

SPRING
Internet-Draft
Intended status: Informational
Expires: September 6, 2018

Z. Ali
K. Talaulikar
C. Filsfils
Cisco Systems
March 5, 2018

Bidirectional Forwarding Detection (BFD) for Segment Routing Policies
for Traffic Engineering
draft-ali-spring-bfd-sr-policy-00

Abstract

Segment Routing (SR) allows a headend node to steer a packet flow along any path using a segment list which is referred to as a SR Policy. Intermediate per-flow states are eliminated thanks to source routing. The header of a packet steered in an SR Policy is augmented with the ordered list of segments associated with that SR Policy. Bidirectional Forwarding Detection (BFD) is used to monitor different kinds of paths between node. BFD mechanisms can be also used to monitor the availability of the path indicated by a SR Policy and to detect any failures. Seamless BFD (SBFD) extensions provide a simplified mechanism which is suitable for monitoring of paths that are setup dynamically and on a large scale.

This document describes the use of Seamless BFD (SBFD) mechanism to monitor the SR Policies that are used for Traffic Engineering (TE) in SR deployments.

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 <https://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 September 6, 2018.

Copyright Notice

Copyright (c) 2018 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 (<https://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	2
2. Choice of SBFD over BFD	3
3. Procedures	4
4. IANA Considerations	5
5. Security Considerations	6
6. Contributors	6
7. Acknowledgements	6
8. References	6
8.1. Normative References	6
8.2. Informative References	7
Authors' Addresses	8

1. Introduction

Segment Routing (SR) ([I-D.ietf-spring-segment-routing]) allows a headend node to steer a packet flow along any path for specific objectives like Traffic Engineering (TE) and to provide it treatment according to the specific established service level agreement (SLA) for it. Intermediate per-flow states are eliminated thanks to source routing. The headend node steers a flow into an SR Policy. The header of a packet steered in an SR Policy is augmented with the ordered list of segments associated with that SR Policy. SR Policy [I-D.filsfils-spring-segment-routing-policy] specifies the concepts of SR Policy and steering into an SR Policy.

SR Policy state is instantiated only on the head-end node and any intermediate node or the endpoint node does not require any state to be maintained or instantiated for it. SR Policies are not signaled through the network nodes except the signaling required to instantiate them on the head-end in the case of a controller based deployment. This enables SR Policies to scale far better than previous TE mechanisms. This also enables SR Policies to be instantiated dynamically and on demand basis for steering specific traffic flows corresponding to service routes as they are signaled. These automatic steering and signaling mechanisms for SR Policies are described in SR Policy [I-D.filsfils-spring-segment-routing-policy].

There is a requirement to continuously monitor the availability of the path corresponding to the SR Policy along the nodes in the network and to signal any failures detected to the head-end node so that it could take corrective action to restore service. The corrective actions may be either to invalidate the candidate path that has experienced failure and to switch to another candidate path within the same SR Policy OR to activate another backup SR Policy or candidate path for end-to-end path protection. These mechanisms are beyond the scope of this document.

Bidirectional Forwarding Detection (BFD) mechanisms have been specified for use for monitoring of unidirectional MPLS LSPs via BFD MPLS [RFC5884]. Seamless BFD [RFC7880] defines a simplified mechanism for using BFD with a large proportion of negotiation aspects eliminated, thus providing benefits such as quick provisioning, as well as improved control and flexibility for network nodes initiating path monitoring. When BFD or SBFD is used for verification of such unidirectional LSP paths, the reverse path is via the shortest path from the tail-end router back to the head-end router as determined by routing.

The SR Policy is essentially a unidirectional path through the network. This document describes the use of BFD and more specifically SBFD for monitoring of SR Policy paths through the network. SR can be instantiated using both MPLS and IPv6 dataplanes. The mechanism described in this document applies to both these instantiations of SR Policy.

2. Choice of SBFD over BFD

BFD MPLS [RFC5884] describes a mechanism where LSP Ping [RFC8029] is used to bootstrap the BFD session over an MPLS TE LSP path. The LSP Ping mechanism was extended to support SR LSPs via SR LSP Ping [RFC8287] and a similar mechanism could have been considered for BFD monitoring of SR Policies on MPLS data-plane. However, this document

proposes instead to use SBFD mechanism as it is more suitable for SR Policies.

Some of the key aspects of SR Policies that are considered in arriving at this decision are as follows:

- o SR Policies do not require any signaling to be performed through the network nodes in order to be setup. They are simply instantiated on the head-end node via provisioning or even dynamically by a controller via BGP SR-TE [I-D.ietf-idr-segment-routing-te-policy] or using PCEP (PCEP SR [I-D.ietf-pce-segment-routing], PCE Initiated [RFC8281], PCEP Stateful [RFC8231]).
- o SR Policies result in state being instantiated only on the head-end node and no other node in the network.
- o In many deployments, SR Policies are instantiated dynamically and on-demand or in the case of automated steering for BGP routes, when routes are learnt with specific color communities (refer SR Policy [I-D.filsfils-spring-segment-routing-policy] for details).
- o SR Policies are expected to be deployed in much higher scale.
- o SR Policies can be instantiated both for MPLS and IPv6 data-planes and hence a monitoring mechanism which works for both is desirable.

In view of the above, the BFD mechanism to be used for monitoring them needs to be simple, lightweight, one that does not result in instantiation of per SR Policy state anywhere but the head-end and which can be setup and deleted dynamically, on-demand and at scale. The SBFD extensions provide this support as described in Seamless BFD [RFC7880]. Furthermore, SBFD Use-Cases [RFC7882] clarifies the applicability in the Centralized TE and SR scenarios.

3. Procedures

The general procedures and mechanisms for SBFD operations are specified in Seamless BFD [RFC7880]. This section describes the specifics related to SBFD use for SR Policies.

SR Policies are represented on a head-end router as <color,endpoint IP address> tuple. The SRTE process on the head-end determines the tail-end node of a SR Policy on the basis of the endpoint IP address. In the cases where the SR Policy endpoint is outside the domain of the head-end node, this information is available with the centralized

controller that computed the multi-domain SR Policy path for the head-end.

In order to enable SBFD monitoring for a given SR Policy, the SBFD Discriminator for the tail-end node (i.e. one with the endpoint IP address) which is going to be the SBFD Reflector is required. ISIS SBFD [RFC7883] and OSPF SBFD [RFC7884] describe the extensions to the ISIS and OSPF link state routing protocols that allow all nodes to advertise their SBFD Discriminators across the network. BGP-LS SBFD [I-D.li-idr-bgp-ls-sbfd-extensions] describes extensions for advertising the SBFD discriminators via BGP-LS across domains and to a controller. Thus, either the SRTE head-end node or the controller, as the case may be, have the SBFD Discriminator of the tail-end node of the SR Policy available.

The SRTE Process can straightaway instantiate the SBFD mechanism on the SR Policy as soon as it is provisioned in the forwarding to start verification of the path to the endpoint. No signaling or provisioning is required for the tail-end node on a per SR Policy basis and it just performs its role as a stateless SBFD Reflector. The return path used by SBFD is via the normal IP routing back to the head-end node. Once the specific SR Policy path is verified via SBFD, then it is considered as active and may be used for traffic steering.

The SBFD monitoring continues for the SR Policy and any failure is notified to the SRTE process. In response to the failure of a specific candidate path, the SRTE process may trigger any of the following based on local policy or implementation specific aspects which are outside the scope of this document:

- o Trigger path-protection for the SR Policy
- o Declare the specific candidate path as invalid and switch to using the next valid candidate path based on preference
- o If no alternate candidate path is available, then handle the steering over that SR Policy based on its invalidation policy (e.g. drop or switch to best effort routing).

4. IANA Considerations

None

5. Security Considerations

Procedures described in this document do not affect the BFD or Segment Routing security model. See the 'Security Considerations' section of [RFC7880] for a discussion of SBFD security and to [I-D.ietf-spring-segment-routing] for analysis of security in SR deployments.

6. Contributors

Nagendra Kumar
Cisco Systems Inc.

Email: naikumar@cisco.com

Mallik Mudigonda
Cisco Systems Inc.

Email: mmudigon@cisco.com

7. Acknowledgements

8. References

8.1. Normative References

- [I-D.filsfils-spring-segment-routing-policy]
Filsfils, C., Sivabalan, S., Raza, K., Liste, J., Clad, F., Talaulikar, K., Ali, Z., Hegde, S., daniel.voyer@bell.ca, d., Lin, S., bogdanov@google.com, b., Krol, P., Horneffer, M., Steinberg, D., Decraene, B., Litkowski, S., and P. Mattes, "Segment Routing Policy for Traffic Engineering", draft-filsfils-spring-segment-routing-policy-05 (work in progress), February 2018.
- [I-D.ietf-spring-segment-routing]
Filsfils, C., Previdi, S., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", draft-ietf-spring-segment-routing-15 (work in progress), January 2018.
- [I-D.li-idr-bgp-ls-sbfd-extensions]
Li, Z., Aldrin, S., Tantsura, J., Mirsky, G., and S. Zhuang, "BGP Link-State Extensions for Seamless BFD", draft-li-idr-bgp-ls-sbfd-extensions-01 (work in progress), April 2017.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC7880] Pignataro, C., Ward, D., Akiya, N., Bhatia, M., and S. Pallagatti, "Seamless Bidirectional Forwarding Detection (S-BFD)", RFC 7880, DOI 10.17487/RFC7880, July 2016, <<https://www.rfc-editor.org/info/rfc7880>>.
- [RFC7882] Aldrin, S., Pignataro, C., Mirsky, G., and N. Kumar, "Seamless Bidirectional Forwarding Detection (S-BFD) Use Cases", RFC 7882, DOI 10.17487/RFC7882, July 2016, <<https://www.rfc-editor.org/info/rfc7882>>.
- [RFC7883] Ginsberg, L., Akiya, N., and M. Chen, "Advertising Seamless Bidirectional Forwarding Detection (S-BFD) Discriminators in IS-IS", RFC 7883, DOI 10.17487/RFC7883, July 2016, <<https://www.rfc-editor.org/info/rfc7883>>.
- [RFC7884] Pignataro, C., Bhatia, M., Aldrin, S., and T. Ranganath, "OSPF Extensions to Advertise Seamless Bidirectional Forwarding Detection (S-BFD) Target Discriminators", RFC 7884, DOI 10.17487/RFC7884, July 2016, <<https://www.rfc-editor.org/info/rfc7884>>.

8.2. Informative References

- [I-D.ietf-idr-segment-routing-te-policy] Previdi, S., Filsfils, C., Jain, D., Mattes, P., Rosen, E., and S. Lin, "Advertising Segment Routing Policies in BGP", draft-ietf-idr-segment-routing-te-policy-02 (work in progress), March 2018.
- [I-D.ietf-pce-segment-routing] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "PCEP Extensions for Segment Routing", draft-ietf-pce-segment-routing-11 (work in progress), November 2017.
- [RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <<https://www.rfc-editor.org/info/rfc5884>>.

- [RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <<https://www.rfc-editor.org/info/rfc8029>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8287] Kumar, N., Ed., Pignataro, C., Ed., Swallow, G., Akiya, N., Kini, S., and M. Chen, "Label Switched Path (LSP) Ping/Traceroute for Segment Routing (SR) IGP-Prefix and IGP-Adjacency Segment Identifiers (SIDs) with MPLS Data Planes", RFC 8287, DOI 10.17487/RFC8287, December 2017, <<https://www.rfc-editor.org/info/rfc8287>>.

Authors' Addresses

Zafar Ali
Cisco Systems

Email: zali@cisco.com

Ketan Talaulikar
Cisco Systems

Email: ketant@cisco.com

Clarence Filsfils
Cisco Systems

Email: cfilsfil@cisco.com

Network Working Group
Internet Draft
Intended Status: Informational
Expiration Date: February 4, 2019

E. Chen
N. Shen
Cisco Systems
R. Raszuk
Bloomberg LP
August 3, 2018

Unsolicited BFD for Sessionless Applications
draft-chen-bfd-unsolicited-03.txt

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/lid-abstracts.html>

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>

This Internet-Draft will expire on February 4, 2019.

Copyright Notice

Copyright (c) 2018 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.

Abstract

For operational simplification of "sessionless" applications using BFD, in this document we present procedures for "unsolicited BFD" that allow a BFD session to be initiated by only one side, and be established without explicit per-session configuration or registration by the other side (subject to certain per-interface or per-router policies).

1. Introduction

The current implementation and deployment practice for BFD ([RFC5880] and [RFC5881]) usually requires BFD sessions be explicitly configured or registered on both sides. This requirement is not an issue when an application like BGP [RFC4271] has the concept of a "session" that involves both sides for its establishment. However, this requirement can be operationally challenging when the prerequisite "session" does not naturally exist between two endpoints in an application. Simultaneous configuration and coordination may be required on both sides for BFD to take effect. For example:

- o When BFD is used to keep track of the "liveness" of the nexthop of static routes. Although only one side may need the BFD functionality, currently both sides need to be involved in specific configuration and coordination and in some cases static routes are created unnecessarily just for BFD.
- o When BFD is used to keep track of the "liveness" of the third-party nexthop of BGP routes received from the Route Server [RFC7947] at an Internet Exchange Point (IXP). As the third-party nexthop is different from the peering address of the Route Server, for BFD to work, currently two routers peering with the Route Server need to have routes and nexthops from each other (although indirectly via the Router Server), and the nexthop of each router must be present at the same time. These issues are also discussed in [I-D.ietf-idr-rs-bfd].

Clearly it is beneficial and desirable to reduce or eliminate unnecessary configurations and coordination in these "sessionless" applications using BFD.

In this document we present procedures for "unsolicited BFD" that allow a BFD session to be initiated by only one side, and be established without explicit per-session configuration or

registration by the other side (subject to certain per-interface or per-router policies).

With "unsolicited BFD" there is potential risk for excessive resource usage by BFD from "unexpected" remote systems. To mitigate such risks, several mechanisms are recommended in the Security Considerations section.

Compared to the "Seamless BFD" [RFC7880], this proposal involves only minor procedural enhancements to the widely deployed BFD itself. Thus we believe that this proposal is inherently simpler in the protocol itself and deployment. As an example, it does not require the exchange of BFD discriminators over an out-of-band channel before the BFD session bring-up.

When BGP Add-Path [RFC7911] is deployed at an IXP using the Route Server, multiple BGP paths (when exist) can be made available to the clients of the Router Server as described in [RFC7947]. The "unsolicited BFD" can be used in BGP route selection by these clients to eliminate paths with "inaccessible nexthops".

1.1. Specification of Requirements

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 [RFC2119].

2. Procedures for Unsolicited BFD

With "unsolicited BFD", one side takes the "Active role" and the other side takes only the "Passive role" as described in [RFC5880].

On the passive side, the "unsolicited BFD" SHOULD be configured explicitly on an interface. The BFD parameters can be either per-interface or per-router based. It MAY also choose to use the parameters that the active side uses in its BFD Control packets. The "Discriminator", however, MUST be chosen to allow multiple unsolicited BFD sessions.

The active side initiates the BFD Control packets as specified in [RFC5880]. The passive side does not initiate the BFD Control packets.

When the passive side receives a BFD Control packet from the active side with 0 as the "remote-discriminator", and it does not find an existing session with the same source address as in the packet and

"unsolicited BFD" is allowed on the interface by local policy, it SHOULD then create a matching BFD session toward the active side (based on the source address and destination address in the BFD Control packet) as if the session were locally registered. It would then start sending the BFD Control packets and perform necessary procedure for bringing up, maintaining and tearing down the BFD session. If the BFD session fails to get established within certain specified time, or if an established BFD session goes down, the passive side would stop sending BFD Control packets and delete the BFD session created until the BFD Control packets is initiated by the active side again.

The "Passive role" may change to the "Active role" when a local client registers for the same BFD session, and from the "Active role" to the "Passive role" when there is no longer any locally registered client for the BFD session.

3. IANA Considerations

This documents makes no IANA requests.

4. Security Considerations

The same security considerations as those described in [RFC5880] and [RFC5881] apply to this document. With "unsolicited BFD" there is potential risk for excessive resource usage by BFD from "unexpected" remote systems. To mitigate such risks, the following measures are RECOMMENDED:

- o Limit the feature to specific interfaces, and to a single-hop BFD with "TTL=255" [RFC5082]. In addition make sure the source address of an incoming BFD packet belongs to the subnet of the interface from which the BFD packet is received.
- o Apply "access control" to allow BFD packets only from certain subnets or hosts.
- o Deploy the feature only in certain "trustworthy" environment, e.g., at an IXP, or between a provider and its customers.
- o Adjust BFD parameters as needed for the particular deployment and scale.
- o Use BFD authentication.

5. Acknowledgments

TBD

6. References

6.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., Ed., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, October 2007.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<http://www.rfc-editor.org/info/rfc5880>>.
- [RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, DOI 10.17487/RFC5881, June 2010, <<http://www.rfc-editor.org/info/rfc5881>>.

6.2. Informative References

- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <<http://www.rfc-editor.org/info/rfc4271>>.
- [RFC7880] Pignataro, C., Ward, D., Akiya, N., Bhatia, M., and S. Pallagatti, "Seamless Bidirectional Forwarding Detection (S-BFD)", RFC 7880, DOI 10.17487/RFC7880, July 2016, <<http://www.rfc-editor.org/info/rfc7880>>.
- [RFC7911] Walton, D., Retana, A., Chen, E., and J. Scudder, "Advertisement of Multiple Paths in BGP", RFC 7911, DOI 10.17487/RFC7911, July 2016, <<http://www.rfc-editor.org/info/rfc7911>>.
- [RFC7947] Jasinska, E., Hilliard, N., Raszuk, R., and N. Bakker, "Internet Exchange BGP Route Server", RFC 7947,

DOI 10.17487/RFC7947, September 2016,
<<http://www.rfc-editor.org/info/rfc7947>>.

[I-D.ietf-idr-rs-bfd]

Bush, R., J. Haas, J. Scudder, A. Nipper, and T. King,
"Making Route Servers Aware of Data Link Failures at
IXPs", draft-ietf-idr-rs-bfd-03 (work in progress), July
2017.

7. Authors' Addresses

Enke Chen
Cisco Systems
560 McCarthy Blvd.
Milpitas, CA 95035
USA

Email: enkechen@cisco.com

Naiming Shen
Cisco Systems
560 McCarthy Blvd.
Milpitas, CA 95035
USA

Email: naiming@cisco.com

Robert Raszuk
Bloomberg LP
731 Lexington Ave
New York City, NY 10022
USA

Email: robert@raszuk.net

INTERNET-DRAFT
Intended status: Proposed Standard

V. Govindan
M. Mudigonda
A. Sajassi
Cisco Systems
G. Mirsky
ZTE
D. Eastlake
Huawei
May 25, 2018

Expires: November 24, 2018

Fault Management for EVPN networks
draft-gsm-bess-evpn-bfd-01

Abstract

This document specifies a proactive, in-band network OAM mechanism to detect loss of continuity and miss-connection faults that affect unicast and multi-destination paths, used by Broadcast, unknown Unicast and Multicast traffic, in an EVPN network. The mechanisms proposed in the draft use the widely adopted Bidirectional Forwarding Detection (BFD) protocol.

Status of this Memo

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

Distribution of this document is unlimited. Comments should be sent to the authors or the BESSq working group mailing list: bess@ietf.org.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/lid-abstracts.html>. The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Table of Contents

1. Introduction.....	3
1.1 Terminology.....	3
2. Scope of this Document.....	4
3. Motivation for Running BFD at the EVPN Network Layer....	4
4. Fault Detection of Unicast Traffic.....	6
5. Fault Detection for BUM Traffic.....	7
5.1 Ingress Replication.....	7
5.2 Label Switched Multicast.....	7
6. BFD Packet Encapsulation.....	8
6.1 Using GAL/G-ACh Encapsulation Without IP Headers.....	8
6.1.1 Ingress Replication.....	8
6.1.1.1 Alternative Encapsulation Format.....	8
6.1.2 LSM (Label Switched Multicast).....	9
6.1.3 Unicast.....	9
6.1.3.1 Alternative Encapsulation Format.....	9
6.2 Using IP Headers.....	10
7. Scalability Considerations.....	11
8. IANA Considerations.....	12
9. Security Considerations.....	13
Normative References.....	14
Informative References.....	15
Authors' Addresses.....	17

1. Introduction

[I-D.eastlake-bess-evpn-oam-req-frmwk] and [I-D.ooamdt-rtgwg-ooam-requirement] outline the OAM requirements of Ethernet VPN networks [RFC7432]. This document proposes mechanisms for proactive fault detection at the network (overlay) OAM layer of EVPN. EVPN fault detection mechanisms need to consider unicast traffic separately from Broadcast, unknown Unicast, and Multicast (BUM) traffic since they map to different FECs in EVPN, hence this document proposes different fault detection mechanisms to suit each type using the principles of [RFC5880], [RFC5884] and Point-to-multipoint BFD [I-D.ietf-bfd-multipoint] and [I-D.ietf-bfd-multipoint-active-tail]. Packet loss and packet delay measurement are out of scope for this document.

1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following acronyms are used in this document.

BUM - Broadcast, Unknown Unicast, and Multicast

CC - Continuity Check

CV - Connectivity Verification

FEC - Forwarding Equivalency Class

GAL - Generic Associated Channel Label

LSM - Label Switched Multicast (P2MP)

LSP - Label Switched Path

MP2P - Multi-Point to Point

OAM - Operations Administration, and Maintenance

P2MP - Point to Multi-Point (LSM)

PE - Provider Edge

PHP - Penultimate Hop Popping

2. Scope of this Document

This document specifies proactive fault detection for EVPN [RFC7432] using BFD mechanisms for:

- o Unicast traffic.
- o BUM traffic using Multi-point-to-Point (MP2P) tunnels (ingress replication).
- o BUM traffic using Point-to-Multipoint (P2MP) tunnels (LSM).

This document does not discuss BFD mechanisms for:

- o EVPN variants like PBB-EVPN [RFC7623]. This will be addressed in future versions.
- o Integrated Routing and Bridging (IRB) solution based on EVPN [I-D.ietf-bess-evpn-inter-subnet-forwarding]. This will be addressed in future versions.
- o EVPN using other encapsulations like VxLAN, NVGRE and MPLS over GRE [RFC8365].
- o BUM traffic using MP2MP tunnels will also be addressed in a future version of this document.

This specification describes procedures only for BFD asynchronous mode. BFD demand mode is outside the scope of this specification. Further, the use of the Echo function is outside the scope of this specification.

3. Motivation for Running BFD at the EVPN Network Layer

The choice of running BFD at the network layer of the OAM model for EVPN [I-D.eastlake-bess-evpn-oam-req-frmwk] and [I-D.ooamdt-rtgwg-ooam-requirement] was made after considering the following:

- o In addition to detecting link failures in the EVPN network, BFD sessions at the network layer can be used to monitor the successful programming of labels used for setting up MP2P and P2MP EVPN tunnels transporting Unicast and BUM traffic. The scope of reachability detection covers the ingress and the egress EVPN PE nodes and the network connecting them.
- o Monitoring a representative set of path(s) or a particular path among the multiple paths available between two EVPN PE nodes could be done by exercising the entropy labels when they are used.

However paths that cannot be realized by entropy variations cannot be monitored. Fault monitoring requirements outlined by [I-D.eastlake-bess-evpn-oam-req-frmwk] are addressed by the mechanisms proposed by this draft.

Successful establishment and maintenance of BFD sessions between EVPN PE nodes does not fully guarantee that the EVPN service is functioning. For example, an egress EVPN-PE can understand the EVPN label but could switch data to incorrect interface. However, once BFD sessions in the EVPN Network Layer reach UP state, it does provide additional confidence that data transported using those tunnels will reach the expected egress node. When the BFD session in EVPN overlay goes down that can be used as an indication of a Loss-of-Connectivity defect in the EVPN underlay that would cause EVPN service failure.

4. Fault Detection of Unicast Traffic

The mechanisms specified in BFD for MPLS LSPs [RFC5884] [RFC7726] can be applied to bootstrap and maintain BFD sessions for unicast EVPN traffic. The discriminators required for de-multiplexing the BFD sessions MUST be exchanged using EVPN LSP ping specifying the Unicast EVPN FEC [I-D.jain-bess-evpn-lsp-ping] before establishing the BFD session. This is needed since the MPLS label stack does not contain enough information to disambiguate the sender of the packet.

The usage of MPLS entropy labels takes care of the requirement to monitor various paths of the multi-path server layer network [RFC6790]. Each unique realizable path between the participating PE routers MAY be monitored separately when entropy labels are used. The multi-path connectivity between two PE routers MUST be tracked by at least one representative BFD session, but in that case the granularity of fault-detection would be coarser. The PE node receiving the EVPN LSP ping MUST allocate BFD discriminators using the procedures defined in [RFC7726]. Once the BFD session for the EVPN label is UP, the ends of the BFD session MUST NOT change the local discriminator values of the BFD Control packets they generate, unless they first brings down the session as specified in [RFC5884].

5. Fault Detection for BUM Traffic

5.1 Ingress Replication

Ingress replication uses separate MP2P tunnels for transporting BUM traffic from the ingress PE (head) to a set of one or more egress PEs (tails). The fault detection mechanism specified by this document takes advantage of the fact that a unique copy is made by the head for each tail. Another key aspect to be considered in EVPN is the advertisement of the inclusive multicast route. The BUM traffic flows from a head node to a particular tail only after the head receives the inclusive multicast route containing the BUM EVPN label (downstream allocated) corresponding to the MP2P tunnel.

The head-end PE performing ingress replication MUST initiate an EVPN LSP ping using the inclusive multicast FEC [I-D.jain-bess-evpn-lsp-ping] upon receiving an inclusive multicast route from a tail to bootstrap the BFD session. There MAY exist multiple BFD sessions between a head PE and an individual tail due to the usage of entropy labels [RFC6790] for an inclusive multicast FEC. The PE node receiving the EVPN LSP ping MUST allocate BFD discriminators using the procedures defined in [RFC7726]. Once the BFD session for the EVPN label is UP, the ends of the BFD session MUST NOT change the local discriminator values of the BFD Control packets they generate, unless they first bring down the session as specified in [RFC5884].

5.2 Label Switched Multicast

Fault detection for BUM traffic distributed by a Label Switched Multicast (LSM) using a P2MP tunnel is done with active tail multipoint BFD in the reliable head notification scenario (see [I-D.ietf-bfd-multipoint] and [I-D.ietf-bfd-multipoint-active-tail] particularly Section 3.4).

TBD...

