Internet Engineering Task Force   D. Katz
Internet-Draft   Juniper Networks
Intended status: Standards Track   D. Ward
Expires: November 7, 2015   Cisco Systems
                                                      S. Pallagatti, Ed.
                                                         Juniper Networks
                                                             May 6, 2015

BFD for Multipoint Networks
draft-ietf-bfd-multipoint-06

Abstract

   This document describes extensions to the Bidirectional Forwarding
   Detection (BFD) protocol for its use in multipoint and multicast
   networks. Comments on this draft should be directed to rtg-
   bfd@ietf.org.

Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on November 7, 2015.

Copyright Notice

Table of Contents

1. Introduction

The Bidirectional Forwarding Detection protocol [RFC5880] specifies a
method for verifying unicast connectivity between a pair of systems.
This document defines a method for using BFD to provide verification

of multipoint or multicast connectivity between a multipoint sender
(the "head") and a set of one or more multipoint receivers (the
"tails").

As multipoint transmissions are inherently unidirectional, this
mechanism purports only to verify this unidirectional connectivity.
Although this seems in conflict with the "Bidirectional" in BFD, it
is a natural fit for that protocol.

This application of BFD allows for the tails to detect a lack of
connectivity from the head. Due to unidirectional nature, virtually
all options and timing parameters are controlled by the head.

As an option, the tail may notify the head of the lack of multipoint
connectivity. Details of tail notification to head are outside the
scope of this document.

Throughout this document, the term "multipoint" is defined as a
mechanism by which one or more systems receive packets sent by a
single sender. This specifically includes such things as IP
multicast and point-to-multipoint MPLS.

This document effectively modifies and adds to the base BFD
specification. It is the intention of the authors to fold these
extensions into the base specification at the appropriate time.

2. Goals

The primary goal of this mechanism is to allow tails to rapidly
detect the fact that multipoint connectivity from the head has
failed.

Another goal is for the mechanism to work on any multicast or
multipoint medium.

A further goal is to support multiple, overlapping multipoint paths,
as well as multipoint paths with multiple heads, and to allow point-
to-point BFD sessions to operate simultaneously among the systems
participating in Multipoint BFD.

A final goal is to integrate multipoint operation into the base
specification in such a way as to make it relatively easy to support
both multipoint and point-to-point operation in a single
implementation.

It is a non-goal for this protocol to verify point-to-point
connectivity between the head and any tails. This can be done

   independently (and with no penalty in protocol overhead) by using
   point-to-point BFD.

3. Overview

   The heart of this protocol is the periodic transmission of BFD
   Control packets along a multipoint path, from the head to all tails
   on the tree. The contents of the BFD packets provide the means for
   the tails to calculate the detection time for path failure. If no
   BFD Control packets are received by a tail for a detection time, the
   tail declares the path to have failed. For some applications this is
   the only mechanism necessary; the head can remain ignorant of the
   tails.

   Head may wish to be alerted to the tails' connectivity (or lack
   thereof). Details of how head keeps track of tails and how tails
   alert it's connectivity to head are outside scope of this document.

   Although this document describes a single head and a set of tails
   spanned by a single multipoint path, the protocol is capable of
   supporting (and discriminating between) more than one multipoint path
   at both heads and tails. Furthermore, the same head and tail may
   share multiple multipoint paths, and a multipoint path may have
   multiple heads.

4. Protocol Details

   This section describes the operation of Multipoint BFD in detail.

4.1. Multipoint BFD Control Packets

   Multipoint BFD Control packets (packets sent by the head over a
   multipoint path) are explicitly marked as such, via the setting of
   the M bit. This means that Multipoint BFD does not depend on the
   recipient of a packet to know whether the packet was received over a
   multipoint path. This can be useful in scenarios where this
   information may not be available to the recipient.

4.2. Session Model

   Multipoint BFD is modeled as a set of sessions of different types.
   The elements of procedure differ slightly for each type.

   Point-to-point sessions, as described in [BFD], are of type
   PointToPoint.

