LSR Working Group
Internet-Draft
Intended status: Standards Track
Expires: 7 November 2025

S. Litkowski
Cisco Systems
Y. Qu
Futurewei Technologies
A. Lindem
LabN Consulting, L.L.C.
I. Chen
The MITRE Corporation
J. Tantsura
Nvidia
6 May 2025

A YANG Data Model for IS-IS Segment Routing over the MPLS Data Plane
draft-ietf-isis-sr-yang-31

Abstract

This document defines a YANG data model that can be used to manage IS-IS Extensions for Segment Routing over the MPLS data plane.

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 7 November 2025.

Copyright Notice

Table of Contents

1. Overview 2
1.1. Requirements Language 2
1.2. Tree Diagrams 2
2. Design of the IS-IS MPLS Segment Routing Module 3
2.1. Segment Routing Activation 3
2.2. Advertising Mapping Server Policy 3
2.3. IP Fast Reroute 3
3. IS-IS MPLS Segment Routing over MPLS YANG Module 3
4. Security Considerations 23
5. Acknowledgements 25
6. IANA Considerations 25
7. References 25
7.1. Normative References 25
7.2. Informative References 27
Appendix A. A Configuration Example 28
Appendix B. IS-IS MPLS Segment Routing Module Tree 30
Authors' Addresses 35

1. Overview

This document defines a device YANG data model [RFC7950] that can be used to manage IS-IS Extensions for Segment Routing [RFC8667] over the MPLS data plane. It is an augmentation to the IS-IS YANG data model [RFC9130].

The YANG data model in this document conforms to the Network Management Datastore Architecture (NMDA) [RFC8342].

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.

1.2. Tree Diagrams

This document uses the graphical representation of data model defined in [RFC8340].

2. Design of the IS-IS MPLS Segment Routing Module

This document defines a YANG data model for IS-IS Extensions for Segment Routing over the MPLS data plane. It is an augmentation of the IS-IS base model.

The IS-IS SR MPLS YANG module requires support for the base segment routing module [RFC9020], which defines the global segment routing management independent of any specific routing protocol configuration, and support of IS-IS base model [RFC9130] which defines basic IS-IS configuration and state.

The ietf-isis-sr-mpls data module defines both the data nodes to configure IS-IS segment routing MPLS extensions and the additions to the IS-IS Link State Protocol Data Units (LSPs) necessary to support MPLS segment routing.

2.1. Segment Routing Activation

Activation of IS-IS SR MPLS is done by setting the "enable" leaf to true. This triggers advertisement of SR MPLS extensions based on the configuration parameters that have been set up using the base segment routing module.

2.2. Advertising Mapping Server Policy

The base segment routing module defines mapping server policies. By default, IS-IS will not advertise or process any mapping server entry. The IS-IS SR MPLS module allows the advertisement of one or multiple mapping server policies through the "bindings/advertise/ policies" leaf-list. The "bindings/receive" leaf controls the reception and process of mapping server entries.

2.3. IP Fast Reroute

IS-IS SR MPLS model augments the fast-reroute container under interface. It brings the ability to activate TI-LFA (Topology Independent LFA) and also enhances remote LFA to use segment-routing tunneling instead of LDP.

