IS-IS Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 25, 2021

S. Litkowski
Cisco Systems
Y. Qu
Futurewei
P. Sarkar
Individual
I. Chen
The MITRE Corporation
J. Tantsura
Apstra

February 21, 2021

YANG Data Model for IS-IS Segment Routing
draft-ietf-isis-sr-yang-10

Abstract

This document defines a YANG data module that can be used to configure and manage IS-IS Segment Routing, as well as a YANG data module for the management of Signaling Maximum SID Depth (MSD) Using IS-IS.

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 August 25, 2021.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Overview 2
  1.1. Requirements Language 3
  1.2. Tree Diagrams 3
2. IS-IS MSD 3
  2.1. IS-IS MSD YANG Module 4
3. IS-IS Segment Routing 8
  3.1. IS-IS Segment Routing configuration 11
    3.1.1. Segment Routing activation 11
    3.1.2. Advertising mapping server policy 11
    3.1.3. IP Fast reroute 11
  3.2. IS-IS Segment Routing YANG Module 11
4. Security Considerations 26
5. Contributors 27
6. Acknowledgements 27
7. IANA Considerations 27
8. Normative References 28
Authors' Addresses 30

1. Overview

YANG [RFC6020] [RFC7950] is a data definition language used to define the contents of a conceptual data store that allows networked devices to be managed using NETCONF [RFC6241]. YANG is proving relevant beyond its initial confines, as bindings to other interfaces (e.g., ReST) and encodings other than XML (e.g., JSON) are being defined. Furthermore, YANG data models can be used as the basis for implementation of other interfaces, such as CLI and programmatic APIs.

This document defines a YANG data module that can be used to configure and manage IS-IS Segment Routing [RFC8667] and it is an augmentation to the IS-IS YANG data model.

This document also defines a YANG data module for the management of Signaling Maximum SID Depth (MSD) Using IS-IS [RFC8491], which augments the base IS-IS YANG data model.

The YANG modules in this document conform 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 models defined in [RFC8340].

2. IS-IS MSD

This document defines a model for Signaling Maximum SID Depth (MSD) Using IS-IS[RFC8667]. It is an augmentation of the IS-IS base model.

The figure below describes the overall structure of the isis-msd YANG module:

```
module: ietf-isis-msd
  augment /rt:routing/rt:control-plane-protocols
          /rt:control-plane-protocol/isis:isis/isis:database
          /isis:levels/isis:lsp/isis:router-capabilities:
    +--ro node-msd-tlv
       +--ro node-msds* [msd-type]
          +--ro msd-type     identityref
          +--ro msd-value?   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:
    +--ro link-msd-sub-tlv
       +--ro link-msds* [msd-type]
          +--ro msd-type     identityref
          +--ro msd-value?   uint8
  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:
    +--ro link-msd-sub-tlv
       +--ro link-msds* [msd-type]
          +--ro msd-type     identityref
          +--ro msd-value?   uint8
```