   The head has a session of type MultipointHead that is bound to a
   multipoint path. Multipoint BFD Control packets are sent by this

session over the multipoint path, and no BFD Control packets are
received by it.

Each tail has a session of type MultipointTail associated with a
multipoint path. These sessions receive BFD Control packets from the
head over multipoint path.

4.3. Session Failure Semantics

The semantics of session failure are subtle enough to warrant further
explanation.

MultipointHead sessions cannot fail (since they are controlled
administratively.)

If a MultipointTail session fails, it means that the tail definitely
has lost contact with the head (or the head has been administratively
disabled) and the tail should take appropriate action.

4.4. State Variables

Multipoint BFD introduces some new state variables, and modifies the
usage of a few existing ones.

4.4.1. New State Variables

A number of state variables are added to the base specification in
support of Multipoint BFD.

   bfd.SessionType

      The type of this session. Allowable values are:

         PointToPoint: Classic point-to-point BFD.

         MultipointHead: A session on the head responsible for the
         periodic transmission of multipoint BFD Control packets
         along the multipoint path.

         MultipointTail: A multipoint session on a tail.

      This variable MUST be initialized to the appropriate type when
      the session is created, according to the rules in Section 4.13

   bfd.SilentTail

       Always set to 1, a tail will never transmit any BFD Control
       packets to the head under any circumstances. Setting to 0 is
       outside the scope of this document.

       This variable is only pertinent when bfd.SessionType is
       MultipointTail.

4.4.2. State Variable Initialization and Maintenance

   Some state variables defined in section 6.8.1 of the [RFC5880] needs
   to be initialized or manipulated differently depending on the session
   type.

      bfd.RequiredMinRxInterval

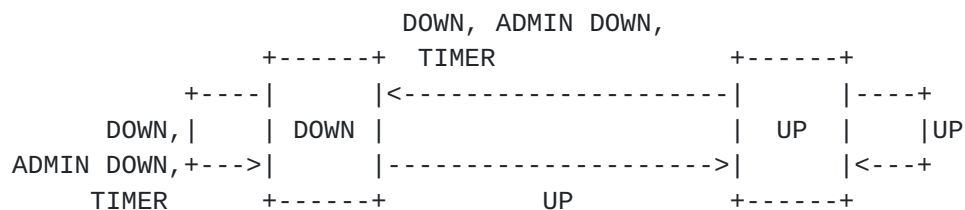   This variable MUST be set to 0 for session type MultipointHead.

      bfd.DemandMode

         This variable MUST be initialized to 1 for session type
         MultipointHead and MUST be initialized to 0 for session type
         MultipointTail.

4.5. State Machine

   The BFD state machine works slightly differently in the multipoint
   application. In particular, since there is a many-to-one mapping,
   three-way handshakes for session establishment and teardown are
   neither possible nor appropriate. As such there is no Init state.
   Session of type MultipointHead MUST NOT send BFD control packets with
   the State field being set to INIT, and must be ignored on receipt.

   The following diagram provides an overview of the state machine for
   session type MultipointTail. The notation on each arc represents the
   state of the remote system (as received in the State field in the BFD
   Control packet) or indicates the expiration of the Detection Timer.

```
                        DOWN, ADMIN DOWN,
                  +------+  TIMER                   +------+
             +----|      |<--------------------|      |----+
       DOWN,|     | DOWN |                     |  UP  |    |UP
  ADMIN DOWN,+--->|      |-------------------->|      |<---+
       TIMER      +------+          UP         +------+
```

   Sessions of type MultipointHead never receive packets and have no
   Detection Timer, and as such all state transitions are
   administratively driven.

4.6. Session Establishment

Unlike Point-to-point BFD, Multipoint BFD provides a form of
discovery mechanism for tails to discover the head. The minimum
amount of a priori information required both on the head and tails is
the binding to the multipoint path over which BFD is running. The
head transmits Multipoint BFD packets on that tree, and the tails
listen for BFD packets on that tree. All other information MAY be
determined dynamically.

