IS-IS Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 3, 2019

S. Litkowski
Orange
D. Yeung
Arrcus, Inc
A. Lindem
Cisco Systems
J. Zhang
Juniper Networks
L. Lhotka
CZ.NIC
July 2, 2018

YANG Data Model for IS-IS protocol
draft-ietf-isis-yang-isis-cfg-21

Abstract

This document defines a YANG data model that can be used to configure and manage IS-IS protocol on network elements.

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 January 3, 2019.

Copyright Notice

Copyright (c) 2018 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. Introduction 3
  1.1. Tree diagram 3
2. Design of the Data Model 3
  2.1. IS-IS Configuration 9
  2.2. Multitopology Parameters 10
  2.3. Per-Level Parameters 10
  2.4. Per-Interface Parameters 12
  2.5. Authentication Parameters 23
  2.6. IGP/LDP synchronization 23
  2.7. ISO parameters 24
  2.8. IP FRR 24
  2.9. Operational States 24
3. RPC Operations 25
4. Notifications 25
5. Interaction with Other YANG Modules 26
6. IS-IS YANG Module 27
7. Security Considerations 94
8. Contributors 95
9. Acknowledgements 95
10. IANA Considerations 95
11. Change log for ietf-isis YANG module 96
  11.1. From version -19 to version -20 96
  11.2. From version -18 to version -19 96
  11.3. From version -17 to version -18 96
  11.4. From version -16 to version -17 96
  11.5. From version -15 to version -16 96
  11.6. From version -14 to version -15 97
  11.7. From version -13 to version -14 97
  11.8. From version -12 to version -13 97
  11.9. From version -09 to version -12 97
  11.10. From version -08 to version -09 98

11.11. From version -07 to version -08 98
11.12. From version -05 to version -07 98
11.13. From version -03 to version -05 98
11.14. From version -02 to version -03 99
11.15. From version -01 to version -02 99
11.16. From version -00 to version -01 99
12. Normative References 100
Appendix A. Example of IS-IS configuration in XML 101
Authors' Addresses 103

1. Introduction

This document defines a YANG data model for IS-IS routing protocol.

The data model covers configuration of an IS-IS routing protocol instance as well as operational states.

1.1. Tree diagram

A simplified graphical representation of the data model is presented in Section 2.

The meaning of the symbols in these diagrams is as follows:

- o Brackets "[" and "]" enclose list keys.
- o Curly braces "{" and "}" contain names of optional features that make the corresponding node conditional.
- o Abbreviations before data node names: "rw" means configuration (read-write), and "ro" state data (read-only).
- o Symbols after data node names: "?" means an optional node and "*" denotes a "list" or "leaf-list".
- o Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").
- o Ellipsis ("...") stands for contents of subtrees that are not shown.

2. Design of the Data Model

The IS-IS YANG module augments the "control-plane-protocol" list in ietf-routing module with specific IS-IS parameters.

The module is designed as per NMDA (Network Management Datastore Architecture).

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

```
module: ietf-isis
  augment /rt:routing/rt:ribs/rt:rib/rt:routes/rt:route:
    +--ro metric?       uint32
    +--ro tag*          uint64
    +--ro route-type?   enumeration
  augment /if:interfaces/if:interface:
    +--rw clns-mtu?   uint16
  augment /rt:routing/rt:control-plane-protocols/
    |      rt:control-plane-protocol:
    +--rw isis
       +--rw enable?                   boolean {admin-control}?
       +--rw level-type?               level
       +--rw system-id?                system-id
       +--rw maximum-area-addresses?
       |      uint8 {maximum-area-addresses}?
       +--rw area-address*             area-address
       +--rw mpls
       |  +--rw ipv4-router-id?  inet:ipv4-address {ipv4-router-id}?
       |  +--rw ipv6-router-id?  inet:ipv6-address {ipv6-router-id}?
       |  +--rw ldp
       |        ...
       +--rw auto-cost {auto-cost}?
       |  +--rw reference-bandwidth?   uint32
       |  +--rw enable?                boolean
       +--rw lsp-mtu?                  uint16
       +--rw lsp-lifetime?             uint16
       +--rw lsp-refresh?
       |      rt-types:timer-value-seconds16 {lsp-refresh}?
       +--rw graceful-restart {graceful-restart}?
       |  +--rw enable?             boolean
       |  +--rw restart-interval?   rt-types:timer-value-seconds16
       |  +--rw helper-enable?      boolean
       +--rw nsr {nsr}?
       |  +--rw enable?   boolean
       +--rw node-tags {node-tag}?
       |  +--rw node-tag* [tag]
       |        ...
       +--rw authentication
       |  +--rw (authentication-type)?
       |  |     ...
       |  +--rw level-1
       |  |     ...
       |  +--rw level-2
       |        ...
```

```
          +--rw metric-type
          |  +--rw value?     enumeration
          |  +--rw level-1
          |  |     ...
          |  +--rw level-2
          |        ...
          +--rw default-metric
          |  +--rw value?     wide-metric
          |  +--rw level-1
          |  |     ...
          |  +--rw level-2
          |        ...
          +--rw afs {nlpid-control}?
          |  +--rw af* [af]
          |        ...
          +--rw preference
          |  +--rw (granularity)?
          |        ...
          +--rw overload
          |  +--rw status?   boolean
          +--rw overload-max-metric {overload-max-metric}?
          |  +--rw timeout?   rt-types:timer-value-seconds16
          +--rw fast-reroute {fast-reroute}?
          |  +--rw lfa {lfa}?
          |  +--ro protected-routes
          |  |     ...
          |  +--ro nonprotected-routes
          |  |     ...
          |  +--ro protection-statistics* [frr-protection-method]
          |        ...
          +--rw spf-control
          |  +--rw ietf-spf-delay {ietf-spf-delay}?
          |        ...
          +--rw topologies {multi-topology}?
          |  +--rw topology* [name]
          |        ...
          +--rw interfaces
          |  +--rw interface* [name]
          |        ...
          +--ro spf-log
          |  +--ro event* [id]
          |        ...
          +--ro lsp-log
          |  +--ro event* [id]
          |        ...
          +--ro hostnames
          |  +--ro hostname* [system-id]
          |        ...
```

```
          +--ro database
          |  +--ro level-db* [level]
          |        ...
          +--ro local-rib
          |  +--ro route* [prefix]
          |        ...
          +--ro system-counters
             +--ro level* [level]
                   ...

     rpcs:
       +---x clear-adjacency
       |  +---w input
       |     +---w routing-protocol-instance-name    instance-state-ref
       |     +---w level?                            level
       |     +---w interface?                        string
       +---x clear-database
          +---w input
             +---w routing-protocol-instance-name    instance-state-ref
             +---w level?                            level

     notifications:
       +---n database-overload
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro overload?                enumeration
       +---n lsp-too-large
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro pdu-size?                uint32
       |  +--ro lsp-id?                  lsp-id
       +---n if-state-change
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro state?                   if-state-type
       +---n corrupted-lsp-detected
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
```

```
       |  +--ro lsp-id?                  lsp-id
       +---n attempt-to-exceed-max-sequence
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro lsp-id?                  lsp-id
       +---n id-len-mismatch
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro pdu-field-len?           uint8
       |  +--ro raw-pdu?                 binary
       +---n max-area-addresses-mismatch
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro max-area-addresses?      uint8
       |  +--ro raw-pdu?                 binary
       +---n own-lsp-purge
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro lsp-id?                  lsp-id
       +---n sequence-number-skipped
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro lsp-id?                  lsp-id
       +---n authentication-type-failure
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro raw-pdu?                 binary
```

```
       +---n authentication-failure
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro raw-pdu?                 binary
       +---n version-skew
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro protocol-version?        uint8
       |  +--ro raw-pdu?                 binary
       +---n area-mismatch
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro raw-pdu?                 binary
       +---n rejected-adjacency
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro raw-pdu?                 binary
       |  +--ro reason?                  string
       +---n protocols-supported-mismatch
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
       |  +--ro interface-level?         level
       |  +--ro extended-circuit-id?     extended-circuit-id
       |  +--ro raw-pdu?                 binary
       |  +--ro protocols*               uint8
       +---n lsp-error-detected
       |  +--ro routing-instance?        string
       |  +--ro routing-protocol-name?   string
       |  +--ro isis-level?              level
       |  +--ro interface-name?          string
```

```
      |  +--ro interface-level?         level
      |  +--ro extended-circuit-id?     extended-circuit-id
      |  +--ro lsp-id?                  lsp-id
      |  +--ro raw-pdu?                 binary
      |  +--ro error-offset?            uint32
      |  +--ro tlv-type?                uint8
      +---n adjacency-state-change
      |  +--ro routing-instance?        string
      |  +--ro routing-protocol-name?   string
      |  +--ro isis-level?              level
      |  +--ro interface-name?          string
      |  +--ro interface-level?         level
      |  +--ro extended-circuit-id?     extended-circuit-id
      |  +--ro neighbor?                string
      |  +--ro neighbor-system-id?      system-id
      |  +--ro state?                   adj-state-type
      |  +--ro reason?                  string
      +---n lsp-received
      |  +--ro routing-instance?        string
      |  +--ro routing-protocol-name?   string
      |  +--ro isis-level?              level
      |  +--ro interface-name?          string
      |  +--ro interface-level?         level
      |  +--ro extended-circuit-id?     extended-circuit-id
      |  +--ro lsp-id?                  lsp-id
      |  +--ro sequence?                uint32
      |  +--ro received-timestamp?      yang:timestamp
      |  +--ro neighbor-system-id?      system-id
      +---n lsp-generation
         +--ro routing-instance?        string
         +--ro routing-protocol-name?   string
         +--ro isis-level?              level
         +--ro lsp-id?                  lsp-id
         +--ro sequence?                uint32
         +--ro send-timestamp?          yang:timestamp
```

2.1. IS-IS Configuration

The IS-IS configuration container is divided in:

o Global parameters.

o Per interface configuration (see Section 2.4).

Additional modules may be created this to support any additional parameters. These additional modules should augment the ietf-isis module.

The model implements features, thus some of the configuration statement becomes optional. As an example, the ability to control the administrative state of a particular IS-IS instance is optional. By advertising the feature "admin-control", a device communicates to the client that it supports the ability to shutdown a particular IS-IS instance.

The global configuration contains usual IS-IS parameters such as lsp-mtu, lsp-lifetime, lsp-refresh, default-metric...

2.2. Multitopology Parameters

The model supports multitopology (MT) IS-IS as defined in [RFC5120].

The "topologies" container is used to enable support of MT extensions.

The "name" used in the topology list should refer to an existing RIB of the device.

Some specific parameters could be defined on a per topology basis both at global level and at interface level: for example, an interface metric can be defined per topology.

Multiple address families (like IPv4 or IPv6) can also be activated within the default topology. This can be achieved using the "afs" container (requiring "nlpid-control" feature to be advertised).

2.3. Per-Level Parameters

Some parameters allow a per level configuration. In this case, the parameter is modeled as a container with three configuration locations:

- o a top level container: corresponds to level-1-2, so the configuration applies to both levels.

- o a level-1 container: corresponds to level-1 specific parameters.

- o a level-2 container: corresponds to level-2 specific parameters.

```
+--rw priority
|  +--rw value?     uint8
|  +--rw level-1
|  |  +--rw value?   uint8
|  +--rw level-2
|     +--rw value?   uint8
```

Example:

```
<priority>
    <value>250</value>
                <level-1>
                        <value>100</value>
                </level-1>
                <level-2>
                        <value>200</value>
                </level-2>
</priority>
```

An implementation SHOULD prefer a level specific parameter over a level-all parameter. As example, if the priority is 100 for the level-1, 200 for the level-2 and 250 for the top level configuration, the implementation should use 100 for the level-1 and 200 for the level-2.

Some parameters like "overload bit" and "route preference" are not modeled to support a per level configuration. If an implementation supports per level configuration for such parameter, this implementation SHOULD augment the current model by adding both level-1 and level-2 containers and SHOULD reuse existing configuration groupings.

Example of augmentation:

```
augment "/rt:routing/" +
        "rt:control-plane-protocols/rt:control-plane-protocol"+
        "/isis:isis/isis:overload" {
        when "rt:type = 'isis:isis'" {
      description
      "This augment IS-IS routing protocol when used";
    }
    description
     "This augments IS-IS overload configuration
      with per level configuration.";

        container level-1 {
                uses isis:overload-global-cfg;
                description
                 "Level 1 configuration.";
        }
        container level-2 {
                uses isis:overload-global-cfg;
                description
                 "Level 2 configuration.";
        }

}
```

If an implementation does not support per level configuration for a parameter modeled with per level configuration, the implementation SHOULD advertise a deviation to announce the non support of the level-1 and level-2 containers.

Finally, if an implementation supports per level configuration but does not support the level-1-2 configuration, it SHOULD also advertise a deviation.

2.4. Per-Interface Parameters

The per-interface section of the IS-IS instance describes the interface specific parameters.

The interface is modeled as a reference to an existing interface defined in the "ietf-interfaces" YANG model.

Each interface has some interface-specific parameters that may have a different per level value as described in previous section. An interface-specific parameter always override an IS-IS global parameter.

