SPRING Working Group
Internet-Draft
Intended status: Standards Track
Expires: November 11, 2021

X. Geng
M. Chen
F. Yang, Ed.
Huawei Technologies
P. Camarillo
Cisco Systems, Inc.
G. Mishra
Verizon Inc.
May 10, 2021

Segment Routing for Redundancy Protection
draft-geng-spring-sr-redundancy-protection-03

Abstract

Redundancy protection provides a mechanism to achieve the high reliability of the service transmission in network. This document extends the capabilities in SR paradigm to support the redundancy protection in a DetNet environment, including the definitions of new Segments and a variation of Segment Routing Policy. The new mechanism applies equally to both Segment Routing with MPLS data plane (SR-MPLS) and Segment Routing with IPv6 data plane (SRv6).

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 .

Status of This Memo

Table of Contents

1. Introduction

Redundancy Protection provides a mechanism to achieve the high reliability of the service transmission in network. Specifically, packets of flows are replicated into two or more copies, which are transported through different paths in parallel. When copies of packets are merged at network node, the redundant packets are eliminated to guarantee one copy of the flow is successfully transmitted.

Redundancy protection targets to the services especially requires ultra reliable transmission, for example 5G URLLC services including Cloud VR/Game, remote surgery, harbor crane lifting, and differential protection in electrical utilities etc. Redundancy protection can also be used as Packet Replication and Elimination Function for Deterministic Networking defined in [RFC8655]. At last, it also bring values to the existing services in legacy network, for example IPTV service and financial private line in fixed broadband network, as well as the video service in data center interconnect.

Segment Routing (SR) leverages the source routing paradigm. An ingress node steers a packet through an ordered list of instructions, called "segments". A segment can be associated to an arbitrary processing of the packet in the node identified by the segment.

This document extends the capabilities in SR paradigm to support the redundancy protection in a DetNet environment, including the definitions of new Segments and a variation of Segment Routing Policy. The new mechanism applies equally to both Segment Routing with MPLS data plane (SR-MPLS) [RFC8660] and Segment Routing with IPv6 data plane (SRv6) [RFC8986].

2. Terminology

2.1. 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.

2.2. Terminology and Conventions

This document uses the terminology defined in [RFC8402] [RFC8964], and it also introduces the following new terms:

Redundancy Node: the start point of redundancy protection, which is a network device that could implement packet replication.

Merging Node: the end point of redundancy protection, which is a network node that could implement packet elimination.

Redundancy Policy: an extended SR policy which includes more than one active segment lists to support redundancy protection.

Flow Identification: information in the SR packet to indicate one flow.

Sequence Number: information in the SR packet to indicate the packet sequence of one flow.

Editor's Note: Similar mechanism is defined as "Service Protection" in the [RFC8655]. In this document, we define a new term "Redundancy Protection" to distinguish with other service protection method. Some of the terms are similar as [RFC8655].

3. Redundancy Protection in Segment Routing Scenario

```
                |                                         |
                |<------------- SRv6 Domain ------------->|
                |                                         |
                |                  +---+                  |
                |            +-----+R3 +-----+            |
                |            |     +---+     |            |
              +-+-+        +-+-+           +-+-+        +-+-+
       -------+R1 +--------+Red|           |Mer+--------+R2 +--------
              +---+        +-+-+           +-+-+        +---+
                             |     +---+     |
                             +-----+R4 +-----+
                                   +---+
```

Figure 1: Example Scenario of Redundancy Protection in SRv6 Domain

This figure shows an example of redundancy protection used in SRv6 domain. R1, R2, R3, R4, Red and Mer are SR-capable nodes. When a flow is sent into SRv6 domain, the process is:

1) R1 receives the traffic flow and encapsulates packets with a list of segments destined to R2, which is instantiated as an ordered list of SRv6 SIDs.

2) When the packet flow arrives in Red node, known as Redundancy Node, each packet is replicated into two or more copies. Each copy of the packet is encapsulated with a new segment list, which represents different forwarding paths (e.g., Disjoint Paths). The last SID in the segment list is always a SID instantiated on the Merging node (Mer) .

3) At the same time, the Flow Identification and Sequence Number are added to each of the replicas. Flow identification identifies the specific flow, and sequence number distinguishes the packet sequence of a flow.

4) The multiple replicas go through different paths until the Mer node. The first received packet of the flow is transmitted from Merging Node to R2, and the redundant packets are eliminated.

5) When there is any failures or packet loss in one path, the service continues undisrupted through the other path without break.

6) Sometimes, the packet will arrive out of order because of redundancy protection, the function of reordering may be also necessary on Merging Node. In such case the Merging node may include a reordering function. This is implementation specific and out of the scope of this document.

