

MPLS Working Group
Internet-Draft
Intended status: Standards Track
Expires: 21 January 2026

J. Rajamanickam, Ed.
R. Gandhi, Ed.
Cisco Systems, Inc.
R. Zigler
Broadcom
H. Song
Futurewei Technologies
K. Kompella
Juniper Networks
20 July 2025

MPLS Network Action (MNA) Sub-Stack Solution
draft-ietf-mpls-mna-hdr-13

Abstract

This document defines the MPLS Network Actions (MNA) sub-stack solution for carrying Network Actions and Ancillary Data in the label stack. MNA can be used to influence packet forwarding decisions, carry additional Operations, Administration, and Maintenance information in the MPLS packet or perform user-defined operations. This solution document specifies In-stack network action and In-stack data specific requirements found in "Requirements for MPLS Network Actions". This document follows the architectural framework for the MNA technologies specified in "MPLS Network Actions (MNA) Framework".

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 21 January 2026.

Copyright Notice

Copyright (c) 2025 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	3
2. Conventions Used in This Document	3
2.1. Requirements Language	3
2.2. Abbreviations	4
3. Overview	6
4. Label Stack Entry Formats	6
4.1. LSE Format A: The MNA Sub-Stack Indicator	6
4.2. LSE Format B: The initial opcode	6
4.3. LSE Format C: Subsequent opcodes	7
4.4. LSE Format D: Additional Data	8
5. The MNA Sub-Stack	9
5.1. Opcodes	9
5.2. Data	9
5.3. Scope	10
5.4. Unknown Network Action Handling	11
5.5. Ordering	12
6. Special Opcodes	12
6.1. bSPL Protection	12
6.2. Flag-Based NAIs without AD	12
6.3. No-Operation Opcode	13
6.4. Extension Opcode	13
7. NAS placement in the Label Stack	13
7.1. Actions when Pushing Labels	14
8. Node Capability Signaling	15
9. Processing the Network Action Sub-Stack	15
9.1. Encapsulating Node Responsibilities	15
9.2. Transit Node Responsibilities	16
9.3. Penultimate Node Responsibilities	16
9.4. Egress Node Responsibilities	16
10. Network Action Indicator Opcode Definition	16
11. Backward Compatibility	17
12. Implementation Status	18
12.1. University of Tuebingen Implementation	18
13. Security Considerations	18
14. IANA Considerations	19
14.1. MNA bSPL Label	19
14.2. MPLS Network Actions Parameters	19

14.3. Network Action Flags Without Ancillary Data	19
14.4. Network Action Opcodes	20
15. Examples	21
15.1. Network Action Encoding Examples	21
15.1.1. Network Action Flags without AD	21
15.1.2. Network Action Opcode with AD	22
15.1.3. Network Action Opcode with more AD with Format-B . .	23
15.1.4. Network Action Opcode with more AD with Format C . .	23
15.2. Network Action Processing Order	24
15.2.1. Network Action Processing Order	24
16. References	25
16.1. Normative References	25
16.2. Informative References	26
Acknowledgments	27
Contributors	27
Authors' Addresses	29

1. Introduction

[RFC3032] defines the encoding of the MPLS label stack, the basic structure used to define a forwarding path. Forthcoming applications require MPLS packets to perform special network actions and carry optional Ancillary Data (AD) that can affect the packet forwarding decision or trigger Operations, Administration, and Maintenance (OAM) logging, for example. Ancillary Data can be used to carry additional information, such as a network slice identifier or an entropy value for load-balancing. Several MNA applications are described in [RFC9791].

This solution document specifies In-stack network action and In-stack data (ISD) specific requirements found in [RFC9613].

This document defines the syntax and semantics of network actions and ancillary data encoded in an MPLS Label Stack. In-stack actions and ancillary data are contained in a Network Action Sub-Stack (NAS), which is recognized by a new base Special Purpose Label (bSPL). This document follows the framework specified in [RFC9789].

2. Conventions Used in This Document

2.1. 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] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2.2. Abbreviations

The terminology defined in [RFC9789] and [RFC9613] is used in this document.

Abbreviation	Meaning	Reference
AD	Ancillary Data	[RFC9613]
bSPL	Base Special Purpose Label	[RFC9017]
BOS	Bottom Of Stack	[RFC3032]
HBH	Hop-By-Hop Scope	[RFC9789]
I2E	Ingress-To-Egress Scope	[RFC9789]
IHS	I2E, HBH, or Select Scope	[RFC9789], This document
ISD	In-stack Data	[RFC9613]
LSE	Label Stack Entry	[RFC3032]
MNA	MPLS Network Actions	[RFC9789]
NAI	Network Action Indicator	[RFC9613]
NAL	Network Action Length	This document
NAS	Network Action Sub-Stack	[RFC9789]
NASI	Network Action Sub-Stack Indicator	This document
NASL	Network Action Sub-Stack Length	This document
OAM	Operations, Administration, and Maintenance	[RFC6291]
RLD	Readable Label Depth	[RFC9789]
TC	Traffic Class	[RFC5462]
TTL	Time To Live	[RFC3032]

Table 1: Abbreviations

3. Overview

The MPLS Network Action Sub-Stack (NAS) is a set of Label Stack Entries (LSEs) that appear as part of an MPLS Label Stack and serve to encode information about the network actions that should be invoked for the packet. Multiple NASes may appear in a label stack.

This document describes how network actions and their optional ancillary data are encoded as part of an NAS as a stack of LSEs. Mechanisms that allow sharing of ancillary data AD between multiple network actions encoded in the same NAS can be described in other documents and do not rely on any explicit provision in the encodings described in this document.

