Workgroup: Network Working Group
Internet-Draft:
draft-ietf-spring-sr-replication-segment-09
Published: 6 October 2022
Intended Status: Standards Track
Expires: 9 April 2023
Authors: D. Voyer, Ed. C. Filsfils
Bell Canada Cisco Systems, Inc.
R. Parekh H. Bidgoli Z. Zhang
Cisco Systems, Inc. Nokia Juniper Networks

SR Replication Segment for Multi-point Service Delivery

Abstract

This document describes the SR Replication segment for Multi-point service delivery. A SR Replication segment allows a packet to be replicated from a Replication Node to Downstream nodes.

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 9 April 2023.

Copyright Notice

Table of Contents

1. Introduction

We define a new type of segment for Segment Routing [RFC8402], called Replication segment, which allows a node (henceforth called as Replication Node) to replicate packets to a set of other nodes (called Downstream Nodes) in a Segment Routing Domain. Replication segments provide building blocks for Point-to-Multipoint Service delivery via SR Point-to-Multipoint (SR P2MP) policy. A Replication segment can replicate packet to directly connected nodes or to downstream nodes (without need for state on the transit routers). This document focuses on the Replication segment building block. The use of one or more stitched Replication segments constructed for SR P2MP Policy tree is specified in [I-D.ietf-pim-sr-p2mp-policy].

2. Replication Segment

In a Segment Routing Domain, a Replication segment is a logical construct which connects a Replication Node to a set of Downstream Nodes. A Replication segment is a local segment instantiated at a Replication node. It can be either provisioned locally on a node or programmed by a PCE. Replication segments apply equally to both SR-MPLS and SRv6 instantiations of Segment Routing.

A Replication segment is identified by the tuple <Replication-ID, Node-ID>, where:

- *Replication-ID: An identifier for a Replication segment that is unique in context of the Replication Node.

- *Node-ID: The address of the Replication Node that the Replication segment is for. Note that the root of a multi-point service is also a Replication Node.

In simplest case, Replication-ID can be a 32-bit number, but it can be extended or modified as required based on specific use of a Replication segment. When the PCE signals a Replication segment to its node, the <Replication-ID, Node-ID> tuple identifies the segment. Examples of such signaling and extension are described in [I-D.ietf-pim-sr-p2mp-policy].

A Replication segment includes the following elements:

- *Replication SID: The Segment Identifier of a Replication segment. This is a SR-MPLS label or a SRv6 SID [RFC8402].

- *Downstream Nodes: Set of nodes in Segment Routing domain to which a packet is replicated by the Replication segment.

- *Replication State: See below.

The Downstream Nodes and Replication State of a Replication segment can change over time, depending on the network state and leaf nodes of a multi-point service that the segment is part of.

Replication SID identifies the Replication segment in the forwarding plane. At a Replication node, the Replication SID is the equivalent of Binding SID [RFC9256] of a Segment Routing Policy.

Replication State is a list of replication branches to the Downstream Nodes. In this document, each branch is abstracted to a <Downstream Node, Downstream Replication SID> tuple.

In a branch tuple, <Downstream Node> represents the reachability from the Replication Node to the Downstream Node. In its simplest form, this MAY be specified as an interface or nexthop if downstream node is adjacent to the Replication Node. The reachability may be specified in terms of Flex-Algo path (including the default algo) [I-D.ietf-lsr-flex-algo], or specified by an SR explicit path represented either by a SID-list (of one or more SIDs) or by a Segment Routing Policy [RFC9256].

A packet is steered into a Replication segment at a Replication Node in two ways:

- When the Active Segment [RFC8402] is a locally instantiated Replication SID

- By the root of a multi-point service based on local configuration outside the scope of this document.

In either case, the packet is replicated to each Downstream node in the associated Replication state.

If a Downstream Node is an egress (aka leaf) of the multi-point service, i.e. no further replication is needed, then that leaf node's Replication segment will not have any Replication State and the operation is NEXT. At an egress node, the Replication SID MAY be used to identify that portion of the multi-point service. Notice that the segment on the leaf node is still referred to as a Replication segment for the purpose of generalization.

A node can be a bud node, i.e. it is a Replication Node and a leaf node of a multi-point service at the same time [I-D.ietf-pim-sr-p2mp-policy].

