Network Working Group
Internet Draft
Intended status: Standards Track
Updates: 6371 (if approved)
Expires: April 24, 2012

Sami Boutros (Ed.)
Siva Sivabalan (Ed.)
Cisco Systems, Inc.

Rahul Aggarwal (Ed.)
Arktan, Inc.

Martin Vigoureux (Ed.)
Alcatel-Lucent

Xuehui Dai (Ed.)
ZTE Corporation

October 24, 2011

MPLS Transport Profile lock Instruct and Loopback Functions
draft-ietf-mpls-tp-li-lb-08.txt

Abstract

Two useful Operations, Administration, and Maintenance (OAM) functions in a transport network are "lock" and "loopback". The lock function enables an operator to lock a transport path such that it does not carry client traffic, but can continue to carry OAM messages and may carry test traffic. The loopback function allows an operator to set a specific node on the transport path into loopback mode such that it returns all received data.

This document specifies the lock function for MPLS networks and describes how the loopback function operates in MPLS networks.

This document updates RFC 6371 section 7.1.1 and 7.1.2.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and 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

1. Introduction

Two useful Operations, Administration, and Maintenance (OAM) functions in a transport network are "lock" and "loopback". This document discusses these functions in the context of MPLS networks.

- The lock function enables an operator to lock a transport path such that it does not carry client traffic. As per RFC 5860 [1], lock is an administrative state in which it is expected that no client traffic may be carried. However, test traffic and OAM messages can still be mapped onto the locked transport path. The lock function may be applied to to Label Switched Paths (LSPs), Pseudowires (PWs) (including multi-segment Pseudowires) (MS-PWs), and bidirectional MPLS Sections as defined in RFC 5960 [9]).

- The loopback function allows an operator to set a specific node on a transport path into loopback mode such that it returns all received data. Loopback can be applied at a Maintenance Entity Group End Point (MEP) or a Maintenance Entity Group Intermediate Point (MIP) on a co-routed bidirectional LSP, on a PW, or on an bidirectional MPLS Section. It can also be applied at a MEP on an associated bidirectional LSP.

  Loopback is used to test the integrity of the transport path to and from the node that is performing loopback. It requires that the transport is locked and that a MEP on the transport path sends test data which it also validates on receipt.

This document specifies the lock function for MPLS networks and describes how the loopback function operates in MPLS networks.

This document updates RFC 6371 section 7.1.1 [6].

1.1. Updates RFC 6371

This document updates section 7.1.1 and 7.1.2 of RFC 6371 [6].

That framework makes the assumption that the Lock Instruct message is used to independently enable locking and requires a response message.

The mechanism defined in this document requires that a lock instruction is sent by management to both ends of the locked transport path and that the Lock Instruct message does not require a response.

2. Terminology and Conventions

2.1. 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].

2.2. Acronyms and Terms

ACH: Associated Channel Header

MEG: Maintenance Entity Group

MEP: Maintenance Entity Group End Point

MIP: Maintenance Entity Group Intermediate Point

MPLS-TP: MPLS Transport Profile

MPLS-TP LSP: Bidirectional Label Switch Path

TLV: Type Length Value

TTL: Time To Live

LI: Lock Instruct

NMS: Network Management System

Transport path: MPLS-TP LSP or MPLS PW

3. Lock Function

Lock is used to request a MEP to take a transport path out of service for administrative reasons. For example, Lock can be used to allow some form of maintenance to be done for a transport path. Lock is also a prerequisite of the Loopback function described in Section 4.

The NMS or a management process initiates a Lock by sending a Lock command to a MEP. The MEP takes the transport path out of service, that is, it stops injecting or forwarding traffic onto the transport path.

To properly lock a transport path (for example, to ensure that a loopback test can be performed), both directions of the transport path must be taken out of service so a Lock command is sent to the MEPs at both ends of the path. This ensures that no traffic is sent in either direction. Thus, the Lock function can be realized entirely using the management plane.

However, dispatch of messages in the management plane to the two MEPs may present coordination challenges. It is desirable that the lock be achieved in a coordinated way within a tight window, and this may be difficult with a busy management plane. In order to provide additional coordination, an LI OAM message can additionally be sent. A MEP locks a transport path when it receives a command from a management process or when it receives an LI message as described in Section 6.

