Bryant, S., "A Primer on the Development of MPLS", draft-bryant-mpls-dev-primer-01 (work in progress), December 2021.

[I-D.saad-mpls-miad-usecases] Saad, T., Makhijani, K., Song, H., and G. Mirsky, "Use Cases for MPLS Function Indicators and Ancillary Data", draft-saad-mpls-miad-usecases-01 (work in progress), March 2022.

[I-D.song-mpls-extension-header] Song, H., Li, Z., Zhou, T., Andersson, L., and Z. Zhang, "MPLS Extension Header", draft-song-mpls-extension-header-06 (work in progress), January 2022.

[RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <https://www.rfc-editor.org/info/rfc3031>.

[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>.

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003, <https://www.rfc-editor.org/info/rfc3552>.

[RFC4221] Nadeau, T., Srinivasan, C., and A. Farrel, "Multiprotocol Label Switching (MPLS) Management Overview", RFC 4221, DOI 10.17487/RFC4221, November 2005, <https://www.rfc-editor.org/info/rfc4221>.

[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>.

[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>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <https://www.rfc-editor.org/info/rfc6973>.

[RFC9088] Xu, X., Kini, S., Psenak, P., Filsfils, C., Litkowski, S., and M. Bocci, "Signaling Entropy Label Capability and Entropy Readable Label Depth Using IS-IS", RFC 9088, DOI 10.17487/RFC9088, August 2021, <https://www.rfc-editor.org/info/rfc9088>.

Authors' Addresses

Matthew Bocci
Nokia

Email: matthew.bocci@nokia.com

Stewart Bryant
University of Surrey 5GIC

Email: sb@stewartbryant.com