Internet
Internet-Draft
Intended status: Standards Track
Expires: December 23, 2021

A. Lindem
S. Litkowski
Cisco Systems
Y. Qu
Futurewei
June 21, 2021

IS-IS YANG Model Augmentations for Additional Features - Version 1
draft-ietf-lsr-isis-yang-augmentation-v1-01

Abstract

This document defines YANG data modules augmenting the IETF IS-IS YANG model to provide support for IS-IS Minimum Remaining Lifetime as defined in RFC 7987, ,IS-IS Application-Specific Link Attributes as defined in RFC 8919, and IS-IS Flexible Algorithm.

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 December 23, 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 diagram 3
2. YANG Module for IS-IS Minimum Remaining Lifetime 3
3. YANG Module for IS-IS Application-Specific Link Attributes . 5
4. YANG Module for IS-IS Flexible Algorithm 14
5. Security Considerations 25
6. IANA Considerations 26
7. Acknowledgements 27
8. References 27
8.1. Normative References 27
8.2. Informative References 29
Authors' Addresses 29

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 YANG data modules augmenting the IETF IS-IS YANG model [I-D.ietf-isis-yang-isis-cfg], which itself augments [RFC8349], to provide support for configuration and operational state for the following IS-IS features:

RFC7987: IS-IS Minimum Remaining Lifetime[RFC7987].

RFC8919: IS-IS Application-Specific Link Attributes[RFC8919].

RFCxxxx: IGP Flexible Algorithm [I-D.ietf-lsr-flex-algo].

The augmentations defined in this document requires support for the IS-IS base model[I-D.ietf-isis-yang-isis-cfg] which defines basic IS-IS configuration and state. The IS-IS YANG model augments the ietf-routing YANG model defined in [RFC8349].

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 diagram

Tree diagrams used in this document follow the notation defined in [RFC8340].

2. YANG Module for IS-IS Minimum Remaining Lifetime

This document defines a YANG module for IS-IS Minimum Remaining Lifetime as defined in [RFC7987]. It is an augmentation of the IS-IS base model.

```
module: ietf-isis-remaining-lifetime

  notifications:
    +---n corrupt-remaining-lifetime
       +--ro routing-protocol-name?   -> /rt:routing
                                          /control-plane-protocols
                                          /control-plane-protocol/name
       +--ro isis-level?              level
       +--ro lsp-id?                  isis:lsp-id
```

```
  <CODE BEGINS> file "ietf-isis-remaining-lifetime@2021-01-05.yang"
module ietf-isis-remaining-lifetime {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-isis-remaining-lifetime";

  prefix isis-remaining-lifetime;

  import ietf-isis {
    prefix "isis";
  }

  organization
    "IETF LSR - Link State Routing Working Group";

  contact
     "WG Web:   <http://tools.ietf.org/wg/lsr>
      WG List:  <mailto:lsr@ietf.org>
```

```
       Author:   Yingzhen Qu
                 <mailto:yqu@futurewei.com>
       Author:   Acee Lindem
                 <mailto:acee@cisco.com>
       Author:   Stephane Litkowski
                 <mailto:slitkows.ietf@gmail.com>";

   description
     "This YANG module defines the configuration and operational
      state for IS-IS Minimum Remaining Lifetime feature as defined
      in RFC 7987.

      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
      (http://trustee.ietf.org/license-info).

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

   reference "RFC XXXX";

   revision 2021-01-05 {
     description
       "Initial version";
     reference
       "RFC XXXX: A YANG Data Model for IS-IS.";
   }

   notification corrupt-remaining-lifetime {
     uses isis:notification-instance-hdr;
     leaf lsp-id {
       type isis:lsp-id;
       description "LSP ID";
     }
     description
       "This notification is sent when the system
        detects correpted lifetime of an LSP.";
     reference "RFC 7987: IS-IS Minimum Remaining Lifetime";
   }
 }
   <CODE ENDS>
```

