Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: July 8, 2017

A. Clemm
Huawei
J. Medved
Cisco
R. Varga
Pantheon Technologies SRO
X. Liu
Ericsson
H. Ananthakrishnan
Packet Design
N. Bahadur
Bracket Computing

January 4, 2017

A YANG Data Model for Layer 3 Topologies
draft-ietf-i2rs-yang-l3-topology-08.txt

Abstract

This document defines a YANG data model for layer 3 network topologies.

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 July 8, 2017.

Table of Contents

1. Introduction

This document introduces a YANG [RFC7950] [RFC6991] data model for Layer 3 network topologies, specifically Layer 3 Unicast. The model allows an application to have a holistic view of the topology of a Layer 3 network, all contained in a single conceptual YANG datastore. The data model builds on top of, and augments, the data model for network topologies defined in [I-D.draft-ietf-i2rs-yang-network-topo].

This document also shows how the model can be further refined to cover different Layer 3 Unicast topology types. For this purpose, example models are introduced that cover IS-IS [RFC1195] and OSPF [RFC2328]. Those examples are intended purely for illustrative purposes; we expect that full-blown IS-IS and OSPF models will be more comprehensive and refined than the examples shown here.

There are multiple applications for a topology data model. A number of use cases have been defined in section 6 of [I-D.draft-ietf-i2rs-usecase-reqs-summary]. For example, nodes within the network can use the data model to capture their understanding of the overall network topology and expose it to a network controller. A network controller can then use the instantiated topology data to compare and reconcile its own view of the network topology with that of the network elements that it controls. Alternatively, nodes within the network could propagate this understanding to compare and reconcile this understanding either amongst themselves or with help of a controller. Beyond the network element itself, a network controller might even use the data model to represent its view of the topology that it controls and expose it to applications north of itself.

The data model for Layer 3 Unicast topologies defined in this document is specified in a YANG module "ietf-l3-unicast-topology". To do so, it augments general network topology model defined in [I-D.draft-ietf-i2rs-yang-network-topo] with information specific to Layer 3 Unicast. This way, the general topology model is extended to be able to meet the needs of Layer 3 Unicast topologies.

Information that is kept in the Traffic Engineering Database (TED) will be specified in a separate model [I-D.draft-ietf-teas-yang-te-topo] and outside the scope of this specification.

2. Definitions and Acronyms

As this document defines a YANG data model, in this document many terms are used that have been defined in conjunction with YANG [RFC7950] and Netconf [RFC6241]. Some terms, such as datastore and data tree, are repeated here for clarity and to put them in context.

Datastore: A conceptual place to store and access information, such as instantiated YANG data.

Data tree: An instantiated tree of data modeled with YANG, in which individual data items are represented by data nodes which are arranged in hierarchical manner.

Data subtree: An instantiated data node and the data nodes that are hierarchically contained within it.

HTTP: Hyper-Text Transfer Protocol

IGP: Interior Gateway Protocol

IS-IS: Intermediate System to Intermediate System protocol

LSP: Label Switched Path

NETCONF: Network Configuration Protocol

OSPF: Open Shortest Path First, a link state routing protocol

URI: Uniform Resource Identifier

ReST: Representational State Transfer, a style of stateless interface and protocol that is generally carried over HTTP

SRLG: Shared Risk Link Group

TED: Traffic Engineering Database

YANG: A data definition language for NETCONF

3. Model Structure

The Layer 3 Unicast topology model is defined by YANG module "l3-unicast-topology". The relationship of this module with other YANG modules is roughly depicted in the figure below.

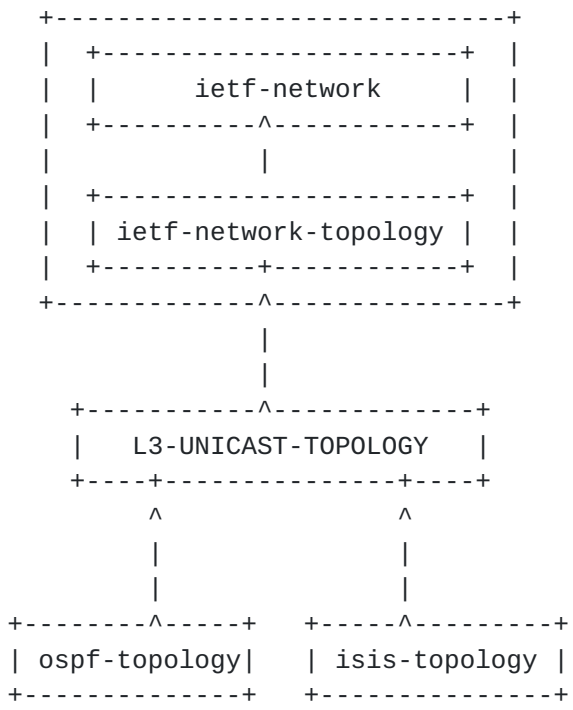

Figure 1: Overall model structure

YANG modules "ietf-network" and "ietf-network-topology" collectively define the basic network topology model. YANG module "ietf-l3-unicast-topology" augments those models with additional definitions needed to represent Layer 3 Unicast topologies. This module in turn can be augmented by YANG modules with additional definitions for specific types of Layer 3 Unicast topologies, such as OSPF and for IS-IS topologies.

