

Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: April 21, 2014

N. Akiya  
C. Pignataro  
D. Ward  
Cisco Systems  
M. Bhatia  
Alcatel-Lucent  
P. K. Santosh  
Juniper Networks  
October 18, 2013

Seamless Bidirectional Forwarding Detection (BFD) with MPLS Label  
Verification Extension  
draft-akiya-bfd-seamless-base-02

## Abstract

This document defines a simplified mechanism to use Bidirectional Forwarding Detection (BFD) with large portions of negotiation aspects eliminated, that allows full and partial reachability verification. For MPLS based BFD, extensions to the generic mechanism are defined that allows BFD to perform a level of label verification.

## 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 April 21, 2014.

## Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1.	Introduction . . . . .	3
2.	Seamless BFD Overview . . . . .	4
3.	Terminology . . . . .	5
4.	BFD Target Identifier Types . . . . .	5
5.	Reserved BFD Discriminators . . . . .	6
6.	BFD Target Identifier Table . . . . .	6
7.	Reflector BFD Session . . . . .	6
8.	Full Reachability Validations . . . . .	7
8.1.	Initiator Behavior . . . . .	7
8.1.1.	Initiator State machine . . . . .	8
8.2.	Responder Behavior . . . . .	9
8.2.1.	Responder Demultiplexing . . . . .	9
8.2.2.	Reflector BFD Session Procedures . . . . .	9
8.3.	Further Packet Details . . . . .	11
8.4.	Diagnostic Values . . . . .	12
8.5.	The Poll Sequence . . . . .	12
8.6.	Control Plane Independent (C) . . . . .	12
8.7.	Additional Initiator Behavior . . . . .	12
8.8.	Additional Responder Behavior . . . . .	12
9.	Partial Reachability Validations . . . . .	13
10.	MPLS Label Verifications . . . . .	13
10.1.	MPLS Label Verifications Mechanism . . . . .	13
10.2.	Localhost Address Usage . . . . .	15
11.	Scaling Aspect . . . . .	15
12.	Co-existence with Traditional BFD . . . . .	15
13.	BFD Echo . . . . .	15
14.	Summary . . . . .	15
15.	Security Considerations . . . . .	17
16.	IANA Considerations . . . . .	17
17.	Acknowledgements . . . . .	18
18.	Contributing Authors . . . . .	18
19.	References . . . . .	18
19.1.	Normative References . . . . .	18

19.2. Informative References . . . . .	19
Authors' Addresses . . . . .	19

## 1. Introduction

Bidirectional Forwarding Detection (BFD), [RFC5880] and related documents, has efficiently generalized the failure detection mechanism for multiple protocols and applications. There are some improvements which can be made to better fit existing technologies. There is a possibility of evolving BFD to better fit new technologies. This document focuses on several aspects of BFD in order to further improve efficiency, to expand failure detection coverage and to allow BFD usage for wider scenarios.

- o There are scenarios when only one side of the BFD, not both, are interested in verifying connectivity between a pair of systems. One example is when a static route uses BFD to validate reachability to the nexthop IP router. Another example is when a uni-directional tunnel uses BFD to validate reachability to the egress node. In such scenarios, regular BFD requires sessions to be provisioned on target nodes (ex: static route nexthop node, egress of RSVP-TE unidirectional LSP) which adds minimal value, if any, to those egress nodes.
- o BFD provides data delivery confidence when reachability validation is performed prior to traffic utilizing specific paths/LSPs. However this comes with a cost where traffic is prevented to use such paths/LSPs until BFD is able to validate the reachability, which could take seconds due to BFD session bring-up sequences [RFC5880], LSP ping bootstrapping [RFC5884], etc. S-BFD addresses these problems by eliminating the three-way handshake mechanism during bootstrap of BFD sessions resulting in faster reachability validation of BFD provisioned paths/LSPs. In addition, it is expected that some MPLS technologies will require traffic engineered LSPs to get created dynamically, driven by external applications (ex: SDN). It would be desirable to perform BFD validation very quickly to allow applications to utilize dynamically created LSPs in timely manner.
- o Existing BFD standards provide a good mechanism to verify end-to-end reachability. They however, do not allow BFD to perform partial reachability validations: ingress to transit, transit to transit and transit to egress.
- o [RFC5884] defines a mechanism to run BFD on existing MPLS technologies. It is used to perform end-to-end LSP liveness check for detecting MPLS data plane failures. This mechanism, however, lacks the ability to validate traversal of the intended

