Workgroup: MPLS Working Group
Internet-Draft:
draft-andersson-mpls-mna-label-stack-operations-00
Published: 26 September 2022
Intended Status: Standards Track
Expires: 30 March 2023
Authors: L. Andersson, Bronze Dragon Consulting
J. Guichard, Futurewei Technologies
H. Song, Futurewei Technologies
S. Bryant, University of Surrey

MPLS Label Stack Operations in Networks with MNA Incremental Deployment

Abstract

MPLS Network Action (MNA) allows MPLS packet to carry instruction and data for in-network services and functions in an MPLS network. This document describes the FEC-based optimized operations on the MPLS label stack when the network is mixed with LSRs which are capable or incapable of processing MNAs.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

MPLS Network Actions (MNA) is used to support actions for Label Switched Paths (LSPs) and/or MPLS packets in addition to the normal forwarding. [I-D.ietf-mpls-mna-fwk] provides the architectural framework for MNA and [I-D.ietf-mpls-mna-requirements] provides the design requirements for MNA. MNA can support actions encoded within or below the label stack. [I-D.andersson-mpls-eh-architecture] describes some further architectural concepts for MNA.

This document provides the operating procedures for MNA-capable and non-MNA-capable LSRs where MNA encoding are carried within or below the MPLS label stack. We show that MNAs can be gradually introduced into an existing MPLS network. The capability to handle MNAs is announced throughout the MPLS network, and LSRs that do not understand this information simply ignore it.

The MNAs are carried below the top label and the presence of MNAs are indicated by a bSPL in the label stack.

The MNA use cases can be found in [I-D.ietf-mpls-mna-usecases]. A post-stack extension header may for example be used when it is required that the packet carry a large instruction header and/or metadata for an MNA [I-D.song-mpls-extension-header].

Only MNA capable LSRs will process MNAs, LSRs that are non-MNA-capable will ignore the MNA and forward the packet as if the information was not there.

1.1. Requirement 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.

2. Operations on an MPLS Label Stack in an MNA capable network

This document provides a set of examples to show the operations performed on MPLS encapsulated packets in a network where MPLS MNAs are used. The document does also illustrated the procedures for processing of the information carried within the MPLS label stack to indicate the presence of MNAs below the top label.

2.1. Physical Topology

Assume a physical topology that includes both MNA capable LSRs and non-MNA capable LSRs. The topology is intentionally kept quite simple.

```
+----+      +----+      +----+      +----+      +----+      +----+
|    |      |    |      |    |      |    |      |    |      |    |
| A  +------+ b  +------+ c  +------+ D  +------+ E  +------+ F  +
|    |      |    |      |    |      |    |      |    |      |    |
+----+      +----+      +----+      +----+      +----+      +----+
```

Legend:
A, D, E, and F are MNA capable LSRs

b and c are non-MNA capable LSRs.

Figure 1: MNA topology I

LDP Downstream on Demand (DoD) or Downstream Unsolicited (DU), RSVP-TE, an IGP or a centralized controller could be used to create the label mappings between the LSRs in an MNA capable network. Referring to Figure 1, and using LDP DU for illustration, creation of an MNA path used by A to send MPLS encapsulated packets with MNAs to F is as show below.

For prefix F reachable at LSR F:

* F advertises labels F:[ldp: implicit-null, MNA-FEC: implicit-null] to E

* E advertises labels F:[ldp: 101, MNA-FEC: 201] to D

* D advertises label F:[ldp: 102] to c

* c advertises label F:[ldp: 103] to b

* b advertises label F:[ldp: 104] to A

This will result in installed labels as shown in Figure 2.

```
+---+       +---+       +---+       +---+       +---+       +---+
|   |..104..|   |..103..|   |..102..|   |..101..|   |..php..|   |
| A +-------+ b +-------+ c +-------+ D +-------+ E +-------+ F +
|   |       |   |       |   |       |   |..201..|   |..php..|   |
+---+       +---+       +---+       +---+       +---+       +---+

Legend:
A, D, E and F are MNA capable nodes.
b and are non-MNA capable nodes.
```

Figure 2: MNA topology II

2.2. A day in the life of a packet

This section provides examples of forwarding for some common scenarios in networks with a mix of MNA-capable and non-MNA-capable LSRs and packets with and without MNAs encoded.

The examples assume the use of post-stack extension headers. The process is equally applicable to in-stack MNAs.