4. Layer 3 Unicast Topology Model Overview

The Layer 3 Unicast topology model is defined by YANG module "ietf-l3-unicast-topology" and depicted in the following diagram. Brackets enclose list keys, "rw" means configuration, "ro" operational state data, "?" designates optional nodes, "*" designates nodes that can have multiple instances. Parantheses enclose choice and case nodes. The prefix "nd:" refers to the YANG module for networks; the prefix "lnk:" refers to the YANG module for network topology. In the interest of brevity, notifications are not depicted.

```
module: ietf-l3-unicast-topology
augment /nd:networks/nd:network/nd:network-types:
   +--rw l3-unicast-topology!
augment /nd:networks/nd:network:
   +--rw l3-topology-attributes
      +--rw name?   string
      +--rw flag*   l3-flag-type
augment /nd:networks/nd:network/nd:node:
   +--rw l3-node-attributes
      +--rw name?        inet:domain-name
      +--rw flag*        node-flag-type
      +--rw router-id*   inet:ip-address
      +--rw prefix* [prefix]
         +--rw prefix    inet:ip-prefix
         +--rw metric?   uint32
         +--rw flag*     prefix-flag-type
augment /nd:networks/nd:network/lnk:link:
   +--rw l3-link-attributes
      +--rw name?     string
      +--rw flag*     link-flag-type
      +--rw metric?   uint32
augment /nd:networks/nd:network/nd:node/lnk:termination-point:
   +--rw l3-termination-point-attributes
      +--rw (termination-point-type)?
         +--:(ip)
         |  +--rw ip-address*      inet:ip-address
         +--:(unnumbered)
         |  +--rw unnumbered-id?   uint32
         +--:(interface-name)
            +--ro interface-name?  string
```

The module augments the original ietf-network and ietf-network-topology modules as follows:

- A new network topology type is introduced, l3-unicast-topology. The corresponding container augments the network-types of the ietf-network module.

- Additional topology attributes are introduced, defined in a grouping, which augments the "network" list of the network module. The attributes include a name for the topology, as well as a set of flags (represented through a leaf-list). Each type of flag is represented by a separate identity. This allows to introduce additional flags in augmenting modules using additional identities without needing to revise this module.

- Additional data objects for nodes are introduced by augmenting the "node" list of the network module. New objects include again a

set of flags, as well as a list of prefixes. Each prefix in turn includes an ip prefix, a metric, and a prefix-specific set of flags.

- Links (in the ietf-network-topology module) are augmented with a set of parameters as well, allowing to associate a link with a link name, another set of flags, and a link metric.

- Termination points (in the ietf-network-topology module as well) are augmented with a choice of IP address, identifier, or name.

In addition, the module defines a set of notifications to alert clients of any events concerning links, nodes, prefixes, and termination points. Each notification includes an indication of the type of event, the topology from which it originated, and the affected node, or link, or prefix, or termination point. In addition, as a convenience to applications, additional data of the affected node, or link, or termination point (respectively) is included. While this makes notifications larger in volume than they would need to be, it avoids the need for subsequent retrieval of context information, which also might have changed in the meantime.

