IS-IS Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 7, 2017

S. Litkowski
Orange
Y. Qu
Cisco
P. Sarkar
Individual
I. Chen
Kuatro Technologies
J. Tantsura
Individual

November 03, 2016

YANG Data Model for IS-IS Segment Routing
draft-ietf-isis-sr-yang-00

Abstract

This document defines a YANG data model that can be used to configure and manage IS-IS Segment Routing ([I-D.ietf-isis-segment-routing-extensions].

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

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 http://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 May 7, 2017.

Copyright Notice

Copyright (c) 2016 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 (http://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
2. IS-IS Segment Routing 3
3. IS-IS Segment Routing configuration 5
3.1. Segment Routing activation 5
3.2. Advertising mapping server policy 5
3.3. IP Fast reroute 5
4. IS-IS Segment Routing YANG Module 6
5. Security Considerations 20
6. Contributors 20
7. Acknowledgements 20
8. IANA Considerations 21
9. Change log for ietf-isis-sr YANG module 21
9.1. From isis document version -12 to isis-sr document version -00 21
9.2. From isis document version -12 to version -13 21
9.3. From isis document version -09 to version -11 21
9.4. From isis document version -08 to version -09 21
9.5. From isis document version -07 to version -08 21
10. Normative References 21
Authors' Addresses 22

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 model that can be used to configure and manage IS-IS Segment Routing and it is an augmentation to the IS-IS YANG data model.

2. 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 receive?     boolean
   +--rw protocol-srgb {sr: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 advertise-adj-group-sid* [group-id]
         |     ...
         +--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/isi
s:remote-lfa:
   +--rw use-segment-routing-path?   boolean {remote-lfa-sr}?
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
```

```
otocol/isis:isis:
   +--ro segment-routing
   |  +--ro enabled?    boolean
   |  +--ro bindings
   |     +--ro advertise
   |     |     ...
```

```
   |     +--ro receive?     boolean
   +--ro protocol-srgb {sr:protocol-srgb}?
      +--ro srgb* [lower-bound upper-bound]
         +--ro lower-bound    uint32
         +--ro upper-bound    uint32
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:interfaces/isis:interface:
   +--ro segment-routing
      +--ro adjacency-sid
         +--ro advertise-adj-group-sid* [group-id]
         |     ...
         +--ro advertise-protection?      enumeration
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:interfaces/isis:interface/isis:adjacencies/isis:ad
jacency:
   +--ro adjacency-sid* [value]
      +--ro af?                     identityref
      +--ro value                   uint32
      +--ro weight?                 uint8
      +--ro protection-requested?   boolean
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:database/isis:level-db/isis:lsp/isis:extended-is-n
eighbor/isis:neighbor:
   +--ro sid-list* [value]
      +--ro flags?         bits
      +--ro weight?        uint8
      +--ro neighbor-id?   isis:system-id
      +--ro value          uint32
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:database/isis:level-db/isis:lsp/isis:mt-is-neighbo
r/isis:neighbor:
   +--ro sid-list* [value]
      +--ro flags?         bits
      +--ro weight?        uint8
      +--ro neighbor-id?   isis:system-id
      +--ro value          uint32
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:database/isis:level-db/isis:lsp/isis:extended-ipv4
-reachability/isis:prefixes:
   +--ro sid-list* [value]
      +--ro flags?       bits
      +--ro algorithm?   uint8
      +--ro value        uint32
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:database/isis:level-db/isis:lsp/isis:mt-extended-i
pv4-reachability/isis:prefixes:
   +--ro sid-list* [value]
      +--ro flags?       bits
```

```
      +--ro algorithm?   uint8
      +--ro value        uint32
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:database/isis:level-db/isis:lsp/isis:ipv6-reachabi
lity/isis:prefixes:
   +--ro sid-list* [value]
      +--ro flags?       bits
      +--ro algorithm?   uint8
      +--ro value        uint32
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:database/isis:level-db/isis:lsp/isis:mt-ipv6-reach
ability/isis:prefixes:
   +--ro sid-list* [value]
      +--ro flags?       bits
      +--ro algorithm?   uint8
```