Some parameters like hello-padding are defined as containers to allow easy extension by vendor specific modules.

```
       +--rw interfaces
       |  +--rw interface* [name]
       |     +--rw name                       if:interface-ref
       |     +--rw level-type?                level
       |     +--rw lsp-pacing-interval?
       |     |       rt-types:timer-value-milliseconds
       |     +--rw lsp-retransmit-interval?
       |     |       rt-types:timer-value-seconds16
       |     +--rw passive?                   boolean
       |     +--rw csnp-interval?
       |     |       rt-types:timer-value-seconds16
       |     +--rw hello-padding
       |     |  +--rw enable?   boolean
       |     +--rw mesh-group-enable?         mesh-group-state
       |     +--rw mesh-group?                uint8
       |     +--rw interface-type?            interface-type
       |     +--rw enable?                    boolean {admin-control}?
       |     +--rw tag*                       uint32 {prefix-tag}?
       |     +--rw tag64*                     uint64 {prefix-tag64}?
       |     +--rw node-flag?                 boolean {node-flag}?
       |     +--rw hello-authentication
       |     |  +--rw (authentication-type)?
       |     |  |  +--:(key-chain) {key-chain}?
       |     |  |  |  +--rw key-chain?        key-chain:key-chain-ref
       |     |  |  +--:(password)
       |     |  |     +--rw key?                string
       |     |  |     +--rw crypto-algorithm?   identityref
       |     |  +--rw level-1
       |     |  |  +--rw (authentication-type)?
       |     |  |     +--:(key-chain) {key-chain}?
       |     |  |     |  +--rw key-chain?      key-chain:key-chain-ref
       |     |  |     +--:(password)
       |     |  |        +--rw key?                string
       |     |  |        +--rw crypto-algorithm?   identityref
       |     |  +--rw level-2
       |     |     +--rw (authentication-type)?
       |     |        +--:(key-chain) {key-chain}?
       |     |        |  +--rw key-chain?      key-chain:key-chain-ref
       |     |        +--:(password)
       |     |           +--rw key?                string
       |     |           +--rw crypto-algorithm?   identityref
       |     +--rw hello-interval
       |     |  +--rw value?     rt-types:timer-value-seconds16
       |     |  +--rw level-1
       |     |  |  +--rw value?   rt-types:timer-value-seconds16
       |     |  +--rw level-2
       |     |     +--rw value?   rt-types:timer-value-seconds16
       |     +--rw hello-multiplier
```

```
        |     |  +--rw value?     uint16
        |     |  +--rw level-1
        |     |  |  +--rw value?   uint16
        |     |  +--rw level-2
        |     |     +--rw value?   uint16
        |     +--rw priority
        |     |  +--rw value?     uint8
        |     |  +--rw level-1
        |     |  |  +--rw value?   uint8
        |     |  +--rw level-2
        |     |     +--rw value?   uint8
        |     +--rw metric
        |     |  +--rw value?     wide-metric
        |     |  +--rw level-1
        |     |  |  +--rw value?   wide-metric
        |     |  +--rw level-2
        |     |     +--rw value?   wide-metric
        |     +--rw bfd {bfd}?
        |     |  +--rw enable?                          boolean
        |     |  +--rw local-multiplier?                multiplier
        |     |  +--rw (interval-config-type)?
        |     |     +--:(tx-rx-intervals)
        |     |     |  +--rw desired-min-tx-interval?    uint32
        |     |     |  +--rw required-min-rx-interval?   uint32
        |     |     +--:(single-interval) {single-minimum-interval}?
        |     |        +--rw min-interval?               uint32
        |     +--rw afs {nlpid-control}?
        |     |  +--rw af* [af]
        |     |     +--rw af    iana-rt-types:address-family
        |     +--rw mpls
        |     |  +--rw ldp
        |     |     +--rw igp-sync?   boolean {ldp-igp-sync}?
        |     +--rw fast-reroute {fast-reroute}?
        |     |  +--rw lfa {lfa}?
        |     |     +--rw candidate-disabled?   boolean
        |     |     +--rw enable?               boolean
        |     |     +--rw remote-lfa {remote-lfa}?
        |     |     |  +--rw enable?   boolean
        |     |     +--rw level-1
        |     |     |  +--rw candidate-disabled?   boolean
        |     |     |  +--rw enable?               boolean
        |     |     |  +--rw remote-lfa {remote-lfa}?
        |     |     |     +--rw enable?   boolean
        |     |     +--rw level-2
        |     |        +--rw candidate-disabled?   boolean
        |     |        +--rw enable?               boolean
        |     |        +--rw remote-lfa {remote-lfa}?
        |     |           +--rw enable?   boolean
```

```
        |     +--rw topologies {multi-topology}?
        |     |  +--rw topology* [name]
        |     |     +--rw name
        |     |     |      -> ../../../../../../../../rt:ribs/rib/name
        |     |     +--rw metric
        |     |        +--rw value?     wide-metric
        |     |        +--rw level-1
        |     |        |  +--rw value?   wide-metric
        |     |        +--rw level-2
        |     |           +--rw value?   wide-metric
        |     +--ro adjacencies
        |     |  +--ro adjacency* []
        |     |     +--ro neighbor-systype?              level
        |     |     +--ro neighbor-sysid?                system-id
        |     |     +--ro neighbor-extended-circuit-id?
        |     |     |      extended-circuit-id
        |     |     +--ro neighbor-snpa?                 snpa
        |     |     +--ro usage?                         level
        |     |     +--ro hold-timer?
        |     |     |       rt-types:timer-value-seconds16
        |     |     +--ro neighbor-priority?             uint8
        |     |     +--ro lastuptime?                    yang:timestamp
        |     |     +--ro state?                         adj-state-type
        |     +--ro event-counters
        |     |  +--ro adjacency-changes?            uint32
        |     |  +--ro adjacency-number?             uint32
        |     |  +--ro init-fails?                   uint32
        |     |  +--ro adjacency-rejects?            uint32
        |     |  +--ro id-len-mismatch?              uint32
        |     |  +--ro max-area-addresses-mismatch?  uint32
        |     |  +--ro authentication-type-fails?    uint32
        |     |  +--ro authentication-fails?         uint32
        |     |  +--ro lan-dis-changes?              uint32
        |     +--ro packet-counters
        |        +--ro level* [level]
        |           +--ro level      level-number
        |           +--ro iih
        |           |  +--ro in?    uint32
        |           |  +--ro out?   uint32
        |           +--ro ish
        |           |  +--ro in?    uint32
        |           |  +--ro out?   uint32
        |           +--ro esh
        |           |  +--ro in?    uint32
        |           |  +--ro out?   uint32
        |           +--ro lsp
        |           |  +--ro in?    uint32
        |           |  +--ro out?   uint32
```

```
       |           +--ro psnp
       |           |  +--ro in?    uint32
       |           |  +--ro out?   uint32
       |           +--ro csnp
       |           |  +--ro in?    uint32
       |           |  +--ro out?   uint32
       |           +--ro unknown
       |              +--ro in?    uint32
       |              +--ro out?   uint32
       +--ro spf-log
       |  +--ro event* [id]
       |     +--ro id                    uint32
       |     +--ro spf-type?             enumeration
       |     +--ro level?                level-number
       |     +--ro schedule-timestamp?   yang:timestamp
       |     +--ro start-timestamp?      yang:timestamp
       |     +--ro end-timestamp?        yang:timestamp
       |     +--ro trigger-lsp* [lsp]
       |        +--ro lsp         lsp-id
       |        +--ro sequence?   uint32
       +--ro lsp-log
       |  +--ro event* [id]
       |     +--ro id                    uint32
       |     +--ro level?                level-number
       |     +--ro lsp
       |     |  +--ro lsp?        lsp-id
       |     |  +--ro sequence?   uint32
       |     +--ro received-timestamp?   yang:timestamp
       |     +--ro change?               identityref
       +--ro hostnames
       |  +--ro hostname* [system-id]
       |     +--ro system-id    system-id
       |     +--ro hostname?    string
       +--ro database
       |  +--ro level-db* [level]
       |     +--ro level    level-number
       |     +--ro lsp* [lsp-id]
       |        +--ro decoded-completed?               boolean
       |        +--ro raw-data?                        yang:hex-string
       |        +--ro lsp-id                           lsp-id
       |        +--ro checksum?                        uint16
       |        +--ro remaining-lifetime?              uint16
       |        +--ro sequence?                        uint32
       |        +--ro attributes?                      bits
       |        +--ro ipv4-addresses*                  inet:ipv4-address
       |        +--ro ipv6-addresses*                  inet:ipv6-address
       |        +--ro ipv4-te-routerid?                inet:ipv4-address
       |        +--ro ipv6-te-routerid?                inet:ipv6-address
```

```
        |        +--ro protocol-supported*              uint8
        |        +--ro dynamic-hostname?                string
        |        +--ro authentication
        |        |  +--ro authentication-type?   string
        |        |  +--ro authentication-key?    string
        |        +--ro mt-entries
        |        |  +--ro topology* []
        |        |     +--ro MT-ID?        uint16
        |        |     +--ro attributes?   bits
        |        +--ro router-capabilities* []
        |        |  +--ro flags?       bits
        |        |  +--ro node-tags {node-tag}?
        |        |  |  +--ro node-tag* []
        |        |  |     +--ro tag?   uint32
        |        |  +--ro binary?      binary
        |        +--ro is-neighbor
        |        |  +--ro neighbor* []
        |        |     +--ro neighbor-id?      system-id
        |        |     +--ro i-e?              boolean
        |        |     +--ro default-metric?   std-metric
        |        |     +--ro delay-metric
        |        |     |  +--ro metric?      std-metric
        |        |     |  +--ro supported?   boolean
        |        |     +--ro expense-metric
        |        |     |  +--ro metric?      std-metric
        |        |     |  +--ro supported?   boolean
        |        |     +--ro error-metric
        |        |        +--ro metric?      std-metric
        |        |        +--ro supported?   boolean
        |        +--ro extended-is-neighbor
        |        |  +--ro neighbor* []
        |        |     +--ro neighbor-id?   system-id
        |        |     +--ro metric?        wide-metric
        |        +--ro ipv4-internal-reachability
        |        |  +--ro prefixes* []
        |        |     +--ro up-down?          boolean
        |        |     +--ro i-e?              boolean
        |        |     +--ro ip-prefix?        inet:ipv4-address
        |        |     +--ro prefix-len?       uint8
        |        |     +--ro default-metric?   std-metric
        |        |     +--ro delay-metric
        |        |     |  +--ro metric?      std-metric
        |        |     |  +--ro supported?   boolean
        |        |     +--ro expense-metric
        |        |     |  +--ro metric?      std-metric
        |        |     |  +--ro supported?   boolean
        |        |     +--ro error-metric
        |        |        +--ro metric?      std-metric
```

```
        |        |        +--ro supported?   boolean
        |        +--ro ipv4-external-reachability
        |        |  +--ro prefixes* []
        |        |     +--ro up-down?          boolean
        |        |     +--ro i-e?              boolean
        |        |     +--ro ip-prefix?        inet:ipv4-address
        |        |     +--ro prefix-len?       uint8
        |        |     +--ro default-metric?   std-metric
        |        |     +--ro delay-metric
        |        |     |  +--ro metric?      std-metric
        |        |     |  +--ro supported?   boolean
        |        |     +--ro expense-metric
        |        |     |  +--ro metric?      std-metric
        |        |     |  +--ro supported?   boolean
        |        |     +--ro error-metric
        |        |        +--ro metric?      std-metric
        |        |        +--ro supported?   boolean
        |        +--ro extended-ipv4-reachability
        |        |  +--ro prefixes* []
        |        |     +--ro up-down?                 boolean
        |        |     +--ro ip-prefix?               inet:ipv4-address
        |        |     +--ro prefix-len?              uint8
        |        |     +--ro metric?                  wide-metric
        |        |     +--ro tag*                     uint32
        |        |     +--ro tag64*                   uint64
        |        |     +--ro external-prefix-flag?    boolean
        |        |     +--ro readvertisement-flag?    boolean
        |        |     +--ro node-flag?               boolean
        |        |     +--ro ipv4-source-router-id?   inet:ipv4-address
        |        |     +--ro ipv6-source-router-id?   inet:ipv6-address
        |        +--ro mt-is-neighbor
        |        |  +--ro neighbor* []
        |        |     +--ro MT-ID?         uint16
        |        |     +--ro neighbor-id?   system-id
        |        |     +--ro metric?        wide-metric
        |        +--ro mt-extended-ipv4-reachability
        |        |  +--ro prefixes* []
        |        |     +--ro MT-ID?                   uint16
        |        |     +--ro up-down?                 boolean
        |        |     +--ro ip-prefix?               inet:ipv4-address
        |        |     +--ro prefix-len?              uint8
        |        |     +--ro metric?                  wide-metric
        |        |     +--ro tag*                     uint32
        |        |     +--ro tag64*                   uint64
        |        |     +--ro external-prefix-flag?    boolean
        |        |     +--ro readvertisement-flag?    boolean
        |        |     +--ro node-flag?               boolean
        |        |     +--ro ipv4-source-router-id?   inet:ipv4-address
```

```
        |        |     +--ro ipv6-source-router-id?   inet:ipv6-address
        |        +--ro mt-ipv6-reachability
        |        |  +--ro prefixes* []
        |        |     +--ro MT-ID?                   uint16
        |        |     +--ro up-down?                 boolean
        |        |     +--ro ip-prefix?               inet:ipv6-address
        |        |     +--ro prefix-len?              uint8
        |        |     +--ro metric?                  wide-metric
        |        |     +--ro tag*                     uint32
        |        |     +--ro tag64*                   uint64
        |        |     +--ro external-prefix-flag?    boolean
        |        |     +--ro readvertisement-flag?    boolean
        |        |     +--ro node-flag?               boolean
        |        |     +--ro ipv4-source-router-id?   inet:ipv4-address
        |        |     +--ro ipv6-source-router-id?   inet:ipv6-address
        |        +--ro ipv6-reachability
        |           +--ro prefixes* []
        |              +--ro up-down?                 boolean
        |              +--ro ip-prefix?               inet:ipv6-address
        |              +--ro prefix-len?              uint8
        |              +--ro metric?                  wide-metric
        |              +--ro tag*                     uint32
        |              +--ro tag64*                   uint64
        |              +--ro external-prefix-flag?    boolean
        |              +--ro readvertisement-flag?    boolean
        |              +--ro node-flag?               boolean
        |              +--ro ipv4-source-router-id?   inet:ipv4-address
        |              +--ro ipv6-source-router-id?   inet:ipv6-address
        +--ro local-rib
        |  +--ro route* [prefix]
        |     +--ro prefix       inet:ip-prefix
        |     +--ro next-hops
        |     |  +--ro next-hop* [next-hop]
        |     |     +--ro outgoing-interface?   if:interface-ref
        |     |     +--ro next-hop              inet:ip-address
        |     +--ro metric?      uint32
        |     +--ro level?       level-number
        |     +--ro route-tag?   uint32
        +--ro system-counters
           +--ro level* [level]
              +--ro level                            level-number
              +--ro corrupted-lsps?                  uint32
              +--ro authentication-type-fails?       uint32
              +--ro authentication-fails?            uint32
              +--ro database-overload?               uint32
              +--ro own-lsp-purge?                   uint32
              +--ro manual-address-drop-from-area?   uint32
              +--ro max-sequence?                    uint32
```

```
             +--ro sequence-number-skipped?         uint32
             +--ro id-len-mismatch?                 uint32
             +--ro partition-changes?               uint32
             +--ro lsp-errors?                      uint32
             +--ro spf-runs?                        uint32

   rpcs:
    +---x clear-adjacency
    |  +---w input
    |     +---w routing-protocol-instance-name    instance-state-ref
    |     +---w level?                            level
    |     +---w interface?                        string
    +---x clear-database
       +---w input
          +---w routing-protocol-instance-name    instance-state-ref
          +---w level?                            level

   notifications:
    +---n database-overload
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro overload?                enumeration
    +---n lsp-too-large
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro pdu-size?                uint32
    |  +--ro lsp-id?                  lsp-id
    +---n if-state-change
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro state?                   if-state-type
    +---n corrupted-lsp-detected
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro lsp-id?                  lsp-id
    +---n attempt-to-exceed-max-sequence
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
```

```
    |  +--ro isis-level?              level
    |  +--ro lsp-id?                  lsp-id
    +---n id-len-mismatch
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro pdu-field-len?           uint8
    |  +--ro raw-pdu?                 binary
    +---n max-area-addresses-mismatch
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro max-area-addresses?      uint8
    |  +--ro raw-pdu?                 binary
    +---n own-lsp-purge
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro lsp-id?                  lsp-id
    +---n sequence-number-skipped
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro lsp-id?                  lsp-id
    +---n authentication-type-failure
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro raw-pdu?                 binary
    +---n authentication-failure
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
```

```
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro raw-pdu?                 binary
    +---n version-skew
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro protocol-version?        uint8
    |  +--ro raw-pdu?                 binary
    +---n area-mismatch
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro raw-pdu?                 binary
    +---n rejected-adjacency
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro raw-pdu?                 binary
    |  +--ro reason?                  string
    +---n protocols-supported-mismatch
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro raw-pdu?                 binary
    |  +--ro protocols*               uint8
    +---n lsp-error-detected
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro lsp-id?                  lsp-id
    |  +--ro raw-pdu?                 binary
```

```
    |  +--ro error-offset?            uint32
    |  +--ro tlv-type?                uint8
    +---n adjacency-state-change
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro neighbor?                string
    |  +--ro neighbor-system-id?      system-id
    |  +--ro state?                   adj-state-type
    |  +--ro reason?                  string
    +---n lsp-received
    |  +--ro routing-instance?        string
    |  +--ro routing-protocol-name?   string
    |  +--ro isis-level?              level
    |  +--ro interface-name?          string
    |  +--ro interface-level?         level
    |  +--ro extended-circuit-id?     extended-circuit-id
    |  +--ro lsp-id?                  lsp-id
    |  +--ro sequence?                uint32
    |  +--ro received-timestamp?      yang:timestamp
    |  +--ro neighbor-system-id?      system-id
    +---n lsp-generation
       +--ro routing-instance?        string
       +--ro routing-protocol-name?   string
       +--ro isis-level?              level
       +--ro lsp-id?                  lsp-id
       +--ro sequence?                uint32
       +--ro send-timestamp?          yang:timestamp
```

