BFD Working Group
Internet Draft
Intended status: Informational
Expires: July 22, 2026

C. Lin
New H3C Technologies
W. Cheng
W. Jiang
China Mobile
R. Chen
ZTE Corporation

January 22, 2026

BFD Path Consistency over SR
draft-lin-bfd-path-consistency-over-sr-06

Abstract

Bidirectional Forwarding Detection (BFD) can be used to monitor paths between nodes.

U-BFD defined in [I-D.ietf-bfd-unaffiliated-echo] can effectively reduce the device equipment.

Seamless BFD (S-BFD) provides a simplified mechanism which is suitable for monitoring of paths that are setup dynamically and on a large scale network.

In SR network, BFD can also be used to monitor SR paths. When a headend use BFD to monitor the segment list/CPath of SR Policy, the forward path of control packet is indicated by segment list, the reverse path of response control packet is via the shortest path from the reflector back to the initiator (headend) as determined by routing. The forward path and reverse path of control packet are likely inconsistent going through different intermediate nodes or links.

This document describes a method to keep the forward path and reverse path consistent when using S-BFD or U-BFD to detect SR Policy

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), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

This Internet-Draft will expire on July 22 2026.

Table of Contents

1. Introduction

Segment Routing (SR) allows a headend node to steer a packet flow along any path. Per-path states of Intermediate nodes are eliminated thanks to source routing. The headend node steers a flow into an SR Policy. The packets steered into an SR Policy carry an ordered list of segments associated with that SR Policy.

SR can be instantiated on the MPLS data plane (MPLS-SR) and the IPv6 data plane (SRv6). On the MPLS-SR data plane, a segment is encoded as an MPLS label, and an ordered list of segments is encoded as a stack of labels. On the SRv6 data plane, a segment is encoded as an IPv6 address (SRv6 SID) [RFC8986], and an ordered list of segments is encoded as an ordered list of SRv6 SIDs in the SR header (SRH) [RFC8754].

BFD Echo function was originally defined in [RFC5880] and [RFC5881], where the remote system is required to loop the BFD Echo packets back to the local system. To support BFD Echo Function, some negotiations between the local system and remote system are needed, and both the local and remote system need to maintain the BFD session state.

Unaffiliated BFD Echo Function (U-BFD) is defined in [I-D.ietf-bfd-unaffiliated-echo].Where the destination IP address of the BFD Echo packets is set to one of the IP addresses of the local system. Therefore, the Echo packets can be automatically looped back (through normal IP forwarding) by the remote system to the local system. With U-BFD, the remote system does not need to support any BFD related functions and maintain any session states. This further simplifies the BFD Echo Function process at the remote system hence greatly increases scalability.

Seamless BFD (S-BFD) defined in [RFC7880] provides a simplified mechanism which is suitable for monitoring of paths that are setup dynamically and on a large scale network.

In the SR network, the headend node could use BFD(S-BFD or U-BFD) to monitor the connectivity of the SR path to implement path switching. When a headend use BFD to monitor the segment list/CPath of SR Policy, the forward path of control packet is indicated by segment

list, the reverse path of response control packet is via the shortest path from the reflector back to the initiator (headend) as determined by routing. The forward path and reverse path of control packet are likely inconsistent going through different intermediate nodes or links.

The inconsistency impacts the detecting result. If the forward path is up and reverse path is down, then the BFD session will be down. If there are multiple path (segment list) in a SR Policy between a headend (local system) node and a tailend(remote system) node, multiple BFD session will be created for each path. Each BFD session uses corresponding path to send control packet, but the reverse path is identical for all BFD sessions. If the reverse path is down, all sessions will be down. Then the SR Policy is down.

The consistency of forward and reverse path of the same BFD session should be guaranteed.

This document describes how to ensure the consistency of the forward path and the reverse path when using BFD to detect SR Policy.

1.1. 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. Path consistency for BFD in SR network

Monitor SR Policy using BFD is usually based on segment list. BFD session is created for each segment list and associated with the segment list.

Referring to the following topology, there are two paths between Node A and D, and All nodes allocate end.x Segments on SRv6 data plane or adjacency SIDs on SR-MPLS data plane. Node A and D are headend and tailend nodes of each other, and SR policy is created on A and D respectively.

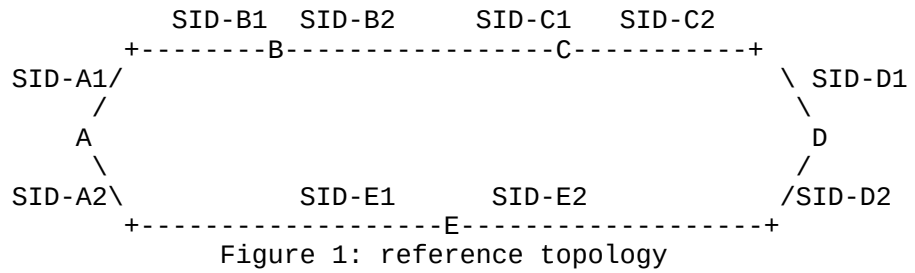


Figure 1: reference topology

Assuming that the deployed SR policy has one candidate path and each path has two segment lists. For ease of description, segment lists with the same number on Node A and D are forward and reverse paths to each other.