This document defines an LI message for MPLS OAM. The LI message is based on a new ACH Type as well as an existing TLV. This is a common mechanism applicable to lock LSPs, PWs, and bidirectional MPLS Sections.

4. Loopback Function

This section provides a description of the Loopback function within an MPLS network. This function is achieved through management commands and so there is no protocol specification necessary. However, the Loopback function is dependent on the Lock function and so it is appropriate to describe it in this document.

The Loopback function is used to test the integrity of a transport path from a MEP up any other node in the same MEG. This is achieved by setting the target node into loopback mode, and transmitting a pattern of test data from the MEP. The target node loops all received data back toward the originator, and the MEP extracts the test data and compares it with what it sent.

Loopback is a function that enables a receiving MEP or MIP to return traffic to the sending MEP when in the loopback state. This state corresponds to the situation where, at a given node, a forwarding plane loop is configured and the incoming direction of a transport path is cross-connected to the outgoing reverse direction. Therefore, except in the case of early TTL expiry, traffic sent by the source will be received by that source.

Data plane loopback is an out-of-service function, as required in section 2.2.5 of RFC 5860 [1]. This function loops back all traffic (including user data and OAM). The traffic can be originated from one internal point at the ingress of a transport path within an interface or inserted from input port of an interface using an external test equipment. The traffic is looped back unmodified (other than normal per hop processing such as TTL decrement) in the direction of the point of origin by an interface at either an intermediate node or a terminating node.

It should be noted that data plane loopback function itself is applied to data plane loopback points residing on different interfaces from MIPs/MEPs. All traffic (including both payload and OAM) received on the looped back interface is sent on the reverse direction of the transport path.

For data plane loopback at an intermediate point in a transport path, the loopback needs to be configured to occur at either the ingress or egress interface. This is done using management.

The management plane can be used to configure the Loopback function. The management plane must ensure that the two MEPs are locked before it requests setting MEP or MIP in the loopback state.

The nature of test data and the use of loopback traffic to measure packet loss, delay, and delay variation is outside the scope of this document.

4.1. Operational Prerequisites

Obviously, for the Loopback function to operate, there are several prerequisites:

- There must be a return path, so transport path under test must be bidirectional.

- The node in loopback mode must be on both the forward and return paths. This is possible for all MEPs and MIPs on a co-routed bidirectional LSP, on a PW, or on a bidirectional MPLS Section, but is only possible on for MEPs on associated bidirectional LSPs.

- The transport path cannot deliver client data when one of its nodes is in loopback mode, so it is important that the transport path is locked before loopback is enabled.

- Management plane coordination between the node in loopback mode and the MEP sending test data is required. The MEP must not send test

data until loopback has been properly configured because this would result in the test data continuing toward the destination.

- The TTL of the test packets must be set sufficiently large to account for both directions of the transport path under test otherwise the packets will not be returned to the originating MEP.

- OAM messages intended for delivery to nodes along the transport path under test can be delivered by correct TTL expiry. However, OAM messages cannot be delivered to points beyond the loopback node until the loopback condition is lifted.

5. Lock Instruct Message

5.1. Message Identification

The Lock Instruct Message is carried in the Associated Channel Header (ACh) described in [4]. it is identified by a new PW ACh Type of 0xHH (to be assigned by IANA).

5.2. LI Message Format

The format of an LI Message is shown below.