6. BFD Packet Encapsulation

6.1 Using GAL/G-ACh Encapsulation Without IP Headers

This section describes use of the Generic Associated Channel Label (GAL/G-ACh).

6.1.1 Ingress Replication

The packet contains the following labels: LSP label (transport) when not using PHP (Penultimate Hop Popping), the optional entropy label, the BUM label and the SH label [RFC7432] (where applicable). The G-ACh type is set to TBD1. The G-ACh payload of the packet MUST contain the L2 header (in overlay space) followed by the IP header encapsulating the BFD packet. The MAC address of the inner packet is used to validate the <EVI, MAC> in the receiving node. The discriminator values of BFD are obtained through negotiation through the out-of-band EVPN LSP ping.

6.1.1.1 Alternative Encapsulation Format

A new TLV can be defined as proposed in Sec 3 of [RFC6428] to include the EVPN FEC information as a TLV following the BFD Control packet.

The format of the TLV can be reused from the EVPN Inclusive Multicast sub-TLV proposed by Fig 2 of [I-D.jain-bess-evpn-lsp-ping].

A new type (TBD3) to indicate the EVPN Inclusive Multicast SubTLV is requested from the "CC/ CV MEP-ID TLV" registry [RFC6428].

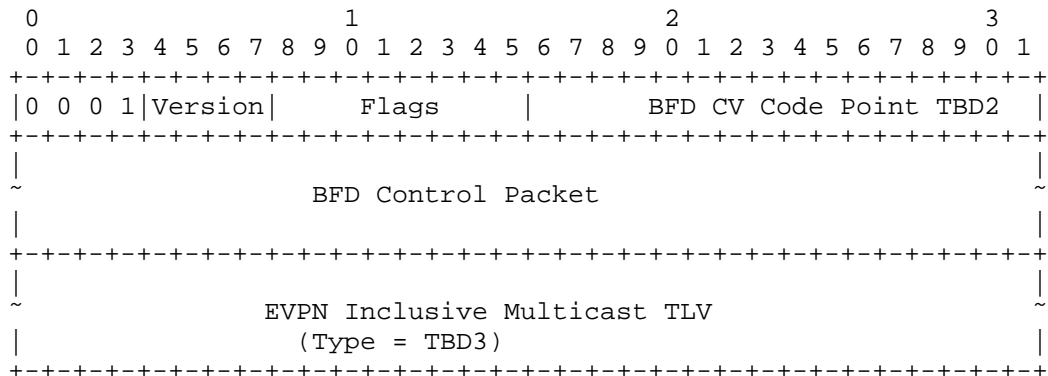


Figure 1: BFD-EVPN CV Message for EVPN Multicast
(Ingress Replication)

6.1.2 LSM (Label Switched Multicast)

TBD...

6.1.3 Unicast

The packet contains the following labels: LSP label (transport) when not using PHP, the optional entropy label and the EVPN Unicast label. The G-ACh type is set to TBD1. The G-ACh payload of the packet MUST contain the L2 header (in overlay space) followed by the IP header encapsulating the BFD packet. The MAC address of the inner packet is used to validate the <EVI, MAC> in the receiving node. The discriminator values for BFD are obtained through negotiation using the out-of-band EVPN ping.

6.1.3.1 Alternative Encapsulation Format

A new TLV can be defined as proposed in Sec 3 of [RFC6428] to include the EVPN FEC information as a TLV following the BFD Control packet. The format of the TLV can be reused from the EVPN MAC sub-TLV proposed by Figure 1 of [I-D.jain-bess-evpn-lsp-ping]. A new type (TBD4) to indicate the EVPN MAC SubTLV is requested from the "CC/ CV MEP-ID TLV" registry [RFC6428].

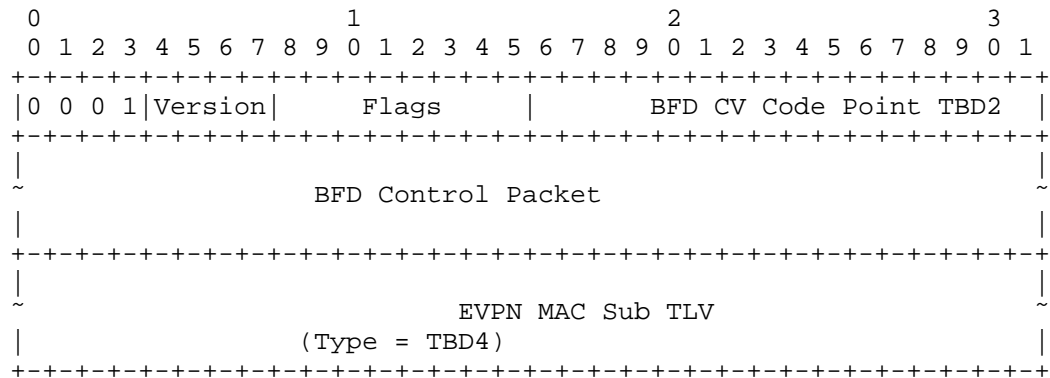


Figure 2: BFD-EVPN CV Message for EVPN Unicast

6.2 Using IP Headers

The encapsulation option using IP headers will not be suited for EVPN, as using different values in the destination IP address for data and OAM (BFD) packets could cause the BFD packets to follow a different path than that of data packets. Hence this option MUST NOT be used for EVPN.

7. Scalability Considerations

The mechanisms proposed by this draft could affect the packet load on the network and its elements especially when supporting configurations involving a large number of EVIs. The option of slowing down or speeding up BFD timer values can be used by an administrator or a network management entity to maintain the overhead incurred due to fault monitoring at an acceptable level.

8. IANA Considerations

IANA is requested to assign two channel types from the "Pseudowire Associated Channel Types" registry in [RFC4385] as follows.

Value	Description	Reference
-----	-----	-----
TBD1	EFD-EVPN CC	[this document]
TBD2	BFD-EVPN CV	[this document]

Ed Note: Do we need a CC code point? TBD

IANA is requested to assign the following code-points from the "CC/CV MEP-ID TLV" registry [RFC6428].

Value	Name	Reference
-----	-----	-----
TBD3	EVPN inclusive multicast	[this document]
TBD4	EVPN unicast	[this document]

9. Security Considerations

Security considerations discussed in [RFC5880], [RFC5883], and [RFC8029] apply.

MPLS security considerations [RFC5920] apply to BFD Control packets encapsulated in a MPLS label stack. When BFD Control packets are routed, the authentication considerations discussed in [RFC5883] should be followed.

Normative References

- [I-D.ietf-bess-evpn-inter-subnet-forwarding] Sajassi, A., Salam, S., Thoria, S., Rekhter, Y., Drake, J., Yong, L., and L. Dunbar, "Integrated Routing and Bridging in EVPN", draft-ietf-bess-evpn-inter-subnet-forwarding-03 (work in progress), October 2015.
- [I-D.ietf-bfd-multipoint] Katz, D., Ward, D., and J. Networks, "BFD for Multipoint Networks", draft-ietf-bfd-multipoint-16 (work in progress), April 2016.
- [I-D.ietf-bfd-multipoint-active-tail] Katz, D., Ward, D., and J. Networks, "BFD Multipoint Active Tails.", draft-ietf-bfd-multipoint-active-tail-07 (work in progress), May 2016.
- [I-D.jain-bess-evpn-lsp-ping] Jain, P., Boutros, S., and S. Salam, "LSP-Ping Mechanisms for EVPN and PBB-EVPN", draft-jain-bess-evpn-lsp-ping-06 (work in progress), May 2016.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <<http://www.rfc-editor.org/info/rfc4385>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<http://www.rfc-editor.org/info/rfc5880>>.
- [RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <<https://www.rfc-editor.org/info/rfc5883>>.
- [RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <<http://www.rfc-editor.org/info/rfc5884>>.
- [RFC6425] Saxena, S., Ed., Swallow, G., Ali, Z., Farrel, A., Yasukawa, S., and T. Nadeau, "Detecting Data-Plane Failures in Point-to-Multipoint MPLS - Extensions to LSP Ping", RFC 6425, DOI 10.17487/RFC6425, November 2011, <<https://www.rfc-editor.org/info/rfc6425>>.

- [RFC6428] Allan, D., Ed., Swallow, G., Ed., and J. Drake, Ed., "Proactive Connectivity Verification, Continuity Check, and Remote Defect Indication for the MPLS Transport Profile", RFC 6428, DOI 10.17487/RFC6428, November 2011, <<http://www.rfc-editor.org/info/rfc6428>>.
- [RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <<http://www.rfc-editor.org/info/rfc6790>>.
- [RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <<http://www.rfc-editor.org/info/rfc7432>>.
- [RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <<http://www.rfc-editor.org/info/rfc7623>>.
- [RFC7726] Govindan, V., Rajaraman, K., Mirsky, G., Akiya, N., and S. Aldrin, "Clarifying Procedures for Establishing BFD Sessions for MPLS Label Switched Paths (LSPs)", RFC 7726, DOI 10.17487/RFC7726, January 2016, <<http://www.rfc-editor.org/info/rfc7726>>.
- [RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <<https://www.rfc-editor.org/info/rfc8029>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <<https://www.rfc-editor.org/info/rfc8365>>.

Informative References

- [I-D.ooamdt-rtgwg-ooam-requirement] Kumar, N., Pignataro, C., Kumar, D., Mirsky, G., Chen, M., Nordmark, E., Networks, J., and D. Mozes, "Overlay OAM Requirements", draft-ooamdt-rtgwg-

oam-requirement-02 (work in progress), March 2016.

[I-D.eastlake-bess-evpn-oam-req-frmwk] Salam, S., Sajassi, A., Aldrin, S., and J. Drake, "EVPN Operations, Administration and Maintenance Requirements and Framework", draft-eastlake-bess-evpn-oam-req-frmwk-00 (work in progress), May 2018.

[RFC5920] Fang, L., Ed., "Security Framework for MPLS and GMPLS Networks", RFC 5920, DOI 10.17487/RFC5920, July 2010, <<https://www.rfc-editor.org/info/rfc5920>>.

Authors' Addresses

Vengada Prasad Govindan
Cisco Systems

Email: venggovi@cisco.com

Mudigonda Mallik
Cisco Systems

Email: mmudigon@cisco.com

Ali Sajassi
Cisco Systems
170 West Tasman Drive
San Jose, CA 95134, USA

Email: sajassi@cisco.com

Gregory Mirsky
ZTE Corp.

Email: gregimirsky@gmail.com

Donald Eastlake, 3rd
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Copyright, Disclaimer, and Additional IPR Provisions

Copyright (c) 2018 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.

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 19, 2019

J. Haas
Juniper Networks, Inc.
A. Fu
Bloomberg
October 16, 2018

BFD Encapsulated in Large Packets
draft-haas-bfd-large-packets-01

Abstract

The Bidirectional Forwarding Detection (BFD) protocol is commonly used to verify connectivity between two systems. BFD packets are typically very small. It is desirable in some circumstances to know that not only is the path between two systems reachable, but also that it is capable of carrying a payload of a particular size. This document discusses thoughts on how to implement such a mechanism using BFD in Asynchronous mode.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC2119] only when they appear in all upper case. They may also appear in lower or mixed case as English words, without normative meaning.

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 19, 2019.

Copyright Notice

Copyright (c) 2018 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	2
2. BFD Encapsulated in Large Packets	3
3. Implementation and Deployment Considerations	3
4. Security Considerations	3
5. IANA Considerations	4
6. References	4
6.1. Normative References	4
6.2. Informative References	4
Authors' Addresses	5

1. Introduction

The Bidirectional Forwarding Detection (BFD) [RFC5880] protocol is commonly used to verify connectivity between two systems. However, some applications may require that the Path MTU [RFC1191] between those two systems meets a certain minimum criteria. When the Path MTU decreases below the minimum threshold, those applications may wish to consider the path unusable.

BFD may be encapsulated in a number of transport protocols. An example of this is single-hop BFD [RFC5881]. In that case, the link MTU configuration is typically enough to guarantee communication between the two systems for that size MTU. BFD Echo mode (Section 6.4 of [RFC5880]) is sufficient to permit verification of the Path MTU of such directly connected systems. Previous proposals ([I-D.haas-xiao-bfd-echo-path-mtu]) have been made for testing Path MTU for such directly connected systems. However, in the case of multi-hop BFD [RFC5883], this guarantee does not hold.

The encapsulation of BFD in multi-hop sessions is a simple UDP packet. The BFD elements of procedure (Section 6.8.6 of [RFC5880])

covers validating the BFD payload. However, the specification is silent on the length of the encapsulation that is carrying the BFD PDU. While it is most common that the transport protocol payload (i.e. UDP) length is the exact size of the BFD PDU, this is not required by the elements of procedure. This leads to the possibility that the transport protocol length may be larger than the contained BFD PDU.

2. BFD Encapsulated in Large Packets

Support for BFD between two systems is typically configured, even if the actual session may be dynamically created by a client protocol. A new BFD variable is defined in this document:

`bfd.PaddedPduSize`

The BFD transport protocol payload size is increased to this value. The contents of this additional payload **MUST** be zero. The minimum size of this variable **MUST NOT** be smaller than permitted by the element of BFD procedure; 24 or 26 - see Section 6.8.6 of [RFC5880].

The Don't Fragment bit (Section 2.3 of [RFC0791]) of the IP payload, when using IPv4 encapsulation, **MUST** be set.

3. Implementation and Deployment Considerations

While this document proposes no change to the BFD protocol, implementations may not permit arbitrarily padded transport PDUs to carry BFD packets. While Section 6 of [RFC5880] warns against excessive pedantry, implementations may not work with this mechanism without additional support. Additional changes to the base BFD protocol may be required to permit negotiation of this functionality and the padding value.

It is also worthy of note that even if an implementation can function with larger transport PDUs, that additional packet size may have impact on BFD scaling.

This mechanism also can be applied to other forms of BFD, including S-BFD [RFC7880].

4. Security Considerations

This document does not change the underlying security considerations of the BFD protocol or its encapsulations.

5. IANA Considerations

This document introduces no additional considerations to IANA.

6. References

6.1. Normative References

- [RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <<https://www.rfc-editor.org/info/rfc791>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, DOI 10.17487/RFC5881, June 2010, <<https://www.rfc-editor.org/info/rfc5881>>.
- [RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <<https://www.rfc-editor.org/info/rfc5883>>.
- [RFC7880] Pignataro, C., Ward, D., Akiya, N., Bhatia, M., and S. Pallagatti, "Seamless Bidirectional Forwarding Detection (S-BFD)", RFC 7880, DOI 10.17487/RFC7880, July 2016, <<https://www.rfc-editor.org/info/rfc7880>>.

6.2. Informative References

- [I-D.haas-xiao-bfd-echo-path-mtu] Haas, J. and M. Xiao, "Application of the BFD Echo function for Path MTU Verification or Detection", draft-haas-xiao-bfd-echo-path-mtu-01 (work in progress), July 2011.
- [RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, DOI 10.17487/RFC1191, November 1990, <<https://www.rfc-editor.org/info/rfc1191>>.

Authors' Addresses

Jeffrey Haas
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089
US

Email: jhaas@juniper.net

Albert Fu
Bloomberg

Email: afu14@bloomberg.net

Network Working Group
Internet-Draft
Updates: 5880 (if approved)
Intended status: Standards Track
Expires: 30 September 2022

M. Jethanandani
Kloud Services
S. Agarwal
Cisco Systems, Inc
A. Mishra
O3b Networks
A. Saxena
Ciena Corporation
A. Dekok
Network RADIUS SARL
29 March 2022

Secure BFD Sequence Numbers
draft-ietf-bfd-secure-sequence-numbers-09

Abstract

This document describes two new BFD Authentication mechanism, Meticulous Keyed ISAAC, and Meticulous Keyed FNV1A. These mechanisms can be used to authenticate BFD packets, and secure the sequence number exchange, with less CPU time cost than using MD5 or SHA1, with the tradeoff of decreased security. This document updates RFC 5880.

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 <https://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 30 September 2022.

Copyright Notice

Copyright (c) 2022 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
2. Requirements Language	3
3. Meticulous Keyed ISAAC	3
4. Meticulous Keyed FNV1A	5
5. Operation	6
5.1. Seeding and Operation of ISAAC	7
5.2. Secret Key	8
5.3. Seeding ISAAC	8
6. Meticulous Keyed ISAAC Authentication	9
7. Meticulous Keyed FNV1A Authentication	10
7.1. Calculation of the FNV-1a Digest	12
8. IANA Considerations	12
9. Security Considerations	13
9.1. Spoofing	14
9.2. Re-Use of keys	14
10. Acknowledgements	15
11. References	15
11.1. Normative References	15
11.2. Informative References	15
Authors' Addresses	15

1. Introduction

BFD [RFC5880] defines a number of authentication mechanisms, including Simple Password (Section 6.7.2), and various other methods based on MD5 and SHA1 hashes. The benefit of using cryptographic hashes is that they are secure. The downside to cryptographic hashes is that they are expensive and time consuming on resource-constrained hardware.

When BFD packets are unauthenticated, it is possible for an attacker to forge, modify, and/or replay packets on a link. These attacks have a number of side effects. They can cause parties to believe that a link is down, or they can cause parties to believe that the link is up when it is, in fact, down. The goal of these methods is to prevent spoofing of the BFD session by someone who could guess the next sequence number. We therefore define simple and fast Auth Type methods which allow parties to detect and prevent both spoofed sequence numbers, and spoofed packets.

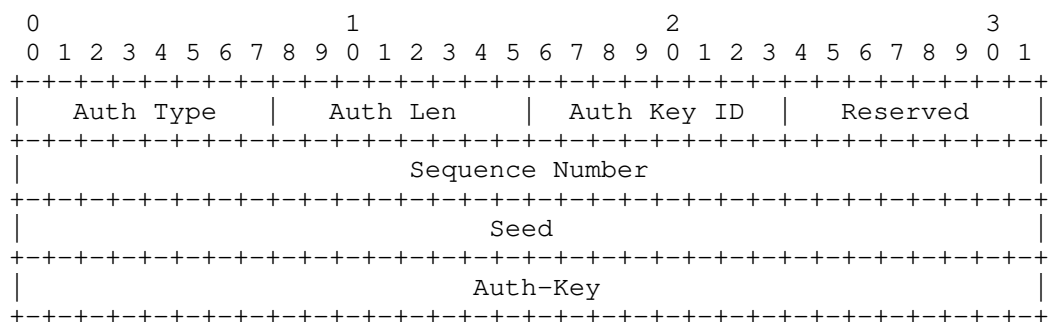
This document proposes the use of Authentication methods which provides meticulous keying, but which have less impact on resource constrained systems. The algorithms chosen are ISAAC [ISAAC], which is a fast cryptographic random number generator, and FNV-1a FNV1A [FNV1A] which is a fast (but non-cryptographic) hash. ISAAC has been subject to significant cryptanalysis in the past thirty years, and has not yet been broken. Similarly, FNV-1a is fast, and while not cryptographically secure, it is has good hashing properties.

2. 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].

3. Meticulous Keyed ISAAC

If the Authentication Present (A) bit is set in the header, and the State (Sta) field equals 3 (Up), and the Authentication Type field contains TB1 (Meticulous Keyed ISAAC), the Authentication Section has the following format:



Auth Type

The Authentication Type, which in this case is TB1 (Meticulous Keyed ISAAC). If the State (Sta) field value is not 3 (Up), then Meticulous Keyed ISAAC MUST NOT be used.

Auth Len

The length of the Authentication Section, in bytes. For Meticulous Keyed ISAAC authentication, the length is 16.

Auth Key ID

The authentication key ID in use for this packet. This allows multiple keys to be active simultaneously.

Reserved

This byte MUST be set to zero on transmit, and ignored on receipt.

Sequence Number

The sequence number for this packet. For Meticulous Keyed ISAAC Authentication, this value is incremented for each successive packet transmitted for a session. This provides protection against replay attacks.

Seed

A 32-bit (4 octet) seed which is used in conjunction with the shared key in order to configure and initialize the ISAAC pseudo-random-number-generator (PRNG). It is used to identify and distinguish "streams" of random numbers which are generated by ISAAC.

Auth-Key

This field carries the 32-bit (4 octet) ISAAC output which is associated with the Sequence Number. The ISAAC PRNG MUST be configured and initialized as given in section TBD.

Note that the Auth-Key here does not include any summary or hash of the packet. The packet itself is completely unauthenticated.

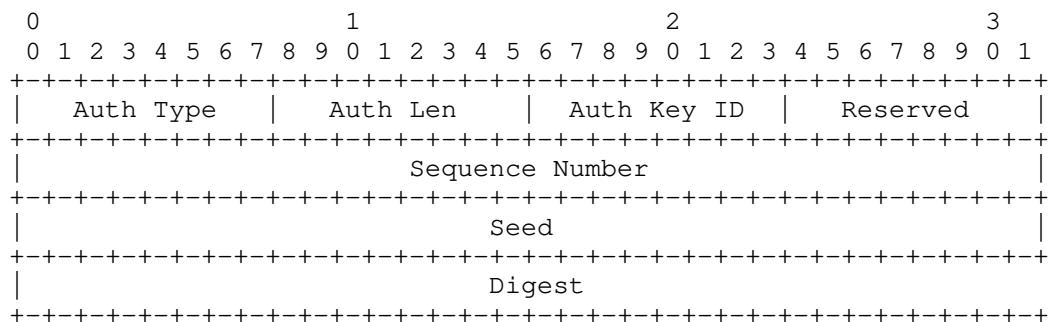
The purpose of this method is to secure the sequence number exchange, and to both detect and prevent spoofing of sequence numbers. In some cases, it is acceptable to not authenticate the entire packet, in which case this method may be used.

When the receiving party receives a BFD packet with an expected sequence number, and the correct corresponding ISAAC output, it knows that only the authentic sending party could have sent that message. The sending party is therefore alive/up, and intended to send the message.

While the rest of the contents of the BFD packet are unauthenticated and may be modified by an attacker, the same is true of stronger Auth Types, such as MD5 or SHA1. The Auth Type methods are not designed to prevent such attacks. Instead, they are designed to prevent an attacker from spoofing identities, and an attacker from artificially keeping a session "Up".

4. Meticulous Keyed FNV1A

If the Authentication Present (A) bit is set in the header, and the State (Sta) field equals 3 (Up), and the Authentication Type field contains TB2 (Meticulous Keyed FNV1A), the Authentication Section has the following format:



Auth Type

The Authentication Type, which in this case is TB2 (Meticulous Keyed FNV1A). If the State (Sta) field value is not 3 (Up), then Meticulous Keyed FNV1A MUST NOT be used.

Auth Len

The length of the Authentication Section, in bytes. For Meticulous Keyed FNV1A authentication, the length is 16.

Auth Key ID

The authentication key ID in use for this packet. This allows multiple keys to be active simultaneously.

Reserved

This byte MUST be set to zero on transmit, and ignored on receipt.

Sequence Number

The sequence number for this packet. For Meticulous Keyed FNV1A Authentication, this value is incremented for each successive packet transmitted for a session. This provides protection against replay attacks.

Seed

A 32-bit (4 octet) seed which is used in conjunction with the shared key in order to configure and initialize the ISAAC PRNG. It is also used to identify and distinguish "streams" of random numbers which are generated by ISAAC.

Digest

This field carries the 32-bit (4 octet) FNV1A digest associated with the Sequence Number. The ISAAC PRNG MUST be configured and initialized as given in section TBD.

Note that the ISAAC PRNG output is still used with this authentication type. The FNV1A hash is fast, but it is not secure. In order to reach an acceptable level of security with FNV1A, we use ISAAC to generate secure per-packet "signing keys". These per-packet keys are then used with FNV1A in order to perform a keyed of hash the packet, and therefore create the Digest.

5. Operation

BFD requires fast and reasonably secure authentication of messages which are exchanged. Methods using MD5 or SHA1 are CPU intensive, and can negatively impact systems with limited CPU power.

We use ISAAC here as a way to generate an infinite stream of pseudo-random numbers. With Meticulous Keyed ISAAC, these numbers are used as a signal that the sending party is authentic. That is, only the sending party can generate the numbers. Therefore if the receiving party sees a correct number, then only the sending party could have generated that number. The sender is therefore authentic, even if the packet contents are not necessarily trusted.

Note that since the packets are not signed with this authentication type, the Meticulous Keyed ISAAC method MUST NOT be used to signal BFD state changes. For BFD state changes, and a more optimized way

to authenticate packets, please refer to BFD Authentication [I-D.ietf-bfd-optimizing-authentication]. Instead, the packets containing Meticulous Keyed ISAAC are only a signal that the sending party is still alive, and that the sending party is authentic. That is, these Auth Type methods must only be used when `bfd.SessionState=Up`, and the State (Sta) field equals 3 (Up).

If slightly more security is desired, the packets can be authenticated via the Meticulous Keyed FNV1A method. This method is similar to the Meticulous Keyed ISAAC authentication type, except that the FNV-1A hash function is used to hash a combination of the packet, and per-packet ISAAC pseudo-random number. If the receiving party is able to validate the hash, then the receiver knows both that the sender is authentic, and that the packet contents have likely not been modified.

As this hash function is not very secure, this method can be used only in situations where the Meticulous Keyed ISAAC method can be used. The Meticulous Keyed FNV1A method MUST NOT be used to signal BFD state changes.

5.1. Seeding and Operation of ISAAC

The ISAAC PRNG state is initialized with the 32-bit Seed, followed by the secret key, and then the rest of the state is filled with zeros. The internal state of ISAAC is 1024 bytes, so the secret key is limited to 1020 bytes in length.

The origin of the Seed field is discussed later in this document. For now, we note that each time a new Seed is used, the `bfd.XmitAuthSeq` value MUST be set to zero.

Once the state has been initialized, the standard ISAAC initial mixing function is run. Once this operation has been performed, ISAAC will be able to produce 256 random numbers at near-zero cost. When all 256 numbers are consumed, the ISAAC mixing function is run, which then results in another set of 256 random numbers

ISAAC can be thought of here as producing an infinite stream of numbers, based on a secret key, where the numbers are produced in "pages" of 256 32-bit values. This property of ISAAC allows for essentially zero-cost "seeking" within a page. The expensive operation of mixing is performed only once per 256 packets, which means that most BFD packet exchanges can be fast and efficient.

The Sequence number is used to "seek" within a the stream of 32-bit numbers produced by ISAAC. The sending party increments the Sequence Number on every packet sent, to indicate to the receiving party where it is in the sequence.

The receiving party can then look at the Sequence Number to determine which particular PRNG value is being used in the packet. The Sequence Number thus permits the two parties to synchronise if/when a packet or packets are lost. Incrementing the Sequence Number for every packet also prevents the re-use of any individual pseudo-random number which was derived from ISAAC.