LSP path. Specifically it cannot detect failures where one of the nodes along the LSP incorrectly label switches the BFD packet, as long as it reaches the intended LSP egress node. The likelihood of this issue being seen depends on deployed MPLS technologies. With MPLS technologies that use downstream label allocation scheme (ex: RSVP, LDP), the incoming label itself provides a level of check as a node will drop any packet containing non-self-advertised label as the top label or will get delivered to unintended egress node. The issue is less likely to be seen for such MPLS technologies. With MPLS technologies such as Segment Routing (SR), incoming label can often be a label allocated and advertised by a node that is multiple downstream hops away. For such MPLS technologies, issue will be more likely to be seen. [RFC4379] can detect such broken LSPs, but it is often difficult to run this technology at the rate which BFD is capable of.

- o A node may desire to establish multiple BFD sessions to a network target. One such scenario is when different applications on a system require running BFD to the same remote target with different failure detection time requirements. Another scenario is when there are multiple unnumbered logical interfaces between a pair of network nodes. A third scenario can be envisaged where a node hosts multiple BFD sessions to the same remote target on different parts of the system (e.g. different CPUs) in order to provide local redundancy when using BFD to validate paths/LSPs. Such a setup may be used to provide resiliency against local faults that can otherwise impact BFD sessions used to monitor paths/LSPs.

This specification provides solutions to above issues by defining a generic mechanism to use Bidirectional Forwarding Detection (BFD) with large portions of negotiation aspects eliminated, that allows full and partial reachability validation. For MPLS based BFD, extensions to the generic mechanism are defined for BFD to perform a level of label verifications. Because the mechanism eliminates much of negotiation aspects of the BFD protocol, "Seamless BFD" has been chosen as the name for this mechanism.

## 2. Seamless BFD Overview

To operate Seamless BFD, set of network entities are first selected. Each network node hosting selected network entities then assigns a special BFD discriminator to each selected local network entity. These network nodes will also create a BFD session instance that listens for incoming BFD control packets with "your discriminator" having local special BFD discriminators. Mappings between selected network entities and corresponding special BFD discriminators are known to other network nodes belonging in the same network. The

mechanism of disseminating the special BFD discriminators is beyond the scope of this specification. A network node in such network is then able to send a BFD control packet to a particular target with corresponding special BFD discriminator as "your discriminator". Target network node, upon reception of such BFD control packet, will transmit a response BFD control packet back to the sender.

Example: IPv4 address 1.2.3.4 is selected as the Seamless BFD target. Node hosting IPv4 address 1.2.3.4 reserves the BFD discriminator 0x01020304, and creates a BFD session instance in listening mode. Node X sends a BFD control packet with destination IP address 1.2.3.4, source IP address X, "your discriminator"=0x01020304 and "my discriminator"=<locally assigned discriminator>. Node hosting IPv4 address 1.2.3.4 will receive this packet, swaps received "your discriminator"/"my discriminator" and generates a response BFD control packet destined to X.

### 3. Terminology

The reader is expected to be familiar with the BFD, IP, MPLS and SR terminology and protocol constructs. This section describes several new terminology introduced by Seamless BFD.

- o BFD Target Identifier: Network entity that is provisioned as a target of Seamless BFD.
- o BFD Target Identifier Type: Type of network entity that is provisioned as a target of Seamless BFD.
- o BFD Target Identifier Table: A table containing BFD target identifier type, BFD target identifier and corresponding BFD discriminator.
- o Reflector BFD Session: A BFD session listening for incoming BFD control packets destined for local BFD target identifier(s).

