

Network Working Group
Internet-Draft
Expires: January 18, 2006

I. Minei
K. Kompella
Juniper Networks
J-L. Le Roux
France Telecom
L. Fang
AT&T
L. Wang
Telenor
S. Amante
Level 3 Communications, LLC
July 17, 2005

**Label Distribution Protocol Extensions for Point-to-Multipoint Label
Switched Paths
draft-minei-mpls-ldp-p2mp-01**

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on January 18, 2006.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

This document describes extensions to the Label Distribution Protocol (LDP) for the setup of point to multi-point (P2MP) Label Switched Paths (LSPs) in Multi-Protocol Label Switching (MPLS) networks. The solution relies on LDP without requiring a multicast routing protocol in the network. Protocol elements and procedures for this solution are described. There can be various applications for P2MP LSPs such as IP multicast. Specification of how such applications can use a LDP signaled P2MP LSP is outside the scope of this document.

Table of Contents

1.	Introduction	3
1.1	Conventions used in this document	3
1.2	Terminology	3
2.	Protocol Operation	5
2.1	The P2MP FEC Element	5
2.2	Using the P2MP FEC Element	6
2.2.1	Label Map	7
2.2.2	Label Withdraw	8
3.	Shared Trees	10
4.	Security considerations	11
5.	Acknowledgments	12
6.	References	13
6.1	Normative References	13
6.2	Informative References	13
	Authors' Addresses	13
	Intellectual Property and Copyright Statements	15

1. Introduction

The LDP protocol is described in [\[1\]](#). It defines mechanisms for setting up point to point (P2P) and multi-point to point (MP2P) LSPs in the network. This document describes extensions to LDP for setting up point to multi-point (P2MP) LSPs. Specifically, this document describes how a P2MP LSP can be set up that allows traffic from a single root (or ingress) node to be delivered to a number of leaf (or egress) nodes. Only a single copy of the packet will be sent on any link traversed by the P2MP LSP (see note at end of [Section 2.2.1](#)). This is accomplished without the use of a multicast protocol in the network. There can be several P2MP LSPs rooted at a given ingress node, each with its own identifier.

The solution assumes that the leaf nodes of the P2MP LSP know the root node and identifier of the P2MP LSP to which they belong. The mechanisms for the distribution of this information are outside the scope of this document. The specification of how an application can use a P2MP LSP signaled by LDP is also outside the scope of this document.

Interested readers may also wish to peruse the documents [\[4\]](#) and [\[6\]](#).

1.1 Conventions used in this document

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](#) [\[2\]](#).

1.2 Terminology

The following terminology is taken from [\[4\]](#).

P2P LSP: An LSP that has one Ingress LSR and one Egress LSR.

P2MP LSP: An LSP that has one Ingress LSR and one or more Egress LSRs.

MP2P LSP: A LSP that has one or more Ingress LSRs and one unique Egress LSR.

MP2MP LSP: A LSP that has one or more Ingress LSRs and one or more Egress LSRs.

Ingress LSR: Source of the P2MP LSP, also referred to as root node.

Egress LSR: One of potentially many destinations of an LSP, also referred to as leaf node in the case of P2MP and MP2MP LSPs.

Transit LSR: An LSR that has one or more directly connected downstream LSRs.

Bud LSR: An LSR that is an egress but also has one or more directly connected downstream LSRs.

2. Protocol Operation

A P2MP LSP consists of a single root node, zero or more transit nodes and one or more leaf nodes. Leaf nodes initiate P2MP LSP setup and tear-down. Leaf nodes also install forwarding state to deliver the traffic received on a P2MP LSP to wherever it needs to go; how this is done is outside the scope of this document. Transit nodes install MPLS forwarding state and propagate the P2MP LSP setup (and tear-down) toward the root. The root node installs forwarding state to map traffic into the P2MP LSP; how the root node determines which traffic should go over the P2MP LSP is outside the scope of this document.

For the setup of a P2MP LSP with LDP, we define one new protocol entity, the P2MP FEC Element to be used in the FEC TLV. The description of the P2MP FEC Element follows.

2.1 The P2MP FEC Element

The P2MP FEC Element consists of the address of the root of the P2MP LSP and an opaque identifier. The opaque identifier is unique within the context of the root node. The combination of (root LSR address, opaque identifier) uniquely identifies a P2MP LSP within the MPLS network.