4. Label Stack Entry Formats

The NAS uses a variety of different formats of LSEs for different purposes. This section describes the syntax of the various formats while the overall structure of the NAS and the semantics of the various LSEs are described in the sections below.

4.1. LSE Format A: The MNA Sub-Stack Indicator

LSE Format A is an LSE as described in [RFC3032] and [RFC5462]. The label value is an IANA-assigned value (TBA) for the MNA bSPL label from the "Base Special-Purpose MPLS Label Values" registry to indicate the presence of MNA in the packet and the beginning an MNA Sub-Stack in the label stack.

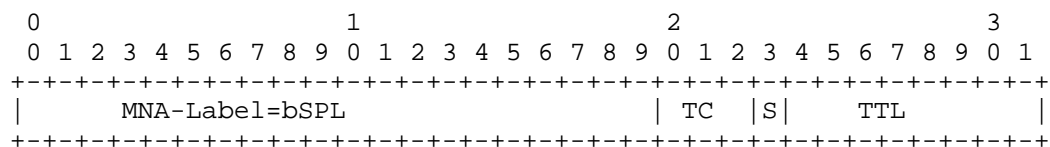


Figure 1: LSE Format A: The MNA Sub-Stack Indicator

- * S (1 bit): The Bottom of Stack [RFC3032]. MUST be set to 0 on transmitted packets. If a packet is received with S bit set to 1, then the packet MUST be dropped.

4.2. LSE Format B: The initial opcode

LSE Format B is used to encode the first opcode in the NAS, plus a number of other fields about the NAS. This LSE cannot carry more than 13 bits of data.

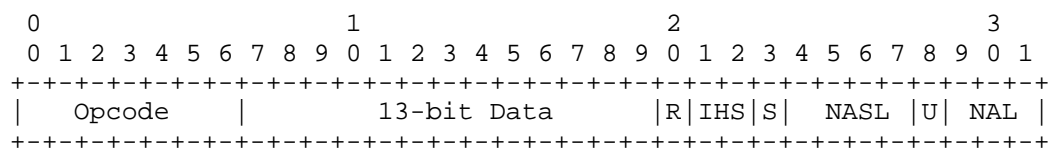


Figure 2: LSE Format B: The initial opcode

- * Opcode (7 bits): The operation code for this LSE. See Section 5.1.
- * Data (13 bits): Opcode-specific data.
- * R (1 bit): Reserved. This bit MUST be transmitted as zero and ignored upon receipt.
- * IHS (2 bits): The scope of the sub-stack. See Section 5.3.
- * S (1 bit): The Bottom of Stack [RFC3032]. If NASL value is non-zero, then S bit MUST be 0. If a packet is received with S bit set to 1 and a non-zero NASL value, then the packet MUST be dropped. The encapsulating node MUST ensure that S bit is set to 1 only in the Last LSE in the MPLS header.
- * NASL (4 bits): The Network Action Sub-Stack Length (NASL). The number of additional LSEs in the sub-stack, not including the leading Format A LSE and the Format B LSE.
- * U (1 bit): Unknown Network Action Handling. See Section 5.4.
- * NAL (3 bits): Network Action Length. The number of LSEs of additional data, encoded in LSE Format D (Section 4.4) following this LSE. The NAL value MUST be less than or equal to the NASL value in Format B, if not the packet MUST be dropped.

NOTE: Format A and B LSEs MUST be present when a Format C or D LSE is to be carried in the NAS.

4.3. LSE Format C: Subsequent opcodes

LSE Format C is used to encode the subsequent opcodes in the NAS.

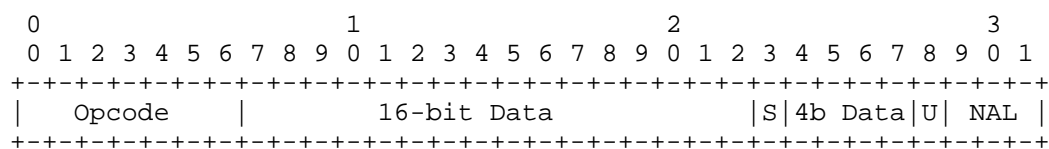


Figure 3: LSE Format C: Subsequent opcodes

- * Opcode (7 bits): The operation code for this LSE. See Section 5.1.
- * Data (16 bits + 4 bits): Opcode specific data
- * S (1 bit): The Bottom of Stack [RFC3032]. If NAL value is non-zero and if S bit is set to 1, then the packet MUST be dropped. If this is not the last LSE in the NAS and if S bit is set to 1 then the packet MUST be dropped. The encapsulating node MUST ensure that S bit is set to 1 only in the Last LSE.
- * U (1 bit): Unknown Network Action Handling. See Section 5.4.
- * NAL (3 bits): Network Action Length. The number of LSEs of additional data, encoded in LSE Format D (Section 4.4) following this LSE. The NAL value MUST be less than or equal to the NASL value in Format B, if not the packet MUST be dropped.

4.4. LSE Format D: Additional Data

LSE Format D is used to encode additional data that did not fit in the LSE with the preceding opcode.