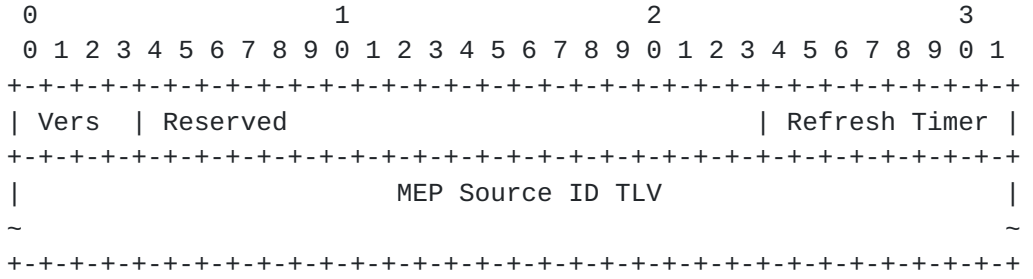

Figure 1: MPLS Lock Instruct Message Format

Version: The Version Number is currently 1. (Note: the version number is to be incremented whenever a change is made that affects the ability of an implementation to correctly parse or process the message. These changes include any syntactic or semantic changes made to any of the fixed fields, or to any Type-Length-Value (TLV) or sub-TLV assignment or format that is defined at a certain version number. The version number may not need to be changed if an optional TLV or sub-TLV is added.)

Reserved: The reserved field MUST be set to zero on transmission and SHOULD be ignored on receipt.

Refresh Timer: The maximum time between successive LI messages

specified in seconds. The default value is 1. The value 0 is not permitted. When a lock is applied, a refresh timer is chosen. This value MUST NOT be changed for the duration of that lock. A node receiving a LI message with a changed refresh timer MAY ignore the new value and continue to apply the old value.

MEP Source ID TLV: This is one of the three MEP Source ID TLVs defined in [3] and identifies the MEP that originated the LI message.

6. Operation of the Lock Function

6.1. Locking a Transport Path

When a MEP receives a Lock command from an NMS or through some other management process, it MUST take the transport path out of service. That is, it MUST stop injecting or forwarding traffic onto the LSP, PW, or bidirectional Section that has been locked.

As soon as the transport path has been locked, the MEP MUST send an LI message targeting the MEP at the other end of the locked transport path. The source MEP MUST set the Refresh Timer value in the LI message and MUST retransmit the LI message at the frequency indicated by the value set.

When locking a transport path, the NMS or management process is required to send a Lock command to both ends of the transport path. Thus a MEP may receive either the management command or an LI message first. A MEP MUST take the transport path out of service immediately in either case, but only sends LI messages itself after it has received a management lock command. Thus, a MEP is locked if either Lock was requested by management (and, as a result, the MEP is sending LI messages), or it is receiving LI messages from the remote MEP.

Note that a MEP that receives an LI message MUST identify the correct transport path and validate the message. The label stack on the received message is used to identify the transport path to be locked:

- If no matching label binding exists then there is no corresponding transport path and the received LI message is in error.

- If the transport path can be identified, but there is no return path (for example, the transport path was unidirectional) then the lock cannot be applied by the receiving MEP.

- If the transport path is suitable for locking but the source MEP-ID identifies an unexpected MEP for the MEG to which the receiving MEP belongs, the received LI message is in error.

When an errored LI message is received, the receiving MEP MUST NOT apply a lock. A MEP receiving errored LI messages SHOULD perform local diagnostic actions (such as counting the messages) and MAY log the messages.

A MEP keeps a transport path locked as long as it is either receiving the periodic LI messages or has an in-force Lock command from management. (see Section 6.2 for an explanation of unlocking a MEP). Note that in some scenarios (such as the use of loopback as described in Section 4) LI messages will not continue to be delivered on a locked transport path. This is why a transport path is considered locked while there is an in-force Lock command from a management process regardless of whether LI messages are being received.

6.2. UnLocking a Transport Path

Unlock is used to request a MEP to bring the previously locked transport path back in service.

When a MEP receives an Unlocked command from a management process it MUST cease sending LI messages. However, as described in Section 6.1, if the MEP is still receiving LI messages, the transport path MUST remain out of service. Thus, to unlock a transport path, the management process has to send an Unlock command to the MEPs at both ends.

When a MEP has been unlocked and has not received an LI message for a multiple of 3.5 times the Refresh Timer on the LI message (or has never received an LI message), the MEP unlocks the transport path and puts it back into service.

7. Security Considerations

MPLS-TP is a subset of MPLS and so builds upon many of the aspects of the security model of MPLS. MPLS networks make the assumption that it is very hard to inject traffic into a network, and equally hard to cause traffic to be directed outside the network. For more information on the generic aspects of MPLS security, see [7].