In this example, service protection is supported by utilizing two packet flows transmitted over two forwarding paths. It is noted that there is no limitation of the number of replicas. For a unidirectional flow, Red node supports replication function, and Mer node supports elimination function. Reordering function MAY be required in combination of elimination function on merging node. To minimize the jitter caused by random packet loss, the disjoint paths are recommended to have similar path forwarding delay.

4. Segment to Support Redundancy Protection

To achieve the packet replication and elimination functions, Redundancy Segment and Merging Segment, as well as the SR over MPLS forwarding behavior and SRv6 Endpoint Behavior are introduced.

4.1. Redundancy Segment

Redundancy Segment is a variation of Binding SID, and associated with a Redundancy Policy on the redundancy node. Redundancy segment is associated with service instructions, indicating the following operations:

o Steering the packet into the corresponding redundancy policy

o Flow identification and sequence number encapsulation in packets

o Packet replication and segment encapsulation based on the information of redundancy policy, e.g., the number of replication copies, a segment or an ordered list of segments with a topological instruction

4.1.1. SR over MPLS

In the case of SR over MPLS, IETF Deterministic Networking working group has defined Packet Replication/Elimination/Ordering Functions in DetNet MPLS data plane [RFC8964]. The support of redundancy protection in SR over MPLS data plane keeps consistent with the PRF and REF functions defined in DetNet MPLS data plane.

In SR over MPLS, Redundancy Segment acts as DetNet S-Label to explicitly identify the replication function on redundancy node. Redundancy segment is allocated from redundancy node, and is provisioned to the ingress node of SR domain by the controller plane via PCEP, BGP, or NetConf protocols.

When the Active Segment is a Redundancy Segment, a NEXT operation is performed and a redundancy policy is associated. Via redundancy policy, flow identification is assigned to redundancy node and acts as the elimination serivce label (S-Label). Sequence number is generated and encapsulated in DetNet Control Word (d-CW). The packets of a flow is replicated to multiple replicas, and encapsulated with a new MPLS label stack including the d-CW, S-Label and forwarding sub-layer LSPs, determined by the candidate paths of redundancy policy.

4.1.2. SRv6

In the case of SRv6, a new behavior End.R for Redundancy Segment is defined. An instance of a redundancy SID is associated with a redundancy policy B and a source address A. In the following description, End.R behavior is specified in the encapsulation mode. The End.R behavior in the insertion mode is for further study.

When an SRv6-capable node (N) receives an IPv6 packet whose destination address matches a local IPv6 address instantiated as an SRv6 SID (S), and S is a Redundancy SID, N does:

```
S01.When an SRH is processed {
S02.   If (Segments Left == 0) {
S03.        Remove the IPv6 header
S04    }
S05.   If (Segments Left>0)   {
S06.        Decrement IPv6 Hop Limit by 1
S07.        Decrement Segments Left by 1
S08.        Update IPv6 DA with Segment List[Segments Left]
S09.   }
S10.   Add flow identification and sequence number to packet
S11.   Duplicate the packets (as many replicas as active SID lists in B)
S12.   Push the new IPv6 headers to each replica. The IPv6 header
       contains an SRH with a different SID List
S13.   Set the outer IPv6 SA to A
S14.   Set the outer IPv6 DA to the first SID of new SRH SL
S15.   Set the outer Payload Length, Traffic Class, Flow Label,
       Hop Limit and Next-Header fields
S16.   Submit the packet to the egress IPv6 FIB lookup
       for transmission to the new destination
S17.}
```

4.2. Merging Segment

Merging Segment is associated with service instructions, indicates the following operations:

- o Packet merging and elimination: forward the first received packets and eliminate the redundant packets

In order to eliminate the redundant packet of a flow, merging node utilizes sequence number to evaluate the redundant status of a packet. Note that implementation specific mechanism could be applied to control the amount of state monitored on sequence number, so that system memory usage can be limited at a reasonable level.

As merging node needs to maintain the state of flows, a centralized controller should have a knowledge of merging nodes capability, and never provision the redundancy policy to redundancy node when the computation result goes beyond the flow recovery capability of merging node. The capability advertisement of merging node will be specified separately elsewhere, which is not within the scope of this document.

4.2.1. SR over MPLS

In the case of SR over MPLS, being consistent with [RFC8964], Merging Segment always stays at last of MPLS label stack as DetNet S-Label. When the Active Segment is a Merging Segment, a NEXT operation is performed. The packet is identified to a particular flow according to the service label. Sequence number encapsulated in DetNet control word is used to determine whether the packet is redundant.