3. IS-IS MPLS Segment Routing over MPLS YANG Module

[RFC6991], [RFC8102], [RFC8294], [RFC8349], [RFC8667], [RFC9020], [RFC9130], and [I-D.ietf-rtgwg-segment-routing-ti-lfa] are referenced in the YANG module.

```
   <CODE BEGINS> file "ietf-isis-sr-mpls@2025-05-06.yang"
   module ietf-isis-sr-mpls {
     yang-version 1.1;
     namespace "urn:ietf:params:xml:ns:yang:ietf-isis-sr-mpls";
     prefix isis-sr-mpls;

     import ietf-routing {
       prefix rt;
       reference
         "RFC 8349: A YANG Data Model for Routing
                    Management (NMDA Version)";
     }
     import ietf-inet-types {
       prefix inet;
       reference
         "RFC 6991: Common YANG Data Types";
     }
     import ietf-segment-routing-common {
       prefix sr-cmn;
       reference
         "RFC 9020: YANG Data Model for Segment Routing";
     }
     import ietf-segment-routing-mpls {
       prefix sr-mpls;
       reference
         "RFC 9020: YANG Data Model for Segment Routing";
     }
     import ietf-isis {
       prefix isis;
       reference
         "RFC 9130: YANG Data Model for IS-IS Protocol";
     }
     import iana-routing-types {
       prefix iana-rt-types;
       reference
         "RFC 8294: Common YANG Data Types for the Routing Area";
     }
     import ietf-routing-types {
       prefix rt-types;
       reference
         "RFC 8294: Common YANG Data Types for the Routing Area";
     }

     organization
       "IETF LSR - Link State Routing Working Group";
     contact
       "WG Web:   <https://datatracker.ietf.org/wg/lsr/>
        WG List:  <mailto:lsr@ietf.org>
```

```
        Author:    Stephane Litkowski
                   <mailto:slitkows.ietf@gmail.com>
        Author:    Yingzhen Qu
                   <mailto:yingzhen.ietf@gmail.com>
        Author:    Acee Lindem
                   <mailto:acee.ietf@gmail.com>
        Author:    Ing-Wher Chen
                   <mailto:ingwherchen@mitre.org>
        Author:    Jeff Tantsura
                   <mailto:jefftant.ietf@gmail.com>
      ";
    description
      "The YANG module defines the generic configuration and
       operational state for Segment Routing ISIS extensions for the
       MPLS data plane.

       This YANG model conforms to the Network Management
       Datastore Architecture (NMDA) as described in RFC 8342.

       Copyright (c) 2025 IETF Trust and the persons identified as
       authors of the code.  All rights reserved.

       Redistribution and use in source and binary forms, with or
       without modification, is permitted pursuant to, and subject to
       the license terms contained in, the Revised BSD License set
       forth in Section 4.c of the IETF Trust's Legal Provisions
       Relating to IETF Documents
       (https://trustee.ietf.org/license-info).

       This version of this YANG module is part of RFC XXXX
       (https://www.rfc-editor.org/info/rfcXXXX); see the RFC itself
       for full legal notices.

       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 (RFC 2119) (RFC 8174) when, and only when,
       they appear in all capitals, as shown here.

       This version of this YANG module is part of RFC XXXX;
       see the RFC itself for full legal notices.";

    // RFC Ed.: Replace XXXX with actual RFC number and remove
    // this note

    revision 2025-05-06 {
      description
        "Initial revision.";
```

```
      reference
        "RFC XXXX: A YANG Data Model for IS-IS Segment Routing over
                   the MPLS Data Plane";
    }

    // RFC Ed.: Replace XXXX with actual RFC number.
    // Also, replace '2025-04-22' with the module publication date
    // in the format (YYYY-MM-DD) and remove this note.

    /* Features */

    feature remote-lfa-sr {
      description
        "Enhance rLFA to use SR path.";
      reference
        "RFC 8102: Remote-LFA Node Protection and Manageability";
    }

    feature ti-lfa {
      description
        "Topology-Independent Loop-Free Alternate (TI-LFA)
         computation using segment routing.";
      reference
        "draft-ietf-rtgwg-segment-routing-ti-lfa -
         Topology-Independent Fast Reroute using Segment Routing";
    }

    /* Identities */

    identity sr-capability {
      description
        "Base identity for ISIS SR-Capabilities sub-TLV flags";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing";
    }

    identity mpls-ipv4 {
      base sr-capability;
      description
        "If set, then the router is capable of
         processing SR MPLS encapsulated IPv4 packets
         on all interfaces.";
    }

    identity mpls-ipv6 {
      base sr-capability;
      description
        "If set, then the router is capable of
```

```
          processing SR MPLS encapsulated IPv6 packets
          on all interfaces.";
      }

      identity prefix-sid-flag {
        description
          "Base identity for prefix SID sub-TLV flags.";
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.1";
      }

      identity r-flag {
        base prefix-sid-flag;
        description
          "Re-advertisement Flag.";
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.1";
      }

      identity n-flag {
        base prefix-sid-flag;
        description
          "Node-SID Flag.";
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.1";
      }

      identity p-flag {
        base prefix-sid-flag;
        description
          "No-PHP (No Penultimate Hop-Popping) Flag.";
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.1";
      }

      identity e-flag {
        base prefix-sid-flag;
        description
          "Explicit NULL Flag.";
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.1";
      }

      identity v-flag {
        base prefix-sid-flag;
        description
          "Value Flag.";
        reference
```

```
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.1";
    }

    identity l-flag {
      base prefix-sid-flag;
      description
        "Local Flag.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.1";
    }

    identity adj-sid-flag {
      description
        "Base identity for Adj-SID sub-TLV flags.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
    }

    identity f-flag {
      base adj-sid-flag;
      description
        "Address-Family flag.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
    }

    identity b-flag {
      base adj-sid-flag;
      description
        "Backup flag.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
    }

    identity vi-flag {
      base adj-sid-flag;
      description
        "Value/Index flag - corresponds to V-Flag in reference.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
    }

    identity lg-flag {
      base adj-sid-flag;
      description
        "Local/Global flag - corresponds to L-Flag in reference.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
```

```
  }

  identity s-flag {
    base adj-sid-flag;
    description
      "Set flag - corresponds to S-Flag in reference..";
    reference
      "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
  }

  identity pe-flag {
    base adj-sid-flag;
    description
      "Persistent flag - corresponds to P-Flag in reference.";
    reference
      "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
  }

  identity sid-binding-flag {
    description
      "Base identity for sid binding tlv flags.";
    reference
      "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.4";
  }

  identity af-flag {
    base sid-binding-flag;
    description
      "Address-Family flag - corresponds to F-Flag in reference.";
    reference
      "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.4";
  }

  identity m-flag {
    base sid-binding-flag;
    description
      "Mirror Context flag.";
    reference
      "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.4";
  }

  identity sf-flag {
    base sid-binding-flag;
    description
      "S flag. If set, the binding label TLV should be flooded
       across the entire routing domain - corresponds to S-Flag in
       reference.";
    reference
```

```
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.4";
    }

    identity d-flag {
      base sid-binding-flag;
      description
        "Leaking flag.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.4";
    }

    identity a-flag {
      base sid-binding-flag;
      description
        "Attached flag.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.4";
    }

    /* Groupings */

    grouping sid-tlv-encoding {
      description
        "SID TLV Encoding -  20-bit label or 32-bit SID index whose