5. Layer 3 Unicast Topology YANG Module

```
<CODE BEGINS> file "ietf-l3-unicast-topology@2017-01-04.yang"
module ietf-l3-unicast-topology {
  yang-version 1.1;
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-l3-unicast-topology";
  prefix "l3t";
  import ietf-network {
    prefix "nd";
  }
  import ietf-network-topology {
    prefix "lnk";
  }
  import ietf-inet-types {
    prefix "inet";
  }
  organization
    "IETF I2RS (Interface to the Routing System) Working Group";
  contact
    "WG Web:    <http://tools.ietf.org/wg/i2rs/>
     WG List:   <mailto:i2rs@ietf.org>
     WG Chair:  Susan Hares
                <mailto:shares@ndzh.com>
     WG Chair:  Russ White
                <mailto:russ@riw.us>
```

```
       Editor:    Alexander Clemm
                  <mailto:ludwig@clemm.org>
       Editor:    Jan Medved
                  <mailto:jmedved@cisco.com>
       Editor:    Robert Varga
                  <mailto:robert.varga@pantheon.sk>
       Editor:    Xufeng Liu
                  <mailto:xliu@kuatrotech.com>
       Editor:    Nitin Bahadur
                  <mailto:nitin_bahadur@yahoo.com>
       Editor:    Hariharan Ananthakrishnan
                  <mailto:hari@packetdesign.com>";
     description
       "This module defines a model for Layer 3 Unicast
        topologies.
        Copyright (c) 2017 IETF Trust and the persons identified as
        authors of the code.  All rights reserved.
        Redistribution and use in source and binary forms, with or
        without modification, is permitted pursuant to, and subject
        to the license terms contained in, the Simplified BSD License
        set forth in Section 4.c of the IETF Trust's Legal Provisions
        Relating to IETF Documents
        (http://trustee.ietf.org/license-info).
        This version of this YANG module is part of
        draft-ietf-i2rs-yang-l3-topology-08;
        see the RFC itself for full legal notices.
        NOTE TO RFC EDITOR: Please replace above reference to
        draft-ietf-i2rs-yang-l3-topology-08 with RFC
        number when published (i.e. RFC xxxx).";
     revision "2017-01-04" {
       description
         "Initial revision.
          NOTE TO RFC EDITOR: Please replace the following reference
          to draft-ietf-i2rs-yang-l3-topology-08 with
          RFC number when published (i.e. RFC xxxx).";
       reference
         "draft-ietf-i2rs-yang-l3-topology-08";
     }

     identity flag-identity {
       description "Base type for flags";
     }

     typedef l3-event-type {
       type enumeration {
         enum "add" {
           description
             "An Layer 3 node or link or prefix or termination-point has
```

```
            been added";
        }
        enum "remove" {
          description
            "An Layer 3 node or link or prefix or termination-point has
            been removed";
        }
        enum "update" {
          description
            "An Layer 3 node or link or prefix or termination-point has
            been updated";
        }
      }
      description "Layer 3 Event type for notifications";
    }

    typedef prefix-flag-type {
      type identityref {
        base "flag-identity";
      }
      description "Prefix flag attributes";
    }

    typedef node-flag-type {
      type identityref {
        base "flag-identity";
      }
      description "Node flag attributes";
    }

    typedef link-flag-type {
      type identityref {
        base "flag-identity";
      }
      description "Link flag attributes";
    }

    typedef l3-flag-type {
      type identityref {
        base "flag-identity";
      }
      description "L3 flag attributes";
    }

    grouping l3-prefix-attributes {
      description
        "L3 prefix attributes";
      leaf prefix {
```

```
        type inet:ip-prefix;
        description
          "IP prefix value";
      }
      leaf metric {
        type uint32;
        description
          "Prefix metric";
      }
      leaf-list flag {
        type prefix-flag-type;
        description
          "Prefix flags";
      }
    }
    grouping l3-unicast-topology-type {
      description "Identify the topology type to be L3 unicast.";
      container l3-unicast-topology {
        presence "indicates L3 Unicast Topology";
        description
          "The presence of the container node indicates L3 Unicast
          Topology";
      }
    }
    grouping l3-topology-attributes {
      description "Topology scope attributes";
      container l3-topology-attributes {
        description "Containing topology attributes";
        leaf name {
          type string;
          description
            "Name of the topology";
        }
        leaf-list flag {
          type l3-flag-type;
          description
            "Topology flags";
        }
      }
    }
    grouping l3-node-attributes {
      description "L3 node scope attributes";
      container l3-node-attributes {
        description
          "Containing node attributes";
        leaf name {
          type inet:domain-name;
          description
```

```
            "Node name";
        }
        leaf-list flag {
          type node-flag-type;
          description
            "Node flags";
        }
        leaf-list router-id {
          type inet:ip-address;
          description
            "Router-id for the node";
        }
        list prefix {
          key "prefix";
          description
            "A list of prefixes along with their attributes";
          uses l3-prefix-attributes;
        }
      }
    }
    grouping l3-link-attributes {
      description
        "L3 link scope attributes";
      container l3-link-attributes {
        description
          "Containing link attributes";
        leaf name {
          type string;
          description
            "Link Name";
        }
        leaf-list flag {
          type link-flag-type;
          description
            "Link flags";
        }
        leaf metric {
          type uint32;
          description
              "Link Metric";
        }
      }
    }
    grouping l3-termination-point-attributes {
      description "L3 termination point scope attributes";
      container l3-termination-point-attributes {
        description
          "Containing termination point attributes";
```

```
        choice termination-point-type {
          description
            "Indicates the termination point type";
          case ip {
            leaf-list ip-address {
              type inet:ip-address;
              description
                "IPv4 or IPv6 address.";
            }
          }
          case unnumbered {
            leaf unnumbered-id {
              type uint32;
              description
                "Unnumbered interface identifier.
                 The identifier will correspond to the ifIndex value
                 of the interface, i.e. the ifIndex value of the
                 ifEntry that represents the interface in
                 implementations where the Interfaces Group MIB
                 (RFC 2863) is supported.";
              reference
                "RFC 2863: The Interfaces Group MIB";
            }
          }
          case interface-name {
            leaf interface-name {
              type string;
              description
                "A name of the interface.  The name can (but does not
                 have to) correspond to an interface reference of a
                 containing node's interface, i.e. the path name of a
                 corresponding interface data node on the containing
                 node reminiscent of data type if-ref defined in
                 RFC 7223. It should be noted that data type if-ref of
                 RFC 7223 cannot be used directly, as this data type
                 is used to reference an interface in a datastore of
                 a single node in the network, not to uniquely
                 reference interfaces across a network.";
            }
          }
        }
      }
    }
    augment "/nd:networks/nd:network/nd:network-types" {
      description
        "Introduce new network type for L3 unicast topology";
      uses l3-unicast-topology-type;
    }
```

```
    augment "/nd:networks/nd:network" {
      when "nd:network-types/l3-unicast-topology" {
        description
          "Augmentation parameters apply only for networks with
          L3 unicast topology";
      }
      description
          "L3 unicast for the network as a whole";
      uses l3-topology-attributes;
    }
    augment "/nd:networks/nd:network/nd:node" {
      when "../nd:network-types/l3-unicast-topology" {
        description
          "Augmentation parameters apply only for networks with
          L3 unicast topology";
      }
      description
          "L3 unicast node level attributes ";
      uses l3-node-attributes;
    }
    augment "/nd:networks/nd:network/lnk:link" {
      when "../nd:network-types/l3-unicast-topology" {
        description
          "Augmentation parameters apply only for networks with
          L3 unicast topology";
      }
      description
        "Augment topology link attributes";
      uses l3-link-attributes;
    }
    augment "/nd:networks/nd:network/nd:node/"
           +"lnk:termination-point" {
      when "../../nd:network-types/l3-unicast-topology" {
        description
          "Augmentation parameters apply only for networks with
          L3 unicast topology";
      }
      description "Augment topology termination point configuration";
      uses l3-termination-point-attributes;
    }
    notification l3-node-event {
      description
        "Notification event for L3 node";
      leaf l3-event-type {
        type l3-event-type;
        description
          "Event type";
      }
```

```
      uses nd:node-ref;
      uses l3-unicast-topology-type;
      uses l3-node-attributes;
    }
    notification l3-link-event {
      description
        "Notification event for L3 link";
      leaf l3-event-type {
        type l3-event-type;
        description
          "Event type";
      }
      uses lnk:link-ref;
      uses l3-unicast-topology-type;
      uses l3-link-attributes;
    }
    notification l3-prefix-event {
      description
        "Notification event for L3 prefix";
      leaf l3-event-type {
        type l3-event-type;
        description
          "Event type";
      }
      uses nd:node-ref;
      uses l3-unicast-topology-type;
      container prefix {
        description
          "Containing L3 prefix attributes";
        uses l3-prefix-attributes;
      }
    }
    notification termination-point-event {
      description
        "Notification event for L3 termination point";
      leaf l3-event-type {
        type l3-event-type;
        description
          "Event type";
      }
      uses lnk:tp-ref;
      uses l3-unicast-topology-type;
      uses l3-termination-point-attributes;
    }
  }

  <CODE ENDS>
```