Node A:

SR Policy A-D
Candidate Path1
Segment list1
SID-A1, SID-B2, SID-C2
Segment list2
SID-A2, SID-E2

Node D:

SR Policy D-A
Candidate Path1
Segment list1
SID-D1, SID-C1, SID-B1
Segment list2
SID-D2, SID-E1

Both Node A and Node D serve as head nodes and need to detect the connectivity of the segment list of a SR Policy. Regardless of whether S-BFD or U-BFD is used, there is a requirement for BFD packet path consistency.

2.1. S-BFD

When node A is the S-BFD initiator, S-BFD sessions for segment list1 and segment list2 could be created respectively. Node A will use the associated segment list to encapsulate IPv6 header and SRH of the control packet.

As the S-BFD reflector, after Node D receives the S-BFD control packet, the response control packet should be able to return along the same path to avoid the false detection of the session caused by the inconsistency of the forward and reverse paths.

The control packet of S-BFD session associated with the segment list1 is forwarded to node D according to the segment list1 of node A. The response control packet of node D needs to be returned to

node A according to the segment list1 of node D. Thus the forward and reverse paths of S-BFD packets are ensured to be consistent.

2.2. U-BFD

The working mechanism of U-BFD is that the local system sends a bfd echo packet, and the destination address is the IP address of the local system. After the bfd echo packet reaches the remote system, the remote system returns the bfd echo packet to the local system in the data plane. So U-BFD usually works when there is only one hop between the local and remote systems.

When deploying U-BFD in SR network, local system could creates a U-BFD session for each segment list under the SR Policy, and uses the segment list to encapsulate BFD echo packets. For SR-MPLS encapsulation is the label stack, for SRv6 it is the segment list in SRH. In this way, the U-BFD echo packet can reach the remote system through multiple hops.

When the U-BFD echo packet reaches the remote system, the destination address of the packet has been updated to the IP address of the local system, so the remote system sends the U-BFD echo packet back to the local system on the data plane.

The U-BFD echo packet returned from remote system to local system should follow the same path from local system to remote system.

3. Path consistency for S-BFD

This draft proposes to forward S-BFD control packets and response control packets through the consistent path by path segment.

3.1. Correlate bidirectional path using Path Segment

A Path Segment is defined to identify an SR path. In SR for MPLS data plane (SR-MPLS), Path Segment is defined in [draft-ietf-spring-mpls-path-segment]. In SR for IPv6 data plane (SRv6), Path Segment is defined in [I-D.ietf-spring-srv6-path-segment].

SR(SR-MPLS or SRv6) Path segments can be used to correlate the two unidirectional SR paths at both ends of the paths.

[I-D.ietf-idr-sr-policy-path-segment] proposes an extension to BGP SR Policy distribute SR policies carrying Path Segment and bidirectional path information.

Through this extension, when distributing SR policy to the headend node, reverse path information and path segment of segment list can be carried together.

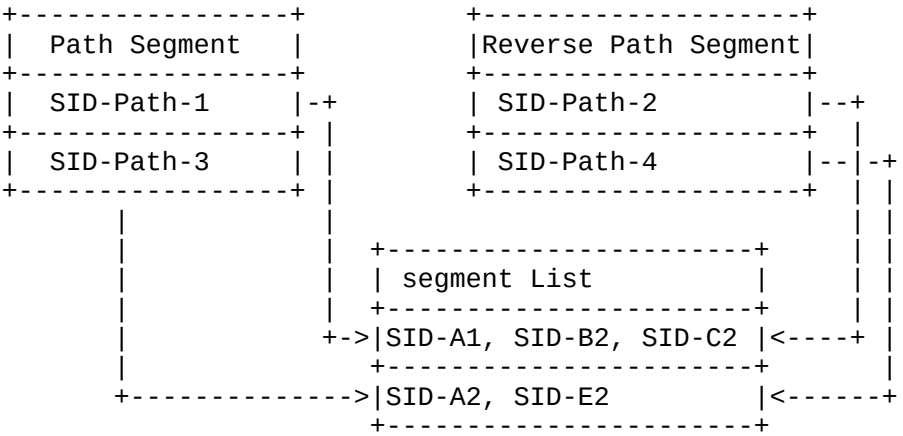
```
Node A                        Node D

SR Policy A-D                       SR Policy D-A
  Candidate Path1                     Candidate Path1
   Segment list1                       Segment list1
       SID-A1, SID-B2, SID-C2           SID-D1, SID-C1, SID-B1
       Path Segment: SID-Path-1        Path Segment: SID-Path-2
       Reverse Path Segment:            Reverse Path Segment:
          SID-Path-2                       SID-Path-1
   Segment list2                       Segment list2
       SID-A2, SID-E2                    SID-D2, SID-E1
       Path Segment: SID-Path-3        Path Segment: SID-Path-4
       Reverse Path Segment:             Reverse Path Segment:
         SID-Path-4                          SID-Path-3
```

In this way, on the headend node in both directions of the forward and reverse paths, the path segment of the paths in both directions can be obtained, and the paths in both directions use the same intermediate links.

The headend node can use path segment in two directions to establish a mapping table. Using this mapping table, the headend node can get the reverse path through the path segment of the forward path.