2.1. SR-MPLS data plane

When the Active Segment is a Replication SID, the processing results in a POP operation and lookup of the associated Replication state. For each replication in the Replication state, the operation is a PUSH of the downstream Replication SID and an optional segment list on to the packet which is forwarded to the Downstream node. For leaf nodes the inner packet is forwarded as per local configuration.

When the root of a multi-point service steers a packet to a Replication segment, it results in a replication to each Downstream node in the associated replication state. The operation is a PUSH of the replication SID and an optional segment list on to the packet which is forwarded to the downstream node.

There MAY be SIDs preceding the SR-MPLS Replication SID in order to guide a packet from a non-adjacent SR node to a Replication Node. A Replication Node MAY replicate a packet to a non-adjacent Downstream Node using SIDs it inserts in the copy preceding the downstream Replication SID. The Downstream Node may be leaf node of the Replication Segment, or another Replication Node, or both in case of bud node. An Anycast SID or BGP PeerSID MUST NOT appear in segment list preceding a Replication SID. There MAY be SIDs after the Replication SID in the segment list of a packet. These SIDs are used to provide additional context for processing a packet locally at the node where the Replication SID is the Active Segment. The processing

of SIDs following the Replication SID MUST NOT forward the SR-MPLS packet to another node.

2.2. SRv6 data plane

In SRv6 [RFC8986], the "Endpoint with replication" behavior (End.Replicate for short) replicates a packet and forwards the packet according to a Replication state.

When processing a packet destined to a local Replication-SID, the packet is replicated to Downstream nodes and/or locally delivered off tree (when this is a bud/leaf node) according to the associated replication state. For replication, the outer header is re-used, and the Downstream Replication SID is written into the outer IPv6 header destination address. If required, an optional segment list may be used on some branches using H.Encaps.Red (while some other branches may not need that). Note that this H.Encaps.Red is independent from the replication segment - it is just used to steer the replicated traffic on a traffic engineered path to a Downstream node. If SRv6 SID compression is possible [I-D.ietf-spring-srv6-srh-compression], the Replication node SHOULD use a CSID container with Downstream Replication SID as the Last uSID in the container instead of H.Encaps.Red.

The above also applies when the Replication segment is for the Root node, whose upstream node has placed the Replication-SID in the header. A local application (e.g. MVPN/EVPN) may also apply H.Encaps.Red and then steer the resulting traffic into the segment. Again note that the H.Encaps.Red is independent of the Replication segment - it is the action of the application (e.g. MVPN/EVPN service). If the service is on a Root node, the two H.Encaps mentioned, one for the service and other in the previous paragraph for replication to Downstream node SHOULD be combined for optimization (to avoid extra IPv6 encapsulation).

For the local delivery on a bud/leaf node, the action associated with Replication-SID is "look at next SID in SRH". The next SID could be a SID with End.DTMC4/6 or End.DT2M local behavior (equivalent of MVPN/EVPN PMSI label in case of tunnel sharing across multiple VPNs). There may also not be a next SID (e.g. MVPN/EVPN with one tunnel per VPN), in which case the Replication-SID is then equivalent to End.DTM4/6 or End.DT2M. Note that decapsulation is not an inherent action of a Replication segment even on a bud/leaf node.

There MAY be SIDs preceding the SRv6 Replication SID in order to guide a packet from a non-adjacent SR node to a Replication Node via an explicit path. A Replication Node MAY steer a replicated packet on an explicit path to a non-adjacent Downstream Node using SIDs it inserts in the copy preceding the downstream Replication SID. The

Downstream Node may be leaf node of the Replication Segment, or another Replication Node, or both in case of bud node. For SRv6, as described in above paragraphs, the insertion of SIDs prior to Replication SID entails a new IPv6 encapsulation with SRH, but this can be optimized on Root node or for compressed SRv6 SIDs. Note that locator of Replication SID is sufficient to guide a packet on IGP shortest path, for default or Flex algo, between non-adjacent nodes. An Anycast SID or BGP PeerSID MUST NOT appear in segment list preceding a Replication SID. There MAY be SIDs after the Replication SID in the SRH of a packet. These SIDs are used to provide additional context for processing a packet locally at the node where the Replication SID is the Active Segment. The processing of SIDs following the Replication SID MUST NOT forward the SRv6 packet to some other node. The restrictions described in this paragraph apply to both un-compressed and compressed SRv6 encapsulation.