6. Extending the Model

The model can be extended for specific Layer 3 Unicast types. Examples include OSPF and IS-IS topologies. In the following, two additional YANG modules are introduced that define simple topology models for OSPF and IS-IS, respectively. These modules intended to serve as examples that illustrate how the general topology model can be refined across multiple levels; they do not constitute full-fledged OSPF and IS-IS topology models which may be more comprehensive and refined than the models that are described here.

6.1. Example 1: OSPF Topology

6.1.1. Model Overview

The following model shows how the Layer 3 Unicast topology model can be extended to cover OSFP topologies. For this purpose, a set of augmentations are introduced in a separate YANG module, "example-ietf-ospf-topology", whose structure is depicted in the following diagram. Like before, brackets enclose list keys, "rw" means configuration, "ro" operational state data, "?" designates optional nodes, "*" designates nodes that can have multiple instances. Parantheses enclose choice and case nodes. A "+" at the end of a line indicates a line break.

```
   module: example-ietf-ospf-topology
   augment /nd:networks/nd:network/nd:network-types/+
      l3t:l3-unicast-topology:
      +--rw ospf!
   augment /nd:networks/nd:network/l3t:l3-topology-attributes:
      +--rw ospf-topology-attributes
         +--rw area-id?   area-id-type
   augment /nd:networks/nd:network/nd:node/l3t:l3-node-attributes:
      +--rw ospf-node-attributes
         +--rw (router-type)?
         |  +--:(abr)
         |  |  +--rw abr?                empty
         |  +--:(asbr)
         |  |  +--rw asbr?               empty
         |  +--:(internal)
         |  |  +--rw internal?           empty
         |  +--:(pseudonode)
         |     +--rw pseudonode?         empty
         +--rw dr-interface-id?     uint32
         +--rw multi-topology-id*   uint8
   augment /nd:networks/nd:network/lnk:link/l3t:l3-link-attributes:
      +--rw ospf-link-attributes
         +--rw multi-topology-id?   uint8
   augment /l3t:l3-node-event:
      +---- ospf!
      +---- ospf-node-attributes
         +---- (router-type)?
         |  +--:(abr)
         |  |  +---- abr?                empty
         |  +--:(asbr)
         |  |  +---- asbr?               empty
         |  +--:(internal)
         |  |  +---- internal?           empty
         |  +--:(pseudonode)
         |     +---- pseudonode?         empty
         +---- dr-interface-id?     uint32
         +---- multi-topology-id*   uint8
   augment /l3t:l3-link-event:
      +---- ospf!
      +---- ospf-link-attributes
         +---- multi-topology-id?   uint8
```

The module augments "ietf-l3-unicast-topology" as follows:

o A new topology type for an OSPF topology is introduced.

- Additional topology attributes are defined in a new grouping which augments l3-topology-attributes of the ietf-l3-unicast-topology module. The attributes include an OSPF area-id identifying the OSPF area.

- Additional data objects for nodes are introduced by augmenting the l3-node-attributes of the l3-unicast-topology module. New objects include router-type, dr-interface-id for pseudonodes, list of multi-topology-ids, ospf node capabilities, and traffic engineering attributes.