The Sequence Number can increment without bounds, though it can wrap once it reaches the limit of the 32-bit counter field. ISAAC has a cycle length of 2^{8287} , so there is no issue with using more than 2^{32} values from it.

The result of the above operation is an infinite series of numbers which are unguessable, and which can be used to authenticate the sending party.

5.2. Secret Key

For interoperability, the management interface by which the key is configured MUST accept ASCII strings, and SHOULD also allow for the configuration of any arbitrary binary string in hexadecimal form. Other configuration methods MAY be supported.

The secret Key is mixed with the Seed before being used in ISAAC. If instead ISAAC was initialized without a Seed, then an attacker could pre-compute ISAAC states for many keys, and perform an off-line dictionary attack. The addition of the Seed makes these attacks infeasible.

As a result, it is safe to use the same secret Key for the Auth Types defined here, and also for other Auth Types.

5.3. Seeding ISAAC

The value of the Seed field SHOULD be derived from a secure source. Exactly how this can be done is outside of the scope of this document.

The Seed value SHOULD remain the same for the duration of a BFD session. The Seed value MAY change when the BFD state changes.

If the sending party changes its Seed value, `bfd.XmitAuthSeq` value MUST be set to zero, otherwise the receiving party would be unable to synchronize its sequence of numbers produced by the ISAAC generator. There is no way to signal or negotiate Seed changes. The receiving party MUST remember the current Seed value, and then detect if the Seed changes. Note that the Seed value MUST NOT change unless sending party has signalled a BFD state change with a packet that is authenticated using a more secure Auth Type method.

6. Meticulous Keyed ISAAC Authentication

In this method of authentication, one or more secret keys (with corresponding key IDs) are configured in each system. One of the keys is used to seed the ISAAC PRNG. The output of ISAAC (I) is used to signal that the sender is authentic. To help avoid replay attacks, a sequence number is also carried in each packet. For Meticulous Keyed ISAAC, the sequence number is incremented on every packet.

The receiving system accepts the packet if the key ID matches one of the configured Keys, the Auth-Key derived from the selected Key, Seed, and Sequence Number matches the Auth-Key carried in the packet, and the sequence number is strictly greater than the last sequence number received (modulo wrap at 2^{32})

Transmission Using Meticulous Keyed ISAAC Authentication

The Auth Type field MUST be set to TBD1 (Meticulous Keyed ISAAC). The Auth Len field MUST be set to 16. The Auth Key ID field MUST be set to the ID of the current authentication key. The Sequence Number field MUST be set to `bfd.XmitAuthSeq`.

The Seed field MUST be set to the value of the current seed used for this sequence.

The Auth-Key field MUST be set to the output of ISAAC, which depends on the secret Key, the current Seed, and the Sequence Number.

For Meticulous Keyed ISAAC, `bfd.XmitAuthSeq` MUST be incremented on each packet, in a circular fashion (when treated as an unsigned 32-bit value). The `bfd.XmitAuthSeq` MUST NOT be incremented by more than one for a packet.

Receipt using Meticulous Keyed ISAAC Authentication

If the received BFD Control packet does not contain an Authentication Section, or the Auth Type is not correct (TBD2 for Meticulous Keyed ISAAC), then the received packet MUST be discarded.

If the Auth Key ID field does not match the ID of a configured authentication key, the received packet MUST be discarded.

If the Auth Len field is not equal to 16, the packet MUST be discarded.

If the Seed field does not match the current Seed value, the packet MUST be discarded.

If `bfd.AuthSeqKnown` is 1, examine the Sequence Number field. For Meticulous Keyed FNV1A, if the sequence number lies outside of the range of `bfd.RcvAuthSeq+1` to `bfd.RcvAuthSeq+(3*Detect Mult)` inclusive (when treated as an unsigned 32-bit circular number space) the received packet MUST be discarded.

Calculate the current expected output of ISAAC, which depends on the secret Key, the current Seed, and the Sequence Number. If the value does not matches the Auth-Key field, then the packet MUST be discarded.

Note that in some cases, calculating the expected output of ISAAC will result in the creation of a new "page" of 256 numbers. This process will irreversible, and will destroy the current "page". As a result, if the generation of a new output will create a new "page", the receiving party MUST save a copy of the entire ISAAC state before proceeding with this calculation. If the outputs match, then the saved copy can be discarded, and the new ISAAC state is used. If the outputs do not match, then the saved copy MUST be restored, and the modified copy discarded.

7. Meticulous Keyed FNV1A Authentication

Where slightly more security is needed, the sender can use Meticulous Keyed FNV1A. In this method, each packet is signed with a non-cryptographic hash, FNV-1a [FNV1A]. This hash is reasonably fast, it has good distribution, and collisions are rare. However, it is linear, and potentially reversible. In addition, its output is only 32 bits, and it is not cryptographically strong.

In this methods of authentication, one or more secret keys (with corresponding key IDs) are configured in each system. One of the keys is included in an FNV1A digest calculated over the outgoing BFD Control packet, but the Key itself is not carried in the packet. To

help avoid replay attacks, a sequence number is also carried in each packet. For Meticulous Keyed FNV1A, the sequence number is incremented on every packet.

The receiving system accepts the packet if the key ID matches one of the configured Keys, an FNV-1a digest including the selected key matches the digest carried in the packet, and the sequence number is strictly greater than the last sequence number received (modulo wrap at 2^{32})

Transmission Using Meticulous Keyed FNV1A Authentication

The Auth Type field MUST be set to TBD2 (Meticulous Keyed FNV1A). The Auth Len field MUST be set to 16. The Auth Key ID field MUST be set to the ID of the current authentication key. The Sequence Number field MUST be set to `bfd.XmitAuthSeq`.

The Digest field MUST be set to the value of the FNV-1a digest, as described below.

For Meticulous Keyed FNV1A, `bfd.XmitAuthSeq` MUST be incremented on each packet, in a circular fashion (when treated as an unsigned 32-bit value). The `bfd.XmitAuthSeq` MUST NOT be incremented by more than one for a packet.

Receipt Using Meticulous Keyed FNV1A Authentication

If the received BFD Control packet does not contain an Authentication Section, or the Auth Type is not correct (TBD2 for Meticulous Keyed FNV1A), then the received packet MUST be discarded.

If the Auth Key ID field does not match the ID of a configured authentication key, the received packet MUST be discarded.

If the Auth Len field is not equal to 16, the packet MUST be discarded.

If the Seed field does not match the current Seed value, the packet MUST be discarded.

If `bfd.AuthSeqKnown` is 1, examine the Sequence Number field. For Meticulous Keyed FNV1A, if the sequence number lies outside of the range of `bfd.RcvAuthSeq+1` to `bfd.RcvAuthSeq+(3*Detect Mult)` inclusive (when treated as an unsigned 32-bit circular number space) the received packet MUST be discarded.

Otherwise (bfd.AuthSeqKnown is 0), bfd.AuthSeqKnown MUST be set to 1, and bfd.RcvAuthSeq MUST be set to the value of the received Sequence Number field.

Replace the contents of the Digest field with zeros, and calculate the FNV-1a digest as described below. If the calculated FNV-1a digest is equal to the received value of the Digest field, the received packet MUST be accepted. Otherwise (the digest does not match the Digest field), the received packet MUST be discarded.

7.1. Calculation of the FNV-1a Digest

Unlike other authentication mechanisms, the user-supplied key is not placed into the Auth Key / Digest field, and the packet hashed. As FNV-1a is not a cryptographic hash, such a process would simplify the process for an attacker to "crack" the key.

Instead, for a particular packet "P", and ISAAC pseudo-random number "I", the FNV1A digest "D" is calculated as shown below, where "+" indicates concatenation.

$$D = \text{FNV1A}(I + P + I)$$

Where "+" denotes concatenation. We also note that the Digest field of the packet MUST be initialized to all zeroes before this calculation is performed

The calculated value "D" is then inserted into the packet in the Digest field, and the packet is sent as normal. The receiving party reverses this operation in order to validate the packet.

8. IANA Considerations

This document asks that IANA allocate a new entry in the "BFD Authentication Types" registry.

Address - TBD1

BFD Authentication Type Name - Meticulous Keyed ISAAC

Reference - this document

Address - TBD2

BFD Authentication Type Name - Meticulous Keyed FNV1A

Reference - this document

Note to RFC Editor: this section may be removed on publication as an RFC.

9. Security Considerations

The security of this proposal depends strongly on the length of the secret, and the entropy of the key. It is RECOMMENDED that the key be 16 octets in length or more.

The security of this proposal depends strongly on ISAAC. This generator has been analyzed and has not been broken. Research shows few other CSRNGs which are as simple and as fast as ISAAC. For example, many other generators are based on AES, which is infeasible for resource constrained systems.

The security of this proposal depends on the strength of the FNV-1a hash algorithm. Folding the output of ISAAC into the hash limits the ability of an attacker to reverse the hash, or to perform off-line dictionary attacks. Even if one particular 32-bit per-packet key is found via brute force, that information will be useless, as the next packet will use a different key. And since ISAAC is secure, knowledge of one particular key will give an attacker no ability to predict the next key.

In a keyed algorithm, the key is shared between the two systems. Distribution of this key to all the systems at the same time can be quite a cumbersome task. BFD sessions running a fast rate will require these keys to be refreshed often, which poses a further challenge. Therefore, it is difficult to change the keys during the operation of a BFD session without affecting the stability of the BFD session. Therefore, it is recommended to administratively disable the BFD session before changing the keys.

This method allows the BFD end-points to detect a malicious packet, as the calculated hash value will not match the value found in the packet. The behavior of the session, when such a packet is detected, is based on the implementation. A flood of such malicious packets may cause a BFD session to be operationally down.

As noted earlier with Meticulous Keyed FNV1A, each packet is associated with a unique, per-packet key. This process means that even if an observer sees the Auth-Key, or the FNV-1a hash for one packet, the only information gained will be a key which is never be re-used, and will therefore be useless to an attacker. Further, even if the attacker can "crack" a sequence of packets to obtain a stream of keys, the cryptographic nature of ISAAC makes it impossible for the attacker to derive the input key which is used to "seed" the ISAAC state.

The particular method of hashing was chosen because of the non-cryptographic and reversible nature of the FNV-1a hash. If the digest had been calculated any other way, then an attacker would have significantly less work to do in order to "crack" the hash. In short the per-packet key protects the hash, and the hash protects the per-packet key.

We believe that this construction is reasonably secure, given the constraints. If cryptographic security is desired, then implementors can use MD5 or SHA1 authentication mechanisms

9.1. Spoofing

When Meticulous Keyed ISAAC is used, it is possible for an attacker who can see the packets to observe a particular Auth Key value, and then copy it to a different packet as a "man-in-the-middle" attack. However, the usefulness of such an attack is limited by the requirements that these packets must not signal state changes in the BFD session, and that the key changes on every packet.

Performing such an attack would require an attacker to have the following information and capabilities:

- This is man-in-the-middle active attack.

- The attacker has the contents of a stable packet

- The attacker has managed to deduce the ISAAC key and knows which per-packet key is being used.

The attack is therefore limited to keeping the BFD session up when it would otherwise drop.

However, the usual actual attack which we are protecting BFD from is availability. That is, the attacker is trying to shut down then connection when the attacked parties are trying to keep it up. As a result, the attacks here seem to be irrelevant in practice.

9.2. Re-Use of keys

The strength of the Auth-Type methods is significantly different between the strong one like SHA-1 and ISAAC. While ISAAC has had cryptanalysis, and has not been shown to be broken, that analysis is limited. The question then is whether or not it is safe to use the same key for both Auth Type methods (SHA1 and ISAAC), or should we require different keys for each method?

If we recommend different keys, then it is possible for the two keys to be configured differently on each side of a BFD lin. For example. the strong key can be properly provisioned, which allows to the BFD state machine to advance to Up, Then, when we switch to the weaker Auth Type which uses a different key, that key may not match, and the session will immediatly drop.

We believe that the use of the same key is acceptable, as the Auth Types which use ISAAC also depend on a Seed. The use of the Seed increases the difficulty of breaking the key, and makes off-line dictionary attacks infeasible.

10. Acknowledgements

The authors would like to thank Jeff Haas and Reshad Rahman for their reviews of and suggestions for the document.

11. References

11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.

11.2. Informative References

- [FNV1A] Noll, L. C., "FNV-1a", <http://www.isthe.com/chongo/tech/comp/fnv/index.html#FNV-1a>, 2013.
- [I-D.ietf-bfd-optimizing-authentication] Jethanandani, M., Mishra, A., Saxena, A., and M. Bhatia, "Optimizing BFD Authentication", Work in Progress, Internet-Draft, draft-ietf-bfd-optimizing-authentication-13, 1 August 2021, <<https://www.ietf.org/archive/id/draft-ietf-bfd-optimizing-authentication-13.txt>>.
- [ISAAC] Jenkins, R. J., "ISAAC", <http://www.burtleburtle.net/bob/rand/isaac.html>, 1996.

Authors' Addresses

Mahesh Jethanandani
Kloud Services
Email: mjethanandani@gmail.com

Sonal Agarwal
Cisco Systems, Inc
170 W. Tasman Drive
San Jose, CA 95070
United States of America
Email: agarwaso@cisco.com
URI: www.cisco.com

Ashesh Mishra
O3b Networks
Email: mishra.ashesh@gmail.com

Ankur Saxena
Ciena Corporation
3939 North First Street
San Jose, CA 95134
United States of America
Email: ankurpsaxena@gmail.com

Alan DeKok
Network RADIUS SARL
100 CentrepoinTE Drive #200
Ottawa ON K2G 6B1
Canada
Email: aland@freeradius.org

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 14, 2021

A. Mishra
SES
M. Jethanandani
Kloud Services
A. Saxena
Ciena Corporation
S. Pallagatti
VMware
M. Chen
Huawei
P. Fan
China Mobile
April 12, 2021

BFD Stability
draft-ietf-bfd-stability-10

Abstract

This document describes extensions to the Bidirectional Forwarding Detection (BFD) protocol to measure BFD stability. Specifically, it describes a mechanism for detection of BFD packet loss.

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 <https://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 October 14, 2021.

Copyright Notice

Copyright (c) 2021 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

(<https://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	2
2. Terminology	3
3. Use Cases	3
4. BFD NULL-Authentication Type	3
5. Theory of Operation	3
5.1. Loss Measurement	4
6. ietf-bfd-stability YANG Module	4
6.1. Data Model Overview	4
6.2. YANG Module	5
7. IANA Considerations	9
7.1. The "IETF XML" Registry	9
7.2. The "YANG Module Names" Registry	9
8. Security Consideration	9
9. Contributors	10
10. Acknowledgements	10
11. References	10
11.1. Normative References	10
11.2. Informative References	12
Authors' Addresses	12

1. Introduction

The Bidirectional Forwarding Detection (BFD) [RFC5880] protocol operates by transmitting and receiving BFD control packets, generally at high frequency, over the datapath being monitored. In order to prevent significant data loss due to a datapath failure, BFD session detection time as defined in BFD [RFC5880] is set to the smallest feasible value.

This document proposes a mechanism to detect lost packets in a BFD session in addition to the datapath fault detection mechanisms of BFD. Such a mechanism presents significant value to measure the stability of BFD sessions and provides data to the operators for the cause of a BFD failure.

This document does not propose any BFD extension to measure data traffic loss or delay on a link or tunnel and the scope is limited to BFD packets.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119] and RFC 8174 [RFC8174].

The reader is expected to be familiar with the BFD [RFC5880], Optimizing BFD Authentication [I-D.ietf-bfd-optimizing-authentication] and BFD Secure Sequence Numbers [I-D.ietf-bfd-secure-sequence-numbers].

3. Use Cases

Bidirectional Forwarding Detection as defined in BFD [RFC5880] cannot detect any BFD packet loss if the loss does not last for detection time. This document proposes a method to detect a dropped packet on the receiver. For example, if the receiver receives BFD control packet k at time t but receives packet k+3 at time t+10ms, and never receives packet k+1 and/or k+2, then it has experienced a drop.

This proposal enables BFD implementations to generate diagnostic information on the health of each BFD session that could be used to preempt a failure on a datapath that BFD was monitoring by allowing time for a corrective action to be taken.

In a faulty datapath scenario, an operator can use BFD health information to trigger delay and loss measurement OAM protocol, Connectivity Fault Management (CFM) [IEEE802.1ag] or Loss Measurement (LM)-Delay Measurement (DM)) as defined by A One-way Active Measurement Protocol (OWAMP) [RFC4656] to further isolate the issue.

4. BFD NULL-Authentication Type

The functionality proposed for BFD stability measurement is achieved by appending an authentication section with the NULL Authentication type (as defined in Optimizing BFD Authentication [I-D.ietf-bfd-optimizing-authentication]) to the BFD control packets that do not have authentication enabled.

5. Theory of Operation

This mechanism allows operators to measure the loss of BFD control packets.

When using MD5 or SHA authentication, BFD uses an authentication section that carries the Sequence Number. However, if non-meticulous authentication is being used, or no authentication is in use, then

the non-authenticated BFD control packets MUST include an authentication section with the NULL Authentication type.

5.1. Loss Measurement

Loss measurement counts the number of BFD control packets missed at the receiver during any Detection Time period. The loss is detected by comparing the Sequence Number field in the Auth TLV (NULL or otherwise) in successive BFD control packets. The Sequence Number in each successive control packet generated on a BFD session by the transmitter is incremented by one. This loss count can then be exposed using the YANG module defined in the subsequent section.

The first BFD authentication section with a non-zero sequence number, in a valid BFD control packet, processed by the receiver is used for bootstrapping the logic. When using secure sequence numbers, if the expected values are pre-calculated, the value must be matched to detect lost packets as defined in BFD secure sequence numbers [I-D.ietf-bfd-secure-sequence-numbers].

6. ietf-bfd-stability YANG Module

6.1. Data Model Overview

This YANG module augments the "ietf-bfd" module to add to the per-session set of counters a 'loss-packet-count' for BFD packets that are lost but do not necessarily result in the BFD session going down.

```

module: ietf-bfd-stability
  augment /rt:routing/rt:control-plane-protocols
    /rt:control-plane-protocol/bfd:bfd/bfd-ip-sh:ip-sh
    /bfd-ip-sh:sessions/bfd-ip-sh:session
    /bfd-ip-sh:session-statistics:
    +--ro lost-packet-count?   yang:counter32
  augment /rt:routing/rt:control-plane-protocols
    /rt:control-plane-protocol/bfd:bfd/bfd-ip-mh:ip-mh
    /bfd-ip-mh:session-groups/bfd-ip-mh:session-group
    /bfd-ip-mh:sessions/bfd-ip-mh:session-statistics:
    +--ro lost-packet-count?   yang:counter32
  augment /rt:routing/rt:control-plane-protocols
    /rt:control-plane-protocol/bfd:bfd/bfd-lag:lag
    /bfd-lag:sessions/bfd-lag:session/bfd-lag:member-links
    /bfd-lag:micro-bfd-ipv4/bfd-lag:session-statistics:
    +--ro lost-packet-count?   yang:counter32
  augment /rt:routing/rt:control-plane-protocols
    /rt:control-plane-protocol/bfd:bfd/bfd-lag:lag
    /bfd-lag:sessions/bfd-lag:session/bfd-lag:member-links
    /bfd-lag:micro-bfd-ipv6/bfd-lag:session-statistics:
    +--ro lost-packet-count?   yang:counter32
  augment /rt:routing/rt:control-plane-protocols
    /rt:control-plane-protocol/bfd:bfd/bfd-mpls:mpls
    /bfd-mpls:session-groups/bfd-mpls:session-group
    /bfd-mpls:sessions/bfd-mpls:session-statistics:
    +--ro lost-packet-count?   yang:counter32

```

6.2. YANG Module

This YANG module imports Common YANG Types [RFC6991], A YANG Data Model for Routing [RFC8349], and YANG Data Model for Bidirectional Forwarding Detection (BFD) [I-D.ietf-bfd-yang].

```

<CODE BEGINS> file "ietf-bfd-stability@2021-04-11.yang"
module ietf-bfd-stability {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-bfd-stability";
  prefix "bfds";

  import ietf-yang-types {
    prefix "yang";
    reference
      "RFC 6991: Common YANG Data Types";
  }

  import ietf-routing {
    prefix "rt";
    reference

```

```
    "RFC 8349: A YANG Data Model for Routing Management
      (NMDA version)";
  }

  import ietf-bfd {
    prefix bfd;
    reference
      "I-D.ietf-bfd-yang: YANG Data Model for Bidirectional
        Forwarding Detection.";
  }

  import ietf-bfd-ip-sh {
    prefix bfd-ip-sh;
    reference
      "I-D.ietf-bfd-yang: YANG Data Model for Bidirectional
        Forwarding Detection.";
  }

  import ietf-bfd-ip-mh {
    prefix bfd-ip-mh;
    reference
      "I-D.ietf-bfd-yang: YANG Data Model for Bidirectional
        Forwarding Detection.";
  }

  import ietf-bfd-lag {
    prefix bfd-lag;
    reference
      "I-D.ietf-bfd-yang: YANG Data Model for Bidirectional
        Forwarding Detection.";
  }

  import ietf-bfd-mpls {
    prefix bfd-mpls;
    reference
      "I-D.ietf-bfd-yang: YANG Data Model for Bidirectional
        Forwarding Detection.";
  }

  organization
    "IETF BFD Working Group";

  contact
    "WG Web:  <http://tools.ietf.org/wg/bfd>
    WG List:  <bfd@ietf.org>

    Authors: Mahesh Jethanandani (mjethanandani@gmail.com)
             Ashesh Mishra (mishra.ashesh@gmail.com)
```

Ankur Saxena (ankurpsaxena@gmail.com)
Santosh Pallagatti (santosh.pallagatti@gmail.com)
Mach Chen (mach.chen@huawei.com)
Peng Fan (fanp08@gmail.com).";

description

"This YANG module augments the base BFD YANG model to add attributes related to BFD Stability. In particular it adds a per session count for BFD packets that are lost.

Copyright (c) 2021 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>).

This version of this YANG module is part of RFC XXXX (<https://www.rfc-editor.org/info/rfcXXXX>); see the RFC itself for full legal notices.

The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALL NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'NOT RECOMMENDED', 'MAY', and 'OPTIONAL' in this document are to be interpreted as described in BCP 14 (RFC 2119) (RFC 8174) when, and only when, they appear in all capitals, as shown here.";

```
revision "2021-04-11" {  
  description  
    "Initial Version.";  
  reference  
    "RFC XXXX, BFD Stability.";  
}
```

```
augment "/rt:routing/rt:control-plane-protocols/" +  
  "rt:control-plane-protocol/bfd:bfd/bfd-ip-sh:ip-sh/" +  
  "bfd-ip-sh:sessions/bfd-ip-sh:session/" +  
  "bfd-ip-sh:session-statistics" {  
  leaf lost-packet-count {  
    type yang:counter32;  
    description  
      "Number of BFD packets that were lost without bringing the  
      session down.";  
  }  
}
```

```
    description
      "Augment the 'bfd' container to add attributes related to BFD
      stability.";
  }

  augment "/rt:routing/rt:control-plane-protocols/" +
    "rt:control-plane-protocol/bfd:bfd/bfd-ip-mh:ip-mh/" +
    "bfd-ip-mh:session-groups/bfd-ip-mh:session-group/" +
    "bfd-ip-mh:sessions/bfd-ip-mh:session-statistics" {
    leaf lost-packet-count {
      type yang:counter32;
      description
        "Number of BFD packets that were lost without bringing the
        session down.";
    }
    description
      "Augment the 'bfd' container to add attributes related to BFD
      stability.";
  }

  augment "/rt:routing/rt:control-plane-protocols/" +
    "rt:control-plane-protocol/bfd:bfd/bfd-lag:lag/" +
    "bfd-lag:sessions/bfd-lag:session/bfd-lag:member-links/" +
    "bfd-lag:micro-bfd-ipv4/bfd-lag:session-statistics" {
    leaf lost-packet-count {
      type yang:counter32;
      description
        "Number of BFD packets that were lost without bringing the
        session down.";
    }
    description
      "Augment the 'bfd' container to add attributes related to BFD
      stability.";
  }

  augment "/rt:routing/rt:control-plane-protocols/" +
    "rt:control-plane-protocol/bfd:bfd/bfd-lag:lag/" +
    "bfd-lag:sessions/bfd-lag:session/bfd-lag:member-links/" +
    "bfd-lag:micro-bfd-ipv6/bfd-lag:session-statistics" {
    leaf lost-packet-count {
      type yang:counter32;
      description
        "Number of BFD packets that were lost without bringing the
        session down.";
    }
    description
      "Augment the 'bfd' container to add attributes related to BFD
      stability.";
```

```
    }

    augment "/rt:routing/rt:control-plane-protocols/" +
        "rt:control-plane-protocol/bfd:bfd/bfd-mpls:mpls/" +
        "bfd-mpls:session-groups/bfd-mpls:session-group/" +
        "bfd-mpls:sessions/bfd-mpls:session-statistics" {
        leaf lost-packet-count {
            type yang:counter32;
            description
                "Number of BFD packets that were lost without bringing the
                 session down.";
        }
        description
            "Augment the 'bfd' container to add attributes related to BFD
             stability.";
    }
}
<CODE ENDS>
```

7. IANA Considerations

7.1. The "IETF XML" Registry

This document registers one URIs in the "ns" subregistry of the "IETF XML" registry [RFC3688]. Following the format in [RFC3688], the following registration is requested:

URI: urn:ietf:params:xml:ns:yang:ietf-bfd-stability
Registrant Contact: The IESG
XML: N/A, the requested URI is an XML namespace.

7.2. The "YANG Module Names" Registry

This document registers one YANG module in the "YANG Module Names" registry YANG [RFC6020]. Following the format in YANG [RFC6020], the following registrations are requested:

name: ietf-bfd-stability
namespace: urn:ietf:params:xml:ns:yang:ietf-bfd-stability
prefix: bfds
reference: RFC XXXX

8. Security Consideration

The YANG module specified in this document defines a schema for data that is designed to be accessed via network management protocols such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure

transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446]. The NETCONF Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

The YANG module does not define any writeable/creatable/deletable data nodes.

The only readable data nodes in YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control read access (e.g., via get, get-config, or notification) to these data nodes. The model does not define any readable subtrees and data nodes.

The YANG module does not define any RPC operations.