3. YANG Module for IS-IS Application-Specific Link Attributes

This document defines a YANG module for IS-IS Application-Specific Link Attributes [RFC8919]. It is an augmentation of the IS-IS base model.

```
module: ietf-isis-link-attr
  augment /rt:routing/rt:control-plane-protocols
          /rt:control-plane-protocol/isis:isis/isis:interfaces
          /isis:interface:
    +--rw isis-link-attr
       +--rw (link-attr-op-mode)
          +--:(legacy)
          |  +--rw legacy?         empty
          +--:(transition)
          |  +--rw transition?     empty
          +--:(app-specific)
             +--rw app-specific?   empty
  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 application-specific-link-attributes-sub-tlvs
       +--ro asla-sub-tlvs* []
          +--ro l-flag?         boolean
          +--ro sabm-length?    uint8
          +--ro r-flag?         boolean
          +--ro udabm-length?   uint8
          +--ro sabm
          |  +--ro sabm-bits*   identityref
          +--ro udabm
          +--ro unknown-tlvs
             +--ro unknown-tlv* []
                +--ro type?     uint16
                +--ro length?   uint16
                +--ro value?    yang:hex-string
  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 application-specific-link-attributes-sub-tlvs
       +--ro asla-sub-tlvs* []
          +--ro l-flag?         boolean
          +--ro sabm-length?    uint8
          +--ro r-flag?         boolean
          +--ro udabm-length?   uint8
          +--ro sabm
          |  +--ro sabm-bits*   identityref
```

```
            +--ro udabm
            +--ro unknown-tlvs
               +--ro unknown-tlv* []
                  +--ro type?     uint16
                  +--ro length?   uint16
                  +--ro value?    yang:hex-string
     augment /rt:routing/rt:control-plane-protocols
             /rt:control-plane-protocol/isis:isis/isis:database
             /isis:levels/isis:lsp:
       +--ro application-specific-srlg-tlv
          +--ro as-srlg-tlvs* []
             +--ro neighbor-system-id?   isis:system-id
             +--ro pseudo-node-id?       uint8
             +--ro l-flag?               boolean
             +--ro sabm-length?          uint8
             +--ro r-flag?               boolean
             +--ro udabm-length?         uint8
             +--ro sabm
             |  +--ro sabm-bits*   identityref
             +--ro udabm
             +--ro length-of-sub-tlvs?   uint8
             +--ro link-id-sub-tlvs
             |  +--ro link-local-remote-ids
             |  |  +--ro link-local-id?    union
             |  |  +--ro link-remote-id?   union
             |  +--ro ipv4-interface-addr
             |  |  +--ro ipv4-int-addr?   inet:ipv4-address
             |  +--ro ipv4-neighbor-addr
             |  |  +--ro ipv4-neighbor-addr?   inet:ipv4-address
             |  +--ro ipv6-interface-addr
             |  |  +--ro ipv6-int-addr?   inet:ipv6-address
             |  +--ro ipv6-neighbor-addr
             |     +--ro ipv6-neighbor-addr?   inet:ipv6-address
             +--ro srlgs
                +--ro srlg*   uint32
```

```
  <CODE BEGINS> file "ietf-isis-link-attr@2021-01-05.yang"
module ietf-isis-link-attr {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-isis-link-attr";

  prefix isis-link-attr;

  import ietf-inet-types {
    prefix inet;
    reference "RFC 6991 - Common YANG Data Types";
  }
```

```
  import ietf-routing {
    prefix "rt";
  }

  import ietf-isis {
    prefix "isis";
  }

  organization
    "IETF LSR - Link State Routing Working Group";

  contact
     "WG Web:   <http://tools.ietf.org/wg/lsr>
      WG List:  <mailto:lsr@ietf.org>

      Author:   Yingzhen Qu
                <mailto:yqu@futurewei.com>
      Author:   Acee Lindem
                <mailto:acee@cisco.com>
      Author:   Stephane Litkowski
                <mailto:slitkows.ietf@gmail.com>";

  description
    "This YANG module defines the configuration and operational
     state for IS-IS application specific link attributes feature as
     defined in RFC 8919.

     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
     (http://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";

  revision 2021-01-05 {
    description
      "Initial version";
    reference
      "RFC XXXX: A YANG Data Model for IS-IS.";
  }

  identity sabm-bit {
    description
      "Base identity for sabm bits.";
    reference "RFC 8919: IS-IS Application-Specific Link Attributes";
  }

  identity rsvp-te-bit {
    base sabm-bit;
    description
      "R bit, RSVP-TE.";
  }

  identity sr-policy-bit {
    base sabm-bit;
    description
      "S bit, Segment Routing Policy.";
  }

  identity lfa-bit {
    base sabm-bit;
    description
      "F bit, Loop Free Alternate (LFA). Includes all LFA types.";
  }

  grouping application-identifier-bit-mask {
    description
      "Identification of the set of applications associated with link
       attribute advertisements";

    leaf l-flag {
      type boolean;
      description
        "Legacy Flag. When set, all of the applications
         specified in the bit mask MUST use the legacy
         advertisements.";
    }
    leaf sabm-length {
      type uint8;
      description
```

```
        "Standard Application Identifier Bit Mask Length in
         octets.";
    }
    leaf r-flag {
      type boolean;
      default false;
      description
        "Reserved.";
    }
    leaf udabm-length {
      type uint8;
      description
      "User Defined Application Identifier Bit Mask Length