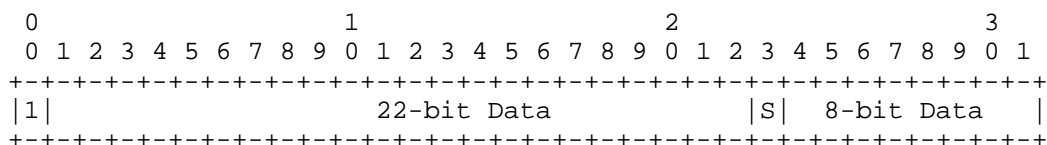


Figure 4: LSE Format D: Additional Data

- * 1 (1 bit): The most significant bit MUST be set. This prevents legacy implementations from misinterpreting this LSE as containing a special purpose label if the data begins with zeros.
- * S (1 bit): The Bottom of Stack [RFC3032]. If this is not the last LSE for the Network Action based on the NAL value and if S bit is set to 1 then the packet MUST be dropped. If this is not the last LSE in the NAS and if S bit is set to 1 then the packet MUST be dropped. The encapsulating node MUST ensure that the S bit is set to 1 only in the Last LSE.
- * Data (22 bits + 8 bits): Opcode specific data

5. The MNA Sub-Stack

The MNA Sub-Stack **MUST** begin with a Format A LSE (Section 4.1). The label value of the LSE contains the MNA bSPL (value TBA) to indicate the presence of the MNA Sub-Stack.

The TC and TTL values of the Format A LSE retain their semantics as defined in [RFC3032] and [RFC5462]. The TTL and TC values in the Format A LSE are copied from the forwarding label at the top of the label stack. The penultimate node on the path copies the TTL and TC values from the preceding LSE to the next LSE on the label stack, overwriting the TTL and TC values of the next LSE, as specified in Section 3.5 of [RFC3443] and Section 2.6.3 of [RFC3270] in the Uniform Mode LSPs. If the node performing this copy is not aware of MNA, this could overwrite the values in the first LSE of the MNA sub-stack.

The second LSE in an NAS **MUST** be a Format B LSE (Section 4.2). This LSE contains an initial opcode plus additional fields that describe the NAS.

The Format B LSE (Section 4.2) could optionally carry additional data in Format D (Section 4.4) LSEs, up to the length encoded in the LSE's NAL value.

An NAS **MAY** contain more Format C (Section 4.3) and Format D (Section 4.4) LSEs, up to the length encoded in the NASL value. All Format D LSEs **MUST** follow a Format C or B LSE and be included in that LSE's NAL value.

5.1. Opcodes

The opcode is a 7-bit field that indicates the semantics of its LSE. Several opcodes are assigned special semantics (Section 6), others act as Network Action Indicators and are assigned through IANA (Section 10 and Section 14.4).

5.2. Data

The data field carries opcode-specific data that is ancillary data for a network action. In the case of opcode 1, the data field carries Flag-Based Network Action Indicators without ancillary data.

Legacy implementations might use the label value (most significant 20 bits) in one or more consecutive LSEs when load-balancing data flows in an ECMP environment. Modifying the first 20 bits in an LSE might alter that packet's path and result in out-of-order delivery of packets. To preserve the stability of the deployed services in ECMP

environments that use information in the label value to load-balance data flows, if a network action encodes data in the given LSE that may be different among packets in the same flow or might change during packet forwarding across the MPLS network, then that data MUST NOT be carried in the most significant 20 bits of a Format B LSE (Section 4.2), a Format C LSE (Section 4.3), or a Format D LSE (Section 4.4). Thus, the available bits for data that can change by a transit node or differ among packets of the same flow in Format A and Format B LSEs are 0, Format C LSE is 7 (bits 20-22 and 25-28) and Format D LSE is 11 (bits 20-22 and 24-31).

Similarly, to preserve service stability, such data also MUST NOT be carried in the most significant 23 bits of these LSEs when the legacy implementation also uses the TC value, in addition to the label value, in all LSEs when computing ECMP decisions.

The available mitigations for these problems are to use additional Format D LSEs to carry the data, or to place the data in Post-Stack Data as described in [RFC9789].

In network deployments where it is known that a load-balancing of data flows is not used, or, otherwise, if only the explicitly signaled entropy value is used, and it is certain that the load-balancing path selection will not be based on the label value of the LSEs, then the data in the label value of the LSEs in ISD MAY be mutable within the data flow without causing the out-of-order delivery of packets.

5.3. Scope

The IHS field in the Format B LSE indicates the scope of the In-stack NAIs encoded in the NAS. Scope defines which nodes along the MPLS path should perform the network actions found within the NAS. The specific values of the IHS field are as follows:

+=====+		
Bits	Scope	
+=====+		
00	I2E	
+-----+		
01	HBH	
+-----+		
10	Select	
+-----+		
11	Reserved	
+-----+		

Table 2: IHS Scope Values

Ingress To Egress (I2E) - The NAS MUST NOT be processed by any node the except the egress node.

Hop-By-Hop (HBH) - All nodes along the path MUST process the NAS.

Select - Only specific nodes along the path that brings NAS to top of the stack will perform the action.

A single NAS carries only one of the three scopes (I2E/HBH/Select). To support multiple scopes for a single packet, multiple NASes MAY be included in a single label stack.

The egress node is included in the HBH scope. This implies that the penultimate node MUST NOT remove a HBH NAS. The egress node MAY receive an NAS at the top of the label stack as discussed in Section 10.

An I2E scope NAS, if present, MUST be encoded after any HBH or Select-scope NASes. This makes it easier for the transit nodes to process a NAS with HBH or Select scope.

5.4. Unknown Network Action Handling

The Unknown Network Action Handling (U) field in a Format B LSE (Section 4.2) and Format C LSE (Section 4.3) is a 1-bit value that defines the action to be taken by a node that does not understand an action within the NAS. The different types of Unknown Network Action Handling actions are defined below.

+=====+	
Bit	Action
+=====+	
0	Skip to the next NA
+-----+	
1	Drop the packet
+-----+	

Table 3: Unknown Network
Action Handling

When a packet with an unknown Network Action is dropped, the node SHOULD maintain a local counter for this event, and MAY send a rate-limited notification to the operator.