- Links are augmented with a multi-topology-id and traffic engineering link attributes.

- Prefixes are augmented with OSPF specific forwarding address.

In addition, the module extends notifications for events concerning Layer 3 nodes, links, termination points, and prefixes with OSPF attributes.

It should be noted that the model defined here represents topology and is intended as an example. It does not define how to configure OSPF routers or interfaces.

6.1.2. OSPF Topology YANG Module

The OSPF Topology YANG Module is specified below. As mentioned, the module is intended as an example for how the Layer 3 Unicast topology model can be extended to cover OSFP topologies, but it is not normative. Accordingly, the module is not delimited with CODE BEGINS and CODE ENDS tags.

```
file "example-ietf-ospf-topology@2017-01-04.yang"
module example-ietf-ospf-topology {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:example-ietf-ospf-topology";
  prefix "ospft";
  import ietf-yang-types {
    prefix "yang";
  }
  import ietf-network {
    prefix "nd";
  }
  import ietf-network-topology {
    prefix "lnk";
  }
  import ietf-l3-unicast-topology {
    prefix "l3t";
```

```
    }
     organization
      "IETF I2RS (Interface to the Routing System) Working Group";
    contact
      "WG Web:    <http://tools.ietf.org/wg/i2rs/>
       WG List:   <mailto:i2rs@ietf.org>
       WG Chair:  Susan Hares
                  <mailto:shares@ndzh.com>
       WG Chair:  Russ White
                  <mailto:russ@riw.us>
       Editor:    Alexander Clemm
                  <mailto:ludwig@clemm.org>
       Editor:    Jan Medved
                  <mailto:jmedved@cisco.com>
       Editor:    Robert Varga
                  <mailto:robert.varga@pantheon.sk>
       Editor:    Xufeng Liu
                  <mailto:xliu@kuatrotech.com>
       Editor:    Nitin Bahadur
                  <mailto:nitin_bahadur@yahoo.com>
       Editor:    Hariharan Ananthakrishnan
                  <mailto:hari@packetdesign.com>";
    description
      "This module defines a model for OSPF network topologies.
       Copyright (c) 2017 IETF Trust and the persons identified as
       authors of the code.  All rights reserved.
       Redistribution and use in source and binary forms, with or
       without modification, is permitted pursuant to, and subject
       to the license terms contained in, the Simplified BSD License
       set forth in Section 4.c of the IETF Trust's Legal Provisions
       Relating to IETF Documents
       (http://trustee.ietf.org/license-info).
       This version of this YANG module is part of
       draft-ietf-i2rs-yang-l3-topology-08;
       see the RFC itself for full legal notices.
       NOTE TO RFC EDITOR: Please replace above reference to
       draft-ietf-i2rs-yang-l3-topology-08 with RFC
       number when published (i.e. RFC xxxx).";
     revision "2017-01-04" {
      description
        "Initial revision.
         NOTE TO RFC EDITOR: Please replace the following reference
         to draft-ietf-i2rs-yang-l3-topology-08 with
         RFC number when published (i.e. RFC xxxx).";
      reference
        "draft-ietf-i2rs-yang-l3-topology-08";
    }
    typedef area-id-type {
```

```
        type yang:dotted-quad;
        description
          "Area ID type.";
      }
      grouping ospf-topology-type {
        description
          "Identifies the OSPF topology type.";
        container ospf {
          presence "indiates OSPF Topology";
          description
            "Its presence identifies the OSPF topology type.";
        }
      }
      augment "/nd:networks/nd:network/nd:network-types/"
        + "l3t:l3-unicast-topology" {
        description
          "Defines the OSPF topology type.";
        uses ospf-topology-type;
      }
      augment "/nd:networks/nd:network/l3t:l3-topology-attributes" {
        when "../nd:network-types/l3t:l3-unicast-topology/ospf" {
          description
            "Augment only for OSPF topology";
        }
        description
          "Augment topology configuration";
        container ospf-topology-attributes {
          description
            "Containing topology attributes";
          leaf area-id {
            type area-id-type;
            description
              "OSPF area ID";
          }
        }
      }
      augment "/nd:networks/nd:network/nd:node/l3t:l3-node-attributes" {
        when "../../nd:network-types/l3t:l3-unicast-topology/ospf" {
          description
            "Augment only for OSPF topology";
        }
        description
          "Augment node configuration";
        uses ospf-node-attributes;
      }
      augment "/nd:networks/nd:network/lnk:link/l3t:l3-link-attributes" {
        when "../../nd:network-types/l3t:l3-unicast-topology/ospf" {
          description
```

```
          "Augment only for OSPF topology";
      }
      description
        "Augment link configuration";
      uses ospf-link-attributes;
    }
    grouping ospf-node-attributes {
      description
        "OSPF node scope attributes";
      container ospf-node-attributes {
        description
          "Containing node attributes";
        choice router-type {
          description
            "Indicates router type";
          case abr {
            leaf abr {
              type empty;
              description
                "The node is ABR";
            }
          }
          case asbr {
            leaf asbr {
              type empty;
              description
                "The node is ASBR";
            }
          }
          case internal {
            leaf internal {
              type empty;
              description
                "The node is internal";
            }
          }
          case pseudonode {
            leaf pseudonode {
              type empty;
              description
                "The node is pseudonode";
            }
          }
        }
        leaf dr-interface-id {
          when "../pseudonode" {
            description
              "Valid only for pseudonode";
```

```
          }
          type uint32;
          default "0";
          description
            "For pseudonodes, DR interface-id";
        }
        leaf-list multi-topology-id {
          type uint8 {
            range "0..127";
          }
          max-elements "128";
          description
            "List of Multi-Topology Identifier up-to 128 (0-127).
             See RFC 4915";
        }
      }
    }
    grouping ospf-link-attributes {
      description
        "OSPF link scope attributes";
      container ospf-link-attributes {
        description
          "Containing OSPF link attributes";
        leaf multi-topology-id {
          type uint8 {
            range "0..127";
          }
          description "Multi topology ID";
        }
      }
    } // ospf-link-attributes
    augment "/l3t:l3-node-event" {
      description
        "OSPF node event";
      uses ospf-topology-type;
      uses ospft:ospf-node-attributes;
    }
    augment "/l3t:l3-link-event" {
      description
        "OSPF link event";
      uses ospf-topology-type;
      uses ospft:ospf-link-attributes;
    }
  }
```