The mapping table of Node A and Node D is shown below:

Node A:



Node D:

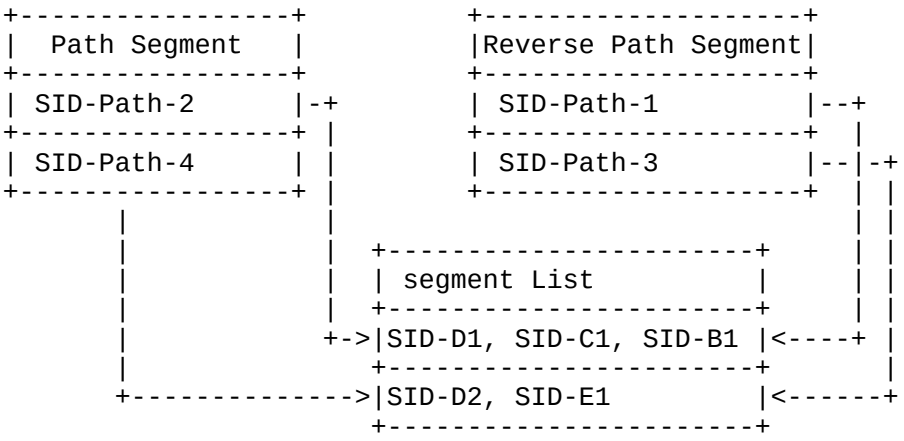


Figure 2: mapping table

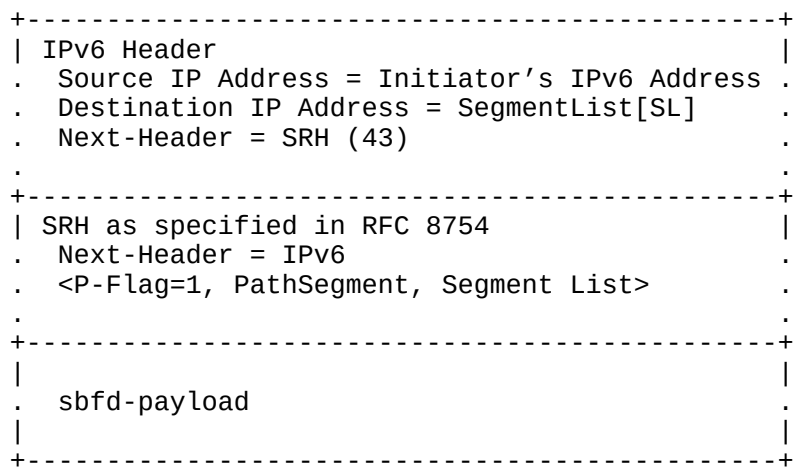
For instance, the S-BFD initiator is Node A in Figure 1, and the S-BFD session is bounded with Segment List1 of Policy A-D. The following sub-section describes the processing of S-BFD in SR-MPLS and SRv6 networks respectively.

3.2. Procedure of S-BFD

3.2.1. S-BFD in SRv6

o S-BFD Initiator procedure

Refer to [I-D.draft-liu-spring-bfd-srv6-policy-encap] for the description of how to encapsulate S-BFD packet with SRv6 Policy. When path segment is used, the encapsulation format of S-BFD control packet is as follows:



NodeA Encapsulates the path segment of segment list1 in SRH, and set SRH.P-Flag.

The S-BFD control packet is encapsulated and forwarded as follows:

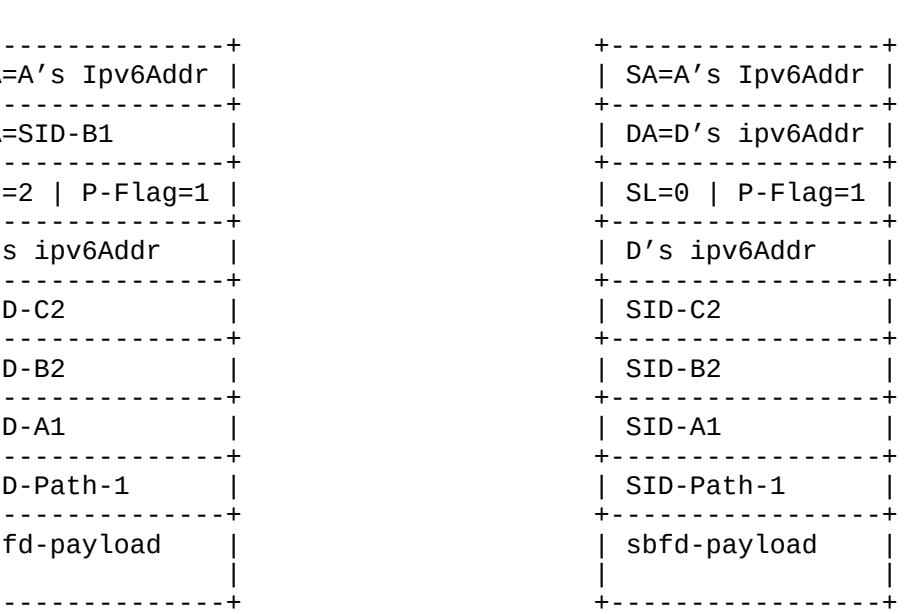


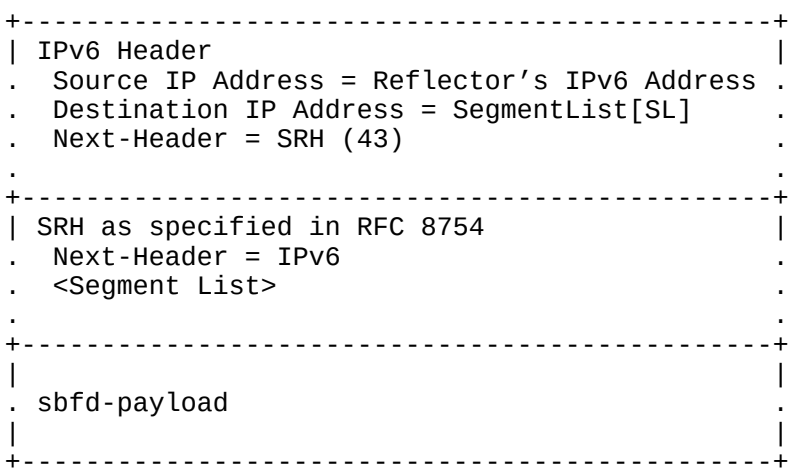
Figure 3: Example of S-BFD control packet