2.1. IS-IS MSD YANG Module

```
<CODE BEGINS> file "ietf-isis-msd@2021-02-21.yang"
module ietf-isis-msd {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-isis-msd";
  prefix isis-msd;

  import ietf-routing {
    prefix rt;
    reference "RFC 8349: A YANG Data Model for Routing
               Management (NMDA Version)";
  }

  import ietf-isis {
    prefix isis;
  }

  organization
    "IETF LSR - LSR Working Group";
  contact
    "WG Web:   <https://tools.ietf.org/wg/mpls/>
     WG List:  <mailto:mpls@ietf.org>

     Author:    Yingzhen Qu
               <mailto:yingzhen.qu@futurewei.com>
     Author:    Acee Lindem
               <mailto:acee@cisco.com>
     Author:    Stephane Litkowski
               <mailto:slitkows.ietf@gmail.com>
     Author:    Jeff Tantsura
               <jefftant.ietf@gmail.com>

    ";
  description
    "The YANG module augments the base ISIS model to
     manage different types of MSDs.

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

     Copyright (c) 2021 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 Simplified 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;
        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.";


     reference "RFC XXXX: YANG Data Model for OSPF MSD.";

     revision 2021-02-21 {
       description
         "Initial Version";
       reference "RFC XXXX: YANG Data Model for ISIS MSD.";
     }

     identity msd-base-type {
       description
         "Base identity for MSD Type";
     }

     identity base-mpls-msd {
       base msd-base-type;
       description
         "Base MPLS Imposition MSD.";
       reference
         "RFC 8491: Singling MSD using IS-IS.";
     }

     identity erld-msd {
       base msd-base-type;
       description
         "ERLD-MSD is defined to advertise the ERLD.";
       reference
         "RFC 8662: Entropy Label for Source Packet Routing in
                    Networking (SPRING) Tunnels";
     }

     grouping node-msd-tlv {
       description
         "Grouping for node MSD.";
       container node-msd-tlv {
         list node-msds {
```

```
          key "msd-type";
          leaf msd-type {
            type identityref {
            base msd-base-type;
            }
            description
              "MSD-Types";
          }
          leaf msd-value {
            type uint8;
            description
              "MSD value, in the range of 0-255.";
          }
          description
            "Node MSD is the smallest link MSD supported by
             the node.";
        }
        description
          "Node MSD is the number of SIDs supported by a node.";
        reference
          "RFC 8476: Signaling Maximum SID Depth (MSD) Using OSPF";
      }
    }

    grouping link-msd-sub-tlv {
      description
        "Link Maximum SID Depth (MSD) grouping for an interface.";
      container link-msd-sub-tlv {
        list link-msds {
          key "msd-type";
          leaf msd-type {
            type identityref {
              base msd-base-type;
            }
            description
              "MSD-Types";
          }
          leaf msd-value {
            type uint8;
            description
              "MSD value, in the range of 0-255.";
          }
          description
            "List of link MSDs";
        }
        description
          "Link MSD sub-tlvs.";
      }
```

```
     }

     /* Node MSD TLV */
     augment "/rt:routing/" +
             "rt:control-plane-protocols/rt:control-plane-protocol"+
             "/isis:isis/isis:database/isis:levels/isis:lsp"+
             "/isis:router-capabilities" {
       when "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment ISIS routing protocol when used";
       }
       description
         "This augments ISIS protocol LSDB router capability.";
       uses node-msd-tlv;
     }


     /* link MSD sub-tlv */
     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" {
       when "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment ISIS routing protocol when used";
       }
       description
         "This augments ISIS protocol LSDB neighbor with
          Link MSD sub-TLV.";

       uses link-msd-sub-tlv;
     }

     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" {
       when "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment ISIS routing protocol when used";
       }
       description
         "This augments ISIS protocol LSDB neighbor.";

       uses link-msd-sub-tlv;
```

```
    }
  }
  <CODE ENDS>
```

3. IS-IS Segment Routing

This document defines a model for IS-IS Segment Routing feature. It is an augmentation of the IS-IS base model.

The IS-IS SR YANG module requires support for the base segment routing module [I-D.ietf-spring-sr-yang], which defines the global segment routing configuration independent of any specific routing protocol configuration, and support of IS-IS base model [I-D.ietf-isis-yang-isis-cfg] which defines basic IS-IS configuration and state.

