Network Working Group A. Zamfir
Internet Draft Z. Ali
Intended status: Experimental Cisco Systems
Expires: October 26, 2011 D. Papadimitriou
Alcatel-Lucent
April 27, 2011

Component Link Recording and Resource Control for TE Links

draft-ietf-mpls-explicit-resource-control-bundle-10.txt

Status of this Memo

Abstract

Record Route is a useful administrative tool that has been used extensively by the service providers. However, when TE links are bundled, identification of label resource in Record Route object (RRO) is not sufficient to determine the component link within a TE link that is being used by a given LSP. In other words, when link bundling is used, resource recording requires mechanisms to specify the component link identifier, along with the TE link identifier and Label. As it is not possible to record component link in the RRO, this document defines the extensions to RSVP-TE [RFC3209] and [RFC3473] to specify component link identifiers for resource recording purposes.

This document also defines the Explicit Route object (ERO) counterpart of the RRO extension. The ERO extensions are needed to perform explicit label/resource control over bundled TE link. Hence, this document defines the extensions to RSVP-TE [RFC3209] and [RFC3473] to specify component link identifiers for explicit

resource control and recording over TE link bundles.

Table of Contents

1. Introduction

In GMPLS networks [RFC3945] where unbundled (being either Packet-Switching Capable, Layer2-Switching Capable, Time Division Multiplexing or Lambda-Switching Capable) Traffic Engineering (TE) Links are used, one of the types of resources that an LSP originator could control and record are the component links used by non-neighboring nodes on the LSP path. The resource control and recording is done by the use of the EXPLICIT_ROUTE object (ERO) and RECORD_ROUTE object (RRO), respectively.

Link Bundling, introduced in [RFC4201], is used to improve routing scalability by reducing the amount of TE related information that need to be flooded and handled by IGP in a TE network. This is accomplished by aggregating and abstracting the TE Link components.

In some cases the component link selection/recording within a TE link is left as a local decision (ERO and RRO contains only TE links). However there are cases when it is desirable for a non-local (e.g., LSP head-end) node to make this selection. The use of such information has found since so far three main applications (while not excluding others unknown at the time of writing): diagnostic, association of component specific attributes for which the bundled information is too coarse (e.g., Shared Risk Link Groups) and thus blocking SRLG-disjoint LSP establishment, allocation of labels at network edges, and notification in case of failures. The latter is useful when a single TE link interconnects two parts of the network. In case one of its components fails notifying a complete TE link failure leaves the network disconnected. In either case, it is required to know which component link within a bundled TE link has been used for a given LSP. For these cases, the TE Link and the Label currently specified in the ERO/RRO are not enough and the component link needs to be specified along with the label. In the case of bi-directional Label Switched Paths (LSP) both upstream and downstream information may be specified. Therefore, explicit resource control and recording over a bundled TE link also requires ability to specify a component link within the TE link.

Another important assumption of this document is that the identifier space used for component link identification are unique for a given node (following [RFC4201]). The reason stems as follows: most experimental developments started with TE links composed by a single component link and then only bundling was added by grouping them. Component links where thus identified such that they could mimic the behavior of TE link processing. This also justifies the experimental status of this document.

This document defines extensions to and describes the use of RSVP-TE

[RFC3209], [RFC3471], [RFC3473] to specify the component link identifier for resource recording and explicit resource control over TE link bundles. Specifically, in this document, component interface identifier RRO and ERO subobjects are defined to complement their Label RRO and ERO counterparts. Furthermore, procedures for processing component interface identifier RRO and ERO subobjects and how they can co-exist with the Label RRO and ERO subobjects are specified.

Conventions used in this document

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

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].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC 2119 significance.

In this document, the characters ">>" preceding an indented line(s) indicates a compliance requirement statement using the key words listed above. This convention aids reviewers in quickly identifying or finding the explicit compliance requirements of this RFC.

2. Terminology

o) TE Link: Unless specified otherwise, it refers to a bundled Traffic Engineering link as defined in [RFC4201]. Furthermore, the terms TE Link and bundled TE Link are used interchangeably in this document.

o) Component (interface) link: refers (locally) to a link that is part of a bundled TE link as described in RFC4201.

o) Component Interface Identifier: Refers to an ID used to uniquely identify a Component Interface. on a bundled link a combination of <component link identifier, label> is sufficient to unambiguously identify the appropriate resources used by an LSP. The IDs used for component link identification are unique for a given node [RFC4201].

3. LSP Resource Recording

LSP Resource Recording refers to the ability to record the resources used by an LSP.