3. Use Cases

In the simplest use case, a single Replication segment includes the root node of a multi-point service and the egress/leaf nodes of the service as all the Downstream Nodes. This achieves Ingress Replication [RFC7988] that has been widely used for MVPN [RFC6513] and EVPN [RFC7432] BUM (Broadcast, Unknown and Multicast) traffic.

Replication segments can also be used as building blocks for replication trees when Replication segments on the root, intermediate Replication Nodes and leaf nodes are stitched together to achieve efficient replication. That is specified in [I-D.ietf-pim-sr-p2mp-policy].

4. IANA Considerations

This document requests IANA to allocate the following codepoints in "SRv6 Endpoint Behaviors" sub-registry of "Segment Routing Parameters" top-level registry.

Value	Hex	Endpoint behavior	Reference
75	0x004B	End.Replicate	[This.ID]

Table 1: IETF - SRv6 Endpoint Behaviors

5. Security Considerations

There are no additional security risks introduced by this design.

6. Acknowledgements

The authors would like to acknowledge Siva Sivabalan, Mike Koldychev, Vishnu Pavan Beeram, Alexander Vainshtein, Bruno

Decraene, Thierry Couture and Joel Halpern for their valuable inputs.

7. Contributors

Clayton Hassen Bell Canada Vancouver Canada

Email: clayton.hassen@bell.ca

Kurtis Gillis Bell Canada Halifax Canada

Email: kurtis.gillis@bell.ca

Arvind Venkateswaran Cisco Systems, Inc. San Jose US

Email: arvvenka@cisco.com

Zafar Ali Cisco Systems, Inc. US

Email: zali@cisco.com

Swadesh Agrawal Cisco Systems, Inc. San Jose US

Email: swaagraw@cisco.com

Jayant Kotalwar Nokia Mountain View US

Email: jayant.kotalwar@nokia.com

Tanmoy Kundu Nokia Mountain View US

Email: tanmoy.kundu@nokia.com

Andrew Stone Nokia Ottawa Canada

Email: andrew.stone@nokia.com

Tarek Saad Juniper Networks Canada

Email:tsaad@juniper.net

Kamran Raza Cisco Systems, Inc. Canada

Email:skraza@cisco.com

8. References

8.1. 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>.

[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>.

[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 10.17487/RFC8986, February 2021, <https://www.rfc-editor.org/info/rfc8986>.

[RFC9256] Filsfils, C., Talaulikar, K., Ed., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", RFC 9256, DOI 10.17487/RFC9256, July 2022, <https://www.rfc-editor.org/info/rfc9256>.

8.2. Informative References

[I-D.filsfils-spring-srv6-net-pgm-illustration] Filsfils, C., Garvia, P. C., Li, Z., Matsushima, S., Decraene, B., Steinberg, D., Lebrun, D., Raszuk, R., and J. Leddy, "Illustrations for SRv6 Network Programming", Work in Progress, Internet-Draft, draft-filsfils-spring-srv6-net-pgm-illustration-04, 30 March 2021, <https://www.ietf.org/archive/id/draft-filsfils-spring-srv6-net-pgm-illustration-04.txt>.

[I-D.ietf-lsr-flex-algo] Psenak, P., Hegde, S., Filsfils, C., Talaulikar, K., and A. Gulko, "IGP Flexible Algorithm", Work in Progress, Internet-Draft, draft-ietf-lsr-flex-algo-25, 6 October 2022, <https://www.ietf.org/archive/id/draft-ietf-lsr-flex-algo-25.txt>.

[I-D.ietf-pim-sr-p2mp-policy] (editor), D. V., Filsfils, C., Parekh, R., Bidgoli, H., and Z. Zhang, "Segment Routing Point-to-Multipoint Policy", Work in Progress, Internet-Draft, draft-ietf-pim-sr-p2mp-policy-05, 2 July 2022, <https://www.ietf.org/archive/id/draft-ietf-pim-sr-p2mp-policy-05.txt>.

[I-D.ietf-spring-srv6-srh-compression] Cheng, W., Filsfils, C., Li, Z., Decraene, B., Cai, D., Voyer, D., Clad, F., Zadok, S., Guichard, J. N., Aihua, L., Raszuk, R., and C. Li, "Compressed SRv6 Segment List Encoding in SRH", Work in Progress, Internet-Draft, draft-ietf-spring-srv6-srh-compression-02, 11 July 2022,

<https://www.ietf.org/archive/id/draft-ietf-spring-srv6-srh-compression-02.txt>.