This document describes a protocol carried in the G-ACh [4], and so is dependent on the security of the G-ACh, itself. The G-ACh is a generalization of the Associated Channel defined in [8]. Thus, this document relies heavily on the security mechanisms provided for the Associated Channel and described in [4] and [8].

A specific concern for the G-ACh is that is can be used to provide a covert channel. This problem is wider than the scope of this document and does not need to be addressed here, but it should be

noted that the channel provides end-to-end connectivity and SHOULD NOT be policed by transit nodes. Thus, there is no simple way of preventing any traffic being carried in the G-ACh between consenting nodes.

A good discussion of the data plane security of an associated channel may be found in [5]. That document also describes some mitigation techniques.

It should be noted that the G-ACh is essentially connection-oriented so injection or modification of control messages specified in this document require the subversion of a transit node. Such subversion is generally considered hard in MPLS networks, and impossible to protect against at the protocol level. Management level techniques are more appropriate.

8. IANA Considerations

8.1. Pseudowire Associated Channel Type

LI OAM requires a unique Associated Channel Type which is assigned by IANA from the Pseudowire Associated Channel Types Registry.

Registry:

Value	Description	TLV Follows	Reference
0xHH	LI	No	[This.I-D]

9. Acknowledgements

The authors would like to thank Loa Andersson, Yoshinori Koike, Alessandro D'Alessandro Gerardo, Shahram Davari, Greg Mirsky, Yaacov Weingarten, Liu Guoman, Matthew Bocci, and Adrian Farrel for their valuable comments.

10. References

10.1. Normative References

[1] Vigoureux, M., Ward, D., and M. Betts, "Requirements for Operations, Administration, and Maintenance (OAM) in MPLS Transport Networks", RFC 5860, May 2010.

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

[3] D. Allan, et. al., Proactive Connectivity Verification, Continuity Check and Remote Defect indication for MPLS

Transport Profile draft-ietf-mpls-tp-cc-cv-rdi-06, work in progress, June 2010

[4] Bocci, M., Vigoureux, M., and S. Bryant, "MPLS Generic Associated Channel", RFC 5586, June 2009.

[5] T. Nadeau, C. Pignataro, "Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires", RFC 5085, Dec 2007.

[6] Busi, I. and Allan, D., "Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks", RFC 6371, September 2011

10.2. Informative References

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

[8] S. Bryant, G. Swallow, L. Martini "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, Feb 2006.

[9] Frost, D., Ed., Bryant, S., Ed., and M. Bocci, Ed., "MPLS Transport Profile Data Plane Architecture", RFC 5960, August 2010.

Editors' Addresses

Sami Boutros
Cisco Systems, Inc.
Email: sboutros@cisco.com

Siva Sivabalan
Cisco Systems, Inc.
Email: msiva@cisco.com

Rahul Aggarwal
Arktan, Inc
EMail: raggarwa_1@yahoo.com

Martin Vigoureux
Alcatel-Lucent.
Email: martin.vigoureux@alcatel-lucent.com

Xuehui Dai
ZTE Corporation.
Email: dai.xuehui@zte.com.cn

Contributing Authors

George Swallow
Cisco Systems, Inc.
Email: swallow@cisco.com

David Ward
Juniper Networks.
Email: dward@juniper.net

Stewart Bryant
Cisco Systems, Inc.
Email: stbryant@cisco.com

Carlos Pignataro
Cisco Systems, Inc.
Email: cpignata@cisco.com

Eric Osborne
Cisco Systems, Inc.
Email: eosborne@cisco.com

Nabil Bitar
Verizon.
Email: nabil.bitar@verizon.com

Italo Busi
Alcatel-Lucent.
Email: italo.busi@alcatel-lucent.com

Lieven Levrau
Alcatel-Lucent.
Email: lieven.levrau@alcatel-lucent.com

Laurent Ciavaglia
Alcatel-Lucent.
Email: laurent.ciavaglia@alcatel-lucent.com

Bo Wu
ZTE Corporation.
Email: wu.bo@zte.com.cn

Jian Yang
ZTE Corporation.
Email: yang_jian@zte.com.cn