o S-BFD Reflector procedure

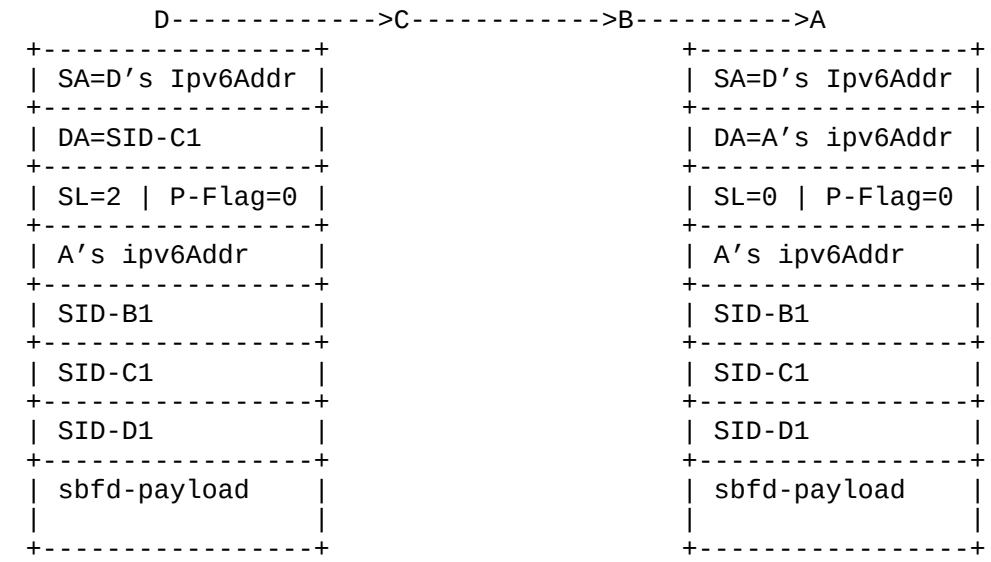
S-BFD control packet is forwarded along the path A->B->C-D. While packet arrives at Node D, SRH.SL is 0 and the destination address is IPv6 address of Node D. Packet is delivered up to the S-BFD module in control plane.

S-BFD module detects SRH.P-flag is set, extracts the path segment of the forward path from SRH, gets the path segment of the reverse path through the mapping table. When responding to S-BFD control packet, S-BFD module uses the segment list associated with path segment of the reverse path to encapsulate SRH.

The encapsulation format of S-BFD response control packet is as follows:



The S-BFD response packet is encapsulated and forwarded as follows:

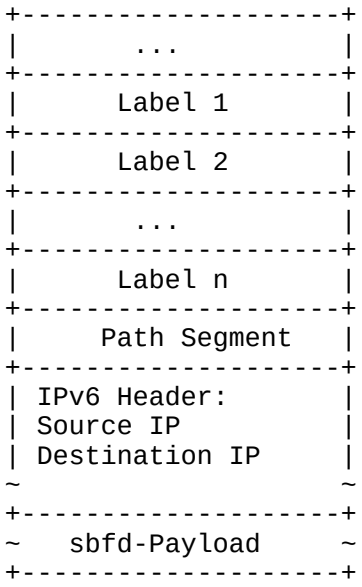


The S-BFD response control packet will be forward along the path D->C->B->A. In this way, the forward and reverse paths of S-BFD are guaranteed to be consistent.

3.2.2. S-BFD in SR-MPLS

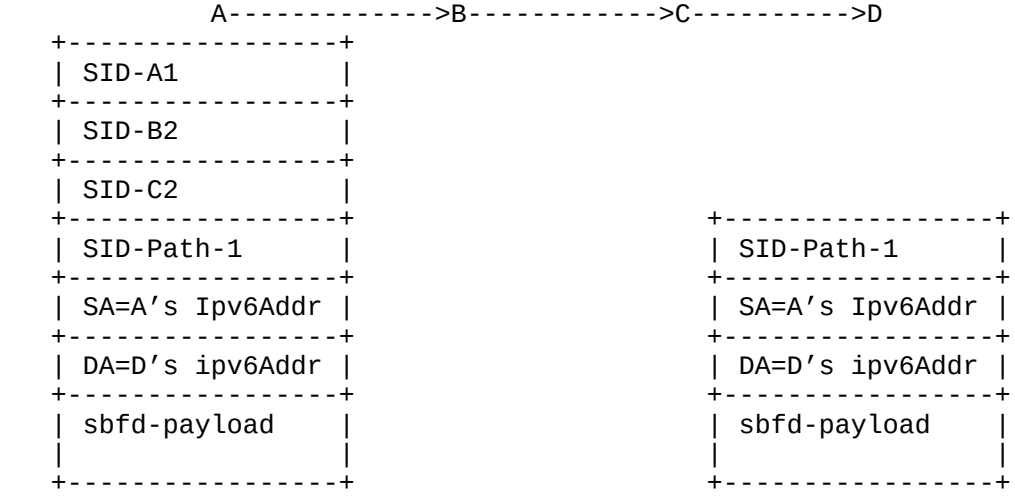
o S-BFD Initiator procedure

The encapsulation format using SR Policy with path segment of S-BFD control packet is as follows:



Node A Encapsulates the segment list1 and path segment in label stack. The source IP is the IPv6 address of Node A, and the destination IP is the IPv6 address of Node D.