```
      +--ro value        uint32
augment /rt:routing-state/rt:control-plane-protocols/rt:control-plane-pr
otocol/isis:isis/isis:database/isis:level-db/isis:lsp:
   +--ro segment-routing-bindings* [fec range]
      +--ro fec        string
      +--ro range      uint16
      +--ro flags?     bits
      +--ro weight?    uint8
      +--ro binding
         +--ro prefix-sid
         |     ...
         +--ro ero-metric?                          uint32
         +--ro ero
         |     ...
         +--ro backup-ero
         |     ...
         +--ro unnumbered-interface-id-ero
         |     ...
         +--ro backup-unnumbered-interface-id-ero
               ...
```

3. IS-IS Segment Routing configuration

3.1. 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.

3.2. 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.

3.3. 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.

4. IS-IS Segment Routing YANG Module

```
<CODE BEGINS> file "ietf-isis-sr@2016-11-03.yang"

module ietf-isis-sr {
  namespace "urn:ietf:params:xml:ns:"
          + "yang:ietf-isis-sr";
  prefix isis-sr;


  import ietf-routing {
    prefix "rt";
  }

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


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

  import ietf-isis {
    prefix "isis";
  }

  organization
   "IETF ISIS Working Group";

  contact
    "WG List:  &lt;mailto:spring@ietf.org&gt;

    Editor:    Stephane Litkowski
          &lt;mailto:stephane.litkowski@orange.com&gt;

        Acee Lindem
          &lt;mailto:acee@cisco.com&gt;
        Yingzhen Qu
          &lt;mailto:yiqu@cisco.com&gt;
        Pushpasis Sarkar
          &lt;mailto:pushpasis.ietf@gmail.com&gt;
        Ing-Wher Chen
          &lt;mailto:ichen@kuatrotech.com&gt;
        Jeff Tantsura
          &lt;mailto:jefftant.ietf@gmail.com&gt;

