

Troubleshoot Make-Before-Break (MBB) for mLDP

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[Theory](#)

[MBB Query and Ack Mechanism](#)

[MBB Capability](#)

[MBB Delay](#)

[MBB for Rerouting](#)

[MBB for Protection](#)

[Theory of mLDP Protection](#)

[Required Configuration](#)

[MBB for FRR Example](#)

[MBB in use](#)

[mLDP Trace](#)

[Configure FRR timer for scale MLDP LSPs](#)

[Conclusion](#)

Introduction

This document describes the behavior of Make-Before-Break (MBB) in Cisco IOS[®] XR.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

- This document is specific to Cisco IOS[®] XR, but it is not restricted to a specific software release or hardware.
- All devices used in this article run Cisco IOS XR 6.5.2. All line cards are third or fourth generation ASR9k line cards.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

Make-Before-Break (MBB) has a purpose: to set up a new mLDP (multipoint Label Distribution Protocol) tree before tearing down the old tree and switching traffic from the old to the new tree without losing multicast traffic. This can be used in two scenarios:

1. When rerouting occurs when a new path becomes available with a better IGP (Interior Gateway Protocol) metric: switch the traffic from the old to the new tree and then delete the old tree.
2. When the multicast traffic is protected on a backup path: switch the traffic from the backup path to the newly signalled native mLDP tree and then delete the backup tree.

If the router knows that the old LSP (Label Switched Path) is broken, it must not wait to start using the new LSP. Waiting does not make sense here, as there is no traffic arriving any longer on the old tree. If the old tree is still working, then the router must not tear down the old tree until the new tree is fully set up.

Theory

MBB is driven by a Query and Ack mechanism as described in RFC 6388. This is the base RFC of mLDP. This Query and Ack mechanism signals when the new tree is ready to forward multicast traffic. In this way, there must not be any packet loss. If the router knows that the old LSP is broken, it must not wait to start using the new LSP. Waiting does not make sense here, as there is no traffic arriving any longer on the old tree. If the old tree is still working, then the router must not tear down the old tree until the new tree is fully set up.

The cases where MBB can help are:

- The existing tree is being rerouted due to a change in the topology, a link up event or a reduction in the IGP cost of a link occurs,
- Reverting back to a native mLDP tree, after receiving the multicast traffic on an actively protected TE tunnel/LFA (Loop-free Alternate), or Ti-LFA (Topology independent LFA) path.

Notice that these two represent good events. An example of a bad event would be a directly connected link going down on a router on the upstream path. MBB cannot help in this case. IP FRR (Fast ReRoute) is needed in this case.

When MBB occurs, there is temporarily more than one upstream neighbor and/or more than one downstream neighbor. In RFC 6388, it is specified that there can be multiple accepting elements. This means that there can be multiple upstream neighbors and upstream label values per tree. An "accepting element" means that the upstream mLDP neighbor is a candidate for accepting traffic on. One accepting element is the active element. The active element is the one for which the MPLS label is installed in the forwarding plane. The other accepting element is the inactive element. This element is the one for which the MPLS label is not yet installed in the forwarding plane. This inactive element is the one for the newly signalled part of the tree with the Query/Ack mechanism and must be short lived, before it transitions to becoming the active accepting element. There can be only two accepting elements per tree: one is the active one, the other is the inactive one. As soon as the Query/Ack signalling finishes or a fixed time delay is reached, the old neighbors are removed from the tree.

Instead of the Query/Ack mechanism, the other implementation choice could be to just delay the switchover to the new LSP by a fixed configurable delay.

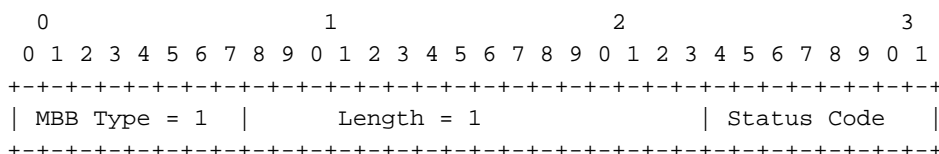
It is important to note that mLDP shares the downstream assigned label space that unicast uses and hence for the MPLS forwarding plane there is in essence no difference between multicast packets or unicast packets. Since the forwarding plane is shared with unicast, certain unicast feature are inherited for multicast, like IP FRR.

The MBB procedures applies to P2MP (Point-to-Multipoint) and MP2MP (Multipoint-to-Multipoint) trees.

MBB Query and Ack Mechanism

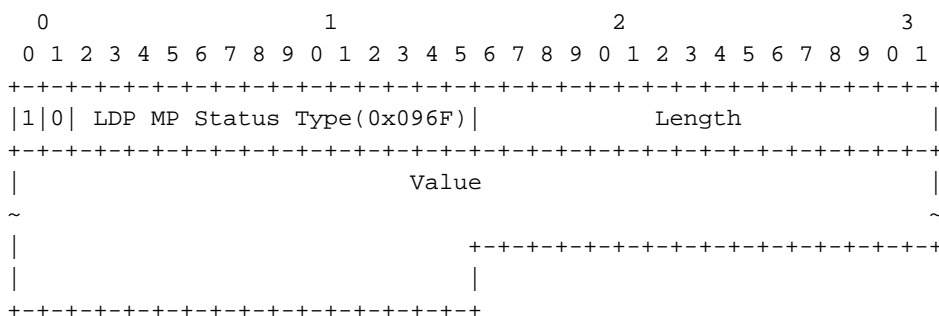
MBB is optional (it is also optional in the RFC), so it must be configured for it to be enabled. When it is configured, there can be an MBB status attached to the Label Mapping Message sent upstream and it can also be attached to an LDP Notification message sent by an upstream router to the downstream router. A router can attach a MBB Status in a LDP MP Status TLV.

The MBB Status is a type of the LDP MP Status Value Element:



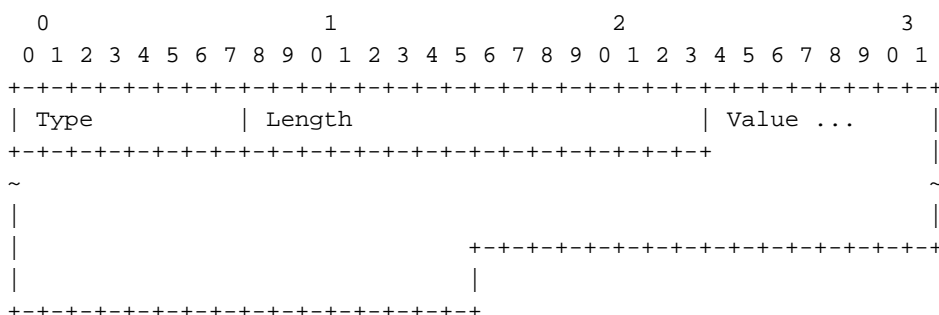
The Status Code is 1 for a MBB request and 2 for a MBB ack.

The LDP MP Status TLV is encoded as follows:



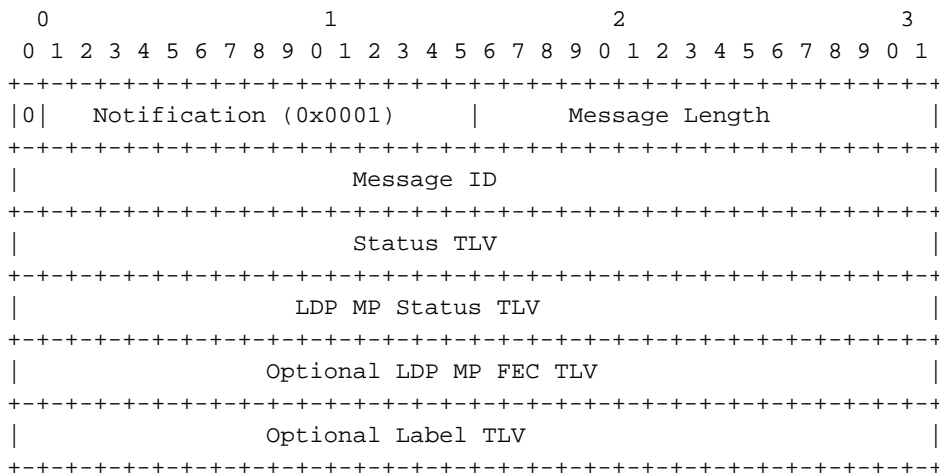
The Value field holds one or more LDP MP Status Value elements.

The LDP MP Status Value Element that is included in the LDP MP Status TLV Value has the next encoding:

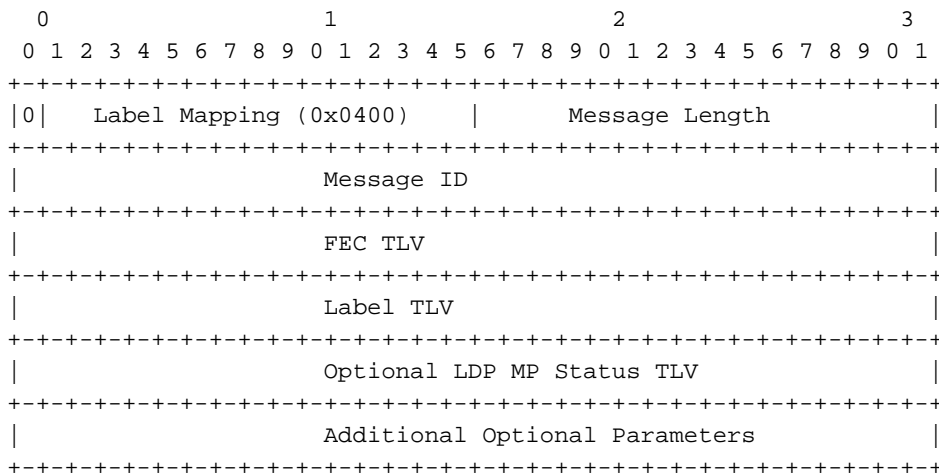


The LDP MP Status TLV can appear either in a Label Mapping message or a LDP Notification message.

In a LDP Notification Message:



In a Label Mapping Message:



The previous describes the dynamic MBB behavior. Another option is to have a static behavior where the switchover to the new tree is only determined by a delay. In this case, the switchover occurs a certain amount of (milli)seconds after the new tree is ready.

Image 1 depicts a capture in Wireshark of the mLDP Label Mapping message. There is a LDP MP Status TLV attached.

```

  Label Mapping Message
    0... .... = U bit: Unknown bit not set
    Message Type: Label Mapping Message (0x400)
    Message Length: 48
    Message ID: 0x000001d3
  FEC
    00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x0)
    TLV Type: FEC (0x100)
    TLV Length: 24
  FEC Elements
    FEC Element 1
      FEC Element Type: MP2MP-up (7)
      FEC Element Address Type: IPv4 (1)
      FEC Element Length: 4
      Root Node Address: 10.100.1.3
      Opaque Length: 14
      Opaque Value: 02000b0010000000200000000000
  Generic Label
    00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x0)
    TLV Type: Generic Label (0x200)
    TLV Length: 4
    .... .... 0000 0101 1101 1100 1000 = Generic Label: 0x05dc8
  LDP MP Status TLV Type
    10.. .... = TLV Unknown bits: Unknown TLV, do not Forward (0x2)
    TLV Type: LDP MP Status TLV Type (0x96F)
    TLV Length: 4
    TLV Value: 01000102

```

Image 1

01000102 decodes to 1 for MBB Type 1, 0001 for Length 1, and 02 for MBB Ack.

Notice that the MBB mechanism applies to the P2MP mLDP FEC (Forwarding Equivalence Class), and the MP2MP Upstream or Downstream FECs.

MBB Capability

A router capable of performing MBB, advertises this in a MBB Capability advertisement on the LDP session to its neighbors.

```

RP/0/RSP1/CPU0:R2#show mpls mldp neighbors
MLDP peer ID      : 10.79.196.14:0, uptime 22:32:06 Up,
Capabilities      : Typed Wildcard FEC, P2MP, MP2MP, MBB
Target Adj       : No
Upstream count    : 0
Branch count      : 0
Label map timer   : never
Policy filter in  :
Path count        : 1
Path(s)           : 10.159.248.201      Bundle-Ether120 No LDP
Adj list          : 10.254.3.36         Bundle-Ether10362
Peer addr list    : 10.79.196.14

```

```
: 10.55.55.1
: 10.196.91.134
: 10.200.30.1
```

MBB is not enabled by default for Cisco IOS XR.

The command "make-before-break" enables the feature and the capability advertisement.

```
mpls ldp
 mldp
 logging notifications
 address-family ipv4
 make-before-break delay 0
```

The MBB does not have a delay by default. Only in a scaled setup, the delay must be increased. The reason is that with many mLDP database entries there can be many mLDP forwarding entries that need to be installed. The time to install these forwarding entries into the data plane of the line cards can take some time.

MBB Delay

Look at image 2.