The procedure for unbundled numbered TE links is described in [RFC3209] and for unbundled unnumbered TE links in [RFC3477]. For the purpose of recording LSP resources used over bundled TE Links, the Component Interface Identifier RRO sub-object is introduced.

3.1. Component Interface Identifier RRO subobject

A new subobject of the Record Route object (RRO) is used to record component interface identifier of a (bundled) TE Link. This subobject has the following format:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|    Type     |     Length    |U| Reserved  (must be zero)    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//   IPv4, IPv6 or unnumbered Component Interface Identifier   //
|                             . . .                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

L: 1 bit

This bit must be set to 0.

Type

Type 10 (TBD): Component Interface identifier IPv4
Type 11 (TBD): Component Interface identifier IPv6
Type 12 (TBD): Component Interface identifier Unnumbered

Length

Component Link Record. & Resource Control for TE Link Bundles

The Length contains the total length of the subobject in bytes, including the Type and Length fields. The Length is 8 bytes for the Component Interface identifier IPv4 and Component Interface identifier Unnumbered types. For Component Interface identifier IPv6 type of sub-object, the length field is 20 bytes.

U: 1 bit

This bit indicates the direction of the component interface.
It is set to 0 for the downstream interface.
It is set to 1 for the upstream interface and is only used for bi-directional LSPs.

3.2. Processing of Component Interface identifier RRO Subobject

If a node desires component link recording, the "Component Link Recording desired" flag (value TBD) should be set in the LSP_ATTRIBUTES object, object that is defined in [RFC5420].

Setting of "Component Link Recording desired" flag is independent of the Label Recording flag in SESSION_ATTRIBUTE object as specified in [RFC3209]. Nevertheless, the following combinations are valid:

1) If both Label and Component Link flags are clear, then neither Labels nor Component Links are recorded.

2) If Label Recording flag is set and Component Link flag is clear, then only Label Recording is performed as defined in [RFC3209].

3) If Label Recording flag is clear and Component Link flag is set, then Component Link Recording is performed as defined in this document.

4) If both Label Recording and Component Link flags are set, then Label Recording is performed as defined in [RFC3209] and also Component Link recording is performed as defined in this document.

In most cases, a node initiates recording for a given LSP by adding the RRO to the Path message. If the node desires Component Link recording and if the outgoing TE link is bundled, then the initial RRO contains the Component Link identifier (numbered or unnumbered) as selected by the sender. As well, the Component Link Recording desired flag is set in the LSP_ATTRIBUTE object. If the node also desires label recording, it sets the Label_Recording flag in the SESSION_ATTRIBUTE object.

When a Path message with the "Component Link Recording desired" flag set is received by an intermediate node, if a new Path message is to be sent for a downstream bundled TE link, the node adds a new Component Link subobject to the RECORD_ROUTE object (RRO) and appends the resulting RRO to the Path message before transmission. Note also that, unlike Labels, Component Link identifiers are always known on receipt of the Path message.

When the destination node of an RSVP session receives a Path message with an RRO and the "Component Link Recording desired" flag set, this indicates that the sender node needs TE route as well as component link recording. The destination node initiates the RRO

process by adding an RRO to Resv messages. The processing mirrors that of the Path messages. The Component Interface Record subobject is pushed onto the RECORD_ROUTE object (RRO) prior to pushing on the node"s IP address. A node MUST NOT push on a Component Interface Record subobject without also pushing on the IP address or unnumbered Interface Id subobject that identifies the TE Link.

When component interfaces are recorded for unidirectional LSPs, the downstream interface is the one identified by the Component Interface subobject. For bi-directional LSPs, component interface RRO subobjects for both downstream and upstream interfaces MUST be included.

4. Signaling Component Interface Identifier in ERO

4.1. Component Interface Identifier ERO subobject

A new oPTIoNAL subobject of the EXPLICIT_ROUTE object (ERO) is used to specify component interface identifier of a bundled TE Link. This Component Interface Identifier subobject has the following format:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|    Type     |     Length    |U|   Reserved (MUST be zero)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//  IPv4, IPv6 or unnumbered Component Interface Identifier    //
|                            . . .                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

L: 1 bit

This bit must be set to 0.

Type

Type 10 (TBD): Component Interface identifier IPv4
Type 11 (TBD): Component Interface identifier IPv6
Type 12 (TBD): Component Interface identifier Unnumbered

Length

The Length contains the total length of the subobject in bytes, including the Type and Length fields. The Length is 8 bytes for the Component Interface identifier types: IPv4 and Component Interface identifier Unnumbered. For Component Interface identifier IPv6 type of sub-object, the length field

is 20 bytes.