         interpretation is dependent on the TLV length (3 for an
         MPLS label or 4 for a 32-bit value)  or the TLV V-Flag and
         L-Flag settings:

          If the  V-Flag is set to 0 and L-Flag is set to 0:
          The SID/Index/Label field is a 4-octet index defining
          the offset in the SID/Label space advertised by this
          router.

          If V-Flag is set to 1 and L-Flag is set to 1: The
          ID/Index/Label field is a 3-octet local label where the
          20 rightmost bits are used for encoding the label value.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing,
         Section 2.1.1.1";
      choice sid {
        case sid-label {
          leaf label-value {
            type uint32 {
              range "0 .. 1048575";
            }
            description
              "A 20-bit MPLS Label";
          }
```

```
        }
        case sid-index {
          leaf index-value {
            type uint32;
            description
              "Index into a label space advertised by this router.";
          }
        }
        description
          "Choice of either a 20-bit MPLS label or 32-bit index into
           an advertised label space.";
      }
    }

    grouping sr-capability {
      description
        "SR capability grouping.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 3.1";
      container sr-capability {
        description
          "SR Capability Flags.";
        leaf-list sr-capability-flag {
          type identityref {
            base sr-capability;
          }
          description
            "SR Capability sub-TLV flags.";
        }
        container global-blocks {
          description
            "Segment Routing Global Blocks.";
          list global-block {
            description
              "Segment Routing Global Block.";
            leaf range-size {
              type rt-types:uint24;
              description
                "The SID range.";
            }
            uses sid-tlv-encoding;
          }
        }
      }
    }

    grouping sr-algorithm {
      description
```

```
        "SR algorithm grouping.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 3.2";
      container sr-algorithms {
        description
          "All SR algorithms.";
        leaf-list sr-algorithm {
          type identityref {
            base sr-cmn:prefix-sid-algorithm;
          }
          description
            "The Segment Routing (SR) algorithms that the router is
             currently using.";
        }
      }
    }

    grouping srlb {
      description
        "SR Local Block grouping.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 3.3";
      container local-blocks {
        description
          "List of SRLBs.";
        list local-block {
          description
            "Segment Routing Local Block.";
          leaf range-size {
            type rt-types:uint24;
            description
              "The SID range.";
          }
          uses sid-tlv-encoding;
        }
      }
    }

    grouping srms-preference {
      description
        "The SRMS preference TLV is used to advertise
         a preference associated with the node that acts
         as an SR Mapping Server.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 3.4";
      container srms-preference {
        description
          "SRMS Preference TLV.";
```

```
        leaf preference {
          type uint8;
          description
            "SRMS preference TLV, value from 0 to 255 with
             255 being the most preferred.";
        }
      }
    }

    grouping adjacency-state {
      description
        "This grouping extends adjacency state.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
      list adjacency-sid {
        config false;
        description
          "List of adjacency Segment IDs.";
        leaf value {
          type uint32;
          description
            "Value of the Adj-SID.";
        }
        leaf address-family {
          type iana-rt-types:address-family;
          description
            "Address-family associated with the
             segment ID";
        }
        leaf weight {
          type uint8;
          description
            "Weight associated with
             the adjacency SID.";
        }
        leaf protection-requested {
          type boolean;
          description
            "Describe if the adjacency SID
             must be protected.";
        }
      }
    }

    grouping prefix-sid-sub-tlv {
      description
        "This grouping defines segment routing prefix segment
         identifier (prefix-sid) sub-tlv.";
```

```
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.1";
        container prefix-sid-sub-tlvs {
          description
            "PrefixSID sub-tlvs.";
          list prefix-sid-sub-tlv {
            description
              "List of prefix-sid sub-tlvs.";
            container prefix-sid-flags {
              description
                "Describes flags associated with the
                 segment ID.";
              leaf-list flag {
                type identityref {
                  base prefix-sid-flag;
                }
                description
                  "Prefix SID sub-TLV flags.";
              }
            }
            leaf algorithm {
              type identityref {
                base sr-cmn:prefix-sid-algorithm;
              }
              description
                "Algorithm to be used for path computation.";
            }
            uses sid-tlv-encoding;
          }
        }
      }

      grouping adjacency-segment-id {
        description
          "This grouping defines segment routing extensions
           for adjacencies.";
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing, Section 2.2";
        container adj-sid-sub-tlvs {
          description
            "Adj-SID optional sub-TLVs.";
          list adj-sid-sub-tlv {
            description
              "List of segments.";
            container adj-sid-flags {
              description
                "Adj-SID sub-TLV flags.";
              leaf-list flag {
```

```
              type identityref {
                base adj-sid-flag;
              }
              description
                "Adj-SID sub-TLV flags list.";
            }
          }
          leaf weight {
            type uint8;
            description
              "The value represents the weight of the Adj-SID
               for the purpose of load balancing.";
          }
          leaf neighbor-id {
            type isis:system-id;
            description
              "Describes the system ID of the neighbor
               associated with the SID value. This is only
               used on LAN adjacencies.";
          }
          uses sid-tlv-encoding;
        }
      }
    }

    grouping sid-binding-tlv {
      description
        "SID/Label binding TLV, type 149.";
      reference
        "RFC 8667: IS-IS Extensions for Segment Routing,
         Section 2.4";
      leaf prefix {
        type inet:ip-prefix;
        description
          "The prefix represents the Forwarding Equivalence
           Class at the tail end of the advertised path.";
      }
      leaf range {
        type uint16;
        description
          "Provides the ability to specify a range of addresses
           and their associated prefix-sids.";
      }
      container sid-binding-flags {
        description
          "Binding TLV flags.";
        leaf-list flag {
          type identityref {
```

```
            base sid-binding-flag;
          }
          description
            "SID Binding TLV flags.";
        }
      }
      list prefix-sid-sub-tlvs {
        description
          "List of prefix-sid sub-tlvs.";
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing,
           Section 2.4.4";
        uses prefix-sid-sub-tlv;
      }
      list sid-sub-tlv {
        description
          "List of prefix-sid sub-tlvs.";
        reference
          "RFC 8667: IS-IS Extensions for Segment Routing,
           Section 2.4.5";
        uses sid-tlv-encoding;
      }
      uses isis:unknown-tlvs;
    }

    /* Configuration */

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis" {
      when "derived-from-or-self(../rt:type, 'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol configuration
         with segment routing for the MPLS data plane.";
      uses sr-mpls:sr-control-plane;
      container protocol-srgb {
        if-feature "sr-mpls:protocol-srgb";
        description
          "Per-protocol Segment Routing Global Block (SRGB).";
        reference