### 4. BFD Target Identifier Types

Number of network entity types (ex: IP address, segment ID) can make use of this mechanism. To differentiate between different network entity types, a value is assigned to each type.

BFD Target Identifier types:

Value	BFD Target Identifier Type
-----	-----
0	Reserved
1	IP (IPv4 Address and Router ID)

## 2 Segment Routing Node Segment ID

Note that IP based BFD from [RFC5885] is supported by this specification, but non-IP based BFD is outside the scope of this document.

Further identifier types to be defined as on need basis.

### 5. Reserved BFD Discriminators

All local network identifiers which are to participate in this mechanism are to have specific BFD discriminators assigned. Assigned BFD discriminators are attached to corresponding identifiers until they are explicitly un-provisioned. BFD discriminators used for this mechanism are considered reserved, and MUST NOT be reused for other BFD sessions.

Some examples of network identifier to BFD discriminator mappings:

- o BFD Target Identifier Type 1: IPv4 address 1.1.1.1 maps to BFD discriminator 0x01010101.
- o BFD Target Identifier Type 2: Node segment ID 0x03E800FF maps to BFD discriminator 0x03E800FF.

### 6. BFD Target Identifier Table

Each network node is responsible for creating and maintaining a table that contains BFD discriminators, BFD target identifier types and BFD target identifiers. Intention of this table is to allow local entities to perform following lookups:

- o BFD discriminator to BFD target identifier type and BFD target identifier
- o BFD target identifier type and BFD target identifier to BFD discriminator

This table is to contain entries for all locally reserved BFD discriminators and corresponding information. This table may need to contain entries from other network nodes, depending on the BFD target identifier type.

### 7. Reflector BFD Session

Each network node MUST create one or more reflector BFD sessions. This reflector BFD session is a session which transmits BFD control

packets in response to received valid locally destined BFD control packets. Specifically, this reflector BFD session is to have following characteristics:

- o MUST NOT transmit any BFD control packets based on local timer expiry.
- o MUST transmit BFD control packet in response to a received valid locally destined BFD control packet.
- o MUST be capable of sending only two states: UP and ADMINDOWN.

One reflector BFD session MAY be responsible for handling received BFD control packets targeted to all local BFD target identifiers, or few reflector BFD sessions MAY each be responsible for subset of local BFD target identifiers. This policy is a local matter, and is outside the scope of this document.

Note that incoming BFD control packets destined to BFD target identifier types may be IPv4, IPv6 or MPLS based. For those BFD target identifier types, implementations MAY either allow the same reflector BFD session to handle all incoming BFD control packets in address family agnostic fashion, or setup multiple reflector BFD sessions to handle incoming BFD control packets with different address families. This policy is again a local matter, and is outside the scope of this document.

## 8. Full Reachability Validations

### 8.1. Initiator Behavior

Any network node can attempt to perform a full reachability validation to any BFD target identifier on other network nodes, as long as destination BFD target identifier is provisioned to use this mechanism. BFD control packets transmitted by the initiator is to have "your discriminator" corresponding to destination BFD target identifier.

A node that initiates a BFD control packet MAY create an active BFD session to periodically send BFD control packets to a target, or a BFD control packet MAY be crafted and sent out on "as needed basis" (ex: BFD ping) without any session presence. In both cases, a BFD instance MUST have unique "my discriminator" value assigned. If a node is to create multiple BFD instances to a same BFD target identifier, then each instance MUST have separate "my discriminator" values assigned. A BFD instance MUST NOT use a discriminator corresponding to one of local BFD target identifiers as "my discriminator". This is to prevent incoming response BFD control

packets ("pong" packets) having "your discriminator" as a discriminator corresponding to the local BFD target identifier.

Below ASCII art describes high level concept of full reachability validations using this mechanism. R2 reserves value XX as BFD discriminator for its BFD target identifier. ASCII art shows that R1 and R4 performing full reachability validation to XX on R2.

```
-- md=50/yd=XX (BFD ping) -->
<-- md=XX/yd=50 (BFD pong) --

                                [*]
R1 ----- R2 ----- R3 ----- R4
                                |   ^
                                |   |
                                |   + - md=60/yd=XX (BFD ping) --
                                + - - -md=XX/yd=60 (BFD pong) -->
```

[\*] Reflector BFD session on R2.