U: 1 bit

This bit indicates the direction of the component interface. It is 0 for the downstream interface. It is set to 1 for the upstream interface and is only used for bi-directional LSPs.

4.2. Processing of Component Interface Identifier ERO Subobject

The Component Interface Identifier ERO subobject follows a subobject containing the IP address, or the link identifier [RFC3477], associated with the TE link on which it is to be used. It is used to identify the component of a bundled TE Link.

The following SHOULD result in "Bad EXPLICIT_ROUTE object" error being sent upstream by a node processing an ERO that contains the Component Interface ID sub-object:

o) The first component interface identifier subobject is not preceded by a sub-object containing an IPv4 or IPv6 address, or an interface identifier [RFC3477], associated with a TE link.

o) The Component Interface Identifier ERO subobject follows a subobject that has the L-bit set.

o) on unidirectional LSP setup, there is a Component Interface Identifier ERO subobject with the U-bit set.

o) Two Component Interface Identifier ERO subobjects with the same U-bit values exist.

If a node implements the component interface identifier subobject, it MUST check if it represents a component interface in the bundled TE Link specified in the preceding subobject that contains the IPv4/IPv6 address or interface identifier of the TE Link. If the content of the component interface identifier subobject does not match a component interface in the TE link, a "Bad EXPLICIT_ROUTE object" error SHOULD be reported as "Routing Problem" (error code 24).

If U-bit of the subobject being examined is cleared (0) and the upstream interface specified in this subobject is acceptable, then the value of the upstream component interface is translated locally in the TLV of the IF_ID RSVP_HOP object [RFC3471]. The local decision normally used to select the upstream component link is bypassed except for local translation into the outgoing interface identifier from the received incoming remote interface identifier.

If this interface is not acceptable, a "Bad EXPLICIT_ROUTE object" error SHOULD be reported as "Routing Problem" (error code 24).

If the U-bit of the subobject being examined is set (1), then the value represents the component interface to be used for upstream traffic associated with the bidirectional LSP. Again, if this interface is not acceptable or if the request is not one for a bidirectional LSP, then a "Bad EXPLICIT_ROUTE object" error SHOULD be reported as "Routing Problem" (error code 24). otherwise, the component interface IP address/ identifier is copied into a TLV sub-object as part of the IF_ID RSVP_HOP object. The local decision normally used to select the upstream component link is bypassed except for local translation into the outgoing interface identifier from the received incoming remote interface identifier.

The IF_ID RSVP_HOP object constructed as above MUST be included in the corresponding outgoing Path message.

Note that, associated with a TE Link sub-object in the ERO, either the (remote) upstream component interface or the (remote) downstream component interface or both may be specified. As specified in [RFC4201] there is no relationship between the TE Link type (numbered or unnumbered) and the Link type of any one of its components.

The Component Interface Identifier ERO subobject is optional. Similarly, presence of the Label ERO sub-objects is not mandatory [RFC3471], [RFC3473]. Furthermore, component interface identifier ERO subobject and Label ERO subobject may be included in the ERO independently of each other. one of the following alternatives applies:

o) When both sub-objects are absent, a node may select any appropriate component link within the TE link and any label on the selected component link.

o) When the Label subobject is only present for a bundled link, then the selection of the component link within the bundle is a local decision and the node may select any appropriate component link, which can assume the label specified in the Label ERO.

o) When only the component interface identifier ERO subobject is present, a node MUST select the component interface specified in the ERO and may select any appropriate label value at the specified component link.

o) When both component interface identifier ERO subobject and Label ERO subobject are present, the node MUST select the locally

corresponding component link and the specified label value on that component link. When present, both subobjects may appear in any relative order to each other but they MUST appear after the TE Link subobject that they refer to.

After processing, the component interface identifier subobjects are removed from the ERO.

Inferred from above, the interface subobject should never be the first subobject in a newly received message. If the component interface subobject is the first subobject in a received ERO, then it SHOULD be treated as a "Bad strict node" error.

Note: Information to construct the Component Interface ERO subobject MAY come from the same mean used to populate the label ERO subobject. Procedures by which an LSR at the head-end of an LSP obtains the information needed to construct the Component Interface subobject are outside the scope of this document.

5. Backward Compatibility

The extensions specified in this document do not affect the processing of the RRO, ERO at nodes that do not support them. A node that does not support the Component Interface RRO subobject but that does support Label subobject SHOULD only insert the Label subobject in the RRO as per [RFC3471] and [RFC3473].