    ";
```

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

     revision 2016-11-03 {
       description
           "Initial revision.";
       reference "RFC XXXX";
     }


     /* Identities */


     /* Features */

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

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

     /* Groupings */

   grouping adjacency-state {
       description
        "This group will extend adjacency state.";
       list adjacency-sid {
         key value;
         leaf af {
           type identityref {
             base rt: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;

         leaf flags {
           type bits {
             bit readvertisment {
               position 7;
               description
                "If set, then the prefix to
               which this Prefix-SID is attached,
               has been propagated by the
               router either from another level
               or from redistribution.";
             }

             bit php {
               position 5;
               description
               "If set, then the penultimate hop MUST NOT
               pop the Prefix-SID before delivering the packet
               to the node
               that advertised the Prefix-SID.";
             }
             bit explicit-null {
               position 4;
               description
```

```
                "If set, any upstream neighbor of
                the Prefix-SID originator MUST replace
                the Prefix-SID with a
                Prefix-SID having an
                Explicit-NULL value (0 for IPv4 and 2 for
                IPv6) before forwarding the packet.";
              }
              bit  value {
                position 3;
                description
                "If set, then the Prefix-SID carries a
                value (instead of an index).
                By default the flag is UNSET.";


              }
              bit  local {
                position 2;
                description
                "If set, then the value/index carried by
                the Prefix-SID has local significance.
                By default the flag is UNSET.";
              }
            }
            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;

         leaf flags {
           type bits {
             bit address-family {
               position 7;
               description
                "If unset, then the Adj-SID refers
               to an adjacency with outgoing IPv4 encapsulation.
               If set then the Adj-SID refers to an adjacency
               with outgoing IPv6 encapsulation.";
             }
             bit backup {
               position 6;
               description
                "If set, the Adj-SID refers to an
               adjacency being protected
               (e.g.: using IPFRR or MPLS-FRR)";
             }
             bit  value {
               position 5;
               description
               "If set, then the SID carries a
               value (instead of an index).
               By default the flag is SET.";


             }
             bit  local {
               position 4;
               description
               "If set, then the value/index carried by
               the SID has local significance.
               By default the flag is SET.";
             }
             bit  set {
               position 3;
               description
               "When set, the S-Flag indicates that the
               Adj-SID refers to a set of adjacencies";
             }
           }

           description
            "Describes flags associated with the
            segment ID.";
         }
```

```
        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.";
        }

        leaf flags {
          type bits {
            bit address-family {
              position 7;
              description
              "If unset, then the Prefix FEC
              carries an IPv4 Prefix.
```

```
              If set then the Prefix FEC carries an
              IPv6 Prefix.";
            }
            bit mirror {
              position 6;
              description
              "Set if the advertised SID/path
              corresponds to a mirrored context.
              ";
            }
            bit flooding {
              position 5;
              description
               "If the S bit is set(1),
               the IS-IS Router CAPABILITY TLV
              MUST be flooded across the entire routing domain.
              If the S bit is
              not set(0), the TLV MUST NOT be leaked between levels.
              This bit MUST NOT be altered during the TLV leaking.";
            }
            bit down {
              position 4;
              description
              "When the IS-IS Router CAPABILITY TLV is
              leaked from level-2 to level-1, the D bit
              MUST be set.  Otherwise, this bit MUST
              be clear.  IS-IS Router capability TLVs
              with the D bit set MUST NOT
              be leaked from level-1 to level-2.
              This is to prevent TLV looping.
              ";
            }
            bit attached {
              position 3;
              description
              "The originator of the SID/Label Binding
              TLV MAY set the A bit in order to signal
              that the prefixes and
              SIDs advertised in the SID/Label Binding
              TLV are directly
              connected to their originators.
              ";
            }

          }
          description
             "Flags of the binding.";
        }
```

```
          leaf weight {
            type uint8;
            description
             "Weight of the path for loadbalancing purpose.";
          }

          container binding {
            container prefix-sid {
              uses prefix-segment-id;
              description
               "Binding prefix SID to the range.";
            }
            leaf ero-metric {
              type uint32;
              description
               "Cost of ERO path.";
            }
            container ero {
              leaf address-family {
                type identityref {
                  base rt:address-family;
                }
                description
                 "Address-family.";
              }

              leaf loose {
                type boolean;
                description
                 "Set to true,
                 if hop is a loose hop.";
              }
              leaf address {
                type string;
                description
                "IP address of a node on the
                path.";
              }

              description
               "Binding ERO path to the range.";
            }
            container backup-ero {
              leaf address-family {
                type identityref {
                  base rt:address-family;
                }
                description
```

```
                "Address-family.";
              }

              leaf loose {
                type boolean;