If BFD control packet is to be sent via IP path, then:

- o Destination IP address MUST be an IP address corresponding to target identifier.
- o Source IP address MUST be a local IP address.
- o IP TTL MUST be 255 for full reachability validations. Partial reachability validations MAY use smaller TTL value (see Section 9).
- o Well-known UDP destination port(s) for IP based S-BFD.

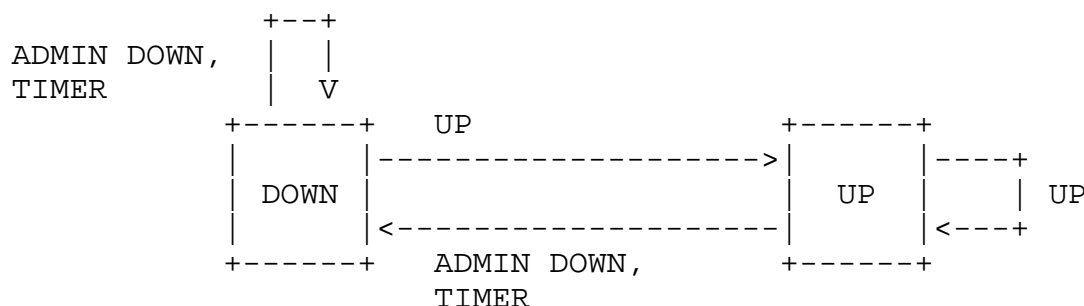
If BFD control packet is to be sent via explicit label switching, then:

- o BFD control packet MUST get imposed with a label stack that is expected to reach the target node.
- o MPLS TTL MUST be 255 for full reachability validations. Partial reachability validations MAY use smaller TTL value (see Section 9).
- o Destination IP address MUST be 127/8 for IPv4 and 0:0:0:0:FFFF:7F00/104 for IPv6.
- o Source IP address MUST be a local IP address.
- o IP TTL=1.
- o Well-known UDP destination port(s) for MPLS based S-BFD

#### 8.1.1. Initiator State machine



The following diagram provides an overview of the initiator state machine. 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.



Note that the above state machine is different from the base BFD specification [RFC5880]. This is because the Init state is no longer applicable for the initiator of the S-BFD session. Another important difference is the transition of the state machine from the Down state to the Up state when a packet with State Up is received by the initiator. The definitions of the states and the events have the same meaning as in the base BFD specification [RFC5880].

## 8.2. Responder Behavior

A network node which receives BFD control packets transmitted by an initiator is referred as responder. Responder, upon reception of BFD control packets, is to perform necessary relevant validations described in [RFC5880]/[RFC5881]/[RFC5883]/[RFC5884]/[RFC5885].

### 8.2.1. Responder Demultiplexing

When responder receives a BFD control packet, if "your discriminator" value is not one of local entries in the BFD target identifier table, then this packet MUST NOT be considered for this mechanism. If "your discriminator" value is one of local entries in the BFD target identifier table, then the packet is determined to be handled by a reflector BFD session responsible for specified BFD targeted identifier. If the packet was determined to be processed further for this mechanism, then chosen reflector BFD session is to transmit a response BFD control packet using procedures described in Section 8.2.2, unless prohibited by local administrative or local policy reasons.

### 8.2.2. Reflector BFD Session Procedures

BFD target identifier type MUST be used to determine further information on how to reach back to the initiator.

In addition, destination IP address of received BFD control packet MUST be examined to determine how to construct response BFD control packet to send back to the initiator.

If destination IP address of received BFD control packet is not 127/8 for IPv4 or 0:0:0:0:0:FFFF:7F00/104 for IPv6, then:

- o Destination IP address MUST be copied from received source IP address.
- o Source IP address MUST be copied from received destination IP address if received destination IP address is a local address. Otherwise local IP address MUST be used.
- o IP TTL MUST be 255.