5.5. Ordering

The network actions encoded in the NAS MUST be processed as if they were processed in the order that they appear in the NAS, from the top of the NAS to the bottom. NAI encoded as flags (see Section 6.2) MUST be processed as if they were processed from the most significant bit to the least significant bit. If a label stack contains multiple NASes, then they MUST be processed as if they were processed in the order that they appear in the label stack, subject to the restrictions in Section 7.

6. Special Opcodes

Below are the special opcodes used to build a basic In-stack MNA solution. In future, additional special opcodes can be defined and their code-points can be assigned from the "Network Action Opcodes" IANA registry.

6.1. bSPL Protection

Opcode: 0

Purpose: Legacy implementations may scan the label stack looking for bSPL values. As long as the opcode field is non-zero, an LSE cannot be misinterpreted as containing a bSPL. Opcode 0 is therefore reserved and is not used.

6.2. Flag-Based NAIs without AD

Opcode: 1

Purpose: Network actions that do not require Ancillary Data do not require an entire LSE. A single flag can be used to indicate each of these network actions.

LSE Formats: B, C, D

Data: The data field carries Network Action Indicators, which should be evaluated from the most significant bit to the least significant bit. If this opcode is used with LSE Format B only, then up to 13 flags may be carried. If this opcode is used with LSE Format C only, then up to 20 flags may be carried. Format D LSEs can be used with format C LSEs to encode more than 20 flags. Flags are assigned from the "Network Action Flags Without Ancillary Data" registry (Section 14.3). If flags need to be evaluated in a different order, multiple LSEs using this opcode may be used to specify the requested order. The Flag-Based Network Action Indicators MUST follow the procedure for data specified in Section 5.2.

Scope: This opcode can be used with any scope.

6.3. No-Operation Opcode

Opcode: 2

Purpose: This opcode is reserved to indicate that this opcode does not perform any Network Action and MUST be skipped.

Scope: Format B.

6.4. Extension Opcode

Opcode: 127

Purpose: This opcode is reserved to extend the current opcode range beyond 127 in future. If this opcode is not supported, then the packet with the opcode 127 MUST be dropped. Use of this opcode is outside the scope of this document.

7. NAS placement in the Label Stack

The node adding an NAS to the label stack places a copy of the NAS where the relevant nodes can read it. Each downstream node along the path has a Readable Label Depth (RLD) the NAS MUST NOT appear at the top of the stack at any MNA incapable node on the path. If the NAS is to be processed by a downstream MNA-capable node, then the entire NAS MUST be placed so that it is within RLD by the time the packet reaches the downstream MNA-capable node and the NAS MUST NOT appear at the top of the stack at any MNA incapable node on the path. When adding the Entropy Label Identifier (bSPL label 7) and Entropy Label as defined in [RFC6790], along with an NAS, the RLD MUST be considered for the placement of both.

If the label stack is deep, several copies of the NAS may need to be encoded in the label stack.

For an NAS with HBH scope, every node will process the top copy of the NAS.

For an NAS with Select scope, it is processed by the node that brings it to the top of stack (for example, in the case of using MPLS label pop operation in Segment Routing) and then the NAS is removed from the stack. The select-scoped NAS needs to be inserted after the forwarding label and before the next forwarding label. It could be inserted before or after a HBH NAS. Note that the case of an NAS with Select scope with MPLS label swap operation (for example, with RSVP-TE LSPs) is for future study.

For I2E scope, only one copy of the NAS needs to be added at the bottom of the stack.

Transit, non-penultimate nodes that pop a forwarding label and expose a copy of an NAS MUST remove it.

A node performing Penultimate Hop Popping (PHP) that pops the forwarding label with only the NAS(es) remaining on the stack MUST NOT remove the NAS(es). Instead, it forwards the packet with the NAS(es) at the top of stack to the next node. Note that the behavior of the PHP node, as defined in [RFC3270] for TC processing, and as defined in [RFC3443] for TTL processing, is not modified regardless of whether the PHP node supports MNA.

The node that receives the NAS at the top of the label stack MUST remove it.

7.1. Actions when Pushing Labels

An MNA-capable node may need to push additional labels as well as push new network actions onto a received packet.

While pushing additional labels on to the label stack of the receive packet, the MNA-capable node MUST verify that the entire top-most NAS with HBH scope is still within the RLD of the downstream MNA-capable nodes. If required, the MNA-capable node MAY create a copy of the top-most NAS with HBH scope and insert it within the RLD of the downstream MNA-capable nodes on the label stack.

When an MNA-capable node needs to push a new NAS with HBH scope on to a received packet that already has an NAS with HBH scope, it SHOULD copy (and merge) the network actions (including their Ancillary Data) from the received top-most NAS with HBH scope in the new NAS with HBH scope. The new NAS MUST be placed within the RLD of the downstream MNA-capable nodes. This behavior can be based on local policy.

The new network actions added MUST NOT conflict with the network actions in the received NAS with HBH scope. The mechanism to resolve such conflicts depend on the network actions and can be based on local policy. The MNA-capable node that pushes entries MUST understand any network actions which it is pushing which may result in a conflict, and MUST resolve any conflicts between new and received network actions. In the usual case of a conflict of duplicating a network action, the definition of the network action will generally give guidance on likely resolutions.

8. Node Capability Signaling

Encapsulating Node is the node that pushes an NAS on to the Label stack.

The encapsulating node MUST make sure that the NAS can be processed by the transit and egress nodes.