                description
                 "Set to true,
                 if hop is a loose hop.";
              }
              leaf address {
                type string;
                description
                "IP address of a node on the
                path.";
              }

              description
               "Binding backup ERO path to the range.";
            }
            container unnumbered-interface-id-ero {
              leaf router-id {
                type string;
                description
                 "Router ID of the node owning the interface.";
              }
              leaf interface-id {
                type uint32;
                description
                "Interface ID on which the path is built.";
              }
              description
               "Binding a path over unnumbered interface.";
            }
            container backup-unnumbered-interface-id-ero {
              leaf router-id {
                type string;
                description
                 "Router ID of the node owning the interface.";
              }
              leaf interface-id {
                type uint32;
                description
                "Interface ID on which the path is built.";
              }
              description
               "Binding a backup path over unnumbered interface.";
            }
            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:controlplane-cfg;
      container protocol-srgb {
           if-feature sr:protocol-srgb;
           uses sr-cmn:srgb-cfg;
           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:igp-interface-cfg;
     }

      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-state/" +
           "rt:control-plane-protocols/rt:control-plane-protocol"+
           "/isis:isis" {
       when "/rt:routing-state/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:controlplane-cfg;
       container protocol-srgb {
            if-feature sr:protocol-srgb;
            uses sr-cmn:srgb-cfg;
            description
              "Per-protocol SRGB.";
          }

     }

      augment "/rt:routing-state/" +
           "rt:control-plane-protocols/rt:control-plane-protocol"+
           "/isis:isis/isis:interfaces/isis:interface" {
       when "/rt:routing-state/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:igp-interface-cfg;
     }

     augment "/rt:routing-state/" +
           "rt:control-plane-protocols/rt:control-plane-protocol"+
           "/isis:isis/isis:interfaces/isis:interface" +
           "/isis:adjacencies/isis:adjacency" {
       when "/rt:routing-state/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-state/" +
          "rt:control-plane-protocols/rt:control-plane-protocol"+
          "/isis:isis/isis:database/isis:level-db/isis:lsp"+
          "/isis:extended-is-neighbor/isis:neighbor" {
       when "/rt:routing-state/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-state/" +
          "rt:control-plane-protocols/rt:control-plane-protocol"+
          "/isis:isis/isis:database/isis:level-db/isis:lsp"+
          "/isis:mt-is-neighbor/isis:neighbor" {
              when "/rt:routing-state/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-state/" +
          "rt:control-plane-protocols/rt:control-plane-protocol"+
          "/isis:isis/isis:database/isis:level-db/isis:lsp"+
          "/isis:extended-ipv4-reachability/isis:prefixes" {
              when "/rt:routing-state/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-state/" +
           "rt:control-plane-protocols/rt:control-plane-protocol"+
           "/isis:isis/isis:database/isis:level-db/isis:lsp"+
           "/isis:mt-extended-ipv4-reachability/isis:prefixes" {
               when "/rt:routing-state/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-state/" +
           "rt:control-plane-protocols/rt:control-plane-protocol"+
           "/isis:isis/isis:database/isis:level-db/isis:lsp"+
           "/isis:ipv6-reachability/isis:prefixes" {
               when "/rt:routing-state/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-state/" +
           "rt:control-plane-protocols/rt:control-plane-protocol"+
           "/isis:isis/isis:database/isis:level-db/isis:lsp"+
           "/isis:mt-ipv6-reachability/isis:prefixes" {
               when "/rt:routing-state/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-state/" +
           "rt:control-plane-protocols/rt:control-plane-protocol"+
           "/isis:isis/isis:database/isis:level-db/isis:lsp" {
           when "/rt:routing-state/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>

5. Security Considerations

Configuration and state data defined in this document are designed to be accessed via the NETCONF protocol [RFC6241].

As IS-IS is an IGP protocol (critical piece of the network), ensuring stability and security of the protocol is mandatory for the network service.

Authors recommends to implement NETCONF access control model ([RFC6536]) to restrict access to all or part of the configuration to specific users.

6. Contributors

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

7. Acknowledgements

TBD.

8. 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: IS-IS WG
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

9. Change log for ietf-isis-sr YANG module

9.1. From isis document version -12 to isis-sr document version -00

- Separate document for IS-IS SR extensions.

9.2. From isis document version -12 to version -13

- Align with new segment routing common module.

9.3. From isis document version -09 to version -11

- Fixed XPATH in 'when' expressions.

9.4. From isis document version -08 to version -09

- Align to draft-ietf-netmod-routing-cfg-23.

9.5. From isis document version -07 to version -08

- Align to draft-ietf-netmod-routing-cfg-21.

10. Normative References

[I-D.ietf-isis-segment-routing-extensions]
Previdi, S., Filsfils, C., Bashandy, A., Gredler, H., Litkowski, S., Decraene, B., and j. jefftant@gmail.com, "IS-IS Extensions for Segment Routing", draft-ietf-isis-segment-routing-extensions-09 (work in progress), October 2016.

[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-13 (work in progress), October 2016.

[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-05 (work in progress), October 2016.

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

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <http://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, <http://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, <http://www.rfc-editor.org/info/rfc6241>.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, DOI 10.17487/RFC6536, March 2012, <http://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, <http://www.rfc-editor.org/info/rfc7950>.

Authors' Addresses

Stephane Litkowski
Orange

Email: stephane.litkowski@orange.com

Yinghzen Qu
Cisco

Email: yiqu@cisco.com

Pushpasis Sarkar
Individual

Email: pushpasis.ietf@gmail.com

Ing-Wher Chen
Kuatro Technologies

Email: ichen@kuatrotech.com

Jeff Tantsura
Individual

Email: jefftant.ietf@gmail.com