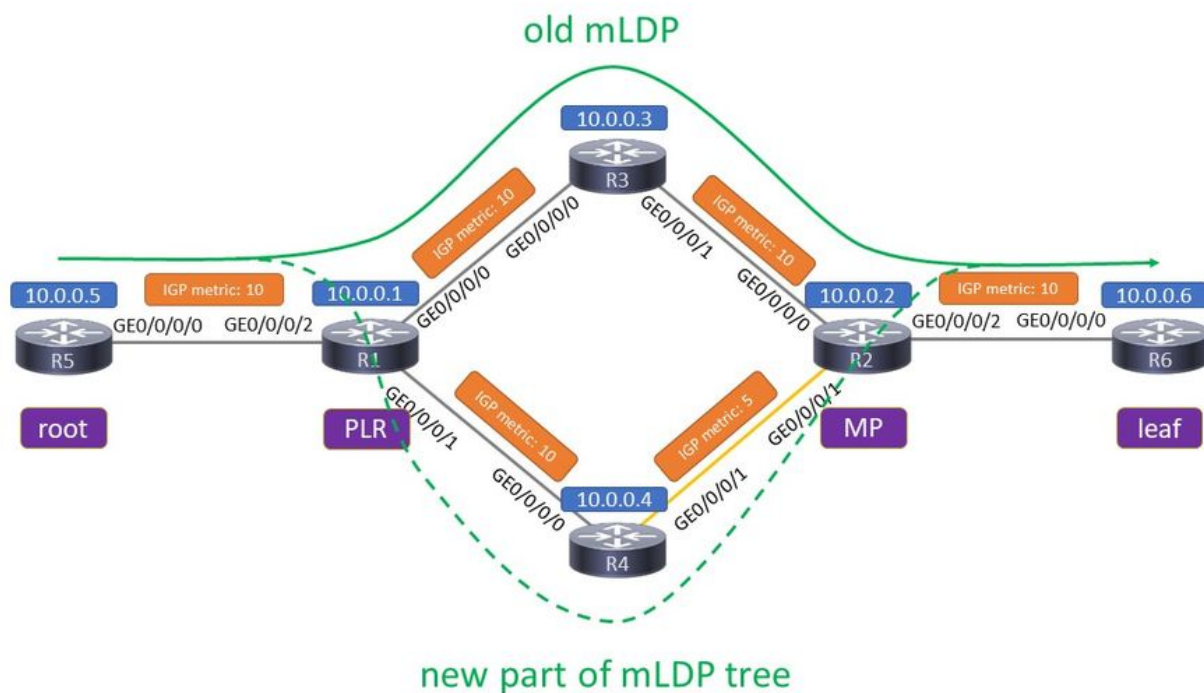


Image 2

There is the old tree and the newly signalled tree. The router where the two trees branch is the Point of Local Repair (PLR). The router where the two trees merge again is the Merge Point (MP). The new part of the mLDP tree is signalled due to the routers discovering a better path. Either, the new link R4 – R2 became available, or the IGP metric on that link was lowered to produce a path with a lower overall metric.

You can configure two delay values for MBB. The first is the delay when MBB is used to have the MP switchover back to a native path. This is the time after the MBB ack is received.

```
RP/0/RP1/CPU0:Router(config-ldp-ml dp-af)#make-before-break delay ?
<0-600> Forwarding delay in seconds
```

A delay of zero means that the newly signalled path is immediately used after the MBB Ack is received on the router where the old and new path diverse, the PLR. The second is the delay for the deletion of the backup path after the MP has switched over to the native path.

```
RP/0/RP1/CPU0:Router(config-ldp-ml dp-af)#make-before-break delay 10 ?
<0-60> Delete delay in seconds
<cr>
```

```
RP/0/RP1/CPU0:Router(config-ldp-ml dp-af)#make-before-break delay 10 10 ?
<cr>
```

Both the switchover delay and the delete delay are used on the MP.

MBB for Rerouting

MBB takes care of setting up a new mLDP tree before the old one is taken down. This only makes sense if the old tree is still present and forwarding traffic. An IGP convergence, such as a link up event, can produce a better path for the mLDP tree. This means a smaller IGP metric towards the root, or towards the leaf if it is a MP2MP mLDP tree.

Look at an example.

Image 3 shows a network before the routing convergence event.

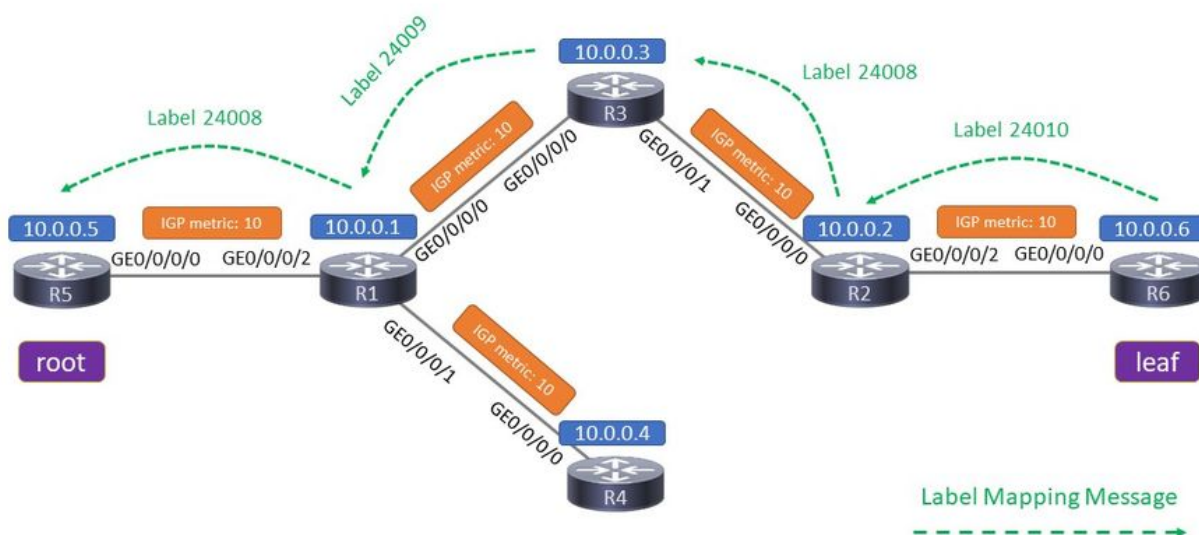


Image 3

R5 is the root router of one mLDP tree and R6 is the leaf router. A P2MP mLDP Tree is signalled with a Label Mapping message (including a MPLS label), from every router towards the root. This

LDP Label Mapping message does not carry a MBB Request.

The mLDP traffic goes left (root) to right (leaf) over the top path. At each link, the indicated MPLS label is on top of the multicast packet.

Image 4 shows the network after the routing convergence event (without MBB).

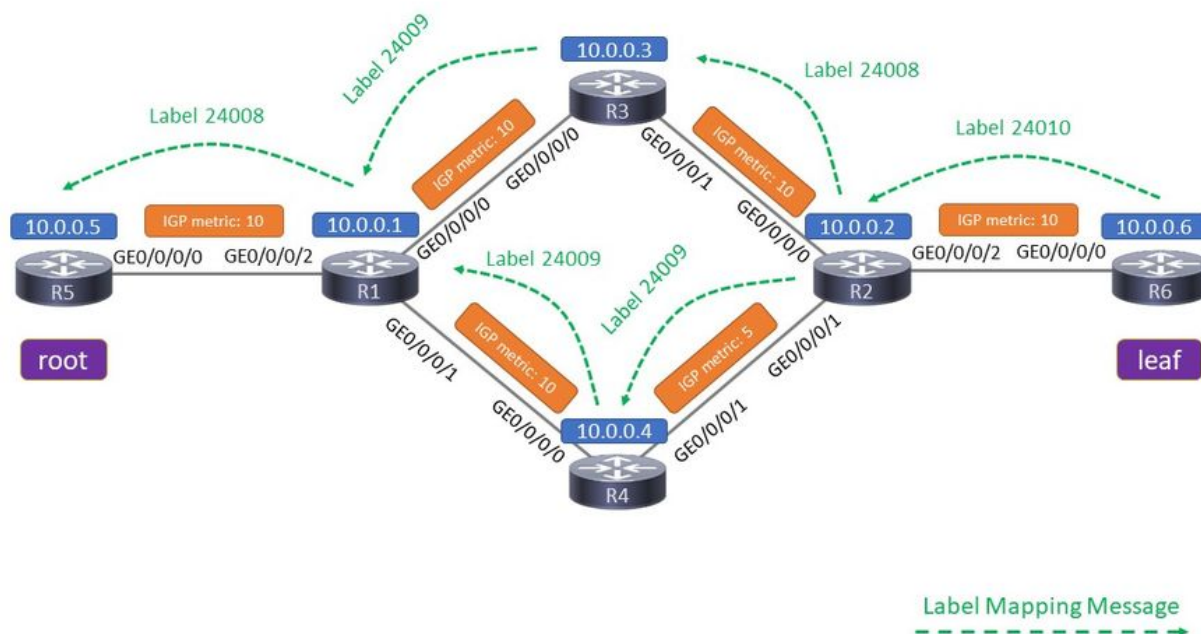


Image 4

The link R4 – R2 is now up. The metric of this link is of a low value so that the bottom path has a lower metric than the top path. Two things need to happen: the IGP adjacency needs to be established over the link and the LDP session needs to be established as well over this new link. Once this LDP session is up, the Label Mapping message is exchanged over this link in order to move the mLDP tree from the top to the bottom.

If MBB is not configured, then there is regular signalling with LDP Label Mapping messages on the bottom path. As soon as the Label Mapping message (without a MBB Request) reaches R1, R1 stops forwarding the multicast traffic on the top path and start forwarding the multicast traffic on the bottom path.

In the end, R1 never forwarded the multicast traffic over the two paths, but only over one: it switched the traffic from the top to the bottom path. The switchover is immediate which could lead to a short period of multicast traffic dropped due to the fact that the control plane signalling from R2 to R1 over R4 could be a bit faster than the time needed for the mLDP entries to be installed in the data plane on the routers on the new path.

There is mLDP logging notification enabled explicitly.

```
RP/0/0/CPU0:Jan 1 16:06:49.778 : mpls_ldp[1180]: %ROUTING-MLDP-5-BRANCH_ADD : 0x00001 [ipv4 10.0.0.105 232.1.1.1] P2MP 10.0.0.5, Add LDP 10.0.0.4:0 branch remote label 24009
```



```
RP/0/0/CPU0:Jan 1 16:06:49.838 : mpls_ldp[1180]: %ROUTING-MLDP-5-BRANCH_DELETE : 0x00001 [ipv4
10.0.0.105 232.1.1.1] P2MP 10.0.0.5, Delete LDP 10.0.0.3:0 branch remote label 24009
```

If MBB is configured, we have as follows.

Note that it is not sufficient to just configure MBB on R1.

This is an example configuration on R2:

```
mpls ldp
 mldp
 logging notifications
 address-family ipv4
  make-before-break delay 60
 !
```

You would want R2 to delay the switchover from the old to the new path with 60 seconds when the LDP session across the link R4-R2 is up. It does not happen. You need to have MBB enabled on every router (or at least R1, R4, and R2) to have the MBB signalling working between R2 and R1 across R4.

You need to have this minimal configuration on every router in order to have MBB signalling enabled.

```
mpls ldp
 mldp
 logging notifications
 address-family ipv4
  make-before-break delay 0
 !
```

Look at image 5.

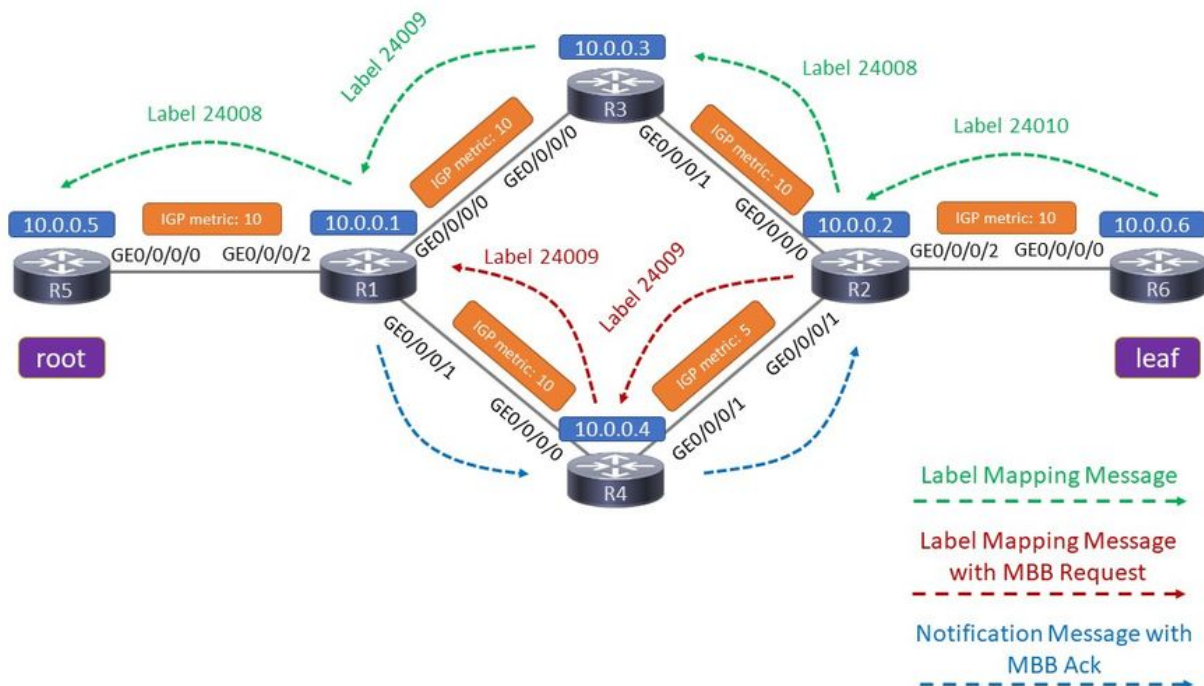


Image 5

All the correct configuration is in place. We look at the events from the beginning, so the situation before the convergence event.