[RFC6513] Rosen, E., Ed. and R. Aggarwal, Ed., "Multicast in MPLS/BGP IP VPNs", RFC 6513, DOI 10.17487/RFC6513, February 2012, <https://www.rfc-editor.org/info/rfc6513>.

[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, <https://www.rfc-editor.org/info/rfc7432>.

[RFC7988] Rosen, E., Ed., Subramanian, K., and Z. Zhang, "Ingress Replication Tunnels in Multicast VPN", RFC 7988, DOI 10.17487/RFC7988, October 2016, <https://www.rfc-editor.org/info/rfc7988>.

Appendix A. Illustration of a Replication Segment

This section illustrates an example of a single Replication segment. Examples showing Replication segment stitched together to form P2MP tree (based on SR P2MP policy) are in [I-D.ietf-pim-sr-p2mp-policy].

Consider the following topology:

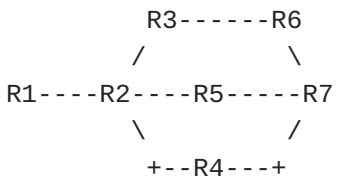

Figure 1: Figure 1

A.1. SR-MPLS

In this example, the Node-SID of a node Rn is N-SIDn and Adjacency-SID from node Rm to node Rn is A-SIDmn. Interface between Rm and Rn is Lmn.

Assume a Replication segment identified with R-ID at Replication Node R1 and downstream Nodes R2, R6 and R7. The Replication SID at node n is R-SIDn. A packet replicated from R1 to R7 has to traverse R4.

The Replication segment state at nodes R1, R2, R6 and R7 is shown below. Note nodes R3, R4 and R5 do not have state for the Replication segment.

Replication segment at R1:

```
Replication segment <R-ID,R1>:
 Replication SID: R-SID1
 Replication State:
   R2: <R-SID2->L12>
   R6: <N-SID6, R-SID6>
   R7: <N-SID4, A-SID47, R-SID7>
```

Replication to R2 steers packet directly to R2 on interface L12. Replication to R6, using N-SID6, steers packet via IGP shortest path to that node. Replication to R7 is steered via R4, using N-SID4 and then adjacency SID A-sID47 to R7.

Replication segment at R2:

```
Replication segment <R-ID,R2>:
 Replication SID: R-SID2
 Replication State:
   R2: <Leaf>
```

Replication segment at R6:

```
Replication segment <R-ID,R6>:
 Replication SID: R-SID6
 Replication State:
   R6: <Leaf>
```

Replication segment at R7:

```
Replication segment <R-ID,R7>:
 Replication SID: R-SID7
 Replication State:
   R7: <Leaf>
```

When a packet is steered into the Replication segment at R1:

- Since R1 is directly connected to R2, R1 performs PUSH operation with just <R-SID2> label for the replicated copy and sends it to R2 on interface L12. R2, as Leaf, performs NEXT operation, pops R-SID2 label and delivers the payload.

- R1 performs PUSH operation with <N-SID6, R-SID6> label stack for the replicated copy to R6 and sends it to R2, the nexthop on IGP shortest path to R6. R2 performs CONTINUE operation on N-SID6 and forwards it to R3. R3 is the penultimate hop for N-SID6; it performs penultimate hop popping, which corresponds to the NEXT operation and the packet is then sent to R6 with <R-SID6> in the label stack. R6, as Leaf, performs NEXT operation, pops R-SID6 label and delivers the payload.

*R1 performs PUSH operation with <N-SID4, A-SID47, R-SID7> label stack for the replicated copy to R7 and sends it to R2, the nexthop on IGP shortest path to R4. R2 is the penultimate hop for N-SID4; it performs penultimate hop popping, which corresponds to the NEXT operation and the packet is then sent to R4 with <A-SID47, R-SID1> in the label stack. R4 performs NEXT operation, pops A-SID47, and delivers packet to R7 with <R-SID7> in the label stack. R7, as Leaf, performs NEXT operation, pops R-SID7 label and delivers the payload.