A node that receives an ERO that contains a Component Link ID subobject SHOULD send "Bad EXPLICIT_ROUTE object" if it does not implement this subobject.

Per [RFC3209], Section 4.4.5, a non-compliant node that receives an RRO that contains Component Interface Identifier sub-objects should ignore and pass them on. This limits the full applicability of if nodes traversed by the LSP are compliant with the proposed extensions.

6. Security Considerations

An implementation of the extensions described in this document does exposes the component interface identifiers to other nodes in the network. If this is considered confidential information the mechanisms described in [RFC5920] should be considered.

7. IANA Considerations

This document introduces the following RSVP protocol elements:

o) Component Interface Identifier RRO subobject of the RECORD_ROUTE object (RRO):

- . IANA registry: RSVP PARAMETERS
- . Registry Name: Class Names, Class Numbers, and Class Types
- . Reference: [RFC3936]
- . Following subobjects have been added to the existing entry for:

21 RECORD_ROUTE
- . Type 10 (TBD): Component Interface identifier IPv4
- . Type 11 (TBD): Component Interface identifier IPv6
- . Type 12 (TBD): Component Interface identifier Unnumbered

o) Component Interface Identifier subobject of the EXPLICIT_ROUTE object (ERO).

- . IANA registry: RSVP PARAMETERS

. Registry Name: Class Names, Class Numbers, and Class Types
. Reference: [RFC3936]
. Following subobjects have been added to the existing entry for:

20 EXPLICIT_ROUTE
. Type 10 (TBD): Component Interface identifier IPv4
. Type 11 (TBD): Component Interface identifier IPv6
. Type 12 (TBD): Component Interface identifier Unnumbered

o) A new "Component Link Recording desired" flag (value TBD) of the LSP_ATTRIBUTES object [RFC5420]:

. Bit Flag: 0x80
. Name: Local Component Link Recording desired

8. References

8.1. Normative References

[RFC2205] R.Braden, et al., "Resource ReSerVation Protocol (RSVP) Version 1, Functional Specification", RFC 2205, September 1997.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2234] Crocker, D. and overell, P. (Editors), "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, Internet Mail Consortium and Demon Internet Ltd., November 1997.

[RFC3209] D.Awduche, et al., "Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.

[RFC3471] L.Berger, et al., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description", RFC 3471, January 2003.

[RFC3473] L.Berger, et al., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, January 2003.

[RFC3477] K.Kompella, et al., "Signaling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", RFC 3477, January 2003.

[RFC3945] E.Mannie, et al., "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", RFC 3945, October 2004.

[RFC4201] K.Kompella, et al., "Link Bundling in MPLS Traffic Engineering", RFC 4201, January 2003.

[RFC5420] A.Farrel, et al., "Encoding of Attributes for Multiprotocol Label Switching (MPLS) Label Switched Path (LSP) Establishment Using Resource ReserVation Protocol-Traffic Engineering (RSVP-TE)", RFC 5420.

[RFC5920] L.Fang, "Security Framework for MPLS and GMPLS Networks", RFC 5920, July 2010.

9. Acknowledgments

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

10. Copyright Notice

Authors Addresses

Anca Zamfir
Cisco Systems Inc.
Email: ancaz@cisco.com

Zafar Ali
Cisco systems Inc.
Email: zali@cisco.com

Dimitri Papadimitriou
Alcatel-Lucent Bell
Email: dimitri.papadimitriou@alcatel-lucent.com

Network Working Group R. Aggarwal
Internet Draft Juniper Networks
Category: Standards Track
Expiration Date: August 2011

J. L. Le Roux
France Telecom

February 02, 2011

MPLS Upstream Label Assignment for LDP

draft-ietf-mpls-ldp-upstream-10.txt

Status of this Memo

Copyright and License Notice

Abstract

This document describes procedures for distributing upstream-assigned labels for Label Distribution Protocol (LDP). It also describes how these procedures can be used for avoiding branch Label Switching Router (LSR) traffic replication on a LAN for LDP point-to-multipoint (P2MP) Label Switched Paths (LSPs).

Table of Contents

1. Specification of requirements

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. Introduction

This document describes procedures for distributing upstream-assigned labels [RFC5331] for Label Distribution Protocol (LDP) [RFC5036]. These procedures follow the architecture for MPLS Upstream Label Assignment described in [RFC5331].

This document describes extensions to LDP that a Label Switching Router (LSR) can use to advertise to its neighboring LSRs whether the LSR supports upstream label assignment.

This document also describes extensions to LDP to distribute upstream-assigned labels.