The top path active is the start. On R1, R3 is the only downstream client.

```
RP/0/0/CPU0:R1#show mpls mldp database
mLDP database
LSM-ID: 0x00001  Type: P2MP  Uptime: 00:19:43
FEC Root       : 10.0.0.5
Opaque decoded  : [ipv4 10.0.0.105 232.1.1.1]
Features       : MBB
Upstream neighbor(s) :
  10.0.0.5:0 [Active] [MBB] Uptime: 00:19:43
    Local Label (D) : 24008
Downstream client(s):
  LDP 10.0.0.3:0      Uptime: 00:03:28
    Next Hop          : 10.1.3.3
    Interface         : GigabitEthernet0/0/0/0
    Remote label (D) : 24009
```

```
RP/0/0/CPU0:R1#show mpls mldp forwarding
mLDP MPLS forwarding database

24008  LSM-ID: 0x00001 flags: None
      24009, NH: 10.1.3.3, Intf: GigabitEthernet0/0/0/0 Role: M
```

On R2, R3 is the only accepting element.

```
RP/0/0/CPU0:R2#show mpls mldp database
mLDP database
LSM-ID: 0x00001  Type: P2MP  Uptime: 00:23:58
FEC Root       : 10.0.0.5
Opaque decoded  : [ipv4 10.0.0.105 232.1.1.1]
Features       : MBB
Upstream neighbor(s) :
  10.0.0.3:0 [Active] [MBB] Uptime: 00:03:19
    Local Label (D) : 24008
Downstream client(s):
  LDP 10.0.0.6:0  Uptime: 00:23:58
    Next Hop      : 10.2.6.6
    Interface     : GigabitEthernet0/0/0/2
    Remote label (D) : 24010
```

```
RP/0/0/CPU0:R2#show mpls mldp forwarding
mLDP MPLS forwarding database

24008  LSM-ID: 0x00001 flags: None
      24010, NH: 10.2.6.6, Intf: GigabitEthernet0/0/0/2 Role: M
```

After the MBB signalling, R2 has two accepting elements, one active, one inactive.

```
Jan  1 16:52:43.700 : mpls_ldp[1180]: %ROUTING-MLDP-5-BRANCH_ADD : 0x00001 [ipv4 10.0.0.105
232.1.1.1] P2MP 10.0.0.5, Add LDP 10.0.0.4:0 branch remote label 240
```

R1 has two downstream clients, R3 and R4:

```
RP/0/0/CPU0:R1#show mpls mldp database
mLDP database
LSM-ID: 0x00001  Type: P2MP  Uptime: 00:22:35
```

```

FEC Root          : 10.0.0.5
Opaque decoded    : [ipv4 10.0.0.105 232.1.1.1]
Features          : MBB
Upstream neighbor(s) :
  10.0.0.5:0 [Active] [MBB] Uptime: 00:22:35
    Local Label (D) : 24008
Downstream client(s):
  LDP 10.0.0.3:0   Uptime: 00:06:20
    Next Hop       : 10.1.3.3
    Interface      : GigabitEthernet0/0/0/0
    Remote label (D) : 24009
  LDP 10.0.0.4:0   Uptime: 00:00:36
    Next Hop       : 10.1.4.4
    Interface      : GigabitEthernet0/0/0/1
    Remote label (D) : 24009

```

R1 is forwarding over both paths:

```
RP/0/0/CPU0:R1#show mpls mldp forwarding
```

```
mLDP MPLS forwarding database
```

```
24008 LSM-ID: 0x00001 flags: None
```

```
  24009, NH: 10.1.3.3, Intf: GigabitEthernet0/0/0/0 Role: M
```

```
  24009, NH: 10.1.4.4, Intf: GigabitEthernet0/0/0/1 Role: M
```

R2 now has two upstream neighbors, one active (R3), and one inactive (R4). This phase is there for 60 seconds, the forwarding delay time.

```
RP/0/0/CPU0:R2#show mpls mldp database
```

```
mLDP database
```

```
LSM-ID: 0x00001 Type: P2MP Uptime: 00:27:00
```

```
FEC Root          : 10.0.0.5
```

```
Opaque decoded    : [ipv4 10.0.0.105 232.1.1.1]
```

```
MBB nbr evaluate  : 00:00:21
```

```
Features          : MBB
```

```
Upstream neighbor(s) :
```

```
  10.0.0.4:0 [Inactive] [MBB] Uptime: 00:00:38
```

```
    Local Label (D) : 24009
```

```
  10.0.0.3:0 [Active] [Delete] [MBB] Uptime: 00:06:22
```

```
    Local Label (D) : 24008
```

```
Downstream client(s):
```

```
  LDP 10.0.0.6:0   Uptime: 00:27:00
```

```
    Next Hop       : 10.2.6.6
```

```
    Interface      : GigabitEthernet0/0/0/2
```

```
    Remote label (D) : 24010
```

```
RP/0/0/CPU0:R2#show mpls mldp forwarding
```

```
mLDP MPLS forwarding database
```

```
24008 LSM-ID: 0x00001 flags: None
```

```
  24010, NH: 10.2.6.6, Intf: GigabitEthernet0/0/0/2 Role: M
```

```
24009 LSM-ID: 0x00001 flags: ED
```

```
  24010, NH: 10.2.6.6, Intf: GigabitEthernet0/0/0/2 Role: M
```

Notice that the local label for each mLDP tree is different. So, R2 has no issue differentiating the incoming mLDP traffic and identifying which incoming mLDP packet belongs to which mLDP tree. R2 only forwards the traffic from one tree at any time. The flag ED means 'Egress Drop' and means that packets arriving with label 24009 is dropped. These are the packets on the tree for which the accepting element is inactive. There is no duplicate traffic arriving at the receivers!

Notice that the outgoing label for each mLDP tree on R2 is the same. So, to R6, a downstream router of R2, it cannot distinguish if the traffic came over the original old (top) path or the new (bottom) path after rerouting.

After 60 seconds, R2 stops forwarding the traffic from the top path, and starts the traffic from the bottom path.

```
RP/0/0/CPU0:R1 Jan  1 16:53:44.236 : mpls_ldp[1180]: %ROUTING-MLDP-5-BRANCH_DELETE : 0x00001
[ipv4 10.0.0.105 232.1.1.1] P2MP 10.0.0.5, Delete LDP 10.0.0.3:0 branch remote label 24009
```

R1 has only one downstream client, R4.

```
RP/0/0/CPU0:R1#show mpls mldp database
mLDP database
LSM-ID: 0x00001  Type: P2MP  Uptime: 00:25:21
FEC Root          : 10.0.0.5
Opaque decoded    : [ipv4 10.0.0.105 232.1.1.1]
Features          : MBB
Upstream neighbor(s) :
  10.0.0.5:0 [Active] [MBB] Uptime: 00:25:21
    Local Label (D) : 24008
Downstream client(s):
  LDP 10.0.0.4:0      Uptime: 00:03:22
    Next Hop          : 10.1.4.4
    Interface         : GigabitEthernet0/0/0/1
    Remote label (D) : 24009
```

```
RP/0/0/CPU0:R1#show mpls mldp forwarding
mLDP MPLS forwarding database

24008 LSM-ID: 0x00001 flags: None
  24009, NH: 10.1.4.4, Intf: GigabitEthernet0/0/0/1 Role: M
```

R2 has only one upstream neighbor:

```
RP/0/0/CPU0:R2#show mpls mldp database
mLDP database
LSM-ID: 0x00001  Type: P2MP  Uptime: 00:29:54
FEC Root          : 10.0.0.5
Opaque decoded    : [ipv4 10.0.0.105 232.1.1.1]
Features          : MBB
Upstream neighbor(s) :
  10.0.0.4:0 [Active] [MBB] Uptime: 00:03:31
    Local Label (D) : 24009
Downstream client(s):
  LDP 10.0.0.6:0      Uptime: 00:29:54
    Next Hop          : 10.2.6.6
    Interface         : GigabitEthernet0/0/0/2
    Remote label (D) : 24010
```

```
RP/0/0/CPU0:R2#show mpls mldp forwarding
mLDP MPLS forwarding database

24009 LSM-ID: 0x00001 flags: None
  24010, NH: 10.2.6.6, Intf: GigabitEthernet0/0/0/2 Role: M
```

The mLDP trace on R2 shows that the MBB signalling was used, that there was a 60 second delay before switching over from the old path to the new path and a subsequent 0 second delay for deleting the old path. After this, R2 sends a Label Withdraw message to R3 for the old path and

receives a Label Release message from R3 as a response.

```
RP/0/0/CPU0:R2#show mpls mldp trace
```

```
Jan 1 16:52:43.370 MLDP GLO 0/0/CPU0 t21 NBR : New LDP peer 10.0.0.4:0 UP cap: f
Jan 1 16:52:43.370 MLDP GLO 0/0/CPU0 t21 NBR : 10.0.0.4:0 LDP Adjacency addr: 10.2.4.4,
Interface: GigabitEthernet0/0/0/1 Add
Jan 1 16:52:43.660 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 ACEL 10.0.0.4:0 installed local label
24009
Jan 1 16:52:43.660 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 P2MP label mapping MBB Request msg to
10.0.0.4:0 Success
Jan 1 16:52:43.660 MLDP LSP 0/0/CPU0 t21 FWD : 0x00001 Label 24009 add path label 24010 intf
GigabitEthernet0/0/0/2 nexthop 10.2.6.6 id 0x00001 Success
Jan 1 16:52:43.660 MLDP GLO 0/0/CPU0 t21 GEN : Root 10.0.0.5 path 10.2.4.4 php nh 10.2.4.4
peer 134a338c:10.0.0.4:0
Jan 1 16:52:43.910 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 P2MP notification from 10.0.0.4:0 root
10.0.16.0 Opaque Len: 83886090 MBB Ack
Jan 1 16:52:43.910 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 Start MBB Notification timer 100 msec
(MBB ack)
Jan 1 16:52:43.910 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 ACEL selection delayed for 60 seconds
(MBB)
Jan 1 16:53:44.156 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 ACEL 10.0.0.3:0 start delete pending
timer at 0 sec
Jan 1 16:53:44.156 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 ACEL 10.0.0.4:0 activate
Jan 1 16:53:44.156 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 update active ident from 10.0.0.3:0 to
10.0.0.4:0
Jan 1 16:53:44.156 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 ACEL 10.0.0.3:0 deactivate
Jan 1 16:53:44.256 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 ACEL 10.0.0.3:0 delete delay timer
expired, delete pending TRUE
Jan 1 16:53:44.256 MLDP LSP 0/0/CPU0 t21 FWD : 0x00001 Label 24008 delete, Success
Jan 1 16:53:44.256 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 ACEL 10.0.0.3:0 binding list Local
Delete
Jan 1 16:53:44.256 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 Released label 24008 to LSD
Jan 1 16:53:44.256 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 P2MP label withdraw msg to 10.0.0.3:0
Success
Jan 1 16:53:44.256 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 ACEL 10.0.0.3:0 remove
Jan 1 16:53:44.256 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 P2MP label release from 10.0.0.3:0
label 24008 root 10.0.0.5 Opaque Len 11
Jan 1 16:53:44.356 MLDP LSP 0/0/CPU0 t21 DB : 0x00001 MBB notification delay timer expired
```

MBB for Protection

The mLDP protection is composed of two main parts: the protection itself and MBB (Make-Before-Break).

Protection

The protection of mLDP traffic is similar to the protection mechanisms of unicast MPLS traffic. As soon as a link failure is detected, the PLR router switches the multicast traffic from trees crossing that link to the backup path. This backup path is a precomputed path that is installed in the forwarding plane. So, as soon as the failure occurs, the multicast traffic can be switched immediately to the backup path.

The protection is for link-down only. There is no node protection for mLDP.

The link-down event must be detected very quickly. This means that BFD (Bidirectional Forwarding Detection) must be used.

MBB

After the protection kicks in, the multicast traffic does not stay on the backup path forever. The traffic must be switched over to a newly calculated native mLDP tree/path. This switchover must occur in such a way that no multicast traffic is lost. MBB is used for this, so that the traffic is only switched over when the newly signalled tree is completely set up and is forwarding traffic. The MP router can then safely switch forwarding the traffic from the old backup tree to the newly signalled tree without traffic loss.

Theory of mLDP Protection

Look at image 6. It shows a network with a link R1 – R2 which is protected with Ti-LFA.