A session of type MultipointHead is created for each multipoint path
over which the head wishes to run BFD. This session runs in the
Active role. Except when terminating BFD service, this session is
always in state Up and always operates in Demand mode. No received
packets are ever demultiplexed to the MultipointHead session. In
this sense it is a degenerate form of a session.

Sessions on the tail MAY be established dynamically, based on the
receipt of a Multipoint BFD Control packet from the head, and are of
type MultipointTail. Tail sessions always take the Passive role.

4.7. Discriminators and Packet Demultiplexing

The use of Discriminators is somewhat different in Multipoint BFD
than in Point-to-point BFD.

The head sends Multipoint BFD Control packets over the MultipointHead
session with My Discr set to a value bound to the multipoint path,
and with Your Discr set to zero.

IP and MPLS multipoint tails MUST demultiplex BFD packets based on a
combination of the source address, My Discriminator and the identity
of the multipoint tree which the Multipoint BFD Control packet was
received from. Together they uniquely identify the head of the
multipoint path. Bootstrapping BFD session to a multipoint LSP in
case of penultimate hop popping is outside the scope of this
document.

Note that, unlike PointToPoint sessions, the discriminator values on
all multipoint session types MUST NOT be changed during the life of a
session. This is a side effect of the more complex demultiplexing
scheme.

4.8. Packet consumption on tails

BFD packets received on tails for a multicast group MUST be consumed
by tails and MUST not be forwarded to receivers. Session of type
MultipointTail MUST identify the packet as BFD with the help of

destination UDP port number "3784" on IP multipoint path. For
multipoint LSP, MultipointTail MUST use destination UDP port "3784"
and IP "127.0.0.0/8" range. Packet identified as BFD packet MUST be
consumed by MultipointTail and demultiplex as described in
Section 4.13.2

4.9. Bringing Up and Shutting Down Multipoint BFD Service

Because there is no three-way handshake in Multipoint BFD, a newly
started head (that does not have any previous state information
available) SHOULD start with bfd.SessionState set to Down and with
bfd.RequiredMinRxInterval set to zero in the MultipointHead session.
The session SHOULD remain in this state for a time equal to
(bfd.DesiredMinTxInterval * bfd.DetectMult). This will ensure that
all MultipointTail sessions are reset (so long as the restarted head
is using the same or larger value of bfd.DesiredMinTxInterval than it
did previously.)

Multipoint BFD service is brought up by administratively setting
bfd.SessionState to Up in the MultipointHead session.

A head may wish to shut down its BFD service in a controlled fashion.
This is desirable because the tails need not wait a detection time
prior to declaring the multipoint session to be down (and taking
whatever action is necessary in that case.)

To shut down a multipoint session a head MUST administratively set
bfd.SessionState in the MultipointHead session to either Down or
AdminDown and SHOULD set bfd.RequiredMinRxInterval to zero. The
session SHOULD send BFD Control packets in this state for a period
equal to (bfd.DesiredMinTxInterval * bfd.DetectMult).

The semantic difference between Down and AdminDown state is for
further discussion.

4.10. Timer Manipulation

Because of the one-to-many mapping, a session of type MultipointHead
SHOULD NOT initiate a Poll Sequence in conjunction with timer value
changes. However to indicate change in packet MultipointHead session
MUST send packet with P bit set. MultipointTail session MUST NOT
reply if packet has M, P bit set and bfd.RequiredMinRxInterval set to
0.

The MultipointHead MUST send bfd.DetectMult packets with P bit set at
the old transmit interval before using the higher value in order to
avoid false detection timeouts at the tails. MultipointHead May also

wait some amount of time before making the changes to the transmit
interval (through configuration).

Change in the value of bfd.RequiredMinRxInterval is outside the scope
of this document.

4.11. Detection Times

Multipoint BFD is inherently asymmetric. As such, each session type
has a different approach to detection times.

Since the MultipointHead session never receives packets, it does not
calculate a detection time.