The usage of MPLS upstream label assignment using LDP for avoiding branch LSR traffic replication on a LAN for LDP point-to-multipoint (P2MP) Label Switched Paths (LSPs) [MLDP] is also described.

3. LDP Upstream Label Assignment Capability

According to [RFC5331], upstream-assigned label bindings MUST NOT be used unless it is known that a downstream LSR supports them. This implies that there MUST be a mechanism to enable an LSR to advertise to its LDP neighbor LSR(s) its support of upstream-assigned labels.

A new Capability Parameter, the LDP Upstream Label Assignment Capability, is introduced to allow an LDP peer to exchange with its peers, its support of upstream label assignment. This parameter follows the format and procedures for exchanging Capability Parameters defined in [RFC5561].

Following is the format of the LDP Upstream Label Assignment Capability Parameter:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|0| Upstream Lbl Ass Cap(IANA)|      Length (= 1)             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1| Reserved    |
+-+-+-+-+-+-+-+-+
```

If an LSR includes the Upstream Label Assignment Capability in LDP Initialization Messages it implies that the LSR is capable of both distributing upstream-assigned label bindings and receiving upstream-assigned label bindings. The reserved bits MUST be set to zero on transmission and ignored on receipt. The Upstream Label Assignment Capability Parameter MUST be carried only in LDP initialization messages and MUST be ignored if received in LDP Capability messages.

4. Distributing Upstream-Assigned Labels in LDP

An optional LDP TLV, Upstream-Assigned Label Request TLV, is introduced. To request an upstream-assigned label an LDP peer MUST include this TLV in a Label Request message.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0|0| Upstream Ass Lbl Req (TBD)|      Length                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Reserved                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

An optional LDP TLV, Upstream-Assigned Label TLV is introduced to signal an upstream-assigned label. Upstream-Assigned Label TLVs are carried by the messages used to advertise, release and withdraw upstream assigned label mappings.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0|0| Upstream Ass Label (TBD)  |      Length                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Reserved                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Label                                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Label field is a 20-bit label value as specified in [RFC3032] represented as a 20-bit number in a 4 octet field as specified in section 3.4.2.1 of RFC5036 [RFC5036].

4.1. Procedures

Procedures for Label Mapping, Label Request, Label Abort, Label Withdraw and Label Release follow [RFC5036] other than the modifications pointed out in this section.

A LDP LSR MUST NOT distribute the Upstream Assigned Label TLV to a neighboring LSR if the neighboring LSR had not previously advertised the Upstream Label Assignment Capability in its LDP Initialization messages. A LDP LSR MUST NOT send the Upstream Assigned Label Request TLV to a neighboring LSR if the neighboring LSR had not previously advertised the Upstream Label Assignment Capability in its LDP Initialization messages.

As described in [RFC5331] the distribution of upstream-assigned labels is similar to either ordered LSP control or independent LSP control of the downstream assigned labels.

When the label distributed in a Label Mapping message is an upstream-assigned label, the Upstream Assigned Label TLV MUST be included in the Label Mapping message. When an LSR receives a Label Mapping message with an Upstream Assigned Label TLV and it does not recognize the TLV, it MUST generate a Notification message with a status code of "Unknown TLV" [RFC5036]. If it does recognize the TLV but is unable to process the upstream label, it MUST generate a Notification message with a status code of "No Label Resources". If the Label Mapping message was generated in response to a Label Request message, the Label Request message MUST contain an Upstream Assigned Label Request TLV. A LSR that generates an upstream assigned label request to a neighbor LSR, for a given FEC, MUST NOT send a downstream label mapping to the neighbor LSR for that FEC unless it withdraws the upstream-assigned label binding. Similarly if an LSR generates a downstream assigned label request to a neighbor LSR, for a given FEC, it MUST NOT send an upstream label mapping to that LSR for that FEC, unless it aborts the downstream assigned label request.

The Upstream Assigned Label TLV may be optionally included in Label Withdraw and Label Release messages that withdraw/release a particular upstream assigned label binding.

5. LDP Tunnel Identifier Exchange