- * The path computation system needs to know the MSD and RLD that can be imposed at the ingress node of a given SR path [RFC8664]. This ensures that the label stack depth of a computed path does not exceed the maximum number of labels (i.e., MSD) the node is capable of imposing and the maximum number of labels that can be read by the MNA-processing nodes in the path. The MSD needs to include the MNA Sub-Stacks to be added.
- * The node responsible for selecting a path through the MPLS network needs to know and consider the MNA-capabilities and RLD of the transit nodes, and the MNA-capabilities of the end point. it supports.
- * Information about the capabilities of the nodes may be configured, collected through management protocols, or distributed by control protocols (such as advertising by routing protocols).
- * The mechanisms by which the capabilities of the nodes are known by the node responsible for selecting a path through the MPLS network are out of scope for this document.

9. Processing the Network Action Sub-Stack

This section defines the specific responsibilities for nodes along an LSP.

9.1. Encapsulating Node Responsibilities

The encapsulating node MAY add NASes to the label stack in accordance with its policies, the placement restrictions in Section 7, and the limitations learned from Section 8.

The encapsulating node MUST NOT add an NAS to the label stack if the egress node does not support MNA.

If there is an existing label stack, the encapsulating node MUST NOT modify the first 20 bits of any LSE in the label stack when the ECMP technique in the network is using the hashing of the labels on the label stack.

If the encapsulating node is also a transit node, then it MUST also follow the rules set out in Section 9.2.

9.2. Transit Node Responsibilities

Transit Node is the node that process an NAS on to the Label stack but does not push any new NAS.

The transit node MUST follow the procedure for data specified in Section 5.2.

Transit nodes MUST process the NASes in the label stack, according to the rules set out in Section 5.5.

A transit node that processes an NAS and does not recognise the value of an opcode MUST follow the rules according to the setting of the Unknown Action Handling value in the NAS as described in (Section 5.4).

9.3. Penultimate Node Responsibilities

In addition to the transit node responsibilities, the penultimate node and penultimate SR-MPLS segment node MUST NOT remove the last copy of an HBH or I2E NAS when it is exposed after removing the forwarding (transport) label. This allows the egress node to process the NAS.

9.4. Egress Node Responsibilities

The egress node MUST remove any NAS it receives.

10. Network Action Indicator Opcode Definition

The following information MUST be defined for new Network Action Indicator opcode request in the document that specifies the Network Action.

A request for a new NAI MUST include the following information:

- * Format: The definition of the new Network Action MUST specify the LSE Formats. The opcode can define Network Action in Format B or C or both B and C. Both Format B and C LSEs MAY optionally carry Format D LSE(s).
- * Scope: The request MUST specify at-least one scope (I2E, HBH, Select) for the Network Action. The request MAY specify more than one scope.

- * Ancillary Data: A request MUST specify the quantity, syntax, and semantics of any associated Ancillary Data. The Ancillary Data MAY be variable length, but the length MUST be computable based on the data present in the NAS.
- * Processing: The request MUST specify the detailed procedure for processing the network action.
- * Interactions: The definition of the new Network Action MUST specify its interaction with other currently defined Network Action if there is any.

An assignment for an NAI MAY make requests from any combination of the "Network Action Opcodes" or "Network Action Flags Without Ancillary Data" assignments. This decision should optimize for eventual encoding efficiency. If the NAI does not require any ancillary data, then a flag is preferred as only one bit is used in the encoding.

11. Backward Compatibility

This section discusses interactions between MNA-capable and legacy, non-MNA-capable nodes.

An MNA-encapsulating node MUST ensure that the MPLS Network Action Sub-Stack indicator is not at the top of the MPLS Label Stack when the packet arrives at a non-MNA-capable node. If such a packet did arrive at a non-MNA-capable node, it will most likely be dropped.

Legacy nodes may scan the label stack, potentially looking for a label value containing a bSPL. To ensure that the LSE formats described herein do not appear to contain a bSPL value, the opcode value of 0 has been reserved. By ensuring that there is a non-zero value in the high order 7 bits, we are assured that the high order 20 bits cannot be misinterpreted as containing a bSPL value (0-15).

The TC and TTL values of the Format A LSE are not re-purposed for encoding, as the penultimate node on the MPLS packet path may propagate TTL from the transport (or forwarding) label to the next label on the label stack, overwriting the TTL on the next label. If the penultimate node is a legacy node, it might perform this action, potentially corrupting other values stored in the TC and TTL values. To protect against this, we retain the TC and TTL values in the Format A LSE.

12. Implementation Status

[Note to the RFC Editor - remove this section before publication, as well as remove the reference to [RFC7942]]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [RFC7942]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

12.1. University of Tuebingen Implementation

The solution defined in the document draft-ietf-mpls-mna-hdr-08 has been implemented using P4 pipeline. The implementation code can be found at <https://github.com/uni-tue-kn/P4-MNA>.

13. Security Considerations

The security considerations in [RFC3032] and [RFC9789] also apply to this document.

In addition, MNA-creates a new dimension in security concerns:

- * The actions of an encapsulating node can affect any or all of the nodes along the path. In the most common and benign situations, such as a syntactically incorrect packet could result in packet loss or corruption.
- * The semantics of a network action are unbounded and may be insecure. A network action could be defined that made arbitrary changes to the memory of the forwarding router, which could then be used by the encapsulating node to compromise every MNA-capable router in the network. The IETF needs to ensure that only secure network actions are defined.
- * The MNA architecture supports locally-defined network actions. For such actions, there will be limited oversight to ensure that the semantics do not create security issues. Implementors and network operators will need to ensure that locally-defined network actions do not compromise the security of the network.