2.5. Authentication Parameters

The module enables authentication configuration through the IETF key-chain module ([I-D.ietf-rtgwg-yang-key-chain]). The IS-IS module imports the "ietf-key-chain" module and reuses some groupings to allow global and per interface configuration of authentication. If a global authentication is configured, an implementation SHOULD authenticate PSNPs, CSNPs and LSPs with the authentication parameters supplied. The authentication of hello PDUs can be activated on a per interface basis.

2.6. IGP/LDP synchronization

[RFC5443] defines a mechanism where IGP needs to be synchronized with LDP. An "ldp-igp-sync" feature has been defined in the model to support this mechanism. The "mpls/ldp/igp-sync" leaf under

"interface" allows activation of the mechanism on a per interface basis. The "mpls/ldp/igp-sync" container in the global configuration is empty on purpose and is not required for the activation. The goal of this empty container is to allow easy augmentation with additional parameters like timers for example.

2.7. ISO parameters

As IS-IS protocol is based on ISO protocol suite, some ISO parameters may be required.

This module augments interface configuration model to support ISO configuration parameters.

The clns-mtu can be defined under the interface.

2.8. IP FRR

This YANG model supports LFA ([RFC5286]) and remote LFA ([RFC7490]) as IP FRR techniques. The "fast-reroute" container may be augmented by other models to support other IPFRR flavors (MRT, TILFA ...).

The current version of the model supports activation of LFA and remote LFA at interface only. The global "lfa" container is present but kept empty to allow augmentation with vendor specific properties like policies.

Remote LFA is considered as a child of LFA. Remote LFA cannot be enabled if LFA is not enabled.

The "candidate-disabled" allows to mark an interface to not be used as a backup.

2.9. Operational States

Operational states are provided in the module in various places:

- o system-counters: provides statistical informations about the global system.

- o interface : provides configuration state informations for each interface.

- o adjacencies: provides state informations about current IS-IS adjacencies.

- o spf-log: provides informations about SPF events on the node. This SHOULD be implemented as a wrapping buffer.

- o lsp-log: provides informations about LSP events on the node (reception of an LSP or modification of local LSP). This SHOULD be implemented as a wrapping buffer and an implementation MAY decide to log refresh LSPs or not.

- o local-rib: provides the IS-IS internal routing table view.

- o database: provides details on the current LSDB.

- o hostnames: provides informations about system-id to hostname mappings.

- o fast-reroute: provides informations about IP FRR.

3. RPC Operations

The "ietf-isis" module defines two RPC operations:

- o clear-isis-database: reset the content of a particular IS-IS database and restart database synchronization with the neighbors.

- o clear-isis-adjacency: restart a particular set of IS-IS adjacencies.

4. Notifications

The "ietf-isis" module introduces some notifications :

database-overload : raised when overload condition is changed.

lsp-too-large : raised when the system tries to propagate a too large PDU.

corrupted-lsp-detected : raised when the system find that an LSP that was stored in memory has become corrupted.

attempt-to-exceed-max-sequence : This notification is sent when the system wraps the 32-bit sequence counter of an LSP.

id-len-mismatch : This notification is sent when we receive a PDU with a different value for the System ID length.

max-area-addresses-mismatch : This notification is sent when we receive a PDU with a different value for the Maximum Area Addresses.

own-lsp-purge : This notification is sent when the system receives a PDU with its own system ID and zero age.

sequence-number-skipped : This notification is sent when the system receives a PDU with its own system ID and different contents. The system has to reissue the LSP with a higher sequence number.

authentication-type-failure : This notification is sent when the system receives a PDU with the wrong authentication type field.

authentication-failure : This notification is sent when the system receives a PDU with the wrong authentication information.

version-skew : This notification is sent when the system receives a PDU with a different protocol version number.

area-mismatch : This notification is sent when the system receives a Hello PDU from an IS that does not share any area address.

rejected-adjacency : This notification is sent when the system receives a Hello PDU from an IS but does not establish an adjacency for some reason.

protocols-supported-mismatch : This notification is sent when the system receives a non pseudonode LSP that has no matching protocol supported.

lsp-error-detected : This notification is sent when the system receives a LSP with a parse error.

adjacency-change : This notification is sent when an IS-IS adjacency moves to Up state or to Down state.

lsp-received : This notification is sent when a LSP is received.

lsp-generation : This notification is sent when a LSP is regenerated.

5. Interaction with Other YANG Modules

The "isis" configuration container augments the "/rt:routing/rt:control-plane-protocols/control-plane-protocol" container of the ietf-routing [I-D.ietf-netmod-routing-cfg] module by defining IS-IS specific parameters.

The "isis" module augments "/if:interfaces/if:interface" with ISO specific parameters.

The "isis" operational state container augments the "/rt:routing-state/rt:control-plane-protocols/control-plane-protocol" container of the ietf-routing module by defining IS-IS specific operational states.

Some IS-IS specific routes attributes are added to route objects of the ietf-routing module by augmenting "/rt:routing-state/rt:ribs/rt:rib/rt:routes/rt:route".

The modules defined in this document use some groupings from ietf-keychain [I-D.ietf-rtgwg-yang-key-chain].