As described in [RFC5331] an upstream LSR Ru MAY transmit an MPLS packet, the top label of which (L) is upstream-assigned, to a downstream LSR Rd, by encapsulating it in an IP or MPLS tunnel. In this case the fact that L is upstream-assigned is determined by Rd by the tunnel on which the packet is received. There must be a mechanism for Ru to inform Rd that a particular tunnel from Ru to Rd will be used by Ru for transmitting MPLS packets with upstream-assigned MPLS labels.

When LDP is used for upstream label assignment, the Interface ID TLV [RFC3472] is used for signaling the Tunnel Identifier. If Ru uses an IP or MPLS tunnel to transmit MPLS packets with upstream assigned labels to Rd, Ru MUST include the Interface ID TLV in the Label Mapping messages along with the Upstream Assigned Label TLV. The IPv4/v6 Next/Previous Hop Address and the Logical Interface ID fields in the Interface ID TLV SHOULD be set to 0 by the sender and ignored by the receiver. The Length field indicates the total length of the TLV, i.e., 4 + the length of the value field in octets. A value field whose length is not a multiple of four MUST be zero-padded so that the TLV is four- octet aligned.

Hence the IPv4 Interface ID TLV has the following format:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0|0|     Type (0x082d)         |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv4 Next/Previous Hop Address (0)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Logical Interface ID (0)                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Sub-TLVs                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The IPv6 Interface ID TLV has the following format:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0|0|     Type (0x082e)         |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv6 Next/Previous Hop Address (0)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Logical Interface ID (0)                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Sub-TLVs                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

As shown in the above figures the Interface ID TLV carries sub-TLVs. Four new Interface ID sub-TLVs are introduced to support RSVP-TE P2MP LSPs, LDP P2MP LSPs, IP Multicast Tunnels and context labels. The sub-TLV value in the sub-TLV acts as the tunnel identifier.

Following are the sub-TLVs that are introduced:

1. RSVP-TE P2MP LSP TLV. Type = 28 (To be assigned by IANA). Value of the TLV is the RSVP-TE P2MP LSP SESSION Object [RFC4875].

Below is the RSVP-TE P2MP LSP TLV format when carried in the IPv4 Interface ID TLV:

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    Type (0x1c)                |  16                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                         P2MP ID                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  MUST be zero                 |      Tunnel ID                |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       Extended Tunnel ID                      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Below is the RSVP-TE P2MP LSP TLV format when carried in the IPv6 Interface ID TLV:

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    Type (0x1c)                |  28                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                         P2MP ID                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  MUST be zero                 |      Tunnel ID                |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       Extended Tunnel ID                      |
    |                                                               |
    |                              .......                          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

This TLV identifies the RSVP-TE P2MP LSP. It allows Ru to tunnel an "inner" LDP P2MP LSP, the label for which is upstream assigned, over an "outer" RSVP-TE P2MP LSP that has leaves <Rd1...Rdn>. The RSVP-TE P2MP LSP IF_ID TLV allows Ru to signal to <Rd1...Rdn> the binding of the inner LDP P2MP LSP to the outer RSVP-TE P2MP LSP. The control plane signaling between Ru and <Rd1...Rdn> for the inner P2MP LSP uses targeted LDP signaling messages

2. LDP P2MP LSP TLV. Type = 29 (To be assigned by IANA). Value of the TLV is the LDP P2MP FEC as defined in [MLDP] and has to be set as per the procedures in [MLDP]. Here is the format of the LDP P2MP FEC as defined in [MLDP]:

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
         |P2MP Type      |        Address Family         | Address Length|
         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
         ~                       Root Node Address                       ~
         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
         |    Opaque Length              |    Opaque Value ...           |
         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
         ~                                                               ~
         |                                                               |
         |                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
         |                               |