- * The MPLS domain border nodes MUST ensure that the MPLS packets with MNA from any domain with a different administrative control can be filtered to prevent entering the provider MPLS domain. The filtering capability MAY be enabled on a per network action basis and it can be based on a local policy. The filtering capability MUST be implemented on those nodes before deploying MNA in the provider MPLS domain. The RLD on the filtering node MUST be higher than the RLD on all other nodes in the provider MPLS domain.
- * The MNA architecture supports modifying the AD on the intermediate nodes, so the critical network functions should either not rely on the data or should be aware of the risks and use other means to verify the security of the whole network.
- * The "private Use" opcodes in "Network Action Opcodes" Section 14.4 and "Network Action Flags Without Ancillary Data" Section 14.3 Registry are subject to the considerations described in [RFC8126].

14. IANA Considerations

14.1. MNA bSPL Label

This document requests that IANA allocate a value (TBA) for the MNA bSPL label from the "Base Special-Purpose MPLS Label Values" registry to indicate the presence of an MNA Sub-Stack in the label stack. The description of the value should be "MPLS Network Actions". The reference should be this document.

14.2. MPLS Network Actions Parameters

This document requests that IANA create a new category called "MPLS Network Actions Parameters" within the "Multiprotocol Label Switching Architecture (MPLS)" category. The registries described below should belong to this new category.

14.3. Network Action Flags Without Ancillary Data

This document requests that IANA create a new registry with the name "Network Action Flags Without Ancillary Data". Registration requests should comply with Section 10. The registration procedure for this registry is "IETF Review", "Experimental Use" and "Private Use" as defined in [RFC8126]. The fields in this registry are "Bit Position" (integer), "Description" (string), and "Reference" (string).

Bit Position refers to the position relative to the most significant bit in LSE Format B or C Data fields and any subsequent Format D LSEs. Bit Position 0 is the most significant bit in an LSE Format B

or C Data field. Bit Position 20 is the most significant bit in the first LSE Format D Data field. There are 20 bits available in LSE Format C and 30 bits available in LSE Format D. There are at most 14 Format D LSEs per opcode (due to NASL limit of 15 and Format D requires Format C LSE), so there are at most $20 + 14 * 30 = 440$ bit positions. The Bit Position is an integer with value 0-469.

The initial assignments for this registry are:

Bit Position	Description	Reference
0-14	IETF Review	This document
15-16	Experimental Use	This document
17-19	Private Use	This document
20-469	IETF Review	This document

Table 4: Network Action Flags Without Ancillary Data Registry

14.4. Network Action Opcodes

This document requests that IANA create a new registry with the name "Network Action Opcodes". Registration requests should comply with Section 10. The registration procedure for this registry is "IETF Review", "Experimental Use" and "Private Use" as defined in [RFC8126]. The fields are "Opcode" (integer), "Description" (string), and "Reference" (string). Opcode is an integer with value 1-126.

Opcode	Description	Reference
1-110	IETF Review	This document
111-114	Experimental Use	This document
115-126	Private Use	This document

Table 5: Network Action Opcodes

IANA has allocated values for the following Network Action Opcodes from "Network Action Opcodes".

Opcode	Description	Reference
0	Reserved	This document
1	Flag-Based Network Action Indicators without AD	This document
2	No operation Opcode	This document
127	Opcode Range Extension Beyond 127	This document

Table 6: Network Action Opcodes

15. Examples

15.1. Network Action Encoding Examples

15.1.1. Network Action Flags without AD

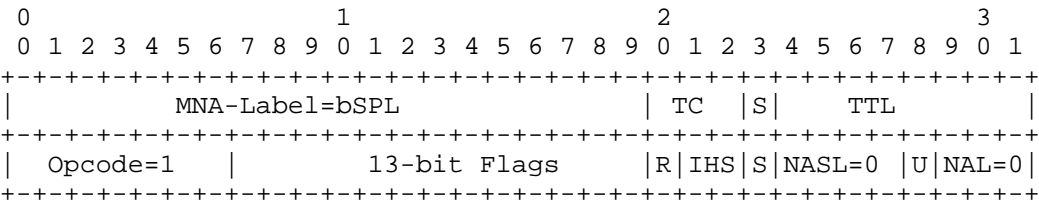


Figure 5: NAS with Network Action Flags

This is an example of an NAS with Flag-Based NAIs without Ancillary Data.

Details:

Opcode=1: This opcode to indicates that the LSE carries Flag-Based NAIs without AD.

Data: The data field carries the Flag-Based NAIs.

S: This is the bottom of stack bit. Set if and only if this LSE is the bottom of the stack.

U: Action to be taken if one of the NAIs are not recognized by the processing node.

NASL: The NASL value is set to 0, as there are no additional LSEs.

NAL: The NAL value is set to 0, as there are no additional AD encoded using Format D.

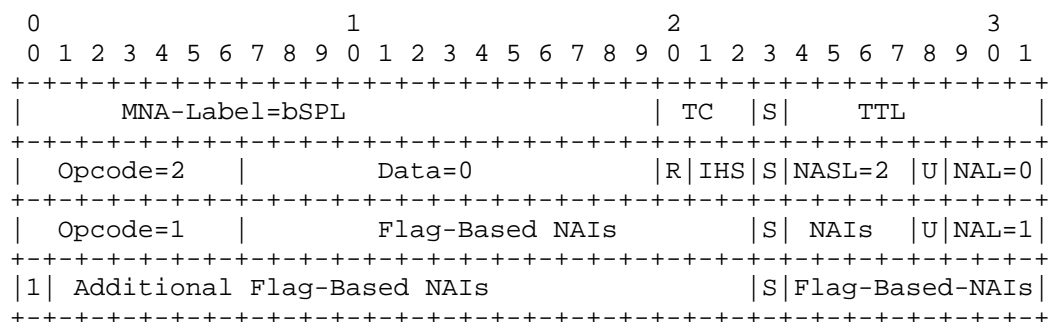


Figure 6: Network Action Flags without AD using LSE Format D

In this example, the NAS contains a Format B LSE with No-Operation Opcode value 2. The next LSE uses Format C, but the Network Action Flag is not in a bit position contained within the Format C LSE, so a single Format D LSE has been added to the NAS to carry the flag.