6. IS-IS YANG Module

```
<CODE BEGINS> file "ietf-yang@2018-07-02.yang"
module ietf-isis {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-isis";

  prefix isis;

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

  import ietf-inet-types {
    prefix inet;
    reference "RFC 6021 - Common YANG Data Types";
  }

  import ietf-yang-types {
    prefix yang;
    reference "RFC 6021 - Common YANG Data Types";
  }

  import ietf-interfaces {
    prefix "if";
    reference "RFC 8343 - A YANG Data Model for Interface
               Management (NDMA Version)";
  }

  import ietf-key-chain {
    prefix "key-chain";
```

```
      reference "RFC 8177 - YANG Data Model for Key Chains";
    }

    import ietf-routing-types {
      prefix "rt-types";
      reference "RFC 8291 - Common YANG Data Types for the
                 Routing Area";
    }

    import iana-routing-types {
      prefix "iana-rt-types";
      reference "RFC 8291 - Common YANG Data Types for the
                 Routing Area";
    }

    import ietf-bfd-types {
      prefix "bfd-types";
      reference "RFC XXXX - YANG Data Model for Bidirectional
                 Forwarding Detection (BFD)";
    }

    organization
      "IETF IS-IS Working Group";

    contact
      "WG List:  &lt;mailto:isis-wg@ietf.org&gt;

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

          Derek Yeung
            &lt;mailto:derek@arrcus.com&gt;
          Acee Lindem
            &lt;mailto:acee@cisco.com&gt;
          Jeffrey Zhang
            &lt;mailto:zzhang@juniper.net&gt;
          Ladislav Lhotka
            &lt;mailto:llhotka@nic.cz&gt;
          Yi Yang
            &lt;mailto:yiya@cisco.com&gt;
          Dean Bogdanovic
            &lt;mailto:deanb@juniper.net&gt;
          Kiran Agrahara Sreenivasa
            &lt;mailto:kkoushik@brocade.com&gt;
          Yingzhen Qu
            &lt;mailto:yiqu@cisco.com&gt;
                 Jeff Tantsura
                   &lt;mailto:jefftant.ietf@gmail.com&gt;
```

```
       ";

     description
       "The YANG module defines a generic configuration model for
        IS-IS common across all of the vendor implementations.";

     revision 2018-07-02 {
       description
         "Initial revision.";
       reference "RFC XXXX";
     }

     /* Identities */

     identity isis {
       base rt:routing-protocol;
       description "Identity for the IS-IS routing protocol.";
     }

     identity isis-adjacency-change {
       description "Identity for the IS-IS routing protocol
                    adjacency state.";
     }

     identity clear-isis-database {
       description "Identity for the IS-IS routing protocol
                    database reset action.";
     }

     identity clear-isis-adjacency {
       description "Identity for the IS-IS routing protocol
                    adjacency reset action.";
     }

     identity lsp-log-reason {
       description "Base identity for an LSP change log reason.";
     }

     identity refresh {
       base lsp-log-reason;
       description
         "Identity used when the LSP log reason is
          a refresh LSP received.";
     }

     identity content-change {
       base lsp-log-reason;
       description
```

```
        "Identity used when the LSP log reason is
         a change in the content of the LSP.";
    }


    /* Feature definitions */

    feature ietf-spf-delay {
      description
        "Support of IETF SPF delay algorithm.";
      reference "RFC XXXX - SPF Back-off algorithm for link
                 state IGPs";
    }
    feature bfd {
      description
        "Support for BFD detection of IS-IS neighbor reachability.";
      reference "RFC 5880 - Bidirectional Forwarding Detection (BFD)
                 RFC 5881 - Bidirectional Forwarding Detection
                 (BFD) for IPv4 and IPv6 (Single Hop)";
    }
    feature key-chain {
      description
        "Support of keychain for authentication.";
      reference "RFC8177 - YANG Data Model for Key Chains";
    }
    feature node-flag {
      description
        "Support of node-flag advertisement
         as prefix attribute";
      reference "RFC7794 - IS-IS Prefix Attributes for
                 Extended IP and IPv6 Reachability";
    }
    feature node-tag {
      description
        "Support of node tag.";
      reference "RFC7917 - Advertising Node Administrative Tags
                 in IS-IS";
    }
    feature ldp-igp-sync {
      description
        "Support of synchronization between IS-IS and LDP.";
      reference "RFC5443 - LDP IGP Synchronization";
    }
    feature fast-reroute {
      description
        "Support of IP Fast Reroute.";
    }
    feature nsr {
```

```
      description
        "Support of Non-Stop Routing.";
    }
    feature lfa {
      description
        "Support of Loop-Free Alternates.";
      reference "RFC5286 - Basic Specification of IP Fast-Reroute:
                 Loop-free Alternates";
    }
    feature remote-lfa {
      description
        "Support of remote Loop Free Alternates.";
      reference "RFC7490 - Remote Loop-Free Alternate Fast Reroute";
    }

    feature overload-max-metric {
      description
        "Support of overload by setting
         all links to max metric.";
    }
    feature prefix-tag {
      description
        "Support for 32-bit prefix tags";
      reference "RFC5130 - A Policy Control Mechanism in
                 IS-IS Using Administrative Tags";
    }
    feature prefix-tag64 {
      description
        "Support for 64-bit prefix tags";
      reference "RFC5130 - A Policy Control Mechanism in
                 IS-IS Using Administrative Tags";
    }
    feature auto-cost {
      description
        "Use an automated assignment of metrics.";
    }
    feature ipv4-router-id {
      description
        "Support of IPv4 router ID configuration under IS-IS.";
    }

    feature ipv6-router-id {
      description
        "Support of IPv6 router ID configuration under IS-IS.";
    }

    feature multi-topology {
      description
```

```
        "Multitopology routing support.";
      reference "RFC5120 - M-IS-IS: Multi Topology Routing in IS-IS";
    }
    feature nlpid-control {
      description
        "This feature controls the advertisement
         of support NLPID within IS-IS configuration.";
    }
    feature graceful-restart {
      description
        "Graceful restart support.";
      reference "RFC5306 - Restart Signaling in IS-IS";
    }

    feature lsp-refresh {
      description
        "Configuration of LSP refresh interval.";
    }

    feature maximum-area-addresses {
      description
        "Support of maximum-area-addresses config.";
    }

    feature admin-control {
      description
        "Control administrative state of IS-IS.";
    }

    /* Type definitions */


    typedef instance-state-ref {
      type leafref {
        path "/rt:routing-state/"
          +"rt:control-plane-protocols/rt:control-plane-protocol/"
          +"rt:name";
      }
      description
        "This type is used for leaves that reference state data for
         an IS-IS protocol instance.";
    }

    typedef circuit-id {
      type uint8;
      description
        "This type defines the circuit ID
         associated with an interface.";
```

```
     }

     typedef extended-circuit-id {
       type uint32;
       description
         "This type defines the extended circuit ID
          associated with an interface.";
     }

     typedef interface-type {
       type enumeration {
         enum broadcast {
           description
             "Broadcast interface type.";
         }
         enum point-to-point {
           description
             "Point-to-point interface type.";
         }
       }
       description
         "This type defines the type of adjacency
          to be established on the interface.
          The interface-type determines the type
          of hello message that is used.";

     }

     typedef level {
       type enumeration {
         enum "level-1" {
           description
             "This enum indicates L1-only capability.";
         }
         enum "level-2" {
           description
             "This enum indicates L2-only capability.";
         }
         enum "level-all" {
           description
             "This enum indicates capability for both levels.";
         }
       }
       default "level-all";
       description
         "This type defines IS-IS level of an object.";

     }
```

```
typedef adj-state-type {
  type enumeration {
    enum "up" {
      description
        "State indicates the adjacency is established.";
    }
    enum "down" {
      description
        "State indicates the adjacency is NOT established.";
    }
    enum "init" {
      description
        "State indicates the adjacency is establishing.";
    }
    enum "failed" {
      description
        "State indicates the adjacency is failed.";
    }
  }
  description
    "This type defines states of an adjacency";
}

typedef if-state-type {
  type enumeration {
    enum "up" {
      description "Up state.";

    }
    enum "down" {
      description "Down state";
    }
  }
  description
    "This type defines the state of an interface";
}

typedef level-number {
  type uint8 {
    range "1 .. 2";
  }
  description
    "This type defines the current IS-IS level.";
}

typedef lsp-id {
  type string {
    pattern
```

```
            '[0-9A-Fa-f]{4}\.[0-9A-Fa-f]{4}\.[0-9A-Fa-f]'
            +'{4}\.[0-9][0-9]-[0-9][0-9]';
        }
        description
          "This type defines the IS-IS LSP ID format using a
           pattern, An example LSP ID is 0143.0438.AeF0.02-01";
      }

      typedef area-address {
        type string {
          pattern '[0-9A-Fa-f]{2}\.([0-9A-Fa-f]{4}\.){0,3}';
        }
        description
          "This type defines the area address format.";
      }

      typedef snpa {
        type string {
          length "0 .. 20";
        }
        description
          "This type defines the Subnetwork Point
           of Attachement (SNPA) format.";
      }

      typedef system-id {
        type string {
          pattern
            '[0-9A-Fa-f]{4}\.[0-9A-Fa-f]{4}\.[0-9A-Fa-f]{4}';
        }
        description
          "This type defines IS-IS system-id using pattern,
           An example system-id is 0143.0438.AeF0";
      }

      typedef wide-metric {
        type uint32 {
          range "0 .. 16777215";
        }
        description
          "This type defines wide style format of IS-IS metric.";
      }

      typedef std-metric {
        type uint8 {
          range "0 .. 63";
        }
        description
```

```
        "This type defines old style format of IS-IS metric.";
    }

    typedef mesh-group-state {
      type enumeration {
        enum "mesh-inactive" {
          description
            "Interface is not part of a mesh group.";
        }
        enum "mesh-set" {
          description
            "Interface is part of a mesh group.";
        }
        enum "mesh-blocked" {
          description
            "LSPs must not be flooded over this interface.";
        }
      }
      description
        "This type describes mesh group state of an interface";
    }


    /* Grouping definitions for configuration and ops state */


    grouping adjacency-state {
      container adjacencies {
        config false;
        list adjacency {
          leaf neighbor-sys-type {
            type level;
            description
              "Level capability of neighboring system";
          }
          leaf neighbor-sysid {
            type system-id;
            description
              "The system-id of the neighbor";
          }
          leaf neighbor-extended-circuit-id {
            type extended-circuit-id;
            description
              "Circuit ID of the neighbor";
          }
          leaf neighbor-snpa {
            type snpa;
            description
```

```
                "SNPA of the neighbor";
            }
            leaf usage {
              type level;
              description
                "Define the level(s) activated on the adjacency.
                 On a p2p link this might be level 1 and 2,
                 but on a LAN, the usage will be level 1
                 between peers at level 1 or level 2 between
                 peers at level 2.";
            }
            leaf hold-timer {
              type rt-types:timer-value-seconds16;
              units seconds;
              description
                "The holding time in seconds for this
                 adjacency. This value is based on
                 received hello PDUs and the elapsed
                 time since receipt.";
            }
            leaf neighbor-priority {
              type uint8 {
                range "0 .. 127";
              }
              description
                "Priority of the neighboring IS for becoming
                 the DIS.";
            }
            leaf lastuptime {
              type yang:timestamp;
              description
                "When the adjacency most recently entered
                 state 'up', measured in hundredths of a
                 second since the last reinitialization of
                 the network management subsystem.
                 The value is 0 if the adjacency has never
                 been in state 'up'.";
            }
            leaf state {
              type adj-state-type;
              description
                "This leaf describes the state of the interface.";
            }

            description
              "List of operational adjacencies.";
          }
          description
```

```
           "This container lists the adjacencies of
            the local node.";
       }
       description
         "Adjacency state";
     }

     grouping fast-reroute-global-state {
       container protected-routes {
         config false;
         list af-stats {
           key "af prefix alternate";

           leaf af {
             type iana-rt-types:address-family;
             description
               "Address-family";
           }
           leaf prefix {
             type string;
             description
               "Protected prefix.";
           }
           leaf alternate {
             type string;
             description
               "Alternate nexthop for the prefix.";
           }
           leaf alternate-type {
             type enumeration {
               enum equal-cost {
                 description
                   "ECMP alternate.";
               }
               enum lfa {
                 description
                   "LFA alternate.";
               }
               enum remote-lfa {
                 description
                   "Remote LFA alternate.";
               }
               enum tunnel {
                 description
                   "Tunnel based alternate
                    (like RSVP-TE or GRE).";
               }
               enum ti-lfa {
```

```
                description
                  "TI-LFA alternate.";
              }
              enum mrt {
                description
                  "MRT alternate.";
              }
              enum other {
                description
                  "Unknown alternate type.";
              }
            }
            description
              "Type of alternate.";
          }
          leaf best {
            type boolean;
            description
              "Indicates if the alternate is the preferred.";
          }
          leaf non-best-reason {
            type string;
            description
              "Information field to describe why the alternate
               is not best.";
          }
          leaf protection-available {
            type bits {
              bit node-protect {
                position 0;
                description
                  "Node protection available.";
              }
              bit link-protect {
                position 1;
                description
                  "Link protection available.";
              }
              bit srlg-protect {
                position 2;
                description
                  "SRLG protection available.";
              }
              bit downstream-protect {
                position 3;
                description
                  "Downstream protection available.";
              }
```

```
              bit other {
                position 4;
                description
                  "Other protection available.";
              }
            }
            description "Protection provided by the alternate.";
          }
          leaf alternate-metric1 {
            type uint32;
            description
              "Metric from Point of Local Repair (PLR) to
               destination through the alternate path.";
          }
          leaf alternate-metric2 {
            type uint32;
            description
              "Metric from PLR to the alternate node";
          }
          leaf alternate-metric3 {
            type uint32;
            description
              "Metric from alternate node to the destination";
          }
          description
            "Per-AF protected prefix statistics.";
        }
        description
          "List of prefixes that are protected.";
      }

      container unprotected-routes {
        config false;
        list af-stats {
          key "af prefix";

          leaf af {
            type iana-rt-types:address-family;

            description "Address-family";
          }
          leaf prefix {
            type string;
            description "Unprotected prefix.";
          }
          description
            "Per AF unprotected prefix statistics.";
        }
```

```
          description
            "List of prefixes that are not protected.";
        }

        list protection-statistics {
          key frr-protection-method;
          config false;
          leaf frr-protection-method {
            type string;
            description "Protection method used.";
          }
          list af-stats {
            key af;

            leaf af {
              type iana-rt-types:address-family;

              description "Address-family";
            }
            leaf total-routes {
              type uint32;
              description "Total prefixes.";
            }
            leaf unprotected-routes {
              type uint32;
              description
                "Total prefixes that are not protected.";
            }
            leaf protected-routes {
              type uint32;
              description
                "Total prefixes that are protected.";
            }
            leaf linkprotected-routes {
              type uint32;
              description
                "Total prefixes that are link protected.";
            }
            leaf nodeprotected-routes {
              type uint32;
              description
                "Total prefixes that are node protected.";
            }
            description
              "Per AF protected prefix statistics.";
          }

          description "Global protection statistics.";
```

```
      }
      description "IPFRR states.";
    }

    grouping notification-instance-hdr {
      description
        "Instance specific IS-IS notification data grouping";
      leaf routing-instance {
        type string;
        description "Name of the routing-instance instance.";
      }
      leaf routing-protocol-name {
        type string;
        description "Name of the IS-IS instance.";
      }
      leaf isis-level {
        type level;
        description "IS-IS level of the instance.";
      }
    }

    grouping notification-interface-hdr {
      description
        "Interface specific IS-IS notification data grouping";
      leaf interface-name {
        type string;
        description "IS-IS interface name";
      }
      leaf interface-level {
        type level;
        description "IS-IS level of the interface.";
      }
      leaf extended-circuit-id {
        type extended-circuit-id;
        description "Eextended circuit-id of the interface.";
      }
    }

    grouping route-content {
      description
        "IS-IS protocol-specific route properties grouping.";
      leaf metric {
        type uint32;
        description "IS-IS metric of a route.";
      }
      leaf-list tag {
        type uint64;
        description
```

```
          "List of tags associated with the route. The leaf
           describes both 32-bit and 64-bit tags.";
      }
      leaf route-type {
        type enumeration {
          enum l2-up-internal {
            description "Level 2 internal route
                         and not leaked to a lower level";
          }
          enum l1-up-internal {
            description "Level 1 internal route
                         and not leaked to a lower level";
          }
          enum l2-up-external {
            description "Level 2 external route
                         and not leaked to a lower level";
          }
          enum l1-up-external {
            description "Level 1 external route
                         and not leaked to a lower level";
          }
          enum l2-down-internal {
            description "Level 2 internal route
                         and leaked to a lower level";
          }
          enum l1-down-internal {
            description "Level 1 internal route
                         and leaked to a lower level";
          }
          enum l2-down-external {
            description "Level 2 external route
                         and leaked to a lower level";
          }
          enum l1-down-external {
            description "Level 1 external route
                         and leaked to a lower level";
          }
        }
        description "IS-IS route type.";
      }
    }

    grouping admin-control {
      leaf enable {
        if-feature admin-control;
        type boolean;
        default true;
        description
```

```
            "Control the administrative state.";
        }
        description
          "Grouping for admin control.";
      }

      grouping fast-reroute-global-cfg {
        description
          "IPFRR global configuration grouping";
        container lfa {
          if-feature lfa;
          description
            "This container may be augmented with global
             parameters for LFA. Creating the container has no
             effect on LFA activation.";
        }
      }

      grouping fast-reroute-if-cfg {
        description
          "IPFRR interface configuration grouping";
        container lfa {
          if-feature lfa;
          uses lfa-if-cfg;
          container level-1 {
            uses lfa-if-cfg;
            description
              "LFA level 1 config";
          }
          container level-2 {
            uses lfa-if-cfg;
            description
              "LFA level 2 config";
          }
          description
            "LFA config";
        }
      }

      grouping ietf-spf-delay-cfg {
        leaf initial-delay {
          type rt-types:timer-value-milliseconds;
          units msec;
          description
            "Delay used while in QUIET state.";
        }
        leaf short-delay {
          type rt-types:timer-value-milliseconds;
```

```
          units msec;
          description
            "Delay used while in SHORT_WAIT state.";
        }
        leaf long-delay {
          type rt-types:timer-value-milliseconds;
          units msec;
          description
