Workgroup: Network Working Group
Internet-Draft:
draft-geng-spring-sr-enhanced-detnet-01
Published: 13 March 2023
Intended Status: Standards Track
Expires: 14 September 2023
Authors: X. Geng Z. Li T. Zhou
Huawei Huawei Huawei

Segment Routing for Enhanced DetNet

Abstract

One of the goals of DetNet is to provide bounded end-to-end latency for critical flows. This document defines how to leverage Segment Routing(SR) and Segment Routing over IPv6 (SRv6) to implement bounded latency. Specifically, new SRv6 SID function is used to specify bounded latency information for a packet. When forwarding devices along the path follow the instructions carried in the packet, the bounded latency is achieved by different implementations based on bounded latency information.

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

Copyright Notice

Table of Contents

1. Introduction

Deterministic Networking(DetNet) provides a capability to carry specified data flows with extremely low data loss rates and bounded latency within a network domain. DetNet is enabled by a group of technologies, such as resource allocation, service protection and explicit routes ([RFC8655]).

Segment Routing(SR) leverages the source routing paradigm. A ingress node steers a packet through an ordered list of instructions, called "segments". When SR is used over the MPLS data plane, SIDs are an MPLS label or an index into an MPLS label space (either SRGB or SRLB).

SR can also be applied over IPv6 data plane using Routing Extension Header(SRH). Besides routing, the segment of SRv6 can indicate functions which are executed locally in the node where they are defined. SRv6 network programming makes it convenient to add sophisticated operations in the network. ([RFC8402])

DetNet data plane is enhanced to facilitate DetNet transit nodes to support end-to-end bounded latency transmission. [I-D.yzz-detnet-enhanced-data-plane] introduces an unified data plane feild for bounded latency, which is called Bounded Latency Information(BLI) BLIis designed to cope with a variety of queuing/scheduling/shaping mechanisms in a uniform format in the data plane.

This document describes how to implement DetNet with SR or SRv6. It can provide : 1. Source routing, which can steer the DetNet flows go through the network according to an explicit route with allocated resource by segment list in SRH; 2. Network programming, which can give packet instructions in every node along the path to guarantee bounded latency. DetNet SR MPLS/SRv6 data plane extensions for enhanced DetNet are defined in this document.

2. Terminology and Conventions

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].

2.1. Terminology

Terminologies for DetNet go along with the definition in [RFC8655]. Other terminologies are defined as follows:

* NH: The IPv6 next-header field.

* SID: A Segment Identifier which represents a specific segment in a segment routing domain([RFC8402]).

* SRH: The Segment Routing Header ([RFC8754]).

2.2. Conventions

Conventions in the document are defined as follows:

* NH=SRH means that NH is 43 with routing type 4.

* A SID list is represented as <S1, S2, S3> where S1 is the first SID to visit, S2 is the second SID to visit and S3 is the last SID to visit along the SR path.

* SRH[SL] represents the SID pointed by the SL field, that is the SLth SID in the Segment List.

*(SA,DA) (S3, S2, S1; SL) represents an IPv6 packet with:

IPv6 header with source and destination addresses SA and DA respectively, and next-header SRH, with SID list <S1, S2, S3> with SegmentsLeft = SL

The payload of the packet is not represented

(S3, S2, S1; SL) represents the same SID list as <S1, S2, S3>, but encoded in the SRH format where the rightmost SID in the SRH is the first SID and the leftmost SID in the SRH is the last SID

3. SRv6 for Enhanced DetNet

To guarantee the end-to-end bounded latency transmission in DetNet network, bounded latency information is required to be conveyed inband with the service data to facilitate the queuing algorithm performed on the DetNet transit nodes. When the bounded latency information is used in DetNet IP data plane or DetNet MPLS data plane, it is carried in IP/UDP or MPLS encapsulations. When the bounded latency information is used in TSN over IP/MPLS data plane, the information used in TSN networks is transparently transmitted IP/ UDP or MPLS encapsulations. Note that, which queuing mechanism is used is a local choice determined by DetNet transit nodes. It is not necessary to be explicitly indicated in packets.

When an SRv6 SID is in the Destination Address field of an IPv6 header of a packet, it is routed through transit nodes in an IPv6 network as an IPv6 address. SRv6 SID consists of LOC:FUNCT:ARG, where a locator (LOC) is encoded in the L most significant bits of the SID, followed by F bits of function (FUNCT) and A bits of arguments (ARG), which is defined in ([RFC8986]).

Bounded Latency Information (BLI) is defined in [I-D.yzz-detnet-enhanced-data-plane] to guide forwarding in network device, which could be initiated in SRv6 data plane. With the characteristics of Segment Routing, the bounded latency information could be coupled with explicit path to provide latency guarantee in each node/ adjacency indicated by the segment list.