      in octets.";
    }
    container sabm {
      leaf-list sabm-bits {
        type identityref {
          base sabm-bit;
        }
        description
          "SABM bits list. This list will contain
           identities for the bits which are set in the
           SABA bits.";
      }
      description
        "Standard Application Identifier Bit Mask.";
    }
    container udabm {
      description
        "User Defined Application Identifier Bit Mask.
         This container is to be augmented by user defined
         applications.";
    }
  }

  grouping application-specific-link-attributes-sub-tlv {
    description
      "Grouping for specification of the applications and
       application-specific attribute values.";

    container application-specific-link-attributes-sub-tlvs {
      list asla-sub-tlvs {
        uses application-identifier-bit-mask;
        uses isis:unknown-tlvs;
        description
          "List of application specific link attributes sub-tlvs.";
      }
```

```
      description
        "Application specific link attributes sub-tlv.";
    }
  }

  grouping application-specific-srlg-tlv {
    description
      "Grouping of a TLV to advertise application-specific
       SRLGs for a given link.";
    container application-specific-srlg-tlv {
      list as-srlg-tlvs {
        leaf neighbor-system-id {
          type isis:system-id;
          description
            "Neighbor System-ID.";
        }
        leaf pseudo-node-id {
          type uint8;
          description
            "Pseudo-node ID.";
        }
        uses application-identifier-bit-mask;
        leaf length-of-sub-tlvs {
          type uint8;
          description
            "Length of sub-tlvs.";
        }

        container link-id-sub-tlvs {
          description
            "Link Identifier sub-TLVs.";
          container link-local-remote-ids {
            description
              "Link local/remote identifier sub-tlv.";
            leaf link-local-id {
              type union {
                type inet:ipv4-address;
                type uint32;
              }
              description
                "Local identifier of the link.
                 It could be an IPv4 address or a local identifier.";
            }
            leaf link-remote-id {
              type union {
                type inet:ipv4-address;
                type uint32;
              }
```

```
              description
                "Remote identifier of the link.
                It could be an IPv4 address or a remotely learned
                identifier.";
            }
          }
          container ipv4-interface-addr {
            leaf ipv4-int-addr {
              type inet:ipv4-address;
              description
                "IPv4 address for the interface.";
            }
            description
              "IPv4 interface address sub-tlv.";
          }
          container ipv4-neighbor-addr {
            leaf ipv4-neighbor-addr {
              type inet:ipv4-address;
              description
                "IPv4 address for a neighboring router
                on this link.";
            }
            description
              "IPv4 neighbor address sub-tlv.";
          }
          container ipv6-interface-addr {
            leaf ipv6-int-addr {
              type inet:ipv6-address;
              description
                "IPv6 address for the interface.";
            }
            description
              "IPv6 interface address sub-tlv.";
          }
          container ipv6-neighbor-addr {
            leaf ipv6-neighbor-addr {
              type inet:ipv6-address;
              description
                "IPv6 address for a neighboring router
                on this link.";
            }
            description
              "IPv6 neighbor address sub-tlv.";
          }
        }

        container srlgs {
          description "List of SRLGs.";
```

```
          leaf-list srlg {
            type uint32;
            description
              "SRLG value of the link.";
          }
        }

        description
          "List of application specific SRLG tlvs.";
      }
      description
        "Application specific SRLG tlv.";
    }
  }

  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 TE attributes per application.";

    container isis-link-attr {
      choice link-attr-op-mode {
        mandatory "true";
        leaf legacy {
          type empty;
          description
            "Only send legacy advertisements.";
        }
        leaf transition {
          type empty;
          description
            "Send both application-specific and legacy advertisements.";
        }
        leaf app-specific{
          type empty;
          description
            "Only send application-specific advertisements.";
        }
        description
          "Link attributes mode";
      }
```

```
      description
        "Link attributes operation mode.";
    }
  }

  /* TLV 22 */
  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 "/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 TLV22.";

       uses application-specific-link-attributes-sub-tlv;
  }

  /* TLV 223 */
  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 "/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 TLV22.";

       uses application-specific-link-attributes-sub-tlv;
  }

  /* application-specific SRLG TLV 238 */
  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 router capability.";

    uses application-specific-srlg-tlv;
  }
}
  <CODE ENDS>
```