MultipointTail sessions cannot influence the transmission rate of the
MultipointHead session using the Required Min Rx Interval field
because of its one-to-many nature. As such, the Detection Time
calculation for a MultipointTail session does not use
bfd.RequiredMinRxInterval in the calculation. The detection time is
calculated as the product of the last received values of Desired Min
TX Interval and Detect Mult.

The value of bfd.DetectMult may be changed at any time on any session
type.

4.12. State Maintenance for Down/AdminDown Sessions

The length of time session state is kept after the session goes down
determines how long the session will continue to send BFD Control
packets (since no packets can be sent after the session is
destroyed.)

4.12.1. MultipointHead Sessions

When a MultipointHead session transitions to states Down or
AdminDown, the state SHOULD be maintained for a period equal to
(bfd.DesiredMinTxInterval * bfd.DetectMult) to ensure that the tails
more quickly detect the session going down (by continuing to transmit
BFD Control packets with the new state.)

4.12.2. MultipointTail Sessions

MultipointTail sessions MAY be destroyed immediately upon leaving Up
state, since tail will transmit no packets.

Otherwise, MultipointTail sessions SHOULD be maintained as long as
BFD Control packets are being received by it (which by definition
will indicate that the head is not Up.)

4.13. **Base Specification Text Replacement**

   The following sections are meant to replace the corresponding
   sections in the base specification.

4.13.1. **Reception of BFD Control Packets**

   The following procedure replaces section 6.8.6 of [RFC5880].

   When a BFD Control packet is received, the following procedure MUST
   be followed, in the order specified. If the packet is discarded
   according to these rules, processing of the packet MUST cease at that
   point.

      If the version number is not correct (1), the packet MUST be
      discarded.

      If the Length field is less than the minimum correct value (24 if
      the A bit is clear, or 26 if the A bit is set), the packet MUST be
      discarded.

      If the Length field is greater than the payload of the
      encapsulating protocol, the packet MUST be discarded.

      If the Detect Mult field is zero, the packet MUST be discarded.

      If the My Discriminator field is zero, the packet MUST be
      discarded.

      Demultiplex the packet to a session according to Section 4.13.2
      below. The result is either a session of the proper type, or the
      packet is discarded (and packet processing MUST cease.)

      If the A bit is set and no authentication is in use (bfd.AuthType
      is zero), the packet MUST be discarded.

      If the A bit is clear and authentication is in use (bfd.AuthType
      is nonzero), the packet MUST be discarded.

      If the A bit is set, the packet MUST be authenticated under the
      rules of section 6.7, based on the authentication type in use
      (bfd.AuthType.) This may cause the packet to be discarded.

      Set bfd.RemoteDiscr to the value of My Discriminator.

      Set bfd.RemoteState to the value of the State (Sta) field.

      Set bfd.RemoteDemandMode to the value of the Demand (D) bit.

Set bfd.RemoteMinRxInterval to the value of Required Min RX
Interval.

If the Required Min Echo RX Interval field is zero, the
transmission of Echo packets, if any, MUST cease.

If a Poll Sequence is being transmitted by the local system and
the Final (F) bit in the received packet is set, the Poll Sequence
MUST be terminated.

If bfd.SessionType is PointToPoint, update the transmit interval
as described in [RFC5880] section 6.8.2.

If bfd.SessionType is PointToPoint, update the Detection Time as
described in [RFC5880] section 6.8.4. Otherwise, update the
Detection Time as described in Section 4.11 above.

If bfd.SessionState is AdminDown

   Discard the packet

If received state is AdminDown

   If bfd.SessionState is not Down

      Set bfd.LocalDiag to 3 (Neighbor signaled session down)

      Set bfd.SessionState to Down

Else

   If bfd.SessionState is Down

      If bfd.SessionType is PointToPoint

         If received State is Down

            Set bfd.SessionState to Init

         Else if received State is Init

            Set bfd.SessionState to Up

      Else (bfd.SessionType is not PointToPoint)

         If received State is Up

            Set bfd.SessionState to Up

      Else if bfd.SessionState is Init

        If received State is Init or Up

          Set bfd.SessionState to Up

      Else (bfd.SessionState is Up)

        If received State is Down

          Set bfd.LocalDiag to 3 (Neighbor signaled session down)

          Set bfd.SessionState to Down

Check to see if Demand mode should become active or not (see
[RFC5880] section 6.6).