Bounded Latency Information is indicated by the allocated SID at each node along the path without maintaining per-flow states at the intermediate and egress nodes. Hence, it naturally supports flow aggregation, and that allows DetNet to support large number of DetNet flows and scale to large networks.

As defined in [I-D.yzz-detnet-enhanced-data-plane], 8 or more Bounded Latency Information Types (BLI Type) are introduced to differentiate the types of BLIs, based on the required information of queuing/scheduling/shaping mechanisms to guarantee bounded latency. Bounded Latency Information Value (BLI Value) is a specified value of a specific type of BLI to provide guidance for packet processing with the meaning of a particular BLI type. The pair <BLI Type, BLI Value> information should be indicated by SRv6 data plane.

The "Endpoint with L3 cross-connect" behavior ("End.X" for short) is a variant of the End behavior. It is the SRv6 instantiation of an Adj-SID ([RFC8402]), and its main use is for traffic-engineering policies.

Two new variations of End.X SID are defined for DetNet bounded latency, which are called End.X.BL and End.X.BLI respectively, and bounded latency information can be defined as functions or arguments in the new types of SID.

Editors Notes: Another option to implement this is to define new flavors. This method will be considered when not only End.X could be combined with BLI.

3.1. End.X.BL: Forwarding the packet with bounded latency guarantee

This document defines End.X.BL, which is used to identify Bounded Latency Information for Enhanced DetNet. End.X.BL a variation of End.X.

End.X.BL SID has two meanings: 1) to identify an interface/link, just like the adjacency SID; 2) to identify the pair <BLI Type and BLI Value> information on the interface/link to guarantee bounded latency. So different End.X.BL SIDs could be allocated to the same interface/link in order to indicated different pairs <BLI Type, BLI Value>.

The SRv6 encapsulation with End.X.BL SIDs is shown as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Next Header  |  Hdr Ext Len  |  Routing Type |  Segment Left |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Last Entry  |     Flags     |              Tag              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Segment List[0]                          |
|                    which is End.X.BL SID                      |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               ...                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Segment List[n]                          |
|                    which is End.X.BL SID                      |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Optional TLVS                        |
|                               ...                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

When N receives a packet destined to S and S is a local End.X.BL SID, N does the following:

```
S01. When an SRH is processed {
S02.   If (Segments Left == 0) {
S03.      Stop processing the SRH, and proceed to process the next
             header in the packet, whose type is identified by
             the Next Header field in the routing header.
S04.   }
S05.   If (IPv6 Hop Limit <= 1) {
S06.      Send an ICMP Time Exceeded message to the Source Address
             with Code 0 (Hop limit exceeded in transit),
             interrupt packet processing, and discard the packet.
S07.   }
S08.   max_LE = (Hdr Ext Len / 2) - 1
S09.   If ((Last Entry > max_LE) or (Segments Left > Last Entry+1)) {
S10.      Send an ICMP Parameter Problem to the Source Address
             with Code 0 (Erroneous header field encountered)
             and Pointer set to the Segments Left field,
             interrupt packet processing, and discard the packet.

S11.   }
S12.   Decrement IPv6 Hop Limit by 1
S13.   Decrement Segments Left by 1
S14.   Update IPv6 DA with Segment List[Segments Left]
S15.   Submit the packet to the IPv6 module for transmission
          to the new destination via a L3 adjacency indicated by the
          End.X.BL SID
S16.   Send the packet out using <BLI Type, BLI Value> indicated by t
          End.X.BL SID with the corresponding bounded latency guarant
          mechanism
S17. }
```

3.2. End.X.BLI: Forward the packet with bounded latency guarantee though BLI

The "Endpoint with forwarding the packet with bounded latency guarantee by BLI" behavior ("End.X.BLI" for short) is a variant of the End behavior.

End.X.BLI SID has two meanings: 1) to identify an interface/link, just like the adjacency SID; 2)to identify the BLI Type to guarantee bounded latency. So different End.X.BLI could be allocated to the same interface/link in order to indicated different types of BLIs. The BLI Value corresponding to the End.X.BLI SID is carried explicitly in the SRv6 packet header.

There are 3 possible options for carrying variable BLI Value associated with the End.X.BLI SID, including:

*Option1: Arguments in End.X.BLI SID

*Option2: SRH TLV for BLI used together with End.X.BLI SID

*Option3: New options in DoH before SRH together with End.X.BLI SID

3.2.1. BLI in Arguments of End.X.BLI SID

The behavior also takes an argument: "Arg.BLI". This argument provides a local BLI Value information for bounded latency guarantee. The SRH with End.X.BLI SIDs is showed as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Next Header  |  Hdr Ext Len  | Routing Type  | Segment Left  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Last Entry  |     Flags     |              Tag              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+---
|                  Location & Function                          |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+End.
|               Bounded Latency Information                     |SID
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+---
|                              ...                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Segment List[n]                          |
|                     which is End.X.BLI SID                    |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Optional TLVS                        |
|                              ...                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where

*Location&Function: the most significant bits that are used for routing and function indication;

*Bounded Latency Information : the least significant bits, which is defined [I-D.yzz-detnet-enhanced-data-plane].