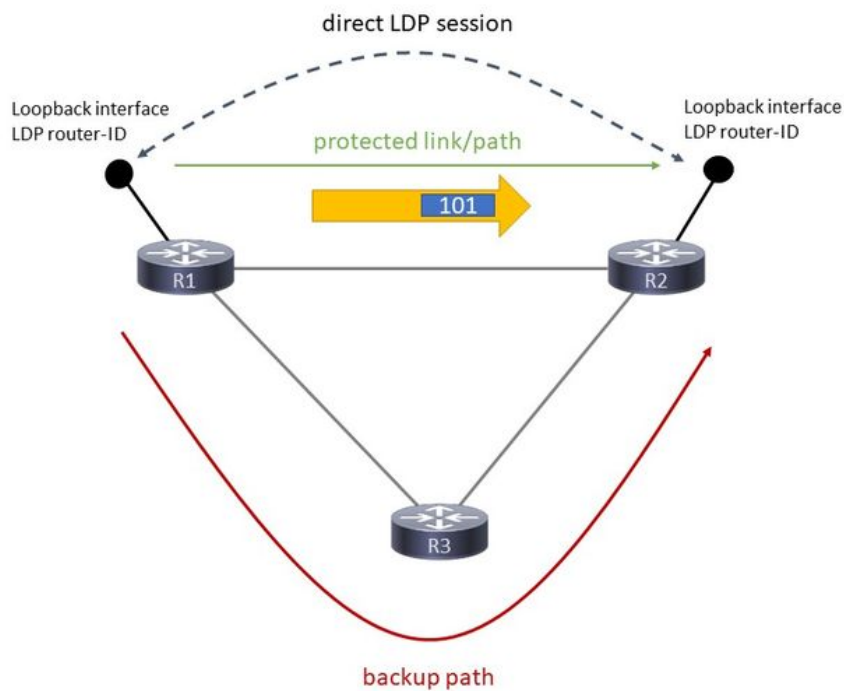


Image 6

The mLDP traffic is forwarded over the link R1 – R2. FRR calculates and installs a backup path via R3.

Look at image 7.

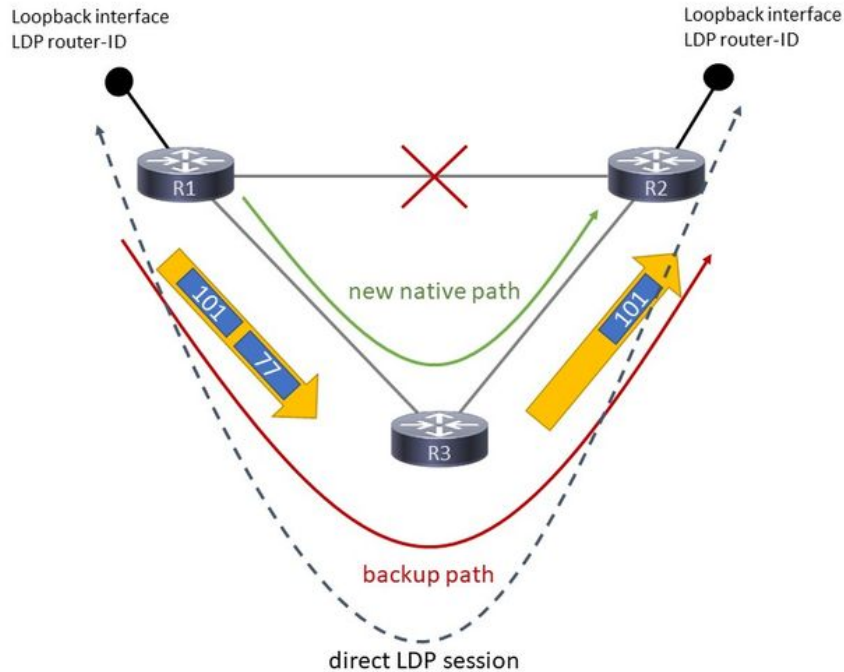


Image 7

Image 7 shows the situation when the protection is active.

When the link R1 – R2 goes down, the LDP session across it, is kept alive by LDP session protection. The LDP session - which is a TCP session – reroutes over R3. This avoid that the label bindings for LDP and mLDP between R1 and R2 are removed. For this LDP session to be able to be routed over R3 and be multi-hop, it must be a targeted LDP session. This is automatically done when LDP session protection is configured.

When the link R1 – R2 goes down, the mLDP traffic can be rerouted in a fast way over R3. For this to work, there must be some form of protection on R1 for the route towards the LDP router-ID of R2. This is achieved by either enabling MPLS Traffic Engineering tunnels, LFA (Loop-free Alternate), or Ti-LFA (Topology Independent LFA). The multicast traffic from R1 to R2 had one mLDP label. When the link R1 – 2 goes down, the multicast traffic gets an extra label when sent to R2. There is Penultimate Hop Popping (PHP), so the traffic is forwarded with one label towards R2. R2 receives this traffic with the same label as when the R1 – R2 link was up. R2 keeps on forwarding this multicast traffic.

This protection is fast. While there is protection for the mLDP traffic, R2 starts to signal a new native path from it towards R1 via R3. So, R2 sends an mLDP label mapping message to R3. R3 does the same towards R1. This is the same process/signalling as always when creating a new mLDP path. While this signalling is going on, R2 keeps forwarding the traffic from the backup mLDP path. When does R2 start forwarding the traffic from the newly created native path? The trigger can be two things: a timed delay or a signalling trigger. The timed delay is something configured. The signalling trigger is the Make-Before-Break behavior (MBB) introduced in mLDP and specified in RFC 6388. When R2 receives the signal from R1, it indicates that the newly native mLDP path is ready, so R2 can start forwarding the traffic from that new mLDP path and stop forwarding the traffic from the backup path.

R1 is called the PLR (Point-of-Local-Repair), it is the router where the protected path and the newly signalled native path branch off. R2 is the MP (Merge Point), the router where the protected

path and the newly signalled native path merge again.

Look at image 8.

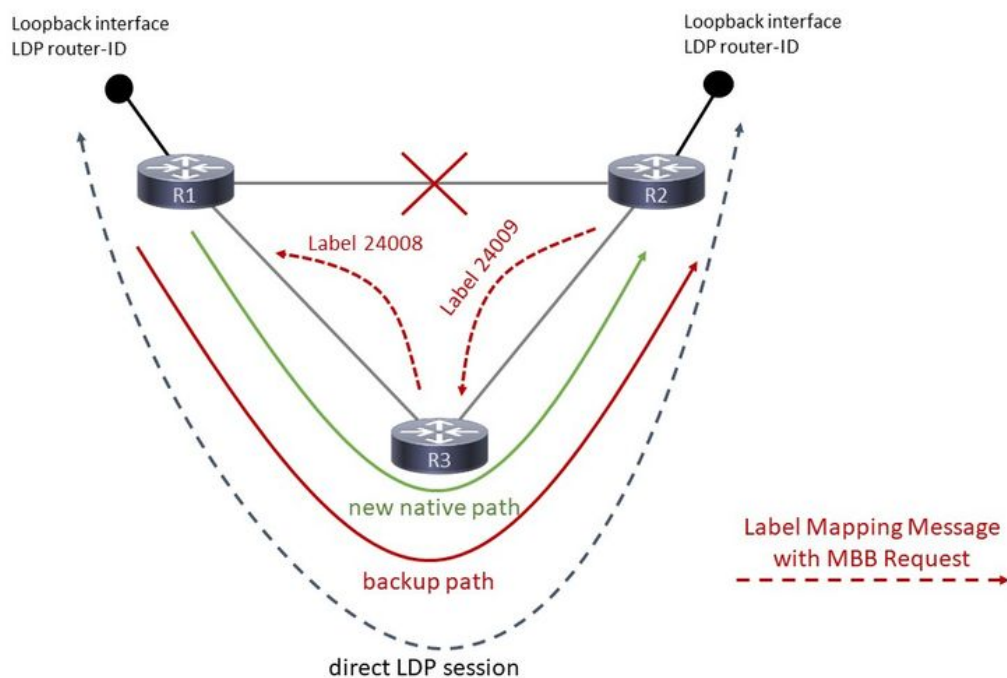


Image 8

Image 8 shows that there a mLDP Label Mapping message from R2 to R3, and from R3 to R1. This Label Mapping message has the MBB Request.

Look at image 9.

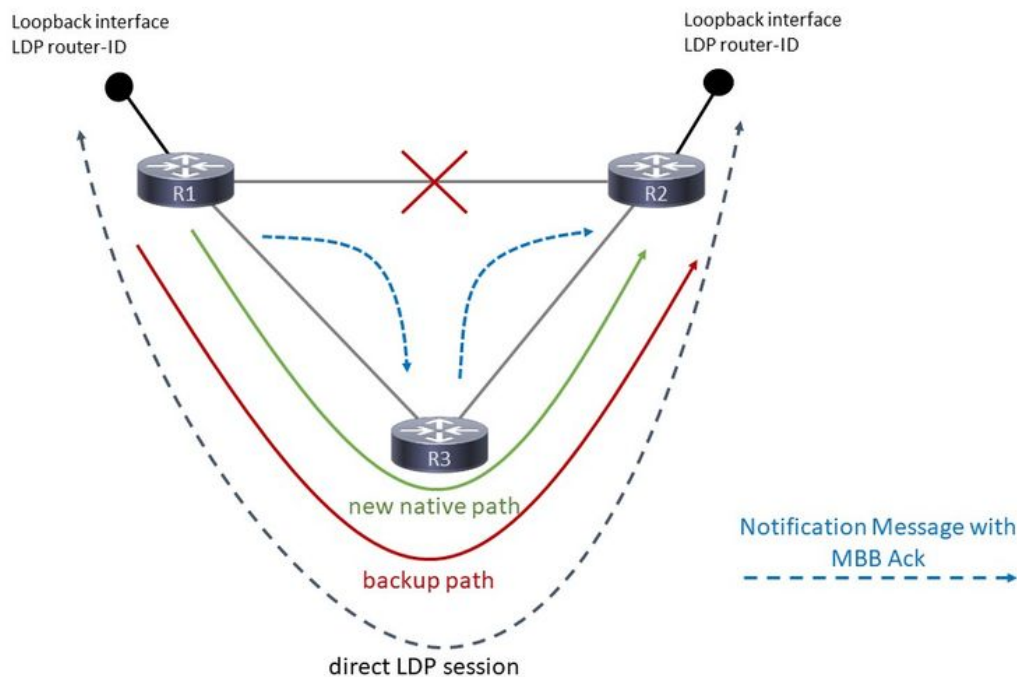


Image 9

R1 answers this signalling with a LDP Notification, carrying the MBB acknowledgement in the reverse direction. So, down the tree. This message travels from R1 to R3, and from R3 to R2. This signals R2, the MP router, that the new native mLDP path is ready. At this point R1 forwards the mLDP traffic twice, once on the backup path and once on the new native path

MBB is used here to have the MP (R2) switchover back to a native path (that was just created). When MBB has finished the signalling part, the MP stops forwarding the mLDP traffic arriving from the backup path and start forwarding the traffic from the newly signalled native path. The MBB is used here to indicate when this newly signalled path is ready. Another possibility is to configure a delay. In that case, the MP stops forwarding the mLDP traffic arriving from the backup path, and start forwarding the traffic from the newly signalled native path after the MBB has signalled that this new native path is ready *and* after the configured delay timer.

When R2 starts forwarding the traffic from the new native path, it stop forwarding the traffic from the backup path and signals the teardown of the backup path by sending a LDP Label Withdraw message for the tree (and a LDP Label Release message).

An additional *delete* delay can be added to remove the old tree in order to allow the platform to program all the forwarding state to the line cards.

After this, there is only the newly signalled native tree. Look at image 10 to see the forwarding of mLDP traffic in this case.

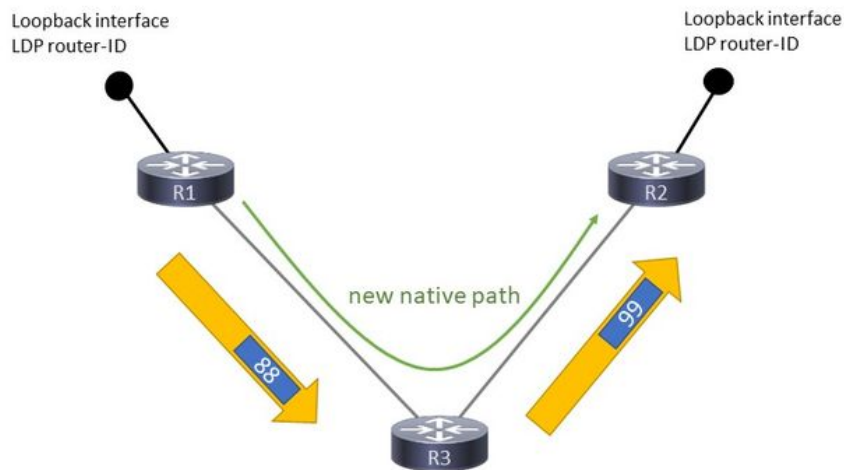


Image 10

Notice that the mLDP traffic has one MPLS label on top again.

Required Configuration

The next three configuration items are required for mLDP FRR (Fast ReRoute) to work.