If destination IP address of received BFD control packet is 127/8 for IPv4 or 0:0:0:0:0:FFFF:7F00/104 for IPv6, then received IP destination MUST be further examined to determine response transport options. If last 23 bits of 127/8 for IPv4 and 0:0:0:0:0:FFFF:7F00/104 for IPv6 is zero, then response SHOULD be label switched but MAY be IP routed. If last 23 bits of 127/8 for IPv4 and 0:0:0:0:0:FFFF:7F00/104 for IPv6 is not zero, then response SHOULD be label switched and SHOULD NOT be IP routed. Description of 23 bits is described in Section 10.

If BFD control packet response is determined to be IP routed, then:

- o Destination IP address MUST be copied from received source IP address.
- o Source IP address MUST be a local address.
- o IP TTL MUST be 255.

If BFD control packet response is determined to be label switched, then:

- o BFD control packet MUST get label switched back to the initiator. Determining the label stack to be imposed on a response BFD control packet is outside the scope of this document.
- o MPLS TTL MUST be 255.
- o Destination IP address MUST be 127/8 for IPv4 and 0:0:0:0:0:FFFF:7F00/104 for IPv6.
- o Source IP address MUST be a local IP address.
- o IP TTL MUST be 1.

Regardless of the response type, BFD control packet being sent by the responder MUST perform following procedures:

- o Copy "my discriminator" from received "your discriminator", and "your discriminator" from received "my discriminator".
- o UDP destination port MUST be same as received UDP destination port.

In addition, reflector BFD session SHOULD transmit response BFD control packet on the same interface on which it received the packet from initiator.

### 8.3. Further Packet Details

Further details of BFD control packets sent by initiator (ex: active BFD session):

- o Well-known UDP destination port assigned for S-BFD.
- o UDP source port as per described in [RFC5881]/[RFC5883]/[RFC5884]/[RFC5885].
- o "my discriminator" assigned by local node.
- o "your discriminator" corresponding to an identifier of target node.
- o "State" MUST be set to a value reflecting local state.
- o "Desired Min TX Interval" MUST be set to a value reflecting local desired minimum transmit interval.
- o "Required Min RX Interval" MUST be zero.
- o "Required Min Echo RX Interval" SHOULD be zero.
- o "Detection Multiplier" MUST be set to a value reflecting locally used multiplier value.

Further details of BFD control packets sent by responder (reflector BFD session):

- o Well-known UDP destination port assigned for S-BFD.
- o UDP source port as described in [RFC5881]/[RFC5883]/[RFC5884]/[RFC5885].
- o "my discriminator" MUST be copied from received "your discriminator".
- o "your discriminator" MUST be copied from received "my discriminator".
- o "State" MUST be UP or ADMINDOWN. Clarification of reflector BFD session state is described in Section 8.8.
- o "Desired Min TX Interval" MUST be copied from received "Desired Min TX Interval".
- o "Required Min RX Interval" MUST be set to a value reflecting how many incoming control packets this reflector BFD session can handle.
- o "Required Min Echo RX Interval" SHOULD be set to zero.
- o "Detection Multiplier" MUST be copied from received "Detection Multiplier".

#### 8.4. Diagnostic Values

Diagnostic value in both directions MAY be set to a certain value, to attempt to communicate further information to both ends. However, details of such are outside the scope of this specification.

#### 8.5. The Poll Sequence

The Poll sequence MUST operate in accordance with [RFC5880].

#### 8.6. Control Plane Independent (C)

Control plane independent (C) bit for BFD instances speaking to a reflector BFD session MUST work according to [RFC5880]. Reflector BFD session also MUST work according to [RFC5880]. Specifically, if reflector BFD session implementation does not share fate with control plane, then response BFD control packets transmitted MUST have control plane independent (C) bit set. If reflector BFD session implementation shares fate with control plane, then response BFD control packets transmitted MUST NOT have control plane independent (C) bit set.