6.2. Example 2: IS-IS Topology

6.2.1. Model Overview

IS-IS topologies are another type of Layer 3 Unicast topology. Like in the case of OSPF topology, a model for IS-IS topology can be defined in a separate module which augments "ietf-l3-unicast-igp-topology". The structure of a corresponding model, "ietf-isis-topology", is depicted in the following diagram. Like before, brackets enclose list keys, "rw" means configuration, "ro" operational state data, "?" designates optional nodes, "*" designates nodes that can have multiple instances. Parantheses enclose choice and case nodes. A "+" at the end of a line indicates a line break.

```
module: example-ietf-isis-topology
augment /nd:networks/nd:network/nd:network-types/+
   l3t:l3-unicast-topology:
   +--rw isis!
augment /nd:networks/nd:network/l3t:l3-topology-attributes:
   +--rw isis-topology-attributes
      +--rw net?   area-address
augment /nd:networks/nd:network/nd:node/l3t:l3-node-attributes:
   +--rw isis-node-attributes
      +--rw iso
      |  +--rw iso-system-id?       system-id
      |  +--rw iso-pseudonode-id?   iso-pseudonode-id
      +--rw net*                 area-address
      +--rw multi-topology-id*   uint16
      +--rw level?               level
augment /nd:networks/nd:network/lnk:link/l3t:l3-link-attributes:
   +--rw isis-link-attributes
      +--rw multi-topology-id?   uint16
augment /l3t:l3-node-event:
   +---- isis!
   +---- isis-node-attributes
      +---- iso
      |  +---- iso-system-id?       system-id
      |  +---- iso-pseudonode-id?   iso-pseudonode-id
      +---- net*                 area-address
      +---- multi-topology-id*   uint16
      +---- level?               level
augment /l3t:l3-link-event:
   +---- isis!
   +---- isis-link-attributes
      +---- multi-topology-id?   uint16
```

The module augments the ietf-l3-unicast-topology as follows:

- A new topology type is introduced for isis.
- Additional topology attributes are introduced in a new grouping which augments "topology-attributes" of the ietf-l3-unicast-topology module. The attributes include an ISIS NET-id identifying the area.
- Additional data objects for nodes are introduced by augmenting "node-attributes" of the ietf-l3-unicast-topology module. New objects include router-type, iso-system-id to identify the router, a list of multi-topology-id, a list of NET ids, and traffic engineering attributes.
- Links are augmented with multi-topology-id and traffic engineering link attributes.

In addition, the module augments nodes and links with IS-IS attributes.

Again, it should be noted that the model defined here represents a topology and is intended as an example. It does not define how to configure IS-IS routers or interfaces.