You need:

- Recursive forwarding for mLDP enabled
- LDP session protection enabled
- LFA (Loop-free Alternate) or Ti-LFA (Topology Independent LFA) under the IGP (Ti-LFA requires Segment Routing). Point-to-Point Traffic Engineering is also possible.

If any of these three are missing, then there is no FRR protection for mLDP. mLDP protects only against link failures, not node failures.

Configuration example

```
mpls ldp
 log
 neighbor
 nsr
 graceful-restart
 session-protection
 !
 igp sync delay on-session-up 25
 mldp
 logging notifications
 address-family ipv4
  make-before-break delay 600 60      <<<<<<
  forwarding recursive              <<<<<<
 !
 !
 router-id 10.79.196.14
 neighbor
 dual-stack transport-connection prefer ipv4
 !
 session protection for LDP-PEERS    <<<<<<
 address-family ipv4
 label
  local
  allocate for host-routes
 !
 !
 !
```

The make-before-break command is optional.

Check that the outgoing interface is protected by LFA or Ti-LFA:

```
router isis IGP
 set-overload-bit on-startup 600
 net 49.0010.0000.0000.0001.00
 segment-routing global-block 100000 150000
 nsf cisco
 log adjacency changes
 lsp-gen-interval maximum-wait 5000 initial-wait 1 secondary-wait 50
 lsp-refresh-interval 1800
 max-lsp-lifetime 1880
 address-family ipv4 unicast
```

```

metric-style wide
fast-reroute per-prefix priority-limit critical
fast-reroute per-prefix tiebreaker lowest-backup-metric index 20
fast-reroute per-prefix tiebreaker node-protecting index 30
fast-reroute per-prefix tiebreaker srlg-disjoint index 10
mpls traffic-eng level-2-only
mpls traffic-eng router-id Loopback145
mpls traffic-eng multicast-intact
spf-interval maximum-wait 7000 initial-wait 1 secondary-wait 50
segment-routing mpls sr-prefer
segment-routing prefix-sid-map advertise-local
spf prefix-priority critical tag 17
mpls ldp auto-config
!
address-family ipv6 unicast
metric-style wide
fast-reroute per-prefix priority-limit critical
fast-reroute per-prefix tiebreaker lowest-backup-metric index 20
fast-reroute per-prefix tiebreaker node-protecting index 30
fast-reroute per-prefix tiebreaker srlg-disjoint index 10
spf-interval maximum-wait 7000 initial-wait 1 secondary-wait 50
segment-routing mpls sr-prefer
spf prefix-priority critical tag 17
!
interface Bundle-Ether10362
circuit-type level-2-only
point-to-point
address-family ipv4 unicast
    fast-reroute per-prefix          <<<<<<
    fast-reroute per-prefix ti-lfa  <<<<<<
metric 420 level 2
mpls ldp sync level 2
!
address-family ipv6 unicast
    fast-reroute per-prefix
    fast-reroute per-prefix ti-lfa
metric 420 level 2
!

```

There is no impact on the protection of the multicast traffic if any of the routers along the new native path do not have MBB configured. The protection is only dependent on the configuration of LDP session protection, recursive forwarding, and FRR, on the PLR. The MBB configuration on the newly native path routers only have a consequence when the traffic is switched from the backup path to the newly signalled tree. If a mLDP router received a Label Mapping message with MBB Request from a downstream router and needs to send a Label Mapping message to an upstream router, but that upstream router does not have MBB enabled, then the mLDP router sends a LDP Notification message to this downstream router as soon as it has sent the Label Mapping message (without the MBB Request) to the upstream router. As such, a regular mLDP tree is the result.

MBB for FRR Example

Look at image 11 for the topology.

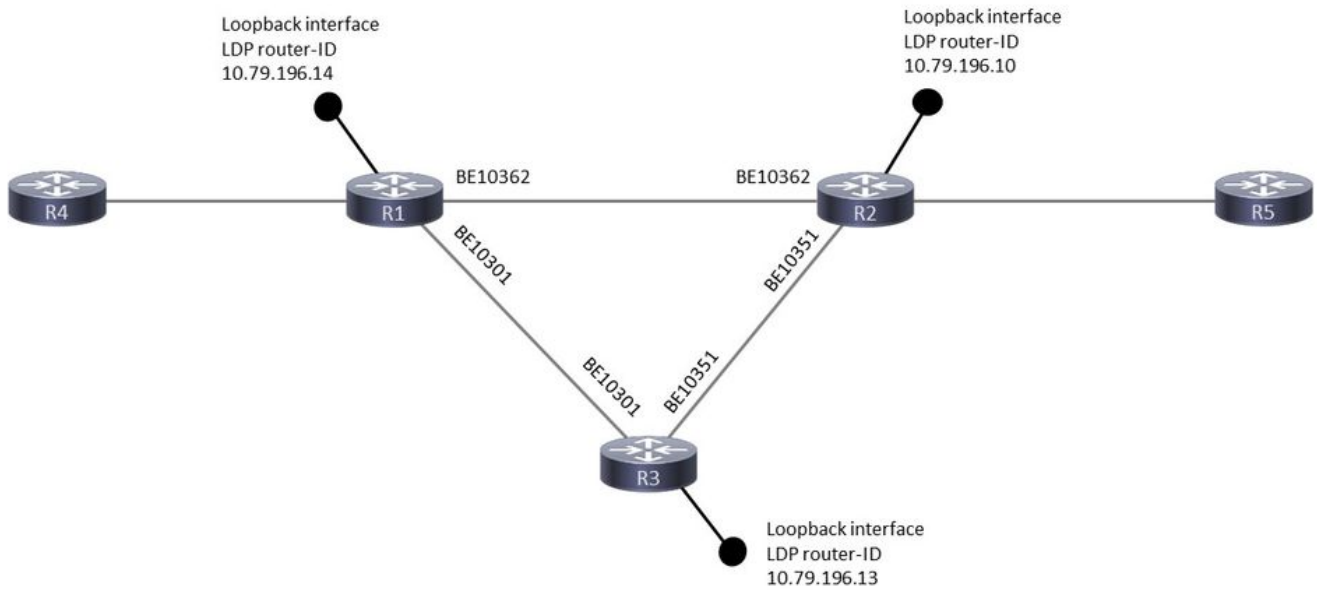


Image 11

When the Link fails between R1 and R2, then the mLDP session between them is protected by a LDP targeted session between them over R3. So, the mLDP session between R1 and R2 remains up even when the link between them is down. This protects the mLDP label bindings between them, they are kept. When the link R1-R2 goes down, the forwarding plane immediately switches over: the outgoing link R1-R2 switches to link R1-R3 in a very fast way, thanks to the Point-to-Point MPLS TE, LFA or Ti-LFA in place. This P2P MPLS TE, LFA or Ti-LFA must protect on R1 the route to the LDP router-ID of R2 in order to switch the forwarding entries for mLDP in a correct way. Finally, the recursive forwarding is needed because the mLDP session switches from a directly connected session, to a remote session, where the LDP router-ID is resolved in a recursive way.

R1 is called the PLR (Point-of-Local-Repair), it is the router where the protected path and the newly signalled native path branch off. R2 is the MP (Merge Point), the router where the protected path and the newly signalled native path merge again.

Check the three requirements:

-LDP protection

For the directly connected LDP (mLDP) neighbor over Bundle-Ether10362, there must also be targeted hello's:

```
RP/0/RP0/CPU0:R1#show mpls ldp discovery 10.79.196.10
```

```
Local LDP Identifier: 10.79.196.14:0
```

```
Discovery Sources:
```

```
Interfaces:
```

```
Bundle-Ether10362 : xmit/rcv
```

```
VRF: 'default' (0x60000000)
```

```
LDP Id: 10.79.196.10:0, Transport address: 10.79.196.10
```

Hold time: 15 sec (local:15 sec, peer:15 sec)
Established: Dec 28 10:23:16.144 (00:02:13 ao)

Targeted Hellos:

10.79.196.14 -> 10.79.196.10 (active), xmit/rcv
LDP Id: 10.79.196.10:0
Hold time: 90 sec (local:90 sec, peer:90 sec)
Established: Dec 28 10:23:30.008 (00:01:59 ago)

-LFA or Ti-LFA under the IGP

Check that the route to the LDP neighbor router-id has a backup path. The RIB (Routing Information Base) and FIB (Forwarding Information Base or CEF) must have this backup path:

RP/0/RP0/CPU0:R1#**show route 10.79.196.10**

Routing entry for 10.79.196.10/32
Known via "isis IGP", distance 115, metric 420, labeled SR
Tag 17, type level-2
Installed Dec 28 10:23:42.659 for 00:07:58
Routing Descriptor Blocks
 10.254.1.144, from 10.79.196.10, **via Bundle-Ether10301, Backup (Local-LFA)**
 Route metric is 2000
 10.254.3.37, from 10.79.196.10, **via Bundle-Ether10362, Protected**
 Route metric is 420
No advertising protos.

RP/0/RP0/CPU0:R1#**show cef 10.79.196.10**

10.79.196.10/32, version 7364, labeled SR, internal 0x1000001 0x83 (ptr 0x788e1f78) [1], 0x0 (0x788ab5a8), 0xa28 (0x79dd1138)
Updated Oct 25 11:32:44.299
Prefix Len 32, traffic index 0, precedence n/a, priority 1
 via 10.254.1.144/32, **Bundle-Ether10301**, 11 dependencies, weight 0, class 0, **backup (Local-LFA)** [flags 0x300]
 path-idx 0 NHID 0x0 [0x78f4e9b0 0x0]
 next hop 10.254.1.144/32
 local adjacency
 local label 100010 labels imposed {100010}
 via 10.254.3.37/32, **Bundle-Ether10362**, 11 dependencies, weight 0, class 0, **protected** [flags 0x400]
 path-idx 1 bkup-idx 0 NHID 0x0 [0x7905e510 0x7905e350]
 next hop 10.254.3.37/32
 local label 100010 labels imposed {ImplNull}

-recursive forwarding for mLDP

The mLDP database entry does not have an outgoing interface in the LFIB if recursive forwarding is applied:

Without recursive forwarding:

RP/0/RP0/CPU0:R1#**show mpls forwarding labels 25426**

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
25426	24440	mLDP/IR: 0x00001	BE10362	10.254.3.37	7893474

With recursive forwarding:

```
RP/0/RP0/CPU0:R1#show mpls forwarding labels 25426
```

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
25426	24440	mLDP/IR: 0x00001		10.79.196.10	2516786878

Notice that there is no outgoing interface anymore for the mLDP forwarding entry. This makes troubleshoot a bit harder.

The MP has the next configuration for mLDP. Note the timers 600 sec and 60 sec. The PLR has the same timers. The PLR forwards the traffic over the backup path *and* the native path for 600 seconds. The 600 seconds delay means that the MP forwards the traffic from the backup path for 600 sec, while dropping the traffic arriving from the native path. 600 seconds is a long time for this timer. It was used in a lab environment in order to provide enough time to capture the output with show commands. The 60 sec delay means that the MP waits the deletion of the MBB path for 60 seconds after it starts forwarding the traffic arriving from the native path and dropping the traffic arriving over the backup path. The correct value for these two delays depends on the network. It needs to be derived from testing the specific network, software, and hardware.

```
mpls ldp
log
neighbor
nsr
graceful-restart
session-protection
!
igp sync delay on-session-up 25
mldp
logging notifications
address-family ipv4
    make-before-break delay 600 60
    forwarding recursive
!
!
router-id 10.79.196.10
neighbor
dual-stack transport-connection prefer ipv4
!
session protection for LDP-PEERS
address-family ipv4
label
    local
        allocate for LDP-PEERS
!
!
!
```

MBB in use

Look at image 12, it shows the forwarding while mLDP is in protecting mode.