The P2MP FEC element is encoded as follows:

```

      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 (TBD)  |           Address Family           | Address Length|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     Root Node Address                                     |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Opaque Identifier Type   |   Opaque Identifier Length   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Opaque Identifier ...   |
.
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Type: The type of the P2MP FEC element is to be assigned by IANA.

Address Family: Two octet quantity containing a value from ADDRESS FAMILY NUMBERS in [3] that encodes the address family for the Root LSR Address.

Address Length: Length of the Root LSR Address in octets.

Root Node Address: A host address encoded according to the Address Family field.

Opaque Identifier Type: The type of Opaque Identifier.

Length: The length of the P2MP Opaque Identifier, in octets.

Opaque Identifier: An opaque identifier of Length octets, padded at the end with zeros so as to be 4-octet aligned.

If Address Family is IPv4, the Address Length MUST be 4; if the Address Family is IPv6, the Address Length MUST be 16. No other Address Lengths are defined at present.

If the Address Length doesn't match the defined length for the Address Family, the receiver SHOULD abort processing the message containing the FEC Element, and send an "Unknown FEC" Notification message to its LDP peer signaling an error.

If a FEC TLV contains a P2MP FEC Element, the P2MP FEC Element MUST be the only FEC Element in the FEC TLV.

2.2 Using the P2MP FEC Element

This section defines the rules for the processing and propagation of the P2MP FEC Element. The following notation is used in the processing rules:

1. P2MP FEC Element <X, Y>: a FEC Element with Root Node Address X and Opaque Identifier Y.
2. P2MP Label Map <X, Y, L>: a Label Map message with a FEC TLV with a single P2MP FEC Element <X, Y> and Label TLV with label L.
3. P2MP Label Withdraw <X, Y, L>: a Label Withdraw message with a FEC TLV with a single P2MP FEC Element <X, Y> and Label TLV with label L.

4. P2MP LSP <X, Y> (or simply <X, Y>): a P2MP LSP with Root Node Address X and Opaque Identifier Y.

The procedures below are organized by the role which the node plays in the P2MP LSP. Node Z knows that it is a leaf node by a discovery process which is outside the scope of this document. During the course of protocol operation, the root node recognizes its role because it owns the Root Node Address. A transit node is any node (other than the root node) that receives a P2MP Label Map message (i.e., one that has leaf nodes downstream of it).

Note that a transit node (and indeed the root node) may also be a leaf node.

2.2.1 Label Map

The following lists procedures for generating and processing P2MP Label Map messages for nodes that participate in a P2MP LSP. An LSR should apply those procedures that apply to it, based on its role in the P2MP LSP.

In the current approach, if there are several receivers for a P2MP LSP on a LAN, packets are replicated over the LAN. This may not be optimal; optimizing this case is for further study.

2.2.1.1 Determining one's 'upstream LSR'

A node Z that is part of P2MP LSP <X, Y> determines the LDP peer U which lies on the best path from Z to the root node X. If there are more than one such LDP peers, only one of them is picked. U is Z's "Upstream LSR" for <X, Y>.

2.2.1.2 Leaf Operation

A leaf node Z of P2MP LSP <X, Y> determines its upstream LSR U for <X, Y> as per [Section 2.2.1.1](#), allocates a label L, and sends a P2MP Label Map <X, Y, L> to U.

2.2.1.3 Transit Node operation

Suppose a transit node Z receives a P2MP Label Map <X, Y, L> over interface I. Z checks whether it already has state for <X, Y>. If not, Z allocates a label L', and installs state to swap L' with L over interface I. Z also determines its upstream LSR U for <X, Y> as per [Section 2.2.1.1](#), and sends a P2MP Label Map <X, Y, L'> to U.

If Z already has state for <X, Y>, then Z adds "swap L, send over interface I" to the nexthop. Z does not send a Label Map message for

P2MP LSP <X, Y>.

2.2.1.4 Root Node Operation

Suppose the root node Z receives a P2MP Label Map <X, Y, L> over interface I. Z checks whether it already has forwarding state for <X, Y>. If not, Z creates forwarding state to push label L onto the traffic that Z wants to forward over the P2MP LSP (how this traffic is determined is outside the scope of this document).

If Z already has forwarding state for <X, Y>, then Z adds "push label L, send over interface I" to the nexthop.

2.2.2 Label Withdraw

The following lists procedures for generating and processing P2MP Label Withdraw messages for nodes that participate in a P2MP LSP. An LSR should apply those procedures that apply to it, based on its role in the P2MP LSP.

2.2.2.1 Leaf Operation