9. Contributors

Manav Bhatia

10. Acknowledgements

Authors would like to thank Nobo Akiya, Jeffery Haas, Dileep Singh, Basil Saji, Sagar Soni, Albert Fu and Mallik Mudigonda who also contributed to this document.

11. References

11.1. Normative References

[I-D.ietf-bfd-optimizing-authentication]

Jethanandani, M., Mishra, A., Saxena, A., and M. Bhatia, "Optimizing BFD Authentication", draft-ietf-bfd-optimizing-authentication-11 (work in progress), July 2020.

[I-D.ietf-bfd-secure-sequence-numbers]

Jethanandani, M., Agarwal, S., Mishra, A., Saxena, A., and A. DeKok, "Secure BFD Sequence Numbers", draft-ietf-bfd-secure-sequence-numbers-07 (work in progress), December 2020.

- [I-D.ietf-bfd-yang]
Rahman, R., Zheng, L., Jethanandani, M., Pallagatti, S.,
and G. Mirsky, "YANG Data Model for Bidirectional
Forwarding Detection (BFD)", draft-ietf-bfd-yang-17 (work
in progress), August 2018.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119,
DOI 10.17487/RFC2119, March 1997,
<<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688,
DOI 10.17487/RFC3688, January 2004,
<<https://www.rfc-editor.org/info/rfc3688>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection
(BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010,
<<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for
the Network Configuration Protocol (NETCONF)", RFC 6020,
DOI 10.17487/RFC6020, October 2010,
<<https://www.rfc-editor.org/info/rfc6020>>.
- [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed.,
and A. Bierman, Ed., "Network Configuration Protocol
(NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011,
<<https://www.rfc-editor.org/info/rfc6241>>.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure
Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011,
<<https://www.rfc-editor.org/info/rfc6242>>.
- [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types",
RFC 6991, DOI 10.17487/RFC6991, July 2013,
<<https://www.rfc-editor.org/info/rfc6991>>.
- [RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF
Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017,
<<https://www.rfc-editor.org/info/rfc8040>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

- [RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <<https://www.rfc-editor.org/info/rfc8341>>.
- [RFC8349] Lhotka, L., Lindem, A., and Y. Qu, "A YANG Data Model for Routing Management (NMDA Version)", RFC 8349, DOI 10.17487/RFC8349, March 2018, <<https://www.rfc-editor.org/info/rfc8349>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.

11.2. Informative References

- [IEEE802.1ag] Institute of Electrical and Electronics Engineers, Inc., "802.1ag - Connectivity Fault Management", September 2007, <<https://www.ieee802.org/1/pages/802.1ag.html>>.
- [RFC4656] Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, DOI 10.17487/RFC4656, September 2006, <<https://www.rfc-editor.org/info/rfc4656>>.

Authors' Addresses

Ashesh Mishra
SES

Email: mishra.ashesh@gmail.com

Mahesh Jethanandani
Kloud Services
CA
USA

Email: mjethanandani@gmail.com

Ankur Saxena
Ciena Corporation
3939 North 1st Street
San Jose, CA 95134
USA

Email: ankurpsaxena@gmail.com
URI: www.ciena.com

Santosh Pallagatti
VMware
Bangalore, Karnataka 560103
India

Email: santosh.pallagatti@gmail.com

Mach Chen
Huawei

Email: mach.chen@huawei.com

Peng Fan
China Mobile
32 Xuanwumen West Street
Beijing, Beijing
China

Email: fanp08@gmail.com

BFD
Internet-Draft
Intended status: Informational
Expires: April 29, 2021

S. Pallagatti, Ed.
VMware
G. Mirsky, Ed.
ZTE Corp.
S. Paragiri
Individual Contributor
V. Govindan
M. Mudigonda
Cisco
October 26, 2020

BFD for VXLAN
draft-ietf-bfd-vxlan-16

Abstract

This document describes the use of the Bidirectional Forwarding Detection (BFD) protocol in point-to-point Virtual eXtensible Local Area Network (VXLAN) tunnels used to form an overlay network.

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 <https://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 29, 2021.

Copyright Notice

Copyright (c) 2020 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 (<https://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	2
2. Conventions Used in this Document	3
2.1. Acronyms	3
2.2. Requirements Language	4
3. Deployment	4
4. Use of the Management VNI	5
5. BFD Packet Transmission over VXLAN Tunnel	6
6. Reception of BFD Packet from VXLAN Tunnel	8
7. Echo BFD	8
8. IANA Considerations	8
9. Security Considerations	9
10. Contributors	9
11. Acknowledgments	9
12. References	10
12.1. Normative References	10
12.2. Informational References	10
Authors' Addresses	11

1. Introduction

"Virtual eXtensible Local Area Network" (VXLAN) [RFC7348] provides an encapsulation scheme that allows building an overlay network by decoupling the address space of the attached virtual hosts from that of the network.

One use of VXLAN is in data centers interconnecting virtual machines (VMs) of a tenant. VXLAN addresses requirements of the Layer 2 and Layer 3 data center network infrastructure in the presence of VMs in a multi-tenant environment by providing a Layer 2 overlay scheme on a Layer 3 network [RFC7348]. Another use is as an encapsulation for Ethernet VPN [RFC8365].

This document is written assuming the use of VXLAN for virtualized hosts and refers to VMs and VXLAN Tunnel End Points (VTEPs) in hypervisors. However, the concepts are equally applicable to non-virtualized hosts attached to VTEPs in switches.

In the absence of a router in the overlay, a VM can communicate with another VM only if they are on the same VXLAN segment. VMs are unaware of VXLAN tunnels as a VXLAN tunnel is terminated on a VTEP.

VTEPs are responsible for encapsulating and decapsulating frames exchanged among VMs.

The ability to monitor path continuity, i.e., perform proactive continuity check (CC) for point-to-point (p2p) VXLAN tunnels, is important. The asynchronous mode of BFD, as defined in [RFC5880], is used to monitor a p2p VXLAN tunnel.

In the case where a Multicast Service Node (MSN) (as described in Section 3.3 of [RFC8293]) participates in VXLAN, the mechanisms described in this document apply and can, therefore, be used to test the continuity of the path between the source NVE and the MSN.

This document describes the use of Bidirectional Forwarding Detection (BFD) protocol to enable monitoring continuity of the path between VXLAN VTEPs that are performing as Network Virtualization Endpoints, and/or between the source NVE and a replicator MSN using a Management VNI (Section 4). All other uses of the specification to test toward other VXLAN endpoints are out of the scope.

2. Conventions Used in this Document

2.1. Acronyms

BFD Bidirectional Forwarding Detection

CC Continuity Check

p2p Point-to-point

MSN Multicast Service Node

NVE Network Virtualization Endpoint

VFI Virtual Forwarding Instance

VM Virtual Machine

VNI VXLAN Network Identifier (or VXLAN Segment ID)

VTEP VXLAN Tunnel End Point

VXLAN Virtual eXtensible Local Area Network

2.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Deployment

Figure 1 illustrates the scenario with two servers, each of them hosting two VMs. The servers host VTEPs that terminate two VXLAN tunnels with VXLAN Network Identifier (VNI) number 100 and 200 respectively. Separate BFD sessions can be established between the VTEPs (IP1 and IP2) for monitoring each of the VXLAN tunnels (VNI 100 and 200). Using a BFD session to monitor a set of VXLAN VNIs between the same pair of VTEPs might help to detect and localize problems caused by misconfiguration. An implementation that supports this specification MUST be able to control the number of BFD sessions that can be created between the same pair of VTEPs. This method is applicable whether the VTEP is a virtual or physical device.

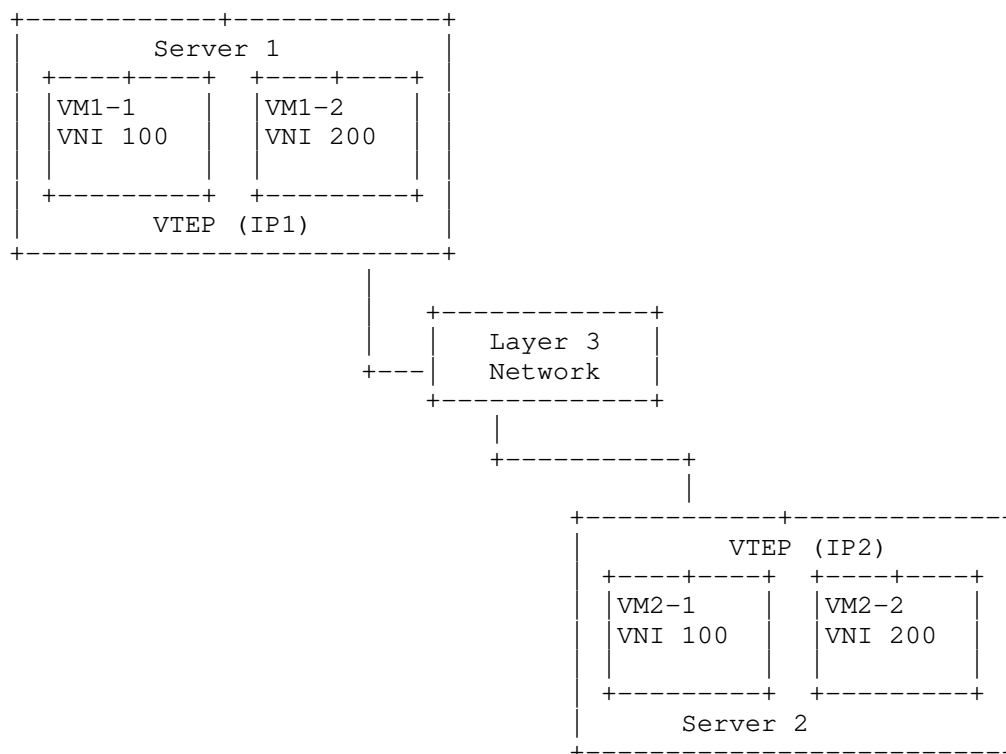


Figure 1: Reference VXLAN Domain

At the same time, a service layer BFD session may be used between the tenants of VTEPs IP1 and IP2 to provide end-to-end fault management (this use case is outside the scope of this document). In such a case, for VTEPs, the BFD Control packets of that session are indistinguishable from data packets.

For BFD Control packets encapsulated in VXLAN (Figure 2), the inner destination IP address SHOULD be set to one of the loopback addresses from 127/8 range for IPv4 or to one of IPv4-mapped IPv6 loopback addresses from `::ffff:127.0.0.0/104` range for IPv6.

4. Use of the Management VNI

In most cases, a single BFD session is sufficient for the given VTEP to monitor the reachability of a remote VTEP, regardless of the number of VNIs. BFD control messages MUST be sent using the Management VNI which acts as the as control and management channel between VTEPs. An implementation MAY support operating BFD on

another (non-Management) VNI although the implications of this are outside the scope of this document. The selection of the VNI number of the Management VNI MUST be controlled through a management plane. An implementation MAY use VNI number 1 as the default value for the Management VNI. All VXLAN packets received on the Management VNI MUST be processed locally and MUST NOT be forwarded to a tenant.

5. BFD Packet Transmission over VXLAN Tunnel

BFD packets MUST be encapsulated and sent to a remote VTEP as explained in this section. Implementations SHOULD ensure that the BFD packets follow the same forwarding path as VXLAN data packets within the sender system.

BFD packets are encapsulated in VXLAN as described below. The VXLAN packet format is defined in Section 5 of [RFC7348]. The value in the VNI field of the VXLAN header MUST be set to the value selected as the Management VNI. The Outer IP/UDP and VXLAN headers MUST be encoded by the sender as defined in [RFC7348].

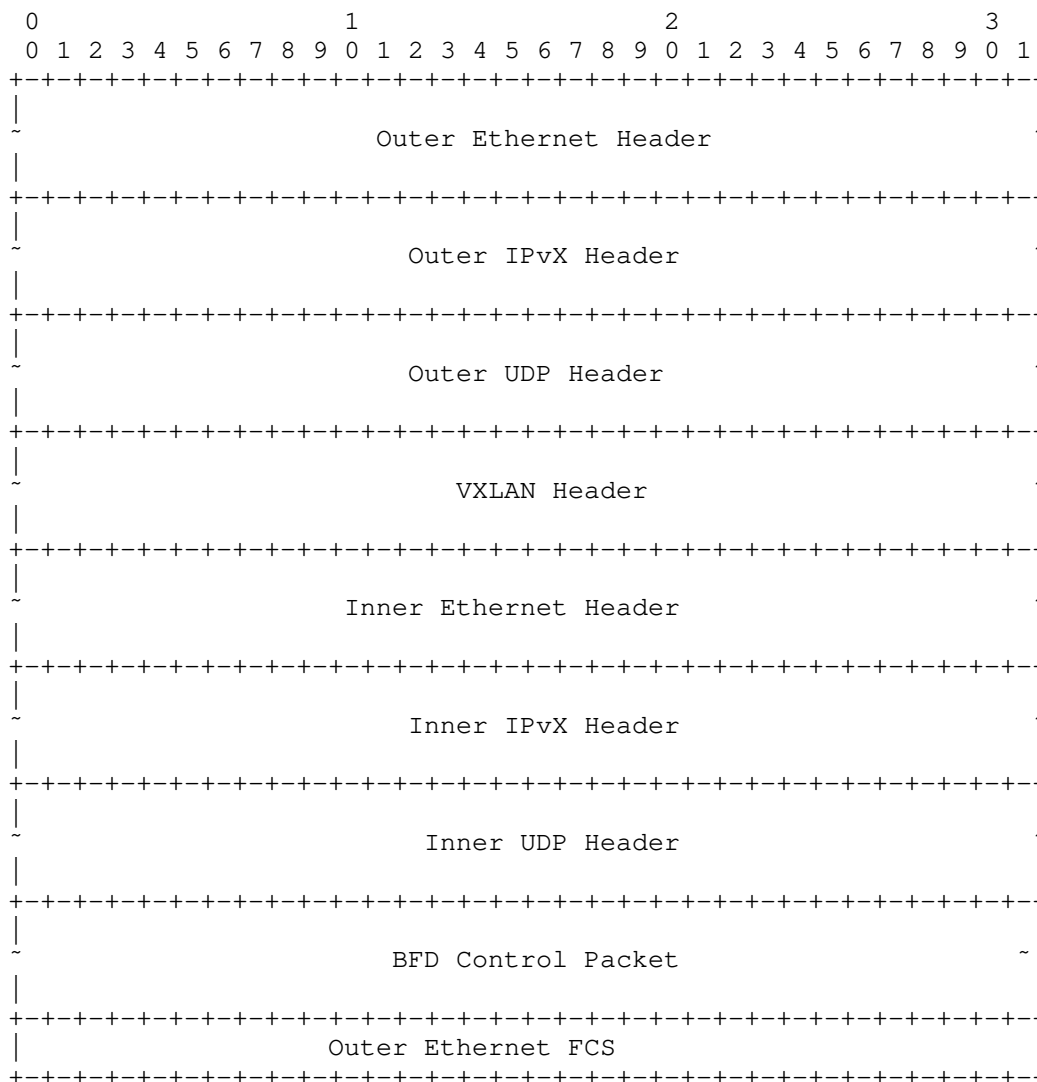


Figure 2: VXLAN Encapsulation of BFD Control Packet

The BFD packet MUST be carried inside the inner Ethernet frame of the VXLAN packet. The choice of Destination MAC and Destination IP addresses for the inner Ethernet frame MUST ensure that the BFD Control packet is not forwarded to a tenant but is processed locally at the remote VTEP. The inner Ethernet frame carrying the BFD Control packet- has the following format:

Ethernet Header:

Destination MAC: A Management VNI, which does not have any tenants, will have no dedicated MAC address for decapsulated traffic. The value (TBD1) SHOULD be used in this field.

Source MAC: MAC address associated with the originating VTEP.

Ethertype: is set to 0x0800 if the inner IP header is IPv4, and is set to 0x86DD if the inner IP header is IPv6.

IP header:

Destination IP: IP address MUST NOT be of one of tenant's IP addresses. The IP address SHOULD be selected from the range 127/8 for IPv4, for IPv6 - from the range ::ffff:127.0.0.0/104. Alternatively, the destination IP address MAY be set to VTEP's IP address.

Source IP: IP address of the originating VTEP.

TTL or Hop Limit: MUST be set to 255 in accordance with [RFC5881].

The fields of the UDP header and the BFD Control packet are encoded as specified in [RFC5881].

6. Reception of BFD Packet from VXLAN Tunnel

Once a packet is received, the VTEP MUST validate the packet. If the packet is received on the management VNI and is identified as BFD control packet addressed to the VTEP, and then the packet can be processed further. Processing of BFD control packets received on non-management VNI is outside the scope of this specification.

The received packet's inner IP payload is then validated according to Sections 4 and 5 in [RFC5881].

7. Echo BFD

Support for echo BFD is outside the scope of this document.

8. IANA Considerations

IANA is requested to assign a single MAC address to the value TBD1 from the "IANA Unicast 48-bit MAC Address" registry from the "Unassigned (small allocations)" block. The Usage field will be "BFD for VXLAN" with a Reference field of this document.

9. Security Considerations

Security issues discussed in [RFC5880], [RFC5881], and [RFC7348] apply to this document.

This document recommends using an address from the Internal host loopback addresses 127/8 range for IPv4 or an IP4-mapped IPv6 loopback address from ::ffff:127.0.0.0/104 range for IPv6 as the destination IP address in the inner IP header. Using such an address prevents the forwarding of the encapsulated BFD control message by a transient node in case the VXLAN tunnel is broken as according to [RFC1812].

A router SHOULD NOT forward, except over a loopback interface, any packet that has a destination address on network 127. A router MAY have a switch that allows the network manager to disable these checks. If such a switch is provided, it MUST default to performing the checks.

The use of IPv4-mapped IPv6 addresses has the same property as using the IPv4 network 127/8, moreover, the IPv4-mapped IPv6 addresses prefix is not advertised in any routing protocol.

If the implementation supports establishing multiple BFD sessions between the same pair of VTEPs, there SHOULD be a mechanism to control the maximum number of such sessions that can be active at the same time.

10. Contributors

Reshad Rahman
rrahman@cisco.com
Cisco

11. Acknowledgments

Authors would like to thank Jeff Haas of Juniper Networks for his reviews and feedback on this material.

Authors would also like to thank Nobo Akiya, Marc Binderberger, Shahram Davari, Donald E. Eastlake 3rd, Anoop Ghanwani, Dinesh Dutt, Joel Halpern, and Carlos Pignataro for the extensive reviews and the most detailed and constructive comments.

12. References

12.1. Normative References

- [RFC1812] Baker, F., Ed., "Requirements for IP Version 4 Routers", RFC 1812, DOI 10.17487/RFC1812, June 1995, <<https://www.rfc-editor.org/info/rfc1812>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, DOI 10.17487/RFC5881, June 2010, <<https://www.rfc-editor.org/info/rfc5881>>.
- [RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <<https://www.rfc-editor.org/info/rfc7348>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

12.2. Informational References

- [RFC8293] Ghanwani, A., Dunbar, L., McBride, M., Bannai, V., and R. Krishnan, "A Framework for Multicast in Network Virtualization over Layer 3", RFC 8293, DOI 10.17487/RFC8293, January 2018, <<https://www.rfc-editor.org/info/rfc8293>>.
- [RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <<https://www.rfc-editor.org/info/rfc8365>>.

Authors' Addresses

Santosh Pallagatti (editor)
VMware

Email: santosh.pallagatti@gmail.com

Greg Mirsky (editor)
ZTE Corp.

Email: gregimirsky@gmail.com

Sudarsan Paragiri
Individual Contributor

Email: sudarsan.225@gmail.com

Vengada Prasad Govindan
Cisco

Email: venggovi@cisco.com

Mallik Mudigonda
Cisco

Email: mmudigon@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 25, 2021

R. Bush
Internet Initiative Japan
J. Haas
J. Scudder
Juniper Networks, Inc.
A. Nipper
C. Dietzel
DE-CIX
September 21, 2020

Making Route Servers Aware of Data Link Failures at IXPs
draft-ietf-idr-rs-bfd-09

Abstract

When BGP route servers are used, the data plane is not congruent with the control plane. Therefore, peers at an Internet exchange can lose data connectivity without the control plane being aware of it, and packets are lost. This document proposes the use of a newly defined BGP Subsequent Address Family Identifier (SAFI) both to allow the route server to request its clients use BFD to track data plane connectivity to their peers' addresses, and for the clients to signal that connectivity state back to the route server.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC2119] only when they appear in all upper case. They may also appear in lower or mixed case as English words, without normative meaning.

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 <https://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 March 25, 2021.

Copyright Notice

Copyright (c) 2020 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 (<https://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	2
2. Definitions	3
3. Overview	4
4. Next Hop Validation	5
4.1. ReachAsk	6
4.2. LocReach	6
4.3. ReachTell	7
4.4. NHIB	7
5. Advertising NH-Reach state in BGP	7
6. Client Procedures for NH-Reach Changes	9
7. Recommendations for Using BFD	9
8. Other Considerations	10
9. Acknowledgments	10
10. IANA Considerations	10
11. Security Considerations	10
12. References	11
12.1. Normative References	11
12.2. Informative References	12
Appendix A. Summary of Document Changes	12
Appendix B. Other Forms of Connectivity Checks	12
Authors' Addresses	13

1. Introduction

In configurations (typically Internet Exchange Points (IXPs)) where EBGp routing information is exchanged between client routers through the agency of a route server (RS) [RFC7947], but traffic is exchanged directly, operational issues can arise when partial data plane connectivity exists among the route server client routers. Since the

data plane is not congruent with the control plane, the client routers on the IXP can lose data connectivity without the control plane - the route server - being aware of it, resulting in significant data loss.

To remedy this, two basic problems need to be solved:

1. Client routers must have a means of verifying connectivity amongst themselves, and
2. Client routers must have a means of communicating the knowledge of the failure (and restoration) back to the route server.

The first can be solved by application of Bidirectional Forwarding Detection [RFC5880]. The second can be solved by exchanging BGP routes which use the NH-Reach Subsequent Address Family Identifier (SAFI) defined in this document.

Throughout this document, we generally assume that the route server being discussed is able to represent different RIBs towards different clients, as discussed in section 2.3.2.1 of [RFC7947]. If this is not the case, the procedures described here to allow BFD to be automatically provisioned between clients still have value; however, the procedures for signaling reachability back to the route server may not.

Throughout this document, we refer to the "route server", "RS" or just "server" and the "client" to describe the two BGP routers engaging in the exchange of information. We observe that there could be other applications for this extension. Our use of terminology is intended for clarity of description, and not to limit the future applicability of the proposal.

[I-D.ietf-idr-bgp-bestpath-selection-criteria] discusses enhancement of the route resolvability condition of section 9.1.2.1 of [RFC4271] to include next hop reachability and path availability checks. This specification represents in part an instance of such, implemented using BFD as the OAM mechanism.

2. Definitions

- o Indirect peer: If a route server is configured such that routes from a given client might be sent to some other client, or vice-versa, those two clients are considered to be indirect peers.
- o Indirect Peer's Address, IPA, next hop: We refer frequently to a next hop. It should generally be clear from context what is intended, almost always an address associated with an indirect peer (the exception, when an indirect peer sends a third party next hop, is discussed in Section 3). In Section 5 we discuss the

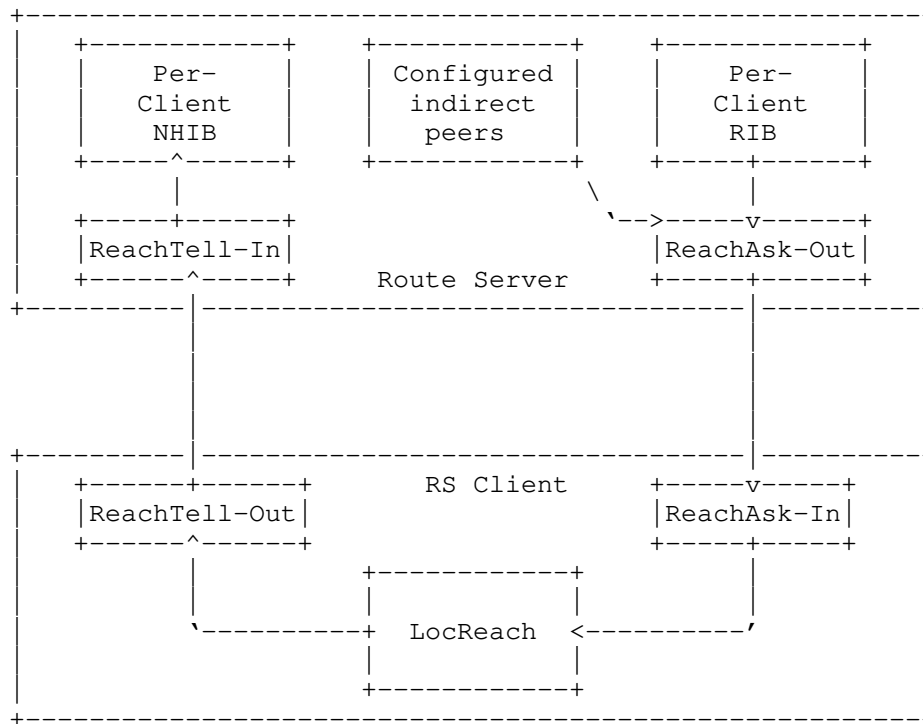
MP-BGP [RFC4760] Next Hop field; this is distinguished by its capitalization and should also be clear from context. Later in that section we define the Indirect Peer's Address field of the NLRI, also called "IPA". It will be clear to the reader that this refers to the "next hops" discussed elsewhere in the document, but we don't use the name "next hop" for this field to avoid confusion with the pre-existing next hop path attribute of [RFC4271] and attribute field of [RFC4760].

- o RS: Route Server. See [RFC7947].

3. Overview

As with the base BGP protocol, we model the function of this extension as the interaction between a conceptual set of databases:

- o ReachAsk: The reachability request database. A database of next hops (host addresses) for which data plane reachability is being queried.
- o ReachAsk-Out: A set of queries sent to the client.
- o ReachAsk-In: A set of queries received from the route server.
- o ReachTell: The reachability response database. A database of responses to ReachAsk queries, indicating what is known about data plane reachability.
- o ReachTell-Out: The responses being sent to the route server.
- o ReachTell-In: The response received from the client.
- o LocReach: The local reachability database.
- o NHIB: Next Hop Information Base. Stores what is known about the client's reachability to its next hops.