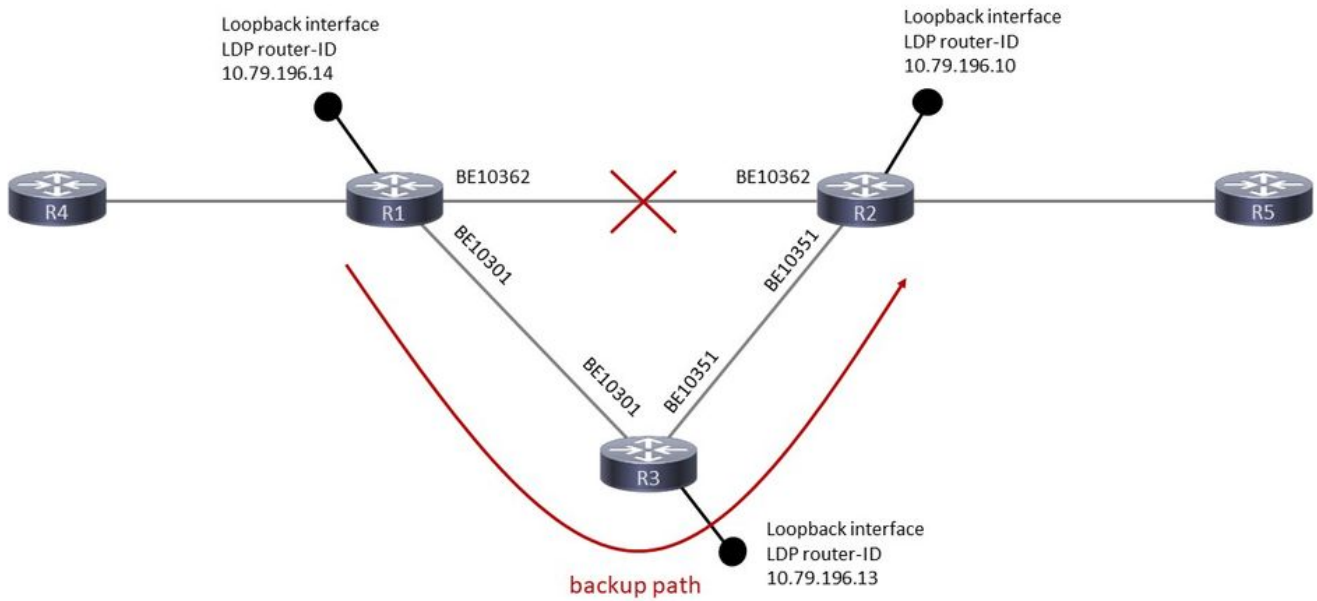


Image 12

Before the outgoing interface is down, this is the LFIB entry for the remote LDP router-ID (R2):

```
RP/0/RP0/CPU0:R1#show mpls forwarding labels 100010
Local  Outgoing  Prefix          Outgoing  Next Hop      Bytes
Label  Label      or ID          Interface  Interface     Switched
-----
100010 Pop          SR Pfx (idx 10) BE10362    10.254.3.37  355616309429
        100010      SR Pfx (idx 10) BE10301    10.254.1.144 0                (!)
```

The (!) indicates a backup path.

This is the mLDP tree database entry on the PLR:

```
RP/0/RP0/CPU0:R1#show mpls mldp database details
mLDP database
LSM-ID: 0x00001  Type: P2MP  Uptime: 3d03h
FEC Root       : 10.79.196.14 (we are the root)
FEC Length     : 12 bytes
FEC Value internal : 02010004000000015C4FC40E
Opaque length  : 4 bytes
Opaque value   : 01 0004 00000001
Opaque decoded : [global-id 1]
Features       : MBB RFWF Trace
Upstream neighbor(s) :
None
Downstream client(s):
LDP 10.79.196.10:0 Uptime: 02:09:09
  Rec Next Hop    : 10.79.196.10
  Remote label (D) : 24440
  LDP MSG ID      : 254705
PIM MDT          Uptime: 3d03h
  Egress intf    : Lmdtvrfone
  Table ID       : IPv4: 0xe0000014 IPv6: 0xe0800014
```

```

HLI                : 0x00001
Ingress            : Yes
Peek               : Yes
PPMP               : Yes

```

This is the mLDP forwarding entry for the tree:

```

RP/0/RP0/CPU0:R1#show mpls mldp forwarding label 25426
mLDP MPLS forwarding database

```

```

25426 LSM-ID: 0x00001 HLI: 0x00001 flags: In Pk
  Lmdtvrfone, RPF-ID: 0, TIDv4: E0000014, TIDv6: E0800014
  24440, NH: 10.79.196.10, Intf: Role: H, Flags: 0x4      Local Label      : 25426 (internal)

```

This is the LFIB (Label Forwarding Instance Base) forwarding entry for the tree:

```

RP/0/RP0/CPU0:R1#show mpls for labels 25426

```

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
25426	24440	mLDP/IR: 0x00001		10.79.196.10	0

The mLDP forwarding entry is protected. The forwarding entry is protected via label 100010, the entry for the remote LDP router ID.

```

RP/0/RP0/CPU0:R1#show mpls for labels 25426 detail

```

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
25426		mLDP/IR: 0x00001 (0x00001)			

```

Updated Dec 28 10:23:42.669
mLDP/IR LSM-ID: 0x00001, MDT: 0x2000660, Head LSM-ID: 0x00001
IPv4 Tableid: 0xe0000014, IPv6 Tableid: 0xe0800014
Flags:IP Lookup:set, Expnulv4:not-set, Expnulv6:not-set
  Payload Type v4:not-set, Payload Type v6:not-set, l2vpn:not-set
  Head:set, Tail:not-set, Bud:not-set, Peek:set, inclusive:not-set
  Ingress Drop:not-set, Egress Drop:not-set
  RPF-ID:0, Encap-ID:0
  Disp-Tun:[ifh:0x0, label:-]
  Platform Data [64]:
    { 0 0 0 96 0 0 0 96
      0 0 0 0 0 0 0 0
      0 0 0 0 0 0 0 0
      0 0 0 96 0 0 0 96
      0 0 0 0 0 0 0 0
      0 0 0 0 0 0 0 0
      0 0 2 9 0 0 2 10
      0 0 0 1 0 0 0 1
    }

```

mpls paths: 1, local mpls paths: 0, **protected mpls paths:**

```

24440      mLDP/IR: 0x00001 (0x00001)  \
                                           10.79.196.10  0

Updated: Dec 28 10:23:42.670
My Nodeid:0x20
Interface Nodeids:
 [ 0x8620 - - - - - ]
Interface Handles:
 [ 0xc0001c0 - - - - - ]

```



```

Backup Interface Nodeids:
  [ 0x8520 - - - - - ]
Backup Interface Handles:
  [ 0xa000400 - - - - - ]
via-label:100010, mpi-flags:0x0 tos_masks:[ primary:0x0 backup:0x0]
Packets Switched: 0

```

This is the forwarding entry in hardware. The routers are ASR9k routers.

RP/0/RP0/CPU0:R1#show mpls for labels 25426 detail hardware ingress location 0/2/CPU0

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
-------------	----------------	--------------	--------------------	----------	----------------

```

-----
25426          mLDP/IR: 0x00001 (0x00001)
Updated Dec 28 10:23:42.674
mLDP/IR LSM-ID: 0x00001, MDT: 0x2000660, Head LSM-ID: 0x00001
IPv4 Tableid: 0xe0000014, IPv6 Tableid: 0xe0800014
Flags:IP Lookup:set, Expnnullv4:not-set, Expnnullv6:not-set
  Payload Type v4:not-set, Payload Type v6:not-set, l2vpn:not-set
  Head:set, Tail:not-set, Bud:not-set, Peek:set, inclusive:not-set
  Ingress Drop:not-set, Egress Drop:not-set
  RPF-ID:0, Encap-ID:0
  Disp-Tun:[ifh:0x0, label:-]
  Platform Data [64]:
    { 0 0 0 96 0 0 0 96
      0 0 0 0 0 0 0 0
      0 0 0 0 0 0 0 0
      0 0 0 96 0 0 0 96
      0 0 0 0 0 0 0 0
      0 0 0 0 0 0 0 0
      0 0 2 9 0 0 2 10
      0 0 0 1 0 0 0 1
    }
mpls paths: 1, local mpls paths: 0, protected mpls paths: 1

```

```

24440          mLDP/IR: 0x00001 (0x00001) \
                                     10.79.196.10  N/A
Updated: Dec 28 10:23:42.674
My Nodeid:0x8420
Interface Nodeids:
  [ 0x8620 - - - - - ]
Interface Handles:
  [ 0xc0001c0 - - - - - ]
Backup Interface Nodeids:
  [ 0x8520 - - - - - ]
Backup Interface Handles:
  [ 0xa000400 - - - - - ]
via-label:100010, mpi-flags:0x0 tos_masks:[ primary:0x0 backup:0x0]
Packets Switched: 0

```

```

LEAF - HAL pd context :
sub-type : MPLS_P2MP, ecd_marked:0, has_collapsed_ldi:0
collapse_bwalk_required:0, ecdv2_marked:0,

```

Leaf H/W Result

```

Leaf H/W Result on NP:0
09000014000000921806352100020900006000020a0000600000a00001010400
vpn_special          =          0 (0x0)
vc_label_vpws       =          0 (0x0)
vc_label_vpls       =          0 (0x0)
pwhe                 =          0 (0x0)
p2mp                =          1 (0x1)

```

```

tp = 0 (0x0)
recursive = 0 (0x0)
non_recursive = 1 (0x1)
flow_label_dispose = 0 (0x0)
receive_entry_type = 0 (0x0)
control_word_enabled = 0 (0x0)
imp_ttl_255 = 0 (0x0)
collapsed = 0 (0x0)
recursive_lsp_stats = 0 (0x0)
vpn_key = 20 (0x14)
Non-recursive:
  rpf_id = 0 (0x0)
  nrldi_ptr = 406817 (0x63521)
P2MP:
  rpf_id = 146 (0x92)
  nrldi_ptr = 146 (0x92)
  mldp_egr_drop = 0 (0x0)
  mldp_ing_drop = 0 (0x0)
  mldp_signal = 0 (0x0)
  mldp_peek = 1 (0x1)
  mldp_tunnel = 1 (0x1)
  p2mp_bud_node = 0 (0x0)
  p2mp_ip_lookup = 0 (0x0)
  per_lc_receivers = 0 (0x0)
  igp_local_label: eos = 1 (0x1)
  igp_local_label: exp = 0 (0x0)
  igp_local_label: label = 25426 (0x6352)
fab_info: fab_mgid = 521 (0x209)
fab_info: fab_slotmask = 96 (0x60)
fab_info: fab_fgid = 150995040 (0x9000060)
backup_fab_info: backup_fab_mgid = 522 (0x20a)
backup_fab_info: backup_fab_slotmask= 96 (0x60)
backup_fab_info: backup_fab_fgid = 167772256 (0xa000060)
rep_node_ndx = 40960 (0xa000)
ecmp_size = 1 (0x1)
stats_ptr = 66560 (0x10400)

```

Leaf H/W Result on NP:1

09000014000000921806352100020900006000020a0000600000a00001010400

...

There is the FGID (Fabric Group Index) and backup FGID. The FGID is used by the switch fabric to forward the multicast traffic to the correct line cards. There is also the MGID (Multicast Group Identifier). The MGID is used to forward the multicast traffic to the correct replication elements on the line cards.

RP/0/RP0/CPU0:R1#**show mrib encap-id**

```

Encap ID Key      : 00000101000000600600020100000000000002
Encap ID Length  : 19
Encap ID Value   : 262145

```

Platform Annotation:

```

Slotmask: Primary: 0x40, Backup: 0x60
MGID:     Primary: 64059, Backup: 64060

```

Flags (Vrflite(v4/v6),Stale,v6): N/N, N, N

Oles:

[1] type: 0x5, len: 12

LSM-ID: 0x00001 MDT: 0x2000660 Turnaround: TRUE

Primary: 0/4/CPU0[1]

Backup: 0/3/CPU0[1]

TableId: 0xe0000014[1001]

Redist History:

client id 31 redist time: 02:01:27 redist flags 0x0

This is how you can look up the MGID entry:

RP0/RP0/CPU0:R1#show controllers mgidprgm mgidindex 521 location 0/2/CPU0

Device	MGID-Bits	Client-Last-Modified
XBAR-0	1	P2MP
XBAR-1	1	P2MP
FIA-0	1	P2MP
FIA-1	0	None
FIA-2	0	None
FIA-3	0	None
FIA-4	0	None
FIA-5	0	None
FIA-6	0	None
FIA-7	0	None

Client	Mask
MFIBV4	0x0
MFIBV6	0x0
L2FIB	0x0
sRP-pseudo-mc	0x0
UT	0x0
Prgm-Svr	0x0
P2MP	0x1
xbar	0x0
UT1	0x0
UT2	0x0
punt_lib	0x0

RP0/RP0/CPU0:R1#show controllers mgidprgm mgidindex 522 location 0/2/CPU0

Device	MGID-Bits	Client-Last-Modified
XBAR-0	1	P2MP
XBAR-1	1	P2MP
FIA-0	1	P2MP
FIA-1	0	None
FIA-2	0	None
FIA-3	0	None
FIA-4	0	Non
FIA-5	0	None
FIA-6	0	None
FIA-7	0	None

Client	Mask
MFIBV4	0x0
MFIBV6	0x0
L2FIB	0x0
sRP-pseudo-mc	0x0
UT	0x0
Prgm-Svr	0x0
P2MP	0x1
xbar	0x0
UT1	0x0
UT2	0x0

punt_lib 0x0

The outgoing interface is now down, and MBB is in use.

Image 13 shows the signalling.