NAL is set to 1 to indicate that Flag-Based NAIs are also encoded in the next LSE.

NASL is set to 2 to indicate that 2 additional LSEs are used.

15.1.2. Network Action Opcode with AD

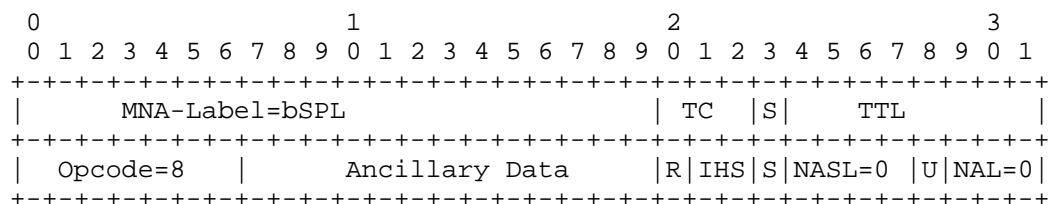


Figure 7: Network action opcode with Ancillary Data

In this example, the NAS is carrying only one Network Action that requires 13 bits of Ancillary Data.

Details on the Second LSE

Opcode=8: A network action allocation is outside of this document.

Data: The data field contains 13 bits of ancillary data.

15.1.3. Network Action Opcode with more AD with Format-B

A network action may require more Ancillary Data than can fit in a single LSE. In this example, a Format D LSE is added to carry additional Ancillary Data.

```

0           1           2           3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|           MNA-Label=bSPL           | TC |S|           TTL           |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Opcode=10 |           Ancillary Data           |R|IHS|S|NASL=1 |U|NAL=1|
+-----+-----+-----+-----+-----+-----+-----+-----+
|1|           Ancillary Data           |S|Ancillary Data |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Figure 8: Network Action With Additional Ancillary Data

In this example, opcode 10 is encoded in Format B and it requires more than one LSE's worth of Ancillary Data, so a Format D LSE is added.

Details on the second LSE:

Opcode=10: An opcode allocation is outside of this document

Ancillary Data: Ancillary data required to process the Network Action opcode 10

NAL: Length of additional LSEs used to encode its Ancillary data

Details on the third LSE:

Ancillary Data: 22 bits of additional Ancillary data.

Ancillary Data: 8 bits of additional Ancillary Data.

15.1.4. Network Action Opcode with more AD with Format C

A network action may require more Ancillary Data than can fit in a single LSE. In this example, a Format D LSE is added to carry additional Ancillary Data.

```

      0               1               2               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|           MNA-Label=bSPL                    | TC | S |      TTL      |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Opcode=2   |           Data=0                    | R | IHS | S | NASL=2 | U | NAL=0 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Opcode=9   |           Ancillary Data                    | S |   AD   | U | NAL=1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 |           Ancillary Data                    | S | Ancillary Data |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Figure 9: Network Action With Additional Ancillary Data

In this example, opcode 9 requires more than one LSE's worth of Ancillary Data, so a Format D LSE is added.

Details on the third LSE:

Opcode=9: An opcode allocation is outside of this document

Ancillary Data: Most significant bits of Ancillary data

AD: 4 bits of additional Ancillary Data

Details on the fourth LSE:

Ancillary Data: 22 bits of additional Ancillary data.

Ancillary Data: 8 bits of additional Ancillary Data.

15.2. Network Action Processing Order

The semantics of a network action can vary widely and the results of processing one network action may affect the processing of a subsequent network action. See Section 5.5.

15.2.1. Network Action Processing Order

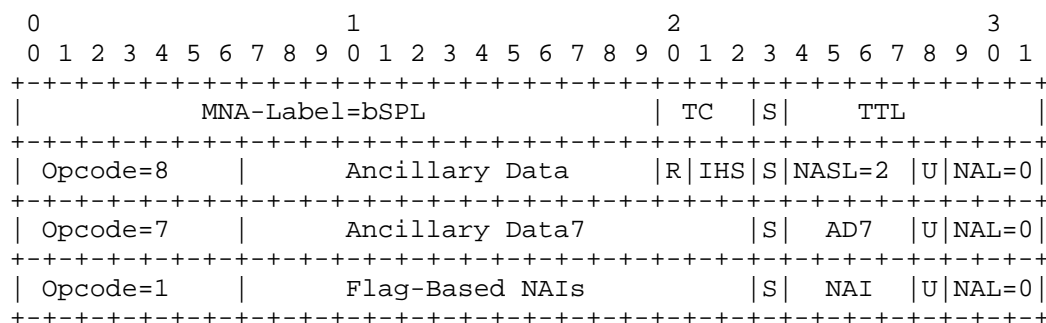


Figure 10: In-stack NA processing order

In this example, opcode 8 is processed first, then opcode 7, and then the network action flags are processed from most significant to least significant.

In a different case, some Flag-Based NAIs may need to be processed before opcode 7 and some Flag-Based NAIs may need to be processed after Opcode 7. This can be done by causing some NAIs to appear earlier in the NAS.