For reference the following shows the full MPLS MNA stack, i.e. including also the post-stack EH specific information and the payload.

```
0                                  31
+--------+--------+--------+--------+
|                                   |
~         MPLS Label Stack          ~
|                                   |
+--------+--------+--------+--------+
|           bSPL for MNA            |
+--------+--------+--------+--------+
|                                   |
~         MPLS Label Stack          ~
|                                   |
+--------+--------+--------+--------+
| Header of Extension Headers (HEH) |
+--------+--------+--------+--------+
|                                   |
~  Extension Header (EH) for MNA 1  ~
|                                   |
+--------+--------+--------+--------+
|                                   |
~  Extension Header (EH) for MNA N  ~
|                                   |
+--------+--------+--------+--------+
|                                   |
~   Upper Layer Protocols/Payload   ~
|                                   |
+--------+--------+--------+--------+
```

Figure 3: Label Stack with MNA

2.2.1. Non-VPN Case

For non-VPN there are two variants, either the MNA is present or it is not.

2.2.1.1. Non-VPN with an MNA in the packet

* A sends packet to b

   - stack = [104, bSPL, HEH, EH, IP]

* b is a legacy router so just swaps [104] to [103], and sends the packet to c

   - stack = [103, bSPL, HEH, EH, IP]

* c is a legacy router so just swaps [103] to [102], and sends the packet to D

   - stack = [102, bSPL, HEH, EH, IP]

* D is an MNA capable LSR and receives the packet with [102] on top of the stack; D scans the packet for an MNA; D finds the MNA and processes it and then swaps the top label to [101] and then sends the packet on to E

   i Note: this goes on the standard FEC because we only announce in the packet there is NO MNA. In this case MNA is present.

   - stack = [101, bSPL, HEH, EH, IP]

* E receives [101] and scans the packet for MNA; it finds the MNA and processes it and then pops the top label and send the packet to F

   - stack = [bSPL, HEH, EH, IP]

      o Note: E is the penultimate hop router so it pops the standard LDP label, and send on the standard FEC to F.

* F receives the packet and scans the packet for MNA; it finds the MNA and processes it. As F is the ultimate hop it pops GAL, and removes bSPL, HEH and EH, processes IP and forwards the packet.

2.2.1.2. Non-VPN without any MNA in the packet

In this example there is no MNA present in the packet.

* A sends packet to b

  - stack = [104, IP]

* b receives the packet, b is a legacy router so it just swaps [104] to [103] and sends the packet to c

  - stack = [103, IP]

* c receives the packet, c is a legacy router so it just swaps [103] to [102], and sends the packet to D

  - stack = [102, IP]

* D receives the packet. Since D is an MNA capable router, it searches the packet for MNA but finds nothing, so D swaps [102] to [201], and sends the packet to E

  - stack = [201, IP]

    o Note: in this case D sends the packet using the MNA-FEC as MNA is *not* present.

    o Note: If downstream is not MNA capable then D sends the packet on the standard FEC.

* E receives the packet [201] and bypasses MNA searching and processing (received on the "no MNA present" FEC; E is penultimate node so it pops MNA-FEC label; and send the packet to F.

  - stack = [IP]; not exactly a "label stack", but listed here for symmetry

* F receives [IP] and routes the packet

2.3. The VPN case

In these two examples there is VPN information in the label stack, in the first there also MNAs in the packet.

2.3.1. VPN with MNA in the packet

* A sends packet to b

  - stack = [104, VPN, bSPL, HEH, EH, IP]

* b receives the packet; b is a legacy router and just swaps [104] to [103] and sends the packet to c

  - stack = [103, VPN, bSPL, HEH, EH, IP]

* c receives the packet; c is a legacy router and just swaps [103] to [102] and sends the packet to D

  - stack = [102, VPN, bSPL, HEH, EH, IP]

* D receives the packet; D is an MNA capable LSR; D will search the packet for MNA and will find and process the MNA; D will then swap [102] to [101] and sends the packet to E

  - stack = [101, VPN, bSPL, HEH, EH, IP]

    o Note: This packet will be sent normal IP standard FEC; only packets that does not include any MNA will be sent on the "no MNA present" FEC.

* E receives the packet; E is MNA capable LSR; E will search the packet for MNA and will find and process the MNA; E will then pop [101] and sends the packet to F

  - stack = [VPN, bSPL, HEH, EH, IP]

    o Note: E is penultimate hop so pops the LDP label and send the packet on normal IP standard FEC; only packets that does not include any MNA will be sent on the "no MNA present" FEC.

* F receives and scans the packet for MNA; it finds an MNA and processes it. As F is the ultimate hop it pops the bSPL and removes HEH and EH, processes the VPN label and forwards the packet.

2.3.2. VPN without MNA in the packet

*A sends packet to b

-stack = [104, VPN, IP]

*b receives the packet; b is a legacy router and just swaps [104] to [103] and sends the packet to c

-stack = [103, VPN, IP]

*c receives the packet; c is a legacy router and just swaps [103] to [102] and sends the packet to D

-stack = [102, VPN, IP]

*D receives the packet; D is MNA capable LSR; D will search the packet for MNA; D will not find any MNA; D will then swap [102] to [201] and sends the packet to E on the "no MNA present" FEC.