The S-BFD control packet is encapsulated and forwarded as follows:



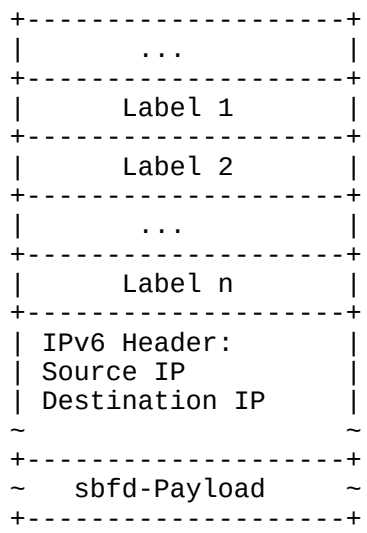
o S-BFD Reflector procedure

S-BFD control packet is forwarded along the path A->B->C-D. When the packet arrives at node D, the top-level label is path segment. Packet with path segment is delivered up to the S-BFD module in control plane.

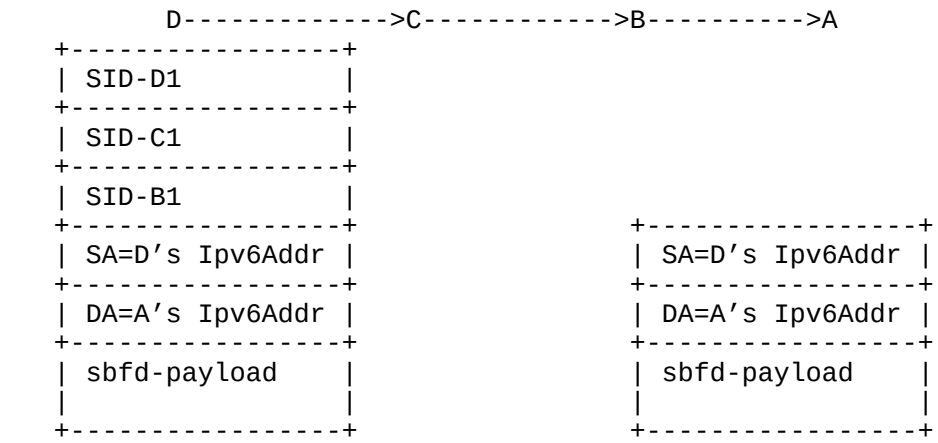
When responding to S-BFD control packet, the S-BFD module uses the mapping table to find the label stack of the reverse path through the path segment to encapsulate the response control packet.

The encapsulation format of S-BFD response control packet is as follows:

The source IP is the IPv6 address of Node D, and the destination IP is the IPv6 address of Node A.



The S-BFD response packet is encapsulated and forwarded as follows:



4. Path consistency for U-BFD

This document proposes to encapsulate the segment list of the return path in the U-BFD echo packet to guide the packet from the remote system to the local system along the same path.

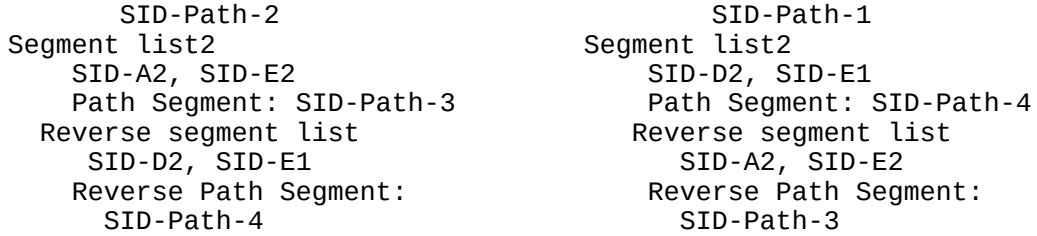
4.1. Getting reverse segment list

[I-D.ietf-idr-sr-policy-path-segment] proposes an extension to BGP SR Policy distribute SR policies carrying reverse path information.

The reverse path information includes reverse segment list and reverse path segment. The reverse path segment can be used for S-BFD path consistency, and the reverse segment list can be used for U-BFD path consistency.

Referring to the example topology, the SR Policy on nodes A and D that contains complete reverse path information is as follows

```
Node A                          Node D

SR Policy A-D                          SR Policy D-A
  Candidate Path1                        Candidate Path1
   Segment list1                          Segment list1
       SID-A1, SID-B2, SID-C2                 SID-D1, SID-C1, SID-B1
       Path Segment: SID-Path-1               Path Segment: SID-Path-2
    Reverse segment list                   Reverse Segment list
       SID-D1, SID-C1, SID-B1                 SID-A1, SID-B2, SID-C2
       Reverse Path Segment:                  Reverse Path Segment:
```