If bfd.RemoteDemandMode is 1, bfd.SessionState is Up, and
bfd.RemoteSessionState is Up, Demand mode is active on the remote
system and the local system MUST cease the periodic transmission
of BFD Control packets (see Section 4.13.3.)

If bfd.RemoteDemandMode is 0, or bfd.SessionState is not Up, or
bfd.RemoteSessionState is not Up, Demand mode is not active on the
remote system and the local system MUST send periodic BFD Control
packets (see see Section 4.13.3.)

If the packet was not discarded, it has been received for purposes
of the Detection Time expiration rules in [RFC5880] section 6.8.4.

4.13.2. Demultiplexing BFD Control Packets

This section is part of the replacement for [RFC5880] section 6.8.6,
separated for clarity.

If the Multipoint (M) bit is set

    If the Your Discriminator field is nonzero, the packet MUST be
    discarded.

    Select a session as based on source address, My Discriminator
    and the identity of the multipoint tree which the Multipoint
    BFD Control packet was received. If a session is found, and
    bfd.SessionType is not MultipointTail, the packet MUST be
    discarded. If a session is not found, a new session of type
    MultipointTail MAY be created, or the packet MAY be discarded.
    This choice is outside the scope of this specification.

Else (Multipoint bit is clear)

> If the Your Discriminator field is nonzero
>
> > Select a session based on the value of Your Discriminator.
> > If no session is found, the packet MUST be discarded.
>
> Else (Your Discriminator is zero)
>
> > If the State field is not Down or AdminDown, the packet MUST
> > be discarded.
> >
> > Otherwise, the session MUST be selected based on some
> > combination of other fields, possibly including source
> > addressing information, the My Discriminator field, and the
> > interface over which the packet was received. The exact
> > method of selection is application-specific and is thus
> > outside the scope of this specification.
> >
> > If a matching session is found, and bfd.SessionType is not
> > PointToPoint, the packet MUST be discarded.
> >
> > If a matching session is not found, a new session of type
> > PointToPoint may be created, or the packet may be discarded.
> > This choice is outside the scope of this specification.
>
> If the State field is Init and bfd.SessionType is not
> PointToPoint, the packet MUST be discarded.

4.13.3. Transmitting BFD Control Packets

The following procedure replaces section 6.8.7 of [RFC5880].

BFD Control packets MUST be transmitted periodically at the rate
determined according to [RFC5880] section 6.8.2, except as specified
in this section.

A system MUST NOT transmit any BFD Control packets if bfd.RemoteDiscr
is zero and the system is taking the Passive role.

A system MUST NOT transmit any BFD Control packets if bfd.SilentTail
is 1.

A system MUST NOT periodically transmit BFD Control packets if Demand
mode is active on the remote system (bfd.RemoteDemandMode is 1,
bfd.SessionState is Up, and bfd.RemoteSessionState is Up) and a Poll
Sequence is not being transmitted.

A system MUST NOT periodically transmit BFD Control packets if
bfd.RemoteMinRxInterval is zero.

If bfd.SessionType is MultipointHead, the transmit interval MUST be
set to bfd.DesiredMinTxInterval (this should happen automatically, as
bfd.RemoteMinRxInterval will be zero.)

If bfd.SessionType is not MultipointHead, the transmit interval MUST
be recalculated whenever bfd.DesiredMinTxInterval changes, or
whenever bfd.RemoteMinRxInterval changes, and is equal to the greater
of those two values. See [BFD] sections 6.8.2 and 6.8.3 for details
on transmit timers.

A system MUST NOT set the Demand (D) bit if bfd.SessionType is
MultipointTail.

A system MUST NOT set the Demand (D) bit if bfd.SessionType
PointToPoint unless bfd.DemandMode is 1, bfd.SessionState is Up, and
bfd.RemoteSessionState is Up.