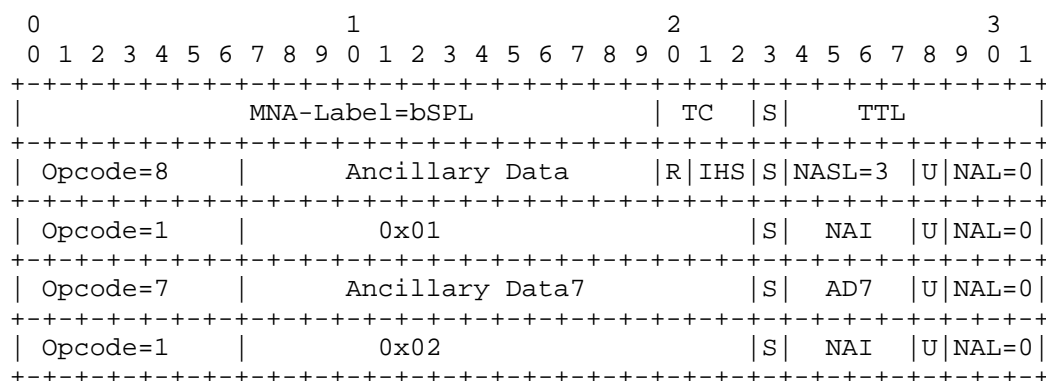


Figure 11: Interleaving network actions

In the above example, opcode 8 is processed first, then Flag-Based NAI 0x01 is processed, then opcode 7 is processed, and finally NAI 0x02 is processed.

16. References

16.1. 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>>.
- [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>>.
- [RFC3443] Agarwal, P. and B. Akyol, "Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks", RFC 3443, DOI 10.17487/RFC3443, January 2003, <<https://www.rfc-editor.org/info/rfc3443>>.
- [RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <<https://www.rfc-editor.org/info/rfc5462>>.
- [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>>.
- [RFC9017] Andersson, L., Kompella, K., and A. Farrel, "Special-Purpose Label Terminology", RFC 9017, DOI 10.17487/RFC9017, April 2021, <<https://www.rfc-editor.org/info/rfc9017>>.
- [RFC9613] Bocci, M., Ed., Bryant, S., and J. Drake, "Requirements for Solutions that Support MPLS Network Actions (MNAs)", RFC 9613, DOI 10.17487/RFC9613, August 2024, <<https://www.rfc-editor.org/info/rfc9613>>.
- [RFC9789] Andersson, L., Bryant, S., Bocci, M., and T. Li, "MPLS Network Actions (MNAs) Framework", RFC 9789, DOI 10.17487/RFC9789, July 2025, <<https://www.rfc-editor.org/info/rfc9789>>.

16.2. Informative References

- [RFC3270] Le Faucheur, F., Ed., Wu, L., Davie, B., Davari, S., Vaananen, P., Krishnan, R., Cheval, P., and J. Heinanen, "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services", RFC 3270, DOI 10.17487/RFC3270, May 2002, <<https://www.rfc-editor.org/info/rfc3270>>.

- [RFC6291] Andersson, L., van Helvoort, H., Bonica, R., Romascanu, D., and S. Mansfield, "Guidelines for the Use of the "OAM" Acronym in the IETF", BCP 161, RFC 6291, DOI 10.17487/RFC6291, June 2011, <<https://www.rfc-editor.org/info/rfc6291>>.
- [RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <<https://www.rfc-editor.org/info/rfc7942>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [RFC8664] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing", RFC 8664, DOI 10.17487/RFC8664, December 2019, <<https://www.rfc-editor.org/info/rfc8664>>.
- [RFC9791] Saad, T., Makhijani, K., Song, H., and G. Mirsky, "Use Cases for MPLS Network Action Indicators and Ancillary Data", RFC 9791, DOI 10.17487/RFC9791, July 2025, <<https://www.rfc-editor.org/info/rfc9791>>.

Acknowledgments

The authors of this document would like to thank the MPLS Working Group Open Design Team for the discussions and comments on this document. The authors would also like to thank Amanda Baber for reviewing the IANA Considerations and providing many useful suggestions. The authors would like to thank Loa Andersson, Stewart Bryant, Greg Mirsky, Joel M. Halpern and Adrian Farrel for reviewing this document and providing many useful suggestions. The authors would like to thank Fabian Ihle and Michael Menth, both from University of Tuebingen, for reviewing and implementing the solution defined in this document in P4 pipeline.

Contributors

The following people have substantially contributed to this document:

Jisu Bhattacharya
Cisco Systems, Inc.
Email: jisu@cisco.com

Bruno Decraene
Orange
Email: bruno.decraene@orange.com

Weiqiang Cheng
China Mobile
Email: chengweiqiang@chinamobile.com

Xiao Min
ZTE Corp.
Email: xiao.min2@zte.com.cn

Luay Jalil
Verizon
Email: luay.jalil@verizon.com

Jie Dong
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China
Email: jie.dong@huawei.com

Tianran Zhou
Huawei Technologies
China
Email: zhoutianran@huawei.com

Bin Wen
Comcast
Email: Bin_Wen@cable.comcast.com

Sami Boutros
Ciena
Email: sboutros@ciena.com

Tony Li
Juniper Networks
United States
Email: tony.li@tony.li

John Drake
Juniper Networks
United States
Email: jdrake@juniper.net

Figure 12

Authors' Addresses

Jaganbabu Rajamanickam (editor)
Cisco Systems, Inc.
Canada
Email: jrajaman@cisco.com

Rakesh Gandhi (editor)
Cisco Systems, Inc.
Canada
Email: rgandhi@cisco.com

Royi Zigler
Broadcom
Email: royi.zigler@broadcom.com

Haoyu Song
Futurewei Technologies
Email: haoyu.song@futurewei.com

Kireeti Kompella
Juniper Networks
United States
Email: kireeti.ietf@gmail.com