Route Server, RS Client, and Reachability Ask and Tell databases with In/Out Queues

In outline, the route server requests its client to track connectivity for all the potential next hops the RS might send to the client, by sending these next hops as ReachAsk "routes". The client tracks connectivity using BFD and reports its connectivity status to the RS using ReachTell "routes". Connectivity status may be that the next hop is reachable, unreachable, or unknown. Once the RS has been informed by the client of its connectivity, it uses this information to influence the route selection the RS performs on behalf of the client. Details are elaborated in the following sections.

4. Next Hop Validation

Below, we detail procedures where a route server tells its client router about other client next hops by sending it ReachAsk routes and the client router verifies connectivity to those other client routers and communicates its findings back to the RS using ReachTell routes. The RS uses the received ReachTell routes as input to the NHIB and hence the route selection process it performs on behalf of the client.

4.1. ReachAsk

The route server maintains a ReachAsk database for each client that supports this proposal, that is, for each client that has advertised support (Section 5) for the NH-Reach SAFI. This database is the union of:

- o The set of next hops found in the associated per-client Loc-RIB (see section 2.3.2.1 of [RFC7947]).
- o The set of addresses of this client's indirect peers (Section 2).
- o The RS MAY also add other entries, for example under configuration control.

We note that under most circumstances, the first (Loc-RIB next hops) set will be a subset of the second (indirect peers) set. For this not to be the case, a client would have to have sent a "third party" next hop [RFC4271] to the server. To cover such a case, an implementation MAY note any such next hops, and include them in its list of indirect peers. (This implies that if a third party next hop for client C is conveyed to client A, not only will C be placed in A's ReachAsk database, but A will be placed in C's ReachAsk database.)

The contents of the ReachAsk database are communicated to the client using the NLRI format and procedures described in Section 5.

4.2. LocReach

The client MUST attempt to track data plane connectivity to each host address depicted in the ReachAsk database. It MAY also track connectivity to other addresses. The use of BFD for this purpose is detailed in Section 6.

For each address being tracked, its state is maintained by the client in a LocReach entry. The state can be:

- o Unknown. Connectivity status is unknown. This may be due to a temporary or permanent lack of feasible OAM mechanism to determine the status.
- o Up. The address has been determined to be reachable.
- o Down. The address has been determined to be unreachable.

The LocReach database is used as input for the ReachTell database; it MAY also be used as input to the client's route resolvability condition (section 9.1.2.1 of [RFC4271]).

4.3. ReachTell

The ReachTell database contains an entry for every entry in the LocReach database.

The contents of the ReachTell database are communicated to the server using the NLRI format and procedures described in Section 5.

4.4. NHIB

The route server maintains a per-client Next Hop Information Base, or NHIB. This contains the information about next hop status received from ReachTell.

In computing its per-client Loc-RIB, the RS uses the content of the related per-client NHIB as input to the route resolvability condition (section 9.1.2.1 of [RFC4271]). The next hop being resolved is looked up in the NHIB and its state determined:

- o Up next hops are considered resolvable.
- o Unknown next hops MAY be considered resolvable. They MAY be less preferred for selection.
- o Down next hops MUST NOT be considered resolvable.
- o If a given next hop is not present in the NHIB, but is present in ReachAsk-Out, either the client has not responded yet (a transient condition) or an error exists. Similar to Unknown next hops, such routes MAY be considered resolvable; they MAY be less preferred.

5. Advertising NH-Reach state in BGP

A new BGP SAFI, the NH-Reach SAFI, is defined in this document. It has been assigned value TBD. A route server or a route server client using the procedures in this document MUST advertise support for this SAFI, for the IPv4 and/or IPv6 Address Family Identifier (AFI). The use of this SAFI with any other AFI is not defined by this document.

NH-Reach NLRI "routes" have a Length of Next Hop Network Address value of 0, therefore they have an empty Network Address of Next Hop field (section 3 of [RFC4760]).

Since as specified here, ReachTell "routes" from different clients populate distinct databases on the RS, there will generally be only a single path per "route"; this implies that route selection need not be performed (or equivalently, that it's trivial to perform).

In the other direction, a client might peer with multiple route servers and receive differing sets of ReachAsk routes from them. An implementation MAY handle this situation by implementing a distinct

ReachAsk and ReachTell per server, but it MAY also handle it by placing all servers' ReachAsk "routes" into a single ReachAsk, and sending the results to all servers from a single ReachTell. This would imply some route server(s) might get ReachTell results they had not asked for, but this is permissible in any case. Again, since the contents of ReachAsk are simply a set of host routes to be tested, route selection over a combined ReachAsk MAY be omitted.

ReachAsk and ReachTell entries are exchanged using the NH-Reach NLRI encoding:

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|T|Reserved|Sta|Indirect Peer's Address (4 or 16 octets)|
+-----+-----+-----+-----+-----+-----+-----+-----+
.      ... Indirect Peer's Address (4 or 16 octets) ...      .
.
+-----+-----+-----+-----+-----+-----+-----+-----+

```

NH-Reach NLRI Format

- o T: Type is a one-bit field that can take the value 0, meaning the NLRI is a ReachAsk entry, or 1, meaning it is a ReachTell entry.
- o Reserved: These five bits are reserved. They MUST be sent as zero and MUST be disregarded on receipt.
- o Sta: State is a two-bit field used to signal the LocReach (Section 4.2) state:
 - * 0 or 3: Unknown.
 - * 1: Up.
 - * 2: Down.

Although either 0 or 3 is to be interpreted as "Unknown", the value 0 MUST be used on transmission. The value 3 MUST be accepted as an alias for 0 on receipt.

- o The Indirect Peer's Address ("IPA") field is an IPv4 or IPv6 host route, depending on whether the AFI is IPv4 or IPv6.

ReachAsk and ReachTell entries MUST NOT be propagated from one BGP peering session to another; the routes are not transitive.

The IPA field is the key for the NH-Reach NLRI type; the information encoded in the top octet is non-key information. It is possible in principle (although unlikely) for two NLRI to be validly present in an UPDATE message with identical IPA fields but different types. However, two NLRI with the same IPA field and different State fields MUST NOT be encoded in the same UPDATE message. If such is

encountered, the receiver MUST behave as though the state "Unknown" was received for the IPA in question.

6. Client Procedures for NH-Reach Changes

When an entry is added to a route server client's ReachAsk-In for a route server peering session, the client will then attempt to verify connectivity to the host depicted by that entry. The procedure described in this specification utilizes BFD.

If no existing BFD session exists to this next hop, a BFD session is provisioned to that IP address and the LocReach reachability state (Section 4.2) is set to Unknown.

If the client cannot establish a BFD session with an entry in its ReachAsk-In, the next hop remains in LocReach with its Reachable state Unknown.

Once the BFD session moves to the Up state, the LocReach reachability state is set to Up.

When the BFD session transitions out of the Up state to the Down state, the LocReach reachability state is set to Down.

If the BFD session transitions out of the Up state to the AdminDown state, the LocReach reachability state is set to Unknown.

When entries are removed from the route server client's ReachAsk-In for a route server peering session, the client MAY delay de-provisioning the BFD peering session. If the client delays de-provisioning the session, it should remove it if the BFD session transitions to the Down or AdminDown states.

7. Recommendations for Using BFD

The RECOMMENDED way a client router can confirm the data plane connectivity to its next hops is available, is the use of BFD in asynchronous mode. Echo mode MAY be used if both client routers running a BFD session support this. The use of authentication in BFD is OPTIONAL as there is a certain level of trust between the operators of the client routers at a particular IXP. If trust cannot be assumed, it is recommended to use pair-wise keys (how this can be achieved is outside the scope of this document). The ttl/hop limit values as described in section 5 [RFC5881] MUST be obeyed in order to shield BFD sessions against packets coming from outside the IXP.

The following values of the BFD configuration of client routers (see section 6.8.1 [RFC5880]) are RECOMMENDED:

- o DesiredMinTxInterval: 1,000,000 (microseconds)
- o RequiredMinRxInterval: 1,000,000 (microseconds)
- o DetectMult: 3

A client router administrator MAY select more appropriate values to meet the special needs of a particular deployment.

8. Other Considerations

For purposes of routing stability, implementations may wish to apply hysteresis ("holddown") to next hops that have transitioned from reachable to unreachable and back.

Implementations MAY restrict the range of addresses with which they will attempt to form BFD relationships. For example, an implementation might by default only allow BFD relationships with peers that share a subnet with the route server. An implementation MAY apply such restrictions by default.

In a route-server environment, use of this feature SHOULD be restricted to consider only routes that are advertised from within the IXP network. This might include checks on AS_PATH length.

9. Acknowledgments

The authors would like to thank Thomas King for his contributions toward this work.

10. IANA Considerations

IANA is requested to allocate a value from the Subsequent Address Family Identifiers (SAFI) Parameters registry for this proposal. Its Description in that registry shall be NH-Reach with a Reference of this RFC.

11. Security Considerations

The mechanism in this document permits a route server client to influence the contents of the route server's Adj-Ribs-Out through its reports of next hop reachability state using the NH-Reach SAFI. Since this state is per-client, if a route server client is able to inject NH-Reach routes for another route server's BGP session to a client, it can cause the route server to select different forwarding than otherwise expected. This issue may be mitigated using transport security on the BGP sessions between the route server and its clients. See [RFC4272].

The NH-Reach SAFI enables the server to trigger creation of a BFD session on its client. A malicious or misbehaving server could trigger an unreasonable number of sessions, a potential resource exhaustion attack. The sedate default timers proposed in Section 7 mitigate this; they also mitigate concerns about use of the client as a source of packets in a flooding attack. An implementation MAY also impose limits on the number of BFD sessions it will create at the request of the server.

The reachability tests between route server clients themselves may be a target for attack. Such attacks may include forcing a BFD session Down through injecting false BFD state. A less likely attack includes forcing a BFD session to stay Up when its real state is Down. These attacks may be mitigated using the BFD security mechanisms defined in [RFC5880].

12. References

12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <<https://www.rfc-editor.org/info/rfc4271>>.
- [RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <<https://www.rfc-editor.org/info/rfc4760>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, DOI 10.17487/RFC5881, June 2010, <<https://www.rfc-editor.org/info/rfc5881>>.
- [RFC7947] Jasinska, E., Hilliard, N., Raszuk, R., and N. Bakker, "Internet Exchange BGP Route Server", RFC 7947, DOI 10.17487/RFC7947, September 2016, <<https://www.rfc-editor.org/info/rfc7947>>.

12.2. Informative References

- [I-D.chen-bfd-unsolicited]
Chen, E., Shen, N., and R. Raszuk, "Unsolicited BFD for Sessionless Applications", draft-chen-bfd-unsolicited-02 (work in progress), January 2018.
- [I-D.ietf-idr-bgp-bestpath-selection-criteria]
Asati, R., "BGP Bestpath Selection Criteria Enhancement", draft-ietf-idr-bgp-bestpath-selection-criteria-12 (work in progress), June 2019.
- [RFC4272] Murphy, S., "BGP Security Vulnerabilities Analysis", RFC 4272, DOI 10.17487/RFC4272, January 2006, <<https://www.rfc-editor.org/info/rfc4272>>.
- [RFC7880] Pignataro, C., Ward, D., Akiya, N., Bhatia, M., and S. Pallagatti, "Seamless Bidirectional Forwarding Detection (S-BFD)", RFC 7880, DOI 10.17487/RFC7880, July 2016, <<https://www.rfc-editor.org/info/rfc7880>>.

Appendix A. Summary of Document Changes

idr-06: Refresh -05.
idr-04 to idr-05: Added reference to "BGP Bestpath Selection Criteria Enhancement" draft. Rename "next hop" field of NLRI to "Indirect Peer's Address". Add suggestion about AS_PATH length checks.
idr-03 to idr-04: Note other forms of connectivity checks.
idr-02 to idr-03: Substantial rewrite. Introduce NLRI format that embeds state.
idr-01 to idr-02: Move from BGP-LS to NH-Reach SAFI. Lots of editorial changes.
idr-00 to idr-01: Add BGP Capability. Move from NH-Cost to BGP-LS.
ymbk-01 to idr-00: No technical changes; adopted by IDR.
ymbk-00 to ymbk-01: Clarifications to BFD procedures. Use BFD state as an input to BGP route selection.

Appendix B. Other Forms of Connectivity Checks

RFC 5880/5881 BFD is a well-deployed feature. For this reason, it was chosen as the connectivity check utilized for nexthop reachability by this document. As other forms of BFD become more widely deployed, they may also be utilized to provide the connectivity check functionality.

Examples of other such BFD mechanisms include:

- o Seamless BFD [RFC7880]
- o Unsolicited BFD for Sessionless Applications
[I-D.chen-bfd-unsolicited]

Implementations MUST support RFC 5880/5881 BFD to be compliant with this specification. Implementations MAY support other forms of connectivity check, including those mechanisms listed above, so long as they provide the ability to fall-back to RFC 5880/5881 BFD.

Authors' Addresses

Randy Bush
Internet Initiative Japan
5147 Crystal Springs
Bainbridge Island, Washington 98110
US

Email: randy@psg.com

Jeffrey Haas
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089
US

Email: jhaas@juniper.net

John G. Scudder
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089
US

Email: jgs@juniper.net

Arnold Nipper
DE-CIX Management GmbH
Lichtstrasse 43i
Cologne 50825
Germany

Email: arnold.nipper@de-cix.net

Internet-Draft Making RSeS aware of IXP Data Link FailuresSeptember 2020

Christoph Dietzel
DE-CIX Management GmbH
Lichtstrasse 43i
Cologne 50825
Germany

Email: christoph.dietzel@de-cix.net

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 2, 2018

N. Gupta
A. Dogra
Cisco Systems, Inc.
C. Docherty
AT&T
G. Mirsky
J. Tantsura
Individual
January 29, 2018

Fast failure detection in VRRP with Point to Point BFD
draft-ietf-rtgwg-vrrp-bfd-p2p-00

Abstract

This document describes how Point to Point Bidirectional Forwarding Detection (BFD) can be used to support sub-second detection of a Master Router failure in the Virtual Router Redundancy Protocol (VRRP).

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 August 2, 2018.

Copyright Notice

Copyright (c) 2018 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. Requirements Language	4
3. Applicability of Point to Point BFD	5
3.1. Extension to VRRP protocol	5
3.2. VRRP Peer Table	6
3.3. VRRP BACKUP ADVERTISEMENT Packet Type	7
3.4. Sample configuration	8
3.5. Critical BFD session	9
3.6. Protocol State Machine	9
3.6.1. Parameters Per Virtual Router	9
3.6.2. Timers	10
3.6.3. VRRP State Machine with Point to Point BFD	10
4. Scalability Considerations	20
5. Operational Considerations	21
6. Applicability to VRRPv2	22
7. IANA Considerations	23
7.1. A New Name Space for VRRP Packet Types	23
8. Security Considerations	24
9. Acknowledgements	25
10. Normative References	26
Authors' Addresses	27

1. Introduction

The Virtual Router Redundancy Protocol (VRRP) provides redundant Virtual gateways in the Local Area Network (LAN), which is typically the first point of failure for end-hosts sending traffic out of the LAN. Fast failure detection of VRRP Master is critical in supporting high availability of services and improved Quality of Experience to users. In VRRP [RFC5798] specification, Backup routers depend on VRRP packets generated at a regular interval by the Master router, to detect the health of the VRRP Master. Faster failure detection can be achieved within VRRP protocol by reducing the Advertisement and Master Down Interval. However, sub second Advert timers, can put extra load on CPU and the network bandwidth which may not be desirable.

Since the VRRP protocol depends on the availability of Layer 3 IPv4 or IPv6 connectivity between redundant peers, the VRRP protocol can interact with the Layer 3 variant of BFD as described in [RFC5881] to achieve a much faster failure detection of the VRRP Master on the LAN. BFD, as specified by the [RFC5880] can provide a much faster failure detection in the range of 150ms, if implemented in the part of a Network device which scales better than VRRP when sub second Advert timers are used.

2. Requirements Language

In this document, several words are used to signify the requirements of the specification. 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]

3. Applicability of Point to Point BFD

BFD for IPv4 or IPv6 (Single Hop) [RFC5881] requires that in order for a BFD session to be formed both peers participating in a BFD session need to know its peer IPv4 or IPV6 address. This poses a unique problem with the definition of the VRRP protocol, that makes the use of BFD for IPv4 or IPv6 [RFC5881] more challenging. In VRRP it is only the Master router that sends Advert packets. This means that a Master router is not aware of any Backup routers, and Backup routers are only aware of the Master router. This also means that a Backup router is not aware of any other Backup routers in the Network.

Since BFD for IPv4 or IPv6 [RFC5881] requires that a session be formed by both peers using a full destination and source address, there needs to be some external means to provide this information to BFD on behalf of VRRP. Once the peer information is made available, VRRP can form BFD sessions with its peer Virtual Router. The BFD session for a given Virtual Router is identified as the Critical Path BFD Session, which is the session that forms between the current VRRP Master router, and the highest priority Backup router. When the Critical Path BFD Session identified by VRRP as having changed state from Up to Down, then this will be interpreted by the VRRP state machine on the highest priority Backup router as a Master Down event. A Master Down event means that the highest priority Backup peer will immediately become the new Master for the Virtual Router.

NOTE: At all times, the normal fail-over mechanism defined in the VRRP [RFC5798] will be unaffected, and the BFD fail-over mechanism will always resort to normal VRRP fail-over.

This draft defines the mechanism used by the VRRP protocol to build a peer table that will help in forming of BFD session and the detection of Critical Path BFD session. If the Critical Path BFD session were to go down, it will signal a Master Down event and make the most preferred Backup router as the VRRP Master router. This requires an extension to the VRRP protocol.

This can be achieved by defining a new type in the VRRP Advert packet, and allowing VRRP peers to build a peer table in any of the operational state, Master or Backup.

3.1. Extension to VRRP protocol

In this mode of operation VRRP peers learn the adjacent routers, and form BFD session between the learnt routers. In order to build the peer table, all routers send VRRP Advert packets whilst in any of the operational states (Master or Backup). Normally VRRP peers only send

Advert packets whilst in the Master state, however in this mode VRRP Backup peers will also send Advert packets with the type field set to BACKUP ADVERTISEMENT type defined in Section 3.3 of this document. The VRRP Master router will still continue to send packets with the Advert type as ADVERTISEMENT as defined in the VRRP protocol. This is to maintain inter-operability with peers complying to VRRP protocol.

Additionally, Advert packets sent from Backup Peers must not use the Virtual router MAC address as the source address. Instead it must use the Interface MAC address as the source address from which the packet is sent from. This is because the source MAC override feature is used by the Master to send Advert packets from the Virtual Router MAC address, which is used to keep the bridging cache on LAN switches and bridging devices refreshed with the destination port for the Virtual Router MAC.

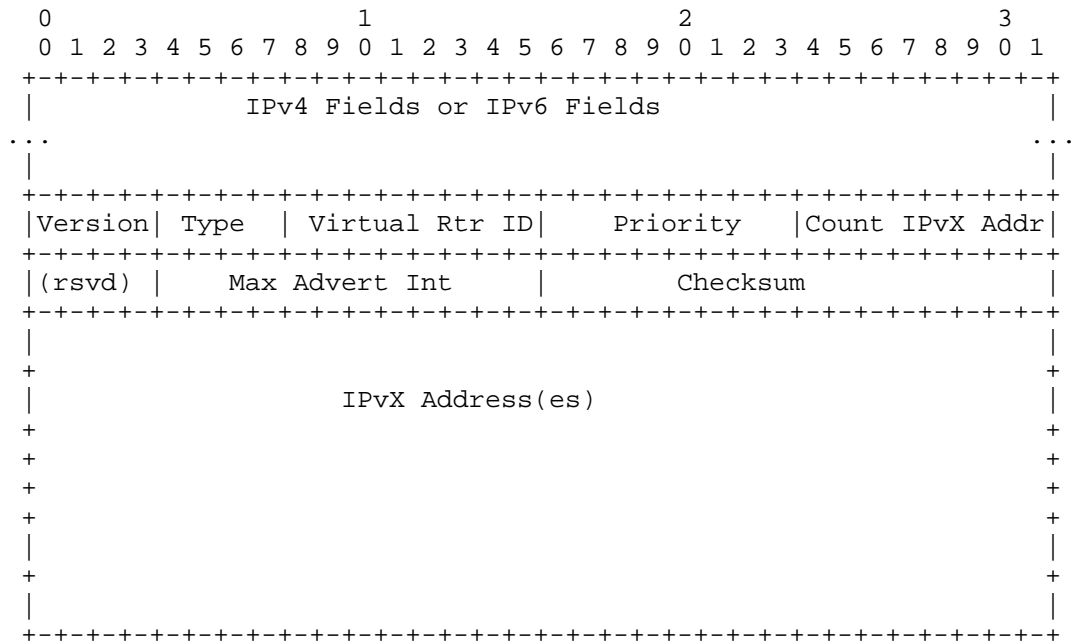
3.2. VRRP Peer Table

VRRP peers can now form the peer table by learning the source address in the ADVERTISEMENT or BACKUP ADVERTISEMENT packet sent by VRRP Master or Backup peers. This allows peers to create BFD sessions with other operational peers.

A peer entry should be removed from the peer table if Advert is not received from a peer for a period of (3 * the Advert interval).

3.3. VRRP BACKUP ADVERTISEMENT Packet Type

The following figure shows the VRRP packet as defined in VRRP [RFC5798] RFC.



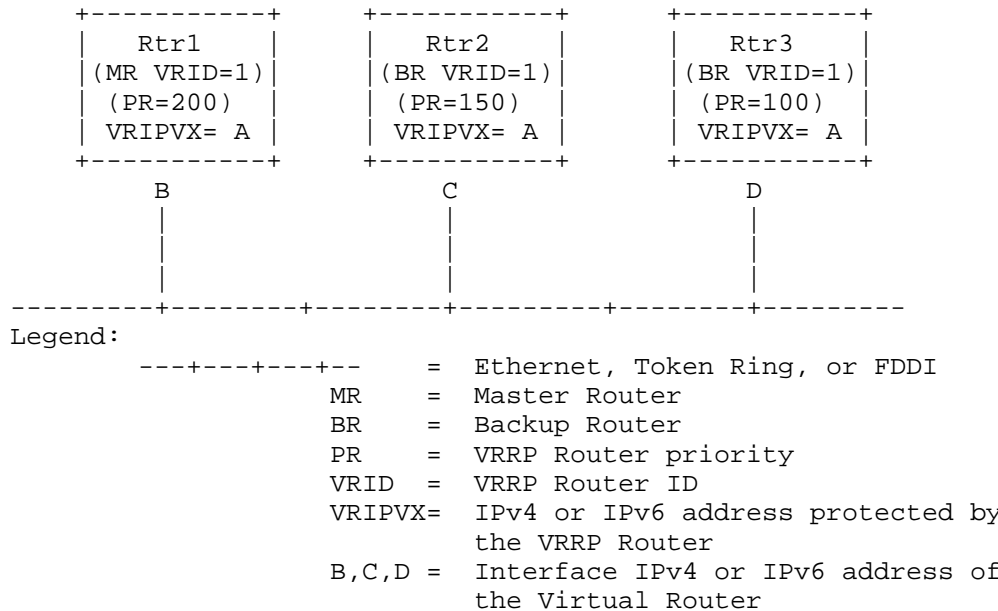
The type field specifies the type of this VRRP packet. The type field can have two values. Type 1 (ADVERTISEMENT) is used by the VRRP Master Router. Type 2 (BACKUP ADVERTISEMENT) is used by the VRRP Backup router. This is to distinguish the packets sent by the VRRP backup Router. VRRP Backup fills Backup_Advertisement_Interval in the Max Advert Int of BACKUP ADVERTISEMENT packet. Rest of the fields in Advert packet remain the same.

- 1 ADVERTISEMENT
- 2 BACKUP ADVERTISEMENT

A packet with unknown type MUST be discarded.

3.4. Sample configuration

The following figure shows a simple network with three VRRP routers implementing one virtual router.



In the above configuration there are three routers on the LAN protecting an IPv4 or IPv6 address associated to a Virtual Router ID 1. Rtr1 is the Master router since it has the highest priority compared to Rtr2 and Rtr3. Now if peer learning extension is enabled on all the peers. Rtr1 will send the Advert packet with type field set to 1. While Rtr2 and Rtr3 will send the Advert packet with type field set to 2. In the above configuration the peer table built at each router is shown below:

Rtr1 Peer table

Peer Address	Priority
C	150
D	100

Rtr2 Peer table

Peer Address	Priority
B	200
D	100

Rtr3 Peer table

Peer Address	Priority
B	200
C	150

Once the peer tables are formed, VRRP on each router can form a BFD sessions with the learnt peers.

3.5. Critical BFD session

The Critical BFD Session is determined to be the session between the VRRP Master and the next best VRRP Backup. Failure of the Critical BFD session indicates that the Master is no longer available and the most preferred Backup will now become Master.

In the above example the Critical BFD session is shared between Rtr1 and Rtr2. If the BFD Session goes from Up to Down state, Rtr2 can treat it as a Master down event and immediately assume the role of VRRP Master router for VRID 1 and Rtr3 will become the critical Backup. If the priorities of two Backup routers are same then the primary IPvX Address of the sender is used to determine the highest priority Backup. Where higher IPvX address has higher priority.

3.6. Protocol State Machine

3.6.1. Parameters Per Virtual Router

Following parameters are added to the VRRP protocol to support this mode of operation.

Backup_Advertisement_Interval	Time interval between BACKUP ADVERTISEMENTS (centiseconds). Default is 100 centiseconds (1 second).
Backup_Adver_Interval	Advertisement interval contained in BACKUP ADVERTISEMENTS received from the Backup (centiseconds). This value is saved by virtual routers used, to compute Backup_Down_Interval.
Backup_Down_Interval	Time interval for VRRP instance to declare Backup down (centiseconds). Calculated as $(3 * \text{Backup_Adver_Interval})$ for each VRRP Backup.
Critical_Backup	Procedure outlined in section 3.4 of this document is used to determine the Critical_Backup at each VRRP Instance.
Critical_BFD_Session	The Critical BFD Session is the session between the VRRP Master and Critical_Backup.

3.6.2. Timers

Following timers are added to the VRRP protocol to support this mode of operation.

Backup_Down_Timer	Timer that fires when BACKUP ADVERTISEMENT has not been heard from a backup peer for Backup_Down_Interval.
Backup_Adver_Timer	Timer that fires to trigger sending of BACKUP ADVERTISEMENT based on Backup_Advertisement_Interval.

3.6.3. VRRP State Machine with Point to Point BFD

Following State Machine replaces the state Machine outlined in section 6.4 of the VRRP protocol [RFC5798] to support this mode of operation. Please refer to the section 6.4 of [RFC5798] for State description.

3.6.3.1. Initialize

Following state machine replaces the state machine outlined in section 6.4.1 of [RFC5798]

```
(100) If a Startup event is received, then:

    (105) - If the Priority = 255 (i.e., the router owns the IPvX
    address associated with the virtual router), then:

        (110) + Send an ADVERTISEMENT

        (115) + If the protected IPvX address is an IPv4 address, then:

            (120) * Broadcast a gratuitous ARP request containing the
            virtual router MAC address for each IP address associated
            with the virtual router.