6.2.2. IS-IS Topology YANG Module

The IS-IS Topology YANG Module is specified as follows. As mentioned, the module is intended as an example for how the Layer 3 Unicast topology model can be extended to cover IS-IS topologies, but it is not normative. Accordingly, the module is not delimited with CODE BEGINS and CODE ENDS tags.

```
file "example-ietf-isis-topology@2017-01-04.yang"
module example-ietf-isis-topology {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:example-ietf-isis-topology";
  prefix "isist";
  import ietf-network {
    prefix "nd";
  }
  import ietf-network-topology {
    prefix "lnk";
  }
  import ietf-l3-unicast-topology {
    prefix "l3t";
  }
   organization
```

```
      "IETF I2RS (Interface to the Routing System) Working Group";
    contact
      "WG Web:    <http://tools.ietf.org/wg/i2rs/>
       WG List:   <mailto:i2rs@ietf.org>
       WG Chair:  Susan Hares
                  <mailto:shares@ndzh.com>
       WG Chair:  Russ White
                  <mailto:russ@riw.us>
       Editor:    Alexander Clemm
                  <mailto:sympotech.com>
       Editor:    Jan Medved
                  <mailto:jmedved@cisco.com>
       Editor:    Robert Varga
                  <mailto:robert.varga@pantheon.sk>
       Editor:    Xufeng Liu
                  <mailto:xliu@kuatrotech.com>
       Editor:    Nitin Bahadur
                  <mailto:nitin_bahadur@yahoo.com>
       Editor:    Hariharan Ananthakrishnan
                  <mailto:hari@packetdesign.com>";
    description
      "This module defines a model for IS-IS network topologies.
       Copyright (c) 2017 IETF Trust and the persons identified as
       authors of the code.  All rights reserved.
       Redistribution and use in source and binary forms, with or
       without modification, is permitted pursuant to, and subject
       to the license terms contained in, the Simplified BSD License
       set forth in Section 4.c of the IETF Trust's Legal Provisions
       Relating to IETF Documents
       (http://trustee.ietf.org/license-info).
       This version of this YANG module is part of
       draft-ietf-i2rs-yang-l3-topology-08;
       see the RFC itself for full legal notices.
       NOTE TO RFC EDITOR: Please replace above reference to
       draft-ietf-i2rs-yang-l3-topology-08 with RFC
       number when published (i.e. RFC xxxx).";
     revision "2017-01-04" {
      description
        "Initial revision.
         NOTE TO RFC EDITOR: Please replace the following reference
         to draft-ietf-i2rs-yang-l3-topology-08 with
         RFC number when published (i.e. RFC xxxx).";
      reference
        draft-ietf-i2rs-yang-l3-topology-08;
    }
    typedef iso-pseudonode-id {
      type string {
        pattern '[0-9a-fA-F]{2}';
```

```
      }
      description
           "ISO pseudonode id for broadcast network.";
    }
    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.";
    }
    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 ISIS system id using a pattern;
        an example of a system id looks like: 0143.0438.AeF0.";
    }
    typedef level {
      type enumeration {
        enum "level-1" {
          description
           "This enum describes L1 only capability.";
        }
        enum "level-2" {
          description
           "This enum describes L2 only capability.";
        }
        enum "level-all" {
          description
           "This enum describes both levels (L1 and L2) capability.";
        }
      }
      default "level-all";
      description
       "This type defines the ISIS level of an object.";
    }
    grouping isis-topology-type {
      description
        "Identifies the ISIS topology type.";
      container isis {
        presence "Indicates ISIS Topology";
        description
          "Its presence identifies the ISIS topology type.";
      }
    }
```

```
    augment "/nd:networks/nd:network/nd:network-types/"
      +"l3t:l3-unicast-topology" {
      description
        "Defines the ISIS topology type.";
      uses isis-topology-type;
    }
    augment "/nd:networks/nd:network/l3t:l3-topology-attributes" {
      when "../nd:network-types/l3t:l3-unicast-topology/isis" {
        description
          "Augment only for ISIS topology";
      }
      description
        "Augment topology configuration";
      container isis-topology-attributes {
        description
          "Containing topology attributes";
        leaf net {
          type area-address;
          description
            "ISO NET ID value";
        }
      }
    }
    augment "/nd:networks/nd:network/nd:node/"+
      "l3t:l3-node-attributes" {
      when "../../nd:network-types/l3t:l3-unicast-topology/isis" {
        description
          "Augment only for ISIS topology";
      }
      description
        "Augment node configuration";
      uses isis-node-attributes;
    }
    augment "/nd:networks/nd:network/lnk:link/l3t:l3-link-attributes" {
      when "../../nd:network-types/l3t:l3-unicast-topology/isis" {
        description
          "Augment only for ISIS topology";
      }
      description
        "Augment link configuration";
      uses isis-link-attributes;
    }
    grouping isis-node-attributes {
      description
        "ISIS node scope attributes";
      container isis-node-attributes {
        description
          "Containing node attributes";
```

```
        container iso {
          description
            "Containing ISO atrributes";
          leaf iso-system-id {
            type system-id;
            description
              "ISO system ID";
          }
          leaf iso-pseudonode-id {
            type iso-pseudonode-id;
            default "00";
            description
              "Pseudonode ID";
          }
        }
        leaf-list net {
          type area-address;
          max-elements 3;
          description
            "List of ISO NET IDs";
        }
        leaf-list multi-topology-id {
          type uint16 {
            range "0..4095";
          }
          max-elements "128";
          description
            "List of Multi Topology Identifier up to 128 (0-127).
            RFC 4915";
        }
        leaf level {
          type level;
          description "Level 1, Level 2 or Level 1 and 2";
        }
      }
    }
    grouping isis-link-attributes {
      description
        "ISIS link scope attributes";
      container isis-link-attributes {
        description
          "Containing link attributes";
        leaf multi-topology-id {
          type uint16 {
            range "0..4095";
          }
          description
            "Multi topology ID";
```

```
        }
      }
    }
    augment "/l3t:l3-node-event" {
      description
        "ISIS node event";
      uses isis-topology-type;
      uses isis-node-attributes;
    }
    augment "/l3t:l3-link-event" {
      description
        "ISIS link event";
      uses isis-topology-type;
      uses isis-link-attributes;
    }
  }
```

7. Interactions with Other YANG Modules

As described in section Section 3, the model builds on top of, and augments, the YANG modules defined in [I-D.draft-ietf-i2rs-yang-network-topo]. Specifically, module ietf-l3-unicast-topology augments modules "ietf-network" and "ietf-network-topology". In addition, the model makes use of data types that have been defined in [RFC6991].