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

```
module: ietf-isis-sr
  augment /rt:routing/rt:control-plane-protocols
          /rt:control-plane-protocol/isis:isis:
    +--rw segment-routing
    |  +--rw enabled?    boolean
    |  +--rw bindings
    |     +--rw advertise
    |     |  +--rw policies*   string
    |     +--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 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:
    +--rw ti-lfa {ti-lfa}?
```

```
       +--rw enable?   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* [value]
       +--ro af?                    iana-rt-types:address-family
       +--ro value                  uint32
       +--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:
    +--ro sr-capability
    |  +--ro sr-capability
    |  |  +--ro sr-capability-bits*   identityref
    |  +--ro global-blocks
    |     +--ro global-block* []
    |        +--ro range-size?    uint32
    |        +--ro sid-sub-tlv
    |           +--ro sid?   uint32
    +--ro sr-algorithms
    |  +--ro sr-algorithm*   uint8
    +--ro local-blocks
    |  +--ro local-block* []
    |     +--ro range-size?    uint32
    |     +--ro sid-sub-tlv
    |        +--ro sid?   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:
    +--ro sid-list* [value]
       +--ro adj-sid-flags
       |  +--ro bits*   identityref
       +--ro weight?          uint8
       +--ro neighbor-id?     isis:system-id
       +--ro 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:
    +--ro sid-list* [value]
       +--ro adj-sid-flags
       |  +--ro bits*   identityref
```

```
      +--ro weight?          uint8
      +--ro neighbor-id?     isis:system-id
      +--ro 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 sid-list* [value]
      +--ro perfix-sid-flags
      |  +--ro bits*   identityref
      +--ro algorithm?          uint8
      +--ro 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 sid-list* [value]
      +--ro perfix-sid-flags
      |  +--ro bits*   identityref
      +--ro algorithm?          uint8
      +--ro 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 sid-list* [value]
      +--ro perfix-sid-flags
      |  +--ro bits*   identityref
      +--ro algorithm?          uint8
      +--ro 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 sid-list* [value]
      +--ro perfix-sid-flags
      |  +--ro bits*   identityref
      +--ro algorithm?          uint8
      +--ro value               uint32
  augment /rt:routing/rt:control-plane-protocols
          /rt:control-plane-protocol/isis:isis/isis:database
          /isis:levels/isis:lsp:
    +--ro segment-routing-bindings* [fec range]
      +--ro fec                  string
      +--ro range                uint16
      +--ro sid-binding-flags
      |  +--ro bits*   identityref
      +--ro binding
         +--ro prefix-sid
            +--ro sid-list* [value]
```

```
               +--ro perfix-sid-flags
               |  +--ro bits*   identityref
               +--ro algorithm?          uint8
               +--ro value               uint32
```

3.1. IS-IS Segment Routing configuration

3.1.1. Segment Routing activation

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

3.1.2. Advertising mapping server policy

The base segment routing module defines mapping server policies. By default, IS-IS will not advertise nor receive any mapping server entry. The IS-IS segment-routing module allows to advertise one or multiple mapping server policies through the "bindings/advertise/policies" leaf-list. The "bindings/receive" leaf allows to enable the reception of mapping server entries.

3.1.3. IP Fast reroute

IS-IS SR 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.