If bfd.SessionType is PointToPoint or MultipointHead, a BFD Control
packet SHOULD be transmitted during the interval between periodic
Control packet transmissions when the contents of that packet would
differ from that in the previously transmitted packet (other than the
Poll and Final bits) in order to more rapidly communicate a change in
state.

The contents of transmitted BFD Control packets MUST be set as
follows:

   Version

      Set to the current version number (1).

   Diagnostic (Diag)

      Set to bfd.LocalDiag.

   State (Sta)

      Set to the value indicated by bfd.SessionState.

   Poll (P)

      Set to 1 if the local system is sending a Poll Sequence or is a
      session of type MultipointHead soliciting the identities of the
      tails, or 0 if not.

Final (F)

   Set to 1 if the local system is responding to a Control packet
   received with the Poll (P) bit set, or 0 if not.

Control Plane Independent (C)

   Set to 1 if the local system's BFD implementation is
   independent of the control plane (it can continue to function
   through a disruption of the control plane.)

Authentication Present (A)

   Set to 1 if authentication is in use on this session
   (bfd.AuthType is nonzero), or 0 if not.

Demand (D)

   Set to bfd.DemandMode if bfd.SessionState is Up and
   bfd.RemoteSessionState is Up. Set to 1 if bfd.SessionType is
   MultipointHead. Otherwise it is set to 0.

Multipoint (M)

   Set to 1 if bfd.SessionType is MultipointHead. Otherwise it is
   set to 0.

Detect Mult

   Set to bfd.DetectMult.

Length

   Set to the appropriate length, based on the fixed header length
   (24) plus any Authentication Section.

My Discriminator

   Set to bfd.LocalDiscr.

Your Discriminator

   Set to bfd.RemoteDiscr.

Desired Min TX Interval

   Set to bfd.DesiredMinTxInterval.

   Required Min RX Interval

      Set to bfd.RequiredMinRxInterval.

   Required Min Echo RX Interval

      Set to 0 if bfd.SessionType is MultipointHead or
      MultipointTail.

   Authentication Section

      Included and set according to the rules in section 6.7 if
      authentication is in use (bfd.AuthType is nonzero.) Otherwise
      this section is not present.

5. Assumptions

   If authentication is in use, all tails must be configured to have a
   common authentication key in order to receive the multipoint BFD
   Control packets.

6. IANA Considerations

   This document has no actions for IANA.

7. Security Considerations

   Implementations that creates MultpointTail sessions dynamically upon
   receipt of Multipoint BFD Control packets MUST implement protective
   measures to prevent infinite number of MultipointTail session being
   created. Below lists some points to be considered in such
   implementations.

      If a Multipoint BFD Control packet did not arrive on a multicast
      tree (ex: on expected interface, with expected MPLS label, etc),
      then a MultipointTail session should not be created.

      If redundant streams are expected for a given multicast stream,
      then the implementations should not create more MultipointTail
      sessions than the number of streams. Additionally, when the
      number of MultipointTail sessions exceeds the number of expected
      streams, then the implementation should generate an alarm to users
      to indicate the anomaly.

      The implementation should have a reasonable upper bound on the
      number of MultipointTail sessions that can be created, with the
      upper bound potentially being computed based on the number of
      multicast streams that the system is expecting.

8. Contributors

   Rahul Aggarwal of Juniper Networks and George Swallow of Cisco
   Systems provided the initial idea for this specification and
   contributed to its development.

9. Normative References

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

   [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection
              (BFD)", RFC 5880, June 2010.

Authors' Addresses

   Dave Katz
   Juniper Networks
   1194 N. Mathilda Ave.
   Sunnyvale, California 94089-1206
   USA

   Email: dkatz@juniper.net

   Dave Ward
   Cisco Systems
   170 West Tasman Dr.
   San Jose, California 95134
   USA

   Email: wardd@cisco.com

   Santosh Pallagatti (editor)
   Juniper Networks
   Embassy Business Park
   Bangalore, KA 560093
   India

   Email: santoshpk@juniper.net