When N receives a packet destined to S and S is a local End.X.BLI SID, N does the following:

```
S01. When an SRH is processed {
S02.   If (Segments Left == 0) {
S03.      Stop processing the SRH, and proceed to process the next
             header in the packet, whose type is identified by
             the Next Header field in the routing header.
S04.   }
S05.   If (IPv6 Hop Limit <= 1) {
S06.      Send an ICMP Time Exceeded message to the Source Address
             with Code 0 (Hop limit exceeded in transit),
             interrupt packet processing, and discard the packet.
S07.   }
S08.   max_LE = (Hdr Ext Len / 2) - 1
S09.   If ((Last Entry > max_LE) or (Segments Left > Last Entry+1)) {
S10.      Send an ICMP Parameter Problem to the Source Address
             with Code 0 (Erroneous header field encountered)
             and Pointer set to the Segments Left field,
             interrupt packet processing, and discard the packet.

S11.   }
S12.   Decrement IPv6 Hop Limit by 1
S13.   Decrement Segments Left by 1
S14.   Update IPv6 DA with Segment List[Segments Left]
S15.   Submit the packet to the IPv6 module for transmission
          to the new destination via a L3 adjacency  indicated by the
          End.X.BLI SID
S16.   Send the packet out using BLI Type indicated by the
          End.X.BLI SID and BLI Value carried in the argument
          with the corresponding bounded latency guarantee mechanism
S17. }
```

3.2.2. BLI in TLV of SRH

Optional TLV defined in SRH could also be extended for BLI, which is used together with End.X.BLI.

3.2.2.1. BLI List TLV

When all or part of the nodes/adjacencies in the explicit path indicated by the segment list request different BLI values corresponding to the End.X.BLI SID to guarantee bounded latency, a BLI List TLV is defined. The SRH with End.X.BLI SIDs is showed as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Next Header   |  Hdr Ext Len  | Routing Type  | Segment Left  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Last Entry   |     Flags     |              Tag              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Segment List[n]                          |
|                    which is End.X.BLI SID                     |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              ...                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Segment List[1]                          |
|                    which is End.X.BLI SID                     |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type(TBD1)   |    Length     |   BLI Left    |   RESERVED    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        BLI List [m]                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               ...                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        BLI List [1]                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Type field is 8 bits in length, and the value is TBD1.

The Length field is 8 bits in length and its value is variable, which depends on the length of BLI list.

BLI Left: 8-bit unsigned integer. Number of BLI remaining, i.e., number of explicitly listed intermediate nodes still to be visited before reaching the final destination.

BLI List[0..m]: 32-bit unsigned integer, representing the nth BLI in the BLI list.

The BLI in the BLI list corresponds to the Segment in the Segment List one by one. The length of BLI List depends on the number of End.X.BLI in the segment list.

When N receives a packet destined to S and S is a local End.X.BLI SID, N does the following:

```
S01. When an SRH is processed {
S02.   If (Segments Left == 0) {
S03.      Stop processing the SRH, and proceed to process the next
             header in the packet, whose type is identified by
             the Next Header field in the routing header.
S04.   }
S05.   If (IPv6 Hop Limit <= 1) {
S06.      Send an ICMP Time Exceeded message to the Source Address
             with Code 0 (Hop limit exceeded in transit),
             interrupt packet processing, and discard the packet.
S07.   }
S08.   max_LE = (Hdr Ext Len / 2) - 1
S09.   If ((Last Entry > max_LE) or (Segments Left > Last Entry+1)) {
S10.      Send an ICMP Parameter Problem to the Source Address
             with Code 0 (Erroneous header field encountered)
             and Pointer set to the Segments Left field,
             interrupt packet processing, and discard the packet.

S11.   }
S12.   Decrement IPv6 Hop Limit by 1
S13.   Decrement Segments Left by 1
S14.   Update IPv6 DA with Segment List[Segments Left]
S15.   Submit the packet to the IPv6 module for transmission
          to the new destination via a L3 adjacency
S16.   Send the packet out using BLI Type indicated by the
          End.X.BLI SID and BLI Value carried by BLI List[BLI Left]
          in SRH TLV and BLI Left--
          with the corresponding bounded latency guarantee mechanism
S17. }
```

3.2.2.2. Shared BLI TLV

When all the nodes/adjacencies in the explicit path indicated by the segment list request the same BLI value to guarantee bounded latency, the Shared BLI TLV is defined. The SRH with End.X.BLI SIDs is showed as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Next Header   |  Hdr Ext Len  | Routing Type  | Segment Left  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Last Entry   |     Flags     |              Tag              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Segment List[n]                          |
|                     which is End.X.BLI SID                    |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            ...                                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      Segment List[1]                          |
|                     which is End.X.BLI SID                    |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type(TBD2)   |    Length     |              RESERVED         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Shared BLI                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Type field is 8 bits in length, and the value is TBD2.