#### 8.7. Additional Initiator Behavior

- o If initiator receives valid BFD control packet in response to transmitted BFD control packet, then initiator SHOULD conclude that packet reached intended target.
- o When a sufficient number of BFD control packets have not arrived as they should, the initiator could declare loss of reachability. The criteria for declaring loss of reachability and the action that would be triggered as a result are outside the scope of this specification.
- o Relating to above bullet item, it is critical for an implementation to understand the latency to/from reflector BFD session on target node. In other words, for very first BFD control packet transmitted, an implementation MUST NOT expect response BFD control packet to be received for time equivalent to sum of latencies: initiator node to target node and target node back to initiator node.

#### 8.8. Additional Responder Behavior

- o BFD control packets transmitted by a reflector BFD session MUST have "Required Min RX Interval" set to a value which reflects how many incoming control packets this reflector BFD session can handle. Responder can control how fast initiators will be sending

BFD control packets to self by ensuring "Required Min RX Interval" reflects a value based on current load.

- o If a reflector BFD session wishes to communicate to some or all initiators that monitored BFD target identifier is "temporarily out of service", then BFD control packets with "state" set to ADMINDOWN are sent to those initiators. Initiators, upon reception of such packets, MUST NOT conclude loss of reachability to corresponding BFD target identifier, and MUST back off packet transmission interval to corresponding BFD target identifier an interval no faster than 1 second. If a reflector BFD session is generating a response BFD control packet for BFD target identifier that is in service, then "state" in response BFD control packets MUST be set to UP.

## 9. Partial Reachability Validations

Same mechanism as described in "Full Reachability Validations" section will be applied with exception of following differences on initiator.

- o When initiator wishes to perform a partial reachability validation towards identifier X upto identifier Y, number of hops to identifier Y is calculated.
- o TTL value based on this calculation is used as the IP TTL or MPLS TTL on top most label, and "your discriminator" of transmitted BFD control packet will carry BFD discriminator corresponding to target transit identifier Y.
- o Imposed label stack or IP destination address will continue to be of identifier X.

## 10. MPLS Label Verifications

This section is only applicable to MPLS based sessions using this mechanism.

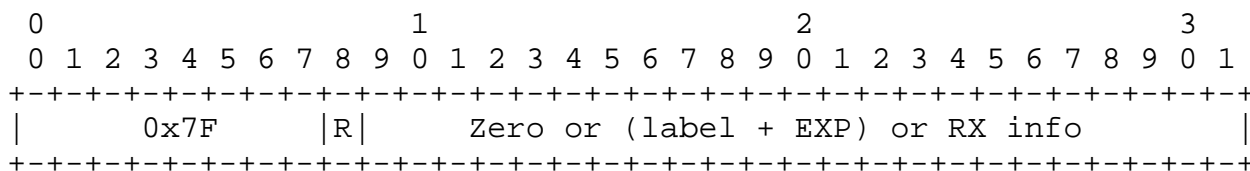
### 10.1. MPLS Label Verifications Mechanism

With full and partial reachability validations, initiator has the ability to determine if target identifier received the packet on any interfaces. This section describes additional mechanism for initiator to determine if target identifier received the packet on a specific interface.

So far for MPLS based sessions, this mechanism makes use of destination IP address of 127/8 range for IPv4 and of

0:0:0:0:0:FFFF:7F00/104 range for IPv6, in both directions. In this section, 127/8 will be used to describe the MPLS label verification mechanism. However, same concept is to be applied to IPv6 range 0:0:0:0:0:FFFF:7F00/104.

When a network node wishes to perform MPLS label verification, BFD control packet will have lower 23 bits of 127/8 destination IP address embedded with non-zero value. One such non-zero value MAY be (label value + EXP) that is used to reach intended target identifier. Receiver of this BFD control packet, if last 23 bits of 127/8 address is not zero, then will embed information reflecting how the packet was received in the lower 23 bits of 127/8 destination IP address in the response BFD control packet. If responder received the BFD control packet on a non-point-to-point interface, source MAC address MAY need to be examined to determine the "RX info" to embed in the returning packet.