4. YANG Module for IS-IS Flexible Algorithm

This document defines a YANG module for IS-IS Flexible Algorithm [I-D.ietf-lsr-flex-algo]. It is an augmentation of the IS-IS base model.

```
module: ietf-isis-flex-algo
  augment /rt:routing/rt:control-plane-protocols
          /rt:control-plane-protocol/isis:isis:
    +--rw isis-flex-algo
       +--rw flex-algo* [algo-number]
          +--rw algo-number             uint8
          +--rw advertise-definition?   boolean
          +--rw admin-groups {te-types:extended-admin-groups,
                              te-types:named-extended-admin-groups}?
          |  +--rw exclude-admin-groups*   -> /te:te/globals
                                              /named-admin-groups
                                              /named-admin-group/name
          |  +--rw include-any-admin-groups*   -> /te:te/globals
                                              /named-admin-groups
                                              /named-admin-group/name
          |  +--rw include-all-admin-groups*   -> /te:te/globals
                                              /named-admin-groups
                                              /named-admin-group/name
          +--rw exclude-srlgs*   -> /te:te/globals
                                    /named-srlgs/named-srlg/name
                                    {te-types:named-srlg-groups}?
          +--rw fast-reroute?           boolean
          +--rw metric-type?            identityref
          +--rw microloop-avoidance?    boolean
          +--rw prefix-metric!
          +--rw priority?               uint8
  augment /rt:routing/rt:control-plane-protocols
          /rt:control-plane-protocol/isis:isis/isis:database
          /isis:levels/isis:lsp/isis:router-capabilities:
    +--ro fad-tlvs
       +--ro fad-tlv* []
          +--ro flex-algo?              uint8
          +--ro metric-type?            identityref
          +--ro calc-type?              uint8
```

```
             +--ro priority?              uint8
             +--ro fa-ex-ag-sub-tlv
             |  +--ro extended-admin-groups*   uint64
             +--ro fa-in-any-ag-sub-tlv
             |  +--ro extended-admin-groups*   uint64
             +--ro fa-in-all-ag-sub-tlv
             |  +--ro extended-admin-groups*   uint64
             +--ro fad-flags-sub-tlv
             |  +--ro fad-flags*   identityref
             +--ro fa-ex-srlg-sub-tlv
             |  +--ro srlgs*   uint32
             +--ro unknown-tlvs
                +--ro unknown-tlv* []
                   +--ro type?     uint16
                   +--ro length?   uint16
                   +--ro value?    yang:hex-string
     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 fapm-sub-tlvs
          +--ro fapm-sub-tlv* []
             +--ro flex-algo?   uint8
             +--ro metric?      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 fapm-sub-tlvs
          +--ro fapm-sub-tlv* []
             +--ro flex-algo?   uint8
             +--ro metric?      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 fapm-sub-tlvs
          +--ro fapm-sub-tlv* []
             +--ro flex-algo?   uint8
             +--ro metric?      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 fapm-sub-tlvs
          +--ro fapm-sub-tlv* []
             +--ro flex-algo?   uint8
             +--ro metric?      uint32
```

```
  <CODE BEGINS> file "ietf-isis-flex-algo@2021-06-18.yang"
module ietf-isis-flex-algo {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-isis-flex-algo";
  prefix isis-flex-algo;

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

  import ietf-te-types {
    prefix te-types;
    reference
      "RFC8776: Common YANG Data Types for Traffic Engineering.";
  }

  import ietf-isis {
    prefix "isis";
  }

  import ietf-te {
    prefix "te";
  }

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

  description
    "The YANG module defines the configuration and operational
     state for ISIS Flexible Algorithm as defined in RFC xxxx.

     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 ISIS Flexible Algorithm.";

  revision 2021-06-18 {
    description