3.2. IS-IS Segment Routing YANG Module

```
<CODE BEGINS> file "ietf-isis-sr@2021-01-11.yang"
module ietf-isis-sr {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:"
          + "yang:ietf-isis-sr";
  prefix isis-sr;


  import ietf-routing {
    prefix "rt";
    reference "RFC 8349 - A YANG Data Model for Routing
               Management (NMDA Version)";
  }

  import ietf-segment-routing-common {
    prefix "sr-cmn";
```

```
     }

     import ietf-segment-routing-mpls {
       prefix "sr-mpls";
     }

     import ietf-isis {
       prefix "isis";
     }

     import iana-routing-types {
       prefix "iana-rt-types";
       reference "RFC 8294 - Common YANG Data Types for the
                  Routing Area";
     }

     organization
      "IETF LSR - LSR Working Group";

     contact
       "WG List:  <mailto:lsr@ietf.org>

       Editor:    Stephane Litkowski
                  <mailto:stephane.litkowski@orange.com>

       Author:    Acee Lindem
                  <mailto:acee@cisco.com>
       Author:    Yingzhen Qu
                  <mailto:yingzhen.qu@futurewei.com>
       Author:    Pushpasis Sarkar
                  <mailto:pushpasis.ietf@gmail.com>
       Author:    Ing-Wher Chen
                  <mailto:ingwherchen@mitre.org>
       Author:    Jeff Tantsura
                  <mailto:jefftant.ietf@gmail.com>
       ";

     description
       "The YANG module defines a generic configuration model for
        Segment routing ISIS extensions common across all of the vendor
        implementations.

       This YANG model conforms to the Network Management
       Datastore Architecture (NMDA) as described in RFC 8242.

       Copyright (c) 2021 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 Simplified 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.";

    reference "RFC XXXX";

    revision 2021-01-11 {
    description
      "Initial revision.";
    reference "RFC XXXX";
    }


    /* Identities */
    identity sr-capability {
      description
        "Base identity for ISIS SR-Capabilities sub-TLV flgs";
    }

    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-bit {
       description
         "Base identity for prefix sid sub-tlv bits.";
     }

     identity r-bit {
       base prefix-sid-bit;
       description
        "Re-advertisement Flag.";
     }

     identity n-bit {
       base prefix-sid-bit;
       description
        "Node-SID Flag.";
     }

     identity p-bit {
       base prefix-sid-bit;
       description
        "No-PHP (No Penultimate Hop-Popping) Flag.";
     }

     identity e-bit {
       base prefix-sid-bit;
       description
        "Explicit NULL Flag.";
     }

     identity v-bit {
       base prefix-sid-bit;
       description
        "Value Flag.";
     }

     identity l-bit {
       base prefix-sid-bit;
       description
        "Local Flag.";
     }

     identity adj-sid-bit {
       description
         "Base identity for adj sid sub-tlv bits.";
     }
```

```
identity f-bit {
  base adj-sid-bit;
  description
    "Address-Family flag.";
}

identity b-bit {
  base adj-sid-bit;
  description
    "Backup flag.";
}

identity vi-bit {
  base adj-sid-bit;
  description
    "Value/Index flag.";
}

identity lo-bit {
  base adj-sid-bit;
  description
    "Local flag.";
}

identity s-bit {
  base adj-sid-bit;
  description
    "Group flag.";
}

identity pe-bit {
  base adj-sid-bit;
  description
    "Persistent flag.";
}

identity sid-binding-bit {
  description
    "Base identity for sid binding tlv bits.";
}

identity af-bit {
  base sid-binding-bit;
  description
    "Address-Family flag.";
}

identity m-bit {
```

```
      base sid-binding-bit;
      description
        "Mirror Context flag.";
    }

    identity sf-bit {
      base sid-binding-bit;
      description
        "S flag. If set, the binding label tlv should be flooded
         across the entire routing domain.";
    }

    identity d-bit {
      base sid-binding-bit;
      description
        "Leaking flag.";
    }

    identity a-bit {
      base sid-binding-bit;
      description
        "Attached flag.";
    }

    /* Features */

    feature remote-lfa-sr {
      description
       "Enhance rLFA to use SR path.";
    }

    feature ti-lfa {
      description
       "Enhance IPFRR with ti-lfa
       support";
    }

    /* Groupings */


    grouping sid-sub-tlv {
      description "SID/Label sub-TLV grouping.";
      container sid-sub-tlv {
        description
          "Used to advertise the SID/Label associated with a
        prefix or adjacency.";
        leaf sid {
          type uint32;
```

```
        description
          "Segment Identifier (SID) - A 20 bit label or
           32 bit SID.";
        }
      }
    }

    grouping sr-capability {
      description
        "SR capability grouping.";
      container sr-capability {
        description
          "Segment Routing capability.";
        container sr-capability {
          leaf-list sr-capability-bits {
            type identityref {
              base sr-capability;
            }
            description "SR Capbility sub-tlv flags list.";
          }
          description
            "SR Capability Flags.";
        }
        container global-blocks {
          description
            "Segment Routing Global Blocks.";
          list global-block {
            description "Segment Routing Global Block.";
            leaf range-size {
              type uint32;
              description "The SID range.";
            }
            uses sid-sub-tlv;
          }
        }
      }
    }

    grouping sr-algorithm {
      description
        "SR algorithm grouping.";
      container sr-algorithms {
        description "All SR algorithms.";
        leaf-list sr-algorithm {
          type uint8;
          description
            "The Segment Routing (SR) algorithms that the router is
             currently using.";
```

```
        }
      }
    }

    grouping srlb {
      description
        "SR Local Block grouping.";
      container local-blocks {
        description "List of SRLBs.";
        list local-block {
          description "Segment Routing Local Block.";
          leaf range-size {
            type uint32;
            description "The SID range.";
          }
          uses sid-sub-tlv;
        }
      }
    }

    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.";
      container srms-preference {
        description "SRMS Preference TLV.";
        leaf preference {
          type uint8 {
            range "0 .. 255";
          }
          description "SRMS preference TLV, vlaue from 0 to 255.";
        }
      }
    }

    grouping adjacency-state {
      description
        "This group will extend adjacency state.";
      list adjacency-sid {
        key value;
        config false;
        leaf af {
          type iana-rt-types:address-family;
          description
            "Address-family associated with the
             segment ID";
        }
        leaf value {
```

```
          type uint32;
          description
            "Value of the Adj-SID.";
        }
        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.";
        }
        description
          "List of adjacency Segment IDs.";
      }