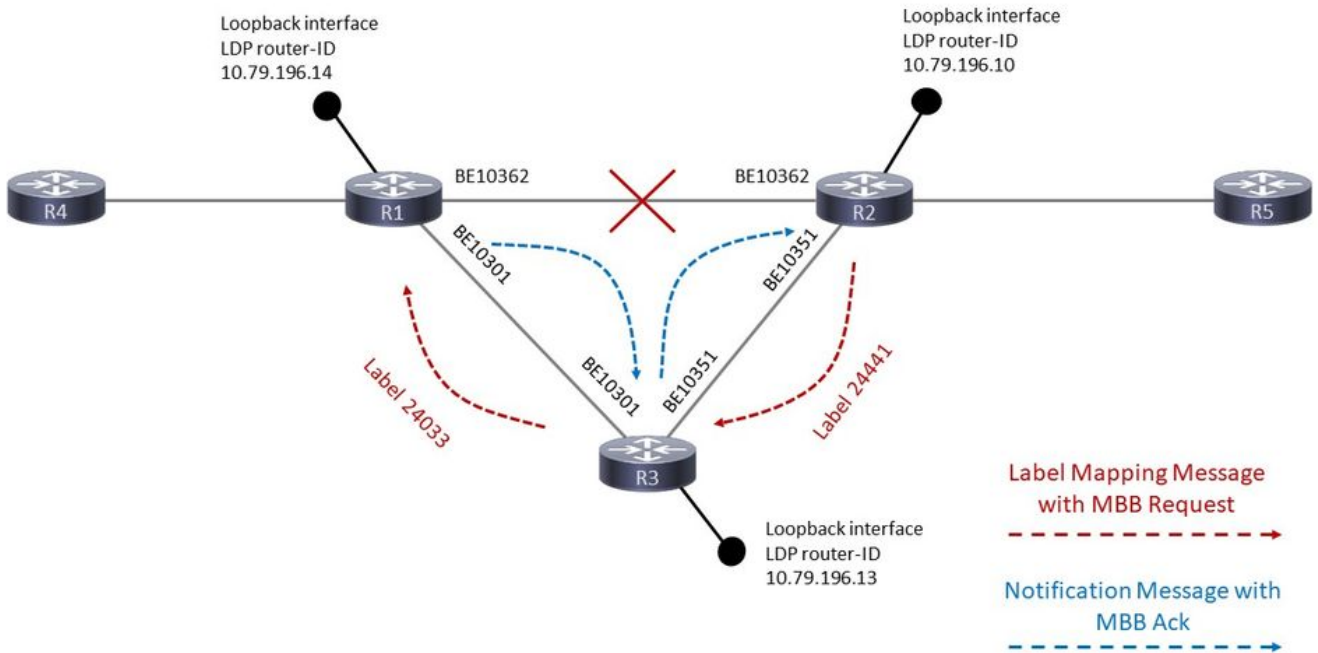


Image 13

R1 now has two forwarding entries for this tree:

```
RP/0/RP0/CPU0:R1#show mpls forwarding labels 25426
```

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
25426	24440	mLDP/IR: 0x00001		10.79.196.10	1834250032
	24033	mLDP/IR: 0x00001		10.79.196.13	1825230386

```
RP/0/RP0/CPU0:R1#show mpls forwarding labels 25426 detail
```

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
25426		mLDP/IR: 0x00001 (0x00001)			
Updated Dec 28 13:07:03.417					
mLDP/IR LSM-ID: 0x00001, MDT: 0x2000660, Head LSM-ID: 0x00001					
IPv4 Tableid: 0xe000014, IPv6 Tableid: 0xe080014					
Flags:IP Lookup:set, Expnulv4:not-set, Expnulv6:not-set					
Payload Type v4:not-set, Payload Type v6:not-set, l2vpn:not-set					
Head:set, Tail:not-set, Bud:not-set, Peek:set, inclusive:not-set					
Ingress Drop:not-set, Egress Drop:not-set					
RPF-ID:0, Encap-ID:0					
Disp-Tun:[ifh:0x0, label:-]					
Platform Data [64]:					
{ 0 0 0 96 0 0 0 96					
0 0 0 0 0 0 0 0					

```
0 0 0 0 0 0 0 0
0 0 0 96 0 0 0 96
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 2 9 0 0 2 10
0 0 0 1 0 0 0 1
}
```

mpls paths: 2, local mpls paths: 0, protected mpls paths:

```
24440      mLDP/IR: 0x00001 (0x00001)  \
                                           10.79.196.10   2230150704
```

```
Updated: Dec 28 13:07:03.245
My Nodeid:0x20
Interface Nodeids:
  [ 0x8520 - - - - - ]
Interface Handles:
  [ 0xa000400 - - - - - ]
Backup Interface Nodeids:
  [ - - - - - ]
Backup Interface Handles:
  [ - - - - - ]
via-label:100010, mpi-flags:0x0 tos_masks:[ primary:0x0 backup:0x0]
Packets Switched: 21039158
```

```
24033      mLDP/IR: 0x00001 (0x00001)  \
                                           10.79.196.13   2221131058
```

```
Updated: Dec 28 13:07:03.417
My Nodeid:0x20
Interface Nodeids:
  [ 0x8520 - - - - - ]
Interface Handles:
  [ 0xa000400 - - - - - ]
Backup Interface Nodeids:
  [ - - - - - ]
Backup Interface Handles:
  [ - - - - - ]
via-label:100013, mpi-flags:0x0 tos_masks:[ primary:0x0 backup:0x0]
Packets Switched: 20954067
```

There are two downstream mLDP clients, R2 and R3:

```
RP/0/RP0/CPU0:R1#show mpls mldp database details
mLDP database
LSM-ID: 0x00001  Type: P2MP  Uptime: 3d04h
FEC Root          : 10.79.196.14 (we are the root)
FEC Length        : 12 bytes
FEC Value internal : 02010004000000015C4FC40E
Opaque length     : 4 bytes
Opaque value      : 01 0004 00000001
Opaque decoded    : [global-id 1]
Features          : MBB RFWF Trace
Upstream neighbor(s) :
  None
Downstream client(s):
LDP 10.79.196.10:0 Uptime: 02:44:09
  Rec Next Hop      : 10.79.196.10
  Remote label (D)  : 24440
  LDP MSG ID        : 254705
LDP 10.79.196.13:0 Uptime: 00:00:48
  Rec Next Hop      : 10.79.196.13
  Remote label (D)  : 24033
  LDP MSG ID        : 98489
PIM MDT           : Uptime: 3d04h
Egress intf       : Lmdtvrfone
```

```

Table ID       : IPv4: 0xe0000014 IPv6: 0xe0800014
HLI           : 0x000001
Ingress       : Yes
Peek          : Yes
PPMP          : Yes
Local Label   : 25426 (internal)

```

The MP (R2) has two upstream neighbors, one is active, the other is inactive:

P/0/RSP1/CPU0:R2#show mpls mldp database details

```

LSM-ID: 0x00002  Type: P2MP  Uptime: 03:45:22
FEC Root       : 10.79.196.14
FEC Length     : 12 bytes
FEC Value internal : 02010004000000015C4FC40E
Opaque length  : 4 bytes
Opaque value    : 01 0004 00000001
Opaque decoded  : [global-id 1]
MBB nbr evaluate : 00:08:18
Features       : MBB RFW Trace
Upstream neighbor(s) :
Is CSI accepting : N
  10.79.196.13:0 [Inactive] [MBB] Uptime: 00:01:42
    Local Label (D) : 24441
Is CSI accepting : N
  10.79.196.14:0 [Active] [Delete] [MBB] Uptime: 02:45:02
    Local Label (D) : 24440
Downstream client(s):
PIM MDT       Uptime: 03:45:22
Egress intf   : Lmdtvrfone
Table ID      : IPv4: 0xe0000013 IPv6: 0xe0800013
RPF ID       : 3
Peek         : Yes
RD           : 3209:92722001

```

The backup interface is gone on R1:

RP/0/RP0/CPU0:R1#show mpls for labels 25426 detail hardware ingress location 0/2/CPU0

Local Label	Outgoing Label	Prefix or ID	Outgoing Interface	Next Hop	Bytes Switched
25426		mLDP/IR: 0x00001 (0x00001)			
Updated Dec 28 13:07:03.418					
mLDP/IR LSM-ID: 0x00001, MDT: 0x2000660, Head LSM-ID: 0x00001					
IPv4 Tableid: 0xe0000014, IPv6 Tableid: 0xe0800014					
Flags:IP Lookup:set, Expnulv4:not-set, Expnulv6:not-set					
Payload Type v4:not-set, Payload Type v6:not-set, l2vpn:not-set					
Head:set, Tail:not-set, Bud:not-set, Peek:set, inclusive:not-set					
Ingress Drop:not-set, Egress Drop:not-set					
RPF-ID:0, Encap-ID:0					
Disp-Tun:[ifh:0x0, label:-]					
Platform Data [64]:					
{ 0 0 0 96 0 0 0 96					
0 0 0 0 0 0 0 0					
0 0 0 0 0 0 0 0					
0 0 0 96 0 0 0 96					
0 0 0 0 0 0 0 0					
0 0 0 0 0 0 0 0					
0 0 2 9 0 0 2 10					
0 0 0 1 0 0 0 1					
}					

mpls paths: 2, local mpls paths: 0, protected mpls paths:

24440 mLDP/IR: 0x00001 (0x00001) \ 10.79.196.10 N/A
Updated: Dec 28 13:07:03.255
My Nodeid:0x8420
Interface Nodeids:
 [0x8520 - - - - -]
Interface Handles:
 [0xa000400 - - - - -]
Backup Interface Nodeids:
 [- - - - -]
Backup Interface Handles:
 [- - - - -]
via-label:100010, mpi-flags:0x0 tos_masks:[primary:0x0 backup:0x0]
Packets Switched: 0

24033 mLDP/IR: 0x00001 (0x00001) \ 10.79.196.13 N/A
Updated: Dec 28 13:07:03.418
My Nodeid:0x8420
Interface Nodeids:
 [0x8520 - - - - -]
Interface Handles:
 [0xa000400 - - - - -]
Backup Interface Nodeids:
 [- - - - -]
Backup Interface Handles:
 [- - - - -]
via-label:100013, mpi-flags:0x0 tos_masks:[primary:0x0 backup:0x0]
Packets Switched: 0

RP/0/RP0/CPU0:R1#show mrib encap-id

Encap ID Key : 00000101000000600600020100000000000002
Encap ID Length : 19
Encap ID Value : 262145

Platform Annotation:
Slotmask: Primary: 0x20, Backup: 0x20
MGID: Primary: 64059, Backup: 64060

Flags (Vrflite(v4/v6),Stale,v6): N/N, N, N

Oles:

[1] type: 0x5, len: 12
LSM-ID: 0x00001 MDT: 0x2000660 Turnaround: TRUE

Primary: 0/3/CPU0[1]

Backup:

TableId: 0xe0000014[1001]

Redist History:

client id 31 redist time: 00:01:22 redist flags 0x0

The MP switched over to the newly signalled native tree, and it is within the 60 seconds before deleting the old tree:

RP/0/RSP1/CPU0:R2#show mpls mldp database details

LSM-ID: 0x00002 Type: P2MP Uptime: 03:53:56
FEC Root : 10.79.196.14
FEC Length : 12 bytes
FEC Value internal : 02010004000000015C4FC40E
Opaque length : 4 bytes
Opaque value : 01 0004 00000001

```
Opaque decoded      : [global-id 1]
Features            : MBB RFWF Trace
Upstream neighbor(s) :
Is CSI accepting    : N
  10.79.196.13:0 [Active] [MBB] Uptime: 00:10:16
    Local Label (D) : 24441
Is CSI accepting    : N
  10.79.196.14:0 [Inactive] [Delete 00:00:44] [MBB] Uptime: 02:53:37
    Local Label (D) : 24440
Downstream client(s):
  PIM MDT           Uptime: 03:53:56
  Egress intf       : Lmdtvrfone
  Table ID          : IPv4: 0xe0000013 IPv6: 0xe0800013
  RPF ID            : 3
  Peek              : Yes
  RD                : 3209:92722001
```

There is the state, after the old tree is deleted:

```
RP/0/RSP1/CPU0:R2#show mpls mldp database details
mLDP database
```

```
LSM-ID: 0x00002 Type: P2MP Uptime: 03:58:03
FEC Root          : 10.79.196.14
FEC Length        : 12 bytes
FEC Value internal : 02010004000000015C4FC40E
Opaque length     : 4 bytes
Opaque value      : 01 0004 00000001
Opaque decoded    : [global-id 1]
Features          : MBB RFWF Trace
Upstream neighbor(s) :
Is CSI accepting     : N
  10.79.196.13:0 [Active] [MBB] Uptime: 00:14:23
    Local Label (D) : 24441
Downstream client(s):
  PIM MDT           Uptime: 03:58:03
  Egress intf       : Lmdtvrfone
  Table ID          : IPv4: 0xe0000013 IPv6: 0xe0800013
  RPF ID            : 3
  Peek              : Yes
  RD                : 3209:92722001
```

The PLR only has one downstream mLDP client:

```
RP/0/RP0/CPU0:R1#show mpls mldp database details
mLDP database
LSM-ID: 0x00001 Type: P2MP Uptime: 3d04h
FEC Root          : 10.79.196.14 (we are the root)
FEC Length        : 12 bytes
FEC Value internal : 02010004000000015C4FC40E
Opaque length     : 4 bytes
Opaque value      : 01 0004 00000001
Opaque decoded    : [global-id 1]
Features          : MBB RFWF Trace
Upstream neighbor(s) :
  None
Downstream client(s):
  LDP 10.79.196.13:0 Uptime: 00:11:13
  Rec Next Hop    : 10.79.196.13
  Remote label (D) : 24033
  LDP MSG ID      : 98489
```



```

PIM MDT                Uptime: 3d04h
Egress intf            : Lmdtvrfone
Table ID               : IPv4: 0xe0000014 IPv6: 0xe0800014
HLI                   : 0x000001
Ingress               : Yes
Peek                  : Yes
PPMP                  : Yes
Local Label           : 25426 (internal)

```

mLDP Trace

The mLDP trace shows the events in more detail.

