TEAS Working Group
Internet-Draft
Updates: 3473 (if approved)
Intended status: Standards Track
Expires: February 12, 2018

X. Zhang, Ed.
Huawei Technologies
V. Beeram, Ed.
Juniper Networks
I. Bryskin
Huawei Technologies
D. Ceccarelli
Ericsson
O. Gonzalez de Dios
Telefonica

August 11, 2017

Network Assigned Upstream-Label
draft-ietf-teas-network-assigned-upstream-label-08

Abstract

This document discusses a Generalized Multi-Protocol Label Switching (GMPLS) Resource reSerVation Protocol with Traffic Engineering (RSVP-TE) mechanism that enables the network to assign an upstream label for a bidirectional Label Switched Path (LSP). This is useful in scenarios where a given node does not have sufficient information to assign the correct upstream label on its own and needs to rely on the downstream node to pick an appropriate label. This document updates RFC3473.

Table of Contents

1. Introduction

The Generalized Multi-Protocol Label Switching (GMPLS) Resource reSerVation Protocol with Traffic Engineering (RSVP-TE) extensions for setting up a bidirectional Label Switched Path (LSP) are specified in [RFC3473]. The bidirectional LSP setup is indicated by the presence of an UPSTREAM_LABEL Object in the PATH message. As per the existing setup procedure outlined for a bidirectional LSP, each upstream node must allocate a valid upstream label on the outgoing interface before sending the initial PATH message downstream. However, there are certain scenarios where it is not desirable or possible for a given node to pick the upstream label on its own. This document defines the protocol mechanism to be used in such scenarios. This mechanism enables a given node to offload the task

of assigning the upstream label for a given bidirectional LSP to nodes downstream in the network. It is meant to be used only for bidirectional LSPs that assign symmetric labels at each hop along the path of the LSP. This document updates [RFC3473] as it defines processing for a special label value.

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 [RFC2119].

2. Unassigned Upstream Label

This document proposes the use of a special label value - "0xFFFFFFFF" (for a 4-octet label) - to indicate an Unassigned Upstream Label. Similar "all-ones" patterns are expected to be used for labels of other sizes. The presence of this value in the UPSTREAM_LABEL object of a PATH message indicates that the upstream node has not assigned an upstream label on its own and has requested the downstream node to provide a label that it can use in both the forward and reverse directions. The presence of this value in the UPSTREAM_LABEL object of a PATH message MUST also be interpreted by the receiving node as a request to mandate symmetric labels for the LSP.

2.1. Processing Rules

The Unassigned Upstream Label is used by an upstream node when it is not in a position to pick the upstream label on its own. In such a scenario, the upstream node sends a PATH message downstream with an Unassigned Upstream Label and requests the downstream node to provide a symmetric label. If the upstream node desires to make the downstream node aware of its limitations with respect to label selection, it MUST specify a list of valid labels via the LABEL_SET object as specified in [RFC3473].

In response, the downstream node picks an appropriate symmetric label and sends it via the LABEL object in the RESV message. The upstream node would then start using this symmetric label for both directions of the LSP. If the downstream node cannot pick the symmetric label, it MUST issue a PATH-ERR message with a "Routing Problem/Unacceptable Label Value" indication.

The upstream node will continue to signal the Unassigned Upstream Label in the PATH message even after it receives an appropriate symmetric label in the RESV message. This is done to make sure that the downstream node would pick a different symmetric label if and

when it needs to change the label at a later time. If the upstream node receives an unacceptable changed label, then the error procedure defined in [RFC3473] MUST be followed.