          "RFC 8402: Segment Routing Architecture, Section 2";
        uses sr-cmn:srgb;
      }
    }
```

```
    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:interfaces/isis:interface" {
      when "derived-from-or-self(../../../rt:type, 'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol configuration
         with segment routing.";
      uses sr-mpls:igp-interface {
        augment "segment-routing/adjacency-sid/adj-sids" {
          when "../../../isis:interface-type = 'broadcast'" {
            description
              "This augments broadcast interface.";
          }
          description
            "This augments LAN interface adj-sid with system-id.";
          leaf neighbor-system-id {
            type isis:system-id;
            mandatory true;
            description
              "Neighbor system ID.";
          }
        }
      }
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:interfaces/isis:interface"
          + "/isis:fast-reroute/isis:lfa" {
      when "derived-from-or-self(../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS interface IP FRR with TILFA.";
      container ti-lfa {
        if-feature "ti-lfa";
        description
          "Topology Independent Loop Free Alternate
           (TI-LFA) support.";
        leaf enabled {
          type boolean;
          default "false";
          description
```

```
            "Enables TI-LFA computation.";
        }
        container selection-tie-breakers {
          description
            "Configure path selection tie-breakers and their
             respective priorities for the TI-LFA computation.";
          container node-protection {
            presence "Presence of container enables the node
                      protection tie-breaker";
            description
              "Enable node protection as a TI-LFA path
               selection tie-breaker. A path providing node
               protection will be selected over one that
               doesn't provide node protection.";
            leaf priority {
              type uint8;
              default "128";
              description
                "Priority for node protection tie-breaker with
                 a lower priority being more preferred.";
            }
          }
          container srlg-disjoint {
            presence "Presence of container enables the SRLG
                      disjoint tie-breaker";
            description
              "Enable SRLG (Shared Resource Link Group)
               disjoint  as a TI-LFA path selection tie-breaker.
               A path providing node a disjoint path for SRLG
               links from the primary path will be selected over
               one that doesn't provide an SRLG disjoint path.";
            leaf priority {
              type uint8;
              default "128";
              description
                "Priority for SRLG disjoint tie-breaker with
                 a lower priority being more preferred.";
            }
          }
        }
      }
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:interfaces/isis:interface"
          + "/isis:fast-reroute/isis:lfa/isis:level-1" {
      when "derived-from-or-self(../../../../../../rt:type,"
```

```
          + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS interface level-1 IP FRR with TILFA.";
      container ti-lfa {
        if-feature "ti-lfa";
        description
          "TI-LFA configuration.";
        leaf enabled {
          type boolean;
          default "false";
          description
            "Enables TI-LFA computation.";
        }
      }
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:interfaces/isis:interface"
          + "/isis:fast-reroute/isis:lfa/isis:level-2" {
      when "derived-from-or-self(../../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS IP interface level-2 FRR with TILFA.";
      container ti-lfa {
        if-feature "ti-lfa";
        description
          "TI-LFA configuration.";
        leaf enabled {
          type boolean;
          default "false";
          description
            "Enables TI-LFA computation.";
        }
      }
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:interfaces/isis:interface"
          + "/isis:fast-reroute/isis:lfa/isis:remote-lfa" {
      when "derived-from-or-self(../../../../../../rt:type,"
```

```
          + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS remoteLFA config with
         use of segment-routing path.";
      leaf use-segment-routing-path {
        if-feature "remote-lfa-sr";
        type boolean;
        default "false";
        description
          "Force remote LFA to use segment routing path instead of LDP
           path. The value of this leaf is in effect only when
           remote-lfa is enabled.";
      }
    }

    /* Operational states */

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:interfaces/isis:interface"
          + "/isis:adjacencies/isis:adjacency" {
      when "derived-from-or-self(../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol configuration
         with segment routing.";
      uses adjacency-state;
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:database/isis:levels/isis:lsp"
          + "/isis:router-capabilities/isis:router-capability" {
      when "derived-from-or-self(../../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB router capability.";
      uses sr-capability;
      uses sr-algorithm;
```

```
      uses srlb;
      uses srms-preference;
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:database/isis:levels/isis:lsp"
          + "/isis:extended-is-neighbor/isis:neighbor/isis:instances"
          + "/isis:instance" {
      when "derived-from-or-self(../../../../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB neighbor.";
      uses adjacency-segment-id;
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:database/isis:levels/isis:lsp"
          + "/isis:mt-is-neighbor/isis:neighbor/isis:instances"
          + "/isis:instance" {
      when "derived-from-or-self(../../../../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB neighbor.";
      uses adjacency-segment-id;
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:database/isis:levels/isis:lsp"
          + "/isis:extended-ipv4-reachability/isis:prefixes" {
      when "derived-from-or-self(../../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB prefix.";
      uses prefix-sid-sub-tlv;
    }
```

```
    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:database/isis:levels/isis:lsp"
          + "/isis:mt-extended-ipv4-reachability/isis:prefixes" {
      when "derived-from-or-self(../../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB prefix.";
      uses prefix-sid-sub-tlv;
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:database/isis:levels/isis:lsp"
          + "/isis:ipv6-reachability/isis:prefixes" {
      when "derived-from-or-self(../../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB prefix.";
      uses prefix-sid-sub-tlv;
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:database/isis:levels/isis:lsp"
          + "/isis:mt-ipv6-reachability/isis:prefixes" {
      when "derived-from-or-self(../../../../../../rt:type,"
         + "'isis:isis')" {
        description
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB prefix.";
      uses prefix-sid-sub-tlv;
    }

    augment "/rt:routing/"
          + "rt:control-plane-protocols/rt:control-plane-protocol"
          + "/isis:isis/isis:database/isis:levels/isis:lsp" {
      when "derived-from-or-self(../../../../rt:type,"
         + "'isis:isis')" {
        description
```

```
          "This augments ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB.";
      container sid-binding-tlvs {
        description
          "List of sid/label binding TLVs.";
        list sid-binding-tlv {
          key "prefix";
          description
            "Sid/label binding TLV, type 149.";
          uses sid-binding-tlv;
        }
      }
      container mt-sid-binding-tlvs {
        description
          "List of multi-topology sid/label binding TLVs.";
        list mt-sid-binding-tlv {
          key "prefix mt-id";
          description
            "Multi-Topology SID/Label binding TLV, type 150.";
          reference
            "RFC 8667: IS-IS Extensions for Segment Routing,
             Section 2.5";
          uses sid-binding-tlv;
          leaf mt-id {
            type uint16;
            description
              "A 12-bit field containing the non-zero ID
               of the topology.";
          }
        }
      }
    }
  }
  <CODE ENDS>
```