            "Delay used while in LONG_WAIT state.";
        }

        leaf hold-down {
          type rt-types:timer-value-milliseconds;
          units msec;
          description
            "Timer used to consider an IGP stability period.";
        }
        leaf time-to-learn {
          type rt-types:timer-value-milliseconds;
          units msec;
          description
            "Duration used to learn all the IGP events
             related to a single component failure.";
        }

        description
          "Grouping for IETF SPF delay configuration.";
      }

      grouping ietf-spf-delay-state {
        leaf current-state {
          type enumeration {
            enum "quiet" {
              description "QUIET state";
            }
            enum "short-wait" {
              description "SHORT_WAIT state";
            }
            enum "long-wait" {
              description "LONG_WAIT state";
            }
          }
          config false;
          description
            "Current SPF backoff algorithm state.";
        }
        leaf remaining-time-to-learn {
          type rt-types:timer-value-milliseconds;
```

```
        units "msec";
        config false;
        description
          "Remaining time until time-to-learn timer fires.";
      }
      leaf remaining-hold-down {
        type rt-types:timer-value-milliseconds;
        units "msec";
        config false;
        description
          "Remaining time until hold-down timer fires.";
      }
      leaf last-event-received {
        type yang:timestamp;
        config false;
        description
          "Time of last IGP event received";
      }
      leaf next-spf-time {
        type yang:timestamp;
        config false;
        description
          "Time when next SPF has been scheduled.";
      }
      leaf last-spf-time {
        type yang:timestamp;
        config false;
        description
          "Time of last SPF computation.";
      }
      description
        "Grouping for IETF SPF delay operational states.";
    }

    grouping local-rib {
      description "Local-rib grouping.";
      container local-rib {
        config false;
        description "Local-rib.";
        list route {
          key "prefix";
          description "List of IS-IS local RIB Routes";
          leaf prefix {
            type inet:ip-prefix;
            description "Destination prefix.";
          }
          container next-hops {
            description "All next hops for the route.";
```

```
            list next-hop {
              key "next-hop";
              description "List of next hop for the route";
              leaf outgoing-interface {
                type if:interface-ref;
                description
                  "Name of the outgoing interface.";
              }
              leaf next-hop {
                type inet:ip-address;
                description "Nexthop address.";
              }
            }
          }
          leaf metric {
            type uint32;
            description "Metric for this route.";
          }
          leaf level {
            type level-number;
            description "Level number for this route.";
          }
          leaf route-tag {
            type uint32;
            description "Route tag for this route.";
          }
        }
      }
    }

    grouping isis-node-tag-cfg {
      description "IS-IS node tag config.";
      container node-tags {
        if-feature node-tag;
        list node-tag {
          key tag;
          leaf tag {
            type uint32;
            description
              "Node tag value.";
          }
          description
            "List of tags.";
        }
        description
          "Container for node tags.";
      }
    }
```

```
     grouping authentication-global-cfg {
       choice authentication-type {
         case key-chain {
           if-feature key-chain;
           leaf key-chain {
             type key-chain:key-chain-ref;
             description
               "Reference to a key-chain.";
           }
         }
         case password {
           leaf key {
             type string;
             description
               "This leaf specifies the authentication key.";
           }
           leaf crypto-algorithm {
             type identityref {
               base key-chain:crypto-algorithm;
             }
             description
               "Cryptographic algorithm associated with key.";
           }
         }
         description "Choice of authentication.";
       }
       description "Grouping for global authentication config.";
     }

     grouping metric-type-global-cfg {
       leaf value {
         type enumeration {
           enum wide-only {
             description
               "Advertise new metric style only (RFC5305)";
           }
           enum old-only {
             description
               "Advertise old metric style only (RFC1195)";
           }
           enum both {
             description "Advertise both metricstyles";
           }
         }
         default wide-only;
         description
           "Type of metric to be generated:
            - wide-only means only new metric style
```

```
               is generated,
             - old-only means that only old style metric
               is generated,
             - both means that both are advertised.
             This leaf is only affecting IPv4 metrics.";
       }
       description
         "Grouping for global metric style config.";
     }

     grouping default-metric-global-cfg {
       leaf value {
         type wide-metric;
         default "10";
         description  "Value of the metric";
       }
       description
         "Global default metric config grouping.";
     }


     grouping overload-global-cfg {
       leaf status {
         type boolean;
         default false;
         description
           "This leaf specifies the overload status.";
       }
       description "Grouping for overload bit config.";
     }

     grouping overload-max-metric-global-cfg {
       leaf timeout {
         type rt-types:timer-value-seconds16;
         units "seconds";
         description
           "Timeout (in seconds) of the overload condition.";
       }
       description
         "Overload maximum metric configuration grouping";
     }

     grouping route-preference-global-cfg {
       choice granularity {
         case detail {
           leaf internal {
             type uint8;
             description
```

```
              "Protocol preference for internal routes.";
          }
          leaf external {
            type uint8;
            description
              "Protocol preference for external routes.";
          }
        }
        case coarse {
          leaf default {
            type uint8;
            description
              "Protocol preference for all IS-IS routes.";
          }
        }
        description
          "Choice for implementation of route preference.";
      }
      description
        "Global route preference grouping";
    }

    grouping hello-authentication-cfg {
      choice authentication-type {
        case key-chain {
          if-feature key-chain;
          leaf key-chain {
            type key-chain:key-chain-ref;
            description "Reference to a key-chain.";
          }
        }
        case password {
          leaf key {
            type string;
            description "Authentication key specification";
          }
          leaf crypto-algorithm {
            type identityref {
              base key-chain:crypto-algorithm;
            }
            description
              "Cryptographic algorithm associated with key.";
          }
        }
        description "Choice of authentication.";
      }
      description "Grouping for hello authentication.";
    }
```

```
     grouping hello-interval-cfg {
       leaf value {
         type rt-types:timer-value-seconds16;
         units "seconds";
         default 10;
         description
           "Interval (in seconds) between successive hello
            messages.";
       }

       description "Interval between hello messages.";
     }

     grouping hello-multiplier-cfg {
       leaf value {
         type uint16;
         default 3;
         description
           "Number of missed hello messages prior to
            declaring the adjacency down.";
       }
       description
           "Number of missed hello messages prior to
            adjacency down grouping.";
     }

     grouping priority-cfg {
       leaf value {
         type uint8 {
           range "0 .. 127";
         }
         default 64;
         description
           "Priority of interface for DIS election.";
       }

       description "Interface DIS election priority grouping";
     }

     grouping metric-cfg {
       leaf value {
         type wide-metric;
         default "10";
         description "Metric value.";
       }
       description "Interface metric grouping";
     }
```

```
     grouping lfa-if-cfg {
       leaf candidate-disabled {
         type boolean;
         default false;
         description
           "Prevent the interface to be used as backup.";
       }
       leaf enable {
         type boolean;
         default false;
         description
           "LFA Activation - this model assumes activation
            of per-prefix LFA.";
       }

       container remote-lfa {
         if-feature remote-lfa;
         leaf enable {
           type boolean;
           default false;
           description
             "Activates rLFA.";
         }
         description "Remote LFA configuration.";
       }
       description "Grouping for LFA interface configuration";
     }

     grouping isis-global-cfg {
       description "IS-IS global configuration grouping";

       uses admin-control;

       leaf level-type {
         type level;
         default "level-all";
         description
           "Level of an IS-IS node - can be level-1-only,
            level-2-only or level-1-2.";
       }

       leaf system-id {
         type system-id;
         description "System-id of the node.";
       }

       leaf maximum-area-addresses {
         if-feature maximum-area-addresses;
```

```
          type uint8;
          default 3;
          description "Maximum areas supported.";
        }

        leaf-list area-address {
          type area-address;
          description
            "List of areas supported by the protocol instance.";
        }

        container mpls {
          leaf ipv4-router-id {
            if-feature ipv4-router-id;
            type inet:ipv4-address;
            description
              "Router ID value that would be used in TLV 134.";
          }
          leaf ipv6-router-id {
            if-feature ipv6-router-id;
            type inet:ipv6-address;
            description
              "Router ID value that would be used in TLV 140.";
          }
          container ldp {
            container igp-sync {
              if-feature ldp-igp-sync;
              description
                "This container may be augmented with global
                 parameters for igp-ldp-sync.";
            }
            description "LDP configuration.";
          }
          description "MPLS configuration";
        }
        container auto-cost {
          if-feature auto-cost;
          leaf reference-bandwidth {
            type uint32;
            units "bps";
            description "Bandwidth for calculating metric.";
          }
          leaf enable {
            type boolean;
            default false;
            description "Enable/disable auto-cost.";
          }
          description "Auto-cost configuration.";
```

```
        }
        leaf lsp-mtu {
          type uint16;
          units "bytes";
          default 1492;
          description
            "Maximum size of an LSP PDU in bytes.";
        }
        leaf lsp-lifetime {
          type uint16 {
            range "1..65535";
          }
          units "seconds";
          description
            "Lifetime of the router's LSPs in seconds.";
        }
        leaf lsp-refresh {
          if-feature lsp-refresh;
          type rt-types:timer-value-seconds16;
          units "seconds";
          description
            "Refresh interval of the router's LSPs in seconds.";
        }
        container graceful-restart {
          if-feature graceful-restart;
          leaf enable {
            type boolean;
            default false;
            description "Enable graceful restart.";
          }
          leaf restart-interval {
            type rt-types:timer-value-seconds16;
            units "seconds";
            description
              "Interval (in seconds) to attempt graceful restart prior
               to failure.";
          }
          leaf helper-enable {
            type boolean;
            default true;
            description
              "Enable local IS-IS router as graceful restart helper.";
          }
          description "Graceful-Restart Configuration.";
        }

        container nsr {
          if-feature nsr;
```

```
          description "Non-Stop Routing (NSR) configuration.";
          leaf enable {
            type boolean;
            default false;
            description "Enable/Disable Non-Stop Routing (NSR).";
          }
        }

        uses isis-node-tag-cfg;

        container authentication {
          uses authentication-global-cfg;

          container level-1 {
            uses authentication-global-cfg;
            description "level-1 specific configuration";
          }
          container level-2 {
            uses authentication-global-cfg;
            description "level-2 specific configuration";
          }
          description "Authentication global configuration for
                       both LSPs and SNPs.";
        }

        container metric-type {
          uses metric-type-global-cfg;
          container level-1 {
            uses metric-type-global-cfg;
            description "level-1 specific configuration";
          }
          container level-2 {
            uses metric-type-global-cfg;
            description "level-2 specific configuration";
          }
          description "Metric style global configuration";
        }

        container default-metric {
          uses default-metric-global-cfg;
          container level-1 {
            uses default-metric-global-cfg;
            description "level-1 specific configuration";
          }
          container level-2 {
            uses default-metric-global-cfg;
            description "level-2 specific configuration";
          }
```

```
        description "Default metric global configuration";
      }

      container afs {
        if-feature nlpid-control;
        list af {
          key af;
          leaf af {
            type iana-rt-types:address-family;
            description "Address-family";
          }
          leaf enable {
            type boolean;
            description "Activate the address family.";
          }
          description
            "List of address families and whether or not they
             are activated.";
        }
        description "Address Family configuration";
      }

      container preference {
        uses route-preference-global-cfg;
        description "Router preference configuration for IS-IS
                     protocol instance route installation";
      }

      container overload {
        uses overload-global-cfg;
        description "Router protocol instance overload state
                     configuration";
      }

      container overload-max-metric {
        if-feature overload-max-metric;
        uses overload-max-metric-global-cfg;
        description
          "Router protocol instance overload maximum
           metric advertisement configuration.";
      }
    }

    grouping isis-global-topologies-cfg {
      description "Per-topology configuration";
      container default-metric {
        uses default-metric-global-cfg;
        container level-1 {
```

```
            uses default-metric-global-cfg;
            description "level-1 specific configuration";
          }
          container level-2 {
            uses default-metric-global-cfg;
            description "level-2 specific configuration";
          }
          description "Default metric per-topology configuration";
        }
        uses isis-node-tag-cfg;
      }

      grouping isis-if-cfg {
        description "Interface configuration grouping";
        leaf level-type {
          type level;
          default "level-all";
          description "IS-IS level of the interface.";
        }
        leaf lsp-pacing-interval {
          type rt-types:timer-value-milliseconds;
          units "milliseconds";
          default 33;
          description
            "Interval (in milli-seconds) between LSP
             transmissions.";
        }
        leaf lsp-retransmit-interval {
          type rt-types:timer-value-seconds16;
          units "seconds";
          description
            "Interval (in seconds) between LSP
             retransmissions.";
        }
        leaf passive {
          type boolean;
          default "false";
          description
            "Indicates whetherthe interface is in passive mode (IS-IS
             not running but network is advertised).";
        }
        leaf csnp-interval {
          type rt-types:timer-value-seconds16;
          units "seconds";
          default 10;
          description
             "Interval (in seconds) between CSNP messages.";
        }
```

```
        container hello-padding {
          leaf enable {
            type boolean;
            default "true";
            description
              "IS-IS Hello-padding activation - enabled by default.";
          }
          description "IS-IS hello padding configuration.";
        }
        leaf mesh-group-enable {
          type mesh-group-state;
          description "IS-IS interface mesh-group state";
        }
        leaf mesh-group {
          when "../mesh-group-enable = 'mesh-set'" {
            description
              "Only valid when mesh-group-enable equals meshset";
          }
          type uint8;
          description "IS-IS interface mesh-group ID.";
        }
        leaf interface-type {
          type interface-type;
          default "broadcast";
          description
            "Type of adjacency to be established on the interface. This
             dictates the type of hello messages that are used.";
        }

        uses admin-control;

        leaf-list tag {
          if-feature prefix-tag;
          type uint32;
          description
            "List of tags associated with the interface.";
        }
        leaf-list tag64 {
          if-feature prefix-tag64;
          type uint64;
          description
            "List of 64-bit tags associated with the interface.";
        }
        leaf node-flag {
          if-feature node-flag;
          type boolean;
          default false;
          description
```

```
            "Set prefix as a node representative prefix.";
        }
        container hello-authentication {
          uses hello-authentication-cfg;
          container level-1 {
            uses hello-authentication-cfg;
            description "level-1 specific configuration";
          }
          container level-2 {
            uses hello-authentication-cfg;
            description "level-2 specific configuration";
          }
          description
            "Authentication type to be used in hello messages.";
        }
        container hello-interval {
          uses hello-interval-cfg;
          container level-1 {
            uses hello-interval-cfg;
            description "level-1 specific configuration";
          }
          container level-2 {
            uses hello-interval-cfg;
            description "level-2 specific configuration";
          }
          description "Interval between hello messages.";
        }
        container hello-multiplier {
          uses hello-multiplier-cfg;
          container level-1 {
            uses hello-multiplier-cfg;
            description "level-1 specific configuration";
          }
          container level-2 {
            uses hello-multiplier-cfg;
            description "level-2 specific configuration";
          }
          description "Hello multiplier configuration.";
        }
        container priority {
          must '../interface-type = "broadcast"' {
            error-message
              "Priority only applies to broadcast interfaces.";
            description "Check for broadcast interface.";
          }
          uses priority-cfg;
          container level-1 {
            uses priority-cfg;
```

```
          description "level-1 specific configuration";
        }
        container level-2 {
          uses priority-cfg;
          description "level-2 specific configuration";
        }
        description "Priority for DIS election.";
      }
      container metric {
        uses metric-cfg;
        container level-1 {
          uses metric-cfg;
          description "level-1 specific configuration";
        }
        container level-2 {
          uses metric-cfg;
          description "level-2 specific configuration";
        }
        description "Metric configuration.";
      }
      container bfd {
        if-feature bfd;
        description "BFD Client Configuration.";
        uses bfd-types:client-cfg-parms;

        reference "draft-ietf-bfd-yang-xx.txt:
                   YANG Data Model for Bidirectional Forwarding
                   Detection (BFD)";
      }
      container afs {
        if-feature nlpid-control;
        list af {
          key af;
          leaf af {
            type iana-rt-types:address-family;
            description  "Address-family";
          }
          description "List of AFs.";
        }
        description "Interface address-families";
      }
      container mpls {
        container ldp {
          leaf igp-sync {
            if-feature ldp-igp-sync;
            type boolean;
            default false;
            description "Enables IGP/LDP synchronization";
```

```
          }
          description "LDP protocol related configuration.";
        }
        description "MPLS configuration for IS-IS interfaces";
      }
    }

    grouping isis-if-topologies-cfg {
      description "IS-IS interface topology configuration.";
      container metric {
        uses metric-cfg;
        container level-1 {
          uses metric-cfg;
          description "level-1 specific configuration";
        }
        container level-2 {
          uses metric-cfg;
          description "level-2 specific configuration";
        }
        description "Metric IS-IS interface configuration.";
      }
    }

    grouping system-counters {
      container system-counters {
        config false;
        list level {
          key level;

          leaf level {
            type level-number;
            description "IS-IS level.";
          }
          leaf corrupted-lsps {
            type uint32;
            description
              "Number of corrupted in-memory LSPs detected.
               LSPs received from the wire with a bad
               checksum are silently dropped and not counted.
               LSPs received from the wire with parse errors
               are counted by lsp-errors.";
          }
          leaf authentication-type-fails {
            type uint32;
            description