-stack = [101, VPN, IP]

oNote: This packet will be sent on the "no MNA pesent" FEC;

*E receives the packet [201] and bypasses MNA processing (received on the "no MNA present" FEC; E is the penultimate node so it pops MNA- FEC label; and send the packet to F on the "no MNA present" FEC.

-stack = [VPN, IP]

oNote: E is penultimate hop so E pops the "no MNA present" label and send the packet to F.

*F receives and scans the packet for MNA; finds no MNA and bypasses MNA processing. As F is the ultimate hop it processes the VPN label and forwards the packet.

2.4. RSVP-TE Tunnel case

The RSVP-TE tunnel is not MNA capable or the capability has been disabled.

Assume a physical topology that includes both MNA capable LSRs and non-MNA capable LSRs, as in the earlier examples. This topology also includes a low cost RSVP-TE tunnel between b and D.

```
   +----+      +----+      +----+      +----+      +----+      +----+
   |    |      |    |      |    |      |    |      |    |      |    |
   | A  +------+ b  +------+ c  +------+ D  +------+ E  +------+ F  +
   |    |      |    |      |    |      |    |      |    |      |    |
   +----+      +----+      +----+      +----+      +----+      +----+
                 | |                     | |
                 | |_____________________| |
                 |_________________________|


Legend:
A, D, E, and F are MNA capable LSRs

b and c are non-MNA capable LSRs.

Nodes that transport the RSVP-TE tunnel are not MNA capable, or the MNA
capability is disabled.
```

Figure 4: MNA topology III

For this example the following assumptions are made:

* An RSVP-TE tunnel has been established between b and D (packets will bypass c)

* F is reachable at b through RSVP-TE tunnel

* LDP is enabled on the RSVP-TE tunnel

For prefix [F]: The following label mappings are sent by the LSRs in the network.

* F advertises labels F: [ldp: implicit-null, MNA-FEC: implicit-null] to E

* E advertises labels F: [ldp: 101, MNA-FEC: 201] to D

* D advertises label F: [ldp: 102] to c and F:[ldp: 102] to b

* c advertises label F: [ldp: 103] to b

* b advertises label F: [ldp: 104] to A

This will result in label mappings like this.

```
   +----+       +----+       +----+       +----+       +----+       +----+
   |    |--104..|    |..103..|    |..102..|    |..101..|    |..php..|    |
   | A  +-------+ b  +-------+ c  +-------+ D  +-- ----+ E  +-------+ F  +
   |    |       |    |       |    |       |    |..201..|    |..php..|    |
   +----+  -    +----+       +----+       +----+       +----+       +----+
                  | |                     | |
                  | +---------------------+ |
                  |      [RSVP, 102]        |
                  +-------------------------+
```

Legend:
A, D, E, and F are MNA capable LSRs

b and c are non-MNA capable LSRs.

Nodes that transport the RSVP-TE tunnel are not MNA capable, or the MNA capability is disabled. [RSVP] represents the series of tunnel top labels.

Figure 5: MNA topology IV

To describe the label stack operations in this case the VPN label stack is used, starting with the case where an MNA is present in the packet.

2.4.1. RSVP Tunnel and MNA present in the packet

*A sends packet to b

-stack = [104, VPN, bSPL, HEH, EH, IP]

*b receives the packet, since b is a legacy router it swaps [104] to [102], the next-hop reachable through the RSVP-TE tunnel; push the ingress RSVP-TE tunnel label and send it via the tunnel to the tunnel endpoint D

-stack = [RSVP, 102, VPN, bSPL, HEH, EH, IP]

*Intermediate tunnel LSRs will forward (swap) based on the RSVP-TE label.

*D receives the packet, D will pop the last RSVP-TE label; since D is an MNA capable router it will search the stack and find the

MNA, after processing the MNA it will swap [102] to [101], and send the packet to E over the normal FEC

- stack = [101, VPN, bSPL, HEH, EH, IP]

  - Note: this will be forwarded on the standard FEC because since the MNA is present in the packet, only packet without any MNA is forwarded on the "no MNA present" FEC.

* E receives the packet [101]; since E is an MNA capable router it will search the stack and find the MNA; after processing the MNA it will pop [101], and send the packet to E over the normal FEC

- stack = [VPN, bSPL, HEH, EH, IP]

  - Note: As E is the penultimate hop it will pop the standard LDP label.

* F receives the packet with the VPN label on top [VPN]; E scans the packet for MNA; it finds the MNA and processes it. As F is the ultimate hop it pops bSPL, and removes HEH and EH, processes VPN label and forwards the packet.