Figure 1

4. Security Considerations

The ietf-isis-sr-mpls YANG module defines a data model that is designed to be accessed via YANG-based management protocols, such as NETCONF [RFC6241] and RESTCONF [RFC8040]. These YANG-based management protocols (1) have to use a secure transport layer (e.g., SSH [RFC4252], TLS [RFC8446], and QUIC [RFC9000]) and (2) have to use mutual authentication.

The Network Configuration Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., "config true", which is the default). All writable data nodes are likely to be reasonably sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) and delete operations to these data nodes without proper protection or authentication can have a negative effect on network operations. The following subtrees and data nodes have particular sensitivities/vulnerabilities:

/isis:isis/segment-routing

/isis:isis/protocol-srgb

/isis:isis/isis:interfaces/isis:interface/segment-routing

/isis:isis/isis:interfaces/isis:interface/isis:fast-reroute/ti-lfa

The ability to disable or enable IS-IS Segment Routing support and/or change Segment Routing configurations can result in a Denial-of-Service (DoS) attack, as this may cause traffic to be dropped or misrouted. Please refer to Section 5 of [RFC8667] for more information on Segment Routing extensions.

Some of the readable data nodes in the modules may be considered sensitive or vulnerable in some network environments. It is thus important to control read access (e.g., via get, get-config, or notification) to these data nodes.

/isis:router-capabilities/sr-capability

/isis:router-capabilities/sr-algorithms

/isis:router-capabilities/local-blocks

/isis:router-capabilities/srms-preference

And the augmentations to the ISIS link state database.

Unauthorized access to any data node of these subtrees can disclose the operational state information of IS-IS protocol on a device.

5. Acknowledgements

The authors would like to thank Pushpasis Sarkar, Derek Yeung,Yi Yang for their major contributions to the draft. Also thank Reshad Rahman, Tom Petch, and Renato Westphal for their thorough reviews and helpful comments.

The authors wish to thank Mohamed Boucadair and Ketan Talaulikar for extensive AD reviews for the IESG Telechat. Thanks to Eric Vyncke and Mahesh Jethanandani for review comments.

Author affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or viewpoints expressed. MITRE has agreed to this document for Public Release, Distribution Unlimited, with Public Release Case Number 19-3033. This approval allows the MITRE employee to release contributions externally and does not assert authority to publish.