4.2. Procedure of U-BFD

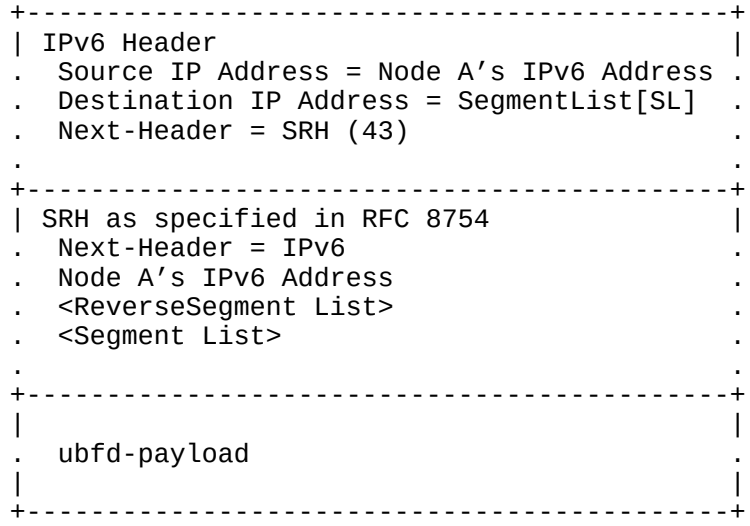
The headend node uses U-BFD to detect a segment list of SR-Policy. In order to achieve path consistency, the reverse segment list can be encapsulated in the U-BFD echo packet at the same time. When the U-BFD echo packet reaches the tailend node of SR-Policy, it will be looped back to the headend node according to the path specified by the reverse segment list.

According to this method, when the segment list in the SR-Policy and its corresponding reverse segment list are planned to pass through the same intermediate link, the U-BFD echo packet’s round-trip path will be consistent.

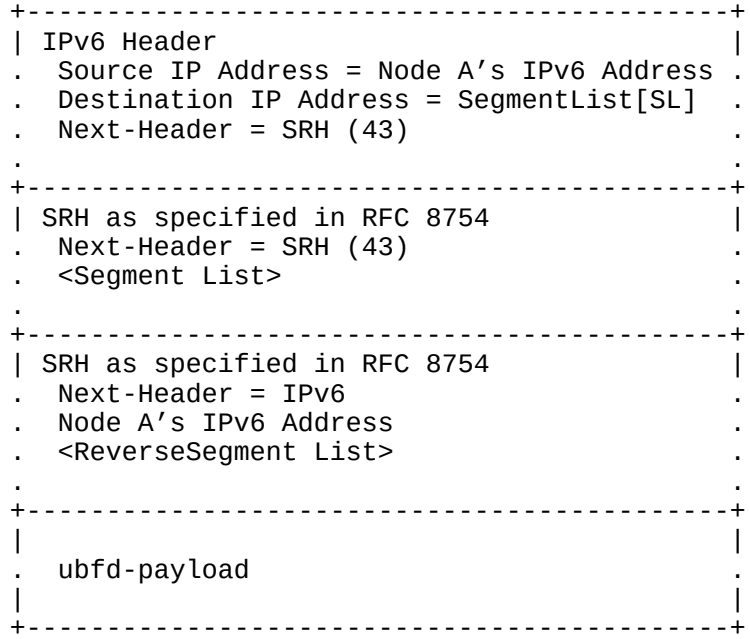
4.2.1. U-BFD in SRv6

In SRv6, the reverse segment list can be encapsulated in one SRH with the forward segment list, or it can be encapsulated in an independent SRH

When the forward and reverse segment lists are in the same SRH, the encapsulation is as follows



When the forward and reverse segment lists are in different SRHs, the encapsulation is as follows



Referring to the sample topology, take node A as the head node and D as the tail node as an example. Node A uses U-BFD to detect segment list1. The forward segment list and reverse segment list and the address of node A are encapsulated in one SRH.

The U-BFD echo packet is encapsulated and forwarded as follows:

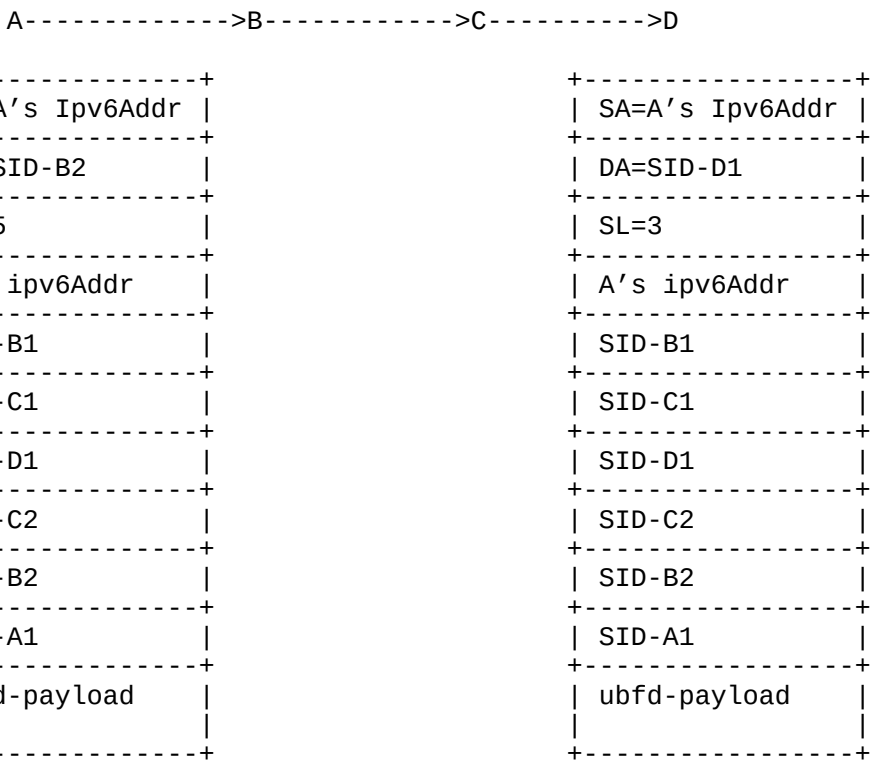
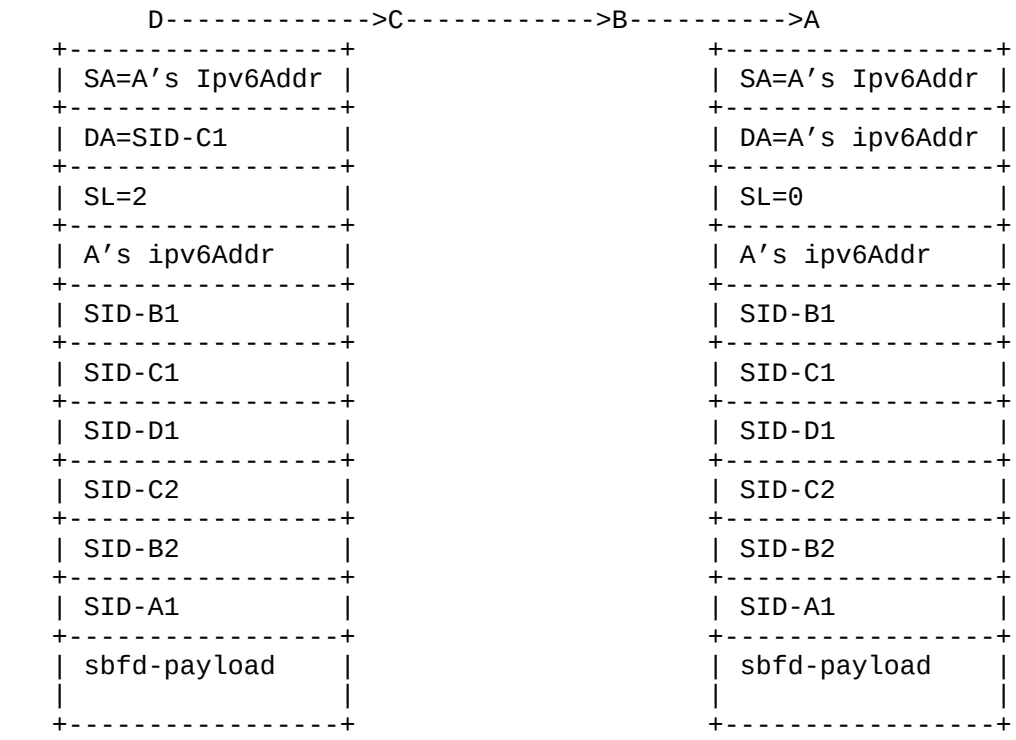


Figure 7: Example of U-BFD echo packet in SRv6

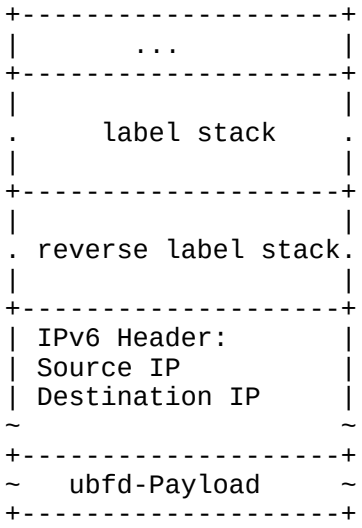
After the u-BFD packet reaches Node D, Node D continues to forward the u-BFD packet according to the SRH, and returns the U-BFD echo packet to Node A.



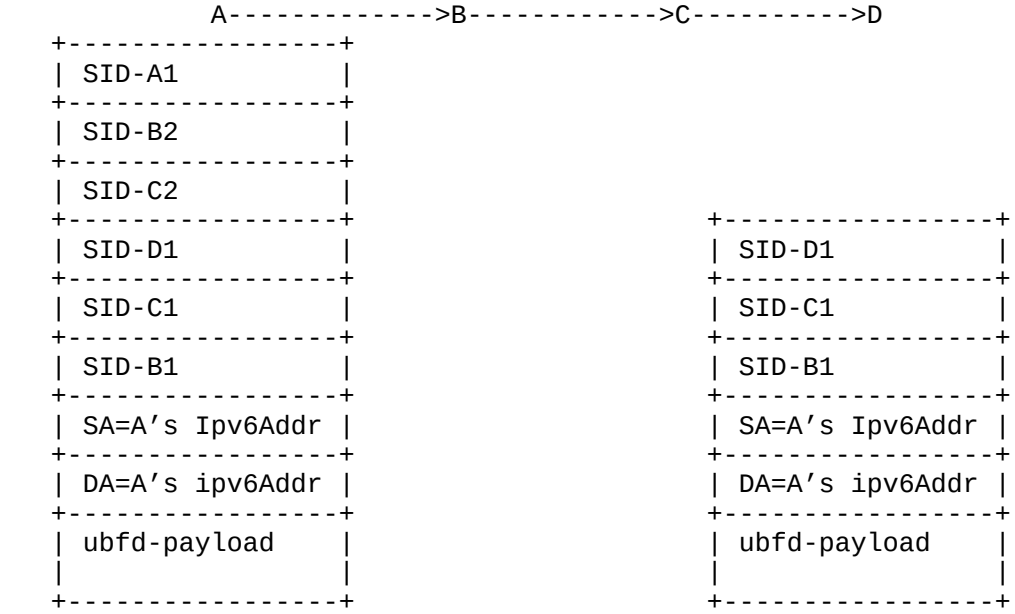
The U-BFD echo packet will be forward along the path D->C->B->A. In this way, the forward and reverse paths of U-BFD are guaranteed to be consistent.