         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Address Family MUST be set to IPv4, the Address Length MUST be set to 4 and the Root Node Address MUST be set to an IPv4 address when the LDP P2MP LSP TLV is carried in the IPv4 Interface ID TLV. The Address Family MUST be set to IPv6, the Address Length MUST be set to 16 and the Root Node Address MUST be set to an IPv6 address when the LDP P2MP LSP TLV is carried in the IPv6 Interface ID TLV.

The TLV value identifies the LDP P2MP LSP. It allows Ru to tunnel an "inner" LDP P2MP LSP, the label for which is upstream assigned, over an "outer" LDP P2MP LSP that has leaves <Rd1...Rdn>. The LDP P2MP LSP IF_ID TLV allows Ru to signal to <Rd1...Rdn> the binding of the inner LDP P2MP LSP to the outer LDP- P2MP LSP. The control plane signaling between Ru and <Rd1...Rdn> for the inner P2MP LSP uses targeted LDP signaling messages

3. IP Multicast Tunnel TLV. Type = 30 (To be assigned by IANA) In this case the TLV value is a <Source Address, Multicast Group Address> tuple. Source Address is the IP address of the root of the tunnel i.e. Ru, and Multicast Group Address is the Multicast Group Address used by the tunnel. The addresses MUST be IPv4 addresses when the IP Multicast Tunnel TLV is included in the IPv4 Interface ID TLV. The addresses MUST be IPv6 addresses when the IP Multicast Tunnel TLV is included in the IPv6 Interface ID TLV.

4. MPLS Context Label TLV. Type = 31 (To be assigned by IANA). In this case the TLV value is a <Source Address, MPLS Context Label> tuple. The Source Address belongs to Ru and the MPLS Context Label is an upstream assigned label, assigned by Ru. The Source Address MUST be set to an IPv4 address when the MPLS Context Label TLV is carried in the IPv4 Interface ID TLV. The Source Address MUST be set to an IPv6 address when the MPLS Context Label TLV is carried in the IPv6 Interface ID TLV. This allows Ru to tunnel an "inner" LDP P2MP LSP, the label of which is upstream assigned, over an "outer" one-hop MPLS LSP, where the outer one-hop LSP has the following property:

+ The label pushed by Ru for the outer MPLS LSP is an upstream assigned context label, assigned by Ru. When <Rd1...Rdn> perform an MPLS label lookup on this label a combination of this label and the incoming interface MUST be sufficient for <Rd1...Rdn> to uniquely determine Ru's context specific label space to lookup the next label on the stack in. <Rd1...Rdn> MUST receive the data sent by Ru with the context specific label assigned by Ru being the top label on the label stack.

Currently the usage of the context label TLV is limited only to LDP P2MP LSPs on a LAN as specified in the next section. The context label TLV MUST NOT be used for any other purposes.

Note that when the outer P2MP LSP is signaled with RSVP-TE or MLDP the above procedures assume that Ru has a priori knowledge of all the <Rd1, ... Rdn>. In the scenario where the outer P2MP LSP is signaled using RSVP-TE, Ru can obtain this information from RSVP-TE. However, in the scenario where the outer P2MP LSP is signaled using MLDP, MLDP does not provide this information to Ru. In this scenario the procedures by which Ru could acquire this information are outside the scope of this document.

6. LDP Point-to-Multipoint LSPs on a LAN

This section describes one application of upstream label assignment using LDP. Further applications are to be described in separate documents.

[MLDP] describes how to setup P2MP LSPs using LDP. On a LAN the solution relies on "ingress replication". A LSR on a LAN, that is a branch LSR for a P2MP LSP, (say Ru) sends a separate copy of a packet that it receives on the P2MP LSP to each of the downstream LSRs on the LAN (say <Rd1...Rdn> that are adjacent to it in the P2MP LSP.

It is desirable for Ru to send a single copy of the packet for the LDP P2MP LSP on the LAN, when there are multiple downstream routers on the LAN that are adjacent to Ru in that LDP P2MP LSP. This requires that each of <Rd1...Rdn> must be able to associate the label L, used by Ru to transmit packets for the P2MP LSP on the LAN, with that P2MP LSP. It is possible to achieve this using LDP upstream-assigned labels with the following procedures.

Consider an LSR Rd that receives the LDP P2MP FEC [MLDP] from its downstream LDP peer. Further the upstream interface to reach LSR Ru which is the next-hop to the P2MP LSP root address, Pr, in the LDP P2MP FEC, is a LAN interface, Li. Further Rd and Ru support upstream-assigned labels. In this case Rd instead of sending a Label Mapping message as described in [MLDP] sends a Label Request message to Ru. This Label Request message MUST contain an Upstream Assigned Label Request TLV.

On receiving this message, Ru sends back a Label Mapping message to Rd with an upstream-assigned label. This message also contains an Interface ID TLV with a MPLS Context Label sub-TLV, as described in the previous section, with the value of the MPLS label set to a value assigned by Ru on inteface Li as specified in [RFC5331]. Processing of the Label Request and Label Mapping messages for LDP upstream-assigned labels is as described in section 4.1. If Ru receives a Label Request for an upstream assigned label for the same P2MP FEC from multiple downstream LSRs on the LAN, <Rd1...Rdn>, it MUST send the same upstream-assigned label to each of <Rd1...Rdn>.