               "Number of authentication type mismatches.";
          }
          leaf authentication-fails {
```

```
              type uint32;
              description
                "Number of authentication key failures.";
            }
            leaf database-overload {
              type uint32;
              description
                "Number of times the database has become
                 overloaded.";
            }
            leaf own-lsp-purge {
              type uint32;
              description
                "Number of times a zero-aged copy of the system's
                 own LSP is received from some other IS-IS node.";
            }
            leaf manual-address-drop-from-area {
              type uint32;
              description
                "Number of times a manual address
                 has been dropped from the area.";
            }
            leaf max-sequence {
              type uint32;
              description
                "Number of times the system has attempted
                 to exceed the maximum sequence number.";
            }
            leaf sequence-number-skipped {
              type uint32;
              description
                "Number of times a sequence number skip has
                 occured.";
            }
            leaf id-len-mismatch {
              type uint32;
              description
                "Number of times a PDU is received with a
                 different value for the ID field length
                 than that of the receiving system.";
            }
            leaf partition-changes {
              type uint32;
              description
                "Number of partition changes detected.";
            }
            leaf lsp-errors {
              type uint32;
```

```
            description
              "Number of LSPs with errors we have received.";
          }
          leaf spf-runs {
            type uint32;
            description
              "Number of times we ran SPF at this level.";
          }
          description
            "List of supported levels.";
        }
        description
          "List counters for the IS-IS protocol instance";
      }
      description "System counters grouping.";
    }

    grouping event-counters {
      container event-counters {
        config false;
        leaf adjacency-changes {
          type uint32;
          description
            "The number of times an adjacency state change has
             occured on this interface.";
        }
        leaf adjacency-number {
          type uint32;
          description
            "The number of adjacencies on this interface.";
        }
        leaf init-fails {
          type uint32;
          description
            "The number of times initialization of this
             interface has failed. This counts events such
             as PPP NCP failures. Failures to form an
             adjacency are counted by adjacency-rejects.";
        }
        leaf adjacency-rejects {
          type uint32;
          description
            "The number of times an adjacency has been
             rejected on this interface.";
        }
        leaf id-len-mismatch {
          type uint32;
          description
```

```
            "The number of times an IS-IS PDU with an ID
             field length different from that for this
             system has been received on this interface.";
        }
        leaf max-area-addresses-mismatch {
          type uint32;
          description
            "The number of times an IS-IS PDU has been
             received on this interface with the
             max area address field differing from that of
             this system.";
        }
        leaf authentication-type-fails {
          type uint32;
          description
            "Number of authentication type mismatches.";
        }
        leaf authentication-fails {
          type uint32;
          description
            "Number of authentication key failures.";
        }
        leaf lan-dis-changes {
          type uint32;
          description
            "The number of times the DIS has changed on this
             interface at this level. If the interface type is
             point-to-point,the count is zero.";
        }
        description "IS-IS interface event counters.";
      }
      description
        "Grouping for IS-IS interface event counters";
    }

    grouping packet-counters {
      container packet-counters {
        config false;
        list level {
          key level;

          leaf level {
            type level-number;
            description "IS-IS level.";
          }
          container iih {
            leaf in {
              type uint32;
```

```
                description "Received IIH PDUs.";
              }
              leaf out {
                type uint32;
                description "Sent IIH PDUs.";
              }
              description "Number of IIH PDUs received/sent.";
            }
            container ish {
              leaf in {
                type uint32;
                description "Received ISH PDUs.";
              }
              leaf out {
                type uint32;
                description "Sent ISH PDUs.";
              }
              description
                "ISH PDUs received/sent.";
            }
            container esh {
              leaf in {
                type uint32;
                description "Received ESH PDUs.";
              }
              leaf out {
                type uint32;
                description "Sent ESH PDUs.";
              }
              description "Number of ESH PDUs received/sent.";
            }
            container lsp {
              leaf in {
                type uint32;
                description "Received LSP PDUs.";
              }
              leaf out {
                type uint32;
                description "Sent LSP PDUs.";
              }
              description "Number of LSP PDUs received/sent.";
            }
            container psnp {
              leaf in {
                type uint32;
                description "Received PSNP PDUs.";
              }
              leaf out {
```

```
              type uint32;
              description "Sent PSNP PDUs.";
            }
            description "Number of PSNP PDUs received/sent.";
          }
          container csnp {
            leaf in {
              type uint32;
              description "Received CSNP PDUs.";
            }
            leaf out {
              type uint32;
              description "Sent CSNP PDUs.";
            }
            description "Number of CSNP PDUs received/sent.";
          }
          container unknown {
            leaf in {
              type uint32;
              description "Received unknown PDUs.";
            }
            leaf out {
              type uint32;
              description "Sent unknown PDUs.";
            }
            description "Number of unknown PDUs received/sent.";
          }
          description
            "List of packet counter for supported llevels.";
        }
        description "Packet counters per IS-IS level.";
      }
      description
        "Grouping for per IS-IS Level packet counters.";
    }

    grouping spf-log {
      container spf-log {
        config false;
        list event {
          key id;

          leaf id {
            type uint32;
            description
              "Event identifier -  purely internal value.";
          }
          leaf spf-type {
```

```
              type enumeration {
                enum full {
                  description "Full SPF computation.";
                }
                enum route-only {
                  description
                    "Route reachability only SPF computation";
                }
              }
              description "Type of SPF computation performed.";
            }
            leaf level {
              type level-number;
              description
                "IS-IS level number for SPF computation";
            }
            leaf schedule-timestamp {
              type yang:timestamp;
              description
                "Timestamp of when the SPF computation was
                 scheduled.";
            }
            leaf start-timestamp {
              type yang:timestamp;
              description
                "Timestamp of when the SPF computation started.";
            }
            leaf end-timestamp {
              type yang:timestamp;
              description
                "Timestamp of when the SPF computation ended.";
            }
            list trigger-lsp {
              key "lsp";
              leaf lsp {
                type lsp-id;
                description
                  "LSPID of the LSP triggering SPF computation.";
              }
              leaf sequence {
                type uint32;
                description
                  "Sequence number of the LSP triggering SPF
                   computation";
              }
              description
                "This list includes the LSPs that triggered the
                 SPF computation.";
```

```
          }
          description
            "List of computation events - implemented as a
             wrapping buffer.";
        }

        description
          "This container lists the SPF computation events.";
      }
      description "Grouping for spf-log events.";
    }

    grouping lsp-log {
      container lsp-log {
        config false;
        list event {
          key id;

          leaf id {
            type uint32;
            description
              "Event identifier -  purely internal value.";
          }
          leaf level {
            type level-number;
            description
              "IS-IS level number for LSP";
          }
          container lsp {
            leaf lsp {

              type lsp-id;
              description
                "LSPID of the LSP.";
            }
            leaf sequence {
              type uint32;
              description
                "Sequence number of the LSP.";
            }
            description
              "LSP identification container - either the received
               LSP or the locally generated LSP.";
          }

          leaf received-timestamp {
            type yang:timestamp;
```

```
            description
              "Timestamp of when the LSP was received. In case
               of local LSP update, the timestamp refers to the
               local LSP update time.";
          }

          leaf change {
            type identityref {
              base lsp-log-reason;
            }
            description "Type of LSP change.";
          }

          description
            "List of LSP events - implemented as a
             wrapping buffer.";
        }

        description
          "LSP reception and local LSP origination events
           container.";

      } description "Grouping for LSP log.";
    }

    grouping hostname-db {
      container hostnames {
        config false;
        list hostname {
          key system-id;
          leaf system-id {
            type system-id;
            description
              "System-id associated with the hostname.";
          }
          leaf hostname {
            type string;
            description
              "Hostname associated with the system ID.";
          }
          description
            "List of system-id/hostname associations.";
        }
        description
          "Hostname to system-id mapping database.";
      }
      description
        "Grouping for hostname to system-id mapping database.";
```

```
     }

     /* Groupings for the LSDB description */

     grouping prefix-reachability-attributes {
       description
         "Grouping for extended reachability attributes of an
          IPv4 or IPv6 prefix.";

       leaf external-prefix-flag {
         type boolean;
         description "External prefix flag.";
       }
       leaf readvertisement-flag {
         type boolean;
         description "Readvertisement flag.";
       }
       leaf node-flag {
         type boolean;
         description "Node flag.";
       }
     }

     grouping prefix-ipv4-source-router-id {
       description
         "Grouping for the IPv4 source router ID of a prefix
          advertisement.";

       leaf ipv4-source-router-id {
         type inet:ipv4-address;
         description "IPv4 Source router ID address.";
       }
     }

     grouping prefix-ipv6-source-router-id {
       description
         "Grouping for the IPv6 source router ID of a prefix
          advertisement.";

       leaf ipv6-source-router-id {
         type inet:ipv6-address;
         description "IPv6 Source router ID address.";
       }
     }

     grouping prefix-attributes-extension {
       description "Prefix extended attributes.";
```

```
        uses prefix-reachability-attributes;
        uses prefix-ipv4-source-router-id;
        uses prefix-ipv6-source-router-id;
      }

      grouping prefix-ipv4-std {
        description
          "Grouping for attributes of an IPv4 standard prefix.";
        leaf up-down {
          type boolean;
          description "Value of up/down bit.";
        }
        leaf i-e {
          type boolean;
          description "Value of I/E bit.";
        }
        leaf ip-prefix {
          type inet:ipv4-address;
          description "IPv4 prefix address";
        }
        leaf prefix-len {
          type uint8;
          description "IPv4 prefix length (in bits)";
        }
        leaf default-metric {
          type std-metric;
          description "Default IS-IS metric for IPv4 prefix";
        }
        container delay-metric {
          leaf metric {
            type std-metric;
            description "IS-IS delay metric for IPv4 prefix";
          }
          leaf supported {
            type boolean;
            default "false";
            description
              "Indicates whether IS-IS delay metric is supported.";
          }
          description "IS-IS delay metric container.";
        }
        container expense-metric {
          leaf metric {
            type std-metric;
            description "IS-IS expense metric for IPv4 prefix";
          }
          leaf supported {
            type boolean;
```

```
            default "false";
            description
              "Indicates whether IS-IS delay metric is supported.";
          }
          description "IS-IS expense metric container.";
        }
        container error-metric {
          leaf metric {
            type std-metric;
            description
              "This leaf describes the IS-IS error metric value";
          }
          leaf supported {
            type boolean;
            default "false";
            description "IS-IS error metric for IPv4 prefix";
          }
          description "IS-IS error metric container.";
        }
      }

      grouping prefix-ipv4-extended {
        description
          "Grouping for attributes of an IPv4 extended prefix.";
        leaf up-down {
          type boolean;
          description  "Value of up/down bit.";
        }
        leaf ip-prefix {
          type inet:ipv4-address;
          description "IPv4 prefix address";
        }
        leaf prefix-len {
          type uint8;
          description "IPv4 prefix length (in bits)";
        }

        leaf metric {
          type wide-metric;
          description "IS-IS wide metric value";
        }
        leaf-list tag {
          type uint32;
          description
            "List of 32-bit tags associated with the IPv4 prefix.";
        }
        leaf-list tag64 {
          type uint64;
```

```
        description
          "List of 32-bit tags associated with the IPv4 prefix.";
      }
      uses prefix-attributes-extension;
    }

    grouping prefix-ipv6-extended {
      description "Grouping for attributes of an IPv6 prefix.";
      leaf up-down {
        type boolean;
        description "Value of up/down bit.";
      }
      leaf ip-prefix {
        type inet:ipv6-address;
        description "IPv6 prefix address";
      }
      leaf prefix-len {
        type uint8;
        description  "IPv4 prefix length (in bits)";
      }
      leaf metric {
        type wide-metric;
        description  "IS-IS wide metric value";
      }
      leaf-list tag {
        type uint32;
        description
          "List of 32-bit tags associated with the IPv4 prefix.";
      }
      leaf-list tag64 {
        type uint64;
        description
          "List of 32-bit tags associated with the IPv4 prefix.";
      }
      uses prefix-attributes-extension;
    }

    grouping neighbor-extended {
      description
       "Grouping for attributes of an IS-IS extended neighbor.";
      leaf neighbor-id {
        type system-id;
        description "System-id of the extended neighbor.";
      }
      leaf metric {
        type wide-metric;
        description "IS-IS wide metric for extended neighbor";
      }
```

```
     }

     grouping neighbor {
       description  "IS-IS standard neighbor grouping.";
       leaf neighbor-id {
         type system-id;
         description "IS-IS neighbor system-id";
       }
       leaf i-e {
         type boolean;
         description
           "Internal or External (I/E) Metric bit value";
       }
       leaf default-metric {
         type std-metric;
         description "IS-IS default metric value";
       }
       container delay-metric {
         leaf metric {
           type std-metric;
           description "IS-IS delay metric value";
         }
         leaf supported {
           type boolean;
           default "false";
           description "IS-IS delay metric supported";
         }
         description "IS-IS delay metric container";
       }
       container expense-metric {
         leaf metric {
           type std-metric;
           description "IS-IS delay expense metric value";
         }
         leaf supported {
           type boolean;
           default "false";
           description "IS-IS delay expense metric supported";
         }
         description "IS-IS delay expense metric container";
       }
       container error-metric {
         leaf metric {
           type std-metric;
           description "IS-IS error metric value";
         }
         leaf supported {
           type boolean;
```

```
            default "false";
            description "IS-IS error metric supported";
          }
          description "IS-IS error metric container";
        }
      }

      grouping lsp-entry {
        description "IS-IS LSP database entry groupting";

        leaf decoded-completed {
          type boolean;
          description "IS-IS LSP body fully decoded.";
        }
        leaf raw-data {
          type yang:hex-string;
          description
            "The hexadecial representation of the complete LSP in
             network-byte order (NBO) as received or originated.";
        }
        leaf lsp-id {
          type lsp-id;
          description "LSP ID of the LSP";
        }
        leaf checksum {
          type uint16;
          description "LSP checksum";
        }
        leaf remaining-lifetime {
          type uint16;
          units "seconds";
          description
            "Remaining lifetime (in seconds) until LSP expiration.";
        }
        leaf sequence {
          type uint32;
          description
            "This leaf describes the sequence number of the LSP.";
        }
        leaf attributes {
          type bits {
            bit partitioned {
              description "Originator partition repair supported";
            }
            bit attached-error {
              description
                "If set, the originator is attached to
                 another area using the referred metric.";
```

```
            }
            bit attached-expense {
              description
                "If set, the originator is attached to
                 another area using the referred metric.";
            }
            bit attached-delay {
              description
                "If set, the originator is attached to
                 another area using the referred metric.";
            }
            bit attached-default {
              description
                "If set, the originator is attached to
                 another area using the referred metric.";
            }
            bit overload {
              description
                "If set, the originator is overloaded,
                 and must be avoided in path calculation.";
            }
          }
          description "LSP attributes";
        }

        leaf-list ipv4-addresses {
          type inet:ipv4-address;
          description
            "List of IPv4 addresses of the IS-IS node - IS-IS
             reference is TLV 132.";
        }

        leaf-list ipv6-addresses {
          type inet:ipv6-address;
          description
            "List of IPv6 addresses of the IS-IS node - IS-IS
             reference is TLV 232.";
        }

        leaf ipv4-te-routerid {
          type inet:ipv4-address;
          description
            "IPv4 Traffic Engineering router ID of the IS-IS node -
             IS-IS reference is TLV 134.";
        }

        leaf ipv6-te-routerid {
          type inet:ipv6-address;
```

```
          description
            "IPv6 Traffic Engineering router ID of the IS-IS node -
             IS-IS reference is TLV 140.";
        }

        leaf-list protocol-supported {
          type uint8;
          description
          "List of supported protocols of the IS-IS node -
           IS-IS reference is TLV 129.";
        }

        leaf dynamic-hostname {
          type string;
          description
            "Host Name of the IS-IS node - IS-IS reference
             is TLV 137.";
        }

        container authentication {
          leaf authentication-type {
            type string;
            description
              "Authentication type to be used with IS-IS node.";
          }
          leaf authentication-key {
            type string;
            description
              "Authentication keyto be used. For security reasons,
               the authentication key MUST NOT be presented in
               plaintext format. It is recommended to use an MD5
               hash to present the authentication-key.";
          }
          description
            "IS-IS node authentication information container -
             IS-IS reference is TLV 10.";
        }

        container mt-entries {
          list topology {
            description
              "List of topologies supported";

            leaf MT-ID {
              type uint16 {
                range "0 .. 4095";
              }
              description
```

```
              "Multi-Topolgoy identifier of topology.";
          }

          leaf attributes {
            type bits {
              bit overload {
                description
                  "If set, the originator is overloaded,
                   and must be avoided in path calculation.";
              }
              bit attached {
                description
                  "If set, the originator is attached to
                  another area using the referred metric.";
              }
            }
            description
              "Attributes of the LSP for the associated
               topology.";
          }
        }
        description
          "IS-IS node topology information container -