    }

    grouping prefix-segment-id {
      description
        "This group defines segment routing extensions
         for prefixes.";

      list sid-list {
        key value;

        container perfix-sid-flags {
          leaf-list bits {
            type identityref {
              base prefix-sid-bit;
            }
            description
              "Prefix SID Sub-TLV flag bits list.";
          }
          description
            "Describes flags associated with the
             segment ID.";
        }

        leaf algorithm {
          type uint8;
          description
            "Algorithm to be used for path computation.";
        }
        leaf value {
```

```
            type uint32;
            description
             "Value of the prefix-SID.";
          }
          description
            "List of segments.";
        }
      }

      grouping adjacency-segment-id {
        description
          "This group defines segment routing extensions
           for adjacencies.";

        list sid-list {
          key value;

          container adj-sid-flags {
            leaf-list bits {
              type identityref {
                base adj-sid-bit;
              }
              description "Adj sid sub-tlv flags list.";
              }
            description "Adj-sid sub-tlv flags.";
          }

          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.";
          }
          leaf value {
            type uint32;
            description
              "Value of the Adj-SID.";
          }
          description
            "List of segments.";
        }
```

```
     }

     grouping segment-routing-binding-tlv {
       list segment-routing-bindings {
         key "fec range";

         leaf fec {
           type string;
           description
           "IP (v4 or v6) range to be bound to SIDs.";
         }

         leaf range {
           type uint16;
           description
             "Describes number of elements to assign
              a binding to.";
         }

         container sid-binding-flags {
           leaf-list bits {
             type identityref {
               base sid-binding-bit;
             }
             description
               "SID Binding TLV flag bits list.";
           }
           description
             "Binding flags.";
         }

         container binding {
           container prefix-sid {
             uses prefix-segment-id;
             description
               "Binding prefix SID to the range.";
           }
           description
             "Bindings associated with the range.";
         }

         description
           "This container describes list of SID/Label bindings.
            ISIS reference is TLV 149.";
       }
       description
         "Defines binding TLV for database.";
     }
```

```
/* Cfg */

augment "/rt:routing/" +
        "rt:control-plane-protocols/rt:control-plane-protocol"+
        "/isis:isis" {
  when "/rt:routing/rt:control-plane-protocols/"+
       "rt:control-plane-protocol/rt:type = 'isis:isis'" {
    description
      "This augment ISIS routing protocol when used";
  }
  description
    "This augments ISIS protocol configuration
     with segment routing.";

  uses sr-mpls:sr-control-plane;
  container protocol-srgb {
    if-feature sr-mpls:protocol-srgb;
    uses sr-cmn:srgb;
    description
      "Per-protocol SRGB.";
  }
}

augment "/rt:routing/" +
        "rt:control-plane-protocols/rt:control-plane-protocol"+
        "/isis:isis/isis:interfaces/isis:interface" {
  when "/rt:routing/rt:control-plane-protocols/"+
       "rt:control-plane-protocol/rt:type = 'isis:isis'" {
    description
      "This augment ISIS routing protocol when used";
  }
  description
    "This augments ISIS protocol configuration
     with segment routing.";

  uses sr-mpls:igp-interface;
}

augment "/rt:routing/" +
        "rt:control-plane-protocols/rt:control-plane-protocol"+
        "/isis:isis/isis:interfaces/isis:interface"+
        "/isis:fast-reroute" {
  when "/rt:routing/rt:control-plane-protocols/"+
       "rt:control-plane-protocol/rt:type = 'isis:isis'" {
    description
      "This augment ISIS routing protocol when used";
  }
  description
```

```
         "This augments ISIS IP FRR with TILFA.";

       container ti-lfa {
         if-feature ti-lfa;
         leaf enable {
           type boolean;
           description
             "Enables TI-LFA computation.";
         }
         description
           "TILFA configuration.";
       }
     }

     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 "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment 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;
         description
           "force remote LFA to use segment routing
            path instead of LDP path.";
       }
     }

     /* Operational states */

     augment "/rt:routing/" +
             "rt:control-plane-protocols/rt:control-plane-protocol"+
             "/isis:isis/isis:interfaces/isis:interface" +
             "/isis:adjacencies/isis:adjacency" {
       when "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment 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" {
       when "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment 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" {
       when "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment 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" {
       when "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment 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 "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment ISIS routing protocol when used";
       }
       description
         "This augments ISIS protocol LSDB prefix.";
          uses prefix-segment-id;
     }

     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 "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment ISIS routing protocol when used";
       }
       description
         "This augments ISIS protocol LSDB prefix.";
       uses prefix-segment-id;
     }

     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 "/rt:routing/rt:control-plane-protocols/"+
            "rt:control-plane-protocol/rt:type = 'isis:isis'" {
         description
           "This augment ISIS routing protocol when used";
       }
       description
         "This augments ISIS protocol LSDB prefix.";
       uses prefix-segment-id;
     }

     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 "/rt:routing/rt:control-plane-protocols/"+
           "rt:control-plane-protocol/rt:type = 'isis:isis'" {
        description
          "This augment ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB prefix.";
      uses prefix-segment-id;
    }

    augment "/rt:routing/" +
            "rt:control-plane-protocols/rt:control-plane-protocol"+
            "/isis:isis/isis:database/isis:levels/isis:lsp" {
      when "/rt:routing/rt:control-plane-protocols/"+
           "rt:control-plane-protocol/rt:type = 'isis:isis'" {
        description
          "This augment ISIS routing protocol when used";
      }
      description
        "This augments ISIS protocol LSDB.";
      uses segment-routing-binding-tlv;
    }


    /* Notifications */

  }
  <CODE ENDS>
```