6. IANA Considerations

The IANA is requested to assign one new URI from the IETF XML registry ([RFC3688]):

```
        URI: urn:ietf:params:xml:ns:yang:ietf-isis-sr-mpls
        Registrant Contact: The IESG.
        XML: N/A, the requested URI is an XML namespace
```

This document also requests one new YANG module name in the YANG Module Names registry ([RFC6020]):

```
        name: ietf-isis-sr-mpls
        namespace: urn:ietf:params:xml:ns:yang:ietf-isis-sr-mpls
        prefix: isis-sr-mpls
  maintained by IANA? N
        reference: RFC XXXX
```

7. References

7.1. Normative References

[I-D.ietf-rtgwg-segment-routing-ti-lfa]
Bashandy, A., Litkowski, S., Filsfils, C., Francois, P., Decraene, B., and D. Voyer, "Topology Independent Fast Reroute using Segment Routing", Work in Progress, Internet-Draft, draft-ietf-rtgwg-segment-routing-ti-lfa-21, 12 February 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-rtgwg-segment-routing-ti-lfa-21>.

[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>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC4252] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Authentication Protocol", RFC 4252, DOI 10.17487/RFC4252, January 2006, <https://www.rfc-editor.org/info/rfc4252>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC8102] Sarkar, P., Ed., Hegde, S., Bowers, C., Gredler, H., and S. Litkowski, "Remote-LFA Node Protection and Manageability", RFC 8102, DOI 10.17487/RFC8102, March 2017, <https://www.rfc-editor.org/info/rfc8102>.

[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>.

[RFC8294] Liu, X., Qu, Y., Lindem, A., Hopps, C., and L. Berger, "Common YANG Data Types for the Routing Area", RFC 8294, DOI 10.17487/RFC8294, December 2017, <https://www.rfc-editor.org/info/rfc8294>.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <https://www.rfc-editor.org/info/rfc8341>.

[RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, DOI 10.17487/RFC8342, March 2018, <https://www.rfc-editor.org/info/rfc8342>.

[RFC8349] Lhotka, L., Lindem, A., and Y. Qu, "A YANG Data Model for Routing Management (NMDA Version)", RFC 8349, DOI 10.17487/RFC8349, March 2018, <https://www.rfc-editor.org/info/rfc8349>.

[RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <https://www.rfc-editor.org/info/rfc8667>.

[RFC9020] Litkowski, S., Qu, Y., Lindem, A., Sarkar, P., and J. Tantsura, "YANG Data Model for Segment Routing", RFC 9020, DOI 10.17487/RFC9020, May 2021, <https://www.rfc-editor.org/info/rfc9020>.

[RFC9130] Litkowski, S., Ed., Yeung, D., Lindem, A., Zhang, J., and L. Lhotka, "YANG Data Model for the IS-IS Protocol", RFC 9130, DOI 10.17487/RFC9130, October 2022, <https://www.rfc-editor.org/info/rfc9130>.

7.2. Informative References

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8792] Watsen, K., Auerswald, E., Farrel, A., and Q. Wu, "Handling Long Lines in Content of Internet-Drafts and RFCs", RFC 8792, DOI 10.17487/RFC8792, June 2020, <https://www.rfc-editor.org/info/rfc8792>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

Appendix A. A Configuration Example

The following is an XML example using the ISIS SR MPLS YANG module defined in this document.

Note: '\' line wrapping per [RFC8792].

```
<?xml version='1.0' encoding='UTF-8'?>
<interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <interface>
    <name>eth0</name>
    <admin-status>up</admin-status>
    <oper-status>up</oper-status>
    <if-index>1</if-index>
    <statistics>
      <discontinuity-time>2024-10-27T14:30:00Z</discontinuity-time>
    </statistics>
  </interface>
</interfaces>
<routing xmlns="urn:ietf:params:xml:ns:yang:ietf-routing">
 <router-id>1.1.1.1</router-id>
 <control-plane-protocols>
   <control-plane-protocol>
     <type xmlns:isis="urn:ietf:params:xml:ns:yang:ietf-isis">\
       isis:isis</type>
     <name>isis</name>
     <isis xmlns="urn:ietf:params:xml:ns:yang:ietf-isis">
       <system-id>1111.2222.3333</system-id>
       <area-address>49.0001.0000.0000.0001</area-address>
       <interfaces>
         <interface>
           <name>eth0</name>
           <segment-routing xmlns="urn:ietf:params:xml:ns:yang:\
             ietf-isis-sr-mpls">
```

```
                <adjacency-sid>
                  <adj-sids>
                    <value>38888</value>
                    <neighbor-system-id>4444.5555.6666\
                    </neighbor-system-id>
                  </adj-sids>
                </adjacency-sid>
              </segment-routing>
            </interface>
          </interfaces>
          <segment-routing xmlns="urn:ietf:params:xml:ns:yang:\
            ietf-isis-sr-mpls">
            <enabled>true</enabled>
          </segment-routing>
          <protocol-srgb xmlns="urn:ietf:params:xml:ns:yang:\
            ietf-isis-sr-mpls">
            <srgb>
              <lower-bound>4000</lower-bound>
              <upper-bound>5000</upper-bound>
            </srgb>
          </protocol-srgb>
          </isis>
       </control-plane-protocol>
    </control-plane-protocols>
   </routing>
```

The following is the same example using JSON format.