4.2.2. U-BFD in SR-MPLS

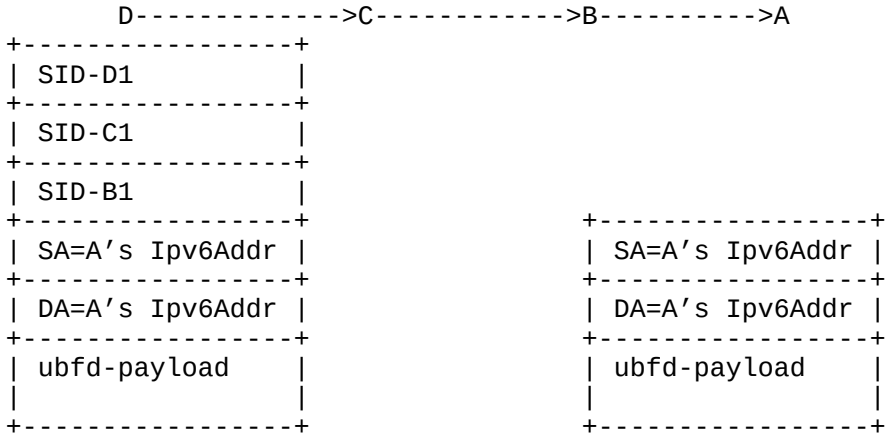
In SR-MPLS, The segment list and the reverse segment list can be encapsulated in the label stack at the same time.



Take node A as the headend node and D as the tailend node as an example, Node A uses U-BFD to detect segment list1. In order to achieve consistent paths, the encapsulation and processing of U-BFD echo packets are as follows



After the u-BFD packet reaches Node D, Node D continues to forward the u-BFD packet according to the label stack, and returns the U-BFD echo packet to Node A.



5. IANA Considerations

This document has no IANA actions.

6. Security Considerations

The security requirements and mechanisms described in [RFC8402] and [RFC8754] also apply to this document.

This document does not introduce any new security consideration.

7. References

7.1. Normative References

[I-D.draft-liu-spring-bfd-srv6-policy-encap] Liu, Y., Cheng, W., Lin, C., Chen, M., "Encapsulation of BFD for SRv6 Policy", draft-liu-spring-bfd-srv6-policy-encap-00(work in progress), October 2022

[I-D.ietf-bfd-unaffiliated-echo] Cheng, W., Wang, R., Min, X., Rahman, R., and R. C. Boddireddy, "Unaffiliated BFD Echo", Work in Progress, Internet-Draft, draft-ietf-bfd-unaffiliated-echo-06, 25 March 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-bfd-unaffiliated-echo-06>.

[I-D.ietf-idr-segment-routing-te-policy] Previdi, S., Filsfils, C., Talaulikar, K., Mattes, P., Jain, D., and S. Lin, "Advertising Segment Routing Policies in BGP", draft-ietf-idr-segment-routing-te-policy-20 (work in progress), July 2022

[I-D.ietf-spring-mpls-path-segment] Cheng, W., Li, H., Chen, M., Gandhi, R., and R. Zigler, "Path Segment in MPLS Based Segment Routing Network",draft-ietf-spring-mpls-path-segment-08 (work in progress), September 2022.

[I-D.ietf-spring-srv6-path-segment] Li, C., Cheng, W., Chen, M., Dhody, D., and Y. Zhu, "Path Segment for SRv6 (Segment Routing in IPv6)", draft-ietf-spring-srv6-path-segment-05 (work in progress),October 2022.

[I-D.ietf-idr-sr-policy-path-segment] Li, C., Li, Z., Yin, Y., Cheng, W., Talaulikar, K., "SR Policy Extensions for Path Segment and Bidirectional Path", draft-ietf-idr-sr-policy-path-segment-07(work in progress), February 2023.

[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>.

[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>.

[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>.

[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>.

[RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <https://www.rfc-editor.org/info/rfc8754>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 0.17487/RFC8986, February 2021, <https://www.rfc-editor.org/info/rfc8986>.

[RFC9252] Dawra, G., Ed., Talaulikar, K., Ed., Raszuk, R., Decraene, B., Zhuang, S., and J. Rabadan, "BGP Overlay Services Based on Segment Routing over IPv6 (SRv6)", RFC 9252, DOI 10.17487/RFC9252, July 2022, <https://www.rfc-editor.org/info/rfc9252>.

Contributors

Yisong Liu contributed to the content of this document.

Authors' Addresses

Changwang Lin
New H3C Technologies
Beijing
China

Email: linchangwang.04414@h3c.com

Weiqiang Cheng
China Mobile
Beijing
CN

Email: chengweiqiang@chinamobile.com

Wenying Jiang
China Mobile
Beijing
CN

Email: jiangwenying@chinamobile.com

Ran Chen
ZTE Corporation
China
Email: chen.ran@zte.com.cn