2.4.2. RSVP Tunnel and no MNA present in the packet

* A sends packet to b

- stack = [104, VPN, IP]

* b receives the packet [104]; be is legacy router and will not search for an MNA; b swaps [104] to [102]; pushes [RSVP] sends packet to D over the RSVP-TE tunnel.

- stack = [RSVP, 102, VPN, IP]

* Intermediate tunnel LSRs will forward (swap) based on the RSVP-TE label.

* D receives pops the tunnel label [RSVP], D is MNA capable and scans the packet for MNA; D finds no MNA is present; pops RSVP-TE label, and then swaps LDP label [102]to [201] and sends the packet to E

- stack = [201, VPN, IP]

  - Note: in this case D sends the packet using the "no MNA present" FEC, since there is no MNA in the packet.

oNote: If the downstream LSR is not MNA capable then D will sends the packet on the standard FEC.

*E receives [201] and bypasses MNA processing since the packet is received on the "no MNA present" FEC; E is the pen-ultimate hop so it pops the MNA-FEC label and forward the packet to F

-stack = [VPN, IP]

*F receives the packet [VPN]; and scans the packet for MNA; it does not find any MNA, and it processes VPN label and forwards the packet.

2.4.3. EH capable RSVP-TE tunnel

The case where an RSVP-TE tunnel is both MNA capable and MNA enabled is for further study.

3. Security Considerations

TBA

4. IANA Considerations

There are no requests for IANA actions in this document.

Note to the RFC Editor - this section can be removed before publication.

5. Acknowledgments

TBA

-

6. References

6.1. Normative References

[I-D.andersson-mpls-eh-architecture] Andersson, L., Guichard, J. N., Song, H., and S. Bryant, "MPLS Extension Header Architecture", Work in Progress, Internet-Draft, draft-andersson-mpls-eh-architecture-03, 5 April 2022, <https://www.ietf.org/archive/id/draft-andersson-mpls-eh-architecture-03.txt>.

[I-D.ietf-mpls-mna-fwk] Andersson, L., Bryant, S., Bocci, M., and T. Li, "MPLS Network Actions Framework", Work in Progress, Internet-Draft, draft-ietf-mpls-mna-fwk-01, 8 September

2022, <https://www.ietf.org/archive/id/draft-ietf-mpls-mna-fwk-01.txt>.

[I-D.ietf-mpls-mna-requirements] Bocci, M., Bryant, S., and J. Drake, "Requirements for MPLS Network Action Indicators and MPLS Ancillary Data", Work in Progress, Internet-Draft, draft-ietf-mpls-mna-requirements-03, 19 August 2022, <https://www.ietf.org/archive/id/draft-ietf-mpls-mna-requirements-03.txt>.

[I-D.ietf-mpls-mna-usecases] Saad, T., Makhijani, K., Song, H., and G. Mirsky, "Use Cases for MPLS Network Action Indicators and MPLS Ancillary Data", Work in Progress, Internet-Draft, draft-ietf-mpls-mna-usecases-00, 19 May 2022, <https://www.ietf.org/archive/id/draft-ietf-mpls-mna-usecases-00.txt>.

[I-D.song-mpls-eh-indicator] Song, H., Li, Z., Zhou, T., and L. Andersson, "Options for MPLS Extension Header Indicator", Work in Progress, Internet-Draft, draft-song-mpls-eh-indicator-05, 27 June 2022, <https://www.ietf.org/archive/id/draft-song-mpls-eh-indicator-05.txt>.

[I-D.song-mpls-extension-header] Song, H., Li, Z., Zhou, T., Andersson, L., Zhang, Z., Gandhi, R., Rajamanickam, J., and J. Bhattacharya, "Support MPLS Network Actions using Post-Stack Extension Headers", Work in Progress, Internet-Draft, draft-song-mpls-extension-header-10, 1 September 2022, <https://www.ietf.org/archive/id/draft-song-mpls-extension-header-10.txt>.

[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>.

[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>.

6.2. Informative References

[RFC7274] Kompella, K., Andersson, L., and A. Farrel, "Allocating and Retiring Special-Purpose MPLS Labels", RFC 7274, DOI

10.17487/RFC7274, June 2014, <https://www.rfc-editor.org/info/rfc7274>.

[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>.

Authors' Addresses

Loa Andersson
Bronze Dragon Consulting

Email: loa@pi.nu

James N Guichard
Futurewei Technologies

Email: james.n.guichard@futurewei.com

Haoyu Song
Futurewei Technologies

Email: haoyu.song@futurewei.com

Stewart Bryant
University of Surrey

Email: stewart.bryant@gmail.com