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "admin-status": "up",
        "oper-status": "up",
        "if-index": 1,
        "statistics": {
          "discontinuity-time": "2024-10-27T07:30:00-07:00"
        }
      }
    ]
  },
  "ietf-routing:routing": {
    "router-id": "1.1.1.1",
    "control-plane-protocols": {
      "control-plane-protocol": [
        {
          "type": "ietf-isis:isis",
```

```
            "name": "isis",
            "ietf-isis:isis": {
              "system-id": "1111.2222.3333",
              "area-address": [
                "49.0001.0000.0000.0001"
              ],
              "interfaces": {
                "interface": [
                  {
                    "name": "eth0",
                    "ietf-isis-sr-mpls:segment-routing": {
                      "adjacency-sid": {
                        "adj-sids": [
                          {
                            "value": 38888,
                            "neighbor-system-id": "4444.5555.6666"
                          }
                        ]
                      }
                    }
                  }
                ]
              },
              "ietf-isis-sr-mpls:segment-routing": {
                "enabled": true
              },
              "ietf-isis-sr-mpls:protocol-srgb": {
                "srgb": [
                  {
                    "lower-bound": 4000,
                    "upper-bound": 5000
                  }
                ]
              }
            }
          }
        ]
      }
    }
  }
```

Appendix B. IS-IS MPLS Segment Routing Module Tree

The figure below describes the overall structure of the ietf-isis-sr-mpls YANG module:

```
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis:
     +--rw segment-routing
     |  +--rw enabled?    boolean
     |  +--rw bindings {mapping-server}?
     |     +--rw advertise
     |     |  +--rw policies*   leafref
     |     +--rw receive?     boolean
     +--rw protocol-srgb {sr-mpls:protocol-srgb}?
        +--rw srgb* [lower-bound upper-bound]
           +--rw lower-bound    uint32
           +--rw upper-bound    uint32
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:interfaces
             /isis:interface:
     +--rw segment-routing
        +--rw adjacency-sid
           +--rw adj-sids* [value]
           |  +--rw value-type?           enumeration
           |  +--rw value                 uint32
           |  +--rw protected?            boolean
           |  +--rw weight?               uint8
           |  +--rw neighbor-system-id    isis:system-id
           +--rw advertise-adj-group-sid* [group-id]
           |  +--rw group-id    uint32
           +--rw advertise-protection?      enumeration
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:interfaces
             /isis:interface/isis:fast-reroute/isis:lfa:
     +--rw ti-lfa {ti-lfa}?
        +--rw enabled?                  boolean
        +--rw selection-tie-breakers
           +--rw node-protection!
           |  +--rw priority?   uint8
           +--rw srlg-disjoint!
              +--rw priority?   uint8
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:interfaces
             /isis:interface/isis:fast-reroute/isis:lfa/isis:level-1:
     +--rw ti-lfa {ti-lfa}?
        +--rw enabled?   boolean
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:interfaces
             /isis:interface/isis:fast-reroute/isis:lfa/isis:level-2:
     +--rw ti-lfa {ti-lfa}?
        +--rw enabled?   boolean
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:interfaces
```

```
            /isis:interface/isis:fast-reroute/isis:lfa
            /isis:remote-lfa:
      +--rw use-segment-routing-path?   boolean {remote-lfa-sr}?
    augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/isis:isis/isis:interfaces
            /isis:interface/isis:adjacencies/isis:adjacency:
      +--ro adjacency-sid* []
         +--ro value?                  uint32
         +--ro address-family?         iana-rt-types:address-family
         +--ro weight?                 uint8
         +--ro protection-requested?   boolean
    augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/isis:isis/isis:database
            /isis:levels/isis:lsp/isis:router-capabilities
            /isis:router-capability:
      +--ro sr-capability
      |  +--ro sr-capability-flag*   identityref
      |  +--ro global-blocks
      |     +--ro global-block* []
      |        +--ro range-size?          rt-types:uint24
      |        +--ro (sid)?
      |           +--:(sid-label)
      |           |  +--ro label-value?   uint32
      |           +--:(sid-index)
      |              +--ro index-value?   uint32
      +--ro sr-algorithms
      |  +--ro sr-algorithm*   identityref
      +--ro local-blocks
      |  +--ro local-block* []
      |     +--ro range-size?          rt-types:uint24
      |     +--ro (sid)?
      |        +--:(sid-label)
      |        |  +--ro label-value?   uint32
      |        +--:(sid-index)
      |           +--ro index-value?   uint32
      +--ro srms-preference
         +--ro preference?   uint8
    augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/isis:isis/isis:database
            /isis:levels/isis:lsp/isis:extended-is-neighbor
            /isis:neighbor/isis:instances/isis:instance:
      +--ro adj-sid-sub-tlvs
         +--ro adj-sid-sub-tlv* []
            +--ro adj-sid-flags
            |  +--ro flag*   identityref
            +--ro weight?              uint8
            +--ro neighbor-id?         isis:system-id
            +--ro (sid)?
```

