

Multipoint BFD for P2MP MPLS LSPs.

draft-chandra-hegde-mpoint-bfd-for-mpls-00

Chandrasekar R
Vikas Hegde

MP-BFD overview

- Basic Functions
 - To verify head to tail(s) multipoint path
 - Head functions
 - “Multipoint Head” session sends BFD packets over MP tree
 - Ignorant of tail(s)
 - Tail functions
 - “Multipoint Tail” session tracking liveness of head over MP tree
 - Must know head identity
 - Tails’ response to BFD timeout is local decision
 - MP type independent (IP multicast or P2mp LSPs)

Overview (contd.)

- Optional Functions
 - Head (“Multipoint client” session)
 - Tail solicitation
 - Requesting tails to notify session failure
 - Verify specific tail
 - Special handling of specific tails
 - Tail
 - Notifying head of session failure
 - Silent tail

MP-BFD for MPLS

- BFD could be used for fast detection of data path failure along the multipoint path at the egress.
- Features like Multicast Live-Live could benefit from running MP-BFD.
- MP-BFD could also be used to monitor the connectivity of MPLS LSPs which do not see continuous traffic.
- Other redundancy concepts like Warm-Standby could also benefit from running MP-BFD.

Gaps in base MP-BFD for MPLS

- MP-BFD is written to be generic.
 - Assumes that the required information to bootstrap a BFD session can be learnt dynamically.
- May not necessarily be true for P2MP MPLS LSPs. Particularly so when PHP is in use.
- Pretty much the same gap which is solved by RFC 5884 for P2P LSPs and vanilla BFD.

Gaps in base MP-BFD for MPLS

- RFC 5884 may not be optimal because:
 - BFD state machine is different for p2mp
 - Dynamic nature of p2mp LSP environment.
 - Egress routers (leaf nodes) are more state aware than the ingress.

Possible Solutions

- Periodic echo requests with “do not reply”
 - Do not solicit replies from egress LSRs when bootstrapping the BFD session to avoid any form of flooding at the ingress.
 - Periodic send at a sedate interval.
 - Tail does not associate a BFD session with an LSP until it receives the P2MP LSP ping.

Possible Solutions

- Application driven ping
 - Applications usually know which tails are being added to the multipoint tree.
 - Application can trigger an expedited ping to a particular leaf (or all leaves) whenever a new leaf is added.
 - Could be used for faster bootstrapping for applications which are aware of the tails (not always the case).
 - Can be used in conjunction with periodic ping to form an optimal solution.

Feedback

- Any comments are much appreciated!
- Questions?