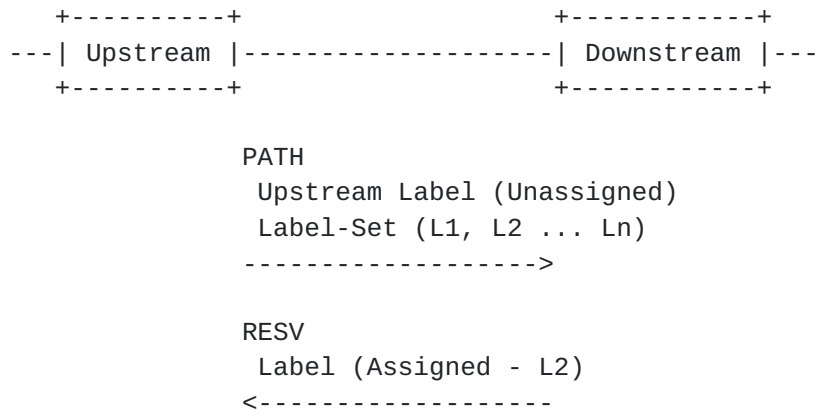


Figure 1

2.2. Backwards Compatibility

If the downstream node is running an implementation that doesn't support the semantics of an Unassigned UPSTREAM LABEL, it will either (a) reject the special label value and generate an error as specified in Section 3.1 of [RFC3473] or (b) accept it and treat it as a valid label.

If the behavior that is exhibited is (a), then there are obviously no backwards compatibility concerns. If there is some existing implementation that exhibits the behavior in (b), then there could be some potential issues. However, at the time of publication, there is no documented evidence of any existing implementation that uses the "all-ones" bit pattern as a valid label. Thus, it is safe to assume that the behavior in (b) will never be exhibited.

3. Use-Case: Wavelength Setup for IP over Optical Networks

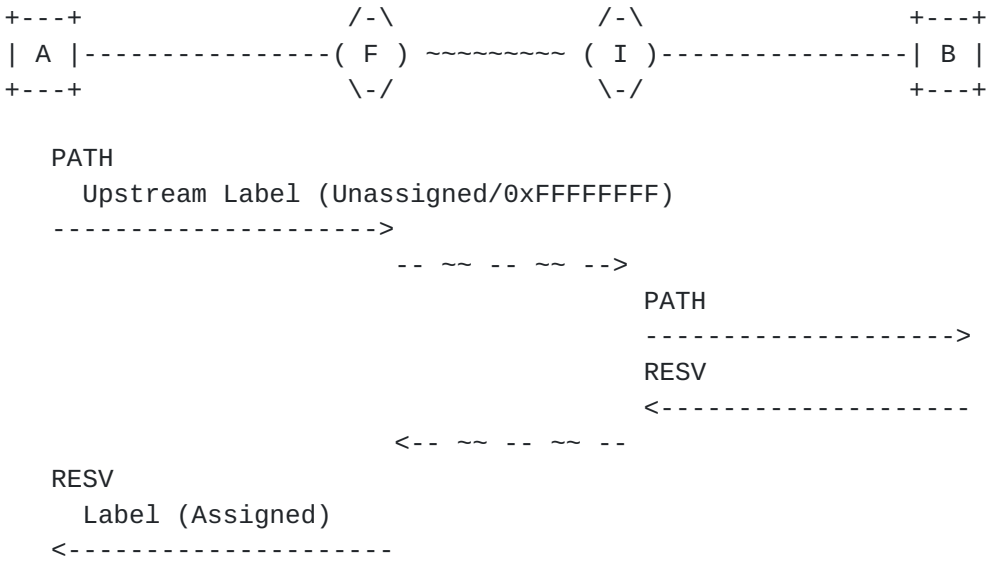
Consider the network topology depicted in Figure 2. Nodes A and B are client IP routers that are connected to an optical Wavelength Division Multiplexing (WDM) transport network. F and I represent WDM nodes. The transponder sits on the router and is directly connected to the add-drop port on a WDM node.

The optical signal originating on "Router A" is tuned to a particular wavelength. On "WDM-Node F", it gets multiplexed with optical

signals at other wavelengths. Depending on the implementation of this multiplexing function, it may not be acceptable to have the router send the signal into the optical network unless it is at the appropriate wavelength. In other words, having the router send signals with a wrong wavelength may adversely impact existing optical trails. If the clients do not have full visibility into the optical network, they are not in a position to pick the correct wavelength in advance.

