SPRING
Internet-Draft
Intended status: Informational
Expires: May 16, 2023

Z. Ali
K. Talaulikar
C. Filsfils
N. Nainar
C. Pignataro
Cisco Systems

November 16, 2022

Bidirectional Forwarding Detection (BFD) for Segment Routing Policies for Traffic Engineering
draft-ali-spring-bfd-sr-policy-10

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 (S-BFD) 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 (S-BFD) 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/.

This Internet-Draft will expire on May 16, 2023.

Table of Contents

1. Introduction

Segment Routing (SR) ([RFC8402]) 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.ietf-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.ietf-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 to rapidly detect any failures in the forwarding path 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 by eliminating the negotiation aspect and the need to maintain per session state entries on the tail end of the policy, thus providing benefits such as quick provisioning, as well as improved control and flexibility for network nodes initiating path monitoring. When BFD or S-BFD 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 S-BFD 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 S-BFD 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 S-BFD 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:

- 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]).

- SR Policies result in state being instantiated only on the head-end node and no other node in the network.

- 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.ietf-spring-segment-routing-policy] for details).

- SR Policies are expected to be deployed in much higher scale.

- 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 and on-demand. The S-BFD extensions provide this support as described in Seamless BFD [RFC7880]. Furthermore, S-BFD Use-Cases [RFC7882] clarifies the applicability in the Centralized TE and SR scenarios.

3. Procedures

The general procedures and mechanisms for S-BFD operations are specified in Seamless BFD [RFC7880]. This section describes the specifics related to S-BFD 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.

3.1. S-BFD Discriminator

In order to enable S-BFD monitoring for a given SR Policy, the S-BFD Discriminator for the tail-end node (i.e. one with the endpoint IP address) which is going to be the S-BFD Reflector is required. ISIS S-BFD [RFC7883] and OSPF S-BFD [RFC7884] describe the extensions to the ISIS and OSPF link state routing protocols that allow all nodes to advertise their S-BFD Discriminators across the network. BGP-LS S-BFD [I-D.ietf-idr-bgp-ls-sbfd-extensions] describes extensions for advertising the S-BFD 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 S-BFD Discriminator of the tail-end node of the SR Policy available.

When the end point IP address configured in the SR policy is IPv4, an implementation may support the use of end point address as the S-BFD Discriminator if SBFDReflector is enabled to associate the end point address as Discriminator for the target identifier.

The selection of S-BFD Discriminator from IGP or end point address is a local implementation matter and can be controlled by configuration knob.

3.2. S-BFD session Initiation by SBFDInitiator

The SRTE Process can straightaway instantiate the S-BFD 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 S-BFD Reflector. The return path used by S-BFD is via the normal IP routing back to the head-end node. Once the specific SR Policy path is verified via S-BFD, then it is considered as active and may be used for traffic steering.

The S-BFD 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:

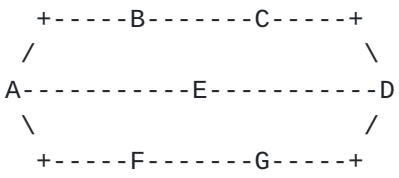
- Trigger path-protection for the SR Policy

- Declare the specific candidate path as invalid and switch to using the next valid candidate path based on preference

- 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).

3.3. Controlled Return Path

S-BFD response from SBFDResponder is IP routed and so the procedure defined in the above sections will receive the response through uncontrolled return path. S-BFD echo packets with relevant stack of segment ID can be used to control the return path.



Forward Paths: A-B-C-D
IP Return Paths: D-E-A

Figure 1: S-BFD Echo Example

Node A sending S-BFD control packets with segment stack {B, C, D} will cause S-BFD control packets to traverse the paths A-B-C-D in the forward direction. The response S-BFD control packets from node D back to node A will be IP routed and will traverse the paths D-E-A. The SBFDInitiator sending such packets can also send S-BFD echo packets with segment stack {B, C, D, C, A}. S-BFD echo packets will u-turn on node D and traverse the paths D-C-B-A. If required, the SBFDInitiator can possess multiple types of S-BFD echo packets, with each having varying return paths. In this particular example, the SBFDInitiator can be sending two types of S-BFD echo packets in addition to S-BFD control packets.

- S-BFD Control Packets
  - Segment Stack: {B, C, D}
  - Return Path: D->E->A
- S-BFD Echo packets #1

* Segment Stack: {B, C, D, C, A}

* Return Path: D->C->B->A

o S-BFD Echo packets #2

* Segment Stack: {B, C, D, G, A}

* Return Path: D->G->F->A

The SBFDInitiator can correlate the result of each packet type to determine the nature of the failure. One such example of failure correlation is described in the figure below.