A.2. SRv6

For SRv6 , we use SID allocation scheme, reproduced below, from Illustrations for SRv6 Network Programming [I-D.filsfils-spring-srv6-net-pgm-illustration]

*2001:db8::/32 is an IPv6 block allocated by a RIR to the operator

*2001:db8:0::/48 is dedicated to the internal address space

*2001:db8:cccc::/48 is dedicated to the internal SRv6 SID space

*We assume a location expressed in 64 bits and a function expressed in 16 bits

*Node k has a classic IPv6 loopback address 2001:db8::k/128 which is advertised in the IGP

*Node k has 2001:db8:cccc:k::/64 for its local SID space. Its SIDs will be explicitly assigned from that block

*Node k advertises 2001:db8:cccc:k::/64 in its IGP

*Function :1:: (function 1, for short) represents the End function with PSP support

*Function :Cn:: (function Cn, for short) represents the End.X function from to Node n

Each node k has:

*An explicit SID instantiation 2001:db8:cccc:k:1::/128 bound to an End function with additional support for PSP

*An explicit SID instantiation 2001:db8:cccc:k:Cj::/128 bound to an End.X function to neighbor J with additional support for PSP

*An explicit SID instantiation 2001:db8:cccc:k:Fk::/128 bound to an End.Replcate function

Assume a Replication segment identified with R-ID at Replication Node R1 and downstream Nodes R2, R6 and R7. The Replication SID at node k, bound to an End.Replcate function, is 2001:db8:cccc:k:Fk::/128. A packet replicated from R1 to R7 has to traverse R4.

The Replication segment state at nodes R1, R2, R6 and R7 is shown below. Note nodes R3, R4 and R5 do not have state for the Replication segment.

Replication segment at R1:

```
Replication segment <R-ID,R1>:
 Replication SID: 2001:db8:cccc:1:F1::0
 Replication State:
   R2: <2001:db8:cccc:2:F2::0->L12>
   R6: <2001:db8:cccc:6:F6::0>
   R7: <2001:db8:cccc:4:C7::0, 2001:db8:cccc:7:F7::0>
```

Replication to R2 steers packet directly to R2 on interface L12. Replication to R6, using 2001:db8:cccc:6:F6::0, steers packet via IGP shortest path to that node. Replication to R7 is steered via R4, using End.X SID 2001:db8:cccc:4:C7::0 at R4 to R7.

Replication segment at R2:

```
Replication segment <R-ID,R2>:
 Replication SID: 2001:db8:cccc:2:F2::0
 Replication State:
   R2: <Leaf>
```

Replication segment at R6:

```
Replication segment <R-ID,R6>:
 Replication SID: 2001:db8:cccc:6:F6::0
 Replication State:
   R6: <Leaf>
```

Replication segment at R7:

```
Replication segment <R-ID,R7>:
 Replication SID: 2001:db8:cccc:7:F7::0
 Replication State:
   R7: <Leaf>
```

When a packet, (A,B2), is steered into the Replication segment at R1:

*Since R1 is directly connected to R2, R1 creates encapsulated replicated copy (2001:db8::1, 2001:db8:cccc:2:F2::0) (A, B2), and

sends it to R2 on interface L12. R2, as Leaf, removes outer IPv6 header and delivers the payload.

*R1 creates encapsulated replicated copy (2001:db8::1, 2001:db8:cccc:6:F6::0) (A, B2) then forwards the resulting packet on the shortest path to 2001:db8:cccc:6::/64. R2 and R3 forward the packet using 2001:db8:cccc:6::/64. R6, as Leaf, removes outer IPv6 header and delivers the payload.

*R1 creates encapsulated replicated copy (2001:db8::1, 2001:db8:cccc:4:C7::0) (2001:db8:cccc:7:F7::0; SL=1) (A, B2) and sends it to R2, the nexthop on IGP shortest path to 2001:db8:cccc:4::/64. R2 forwards packet to R4 using 2001:db8:cccc:4::/64. R4 executes End.X function on 2001:db8:cccc:4:C7::0, performs PSP action, removes SRH and sends resulting packet (2001:db8::1, 2001:db8:cccc:7:F7::0) (A, B2) to R7. R7, as Leaf, removes outer IPv6 header and delivers the payload.

Authors' Addresses

Daniel Voyer (editor)
Bell Canada
Montreal
Canada

Email: daniel.voyer@bell.ca

Clarence Filsfils
Cisco Systems, Inc.
Brussels
Belgium

Email: cfilsfil@cisco.com

Rishabh Parekh
Cisco Systems, Inc.
San Jose,
United States of America

Email: riparekh@cisco.com

Hooman Bidgoli
Nokia
Ottawa
Canada

Email: hooman.bidgoli@nokia.com

Zhaohui Zhang

Juniper Networks

Email: zzhang@juniper.net