The rest of this section examines how the protocol mechanism proposed in this document allows the optical network to select and communicate the correct wavelength to its clients.

3.1. Initial Setup



Initial Setup Sequence

Figure 2

Steps:

o "Router A" does not have enough information to pick an appropriate client wavelength. It sends a PATH message downstream requesting the network to assign an appropriate symmetric label for its use. Since the client wavelength is unknown, the laser is off at the ingress client.

- The downstream node (Node F) receives the PATH message, chooses the appropriate wavelength values and forwards them in appropriate label fields to the egress client ("Router B").

- "Router B" receives the PATH message, turns the laser ON and tunes it to the appropriate wavelength (received in the UPSTREAM_LABEL/ LABEL_SET of the PATH) and sends a RESV message upstream.

- The RESV message received by the ingress client carries a valid symmetric label in the LABEL object. "Router A" turns on the laser and tunes it to the wavelength specified in the network assigned symmetric LABEL.

For cases where the egress-node relies on RSVP signaling to determine exactly when to start using the LSP, implementations may choose to integrate the above sequence with any of the existing graceful setup procedures:

- "RESV-CONF" setup procedure ([RFC2205])

- 2-step "ADMIN STATUS" based setup procedure ("A" bit set in the first step; "A" bit cleared when the LSP is ready for use). ([RFC3473])

3.2. Wavelength Change

After the LSP is set up, the network may decide to change the wavelength for the given LSP. This could be for a variety of reasons including policy reasons, restoration within the core, preemption, etc.

In such a scenario, if the ingress client receives a changed label via the LABEL object in a modified RESV message, it retunes the laser at the ingress to the new wavelength. Similarly, if the egress client receives a changed label via UPSTREAM_LABEL/LABEL_SET in a modified PATH message, it retunes the laser at the egress to the new wavelength.

4. Acknowledgements

The authors would like to thank Adrian Farrel and Chris Bowers for their inputs.

5. Contributors

John Drake
Juniper Networks
Email: jdrake@juniper.net

Gert Grammel
Juniper Networks
Email: ggrammel@juniper.net

Pawel Brzozowski
ADVA Optical Networking
Email: pbrzozowski@advaoptical.com

Zafar Ali
Cisco Systems, Inc.
Email: zali@cisco.com

6. IANA Considerations

This document makes no requests for IANA action.

7. Security Considerations

This document defines a special label value to be carried in the UPSTREAM_LABEL object of a PATH message. This special label value is used to enable the function of requesting network assignment of an upstream label. The changes proposed in this document pertain to the semantics of a specific field in an existing RSVP object and the corresponding procedures. Thus, there are no new security implications raised by this document and the security considerations discussed by [RFC3473] still apply.

For a general discussion on MPLS and GMPLS related security issues, see the MPLS/GMPLS security framework [RFC5920].

8. References

8.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <http://www.rfc-editor.org/info/rfc2205>.

[RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, DOI 10.17487/RFC3473, January 2003, <http://www.rfc-editor.org/info/rfc3473>.

8.2. Informative References

[RFC5920] Fang, L., Ed., "Security Framework for MPLS and GMPLS Networks", RFC 5920, DOI 10.17487/RFC5920, July 2010, <http://www.rfc-editor.org/info/rfc5920>.

Authors' Addresses

Xian Zhang (editor)
Huawei Technologies

Email: zhang.xian@huawei.com

Vishnu Pavan Beeram (editor)
Juniper Networks

Email: vbeeram@juniper.net

Igor Bryskin
Huawei Technologies

Email: igor.bryskin@huawei.com

Daniele Ceccarelli
Ericsson

Email: daniele.ceccarelli@ericsson.com

Oscar Gonzalez de Dios
Telefonica

Email: ogondio@tid.es