      "Initial Version";
    reference "RFC XXXX: YANG Data Model for ISIS Flexible Algorithm.";
  }

  /* Identities */

  identity metric-type {
    description
      "Base identity for route metric types.";
  }

  identity igp-metric {
    base metric-type;
    description
      "Identity for the IGP metric type.";
  }

  identity min-uni-delay {
    base metric-type;
    description
      "Min unidirectional link delay metric type.";
    reference
      "RFC 8570 - IS-IS Traffic Engineering (TE) Metric Extensions";
  }
```

```
  identity te-metric {
    base metric-type;
    description
      "Traffic engineering metric type.";
    reference
      "RFC 5305 - IS-IS Extensions for Traffic Engineering (TE)";
  }

  identity fad-flags {
    description
      "Base identity for ISIS FAD flags.";
  }

  identity m-bit {
    base fad-flags;
    description
      "M bit, when set, the fex-algo specific prefix and ASBR
       metric MUST be used for inter-area and external prefix
       calculation.";
  }

  /* Groupings */
  grouping fa-ex-ag-sub-tlv {
    container fa-ex-ag-sub-tlv {
      leaf-list extended-admin-groups {
        type uint64;
        description
          "Extended administrative group as defined in RFC 7308.";
      }
      description
        "The flex-algo exclude admin group sub-tlv.";
    }
    description
      "The flex-algo exclude admin group sub-tlv.";
  }

  grouping fa-in-any-ag-sub-tlv {
    container fa-in-any-ag-sub-tlv {
      leaf-list extended-admin-groups {
        type uint64;
        description
          "Extended administrative group as defined in RFC 7308.";
      }
      description
        "The flex-algo include-any admin group sub-tlv.";
    }
    description
      "The flex-algo include-any admin group sub-tlv.";
```

```
  }

  grouping fa-in-all-ag-sub-tlv {
    container fa-in-all-ag-sub-tlv {
      leaf-list extended-admin-groups {
        type uint64;
        description
          "Extended administrative group as defined in RFC 7308.";
      }
      description
        "The flex-algo include-all admin group sub-tlv.";
    }
    description
      "The flex-algo include-all admin group sub-tlv.";
  }

  grouping fad-flags-sub-tlv {
    container fad-flags-sub-tlv {
      leaf-list fad-flags {
        type identityref {
          base fad-flags;
        }
        description
          "Flex-algo definition flags list.";
      }
      description
        "ISIS flex-algo definition flags.";
    }
    description
      "The flex-algo definition flags sub-tlv.";
  }

  grouping fa-ex-srlg-sub-tlv {
    container fa-ex-srlg-sub-tlv {
      leaf-list srlgs {
        type uint32;
        description
          "SRLG value as defined in RFC 4203.";
      }
      description
        "The flex-algo exclude SRLG sub-tlv.";
    }
    description
      "The flex-algo exclude SRLG sub-tlv.";
  }

  grouping fad-tlvs {
    container fad-tlvs {
```

```
      list fad-tlv {
        leaf flex-algo {
          type uint8;
          description
            "Flex-algo number, value between 128 and 255 inclusive.";
        }
        leaf metric-type {
          type identityref {
            base metric-type;
          }
            description
              "Type of metric to be used during the calculation.";
        }
        leaf calc-type {
          type uint8 {
            range "0..127";
          }
          description
            "IGP algorithm types, value from 0 to 127 as
            defined under 'Interior Gateway Protocol (IGP)
            Parameter' by IANA.";
        }
        leaf priority {
          type uint8;
            description
              "Priority of the advertisement.";
        }

        uses fa-ex-ag-sub-tlv;
        uses fa-in-any-ag-sub-tlv;
        uses fa-in-all-ag-sub-tlv;
        uses fad-flags-sub-tlv;
        uses fa-ex-srlg-sub-tlv;
        uses isis:unknown-tlvs;

        description
          "List of flex-algo definition TLVs.";
      }
      description
        "ISIS Flexible Algorithm Definition TLV.";
    }
    description
      "ISIS Flexible Algorithm Definition (FAD) TLV.";
  }

  grouping fapm-sub-tlvs {