           IS-IS reference is TLV 229.";
      }

      list router-capabilities {
        leaf flags {
          type bits {
            bit flooding {
              position 0;
              description
                "If the S bit is set, the IS-IS Router CAPABILITY
                 TLV MUST be flooded across the entire routing
                 domain. If the S bit is clear, the TLV MUST NOT
                 be leaked between levels. This bit MUST NOT
                  be altered during the TLV leaking.";
            }
            bit down {
              position 1;
              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 in to prevent
                 TLV looping.";
            }
```

```
          }
          description "Router Capability Flags";
        }
        container node-tags {
          if-feature node-tag;
          list node-tag {
            leaf tag {
              type uint32;
              description "Node tag value.";
            }
            description "List of tags.";
          }
          description "Node Tag container";
        }

        leaf binary {
          type binary;
          description
            "Binary encoding of the IS-IS node capabilities";
        }
        description
          "IS-IS node capabilities container. This container may
           be extended with detailed information -  IS-IS
           reference is TLV 242.";
      }

      container is-neighbor {
        list neighbor {
          uses neighbor;
          description "List of neighbors.";
        }
        description
          "Standard IS neighbors container - IS-IS reference is
           TLV 2.";
      }

      container extended-is-neighbor {
        list neighbor {
          uses neighbor-extended;
          description
            "List of extended IS neighbors";
        }
        description
          "Standard IS extended neighbors container - IS-IS
           reference is TLV 22";
      }

      container ipv4-internal-reachability {
```

```
          list prefixes {
            uses prefix-ipv4-std;
            description "List of prefixes.";
          }
          description
          "IPv4 internal reachability information container - IS-IS
           reference is TLV 128.";
        }

        container ipv4-external-reachability {
          list prefixes {
            uses prefix-ipv4-std;
            description "List of prefixes.";
          }
          description
            "IPv4 external reachability information container -
             IS-IS reference is TLV 130.";
        }

        container extended-ipv4-reachability {
          list prefixes {
            uses prefix-ipv4-extended;
            description "List of prefixes.";
          }
          description
            "IPv4 extended reachability information container -
             IS-IS reference is TLV 135.";
        }

        container mt-is-neighbor {
          list neighbor {
            leaf mt-id {
              type uint16 {
                range "0 .. 4095";
              }
              description "Multi-topology (MT) identifier";
            }
            uses neighbor-extended;
            description "List of neighbors.";
          }
          description
            "IS-IS multi-topology neighbor container - IS-IS
             reference is TLV 223.";
        }

        container mt-extended-ipv4-reachability {
          list prefixes {
            leaf mt-id {
```

```
            type uint16 {
              range "0 .. 4095";
            }
            description  "Multi-topology (MT) identifier";
          }
          uses prefix-ipv4-extended;
          description "List of extended prefixes.";
        }
        description
          "IPv4 multi-topolgy (MT) extended reachability
           information container - IS-IS reference is TLV 235.";
      }

      container mt-ipv6-reachability {
        list prefixes {
          leaf MT-ID {
            type uint16 {
              range "0 .. 4095";
            }
            description "Multi-topology (MT) identifier";
          }
          uses prefix-ipv6-extended;
          description "List of IPv6 extended prefixes.";
        }
        description
          "IPv6 multi-topolgy (MT) extended reachability
           information container - IS-IS reference is TLV 237.";
      }

      container ipv6-reachability {
        list prefixes {
          uses prefix-ipv6-extended;
          description "List of IPv6 prefixes.";
        }
        description
          "IPv6 reachability information container - IS-IS
           reference is TLV 236.";
      }
    }

    grouping lsdb {
      description "Link State Database (LSDB) grouping";
      container database {
        config false;
        list level-db {
          key level;

          leaf level {
```

```
            type level-number;
            description "LSDB level number (1 or 2)";
          }
          list lsp {
            key lsp-id;
            uses lsp-entry;
            description "List of LSPs in LSDB";
          }
          description "LSP list for LSDB level container";
        }
        description "IS-IS Link State database container";
      }
    }




    /* Augmentations */

    augment "/rt:routing/"
      +"rt:ribs/rt:rib/rt:routes/rt:route" {
      when "rt:source-protocol = 'isis:isis'" {
        description "IS-IS-specific route attributes.";
      }
      uses route-content;
      description
        "This augments route object in RIB with IS-IS-specific
         attributes.";
    }


    augment "/if:interfaces/if:interface" {
      leaf clns-mtu {
        type uint16;
        description "CLNS MTU of the interface";
      }
      description "ISO interface config.";
    }

    augment "/rt:routing/rt:control-plane-protocols/"
      +"rt:control-plane-protocol" {
      when "rt:type = 'isis:isis'" {
        description
          "This augment is only valid when routing protocol
           instance type is 'isis'";
      }
      description
        "This augments a routing protocol instance with IS-IS
```

```
          specific parameters.";
       container isis {
         must "count(area-address) > 0" {
           error-message
             "At least one area-address must be configured.";
           description
             "Enforce configuration of at least one area.";
         }

         uses isis-global-cfg;
         container fast-reroute {
           if-feature fast-reroute;
           uses fast-reroute-global-cfg;
           uses fast-reroute-global-state;
           description
             "IP Fast ReRoute (IPFRR) global container";
         }
         container spf-control {
           container ietf-spf-delay {
             if-feature ietf-spf-delay;
             uses ietf-spf-delay-cfg;
             uses ietf-spf-delay-state;
             description "IETF SPF delay algorithm container";
           }
           description
             "SPF computation-related information container";
         }
         container topologies {
           if-feature multi-topology;
           list topology {
             key "name";
             leaf enable {
               type boolean;
               description "Topology enable configuration";
             }
             leaf name {
               type leafref {
                 path "../../../../../../rt:ribs/rt:rib/rt:name";
               }
               description
                 "Routing Information Base (RIB) corresponding
                  to topology.";
             }

             uses isis-global-topologies-cfg;

             description "List of topologies";
           }
```

```
          description "Multi-topology container";
        }
        container interfaces {
          list interface {
            key "name";
            leaf name {
              type if:interface-ref;

              description
                "Reference to the interface within
                 the routing-instance.";
            }
            uses isis-if-cfg;
            container fast-reroute {
              if-feature fast-reroute;
              uses fast-reroute-if-cfg;
              description
                "IP Fast ReRoute (IPFRR) interface container";
            }
            container topologies {
              if-feature multi-topology;
              list topology {
                key name;

                leaf name {
                  type leafref {
                    path "../../../../../../../../"+
                      "rt:ribs/rt:rib/rt:name";
                  }

                  description
                    "Routing Information Base (RIB) corresponding
                     to topology.";
                }
                uses isis-if-topologies-cfg;
                description "List of interface topologies";
              }
              description "Multi-topology container";
            }
            uses adjacency-state;
            uses event-counters;
            uses packet-counters;
            description "List of IS-IS interfaces.";
          }
          description
            "IS-IS interface specific configuration container";
        }
        uses spf-log;
```

```
        uses lsp-log;
        uses hostname-db;
        uses lsdb;
        uses local-rib;
        uses system-counters;

        description
          "IS-IS configuration/state top-level container";
      }
    }


    /* RPC methods */

    rpc clear-adjacency {
      description
        "This RPC request clears a particular set of IS-IS
         adjacencies. If the operation fails due to an internal
         reason, then the error-tag and error-app-tag should be
         set indicating the reason for the failure.";
      input {

        leaf routing-protocol-instance-name {
          type instance-state-ref;
          mandatory "true";
          description
            "Name of the IS-IS protocol instance whose IS-IS
             information is being queried.

             If the corresponding IS-IS instance doesn't exist,
             then the operation will fail with an error-tag of
             'data-missing' and an error-app-tag of
             'routing-protocol-instance-not-found'.";
        }
        leaf level {
          type level;
          description
            "IS-IS level of the adjacency to be cleared. If the
             IS-IS level is level-1-2, both level 1 and level 2
             adjacencies would be cleared.

             If the value provided is different from the one
             authorized in the enum type, then the operation
             SHALL fail with an error-tag of 'data-missing' and
             an error-app-tag of 'bad-isis-level'.";
        }
        leaf interface {
          type string;
```

```
          description
            "IS-IS interface name.

             If the corresponding IS-IS interface doesn't exist,
             then the operation SHALL fail with an error-tag of
             'data-missing' and an error-app-tag of
             'isis-interface-not-found'.";
        }
      }
    }

    rpc clear-database {
      description
        "This RPC request clears a particular  IS-IS database. If
         the operation fails for an IS-IS internal reason, then
         the error-tag and error-app-tag should be set
         indicating the reason for the failure.";
      input {
        leaf routing-protocol-instance-name {
          type instance-state-ref;
          mandatory "true";
          description
            "Name of the IS-IS protocol instance whose IS-IS
             database(s) is/are being cleard.

             If the corresponding IS-IS instance doesn't exist,
             then the operation will fail with an error-tag of
             'data-missing' and an error-app-tag of
             'routing-protocol-instance-not-found'.";
        }
        leaf level {
          type level;
          description
            "IS-IS level of the adjacency to be cleared. If the
             IS-IS level is level-1-2, both level 1 and level 2
             databases would be cleared.

             If the value provided is different from the one
             authorized in the enum type, then the operation
             SHALL fail with an error-tag of 'data-missing' and
             an error-app-tag of 'bad-isis-level'.";
        }
      }
    }


    /* Notifications */
```

```
     notification database-overload {
       uses notification-instance-hdr;

       leaf overload {
         type enumeration {
           enum off {
             description
               "Indicates IS-IS instance has left overload state";
           }
           enum on {
             description
               "Indicates IS-IS instance has entered overload state";
           }

         }
         description "New overload state of the IS-IS instance";
       }
       description
         "This notification is sent when an IS-IS instance
          overload state changes.";
     }

     notification lsp-too-large {
       uses notification-instance-hdr;
       uses notification-interface-hdr;

       leaf pdu-size {
         type uint32;
         description "Size of the LSP PDU";
       }
       leaf lsp-id {
         type lsp-id;
         description "LSP ID";
       }
       description
         "This notification is sent when we attempt to propagate
          an LSP that is larger than the dataLinkBlockSize for the
          circuit.  The notification generation must be throttled
          with at least 5 seconds betweeen successive
          notifications.";
     }

     notification if-state-change {
       uses notification-instance-hdr;
       uses notification-interface-hdr;

       leaf state {
         type if-state-type;
```

```
        description "Interface state.";
      }
      description
        "This notification is sent when an interface
         state change is detected.";
    }

    notification corrupted-lsp-detected {
      uses notification-instance-hdr;
      leaf lsp-id {
        type lsp-id;
        description "LSP ID";
      }
      description
        "This notification is sent when we find that
         an LSP that was stored in memory has become
         corrupted.";
    }

    notification attempt-to-exceed-max-sequence {
      uses notification-instance-hdr;
      leaf lsp-id {
        type lsp-id;
        description "LSP ID";
      }
      description
        "This notification is sent when the system
         wraps the 32-bit sequence counter of an LSP.";
    }

    notification id-len-mismatch {
      uses notification-instance-hdr;
      uses notification-interface-hdr;

      leaf pdu-field-len {
        type uint8;
        description "Size of the ID length in the received PDU";
      }
      leaf raw-pdu {
        type binary;
        description "Received raw PDU.";
      }
      description
        "This notification is sent when we receive a PDU
         with a different value for the System ID length.
         The notification generation must be throttled
         with at least 5 seconds betweeen successive
         notifications.";
```

```
     }

     notification max-area-addresses-mismatch {
       uses notification-instance-hdr;
       uses notification-interface-hdr;

       leaf max-area-addresses {
         type uint8;
         description "Received number of supported areas";
       }
       leaf raw-pdu {
         type binary;
         description "Received raw PDU.";
       }
       description
         "This notification is sent when we receive a PDU
          with a different value for the Maximum Area Addresses.
          The notification generation must be throttled
          with at least 5 seconds betweeen successive
          notifications.";
     }

     notification own-lsp-purge {
       uses notification-instance-hdr;
       uses notification-interface-hdr;
       leaf lsp-id {
         type lsp-id;
         description "LSP ID";
       }
       description
         "This notification is sent when the system receives
          a PDU with its own system ID and zero age.";
     }

     notification sequence-number-skipped {
       uses notification-instance-hdr;
       uses notification-interface-hdr;
       leaf lsp-id {
         type lsp-id;
         description "LSP ID";
       }
       description
         "This notification is sent when the system receives a
          PDU with its own system ID and different contents. The
          system has to reoriginate the LSP with a higher sequence
          number.";
     }
```

```
     notification authentication-type-failure {
       uses notification-instance-hdr;
       uses notification-interface-hdr;
       leaf raw-pdu {
         type binary;
         description "Received raw PDU.";
       }
       description
         "This notification is sent when the system receives a
          PDU with the wrong authentication type field.
          The notification generation must be throttled
          with at least 5 seconds betweeen successive
          notifications.";
     }

     notification authentication-failure {
       uses notification-instance-hdr;
       uses notification-interface-hdr;
       leaf raw-pdu {
         type binary;
         description "Received raw PDU.";
       }
       description
         "This notification is sent when the system receives
          a PDU with the wrong authentication information.
          The notification generation must be throttled with
          with at least 5 seconds betweeen successive
          notifications.";
     }

     notification version-skew {
       uses notification-instance-hdr;
       uses notification-interface-hdr;
       leaf protocol-version {
         type uint8;
         description "Protocol version received in the PDU.";
       }
       leaf raw-pdu {
         type binary;
         description "Received raw PDU.";
       }
       description
         "This notification is sent when the system receives a
          PDU with a different protocol version number.
          The notification generation must be throttled
          with at least 5 seconds betweeen successive
          notifications.";
     }
```

```
     notification area-mismatch {
       uses notification-instance-hdr;
       uses notification-interface-hdr;
       leaf raw-pdu {
         type binary;
         description "Received raw PDU.";
       }
       description
         "This notification is sent when the system receives a
          Hello PDU from an IS that does not share any area
          address. The notification generation must be throttled
          with at least 5 seconds betweeen successive
          notifications.";
     }

     notification rejected-adjacency {
       uses notification-instance-hdr;
       uses notification-interface-hdr;
       leaf raw-pdu {
         type binary;
         description
           "Received raw PDU.";
       }
       leaf reason {
         type string;
         description
           "The system may provide a reason to reject the
            adjacency. If the reason is not available,
            an empty string will be returned.";
       }
       description
         "This notification is sent when the system receives a
          Hello PDU from an IS but does not establish an adjacency
          for some reason. The notification generation must be
          throttled with at least 5 seconds betweeen successive
          notifications.";
     }


     notification protocols-supported-mismatch {
       uses notification-instance-hdr;
       uses notification-interface-hdr;
       leaf raw-pdu {
         type binary;
         description "Received raw PDU.";
       }
       leaf-list protocols {
         type uint8;
```

```
        description
          "List of protocols supported by the remote system.";
      }
      description
        "This notification is sent when the system receives a
         non-pseudonode LSP that has no matching protocols
         supported. The notification generation must be throttled
         with at least 5 seconds betweeen successive
         notifications.";
    }

    notification lsp-error-detected {
      uses notification-instance-hdr;
      uses notification-interface-hdr;
      leaf lsp-id {
        type lsp-id;
        description "LSP ID.";
      }
      leaf raw-pdu {
        type binary;
        description "Received raw PDU.";
      }
      leaf error-offset {
        type uint32;
        description
          "If the problem is a malformed TLV, the error-offset
           points to the start of the TLV. If the problem is with
           the LSP header, the error-offset points to the errant
           byte";
      }
      leaf tlv-type {
        type uint8;
        description
          "If the problem is a malformed TLV, the tlv-type is set
           to the type value of the suspicious TLV. Otherwise,
           this leaf is not present.";
      }
      description
        "This notification is sent when the system  receives an
'        LSP with a parse error. The notification generation must
         be throttled with at least 5 seconds betweeen successive
         notifications.";
    }

    notification adjacency-state-change {
      uses notification-instance-hdr;
      uses notification-interface-hdr;
      leaf neighbor {
```

```
          type string;
          description
            "Name of the neighbor. If the name of the neighbor is
             not available, it is not returned.";
        }
        leaf neighbor-system-id {
          type system-id;
          description "Neighbor system-id";
        }
        leaf state {
          type adj-state-type;

          description "New state of the IS-IS adjacency.";
        }
        leaf reason {
          type string;
          description
            "If the adjacency is going to DOWN,  this leaf provides
             a reason for the adjacency going down. The reason is
             provided as a text. If the adjacency is going to UP, no
             reason is provided.";
        }
        description
          "This notification is sent when an IS-IS adjacency
           moves to Up state or to Down state.";
      }

      notification lsp-received {
        uses notification-instance-hdr;
        uses notification-interface-hdr;

        leaf lsp-id {
          type lsp-id;
          description "LSP ID";
        }
        leaf sequence {
          type uint32;
          description "Sequence number of the received LSP.";
        }
        leaf received-timestamp {
          type yang:timestamp;

          description "Timestamp when the LSP was received.";
        }
        leaf neighbor-system-id {
          type system-id;
          description "Neighbor system-id of LSP sender";
        }
```

```
      description
        "This notification is sent when an LSP is received.
         The notification generation must be throttled with at
         least 5 seconds betweeen successive notifications.";
     }

    notification lsp-generation {
      uses notification-instance-hdr;

      leaf lsp-id {
        type lsp-id;
        description "LSP ID";
      }
      leaf sequence {
        type uint32;
        description "Sequence number of the received LSP.";
      }
      leaf send-timestamp {
        type yang:timestamp;

        description "Timestamp when our LSP was regenerated.";
      }
      description
        "This notification is sent when an LSP is regenerated.
         The notification generation must be throttled with at
         least 5 seconds betweeen successive notifications.";
    }
  }
  <CODE ENDS>
```