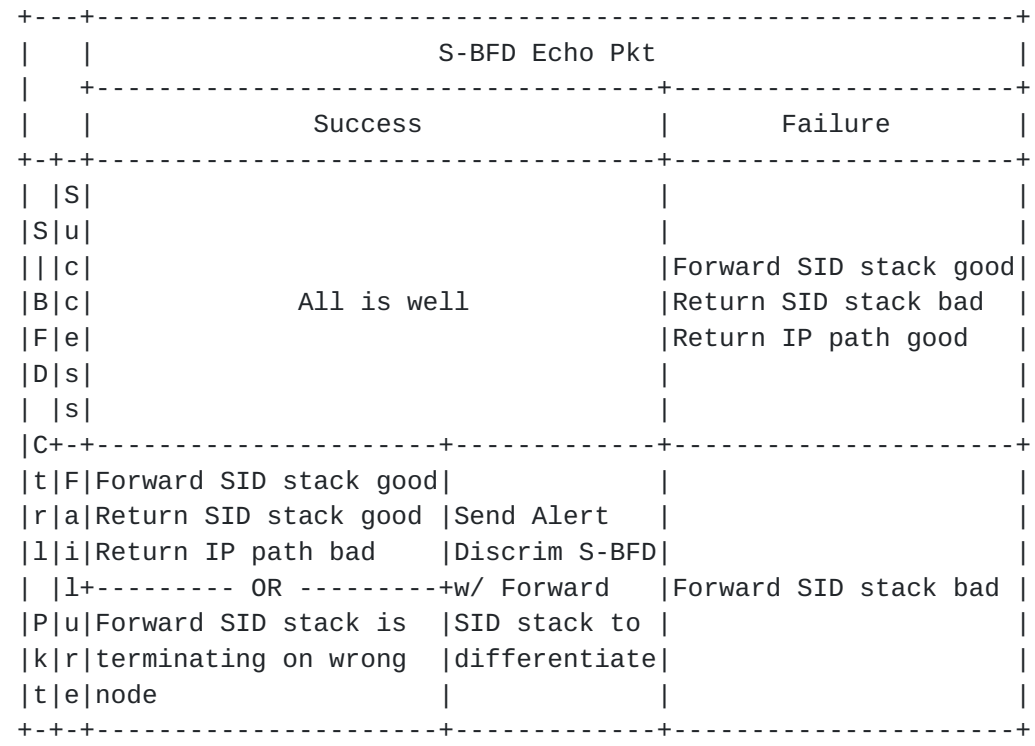


Figure 2: SBFDInitiator Failure Correlation Example

3.4. S-BFD Echo Recommendation

o It is RECOMMENDED to compute and use smallest number of segment stack to describe the return path of S-BFD echo packets to prevent the segment stack being too large. How SBFDInitiator determines when to use S-BFD echo packets and how to identify corresponding

segment stack for the return paths are outside the scope of this document.

- It is RECOMMENDED that SBFDInitiator does not send only S-BFD echo packets. S-BFD echo packets are crafted to traverse the network and to come back to self, thus there is no guarantee that S-BFD echo are u-turning on the intended remote target. On the other hand, S-BFD control packets can verify that segment stack of the forward direction reaches the intended remote target. Therefore, an SBFDInitiator SHOULD send S-BFD control packets when sending S-BFD echo packets.

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 S-BFD security and to [RFC8402] for analysis of security in SR deployments.

6. Contributors

Mallik Mudigonda
Cisco Systems Inc.

Email: mmudigon@cisco.com

7. Acknowledgements

8. References

8.1. Normative References

[I-D.ietf-idr-bgp-ls-sbfd-extensions]
Li, Z., Zhuang, S., Talaulikar, K., Aldrin, S., Tantsura, J., and G. Mirsky, "BGP Link-State Extensions for Seamless BFD", draft-ietf-idr-bgp-ls-sbfd-extensions-02 (work in progress).

[I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Sivabalan, S., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-07 (work in progress).

[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>.

[RFC8402] Filsfils, 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>.

8.2. Informative References

[I-D.ietf-idr-segment-routing-te-policy]
Previdi, S., Filsfils, C., Talaulikar, K., Mattes, P., Rosen, E., Jain, D., and S. Lin, "Advertising Segment Routing Policies in BGP", draft-ietf-idr-segment-routing-te-policy-08 (work in progress)

[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-16 (work in progress).

[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.ietf@gmail.com

Clarence Filsfils
Cisco Systems

Email: cfilsfil@cisco.com

Nagendra Kumar Nainar
Cisco Systems

Email: naikumar@cisco.com

Carlos Pignataro
Cisco Systems

Email: cpignata@cisco.com