    container fapm-sub-tlvs {
      list fapm-sub-tlv {
```

```
        leaf flex-algo {
          type uint8;
          description
            "Flex-algo number, value between 128 and 255
             inclusive.";
        }
        leaf metric {
          type uint32;
          description
            "Prefix metric.";
        }
        description
          "List of flex-algo prefix sub-tlvs.";
      }
      description
        "Flex-algo prefix metric sub-tlvs.";
    }
    description
      "Flexible Algorithm Prefix Metric (FAPM) sub TLVs.";
  }


  /* Configurations */

  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 flexible algorithm.";


    container isis-flex-algo {
      list flex-algo {
        key "algo-number";

        leaf algo-number {
          type uint8 {
            range "128..255";
          }
          description
            "An identifier in the range 128-255 that's associated
            with the Flexible Algorithm Definition.";
```

```
        }

        leaf advertise-definition {
          type boolean;
          default true;
          description
            "Enable to advertise the flex-algo definition.";
        }

        container admin-groups {
          if-feature "te-types:extended-admin-groups";
          if-feature "te-types:named-extended-admin-groups";
          leaf-list exclude-admin-groups {
            type leafref {
              path "/te:te/te:globals/te:named-admin-groups/"
                 + "te:named-admin-group/te:name";
            }
            description
              "Exclude rule used during the flex-algo
              path computation.";
          }
          leaf-list include-any-admin-groups {
            type leafref {
              path "/te:te/te:globals/te:named-admin-groups/"
                 + "te:named-admin-group/te:name";
            }
            description
              "Include-any rule used during the flex-algo
              path computation.";
          }
          leaf-list include-all-admin-groups {
            type leafref {
              path "/te:te/te:globals/te:named-admin-groups/"
                 + "te:named-admin-group/te:name";
            }
            description
              "Include-all rule used during the flex-algo
              path computation.";
          }
          description
            "Specify links for the flex-algo path computation.";
        }

        leaf-list exclude-srlgs {
          if-feature "te-types:named-srlg-groups";
          type leafref {
            path "/te:te/te:globals/te:named-srlgs/te:named-srlg/"
               + "te:name";
```

```
          }
          description
            "Shared Risk Link Groups (SRLGs) to be excluded during
            the flex-algo path computation.";
        }

        leaf fast-reroute {
          type boolean;
          default true;
          description
            "Enable fast reroute.";
        }

        leaf metric-type {
          type identityref {
            base metric-type;
          }
          description
            "Type of metric to be used during the calculation.";
        }

        leaf microloop-avoidance {
          type boolean;
          default true;
          description
            "Enable microloop avoidance.";
        }

        container prefix-metric {
          presence
            "Use flex-algo specific prefix metric.";
          description
            "Use flex-algo prefix metric.";
        }

        leaf priority {
          type uint8;
          description
            "Priority of the advertisement.";
        }

        description
          "List of flex-algo configurations.";
      }
      description
        "Flexible Algorithm configuration.";
    }
  }
```

```
  /* Database */

  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 fad-tlvs;
  }

  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 fapm-sub-tlvs;
  }

  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 fapm-sub-tlvs;
  }

  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 fapm-sub-tlvs;
  }

  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 fapm-sub-tlvs;
  }
}
    <CODE ENDS>
```