        (125) + else // IPv6

            (130) * For each IPv6 address associated with the virtual
            router, send an unsolicited ND Neighbor Advertisement with
            the Router Flag (R) set, the Solicited Flag (S) unset, the
            Override flag (O) set, the target address set to the IPv6
            address of the virtual router, and the target link-layer
            address set to the virtual router MAC address.

        (135) +endif // was protected addr IPv4?

        (140) + Set the Adver_Timer to Advertisement_Interval

        (145) + Transition to the {Master} state

    (150) - else // rtr does not own virt addr

        (155) + Set Master_Adver_Interval to Advertisement_Interval

        (160) + Set the Master_Down_Timer to Master_Down_Interval

        (165) + Set Backup_Adver_Timer to Backup_Advertisement_Interval

        (170) + Transition to the {Backup} state

    (175) -endif // priority was not 255

(180) endif // startup event was recv
```

3.6.3.2. Backup

Following state machine replaces the state machine outlined in section 6.4.2 of [RFC5798]

```
(300) While in this state, a VRRP router MUST do the following:

(305) - If the protected IPvX address is an IPv4 address, then:

    (310) + MUST NOT respond to ARP requests for the IPv4
    address(es) associated with the virtual router.

(315) - else // protected addr is IPv6

    (320) + MUST NOT respond to ND Neighbor Solicitation messages
    for the IPv6 address(es) associated with the virtual router.

    (325) + MUST NOT send ND Router Advertisement messages for the
    virtual router.

(330) -endif // was protected addr IPv4?

(335) - MUST discard packets with a destination link-layer MAC
address equal to the virtual router MAC address.

(340) - MUST NOT accept packets addressed to the
IPvX address(es) associated with the virtual router.

(345) - If a Shutdown event is received, then:

    (350) + Cancel the Master_Down_Timer.

    (355) + Cancel the Backup_Adver_Timer.

    (360) + Cancel Backup_Down_Timers.

    (365) + Remove Peer table.

    (370) + If Critical_BFD_Session Exists:

        (375) * Tear down the Critical_BFD_Session.

    (380) + endif // Critical_BFD_Session Exists?

    (385) + Send a BACKUP ADVERTISEMENT with Priority = 0.

    (390) + Transition to the {Initialize} state.
```



```
(395) -endif // shutdown recv

(400) - If the Master_Down_Timer fires or
      If Critical_BFD_Session transitions from UP to DOWN, then:

(405) + Send an ADVERTISEMENT

(415) + If the protected IPvX address is an IPv4 address, then:

      (420) * Broadcast a gratuitous ARP request on that interface
            containing the virtual router MAC address for each IPv4
            address associated with the virtual router.

(425) + else // ipv6

      (430) * Compute and join the Solicited-Node multicast
            address [RFC4291] for the IPv6 address(es) associated with
            the virtual router.

      (435) * For each IPv6 address associated with the virtual
            router, send an unsolicited ND Neighbor Advertisement with
            the Router Flag (R) set, the Solicited Flag (S) unset, the
            Override flag (O) set, the target address set to the IPv6
            address of the virtual router, and the target link-layer
            address set to the virtual router MAC address.

(440) +endif // was protected addr ipv4?

(445) + Set the Adver_Timer to Advertisement_Interval.

(450) + If the Critical_BFD_Session exists:

      (455) @ Tear Critical_BFD_Session.

(460) + endif // Critical_BFD_Session exists

(465) + Calculate the Critical_Backup.

(470) + If the Critical_Backup exists:

      (475) * Bootstrap Critical_BFD_Session with the
            Critical_Backup.

(480) + endif //Critical_Backup exists?

(485) + Transition to the {Master} state.

(490) -endif // Master_Down_Timer fired
```

```
(485) - If an ADVERTISEMENT is received, then:

    (490) + If the Priority in the ADVERTISEMENT is zero, then:

        (495) * Set the Master_Down_Timer to Skew_Time.

        (500) * If the Critical_BFD_Session exists:

            (505) * Tear Critical_BFD_Session with the Master.

        (510) * endif // Critical_BFD_Session exists

    (515) + else // priority non-zero

        (520) * If Preempt_Mode is False, or if the Priority in the
        ADVERTISEMENT is greater than or equal to the local
        Priority, then:

            (525) @ Set Master_Adver_Interval to Adver Interval
            contained in the ADVERTISEMENT.

            (530) @ Recompute the Master_Down_Interval.

            (535) @ Reset the Master_Down_Timer to
            Master_Down_Interval.

            (540) @ Determine Critical_Backup.

            (545) @ If Critical_BFD_Session does not exists and this
            instance is the Critical_Backup:

                (550) @+ Bootstrap Critical_BFD_Session with Master.

            (555) @ endif //Critical_BFD_Session exists check

        (560) * else // preempt was true or priority was less

            (565) @ Discard the ADVERTISEMENT.

        (570) *endif // preempt test

    (575) +endif // was priority zero?

(580) -endif // was advertisement rcv?

(585) - If a BACKUP ADVERTISEMENT is received, then:

    (590) + If the Priority in the BACKUP ADVERTISEMENT is zero,
```

```
        then:

(595) * Cancel Backup_Down_Timer.

(600) * Remove the Peer from Peer table.

(605) + else // priority non-zero

(610) * Update the peer table with peer information.

(615) * Set Backup_Adver_Interval to Adver Interval
contained in the BACKUP ADVERTISEMENT.

(620) * Recompute the Backup_Down_Interval.

(625) * Reset the Backup_Down_Timer to Backup_Down_Interval.

(630) +endif // was priority zero?

(635) + Recalculate Critical_Backup.

(640) + If Critical_BFD_Session exists and this
instance is not the Critical_Backup:

(645) * Tear Down the Critical_BFD_Session.

(650) + else If Critical_BFD_Session does not exists and this
instance is the Critical_Backup:

(655) * BootStrap Critical_BFD_Session with Master.

(660) + endif // Critical_Backup change

(665) -endif // was backup advertisement recv?

(670) - If Backup_Down_Timer fires, then:

(675) + Remove the Peer from Peer table.

(680) + If Critical_BFD_Session does not exist:

(685) @ Recalculate Critical_Backup.

(690) @ If This instance is the Critical_Backup:

(695) +@ BootStrap Critical_BFD_Session with Master.

(700) @ endif // Critical_Backup change
```

```
(705) + endif // Critical_BFD_Session does not exist?
(710) -endif // Backup_Down_Timer fires?
(715) - If Backup_Adver_Timer fires, then:
    (720) + Send a BACKUP ADVERTISEMENT.
    (725) + Reset the Backup_Adver_Timer to
            Backup_Advertisement_Interval.
(730) -endif // Backup_Down_Timer fires?
(735) endwhile // Backup state
```

3.6.3.3. Master

Following state machine replaces the state machine outlined in section 6.4.3 of [RFC5798]

```
(800) While in this state, a VRRP router MUST do the following:
    (805) - If the protected IPvX address is an IPv4 address, then:
        (810) + MUST respond to ARP requests for the IPv4 address(es)
                associated with the virtual router.
    (815) - else // ipv6
        (820) + MUST be a member of the Solicited-Node multicast
                address for the IPv6 address(es) associated with the virtual
                router.
        (825) + MUST respond to ND Neighbor Solicitation message for
                the IPv6 address(es) associated with the virtual router.
        (830) + MUST send ND Router Advertisements for the virtual
                router.
        (835) + If Accept_Mode is False: MUST NOT drop IPv6
                Neighbor Solicitations and Neighbor Advertisements.
    (840) -endif // ipv4?
    (845) - MUST forward packets with a destination link-layer MAC
            address equal to the virtual router MAC address.
```

(850) - MUST accept packets addressed to the IPvX address(es) associated with the virtual router if it is the IPvX address owner or if Accept_Mode is True. Otherwise, MUST NOT accept these packets.

(855) - If a Shutdown event is received, then:

(860) + Cancel the Adver_Timer.

(865) + Send an ADVERTISEMENT with Priority = 0,

(870) + Cancel Backup_Down_Timers.

(875) + Remove Peer table.

(880) + If Critical_BFD_Session Exists:

(885) * Tear down Critical_BFD_Session

(890) + endif // If Critical_BFD_Session Exists

(895) + Transition to the {Initialize} state.

(900) -endif // shutdown recv

(905) - If the Adver_Timer fires, then:

(910) + Send an ADVERTISEMENT.

(915) + Reset the Adver_Timer to Advertisement_Interval.

(920) -endif // advertisement timer fired

(925) - If an ADVERTISEMENT is received, then:

(930) -+ If the Priority in the ADVERTISEMENT is zero, then:

(935) -* Send an ADVERTISEMENT.

(940) -* Reset the Adver_Timer to Advertisement_Interval.

(945) -+ else // priority was non-zero

(950) -* If the Priority in the ADVERTISEMENT is greater than the local Priority,

(955) -* or

```
(960) -* If the Priority in the ADVERTISEMENT is equal to
the local Priority and the primary IPvX Address of the
sender is greater than the local primary IPvX Address, then:

(965) -@ Cancel Adver_Timer

(970) -@ Set Master_Adver_Interval to Adver Interval
contained in the ADVERTISEMENT

(975) -@ Recompute the Skew_Time

(980) @ Recompute the Master_Down_Interval

(985) @ Set Master_Down_Timer to Master_Down_Interval

(990) If Critical_BFD_Session Exists:

    (995) @+ Tear Critical_BFD_Session

(960) @ endif //Critical_BFD_Session Exists?

(965) @ Calculate Critical_Backup.

(970) @ If this instance is Critical_Backup:

    (975) @+ Bootstrap Critical_BFD_Session with new
        Master.

(980) @ endif // am i Critical_Backup?

(985) @ Transition to the {Backup} state

(990) * else // new Master logic

    (995) @ Discard ADVERTISEMENT

(1000) *endif // new Master detected

(1005) +endif // was priority zero?

(1010) -endif // advert recv

(1015) - If a BACKUP ADVERTISEMENT is received, then:

    (1020) + If the Priority in the BACKUP ADVERTISEMENT is
        zero, then:

        (1025) * Remove the Peer from peer table.
```

```
(1030) + else: // priority non-zero
    (1035) * Update the Peer info in peer table.
    (1040) * Recompute the Backup_Down_Interval
    (1045) * Reset the Backup_Down_Timer to
            Backup_Down_Interval
(1050) + endif // priority in backup advert zero
(1055) + Calculate the Critical_Backup
(1060) + If Critical_BFD_Session doesnot exist:
    (1065) * Bootstrap Critical_BFD_Session
(1070) + else if Critical_BFD_Session exist and
        Critical_Backup changes:
    (1075) + Tear Critical_BFD_Session with old Backup
    (1080) + Bootstrap Critical_BFD_Session with Critical_Backup
(1085) + endif // Critical_BFD_Session check?
(1090) - endif // backup advert recv
(1095) - If Critical_BFD_Session transitions from UP to DOWN,
then:
    (1100) + Cancel Backup_Down_Timer
    (1105) + Delete the Peer info from peer table
    (1200) + Calculate the Critical_Backup
    (1205) + Bootstrap Critical_BFD_Session with Critical_Backup
(1210) - endif // BFD session transition
(1215) endwhile // in Master
```

4. Scalability Considerations

To reduce the number of packets generated at a regular interval, Backup Advert packets may be sent at a reduced rate as compared to Advert packets sent by the VRRP Master.

5. Operational Considerations

A VRRP peer that forms a member of this Virtual Router, but does not support this feature or extension must be configured with the lowest priority, and will only operate as the Router of last resort on failure of all other VRRP routers supporting this functionality.

It is recommended that mechanism defined by this draft, to interface VRRP with BFD should be used when BFD can support more aggressive monitoring timers than VRRP. Otherwise it is desirable not to interface VRRP with BFD for determining the health of VRRP Master.

This Draft does not preclude the possibility of the peer table being populated by means of manual configuration, instead of using the BACKUP ADVERTISEMENT as defined by the Draft.

6. Applicability to VRRPv2

The workings of this Draft can be extended to VRRPv2 [RFC3768], with the introduction of BACKUP ADVERTISEMENT and Peer Table as outlined in the Draft.

7. IANA Considerations

This document requests IANA to create a new name space that is to be managed by IANA. The document defines a new VRRP Packet Type. The VRRP Packet Types are discussed below.

- a) Type 1 (ADVERTISEMENT) defined in section 5.2.2 of [RFC5798]
- b) Type 2 (BACKUP ADVERTISEMENT) defined in section 3.3 of this document

7.1. A New Name Space for VRRP Packet Types

This document defines in Section 3.3 a "BACKUP ADVERTISEMENT" VRRP Packet Type. The new name space has to be created by the IANA and they will maintain this new name space. The field for this namespace is 4-Bits, and IANA guidelines for assignments for this field are as follows:

ADVERTISEMENT	1
BACKUP ADVERTISEMENT	2

Future allocations of values in this name space are to be assigned by IANA using the "Specification Required" policy defined in [IANA-CONS]

8. Security Considerations

Security considerations discussed in [RFC5798], [RFC5880], apply to this document. There are no additional security considerations identified by this draft.

9. Acknowledgements

The authors gratefully acknowledge the contributions of Gerry Meyer, and Mouli Chandramouli, for their contributions to the draft. The authors will also like to thank Jeffrey Haas, Maik Pfeil, Chris Bowers, Vengada Prasad Govindan and Alexander Vainshtein for their comments and suggestions.

10. Normative References

- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, 2010.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, 1997.
- [RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, 2010.
- [RFC5798] Nadas, S., "Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6", RFC 5798, 2010.
- [RFC3768] Hinden, R., "Virtual Router Redundancy Protocol (VRRP)", RFC 3768, 2004.
- [IANA-CONS] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 2434, 1998.

Authors' Addresses

Nitish Gupta
Cisco Systems, Inc.
3265 CISCO Way
San Jose 95134
United States

Phone: +91 80 4429 2530
Email: nitisgup@cisco.com
URI: <http://www.cisco.com/>

Aditya Dogra
Cisco Systems, Inc.
Sarjapur Outer Ring Road
Bangalore 560103
India

Phone: +91 80 4429 2166
Email: adogra@cisco.com
URI: <http://www.cisco.com/>

Colin Docherty
AT&T
23 The Maltings
Haddington, Scotland EH414EF
United Kingdom

Email: colin.docherty@att.com

Greg Mirsky
Individual

Email: gregimirsky@gmail.com

Jeff Tantsura
Individual

Email: jefftant.ietf@gmail.com

Inter-Domain Routing
Internet-Draft
Intended status: Standards Track
Expires: December 31, 2018

Z. Li
Huawei
S. Aldrin
Google, Inc
J. Tantsura
Nuage Networks
G. Mirsky
ZTE Corp.
S. Zhuang
Huawei
K. Talaulikar
Cisco Systems
June 29, 2018

BGP Link-State Extensions for Seamless BFD
draft-li-idr-bgp-ls-sbfd-extensions-02

Abstract

Seamless Bidirectional Forwarding Detection (S-BFD) defines a simplified mechanism to use Bidirectional Forwarding Detection (BFD) with large portions of negotiation aspects eliminated, thus providing benefits such as quick provisioning as well as improved control and flexibility to network nodes initiating the path monitoring. The link-state routing protocols (IS-IS and OSPF) have been extended to advertise the Seamless BFD (S-BFD) Discriminators.

This draft defines extensions to the BGP Link-state address-family to carry the S-BFD Discriminators information via BGP.

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 <https://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 December 31, 2018.

Copyright Notice

Copyright (c) 2018 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 (<https://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	2
2. Terminology	3
3. Problem and Requirement	3
4. BGP-LS Extensions for S-BFD Discriminator	4
5. IANA Considerations	6
6. Manageability Considerations	6
6.1. Operational Considerations	6
6.2. Management Considerations	6
7. Security Considerations	6
8. Acknowledgements	7
9. References	7
9.1. Normative References	7
9.2. Informative References	7
Authors' Addresses	8

1. Introduction

Seamless Bidirectional Forwarding Detection (S-BFD) [RFC7880] defines a simplified mechanism to use Bidirectional Forwarding Detection (BFD) [RFC5880] with large portions of negotiation aspects eliminated, thus providing benefits such as quick provisioning as well as improved control and flexibility to network nodes initiating the path monitoring.

For monitoring of a service path end-to-end via S-BFD, the headend/initiator node needs to know the S-BFD Discriminator of the destination/tail-end node of that service. The link-state routing protocols (IS-IS, OSPF and OSPFv3) have been extended to advertise the S-BFD Discriminators. With this a initiator node can learn the S-BFD discriminator for all nodes within its IGP area/level or optionally within the domain. With networks being divided into multiple IGP domains for scaling and operational considerations, the service endpoints that require end to end S-BFD monitoring often span across IGP domains.

BGP Link-State (BGP-LS) [RFC7752] enables the collection and distribution of IGP link-state topology information via BGP sessions across IGP areas/levels and domains. The S-BFD discriminator(s) of a node can thus be distributed along with the topology information via BGP-LS across IGP domains and even across multiple Autonomous Systems (AS) within an administrative domain.

This draft defines extensions to BGP-LS for carrying the S-BFD Discriminators information.

2. Terminology

This memo makes use of the terms defined in [RFC7880].

3. Problem and Requirement

Seamless MPLS [I-D.ietf-mpls-seamless-mpls] extends the core domain and integrates aggregation and access domains into a single MPLS domain. In a large network, the core and aggregation networks can be organized as different ASes. Although the core and aggregation networks are segmented into different ASes, an E2E LSP can be created using hierarchical BGP signaled LSPs based on iBGP labeled unicast within each AS, and eBGP labeled unicast to extend the LSP across AS boundaries. This provides a seamless MPLS transport connectivity for any two service end-points across the entire domain. In order to detect failures for such end to end services and trigger faster protection and/or re-routing, S-BFD MAY be used for the Service Layer (e.g. for MPLS VPNs, PW, etc.) or the Transport Layer monitoring. This brings up the need for setting up S-BFD session spanning across AS domains.

In a similar Segment Routing (SR) [I-D.ietf-spring-segment-routing] multi-domain network, an end to end SR Policy [I-D.ietf-spring-segment-routing-policy] path may be provisioned between service end-points across domains either via local provisioning or by a controller or signalled from a Path Computation

Engine (PCE). Monitoring using S-BFD can similarly be setup for such a SR Policy.

Extending the automatic discovery of S-BFD discriminators of nodes from within the IGP domain to across the administrative domain using BGP-LS enables setting up of S-BFD sessions on demand across IGP domains. The S-BFD discriminators for service end point nodes MAY be learnt by the PCE or a controller via the BGP-LS feed that it gets from across IGP domains and it can signal or provision the remote S-BFD discriminator on the initiator node on demand when S-BFD monitoring is required. The mechanisms for the signaling of the S-BFD discriminator from the PCE/controller to the initiator node and setup of the S-BFD session is outside the scope of this document.

Additionally, the service end-points themselves MAY also learn the S-BFD discriminator of the remote nodes themselves by receiving the BGP-LS feed via a route reflector (RR) or a centralized BGP Speaker that is consolidating the topology information across the domains. The initiator node can then itself setup the S-BFD session to the remote node without a controller/PCE assistance.

While this document takes examples of MPLS and SR paths, the S-BFD discriminator advertisement mechanism is applicable for any S-BFD use-case in general.

4. BGP-LS Extensions for S-BFD Discriminator

The BGP-LS [RFC7752] specifies the Node NLRI for advertisement of nodes and their attributes using the BGP-LS Attribute. The S-BFD discriminators of a node are considered as its node level attribute and advertised as such.

This document defines a new BGP-LS Attribute TLV called the S-BFD Discriminators TLV and its format is as follows:

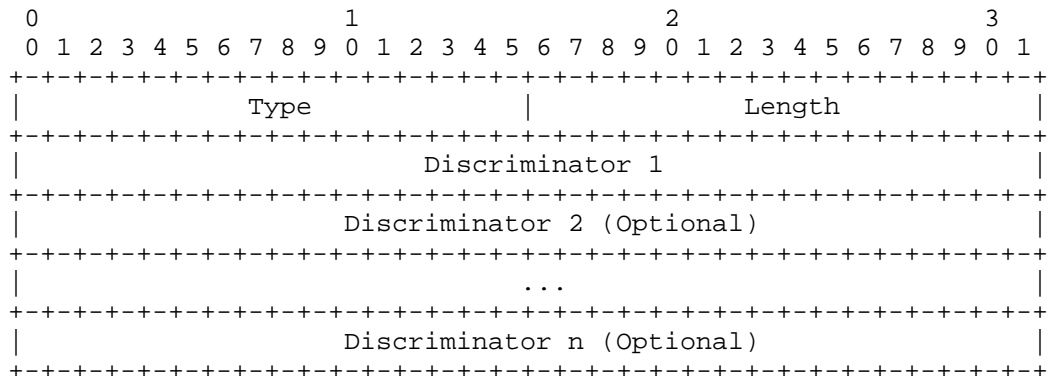


Figure 1: S-BFD Discriminators TLV

where:

- o Type: TBD (see IANA Considerations Section 5)
- o Length: variable. Minimum of 8 octets and increments of 4 octets there on for each additional discriminator
- o Discriminators : multiples of 4 octets, each carrying a S-BFD local discriminator value of the node. At least one discriminator MUST be included in the TLV.

The S-BFD Discriminators TLV can only be added to the BGP-LS Attribute associated with the Node NLRI that originates the corresponding underlying IGP TLV/sub-TLV as described below. This information is derived from the protocol specific advertisements as below..

- o IS-IS, as defined by the S-BFD Discriminators sub-TLV in [RFC7883].
- o OSPFv2/OSPFv3, as defined by the S-BFD Discriminators TLV in [RFC7884].

When the node is not running any of the IGPs but running a protocol like BGP, then the locally provisioned S-BFD discriminators of the node MAY be originated as part of the BGP-LS attribute within the Node NLRI corresponding to the local node.

5. IANA Considerations

This document requests assigning code-points from the registry "BGP-LS Node Descriptor, Link Descriptor, Prefix Descriptor, and Attribute TLVs" based on table below. The column "IS-IS TLV/Sub-TLV" defined in the registry does not require any value and should be left empty.

Code Point	Description	Length
TBD	S-BFD Discriminators TLV	variable

6. Manageability Considerations

This section is structured as recommended in [RFC5706].

The new protocol extensions introduced in this document augment the existing IGP topology information that was distributed via [RFC7752]. Procedures and protocol extensions defined in this document do not affect the BGP protocol operations and management other than as discussed in the Manageability Considerations section of [RFC7752]. Specifically, the malformed NLRIs attribute tests in the Fault Management section of [RFC7752] now encompass the new TLVs for the BGP-LS NLRI in this document.

6.1. Operational Considerations

No additional operation considerations are defined in this document.

6.2. Management Considerations

No additional management considerations are defined in this document.

7. Security Considerations

The new protocol extensions introduced in this document augment the existing IGP topology information that was distributed via [RFC7752]. Procedures and protocol extensions defined in this document do not affect the BGP security model other than as discussed in the Security Considerations section of [RFC7752]. More specifically the aspects related to limiting the nodes and consumers with which the topology information is shared via BGP-LS to trusted entities within an administrative domain.

Advertising the S-BFD Discriminators via BGP-LS makes it possible for attackers to initiate S-BFD sessions using the advertised

information. The vulnerabilities this poses and how to mitigate them are discussed in [RFC7752].

8. Acknowledgements

The authors would like to thank Nan Wu for his contributions to this work.

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.
- [RFC7880] Pignataro, C., Ward, D., Akiya, N., Bhatia, M., and S. Pallagatti, "Seamless Bidirectional Forwarding Detection (S-BFD)", RFC 7880, DOI 10.17487/RFC7880, July 2016, <<https://www.rfc-editor.org/info/rfc7880>>.
- [RFC7883] Ginsberg, L., Akiya, N., and M. Chen, "Advertising Seamless Bidirectional Forwarding Detection (S-BFD) Discriminators in IS-IS", RFC 7883, DOI 10.17487/RFC7883, July 2016, <<https://www.rfc-editor.org/info/rfc7883>>.
- [RFC7884] Pignataro, C., Bhatia, M., Aldrin, S., and T. Ranganath, "OSPF Extensions to Advertise Seamless Bidirectional Forwarding Detection (S-BFD) Target Discriminators", RFC 7884, DOI 10.17487/RFC7884, July 2016, <<https://www.rfc-editor.org/info/rfc7884>>.

9.2. Informative References

- [I-D.ietf-mpls-seamless-mpls] Leymann, N., Decraene, B., Filsfils, C., Konstantynowicz, M., and D. Steinberg, "Seamless MPLS Architecture", draft-ietf-mpls-seamless-mpls-07 (work in progress), June 2014.

- [I-D.ietf-spring-segment-routing]
Filsfils, C., Previdi, S., Ginsberg, L., Decraene, B.,
Litkowski, S., and R. Shakir, "Segment Routing
Architecture", draft-ietf-spring-segment-routing-15 (work
in progress), January 2018.
- [I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Sivabalan, S., daniel.voyer@bell.ca, d.,
bogdanov@google.com, b., and P. Mattes, "Segment Routing
Policy Architecture", draft-ietf-spring-segment-routing-
policy-01 (work in progress), June 2018.
- [RFC5706] Harrington, D., "Guidelines for Considering Operations and
Management of New Protocols and Protocol Extensions",
RFC 5706, DOI 10.17487/RFC5706, November 2009,
<<https://www.rfc-editor.org/info/rfc5706>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection
(BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010,
<<https://www.rfc-editor.org/info/rfc5880>>.

Authors' Addresses

Zhenbin Li
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: lizhenbin@huawei.com

Sam Aldrin
Google, Inc

Email: aldrin.ietf@gmail.com

Jeff Tantsura
Nuage Networks

Email: jefftant.ietf@gmail.com

Greg Mirsky
ZTE Corp.

Email: gregimirsky@gmail.com

Shunwan Zhuang
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: zhuangshunwan@huawei.com

Ketan Talaulikar
Cisco Systems

Email: ketant@cisco.com

BFD Working Group
Internet-Draft
Intended status: Informational
Expires: 8 September 2022

G. Mirsky
Ericsson
7 March 2022

BFD in Demand Mode over Point-to-Point MPLS LSP
draft-mirsky-bfd-mpls-demand-11

Abstract

This document describes procedures for using Bidirectional Forwarding Detection (BFD) in Demand mode to detect data plane failures in Multiprotocol Label Switching (MPLS) point-to-point Label Switched Paths.

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 <https://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 8 September 2022.

Copyright Notice

Copyright (c) 2022 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
2. Conventions used in this document	2
2.1. Terminology	2
3. Use of the BFD Demand Mode	2
3.1. The Applicability of BFD for Multipoint Networks	4
4. IANA Considerations	4
5. Security Considerations	4
6. Normative References	4
7. Informative References	5
Appendix A. Acknowledgements	5
Author's Address	6

1. Introduction

[RFC5884] defined use of the Asynchronous method of Bidirectional Detection (BFD) [RFC5880] to monitor and detect failures in the data path of a Multiprotocol Label Switching (MPLS) Label Switched Path (LSP). Use of the Demand mode, also specified in [RFC5880], has not been defined so far. This document describes procedures for using the Demand mode of BFD protocol to detect data plane failures in MPLS point-to-point (p2p) LSPs.

2. Conventions used in this document

2.1. Terminology

MPLS: Multiprotocol Label Switching

LSP: Label Switched Path

LER: Label switching Edge Router

BFD: Bidirectional Forwarding Detection

p2p: Point-to-Point

3. Use of the BFD Demand Mode

[RFC5880] defines that the Demand mode may be:

- * asymmetric, i.e. used in one direction of a BFD session;
- * switched to and from without bringing BFD session to Down state through using a Poll Sequence.

For the case of BFD over MPLS LSP, ingress Label switching Edge Router (LER) usually acts as Active BFD peer and egress LER acts as Passive BFD peer. The Active peer bootstraps the BFD session by using LSP ping. If the BFD session is configured to use the Demand mode, once the BFD session is in Up state the ingress LER switches to the Demand mode as defined in Section 6.6 [RFC5880]. The egress LER also follows procedures defined in Section 6.6 [RFC5880] and ceases further transmission of periodic BFD control packets to the ingress LER.

In this state BFD peers remains as long as the egress LER is in Up state. The ingress LER can periodically check continuity of a bidirectional path between the ingress and egress LERs by using the Poll Sequence, as described in Section 6.6 [RFC5880]. An implementation that supports using the Poll Sequence as the mechanism for bidirectional path continuity check must control the interval between consecutive Poll Sequences. The Rdefault value could be selected as 1 second.

If the Detection timer at the egress LER expires, the BFD system on LER sends BFD Control packet to the ingress LER with the Poll (P) bit set, Status (Sta) field set to the Down value, and the Diagnostic (Diag) field set to Control Detection Time Expired value. The egress LER periodically transmits these Control packets to the ingress LER until either it receives the valid for this BFD session control packet with the Final (F) bit set from the ingress LER or the defect condition clears and the BFD session state reaches Up state at the egress LER. An implementation that supports this specification provides control of the interval between consecutive Poll messages signaling the expiration of the Detection timer. The default value of the interval can be selected as 1 second.

The ingress LER transmits BFD Control packets over the MPLS LSP with the Demand (D) flag set at negotiated interval per [RFC5880], the greater of `bfd.DesiredMinTxInterval` and `bfd.RemoteMinRxInterval`, until it receives the valid BFD packet from the egress LER with the Poll (P) bit and the Diagnostic (Diag) field value Control Detection Time Expired. Reception of such BFD control packet by the ingress LER indicates that the monitored LSP has a failure and sending BFD control packet with the Final flag set to acknowledge failure indication is likely to fail. Instead, the ingress LER transmits the BFD Control packet to the egress LER over the IP network with:

- * destination IP address is set to the destination IP address of the LSP Ping Echo request message [RFC8029];
- * destination UDP port set to 4784 [RFC5883];

- * Final (F) flag in BFD control packet is set;
- * Demand (D) flag in BFD control packet is cleared.

The ingress LER changes the state of the BFD session to Down and changes rate of BFD Control packets transmission to one packet per second. The ingress LER in Down mode changes to Asynchronous mode until the BFD session comes to Up state once again. Then the ingress LER switches to the Demand mode.

3.1. The Applicability of BFD for Multipoint Networks

[RFC8562] defines the use of BFD in multipoint networks. This specification analyzes the case of p2p LSP. In that scenario, the ingress of the LSP acts as the MultipointHead, and the egress - as MultipointTail. The BFD state machines for MultipointHead, MultipointClient, and MultipointTail don't use the three-way handshakes for session establishment and teardown. As a result, the Init state is absent, and the session transitions to the Up state once the BFD session is administratively enabled. Hence, a BFD session over a p2p LSP, using principles of [RFC8562] or [RFC8563], can be established faster if the MultipointTail has been provisioned with the value of My Discriminator used by the MultipointHead for that BFD session. That value can be provided to the MultipointTail using different mechanisms, e.g., an extension to IGP. Description of mechanism to provide the value of My Discriminator used by the MultipointHead for the particular BFD session is outside the scope of this specification.

Unsolicited notification of the detected failure by the MultipointTail to the MultipointClient performs as described above for the case when the ingress BFD system switches the remote peer into the Demand mode.

4. IANA Considerations

TBD

5. Security Considerations

This document does not introduce new security aspects but inherits all security considerations from [RFC5880], [RFC5884], [RFC7726], [RFC8029], and [RFC6425].

6. Normative References

- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <<https://www.rfc-editor.org/info/rfc5883>>.
- [RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <<https://www.rfc-editor.org/info/rfc5884>>.
- [RFC6425] Saxena, S., Ed., Swallow, G., Ali, Z., Farrel, A., Yasukawa, S., and T. Nadeau, "Detecting Data-Plane Failures in Point-to-Multipoint MPLS - Extensions to LSP Ping", RFC 6425, DOI 10.17487/RFC6425, November 2011, <<https://www.rfc-editor.org/info/rfc6425>>.
- [RFC7726] Govindan, V., Rajaraman, K., Mirsky, G., Akiya, N., and S. Aldrin, "Clarifying Procedures for Establishing BFD Sessions for MPLS Label Switched Paths (LSPs)", RFC 7726, DOI 10.17487/RFC7726, January 2016, <<https://www.rfc-editor.org/info/rfc7726>>.
- [RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <<https://www.rfc-editor.org/info/rfc8029>>.
- [RFC8562] Katz, D., Ward, D., Pallagatti, S., Ed., and G. Mirsky, Ed., "Bidirectional Forwarding Detection (BFD) for Multipoint Networks", RFC 8562, DOI 10.17487/RFC8562, April 2019, <<https://www.rfc-editor.org/info/rfc8562>>.

7. Informative References

- [RFC8563] Katz, D., Ward, D., Pallagatti, S., Ed., and G. Mirsky, Ed., "Bidirectional Forwarding Detection (BFD) Multipoint Active Tails", RFC 8563, DOI 10.17487/RFC8563, April 2019, <<https://www.rfc-editor.org/info/rfc8563>>.

Appendix A. Acknowledgements

TBD

Author's Address

Greg Mirsky
Ericsson
Email: gregimirsky@gmail.com

BFD Working Group
Internet-Draft
Updates: 5798 (if approved)
Intended status: Standards Track
Expires: November 25, 2018

G. Mirsky
ZTE Corp.
J. Tantsura
May 24, 2018

Bidirectional Forwarding Detection (BFD) for Multi-point Networks and
Virtual Router Redundancy Protocol (VRRP) Use Case
draft-mirsky-bfd-p2mp-vrrp-use-case-02

Abstract

This document discusses use of Bidirectional Forwarding Detection (BFD) for multi-point networks to provide Virtual Router Redundancy Protocol (VRRP) with sub-second Master convergence and defines the extension to bootstrap point-to-multipoint BFD session.

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 <https://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 25, 2018.

Copyright Notice

Copyright (c) 2018 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 (<https://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	2
1.1. Conventions used in this document	2
1.1.1. Terminology	3
1.1.2. Requirements Language	3
2. Problem Statement	3
3. Applicability of p2mp BFD	3
3.1. Multipoint BFD Encapsulation	5
4. IANA Considerations	5
5. Security Considerations	5
6. Acknowledgements	5
7. Normative References	5
Authors' Addresses	6

1. Introduction

The [RFC5798] is the current specification of the Virtual Router Redundancy Protocol (VRRP) for IPv4 and IPv6 networks. VRRPv3 allows for faster switchover to a Backup router. Using such capability with software-based implementation of VRRP is may prove challenging. But it still may be possible to deploy VRRP and provide sub-second detection of Master router failure by Backup routers.

Bidirectional Forwarding Detection (BFD) [RFC5880] had been originally defined detect failure of point-to-point (p2p) paths: single-hop [RFC5881], multihop [RFC5883]. Single-hop BFD may be used to enable Backup routers to detect failure of the Master router within 100 msec or faster. [I-D.nitish-vrrp-bfd] demonstrates how, with some extensions to [RFC5798], that can be achieved.

[I-D.ietf-bfd-multipoint] extends [RFC5880] for multipoint and multicast networks, which is precisely characterizes deployment scenarios for VRRP over LAN segment. This document demonstrates how point-to-multipoint (p2mp) BFD can enable faster detection of Master failure and thus minimize service disruption in a VRRP domain. The document also defines the extension to VRRP [RFC5798] to bootstrap a VRRP Backup router to join in p2mp BFD session.

1.1. Conventions used in this document

1.1.1. Terminology

BFD: Bidirectional Forwarding Detection

p2mp: Pont-to-Multipoint

VRRP: Virtual Router Redundancy Protocol

1.1.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Problem Statement

A router may be part of several Virtual Router Redundancy groups, as Master in some and as Backup in others. Supporting sub-second mode for VRRPv3 [RFC5798] for all these roles without specialized support in data plane may prove to be very challenging. BFD already has many implementations based on HW that are capable to support multiple sub-second session concurrently.

3. Applicability of p2mp BFD

[I-D.ietf-bfd-multipoint] may provide the efficient and scaleable solution for fast-converging environment that uses default route rather than dynamic routing. Each redundancy group presents itself as p2mp BFD session with its Master being the root and Backup routers being tails of the p2mp BFD session. Figure 1 displays the extension of VRRP [RFC5798] to bootstrap tail of the p2mp BFD session. Master

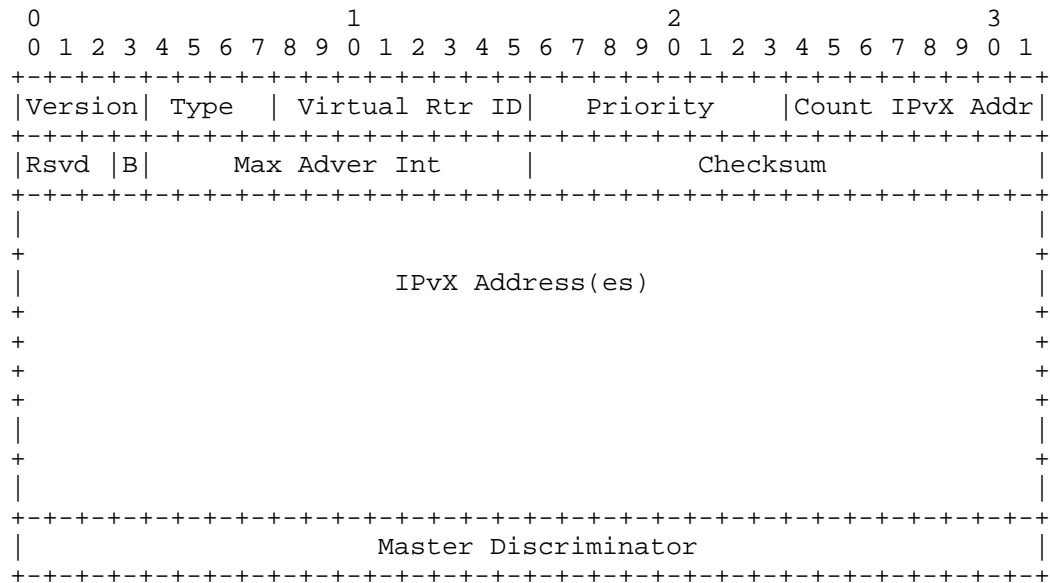


Figure 1: VRRP Extension to Bootstrap P2MP BFD session

where new fields are interpreted as:

B(FD) - one bit flag that indicates that the Master Discriminator field is appended to VRRP packet defined in [RFC5798];

Master Discriminator - My Discriminator value allocated by the root of the p2mp BFD session.

The Master router that is configured to use p2mp BFD to support faster convergence of VRRP starts transmitting BFD control packets with VRID as source IP address and My Discriminator. The same value of My Discriminator MUST be set as value of Master Discriminator field and BFD flag MUST be set in the VRRP packet. Backup router demultiplexes p2mp BFD test sessions based on VRID that it been configured with and the My Discriminator value it learns from the received VRRP packet. When a Backup router detects failure of the Master router it re-evaluates its role in the VRID. As result, the Backup router may become the Master router of the given VRID or continue as a Backup router. If the former is the case, then the new Master router MUST select My Discriminator and start transmitting p2mp BFD control packets using Master IP address as source IP address for p2mp BFD control packets. If the latter is the case, then the Backup router MUST wait for VRRP packet from the new VRRP Master router that will bootstrap new p2mp BFD session.

3.1. Multipoint BFD Encapsulation

The MultipointHead of p2mp BFD session when transmitting BFD control packet:

MUST set TTL value to 1 (though note that VRRP packets have TTL set to 255);

SHOULD use group address VRRP ('224.0.0.18' for IPv4 and 'FF02:0:0:0:0:0:0:12' for IPv6) as destination IP address

MAY use network broadcast address for IPv4 or link-local all nodes multicast group for IPv6 as destination IP address;

MUST set destination UDP port value to 3784 when transmitting BFD control packets, as defined in [I-D.ietf-bfd-multipoint];

MUST use Master IP address as source IP address.

4. IANA Considerations

This document makes no requests for IANA allocations. This section may be deleted by RFC Editor.

5. Security Considerations

Security considerations discussed in [RFC5798], [RFC5880], and [I-D.ietf-bfd-multipoint], apply to this document.

6. Acknowledgements

7. Normative References

[I-D.ietf-bfd-multipoint]

Katz, D., Ward, D., Networks, J., and G. Mirsky, "BFD for Multipoint Networks", draft-ietf-bfd-multipoint-16 (work in progress), April 2018.

[I-D.nitish-vrrp-bfd]

Gupta, N., Dogra, A., Docherty, C., Mirsky, G., and J. Tantsura, "Fast failure detection in VRRP with BFD", draft-nitish-vrrp-bfd-04 (work in progress), August 2016.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

- [RFC5798] Nadas, S., Ed., "Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6", RFC 5798, DOI 10.17487/RFC5798, March 2010, <<https://www.rfc-editor.org/info/rfc5798>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, DOI 10.17487/RFC5881, June 2010, <<https://www.rfc-editor.org/info/rfc5881>>.
- [RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <<https://www.rfc-editor.org/info/rfc5883>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

Authors' Addresses

Greg Mirsky
ZTE Corp.

Email: gregimirsky@gmail.com

Jeff Tantsura

Email: jefftant.ietf@gmail.com

MPLS Working Group
Internet-Draft
Intended status: Standards Track
Expires: 23 March 2022

G. Mirsky
Ericsson
G. Mishra
Verizon Inc.
D. Eastlake
Futurewei Technologies
19 September 2021

BFD for Multipoint Networks over Point-to-Multi-Point MPLS LSP
draft-mirsky-mpls-p2mp-bfd-15

Abstract

This document describes procedures for using Bidirectional Forwarding Detection (BFD) for multipoint networks to detect data plane failures in Multiprotocol Label Switching (MPLS) point-to-multipoint (p2mp) Label Switched Paths (LSPs) and Segment Routing (SR) point-to-multipoint policies with SR-MPLS data plane.

It also describes the applicability of LSP Ping, as in-band, and the control plane, as out-band, solutions to bootstrap a BFD session.

It also describes the behavior of the active tail for head notification.

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 <https://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 23 March 2022.

Copyright Notice

Copyright (c) 2021 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 (<https://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	2
2. Conventions used in this document	3
2.1. Terminology	3
2.2. Requirements Language	3
3. Multipoint BFD Encapsulation	3
3.1. IP Encapsulation of Multipoint BFD	4
3.2. Non-IP Encapsulation of Multipoint BFD	4
4. Bootstrapping Multipoint BFD	5
4.1. LSP Ping	5
4.2. Control Plane	6
5. Operation of Multipoint BFD with Active Tail over P2MP MPLS LSP	6
6. Security Considerations	7
7. IANA Considerations	8
8. Acknowledgements	8
9. References	8
9.1. Normative References	8
9.2. Informative References	10
Authors' Addresses	10

1. Introduction

[RFC8562] defines a method of using Bidirectional Detection (BFD) [RFC5880] to monitor and detect unicast failures between the sender (head) and one or more receivers (tails) in multipoint or multicast networks.

[RFC8562] added two BFD session types - MultipointHead and MultipointTail. Throughout this document, MultipointHead and MultipointTail refer to the value of the `bfd.SessionType` is set on a BFD endpoint.

This document describes procedures for using such modes of BFD protocol to detect data plane failures in Multiprotocol Label Switching (MPLS) point-to-multipoint (p2mp) Label Switched Paths (LSPs) and Segment Routing (SR) point-to-multipoint policies with SR-MPLS data plane

The document also describes the applicability of out-band solutions to bootstrap a BFD session in this environment.

It also describes the behavior of the active tail for head notification.

2. Conventions used in this document

2.1. Terminology

MPLS: Multiprotocol Label Switching

LSP: Label Switched Path

BFD: Bidirectional Forwarding Detection

p2mp: Point-to-Multipoint

FEC: Forwarding Equivalence Class

G-ACh: Generic Associated Channel

ACH: Associated Channel Header

GAL: G-ACh Label

LSR: Label Switching Router

SR: Segment Routing

SR-MPLS: SR with MPLS data plane