The moodel defines a protocol independent YANG data model with layer 3 topology information. It is separate from and not linked with data models that are used to configure routing protocols or routing information. This includes e.g. model "ietf-routing" [RFC8022] and model "ietf-fb-rib" [I-D.draft-acee-rtgwg-yang-rib-extend].

The model obeys the requirements for the ephemeral state found in the document [I-D.draft-ietf-i2rs-ephemeral-state]. For ephemeral topology data that is server provided, the process tasked with maintaining topology information will load information from the routing process (such as OSPF) into the data model without relying on a configuration datastore.

8. IANA Considerations

This document registers the following namespace URI in the "IETF XML Registry" [RFC3688]:

URI: urn:ietf:params:xml:ns:yang:ietf-l3-unicast-topology
Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

This document registers the following YANG module in the "YANG Module Names" registry [RFC6020]:

Name: ietf-l3-unicast-topology
Namespace: urn:ietf:params:xml:ns:yang:ietf-l3-unicast-topology
Prefix: l3t
Reference: draft-ietf-i2rs-yang-l3-topology-08.txt (RFC form)

9. Security Considerations

The YANG module defined in this memo is designed to be accessed via the NETCONF protocol [RFC6241]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The NETCONF access control model [RFC6536] provides the means to restrict access for particular NETCONF users to a pre-configured subset of all available NETCONF protocol operations and content.

In general, Layer 3 Unicast topologies are server-provided and provide ephemeral topology information. As they provide read-only access to clients, they are less vulnerable. That said, the YANG module does in principle allow information to be configurable in certain instances (when the server-provided flag for the topology is set to false). In such cases, a malicious client could introduce topologies that are undesired. For example, a client could remove or add topological links between nodes, which could lead to an undesired and suboptimal topology, which might impact service levels and network utilization. It is therefore important that the NETCONF access control model is vigorously applied to prevent topology configuration by unauthorized clients.

10. Contributors

The model presented in this paper was contributed to by more people than can be listed on the author list. Additional contributors include:

- Vishnu Pavan Beeram, Juniper
- Igor Bryskin, Huawei
- Ken Gray, Cisco
- Aihua Guo, Huawei
- Tom Nadeau, Brocade
- Tony Tkacik

o Aleksandr Zhdankin, Cisco

11. Acknowledgements

We wish to acknowledge the helpful contributions, comments, and suggestions that were received from Ladislav Lhotka, Andy Bierman, Carlos Pignataro, Joel Halpern, Juergen Schoenwaelder, Alia Atlas, Susan Hares, Benoit Claise, and Carl Moberg.

12. References

12.1. Normative References

[I-D.draft-ietf-i2rs-yang-network-topo] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", I-D draft-ietf-i2rs-yang-network-topo-10, January 2017.

[RFC1195] Callon, R., "Use of OSI IS-IS for Routing in TCP/IP and Dual Environments", RFC 1195, December 1990.

[RFC2328] Moy, J., "OSPF Version 2", RFC 2328, April 1998.

[RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, June 2000.

[RFC3688] Mealling, M., "The IETF XML Registry", RFC 3688, January 2004.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, June 2011.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, March 2012.

[RFC6991] Schoenwaelder, J., "Common YANG Data Types", RFC 6991, July 2013.

[RFC7950] Bjorklund, M., "The YANG 1.1 Data Modeling Language", RFC 7950, August 2016.

12.2. Informative References

[I-D.draft-acee-rtgwg-yang-rib-extend]
Lindem, A. and Y. Qu, "YANG Data Model for RIB Extensions", I-D draft-acee-rtgwg-yang-rib-extend-02, October 2016.

[I-D.draft-ietf-i2rs-ephemeral-state]
Haas, J. and S. Hares, "I2RS Ephemeral State Requirements", I-D draft-ietf-i2rs-ephemeral-state-23, November 2016.

[I-D.draft-ietf-i2rs-usecase-reqs-summary]
Hares, S. and M. Chen, "Summary of I2RS Use Case Requirements", I-D draft-ietf-i2rs-usecase-reqs-summary-03, November 2016.

[I-D.draft-ietf-teas-yang-te-topo]
Liu, X., Bryskin, I., Beeram, V., Saad, T., Shah, H., and O. Gonzelz De Dios, "YANG Data Model for TE Topologies", I-D draft-ietf-teas-yang-te-topo-06, October 2016.

[RFC7223] Bjorklund, M., "A YANG Data Model for Routing Management", RFC 7223, May 2014.

[RFC8022] Lhotka, L. and A. Lindem, "A YANG Data Model for Routing Management", RFC 8022, November 2016.

Authors' Addresses

Alexander Clemm
Huawei

EMail: ludwig@clemm.org

Jan Medved
Cisco

EMail: jmedved@cisco.com

Robert Varga
Pantheon Technologies SRO

EMail: robert.varga@pantheon.sk

Xufeng Liu
Ericsson

EMail: xliu@kuatrotech.com

Hariharan Ananthakrishnan
Packet Design

EMail: hari@packetdesign.com

Nitin Bahadur
Bracket Computing

EMail: nitin_bahadur@yahoo.com