If a leaf node Z discovers (by means outside the scope of this document) that it is no longer a leaf of the P2MP LSP, it SHOULD send a Label Withdraw <X, Y, L> to its upstream LSR U for <X, Y>, where L is the label it had previously advertised to U for <X, Y>.

2.2.2.2 Transit Node Operation

If a transit node Z receives a Label Withdraw message <X, Y, L> from a node W, it deletes label L from its forwarding state, and sends a Label Release message with label L to W.

If deleting L from Z's forwarding state for P2MP LSP <X, Y> results in no state remaining for <X, Y>, then Z propagates the Label Withdraw <X, Y, L> to its upstream for <X, Y>.

2.2.2.3 Root Node Operation

The procedure when the root node of a P2MP LSP receives a Label Withdraw message are the same as for transit nodes, except that it would not propagate the Label Withdraw upstream (as it has no upstream).

2.2.2.4 Upstream LSR change

If, for a given node Z participating in a P2MP LSP <X, Y>, the upstream LSR changes, say from U to U', then Z MUST

1. send a Label Withdraw $\langle X, Y, L \rangle$ to U, where L is the label Z had previously sent to U for $\langle X, Y \rangle$;
2. delete all forwarding state for the P2MP LSP $\langle X, Y \rangle$;
3. allocate a label L' for $\langle X, Y \rangle$, and send a Label Map $\langle X, Y, L' \rangle$ to U';
4. install forwarding state for label L'.

3. Shared Trees

The mechanism described above shows how to build a tree with a single root and multiple leaves, i.e., a P2MP LSP. One can use essentially the same mechanism to build Shared Trees with LDP. A Shared Tree can be used by a group of routers that want to multicast traffic among themselves, i.e., each node is both a root node (when it sources traffic) and a leaf node (when any other member of the group sources traffic). A Shared Tree offers similar functionality to a MP2MP LSP, but the underlying multicasting mechanism uses a P2MP LSP. One example where a Shared Tree is useful is video-conferencing. Another is Virtual Private LAN Service (VPLS) [5], where for some types of traffic, each device participating in a VPLS must send packets to every other device in that VPLS.

One way to build a Shared Tree is to build an LDP P2MP LSP rooted at a common point, the Shared Root (SR), and whose leaves are all the members of the group. Each member of the Shared Tree unicasts traffic to the SR (using, for example, the MP2P LSP created by the unicast LDP FEC advertised by the SR); the SR then splices this traffic into the LDP P2MP LSP. The SR may be (but need not be) a member of the multicast group.

A major advantage of this approach is that no further protocol mechanisms beyond the one already described are needed to set up a Shared Tree. Furthermore, a Shared Tree is very efficient in terms of the multicast state in the network, and is reasonably efficient in terms of the bandwidth required to send traffic.

An important consideration in this approach is that a sender will receive its own packets as part of the multicast; thus a sender must be prepared to recognize and discard packets that it itself has sent. For a number of applications (for example, VPLS), this requirement is easy to meet. Another consideration is the various techniques that can be used to splice unicast LDP MP2P LSPs to the LDP P2MP LSP; these will be described in a later revision.

4. Security considerations

The same security considerations apply as for the base LDP specification, as described in [[1](#)].

5. Acknowledgments

The authors would like to thank Nischal Sheth, Yakov Rekhter and Rahul Aggarwal for their suggestions and review.

6. References

6.1 Normative References

- [1] Andersson, L., Doolan, P., Feldman, N., Fredette, A., and B. Thomas, "LDP Specification", [RFC 3036](#), January 2001.
- [2] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [3] Reynolds, J. and J. Postel, "Assigned Numbers", [RFC 1700](#), October 1994.
- [4] Roux, J., "Requirements for multipoint extensions to the Label Distribution Protocol", [draft-leroux-mpls-mp-ldp-reqs-00](#) (work in progress), July 2005.

6.2 Informative References

- [5] Andersson, L. and E. Rosen, "Framework for Layer 2 Virtual Private Networks (L2VPNs)", [draft-ietf-l2vpn-l2-framework-05](#) (work in progress), June 2004.
- [6] Aggarwal, R., "Extensions to RSVP-TE for Point to Multipoint TE LSPs", [draft-ietf-mpls-rsvp-te-p2mp-01](#) (work in progress), January 2005.

Authors' Addresses

Ina Minei
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: ina@juniper.net

Kireeti Kompella
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: kireeti@juniper.net

Jean-Louis Le Roux
France Telecom
2, avenue Pierre-Marzin
Lannion Cedex 22307
France

Email: jeanlouis.leroux@francetelecom.com

Luyuan Fang
AT&T
200 Laurel Avenue, Room C2-3B35
Middletown, NJ 07748
US

Email: luyuanfang@att.com

Lei Wang
Telenor
Snaroyveien 30
Fornebu 1331
Norway

Email: lei.wang@telenor.com

Shane Amante
Level 3 Communications, LLC
1025 Eldorado Blvd
Broomfield, CO 80021
US

Email: Shane.Amante@Level3.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