5. Security Considerations

The YANG modules specified in this document define 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 [RFC5246].

The NETCONF access control model [RFC6536] 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 the modules 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 correspond to the following schema nodes:

/isis:isis/isis:interfaces/isis:interface/isis-link-attr - Modification of link attributes operation mode could result in traffic being redirected or DoS attack.

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. The exposure of the Link State Database (LSDB) will expose the detailed topology of the network. This may be undesirable since both due to the fact that exposure may facilitate other attacks. Additionally, network operators may consider their topologies to be sensitive confidential data.These correspond to the following schema nodes:

/isis:isis/isis:database/isis:levels/isis:lsp/isis:mt-is-neighbor/isis:neighbor/isis:instances/isis:instance/application-specific-link-attributes-sub-tlvs

/isis:isis/isis:database/isis:levels/isis:lsp/application-specific-srlg-tlv

6. IANA Considerations

This document registers URIs in the IETF XML registry [RFC3688]. Following the format in [RFC3688], the following registrations is requested to be made:

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

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

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

This document registers the YANG modules in the YANG Module Names registry [RFC6020].

```
name: ietf-isis-remaining-lifetime
namespace: urn:ietf:params:xml:ns:yang:ietf-isis-remaining-lifetime
prefix: isis-remaining-lifetime
reference: RFC XXXX

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

 name: ietf-isis-flex-algo
 namespace: urn:ietf:params:xml:ns:yang:ietf-isis-flex-algo
 prefix: isis-flex-algo
 reference: RFC XXXX
```

7. Acknowledgements

This document was produced using Marshall Rose's xml2rfc tool.

The YANG model was developed using the suite of YANG tools written and maintained by numerous authors.

8. References

8.1. Normative References

[I-D.ietf-isis-yang-isis-cfg]
Litkowski, S., Yeung, D., Lindem, A., Zhang, J., 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-lsr-flex-algo]
Psenak, P., Hegde, S., Filsfils, C., Talaulikar, K., and A. Gulko, "IGP Flexible Algorithm", draft-ietf-lsr-flex-algo-15 (work in progress), April 2021.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

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

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <https://www.rfc-editor.org/info/rfc5246>.

[RFC5329] Ishiguro, K., Manral, V., Davey, A., and A. Lindem, Ed., "Traffic Engineering Extensions to OSPF Version 3", RFC 5329, DOI 10.17487/RFC5329, September 2008, <https://www.rfc-editor.org/info/rfc5329>.

[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>.

[RFC7987] Ginsberg, L., Wells, P., Decraene, B., Przygienda, T., and H. Gredler, "IS-IS Minimum Remaining Lifetime", RFC 7987, DOI 10.17487/RFC7987, October 2016, <https://www.rfc-editor.org/info/rfc7987>.

[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>.

[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>.

[RFC8919] Ginsberg, L., Psenak, P., Previdi, S., Henderickx, W., and J. Drake, "IS-IS Application-Specific Link Attributes", RFC 8919, DOI 10.17487/RFC8919, October 2020, <https://www.rfc-editor.org/info/rfc8919>.

8.2. Informative References

[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>.

Authors' Addresses

Acee Lindem
Cisco Systems
301 Midenhall Way
Cary, NC 27513

EMail: acee@cisco.com

Stephane Litkowski
Cisco Systems

EMail: slitkows.ietf@gmail.com

Yingzhen Qu
Futurewei
2330 Central Expressway
Santa Clara, CA 95050
USA

EMail: yingzhen.qu@futurewei.com