2.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Multipoint BFD Encapsulation

[RFC8562] uses BFD in the Demand mode from the very start of a point-to-multipoint (p2mp) BFD session. Because the head doesn't receive any BFD Control packet from a tail, the head of the p2mp BFD session transmits all BFD Control packets with the value of Your Discriminator field set to zero. As a result, a tail cannot demultiplex BFD sessions using Your Discriminator, as defined in

[RFC5880]. [RFC8562] requires that to demultiplex BFD sessions, the tail uses the source IP address, My Discriminator, and the identity of the multipoint tree from which the BFD Control packet was received. If the BFD Control packet is encapsulated in IP/UDP, then the source IP address MUST be used to demultiplex the received BFD Control packet as described in Section 3.1. The non-IP encapsulation case is described in Section 3.2.

3.1. IP Encapsulation of Multipoint BFD

[RFC8562] defines IP/UDP encapsulation for multipoint BFD over p2mp MPLS LSP:

- * UDP destination port MUST be set to 3784;
- * destination IP address MUST be set to the loopback address 127.0.0.1/32 for IPv4, or the loopback address ::1/128 for IPv6 [RFC4291]. Note that that is different from how the destination IP address selection is defined in Section 7 [RFC5884]. Firstly, because only one loopback address ::1/128 is defined in IPv6. And also, it is recommended to use the Entropy Label [RFC6790] to discover multiple alternate paths in an MPLS network. Using a single loopback address both for IPv4 and IPv6 encapsulation makes it consistent and more straightforward for an implementation.

The Motivation section [RFC6790] lists several advantages of generating the entropy value by an ingress Label Switching Router (LSR) compared to when a transit LSR infers entropy using the information in the MPLS label stack or payload. Thus this specification further clarifies that:

if multiple alternative paths for the given p2mp LSP Forwarding Equivalence Class (FEC) exist, the MultipointHead SHOULD use Entropy Label [RFC6790] used for LSP Ping [RFC8029] to exercise those particular alternative paths;

or the MultipointHead MAY use the UDP port number as discovered by LSP Ping traceroute [RFC8029] as the source UDP port number to possibly exercise those particular alternate paths.

3.2. Non-IP Encapsulation of Multipoint BFD

In some environments, the overhead of extra IP/UDP encapsulations may be considered burdensome, making the use of more compact G-ACh encapsulation attractive. Also, the validation of the IP/UDP encapsulation of a BFD Control packet in a p2mp BFD session may fail because of a problem related to neither the MPLS label stack nor to BFD. Avoiding unnecessary encapsulation of p2mp BFD over an MPLS LSP

improves the accuracy of the correlation of the detected failure and defect in MPLS LSP. Non-IP encapsulation for multipoint BFD over p2mp MPLS LSP MUST use Generic Associated Channel (G-ACH) Label (GAL) (see [RFC5586]) at the bottom of the label stack followed by an Associated Channel Header (ACH). If a BFD Control packet in PW-ACH encapsulation (without IP/UDP Headers) is to be used in ACH, an implementation would not be able to verify the identity of the MultipointHead and, as a result, will not properly demultiplex BFD packets. Hence, a new channel type value is needed. The Channel Type field in ACH MUST be set to TBA1 value Section 7. To provide the identity of the MultipointHead for the particular multipoint BFD session, a Source Address TLV [RFC7212] MUST immediately follow a BFD Control message.

4. Bootstrapping Multipoint BFD

4.1. LSP Ping

LSP Ping is the part of the on-demand OAM toolset used to detect and localize defects in the data plane and verify the control plane against the data plane by ensuring that the LSP is mapped to the same FEC at both egress and ingress endpoints.

LSP Ping, as defined in [RFC6425], MAY be used to bootstrap MultipointTail. If LSP Ping is used, it MUST include the Target FEC TLV and the BFD Discriminator TLV defined in [RFC5884]. For the case of p2mp MPLS LSP, the Target FEC TLV MUST use sub-TLVs defined in Section 3.1 [RFC6425]. For the case of p2mp SR policy with SR-MPLS data plane, an implementation of this specification MUST follow procedures defined in [RFC8287]. Setting the value of Reply Mode field to "Do not reply" [RFC8029] for the LSP Ping to bootstrap MultipointTail of the p2mp BFD session is RECOMMENDED. Indeed, because BFD over a multipoint network uses BFD Demand mode, the LSP echo reply from a tail has no useful information to convey to the head, unlike in the case of the BFD over a p2p MPLS LSP [RFC5884]. A MultipointTail that receives an LSP Ping that includes the BFD Discriminator TLV:

- * MUST validate the LSP Ping;
- * MUST associate the received BFD Discriminator value with the p2mp LSP;
- * MUST create a p2mp BFD session and set bfd.SessionType = MultipointTail as described in [RFC8562];

- * MUST use the source IP address of LSP Ping, the value of BFD Discriminator from the BFD Discriminator TLV, and the identity of the p2mp LSP to properly demultiplex BFD sessions.

Besides bootstrapping a BFD session over a p2mp LSP, LSP Ping SHOULD be used to verify the control plane against the data plane periodically by checking that the p2mp LSP is mapped to the same FEC at the MultipointHead and all active MultipointTails. The rate of generation of these LSP Ping Echo request messages SHOULD be significantly less than the rate of generation of the BFD Control packets because LSP Ping requires more processing to validate the consistency between the data plane and the control plane. An implementation MAY provide configuration options to control the rate of generation of the periodic LSP Ping Echo request messages.

4.2. Control Plane

The BGP-BFD Attribute [RFC9026] MAY be used to bootstrap multipoint BFD session on a tail.

5. Operation of Multipoint BFD with Active Tail over P2MP MPLS LSP

[RFC8562] defined how the BFD Demand mode can be used in multipoint networks. When applied in MPLS, procedures specified in [RFC8562] allow an egress LSR to detect a failure of the part of the MPLS p2mp LSP from the ingress LSR. The ingress LSR is not aware of the state of the p2mp LSP. [RFC8563], using mechanisms defined in [RFC8562], defined an "active tail" behavior. An active tail might notify the head of the detected failure and responds to a poll sequence initiated by the head. The first method, referred to as Head Notification without Polling, is mentioned in Section 5.2.1 [RFC8563], is the simplest of all described in [RFC8563]. The use of this method in BFD over MPLS p2mp LSP is discussed in this document. Analysis of other methods of a head learning of the state of an MPLS p2mp LSP is outside the scope of this document.

As specified in [RFC8563] for the active tail mode, BFD variables MUST be as follows:

On an ingress LSR:

- * bfd.SessionType is MultipointHead;
- * bfd.RequiredMinRxInterval is set to nonzero, allowing egress LSRs to send BFD Control packets.

On an egress LSR:

- * bfd.SessionType is MultipointTail;
- * bfd.SilentTail is set to zero.

In Section 5.2.1 [RFC8563] is noted that "the tail sends unsolicited BFD packets in response to the detection of a multipoint path failure" but without the specifics on the information in the packet and frequency of transmissions. This document defines below the procedure of an active tail with unsolicited notifications for p2mp MPLS LSP.

Upon detecting the failure of the p2mp MPLS LSP, an egress LSR sends BFD Control packet with the following settings:

- * the Poll (P) bit is set;
- * the Status (Sta) field set to Down value;
- * the Diagnostic (Diag) field set to Control Detection Time Expired value;
- * the value of the Your Discriminator field is set to the value the egress LSR has been using to demultiplex that BFD multipoint session;
- * BFD Control packet MAY be encapsulated in IP/UDP with the destination IP address of the ingress LSR and the UDP destination port number set to 4784 per [RFC5883]. If non-IP encapsulation is used, then a BFD Control packet is encapsulated using PW-ACH encapsulation (without IP/UDP Headers) (0x0007) [RFC5885];
- * these BFD Control packets are transmitted at the rate of one per second until either it receives a control packet valid for this BFD session with the Final (F) bit set from the ingress LSR or the defect condition clears; however to improve the likelihood of notifying the ingress LSR of the failure of the p2mp MPLS LSP, the egress LSR SHOULD initially transmit three BFD Control packets defined above in short succession.

An ingress LSR that has received the BFD Control packet, as described above, sends the unicast IP/UDP encapsulated BFD Control packet with the Final (F) bit set to the egress LSR.

6. Security Considerations

This document does not introduce new security aspects but inherits all security considerations from [RFC5880], [RFC5884], [RFC7726], [RFC8562], [RFC8029], and [RFC6425].

Also, BFD for p2mp MPLS LSP MUST follow the requirements listed in section 4.1 [RFC4687] to avoid congestion in the control plane or the data plane caused by the rate of generating BFD Control packets. An operator SHOULD consider the amount of extra traffic generated by p2mp BFD when selecting the interval at which the MultipointHead will transmit BFD Control packets. The operator MAY consider the size of the packet the MultipointHead transmits periodically as using IP/UDP encapsulation, which adds up to 28 octets, more than 50% of the BFD Control packet length, comparing to G-ACh encapsulation.

7. IANA Considerations

IANA is requested to allocate value (TBA1) from its MPLS Generalized Associated Channel (G-ACh) Types registry.

Value	Description	Reference
TBA1	Multipoint BFD Session	This document

Table 1: Multipoint BFD Session G-ACh Type

8. Acknowledgements

The authors sincerely appreciate the comments received from Andrew Malis, Italo Busi, Shraddha Hegde, and thought stimulating questions from Carlos Pignataro.

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5586] Bocci, M., Ed., Vigoureux, M., Ed., and S. Bryant, Ed., "MPLS Generic Associated Channel", RFC 5586, DOI 10.17487/RFC5586, June 2009, <<https://www.rfc-editor.org/info/rfc5586>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.

- [RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <<https://www.rfc-editor.org/info/rfc5883>>.
- [RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <<https://www.rfc-editor.org/info/rfc5884>>.
- [RFC5885] Nadeau, T., Ed. and C. Pignataro, Ed., "Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)", RFC 5885, DOI 10.17487/RFC5885, June 2010, <<https://www.rfc-editor.org/info/rfc5885>>.
- [RFC6425] Saxena, S., Ed., Swallow, G., Ali, Z., Farrel, A., Yasukawa, S., and T. Nadeau, "Detecting Data-Plane Failures in Point-to-Multipoint MPLS - Extensions to LSP Ping", RFC 6425, DOI 10.17487/RFC6425, November 2011, <<https://www.rfc-editor.org/info/rfc6425>>.
- [RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <<https://www.rfc-editor.org/info/rfc6790>>.
- [RFC7212] Frost, D., Bryant, S., and M. Bocci, "MPLS Generic Associated Channel (G-ACh) Advertisement Protocol", RFC 7212, DOI 10.17487/RFC7212, June 2014, <<https://www.rfc-editor.org/info/rfc7212>>.
- [RFC7726] Govindan, V., Rajaraman, K., Mirsky, G., Akiya, N., and S. Aldrin, "Clarifying Procedures for Establishing BFD Sessions for MPLS Label Switched Paths (LSPs)", RFC 7726, DOI 10.17487/RFC7726, January 2016, <<https://www.rfc-editor.org/info/rfc7726>>.
- [RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <<https://www.rfc-editor.org/info/rfc8029>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

- [RFC8287] Kumar, N., Ed., Pignataro, C., Ed., Swallow, G., Akiya, N., Kini, S., and M. Chen, "Label Switched Path (LSP) Ping/Traceroute for Segment Routing (SR) IGP-Prefix and IGP-Adjacency Segment Identifiers (SIDs) with MPLS Data Planes", RFC 8287, DOI 10.17487/RFC8287, December 2017, <<https://www.rfc-editor.org/info/rfc8287>>.
- [RFC8562] Katz, D., Ward, D., Pallagatti, S., Ed., and G. Mirsky, Ed., "Bidirectional Forwarding Detection (BFD) for Multipoint Networks", RFC 8562, DOI 10.17487/RFC8562, April 2019, <<https://www.rfc-editor.org/info/rfc8562>>.
- [RFC8563] Katz, D., Ward, D., Pallagatti, S., Ed., and G. Mirsky, Ed., "Bidirectional Forwarding Detection (BFD) Multipoint Active Tails", RFC 8563, DOI 10.17487/RFC8563, April 2019, <<https://www.rfc-editor.org/info/rfc8563>>.

9.2. Informative References

- [RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <<https://www.rfc-editor.org/info/rfc4291>>.
- [RFC4687] Yasukawa, S., Farrel, A., King, D., and T. Nadeau, "Operations and Management (OAM) Requirements for Point-to-Multipoint MPLS Networks", RFC 4687, DOI 10.17487/RFC4687, September 2006, <<https://www.rfc-editor.org/info/rfc4687>>.
- [RFC9026] Morin, T., Ed., Kebler, R., Ed., and G. Mirsky, Ed., "Multicast VPN Fast Upstream Failover", RFC 9026, DOI 10.17487/RFC9026, April 2021, <<https://www.rfc-editor.org/info/rfc9026>>.

Authors' Addresses

Greg Mirsky
Ericsson

Email: gregimirsky@gmail.com

Gyan Mishra
Verizon Inc.

Email: gyan.s.mishra@verizon.com

Donald Eastlake, 3rd
Futurewei Technologies
2386 Panoramic Circle
Apopka, FL 32703
United States of America

Email: d3e3e3@gmail.com

SPRING Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 28, 2020

G. Mirsky
ZTE Corp.
J. Tantsura
Apstra, Inc.
I. Varlashkin
Google
M. Chen
Huawei
J. Wenying
CMCC
April 26, 2020

Bidirectional Forwarding Detection (BFD) in Segment Routing Networks
Using MPLS Dataplane
draft-mirsky-spring-bfd-10

Abstract

Segment Routing (SR) architecture leverages the paradigm of source routing. It can be realized in the Multiprotocol Label Switching (MPLS) network without any change to the data plane. A segment is encoded as an MPLS label, and an ordered list of segments is encoded as a stack of labels. Bidirectional Forwarding Detection (BFD) is expected to monitor any existing path between systems. This document defines how to use Label Switched Path Ping to bootstrap a BFD session, control an SR Policy in the reverse direction of the SR-MPLS tunnel, and applicability of BFD Demand mode in the SR-MPLS domain. Also, the document describes the use of BFD Echo with BFD Control packet payload.

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 <https://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 October 28, 2020.

Copyright Notice

Copyright (c) 2020 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 (<https://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	2
1.1. Conventions	3
1.1.1. Terminology	3
1.1.2. Requirements Language	3
2. Bootstrapping BFD Session over Segment Routed Tunnel with MPLS Data Plane	4
3. Use BFD Reverse Path TLV over Segment Routed MPLS Tunnel	5
4. Use Non-FEC Path TLV	5
5. BFD Reverse Path TLV over Segment Routed MPLS Tunnel with Dynamic Control Plane	7
6. Applicability of BFD Demand Mode in SR-MPLS Domain	7
7. Using BFD to Monitor Point-to-Multipoint SR Policy	7
8. Use of Echo BFD in SR-MPLS	8
9. IANA Considerations	8
9.1. Non-FEC Path TLV	8
9.2. Return Code	9
10. Implementation Status	10
11. Security Considerations	10
12. Contributors	11
13. Acknowledgments	11
14. References	11
14.1. Normative References	11
14.2. Informative References	13
Authors' Addresses	13

1. Introduction

[RFC5880], [RFC5881], and [RFC5883] defined the operation of Bidirectional Forwarding Detection (BFD) protocol between the two systems over IP networks. [RFC5884] and [RFC7726] set rules for using BFD Asynchronous mode over point-to-point (p2p) Multiprotocol

Label Switching (MPLS) Label Switched Path (LSP). These latter standards implicitly assume that the remote BFD system, which is at the egress Label Edge Router (LER), will use the shortest path route regardless of the path the BFD system at the ingress LER uses to send BFD Control packets towards it. Throughout this document, references to ingress LER and egress LER are used, respectively, as a shortened version of the "BFD system at the ingress/egress LER".

This document defines the use of LSP Ping for Segment Routing networks over MPLS data plane [RFC8287] to bootstrap and control path of a BFD session from the egress to ingress LER using Segment Routing tunnel with MPLS data plane (SR-MPLS).

1.1. Conventions

1.1.1. Terminology

BFD: Bidirectional Forwarding Detection

BSID: Binding Segment Identifier

FEC: Forwarding Equivalence Class

MPLS: Multiprotocol Label Switching

SR-MPLS Segment Routing with MPLS data plane

LSP: Label Switched Path

LER Label Edge Router

p2p Point-to-point

p2mp Point-to-multipoint

SID Segment Identifier

SR Segment Routing

1.1.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Bootstrapping BFD Session over Segment Routed Tunnel with MPLS Data Plane

Use of an LSP Ping to bootstrap BFD over MPLS LSP is required, as documented in [RFC5884], to establish an association between a fault detection message, i.e., BFD Control message, and the Forwarding Equivalency Class (FEC) of a single label stack LSP in case of Penultimate Hop Popping or when the egress LER distributes the Explicit NULL label to the penultimate hop router. The Explicit NULL label is not advertised as a Segment Identifier (SID) by an SR node but, as demonstrated in section 3.1 [RFC8660] if the operation at the penultimate hop is NEXT; then the egress SR node will receive an IP encapsulated packet. Thus the conclusion is that LSP Ping MUST be used to bootstrap a BFD session in an SR-MPLS domain if there are no other means to bootstrap the BFD session, e.g., using an extension to a dynamic routing protocol as described in [I-D.ietf-bess-mvpn-fast-failover] and [I-D.ietf-pim-bfd-p2mp-use-case].

As demonstrated in [RFC8287], the introduction of Segment Routing network domains with an MPLS data plane requires three new sub-TLVs that MAY be used with Target FEC TLV. Section 6.1 addresses the use of the new sub-TLVs in Target FEC TLV in LSP ping and LSP traceroute. For the case of LSP ping, the [RFC8287] states that:

The initiator, i.e., ingress LER, MUST include FEC(s) corresponding to the destination segment.

The initiator MAY include FECs corresponding to some or all of segments imposed in the label stack by the ingress LER to communicate the segments traversed.

It has been noted in [RFC5884] that a BFD session monitors for defects particular <MPLS LSP, FEC> tuple. [RFC7726] clarified how to establish and operate multiple BFD sessions for the same <MPLS LSP, FEC> tuple. Because only the ingress LER is aware of the SR-based explicit route, the egress LER can associate the LSP ping with BFD Discriminator TLV with only one of the FECs it advertised for the particular segment. Thus this document clarifies that:

When LSP Ping is used to bootstrapping a BFD session for SR-MPLS tunnel the FEC corresponding to the segment to be associated with the BFD session MUST be as the very last sub-TLV in the Target FEC TLV.

If the target segment is an anycast prefix segment ([I-D.ietf-spring-mpls-anycast-segments]) the corresponding Anycast SID MUST be included in the Target TLV as the very last sub-TLV.

Also, for BFD Control packet the ingress SR node MUST use precisely the same label stack encapsulation, especially Entropy Label ([RFC6790]), as for the LSP ping with the BFD Discriminator TLV that bootstrapped the BFD session. Other operational aspects of using BFD to monitor the continuity of the path to the particular Anycast SID, advertised by a group of SR-MPLS capable nodes, will be considered in the future versions of the document.

Encapsulation of a BFD Control packet in Segment Routing network with MPLS data plane MUST follow Section 7 [RFC5884] when the IP/UDP header used and MUST follow Section 3.4 [RFC6428] without IP/UDP header being used.

3. Use BFD Reverse Path TLV over Segment Routed MPLS Tunnel

For BFD over MPLS LSP case, per [RFC5884], egress LER MAY send BFD Control packet to the ingress LER either over IP network or an MPLS LSP. Similarly, for the case of BFD over p2p SR-MPLS tunnel, the egress LER MAY route BFD Control packet over the IP network, as described in [RFC5883], or transmit over a segment tunnel, as described in Section 7 [RFC5884]. In some cases, there may be a need to direct egress LER to use a specific path for the reverse direction of the BFD session by using the BFD Reverse Path TLV and following all procedures as defined in [I-D.ietf-mpls-bfd-directed].

4. Use Non-FEC Path TLV

For the case of MPLS data plane, Segment Routing Architecture [RFC8402] explains that "a segment is encoded as an MPLS label. An ordered list of segments is encoded as a stack of labels."

This document defines a new optional Non-FEC Path TLV. The format of the Non-FEC Path TLV is presented in Figure 1

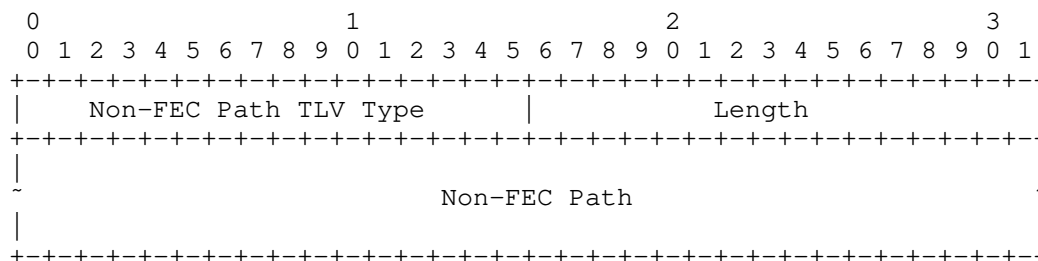


Figure 1: Non-FEC Path TLV Format

Non-FEC Path TLV Type is two octets in length and has a value of TBD1 (to be assigned by IANA as requested in Section 9.1).

Length field is two octets long and defines the length in octets of the Non-FEC Path field.

Non-FEC Path field contains a sub-TLV. Any Non-FEC Path sub-TLV (defined in this document or to be defined in the future) for Non-FEC Path TLV type MAY be used in this field. None or one sub-TLV MAY be included in the Non-FEC Path TLV. If no sub-TLV has been found in the Non-FEC Path TLV, the egress LER MUST revert to using the reverse path selected based on its local policy. If there is more than one sub-TLV, then the Return Code in echo reply MUST be set to value TBD3 "Too Many TLVs Detected" (to be assigned by IANA as requested in Table 4).

Non-FEC Path TLV MAY be used to specify the reverse path of the BFD session identified in the BFD Discriminator TLV. If the Non-FEC Path TLV is present in the echo request message the BFD Discriminator TLV MUST be present as well. If the BFD Discriminator TLV is absent when the Non-FEC Path TLV is included, then it MUST be treated as malformed Echo Request, as described in [RFC8029].

This document defines the Segment Routing MPLS Tunnel sub-TLV that MAY be used with the Non-FEC Path TLV. The format of the sub-TLV is presented in Figure 2.

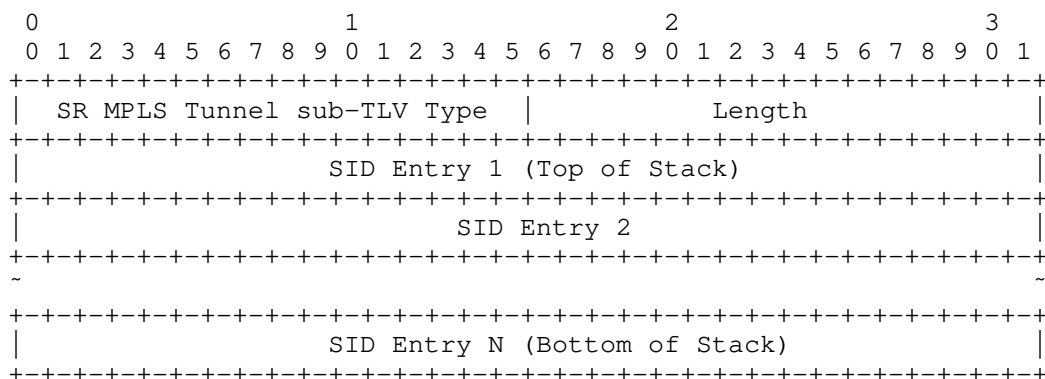


Figure 2: Segment Routing MPLS Tunnel sub-TLV

The Segment Routing MPLS Tunnel sub-TLV Type is two octets in length, and has a value of TBD2 (to be assigned by IANA as requested in Section 9.1).

The egress LER MUST use the Value field as label stack for BFD Control packets for the BFD session identified by the source IP

address of the MPLS LSP Ping packet and the value in the BFD Discriminator TLV. Label Entries MUST be in network order.

5. BFD Reverse Path TLV over Segment Routed MPLS Tunnel with Dynamic Control Plane

When Segment Routed domain with MPLS data plane uses distributed tunnel computation BFD Reverse Path TLV MAY use Target FEC sub-TLVs defined in [RFC8287].

6. Applicability of BFD Demand Mode in SR-MPLS Domain

[I-D.mirsky-bfd-mpls-demand] defines how Demand mode of BFD, specified in sections 6.6 and 6.18.4 of [RFC5880], can be used to monitor uni-directional MPLS LSP. Similar procedures can be following in SR-MPLS to monitor uni-directional SR tunnels:

- o an ingress SR node bootstraps BFD session over SR-MPLS in Async BFD mode;
- o once BFD session is Up, the ingress SR node switches the egress LER into the Demand mode by setting D field in BFD Control packet it transmits;
- o if the egress LER detects the failure of the BFD session, it sends its BFD Control packet to the ingress SR node over the IP network with a Poll sequence;
- o if the ingress SR node receives a BFD Control packet from the remote node in a Demand mode with Poll sequence and Diag field indicating the failure, the ingress SR node transmits BFD Control packet with Final over IP and switches the BFD over SR-MPLS back into Async mode, sending BFD Control packets one per second.

7. Using BFD to Monitor Point-to-Multipoint SR Policy

[I-D.voyer-spring-sr-p2mp-policy] defined variants of SR Policy to deliver point-to-multipoint (p2mp) services. For the given P2MP segment [RFC8562] can be used if, for example, leaves have an alternative source of the multicast service flow to select. In such a scenario, a leaf may switch to using the alternative flow after p2mp BFD detects the failure in the working multicast path. For scenarios where it is required for the root to monitor the state of the multicast tree [RFC8563] can be used. The root may use the detection of the failure of the multicast tree to the particular leaf to restore the path for that leaf or re-instantiate the whole multicast tree.

An essential part of using p2mp BFD is the bootstrapping the BFD session at all the leaves. The root, acting as the MultipointHead, MAY use LSP Ping with the BFD Discriminator TLV. Alternatively, extensions to routing protocols, e.g., BGP, or management plane, e.g., PCEP, MAY be used to associate the particular P2MP segment with MultipointHead's Discriminator. Extensions for routing protocols and management plane are for further study.

8. Use of Echo BFD in SR-MPLS

Echo-BFD [RFC5880] can be used to monitor an SR Policy between the local and the remote BFD peers. As defined in [RFC5880], the remote BFD system does not process the payload of an Echo BFD. Thus it is the local system that demultiplexes the Echo BFD packet matching it to the appropriate BFD session and detects missing Echo BFD packets. A BFD Control packet MAY be used as the payload of Echo BFD. This specification defines the use of Echo BFD in SR-MPLS network with BFD Control packet as the payload. The use of other types of Echo BFD payload is outside the scope of this document. Because the remote BFD system does not process Echo BFD, the value of the Your Discriminator field MUST be set to the discriminator the local BFD system assigned to the given BFD session. My Discriminator field MUST be zeroed. Authentication MUST be set according to the configuration of the BFD session. To ensure that the Echo BFD packet is returned to the sender without being processed, the sender MAY use a Binding SID (BSID) [RFC8402] that has been bound with the SR Policy that ensures the return of a packet to that particular node. A BSID MAY be associated with the SR Policy that is the reverse to the SR Policy programmed onto the BFD Echo packet by the sender.

9. IANA Considerations

9.1. Non-FEC Path TLV

IANA is requested to assign new TLV type from the from Standards Action range of the registry "Multiprotocol Label Switching Architecture (MPLS) Label Switched Paths (LSPs) Ping Parameters - TLVs" as defined in Table 1.

Value	TLV Name	Reference
TBD1	Non-FEC Path TLV	This document

Table 1: New Non-FEC Path TLV

IANA is requested to create new Non-FEC Path sub-TLV registry for the Non-FEC Path TLV, as described in Table 2.

Range	Registration Procedures	Note
0-16383	Standards Action	This range is for mandatory TLVs or for optional TLVs that require an error message if not recognized. Experimental RFC needed
16384-31743	Specification Required	
32768-49161	Standards Action	This range is for optional TLVs that can be silently dropped if not recognized. Experimental RFC needed
49162-64511	Specification Required	
64512-65535	Private Use	

Table 2: Non-FEC Path sub-TLV registry

IANA is requested to allocate the following values from the Non-FEC Path sub-TLV registry as defined in Table 3.

Value	Description	Reference
0	Reserved	This document
TBD2	Segment Routing MPLS Tunnel sub-TLV	This document
65535	Reserved	This document

Table 3: New Segment Routing Tunnel sub-TLV

9.2. Return Code

IANA is requested to create Non-FEC Path sub-TLV sub-registry for the new Non-FEC Path TLV and assign a new Return Code value from the "Multi-Protocol Label Switching (MPLS) Label Switched Paths (LSPs) Ping Parameters" registry, "Return Codes" sub-registry, as follows using a Standards Action value.

Value	Description	Reference
X TBD3	Too Many TLVs Detected.	This document

Table 4: New Return Code

10. Implementation Status

- The organization responsible for the implementation: ZTE Corporation.
- The implementation's name ROSng SW empowers traditional routers, e.g., ZXCTN 6000.
- A brief general description: A list of SIDs can be specified as the Return Path for an SR-MPLS tunnel.
- The implementation's level of maturity: production.
- Coverage: complete
- Version compatibility: draft-mirsky-spring-bfd-06.
- Licensing: proprietary.
- Implementation experience: Appreciate Early Allocation of values for Non-FEC TLV and Segment Routing MPLS Tunnel sub-TLV (using Private Use code points).
- Contact information: Qian Xin qian.xin2@zte.com.cn
- The date when information about this particular implementation was last updated: 12/16/2019

Note to RFC Editor: This section MUST be removed before publication of the document.

11. Security Considerations

Security considerations discussed in [RFC5880], [RFC5884], [RFC7726], and [RFC8029] apply to this document.

12. Contributors

Xiao Min
ZTE Corp.
Email: xiao.min2@zte.com.cn

13. Acknowledgments

Authors greatly appreciate the help of Qian Xin, who provided the information about the implementation of this specification.

14. References

14.1. Normative References

- [I-D.ietf-mpls-bfd-directed]
Mirsky, G., Tantsura, J., Varlashkin, I., and M. Chen,
"Bidirectional Forwarding Detection (BFD) Directed Return
Path", draft-ietf-mpls-bfd-directed-13 (work in progress),
December 2019.
- [I-D.mirsky-bfd-mpls-demand]
Mirsky, G., "BFD in Demand Mode over Point-to-Point MPLS
LSP", draft-mirsky-bfd-mpls-demand-06 (work in progress),
December 2019.
- [I-D.voyer-spring-sr-p2mp-policy]
daniel.voyer@bell.ca, d., Filsfils, C., Parekh, R.,
Bidgoli, H., and Z. Zhang, "SR Replication Policy for P2MP
Service Delivery", draft-voyer-spring-sr-p2mp-policy-03
(work in progress), July 2019.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119,
DOI 10.17487/RFC2119, March 1997,
<<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection
(BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010,
<<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection
(BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881,
DOI 10.17487/RFC5881, June 2010,
<<https://www.rfc-editor.org/info/rfc5881>>.

- [RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, DOI 10.17487/RFC5883, June 2010, <<https://www.rfc-editor.org/info/rfc5883>>.
- [RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <<https://www.rfc-editor.org/info/rfc5884>>.
- [RFC6428] Allan, D., Ed., Swallow, G., Ed., and J. Drake, Ed., "Proactive Connectivity Verification, Continuity Check, and Remote Defect Indication for the MPLS Transport Profile", RFC 6428, DOI 10.17487/RFC6428, November 2011, <<https://www.rfc-editor.org/info/rfc6428>>.
- [RFC7726] Govindan, V., Rajaraman, K., Mirsky, G., Akiya, N., and S. Aldrin, "Clarifying Procedures for Establishing BFD Sessions for MPLS Label Switched Paths (LSPs)", RFC 7726, DOI 10.17487/RFC7726, January 2016, <<https://www.rfc-editor.org/info/rfc7726>>.
- [RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <<https://www.rfc-editor.org/info/rfc8029>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8287] Kumar, N., Ed., Pignataro, C., Ed., Swallow, G., Akiya, N., Kini, S., and M. Chen, "Label Switched Path (LSP) Ping/Traceroute for Segment Routing (SR) IGP-Prefix and IGP-Adjacency Segment Identifiers (SIDs) with MPLS Data Planes", RFC 8287, DOI 10.17487/RFC8287, December 2017, <<https://www.rfc-editor.org/info/rfc8287>>.
- [RFC8402] Filss, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8562] Katz, D., Ward, D., Pallagatti, S., Ed., and G. Mirsky, Ed., "Bidirectional Forwarding Detection (BFD) for Multipoint Networks", RFC 8562, DOI 10.17487/RFC8562, April 2019, <<https://www.rfc-editor.org/info/rfc8562>>.

- [RFC8563] Katz, D., Ward, D., Pallagatti, S., Ed., and G. Mirsky, Ed., "Bidirectional Forwarding Detection (BFD) Multipoint Active Tails", RFC 8563, DOI 10.17487/RFC8563, April 2019, <<https://www.rfc-editor.org/info/rfc8563>>.
- [RFC8660] Bashandy, A., Ed., Filsfils, C., Ed., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with the MPLS Data Plane", RFC 8660, DOI 10.17487/RFC8660, December 2019, <<https://www.rfc-editor.org/info/rfc8660>>.

14.2. Informative References

- [I-D.ietf-bess-mvpn-fast-failover]
Morin, T., Kebler, R., and G. Mirsky, "Multicast VPN fast upstream failover", draft-ietf-bess-mvpn-fast-failover-10 (work in progress), February 2020.
- [I-D.ietf-pim-bfd-p2mp-use-case]
Mirsky, G. and J. Xiaoli, "Bidirectional Forwarding Detection (BFD) for Multi-point Networks and Protocol Independent Multicast - Sparse Mode (PIM-SM) Use Case", draft-ietf-pim-bfd-p2mp-use-case-03 (work in progress), January 2020.
- [I-D.ietf-spring-mpls-anycast-segments]
Sarkar, P., Gredler, H., Filsfils, C., Previdi, S., Decraene, B., and M. Horneffer, "Anycast Segments in MPLS based Segment Routing", draft-ietf-spring-mpls-anycast-segments-02 (work in progress), January 2018.
- [RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <<https://www.rfc-editor.org/info/rfc6790>>.

Authors' Addresses

Greg Mirsky
ZTE Corp.

Email: gregimirsky@gmail.com

Jeff Tantsura
Apstra, Inc.

Email: jefftant.ietf@gmail.com

Ilya Varlashkin
Google

Email: Ilya@nobulus.com

Mach (Guoyi) Chen
Huawei

Email: mach.chen@huawei.com

Jiang Wenying
CMCC

Email: jiangwenying@chinamobile.com