7. 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. Access control to RPCs is also critical as RPC allows to clear protocol datastructures that would definitively impact the network service. This kind of RPC needs only to be used in specific cases by well-known experienced users.

Authors consider that all the configuration is considered as sensitive/vulnerable as well as RPCs. But security teams can decide

to open some part of the configuration to less experienced users depending on the internal organization, for example:

- o User FullWrite: would access to the whole data model. This kind of profile may be restricted to few experienced people.

- o User PartialWrite: would only access to configuration part within /isis/interfaces/interface. So this kind of profile is restricted to creation/modification/deletion of interfaces. This profile does not have access to RPC.

- o User Read: would only access to the operational states.

Unauthorized access to configuration or RPC may cause high damages to the network service.

The "isis/database" may contain authentication information. As presented in the description of the "/isis/database/level-1/lsp/authentication/authentication-key", the authentication MUST never be displayed in a plaintext format for security reason. Authors recommend the usage of MD5 to display or return the authentication-key.

Some authentication-key may also be required in the "isis" writable container. When configuring IS-IS using the NETCONF protocol, authors recommends the usage of secure transport of NETCONF using SSH ([RFC6242]).

8. Contributors

Authors would like to thank Kiran Agrahara Sreenivasa, Dean Bogdanovic, Yingzhen Qu, Yi Yang for their major contributions to the draft.

9. Acknowledgements

TBD.

10. 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
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
namespace: urn:ietf:params:xml:ns:yang:ietf-isis
prefix: isis
reference: RFC XXXX
```

11. Change log for ietf-isis YANG module

11.1. From version -19 to version -20

- o Moved to Yang 1.1
- o Lower case enumerations
- o Add RFC references to features
- o Remove segment-routing feature
- o Modified BFD activation modeling

11.2. From version -18 to version -19

- o Align with draft-ietf-netmod-rfc8022bis.
- o Modify address family types as per draft-ietf-rtgwg-routing-types-17.

11.3. From version -17 to version -18

- o NMDA compliancy.
- o Set some default values.
- o Align with iana-rt-types module.

11.4. From version -16 to version -17

- o Cosmetic fixes.
- o Use of rt-types model.

11.5. From version -15 to version -16

- o Alignment with last IETF key chain model.
- o lsp-log "change" leaf moved as an identity.

- o Incremental SPF removed from spf-log types.

11.6. From version -14 to version -15

- o Alignment with OSPF model done:
  - * Added spf-control container with IETF SPF delay algorithm as a feature.
  - * Added graceful-restart options.
  - * Added nsr as a feature.
  - * Removed per topology FRR. Need to be augmented if necessary.
  - * Created an ldp container within mpls.
  - * Renamed igp-ldp-sync to igp-sync.
  - * Added auto-cost container.
  - * Moved reference-bandwidth under auto-cost container.
  - * Added IS-IS local RIB as operational state.
  - * Added decode-completed and raw-data leaves in the LSDB model.
  - * Modified the notification header.

11.7. From version -13 to version -14

- o Segment Routing extensions are now in a separate document.

11.8. From version -12 to version -13

- o Move feature nlpid-control to container rather than list.
- o Rename multi-topology to topologies to align with OSPF.
- o Rename bfd/enabled to bfd/enable for consistency reason.
- o Add support for NSR with a feature.

11.9. From version -09 to version -12

- o Rename node-tag container to node-tags.

11.10. From version -08 to version -09

- o Added container before af list.
- o Added container before topology list.
- o Aligned LFA if per level cfg.
- o Align to draft-ietf-netmod-routing-cfg-23.

11.11. From version -07 to version -08

- o Remove selector from system-id type.
- o Add some default values.
- o Moved lists to containers+groupings for per level configuration.
- o remove routing-instance as per core routing model v21.
- o added BFD leaf (no more BFD protocol model).
- o changed keychain module reference.

11.12. From version -05 to version -07

- o Move Overload config from list to container.
- o Move Overload-max-metric config from list to container.
- o Move preference config from list to container.
- o Add Node flag in config.
- o Removed BFD config => moved to isis-bfd module.
- o Remove call to routing policy model.

11.13. From version -03 to version -05

- o Correct invalid references to previous versions of core routing model.
- o Remove BFD config and replace by groupings from ietf-bfd.
- o Adding routing-policy support through routing-policy model.

11.14. From version -02 to version -03

- o Reviewed config and op state groupings.
- o Add default value to lfa candidate-disabled.
- o Add enable leaf to isis container to reflect admin state.
- o Move to VRF centric only.
- o Segment routing is part os a separate module.

11.15. From version -01 to version -02

- o Adding IPFRR.
- o Adding igp-ldp-sync.
- o Adding segment-routing.
- o Adding instance reference to operational states.
- o Move AF type from string to identity.
- o Updated router-capability in LSDB description.
- o packet counters moved to interface-packet-counters.
- o Added modification information in lsp-log.
- o Removing igp-ldp-sync timer in IS-IS.
- o Defining hierarchy for operational states.
- o Adding clns-mtu.
- o Adding key-chain.

11.16. From version -00 to version -01

- o Interface metric move from af container to interface container.
- o Hello-padding on interface moved to hello-padding-disable with empty type.
- o three-way-handshake removed.
- o route preference changed to a choice.

- o csnp-authentication/psnp-authentication merged to authentication container.
- o lsp-gen-interval-exp-delay removed.
- o Added overload-max-metric feature.
- o overload-max-metric is in a separate container.
- o Change hello-padding to container.
- o Change bfd to container.
- o Make BFD a feature.
- o Create mpls-te container and put router-id inside.
- o Remove GR helper disable and timers.

12. Normative References

[I-D.ietf-netmod-routing-cfg]
Lhotka, L. and A. Lindem, "A YANG Data Model for Routing Management", draft-ietf-netmod-routing-cfg-25 (work in progress), November 2016.

[I-D.ietf-rtgwg-yang-key-chain]
Lindem, A., Qu, Y., Yeung, D., Chen, I., and Z. Zhang, "Routing Key Chain YANG Data Model", draft-ietf-rtgwg-yang-key-chain-24 (work in progress), April 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>.

[RFC5120] Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, DOI 10.17487/RFC5120, February 2008, <https://www.rfc-editor.org/info/rfc5120>.

[RFC5286] Atlas, A., Ed. and A. Zinin, Ed., "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, DOI 10.17487/RFC5286, September 2008, <https://www.rfc-editor.org/info/rfc5286>.

[RFC5443] Jork, M., Atlas, A., and L. Fang, "LDP IGP Synchronization", RFC 5443, DOI 10.17487/RFC5443, March 2009, <https://www.rfc-editor.org/info/rfc5443>.

[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>.

[RFC7490] Bryant, S., Filsfils, C., Previdi, S., Shand, M., and N. So, "Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)", RFC 7490, DOI 10.17487/RFC7490, April 2015, <https://www.rfc-editor.org/info/rfc7490>.

Appendix A. Example of IS-IS configuration in XML

This section gives an example of configuration of an IS-IS instance on a device. The example is written in XML.

```
<?xml version="1.0" encoding="utf-8"?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <routing xmlns="urn:ietf:params:xml:ns:yang:ietf-routing">
      <name>SLI</name>
      <router-id>1.1.1.1</router-id>
      <description/>
      <interfaces>
        <interface>
          <name>Loopback0</name>
        </interface>
```

```
                  <interface>
            <name>Eth1</name>
          </interface>
        </interfaces>
        <control-plane-protocols>
          <control-plane-protocol>
            <name>ISIS</name>
            <description/>
            <type>isis:isis</type>
            <isis xmlns="urn:ietf:params:xml:ns:yang:ietf-isis">
                            <enable>true</enable>
                            <level-type>level-2</level-type>
                <system-id>87FC.FCDF.4432</system-id>
                <area-address>49.0001</area-address>
                <mpls-te>
                  <ipv4-router-id>1.1.1.1</ipv4-router-id>
                </mpls-te>
                <lsp-lifetime>65535</lsp-lifetime>
                <lsp-refresh>65000</lsp-refresh>
                <metric-type>
                  <value>wide</value>
                </metric-type>
                <default-metric>
                  <value>111111</value>
                </default-metric>
                <afs>
                                  <af>
                                          <af>ipv4-unicast</af>
                                          <enabled>true</enabled>
                                  </af>
                </afs>
                <interfaces>
                  <interface>
                    <name>Loopback0</name>
                    <tag>200</tag>
                    <metric>
                      <value>0</value>
                    </metric>
                                    <passive>true</passive>
                  </interface>
                                  <interface>
                    <name>Eth1</name>
                                    <level-type>level-2</level-type>
                                    <interface-type>point-to-point
                                    </interface-type>
                    <metric>
                      <value>167890</value>
                    </metric>
```

```
                  </interface>
                </interfaces>
            </isis>
          </control-plane-protocol>
        </control-plane-protocols>
    </routing>
    <interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
      <interface>
        <name>Loopback0</name>
        <description/>
        <type/>
        <link-up-down-trap-enable/>
        <ipv4 xmlns="urn:ietf:params:xml:ns:yang:ietf-ip">
          <mtu/>
          <address>
            <ip>1.1.1.1</ip>
            <prefix-length>32</prefix-length>
          </address>
        </ipv4>

      </interface>
          <interface>
        <name>Eth1</name>
        <description/>
        <type/>
        <link-up-down-trap-enable/>
        <ipv4 xmlns="urn:ietf:params:xml:ns:yang:ietf-ip">
          <mtu/>
          <address>
            <ip>10.0.0.1</ip>
            <prefix-length>30</prefix-length>
          </address>
        </ipv4>

      </interface>
    </interfaces>
  </data>
```


Authors' Addresses

Stephane Litkowski
Orange

Email: stephane.litkowski@orange.com

Derek Yeung
Arrcus, Inc

Email: derek@arrcus.com

Acee Lindem
Cisco Systems

Email: acee@cisco.com

Jeffrey Zhang
Juniper Networks

Email: zzhang@juniper.net

Ladislav Lhotka
CZ.NIC

Email: lhotka@nic.cz