4.2.2. SRv6

In the case of SRv6, a new behavior End.M for Merging Segment is defined.

When an SRv6-capable node (N) receives an IPv6 packet whose destination address matches a local IPv6 address instantiated as an SRv6 SID (S), and S is a Merging SID, N does:

```
When an SRH is processed {
 If (Segments Left==0)   {
   Acquire the sequence number of received packet and look it up
     If (state of this sequence number == 0) {
      Set the state of this sequence number to 1
      Remove the outer IPv6+SRH header
      Decrement IPv6 Hop Limit by 1 in inner SRH
      Decrement Segments Left by 1 in inner SRH
      Update IPv6 DA with Segment List[Segments Left] in inner SRH
      Submit the packet to the egress IPv6 FIB lookup and transmit
     }
     ELSE {
          Drop the packet
     }
   }
}
```

5. Meta Data to Support Redundancy Protection

To support the redundancy protection function, Flow Identification and Sequence Number are required. Flow identification identifies the specific flow with target of redundancy protection. Sequence number distinguishes the packets within a flow by specifying the order of packets. Thus, the encapsulation of flow identification and sequence number is required in both SR over MPLS and SRv6 data plane.

In SR over MPLS, being consistent with [RFC8964], flow identification is identified by either redundancy service or merging service, and is encapsulated as the DetNet service label. Note that, the DetNet service label can be different and swapped in the packet transmission. Sequence number is encapsulated in DetNet Control Word (d-CW).

In SRv6, flow identification and sequence number is added at the redundancy node and carried in the packets along the different paths to merging node. Merging node removes flow identifier and sequence number once the elimination and ordering (optional) is accomplished.

6. Segment Routing Policy to Support Redundancy Protection

Redundancy Policy is a variation of SR Policy, and is identified through the tuple <redundancy node, redundancy ID, merging node>. Redundancy node is specified as IPv4/IPv6 address of the headend, which is able to do packet replication. Merging node is specified as IPv4/IPv6 address of the endpoint, which is able to do packet elimination. Redundancy ID could be a specified value of "color" define in section 2.1 of [I-D.ietf-spring-segment-routing-policy], which indicates the SR policy as a redundancy policy. Redundancy ID

could also be used to distinguish different redundancy policies sharing the same redundancy node and merging node.

Redundancy Policy extends SR policy to include more than one ordered lists of segments between redundancy node and merging node, and all the ordered lists of segments are used at the same time to steer the copies of flow into different paths. In redundancy policy, Redundancy Segment MUST be specified, and the last segment of each ordered list of segments MUST be Merging Segment.

7. IANA Considerations

This document requires registration of End.R behavior and End.M behavior in "SRv6 Endpoint Behaviors" sub-registry of "Segment Routing Parameters" registry.

8. Security Considerations

TBD

9. Acknowledgements

The authors would like to thank Bruno Decraene, Ron Bonica, and James Guichard for their valuable comments.

10. References

10.1. Normative References

[I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-11 (work in progress), April 2021.

[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>.

[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>.

[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>.

[RFC8964] Varga, B., Ed., Farkas, J., Berger, L., Malis, A., Bryant, S., and J. Korhonen, "Deterministic Networking (DetNet) Data Plane: MPLS", RFC 8964, DOI 10.17487/RFC8964, January 2021, <https://www.rfc-editor.org/info/rfc8964>.

[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>.

10.2. Informative References

[I-D.ietf-spring-sr-service-programming] Clad, F., Xu, X., Filsfils, C., Bernier, D., Li, C., Decraene, B., Ma, S., Yadlapalli, C., Henderickx, W., and S. Salsano, "Service Programming with Segment Routing", draft-ietf-spring-sr-service-programming-04 (work in progress), March 2021.

[RFC8578] Grossman, E., Ed., "Deterministic Networking Use Cases", RFC 8578, DOI 10.17487/RFC8578, May 2019, <https://www.rfc-editor.org/info/rfc8578>.

[RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, DOI 10.17487/RFC8655, October 2019, <https://www.rfc-editor.org/info/rfc8655>.

Authors' Addresses

Xuesong Geng
Huawei Technologies
China

Email: gengxuesong@huawei.com

Mach(Guoyi) Chen
Huawei Technologies
China

Email: mach.chen@huawei.com

Fan Yang
Huawei Technologies
China

Email: shirley.yangfan@huawei.com

Pablo Camarillo Garvia
Cisco Systems, Inc.
Spain

Email: pcamaril@cisco.com

Gyan Mishra
Verizon Inc.

Email: gyan.s.mishra@verizon.com