9th bit is reserved for the time being and SHOULD be set to zero and SHOULD be ignored on receipt, by both initiator and responder

Initiator receiving back a response will know that packet did reach intended identifier. Initiator can also look into lower 23 bits of IP destination address in received BFD control packet to determine if packet sent was received by intended identifier in expected way (ex: expected RX interface).

When (label + EXP) is being encoded, label is specified in higher 20 bits of 23 bits and EXP is specified in lower 3 bits of 23 bits.

If a response BFD control packet is received, then initiator can conclude that a packet has reached intended node correctly. With information embedded in last 23 bits of response BFD control packet from responder, initiator has the ability to perform further verifications on how responded node received BFD control packet.

## 10.2. Localhost Address Usage

Last 23 bits of 127/8 for IPv4 and 0:0:0:0:0:FFFF:7F00/104 for IPv6 being non-zero is the trigger for responder to embed RX information in the response. When initiator is performing only reachability validations to target identifiers, then last 23 bits of the localhost address SHOULD be zero. This is to ensure unnecessary processing at responder is eliminated. However, last 23 bits of the localhost address MAY be set to a non-zero value to traverse specific ECMP path if required. Obvious side effect is the additional processing at responder to populate the RX info in response packet.

## 11. Scaling Aspect

This mechanism brings forth one noticeable difference in terms of scaling aspect: number of BFD sessions. This specification eliminates the need for egress nodes to have fully active BFD sessions when only one side desires to perform reachability validations. With introduction of reflector BFD concept, egress no longer is required to create any active BFD session per path/LSP basis. Due to this, total number of BFD sessions in a network is reduced.

If traditional BFD technology was used on a network comprised of N nodes, and each node monitored M unidirectional paths/LSPs, then total number of BFD sessions in such network will be:

$$((N - 1) \times M) \times 2$$

Assuming that each network node creates one reflector BFD session to handle all local BFD target identifiers, then total number of BFD sessions in same scenario will be:

$$((N - 1) \times M) + N$$

## 12. Co-existence with Traditional BFD

This mechanism has no issues being deployed with traditional BFDs ([RFC5881]/[RFC5883]/[RFC5884]/[RFC5885]) because BFD discriminators which allow this mechanism to function are explicitly reserved and separate UDP port values are used with S-BFD.

## 13. BFD Echo

BFD echo is outside the scope of this document.

## 14. Summary

Conceptually, Seamless BFD is as a way to perform BFD Echo Mode using BFD control packets. Critical differentiator being that target (ex: egress) is still required to respond. This allows greater control of a session to the initiator while required target (ex: egress) response allows for proper validations.

This section visits each aspect specified in the Introduction (Section 1) and describes how Seamless BFD provides beneficial impacts.

- o Two sided BFD a MUST?

Active BFD session instances are only created on network nodes that desire to validate/monitor reachability to specific targets through specific transports. It is pre-created reflector BFD sessions which operate Seamless BFD functionality at egress in all cases. Thus, it is no longer required for egress to create BFD sessions specific for paths/LSPs which are terminating on own network node. Therefore, Seamless BFD is a nice fit for scenarios where only one side is wanting to perform the BFD check.

- o Faster BFD bring-up?

Reflector BFD sessions are persistent entities provisioned in the network ahead of time, on relevant network nodes. When a network node desires to perform a reachability validation to a particular target, which already has a reflector BFD session monitoring the BFD target identifier, then generating the a Seamless BFD control packet and receiving back a Seamless BFD control packet is all that is required. It is no longer required for egress to create a specific BFD session instance nor for BFD sessions to go through FSM based on sedated bring-up intervals. Thus reachability validation is virtually instantaneous.

- o Why end-to-end only?

Seamless BFD creates separation of transport and intended receiver of the packet. IP destination address or MPLS label stack of BFD control packets describes particular paths while "your discriminator" describes intended receiver of such packets. Thus it is possible to inject BFD control packets from a transit node of a LSP. It is also possible, with careful TTL manipulations, for a network node to test reachability of a path/LSP to a particular transit node.