```
            +--:(sid-label)
            |  +--ro label-value?   uint32
            +--:(sid-index)
               +--ro index-value?   uint32
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:database
             /isis:levels/isis:lsp/isis:mt-is-neighbor/isis:neighbor
             /isis:instances/isis:instance:
     +--ro adj-sid-sub-tlvs
        +--ro adj-sid-sub-tlv* []
           +--ro adj-sid-flags
           |  +--ro flag*   identityref
           +--ro weight?              uint8
           +--ro neighbor-id?         isis:system-id
           +--ro (sid)?
              +--:(sid-label)
              |  +--ro label-value?   uint32
              +--:(sid-index)
                 +--ro index-value?   uint32
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:database
             /isis:levels/isis:lsp/isis:extended-ipv4-reachability
             /isis:prefixes:
     +--ro prefix-sid-sub-tlvs
        +--ro prefix-sid-sub-tlv* []
           +--ro prefix-sid-flags
           |  +--ro flag*   identityref
           +--ro algorithm?           identityref
           +--ro (sid)?
              +--:(sid-label)
              |  +--ro label-value?   uint32
              +--:(sid-index)
                 +--ro index-value?   uint32
   augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:database
             /isis:levels/isis:lsp/isis:mt-extended-ipv4-reachability
             /isis:prefixes:
     +--ro prefix-sid-sub-tlvs
        +--ro prefix-sid-sub-tlv* []
           +--ro prefix-sid-flags
           |  +--ro flag*   identityref
           +--ro algorithm?           identityref
           +--ro (sid)?
              +--:(sid-label)
              |  +--ro label-value?   uint32
              +--:(sid-index)
                 +--ro index-value?   uint32
   augment /rt:routing/rt:control-plane-protocols
```

```
            /rt:control-plane-protocol/isis:isis/isis:database
            /isis:levels/isis:lsp/isis:ipv6-reachability
            /isis:prefixes:
     +--ro prefix-sid-sub-tlvs
        +--ro prefix-sid-sub-tlv* []
           +--ro prefix-sid-flags
           |  +--ro flag*   identityref
           +--ro algorithm?           identityref
           +--ro (sid)?
              +--:(sid-label)
              |  +--ro label-value?   uint32
              +--:(sid-index)
                 +--ro index-value?   uint32
   augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/isis:isis/isis:database
            /isis:levels/isis:lsp/isis:mt-ipv6-reachability
            /isis:prefixes:
     +--ro prefix-sid-sub-tlvs
        +--ro prefix-sid-sub-tlv* []
           +--ro prefix-sid-flags
           |  +--ro flag*   identityref
           +--ro algorithm?           identityref
           +--ro (sid)?
              +--:(sid-label)
              |  +--ro label-value?   uint32
              +--:(sid-index)
                 +--ro index-value?   uint32
   augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/isis:isis/isis:database
            /isis:levels/isis:lsp:
     +--ro sid-binding-tlvs
     |  +--ro sid-binding-tlv* [prefix]
     |     +--ro prefix                 inet:ip-prefix
     |     +--ro range?                 uint16
     |     +--ro sid-binding-flags
     |     |  +--ro flag*   identityref
     |     +--ro prefix-sid-sub-tlvs* []
     |     |  +--ro prefix-sid-sub-tlvs
     |     |     +--ro prefix-sid-sub-tlv* []
     |     |        +--ro prefix-sid-flags
     |     |        |  +--ro flag*   identityref
     |     |        +--ro algorithm?           identityref
     |     |        +--ro (sid)?
     |     |           +--:(sid-label)
     |     |           |  +--ro label-value?   uint32
     |     |           +--:(sid-index)
     |     |              +--ro index-value?   uint32
     |     +--ro sid-sub-tlv* []
```

```
|     |  +--ro (sid)?
|     |     +--:(sid-label)
|     |     |  +--ro label-value?   uint32
|     |     +--:(sid-index)
|     |        +--ro index-value?   uint32
|     +--ro unknown-tlvs
|        +--ro unknown-tlv* []
|           +--ro type?     uint16
|           +--ro length?   uint16
|           +--ro value?    yang:hex-string
+--ro mt-sid-binding-tlvs
   +--ro mt-sid-binding-tlv* [prefix mt-id]
      +--ro prefix                inet:ip-prefix
      +--ro range?                uint16
      +--ro sid-binding-flags
      |  +--ro flag*   identityref
      +--ro prefix-sid-sub-tlvs* []
      |  +--ro prefix-sid-sub-tlvs
      |     +--ro prefix-sid-sub-tlv* []
      |        +--ro prefix-sid-flags
      |        |  +--ro flag*   identityref
      |        +--ro algorithm?           identityref
      |        +--ro (sid)?
      |           +--:(sid-label)
      |           |  +--ro label-value?   uint32
      |           +--:(sid-index)
      |              +--ro index-value?   uint32
      +--ro sid-sub-tlv* []
      |  +--ro (sid)?
      |     +--:(sid-label)
      |     |  +--ro label-value?   uint32
      |     +--:(sid-index)
      |        +--ro index-value?   uint32
      +--ro unknown-tlvs
      |  +--ro unknown-tlv* []
      |     +--ro type?     uint16
      |     +--ro length?   uint16
      |     +--ro value?    yang:hex-string
      +--ro mt-id                 uint16
```

Authors' Addresses

Stephane Litkowski
Cisco Systems
Email: slitkows.ietf@gmail.com

Yingzhen Qu
Futurewei Technologies
Email: yingzhen.ietf@gmail.com

Acee Lindem
LabN Consulting, L.L.C.
Email: acee.ietf@gmail.com

Ing-Wher Chen
The MITRE Corporation
Email: ingwherchen@mitre.org

Jeff Tantsura
Nvidia
Email: jefftant.ietf@gmail.com