4. Security Considerations

The YANG module specified in this document defines a schema for data that is designed to be accessed via network management protocols such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446].

The NETCONF Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a pre-configured 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). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations. These are the subtrees and data nodes and their sensitivity/vulnerability:

/isis:isis/segment-routing

/isis:isis/protocol-srgb

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

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

/isis:router-capabilities/node-msd-tlv

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 this device.

5. Contributors

Authors would like to thank Derek Yeung, Acee Lindem, Yi Yang for their major contributions to the draft.

6. Acknowledgements

MITRE has approved this document for Public Release, Distribution Unlimited, with Public Release Case Number 19-3033.

7. IANA Considerations

The IANA is requested to assign two new URIs from the IETF XML registry ([RFC3688]). Authors are suggesting the following URI:

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

URI: urn:ietf:params:xml:ns:yang:ietf-isis-msd
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]) with the following suggestion :

name: ietf-isis-sr
namespace: urn:ietf:params:xml:ns:yang:ietf-isis-sr
prefix: isis-sr
reference: RFC XXXX

name: ietf-isis-msd
namespace: urn:ietf:params:xml:ns:yang:ietf-isis-msd
prefix: isis-msd
reference: RFC XXXX

8. Normative References

[I-D.ietf-isis-yang-isis-cfg]
Litkowski, S., Yeung, D., Lindem, A., Zhang, Z., and L. Lhotka, "YANG Data Model for IS-IS Protocol", draft-ietf-isis-yang-isis-cfg-42 (work in progress), October 2019.

[I-D.ietf-spring-sr-yang]
Litkowski, S., Qu, Y., Sarkar, P., and J. Tantsura, "YANG Data Model for Segment Routing", draft-ietf-spring-sr-yang-15 (work in progress), December 2017.

[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>.

[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>.

[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>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/info/rfc6242>.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, DOI 10.17487/RFC6536, March 2012, <https://www.rfc-editor.org/info/rfc6536>.

[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>.

[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>.

[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>.

[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>.

[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>.

[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>.

[RFC8491] Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg, "Signaling Maximum SID Depth (MSD) Using IS-IS", RFC 8491, DOI 10.17487/RFC8491, November 2018, <https://www.rfc-editor.org/info/rfc8491>.

[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>.

Authors' Addresses

Stephane Litkowski
Cisco Systems

Email: slitkows.ietf@gmail.com

Yinghzen Qu
Futurewei

Email: yingzhen.qu@futurewei.com

Pushpasis Sarkar
Individual

Email: pushpasis.ietf@gmail.com

Ing-Wher Chen
The MITRE Corporation

Email: ingwherchen@mitre.org

Jeff Tantsura
Apstra

Email: jefftant.ietf@gmail.com