On the PLR

The interface BE10362 goes down:

```

Dec 28 13:07:03.220 MLDP GLO 0/RP0/CPU0 t10704 RIB : Read notification
Dec 28 13:07:03.225 MLDP GLO 0/RP0/CPU0 t10706 RIB : Notify client 'Peer' for prefix:
10.79.196.10/32
Dec 28 13:07:03.225 MLDP GLO 0/RP0/CPU0 t10706 GEN : Checkpoint save neighbor 10.79.196.10:0
canceled, no GR or NSR
Dec 28 13:07:03.227 MLDP GLO 0/RP0/CPU0 t10706 NBR : 10.79.196.10:0 delete adj
2000460/10.254.3.37
Dec 28 13:07:03.227 MLDP GLO 0/RP0/CPU0 t10706 GEN : Checkpoint delete neighbor adj
2000460/10.254.3.37 objid 0 version 0 Failed
Dec 28 13:07:03.227 MLDP GLO 0/RP0/CPU0 t10706 NBR : 10.79.196.10:0 LDP Adjacency addr:
10.254.3.37, Interface: Bundle-Ether10362 Delete
Dec 28 13:07:03.325 MLDP GLO 0/RP0/CPU0 t10706 NBR : 10.79.196.10:0 Check branches for path
change

```

The link was lost, but the LDP adjacency is not lost, it is kept as a targeted session.

The next entries are the new branch over the P router (10.79.196.13):

```

Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 DB : P2MP Label mapping from 10.79.196.13:0
label 24033 root 10.79.196.14 Opaque Len 7
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Add branch LDP 10.79.196.13:0
Label 24033
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Branch LDP 10.79.196.13:0 binding
list Remote Add
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Changing branch LDP 10.79.196.13:0
from None/0.0.0.0 to None/10.79.196.13
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Notify client Add event: 6 root
TRUE
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Add update to PIM Root TRUE
Upstream TRUE Ingress TRUE
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 FWD : 0x00001 Label 25426 add path label 24033
intf None nexthop 10.79.196.13 id 0x00001 Success
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 FWD : 0x00001 Label 25426 set HLI 0x00001
Success
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Notify client Add event: 6 root
TRUE
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Add update to PIM Root TRUE
Upstream TRUE Ingress TRUE
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 FWD : 0x00001 Label 25426 add path label 24033
intf None nexthop 10.79.196.13 id 0x00001 Success
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10706 FWD : 0x00001 Label 25426 set HLI 0x00001
Success
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10705 DB : 0x00001 Add event from mLDP to PIM, ready

```

```
TRUE root TRUE csc_rd 0:0 csc_umh 0.0.0.0, msg_len 50
Dec 28 13:07:03.401 MLDP LSP 0/RP0/CPU0 t10705 DB : 0x00001 Add event from mLDP to PIM, ready
TRUE root TRUE csc_rd 0:0 csc_umh 0.0.0.0, msg_len 50
Dec 28 13:07:05.296 MLDP GLO 0/RP0/CPU0 t10706 NBR : 10.79.196.10:0 to address: 10.254.3.37
mapping deleted
```

The rest is the clean-up. R3 sends the Label Withdraw message and Label Release message to R1:

```
Dec 28 13:18:04.635 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 P2MP label withdraw from
10.79.196.10:0 label 24440 root 10.79.196.14 Opaque Len 7
Dec 28 13:18:04.635 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 P2MP label release msg to
10.79.196.10:0 Success
Dec 28 13:18:04.635 MLDP LSP 0/RP0/CPU0 t10706 FWD : 0x00001 Label 25426 delete path label
24440 intf None nexthop 10.79.196.10 id 0x00001 Success
Dec 28 13:18:04.635 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Branch LDP 10.79.196.10:0 binding
list Remote Delete
Dec 28 13:18:04.635 MLDP LSP 0/RP0/CPU0 t10706 DB : 0x00001 Deleting branch entry LDP
10.79.196.10:0
```

On the MP

The interface to the MP goes down. The adjacency is lost over the link, but the LDP adjacency is kept as a started session:

```
Dec 28 13:05:27.134 MLDP GLO 0/RSP1/CPU0 t31491 NBR : 10.79.196.14:0 delete adj
20003a0/10.254.3.36
Dec 28 13:05:27.134 MLDP GLO 0/RSP1/CPU0 t31491 GEN : Checkpoint delete neighbor adj
20003a0/10.254.3.36 objid 0 version 0 Failed
Dec 28 13:05:27.134 MLDP GLO 0/RSP1/CPU0 t31491 NBR : 10.79.196.14:0 LDP Adjacency addr:
10.254.3.36, Interface: Bundle-Ether10362 Delete
Dec 28 13:05:27.134 MLDP GLO 0/RSP1/CPU0 t31491 GEN : Start path timer for root: 10.79.196.14
Dec 28 13:05:27.134 MLDP GLO 0/RSP1/CPU0 t31491 GEN : Checkpoint save neighbor 10.79.196.14:0
canceled, no GR or NSR
Dec 28 13:05:27.152 MLDP GLO 0/RSP1/CPU0 t31488 RIB : Read notification
Dec 28 13:05:27.152 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Root paths count 1
Dec 28 13:05:27.152 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 None 10.79.196.13
Dec 28 13:05:27.152 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.13:0 added (chkpt
FALSE)
Dec 28 13:05:27.152 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.13:0 binding list
Local Add
Dec 28 13:05:27.152 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.13:0 path changed
from None:0.0.0.0 to None:10.79.196.13
Dec 28 13:05:27.152 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Request label type ACEL ident
10.79.196.13:0 LSD Success
Dec 28 13:05:27.153 MLDP GLO 0/RSP1/CPU0 t31491 RIB : Notify client 'Root' for prefix:
10.79.196.14/32
Dec 28 13:05:27.153 MLDP GLO 0/RSP1/CPU0 t31491 GEN : Root 10.79.196.14 path 10.254.1.184 php
nh 10.254.1.184 peer 72d83798:10.79.196.13:0
Dec 28 13:05:27.153 MLDP GLO 0/RSP1/CPU0 t31491 GEN : mldp_root_get_path: tid e0100000 ifh 0
php_nh 0.0.0.0
Dec 28 13:05:27.153 MLDP GLO 0/RSP1/CPU0 t31491 GEN : Failed to get intf type for ifh 0x0
Dec 28 13:05:27.153 MLDP GLO 0/RSP1/CPU0 t31491 RIB : Notify client 'Peer' for prefix:
10.79.196.14/32
Dec 28 13:05:27.153 MLDP GLO 0/RSP1/CPU0 t31491 GEN : Checkpoint save neighbor 10.79.196.14:0
canceled, no GR or NSR
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Main Entry LSD label 24441 type
ACEL ident 10.79.196.13:0 assigned
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.13:0 installed
local label 24441
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Neighbor 10.79.196.13:0 not MBB
```

capable or worse metric, ignore MBB code 0

MBB kicks in: the 600 seconds is the configured switchover delay

```
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Start MBB Notification timer 100 msec (MBB ack)
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL selection delayed for 600 seconds (MBB)
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 P2MP label mapping msg to 10.79.196.13:0 Success
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL selection delayed for 600 seconds (MBB)
```

The new path via the P router is created:

```
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 FWD : 0x00002 Label 24441 create, Flags: 5 Success
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 FWD : 0x00002 Label 24441 add path lspvif Lmdtvrfone rpf-id 3 tid v4 0xe0000013 v6 0xe0800013 Success
Dec 28 13:05:27.153 MLDP LSP 0/RSP1/CPU0 t31491 FWD : 0x00002 Label 24441 id_val 0 id_type 0
Dec 28 13:05:27.154 MLDP GLO 0/RSP1/CPU0 t31491 GEN : ACEL for local label 24441 label up 1048577
Dec 28 13:05:27.233 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Root paths count 1
Dec 28 13:05:27.233 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 None 10.79.196.13
Dec 28 13:05:27.233 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.13:0 found, retain TRUE, to front TRUE
Dec 28 13:05:27.233 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL selection delayed for 600 seconds (MBB)
Dec 28 13:05:27.234 MLDP GLO 0/RSP1/CPU0 t31491 NBR : 10.79.196.14:0 Check branches for path change
Dec 28 13:05:27.234 MLDP GLO 0/RSP1/CPU0 t31491 GEN : Checking paths for root: 10.79.196.14
Dec 28 13:05:27.234 MLDP GLO 0/RSP1/CPU0 t31491 GEN : mldp_root_get_path: tid e0100000 ifh 0 php_nh 0.0.0.0
Dec 28 13:05:27.350 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 MBB notification delay timer expired
Dec 28 13:05:29.275 MLDP GLO 0/RSP1/CPU0 t31491 NBR : 10.79.196.14:0 to address: 10.254.3.36 mapping deleted
```

The 600 seconds timer expires:

```
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Peer change delay timer expired
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL evaluate
```

The entry is deleted after another 60 seconds.

```
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.14:0 start delete pending timer at 60 sec
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.13:0 activate
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 FWD : 0x00002 Label 24441 create, Flags: 1 Success
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 update active ident from 10.79.196.14:0 to 10.79.196.13:0
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Checkpoint save Main Entry active 10.79.196.13:0 rec_nh 0.0.0.0 rec_rd 0:0 cont...
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Checkpoint save lbl no_label length: 88 obj 80002f60 version 136 Success
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.14:0 deactivate
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 FWD : 0x00002 Label 24440 create, Flags: 5 Success
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 update active ident from 10.79.196.13:0 to 0.0.0.0:0
```

```

Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Checkpoint save Main Entry active
0.0.0.0:0 rec_nh 0.0.0.0 rec_rd 0:0 cont...
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Checkpoint save lbl no_label
length: 88 obj 80002f60 version 137 Success
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 update active ident from
0.0.0.0:0 to 10.79.196.13:0
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Checkpoint save Main Entry active
10.79.196.13:0 rec_nh 0.0.0.0 rec_rd 0:0 cont...
Dec 28 13:15:28.352 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Checkpoint save lbl no_label
length: 88 obj 80002f60 version 138 Success
Dec 28 13:15:28.352 MLDP GLO 0/RSP1/CPU0 t31491 GEN : ACEL for local label 24441 label up
1048577
Dec 28 13:15:28.352 MLDP GLO 0/RSP1/CPU0 t31491 GEN : ACEL for local label 24440 label up
1048577

```

The delete delay timer expires. R3 sends the Label Withdraw message and Label Release message to R1:

```

Dec 28 13:15:28.552 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 MBB notification delay timer
expired
Dec 28 13:16:28.552 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.14:0 delete delay
timer expired, delete pending TRUE
Dec 28 13:16:28.552 MLDP LSP 0/RSP1/CPU0 t31491 FWD : 0x00002 Label 24440 delete, Success
Dec 28 13:16:28.552 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.14:0 binding list
Local Delete
Dec 28 13:16:28.552 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 Released label 24440 to LSD
Dec 28 13:16:28.552 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 P2MP label withdraw msg to
10.79.196.14:0 Success
Dec 28 13:16:28.552 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 ACEL 10.79.196.14:0 remove
Dec 28 13:16:28.557 MLDP LSP 0/RSP1/CPU0 t31491 DB : 0x00002 P2MP label release from
10.79.196.14:0 label 24440 root 10.79.196.14 Opaque Len 7

```

Configure FRR timer for scale MLDP LSPs

In a scaled setup with more than 500 LSPs, when an FRR occurs, the unicast Internet Gateway Protocol (IGP) can converge faster than multicast updates (LMRIB to FIB) for mLDP label updates. As a result, FIB can mark off the FRR bit in 2 seconds after an FRR event, where mLDP label hardware programming is not complete on the egress line card, hosting the backup path. The FRR holdtime is by default 2 seconds.

It is advised to increase this FRR holdtime in a scaled setup.

The command *frr-holdtime* configures the FRR holdtime to be proportional to the scale number of LSPs. The recommended *frr-holdtime* value is either the same, or lesser than the MBB delay timer. This ensures that the egress line card is in FRR state after the primary path down event. When not configured, the default *frr-holdtime*, in seconds, is set to 2.

This command was introduced in 5.3.2.

```
RP/0/RSP1/CPU0:ASR-9906#conf t
```

```
RP/0/RSP1/CPU0:ASR-9906(config)#cef platform ?
lsm Label-switched-multicast parameters
```

```
RP/0/RSP1/CPU0:ASR-9906(config)#cef platform lsm ?
frr-holdtime Time to keep FRR slots programmed post FRR
```

```
RP/0/RSP1/CPU0:ASR-9906(config)#cef platform lsm frr-holdtime ?
```

<3-180> Time in seconds

Conclusion

MBB can work to prevent multicast traffic loss for rerouting in the case of routing convergence and in the case of protecting traffic in the case of a link going down, when switching back the multicast traffic from the backup path to a native path.

MBB must be configured to enable it. It must be configured on all routers.

There must be a MBB forwarding delay configured of several seconds to allow the installation of the newly signalled mLDP tree in the forwarding plane before forwarding the traffic from that mLDP tree.