Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: July 6, 2018

X. Xu
Huawei
H. Assarpour
Broadcom
S. Ma
Juniper

January 2, 2018

MPLS Payload Protocol Identifier
draft-xu-mpls-payload-protocol-identifier-04

Abstract

The MPLS label stack has no explicit protocol identifier field to indicate the protocol type of the MPLS payload. This document proposes a mechanism for containing a protocol identifier field within the MPLS packet, which is useful for any new encapsulation header which may need to be encapsulated with an MPLS header.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The MPLS label stack has no explicit protocol identifier field to indicate the protocol type of the MPLS payload. This document proposes a mechanism for containing a protocol identifier field within the MPLS packet, which is useful for any new encapsulation header which may need to be encapsulated with an MPLS header. With this explicit protocol identifier field, there is no need any more for each new encapsulation header to deal with the notorious first nibble issue associated with MPLS individually. More specifically, there is no need to intentionally avoid the first nibble of each new encapsulation header from being 0100 (IPv4) or 0110 (IPv6) and even worsely misuse the first nibble of each new encapsulation header as an MPLS payload type field (e.g., MPLS-BIER [I-D.ietf-bier-mpls-encapsulation]). The tacit permission of misusing the first nibble of each new encapsulation header as an MPLS payload type field would exhause the valuable nibble space quickly. Furthermore, there is no need to insert one additional label indicating the MPLS payload type when transporting any new encapsulation header over MPLS LSPs (e.g., transporting Network Service Header (NSH) [I-D.ietf-sfc-nsh] over MPLS LSPs) therefore the signalling for that additional label is not needed anymore.

To some extent, this situation is much similar to that of the MPLS reserved label space (a.k.a., the special purpose label space) [RFC7274] . Due to the concern over the scarcity of the special-purpose label space , the extended special purpose label concept is introduced accordingly. Similarily, the IETF MPLS community should take precautions on the the scarcity of the first nibble of the MPLS payload before it is too late.

1.1. 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 RFC 2119 [RFC2119].

2. Terminology

This memo makes use of the terms defined in [RFC3032].

3. Protocol Type Field

The encapsulation format for Protocol Type field is depicted as below:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  PIL                  | EXP |1|      TTL      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0|       Reserved        |         Protocol Type         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              Payload                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1

Protocol Identifier Label (PIL): This field contains a special purpose label with value of <TBD> or an extended special purpose label [RFC7274] with value of <TBD> which indicates that a Protocol Type field appears immediately after the bottom of the label stack.

EXP: The usage of this field is in accordance with the current MPLS specification [RFC3032].

S: The Bottom of Stack (BoS) field is set since the PIL MUST always appear at the bottom of the label stack.

TTL: The usage of this field is in accordance with the current MPLS specification [RFC3032].

Reserved MUST be set to 0 and ignored on reception.

Protocol Type: This field indicates the protocol type of the MPLS payload as per [ETYPES].

Payload: This field contains the MPLS payload which can be an IP packet, an Ethernet frame, or any other type of payload, e.g., Network Service Header (NSH) [I-D.ietf-sfc-nsh].

4. Data Plane Processing of PIL

4.1. Egress LSRs

Suppose egress LSR Y is capable of processing the Protocol Type field contained in MPLS packets. LSR Y indicates this to all ingress LSRs via signaling (see Section 5). LSR Y MUST be prepared to deal with both packets with an imposed Protocol Type field and those without; the PIL will distinguish these cases. If a particular ingress LSR chooses not to impose a Protocol Type field, LSR Y's processing of the received label stack (which might be empty) is as if LSR Y chose not to accept Protocol Type field. If an ingress LSR X chooses to impose the Protocol Type field, then LSR Y will receive an MPLS packet constructed as follows: <Top Label (TL), Application Label (AL), PIL> <Protocol Type field> <remaining MPLS payload>. Note that here the TL could be replaced with an IP-based tunnel [RFC4023] and the AL is optional. LSR Y recognizes TL as the label it distributed to its upstream LSR and pops the TL (note that the TL may be an implicit null label, in which case it doesn't appear in the label stack and LSR Y MUST process the packet starting with the AL label (if present) and/or the PIL.) LSR Y recognizes the PIL with S bit set. LSR Y then processes the Protocol Type field, which will determine how LSR Y processes the MPLS payload.

4.2. Ingress LSRs

If an egress LSR Y indicates via signaling that it can process the Protocol Type field, an ingress LSR X can choose whether or not to insert it into the MPLS packet destined for LSR Y. The ingress LSR X MUST NOT insert the Protocol Type field into that MPLS packet unless the egress LSR X has explicitly announced that it could process it. The steps that ingress LSR X performs to insert the Protocol Type field are as follows:

1. On an incoming packet, identify the application to which the packet belongs and determine whether the Protocol Type field needs to be added to the incoming packet.

2. For packets requiring the insertion of the Protocol Type field, prepend the Protocol Type field to the existing MPLS payload; then, push the PIL on to the label stack with the S bit set.

3. Push the application label (AL) label (if required) on to the label stack.

4. Push the EL and the ELI labels [RFC6790] on to the label stack (if required).

5. Determine the top label (TL) and push it on to the label stack.

6. Determine the output interface and send the packet out.

4.3. Transit LSRs

Transit LSRs MAY operate with no change in forwarding behavior. If a transit LSR recognizes the PIL and the subsequent Protocol Type field, it MAY be allowed to do some additional value-added processing, such as MPLS payload inspection, on the received MPLS packet containing the PIL and the Protocol Type field.

4.4. Penultimate Hop LSRs

No change is needed at penultimate hop LSRs.

5. Signaling for PIL Processing Capability

TBD.

6. Alternative Approaches

As illustrated in Section 3 and Section 4, the existence of the Protocol Type field immediately after the MPLS label stack is indicated by inserting the PIL into an MPLS packet. Alternatively, by setting the first nibble of the 4-octet entry containing the Protocol Type field to a dedicated value (e.g., 1111), the existence of the Protocol Type field could be indicated as well (see Figure 2). In this way, there is no need to insert additional label(s) (i.e., the PIL) into an MPLS packet. As for which approach should be selected in the end, it depends on a wide-scope discussion within the IETF.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Bottom Label             | EXP |1|      TTL      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1 1 1 1|       Reserved         |        Protocol Type         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              Payload                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2

7. Acknowledgements

TBD.

8. IANA Considerations

A special purpose label with value of <TBD> or an extended special purpose label with value of <TBD> for the PIL needs to be assigned by the IANA

9. Security Considerations

TBD.

10. References

10.1. Normative References

[ETYPES]   The IEEE Registration Authority, "IEEE 802 Numbers", 2012.

[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>.

10.2. Informative References

[I-D.ietf-bier-mpls-encapsulation]
Wijnands, I., Rosen, E., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication in MPLS and non-MPLS Networks", draft-ietf-bier-mpls-encapsulation-12 (work in progress), October 2017.

[I-D.ietf-sfc-nsh]
Quinn, P., Elzur, U., and C. Pignataro, "Network Service Header (NSH)", draft-ietf-sfc-nsh-28 (work in progress), November 2017.

[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>.

[RFC4023] Worster, T., Rekhter, Y., and E. Rosen, Ed., "Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)", RFC 4023, DOI 10.17487/RFC4023, March 2005, <https://www.rfc-editor.org/info/rfc4023>.

[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>.

[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>.

Authors' Addresses

Xiaohu Xu
Huawei

Email: xuxh.mail@gmail.com

Hamid Assarpour
Broadcom

Email: hamid.assarpour@broadcom.com

Shaowen Ma
Juniper

Email: mashaowen@gmail.com