Ru transmits the MPLS packet using the procedures defined in [RFC5331] and [RFC5332]. The MPLS packet transmitted by Ru contains as the top label the context label assigned by Ru on the LAN interface, Li. The bottom label is the upstream label assigned by Ru to the LDP P2MP LSP. The top label is looked up in the context of the LAN interface, Li, [RFC5331] by a downstream LSR on the LAN. This lookup enables the downstream LSR to determine the context specific label space to lookup the inner label in.

Note that <Rd1...Rdn> may have more than one equal cost next-hop on the LAN to reach Pr. It MAY be desirable for all of them to send the label request to the same upstream LSR and they MAY select one upstream LSR using the following procedure:

1. The candidate upstream LSRs are numbered from lower to higher IP address

2. The following hash is performed: H = (Sum Opaque value) modulo N, where N is the number of candidate upstream LSRs. Opaque value is defined in [MLDP] and comprises the P2MP LSP identifier.

3. The selected upstream LSR U is the LSR that has the number H.

This allows for load balancing of a set of LSPs among a set of candidate upstream LSRs, while ensuring that on a LAN interface a single upstream LSR is selected. It is also to be noted that the procedures in this section can still be used by Rd and Ru if other LSRs on the LAN do not support upstream label assignment. Ingress replication and downstream label assignment will continue to be used for LSRs that do not support upstream label assignment.

7. IANA Considerations

7.1. LDP TLVs

IANA maintains a registry of LDP TLVs at the registry "Label Distribution Protocol" in the sub-registry called "TLV Type Name Space".

This document defines a new LDP Upstream Label Assignment Capability TLV (Section 3). IANA is requested to assign the value 0x0507 to this TLV.

This document defines a new LDP Upstream-Assigned Label TLV (Section 4). IANA is requested to assign the type value of 0x204 to this TLV.

This document defines a new LDP Upstream-Assigned Label Request TLV (Section 4). IANA is requested to assign the type value of 0x205 to this TLV.

7.2. Interface Type Identifiers

[RFC3472] defines the LDP Interface ID IPv4 and IPv6 TLV. These top-level TLVs can carry sub-TLVs dependent on the interface type. These sub-TLVs are assigned "Interface ID Types". IANA maintains a registry of Interface ID Types for use in GMPLS in the registry "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Parameters" and sub-registry "Interface_ID Types". IANA is requested to make corresponding allocations from this registry as follows:

- RSVP-TE P2MP LSP TLV (requested value 28)

- LDP P2MP LSP TLV (requested value 29)

- IP Multicast Tunnel TLV (requested value 30)

- MPLS Context Label TLV (requested value 31)

8. Security Considerations

The security considerations discussed in RFC 5036, RFC 5331 and RFC 5332 apply to this document.

More detailed discussion of security issues that are relevant in the context of MPLS and GMPLS, including security threats, related defensive techniques, and the mechanisms for detection and reporting, are discussed in "Security Framework for MPLS and GMPLS Networks

[MPLS-SEC].

9. Acknowledgements

Thanks to Yakov Rekhter for his contribution. Thanks to Ina Minei and Thomas Morin for their comments. The hashing algorithm used on LAN interfaces is taken from [MLDP]. Thanks to Loa Andersson, Adrian Farrel and Eric Rosen for their comments and review.

10. References

10.1. Normative References

[RFC5331] R. Aggarwal, Y. Rekhter, E. Rosen, "MPLS Upstream Label Assignment and Context Specific Label Space", RFC5331

[RFC5332] T. Eckert, E. Rosen, R. Aggarwal, Y. Rekhter, RFC5332

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels.", Bradner, March 1997

[RFC5036] L. Andersson, et. al., "LDP Specification", RFC5036.

[RFC4875] R. Aggarwal, D. Papadimitriou, S. Yasukawa [Editors], "Extensions to RSVP-TE for Point to Multipoint TE LSPs", RFC 4875

[MLDP] I. Minei et. al, "Label Distribution Protocol Extensions for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths", draft-ietf-mpls-ldp-p2mp-08.txt

10.2. Informative References

[RFC5561] B. Thomas, K. Raza, S. Aggarwal, R. Aggarwal, JL. Le Roux, "LDP Capabilities", RFC5561

[MPLS-SEC] L. fang, ed, "Security Framework for MPLS and GMPLS Networks", draft-ietf-mpls-mpls-and-gmpls-security-framework-07.txt

[RFC3032] E. Rosen et. al, "MPLS Label Stack Encoding", RFC 3032

[RFC3472] Ashwood-Smith, P. and L. Berger, Editors, " Generalized Multi-Protocol Label Switching (GMPLS) Signaling - Constraint-based Routed Label Distribution Protocol (CR-LDP) Extensions", RFC 3472, January 2003.

11. Author’s Address

Rahul Aggarwal
Juniper Networks
1194 North Mathilda Ave.
Sunnyvale, CA 94089
Phone: +1-408-936-2720
Email: rahul@juniper.net

Jean-Louis Le Roux
France Telecom
2, avenue Pierre-Marzin
22307 Lannion Cedex
France
E-mail: jeanlouis.leroux@orange-ftgroup.com