- o Is it taking the right path?



MPLS label verification aspect of Seamless BFD allows for testing of label programming. If certain MPLS label stack with certain "your discriminator" results in a response packet to be received back, then a node can conclude that the packet reached intended receiver based on imposed MPLS label stack. Also by examining "RX info" of received back BFD control packet, a node can determine if intended receiver received the packet in expected way (ex: on expected incoming interface).

- o Is one really enough?

With Seamless BFD, a network node is free to create any number of BFD session instances to a target, even if encapsulations of all such sessions are exactly the same. Because each BFD session instance will have a unique "my discriminator", response BFD control packets can get demultiplexed correctly into right session.

## 15. Security Considerations

Same security considerations as [RFC5880], [RFC5881], [RFC5883], [RFC5884] and [RFC5885] apply to this document.

Additionally, implementing following measures will strengthen security aspects of this mechanism described by this document.

- o Implementations MUST provide filtering capability based on source IP addresses or source node segment IDs of received BFD control packets: [RFC2827].
- o Implementations MUST NOT act on received BFD control packets containing Martian addresses as source IP addresses.
- o Implementations MUST ensure response target IP addresses or node segment IDs are reachable.

## 16. IANA Considerations

BFD Target Identifier types:

Value	BFD Target Identifier Type
-----	-----
0	Reserved
1	IP (IPv4 Address and Router ID)
2	Segment Routing Node Segment ID

New UDP port number(s) will be requested for S-BFD.

## 17. Acknowledgements

Authors would like to thank Girija Raghavendra Rao, Marc Binderberger, Srihari Raghavan, Vanitha Neelamegam and Vengada Prasad Govindan from Cisco Systems for providing valuable comments.

## 18. Contributing Authors

Tarek Saad  
Cisco Systems  
Email: tsaad@cisco.com

Siva Sivabalan  
Cisco Systems  
Email: msiva@cisco.com

Nagendra Kumar  
Cisco Systems  
Email: naikumar@cisco.com

Mallik Mudigonda  
Cisco Systems  
Email: mmudigon@cisco.com

## 19. References

### 19.1. Normative References

- [I-D.previdi-filsfils-isis-segment-routing]  
Previdi, S., Filsfils, C., Bashandy, A., Horneffer, M., Decraene, B., Litkowski, S., Milojevic, I., Shakir, R., Ytti, S., Henderickx, W., and J. Tantsura, "Segment Routing with IS-IS Routing Protocol", draft-previdi-filsfils-isis-segment-routing-02 (work in progress), March 2013.
- [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.
- [RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, June 2010.
- [RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, June 2010.

- [RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, June 2010.

## 19.2. Informative References

- [I-D.ietf-bfd-on-lags]  
Bhatia, M., Chen, M., Boutros, S., Binderberger, M., and J. Haas, "Bidirectional Forwarding Detection (BFD) on Link Aggregation Group (LAG) Interfaces", draft-ietf-bfd-on-lags-01 (work in progress), June 2013.
- [RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.
- [RFC4379] Kompella, K. and G. Swallow, "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", RFC 4379, February 2006.
- [RFC5885] Nadeau, T. and C. Pignataro, "Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)", RFC 5885, June 2010.
- [RFC6428] Allan, D., Swallow Ed. , G., and J. Drake Ed. , "Proactive Connectivity Verification, Continuity Check, and Remote Defect Indication for the MPLS Transport Profile", RFC 6428, November 2011.

## Authors' Addresses

Nobo Akiya  
Cisco Systems

Email: nobo@cisco.com

Carlos Pignataro  
Cisco Systems

Email: cpignata@cisco.com

Dave Ward  
Cisco Systems

Email: wardd@cisco.com

Manav Bhatia  
Alcatel-Lucent

Email: manav.bhatia@alcatel-lucent.com

Santosh  
Juniper Networks

Email: santoshpk@juniper.net