The Length field is 8 bits in length and its value is variable, which depends on the length of BLI list.

The Shared BLI field is 32 bits in length and corresponds to definition of BLI in [I-D.yzz-detnet-enhanced-data-plane].

When N receives a packet destined to S and S is a local End.X.BLI SID, N does the following:

```
S01. When an SRH is processed {
S02.   If (Segments Left == 0) {
S03.      Stop processing the SRH, and proceed to process the next
             header in the packet, whose type is identified by
             the Next Header field in the routing header.
S04.   }
S05.   If (IPv6 Hop Limit <= 1) {
S06.      Send an ICMP Time Exceeded message to the Source Address
             with Code 0 (Hop limit exceeded in transit),
             interrupt packet processing, and discard the packet.
S07.   }
S08.   max_LE = (Hdr Ext Len / 2) - 1
S09.   If ((Last Entry > max_LE) or (Segments Left > Last Entry+1)) {
S10.      Send an ICMP Parameter Problem to the Source Address
             with Code 0 (Erroneous header field encountered)
             and Pointer set to the Segments Left field,
             interrupt packet processing, and discard the packet.

S11.   }
S12.   Decrement IPv6 Hop Limit by 1
S13.   Decrement Segments Left by 1
S14.   Update IPv6 DA with Segment List[Segments Left]
S15.   Submit the packet to the IPv6 module for transmission
          to the new destination via a L3 adjacency
S16.   Send the packet out using BLI Type indicated by the
          End.X.BLI SID and BLI Value indicated by Shared BLI TLV
          with the corresponding bounded latency guarantee mechanism
S17. }
```

3.2.2.3. BLI Options in DOH before SRH

According to [RFC8200], BLI could also be defined through DOH before SRH for the specified segment. For the case of BLI List, considering that the location of DOH is before SRH, it is not recommended to be defined in DoH, because it will affect the processing efficiency of Segment in SRH. For Shared BLI TLV, it can be carried by the DOH Option. In order for the consistency, this document recommends to use the SRH TLV to carry both information.

4. SR MPLS for Enhanced DetNet

For SR MPLS data plane, this document defines a new segment that is called a BLI Segment, which is used to identify Bounded Latency Information for Enhanced DetNet just like End.BL SID. A BLI Segment is an adjacency segment and allocated from the Segment Routing Local Block (SRLB)[RFC8402]. BLI Segment indicateds <BLI Type, BLI Value> of an interface/link. So different BLI segments could be allocated to the same interface/link in order to indicated different pairs <BLI Type, BLI Value>.

Editors Notes: SR MPLS extension with meta data which is still under discussion will be defined based on the progress of MPLS DT. The possible definition of MPLS segment associated with the variable BLI values like the SRv6 End.X.BLI will be defined in the future version.

5. IANA Considerations

The following codepoints are defined in this document in Segment Routing Header TLVs registry:

Value	Description	Reference
TBD1	BLI List TLV	This document
TBD2	Shared BLI TLV	This document

6. Security Considerations

TBD

7. Normative References

[I-D.ietf-detnet-bounded-latency] Finn, N., Le Boudec, J., Mohammadpour, E., Zhang, J., and B. Varga, "Deterministic Networking (DetNet) Bounded Latency", Work in Progress, Internet-Draft, draft-ietf-detnet-bounded-latency-10, 8 April 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-detnet-bounded-latency-10>.

[I-D.yzz-detnet-enhanced-data-plane] Geng, X., Zhou, T., Zhang, L., and Z. Du, "DetNet Enhanced Data Plane", Work in Progress, Internet-Draft, draft-yzz-detnet-enhanced-data-plane-02, 24 December 2022, <https://datatracker.ietf.org/doc/html/draft-yzz-detnet-enhanced-data-plane-02>.

[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>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/

RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[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>.

[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>.

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

Authors' Addresses

Xuesong Geng
Huawei

Email: gengxuesong@huawei.com

Zhenbin Li
Huawei

Email: lizhenbin@huawei.com

Tianran Zhou
Huawei

Email: zhoutianran@huawei.com