BESS Working Group
Internet Draft
Intended status: Standards Track
Expires: March 19, 2018

Y. Liu
F. Guo
Huawei Technologies
X. Liu
Jabil
R. Kebler
Juniper Networks
M. Sivakumar
Cisco

Sep 19, 2017

Yang Data Model for Multicast in MPLS/BGP IP VPNs
draft-liu-bess-mvpn-yang-05

Status of this Memo

Copyright Notice

Abstract

This document defines a YANG data model that can be used to configure and manage multicast in MPLS/BGP IP VPNs.

Table of Contents

1. Introduction

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

This document defines a YANG data model that can be used to configure and manage Multicast in MPLS/BGP IP VPN (MVPN). It includes Cisco systems' solution [RFC6037], BGP MVPN [RFC6513] [RFC6514] etc. Currently this model is incomplete, but it will support the core MVPN protocols, as well as many other features mentioned in separate MVPN RFCs. In addition, Non-core features described in MVPN standards other than mentioned above RFC in future version.

1.1. 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 RFC-2119 [RFC2119].

1.2. Terminology

The terminology for describing YANG data models is found in [RFC6020].

This draft employs YANG tree diagrams, which are explained in [I-D.ietf-netmod-rfc6087bis].

2. Design of Data model

2.1. Scope of model

The model covers Rosen MVPN [RFC6037], BGP MVPN [RFC6513] [RFC6514]. The representation of some of extension features is not completely specified in this draft of the data model. This model is being circulated in its current form for early oversight and review of the basic hierarchy.

The operational state fields of this model are also incomplete, though the structure of what has been written may be taken as representative of the structure of the model when complete.

This model does not cover other MVPN related protocols such as MVPN Extranet [RFC7900] or MVPN MLDP In-band signaling [RFC7246] etc., these will be covered by future Internet Drafts.

2.2. Optional capabilities

This model is designed to represent the capabilities of MVPN devices with various specifications, including some with basic subsets of the MVPN protocols. The main design goals of this draft are that any major now-existing implementation may be said to support the basic model, and that the configuration of all implementations meeting the specification is easy to express through some combination of the features in the basic model and simple vendor augmentations.

On the other hand, operational state parameters are not so widely designated as features, as there are many cases where the defaulting of an operational state parameter would not cause any harm to the system, and it is much more likely that an implementation without native support for a piece of operational state would be able to

derive a suitable value for a state variable that is not natively supported.

For the same reason, wide constant ranges (for example, timer maximum and minimum) will be used in the model. It is expected that vendors will augment the model with any specific restrictions that might be required. Vendors may also extend the features list with proprietary extensions.

2.3. Position of address family in hierarchy

The current draft contains MVPN IPv4 and IPv6 as separate schema branches in the structure. The reason for this is to inherit l3vpn yang model structure and make it easier for implementations which may optionally choose to support specific address families. And the names of objects may be different between the IPv4 and IPv6 address families.

3. Module Structure

The MVPN YANG model follows the Guidelines for YANG Module Authors (NMDA) [draft-dsdt-nmda-guidelines-01].The MVPN modules define the network-instance-wide configuration and operational state options in a two-level hierarchy as listed below:

Instance level: Only including configuration data nodes now. MVPN configuration attributes for the entire routing instance, including route-target, I-PMSI tunnel and S-PMSI number, common timer etc.

PMSI tunnel level: MVPN configuration attributes applicable to the I-PMSI and per S-PMSI tunnel configuration attributes, including tunnel mode, tunnel specific parameters and threshold etc. MVPN PMSI tunnel operational state attributes applicable to the I-PMSI and per S-PMSI tunnel operational state attributes, including tunnel mode, tunnel role, tunnel specific parameters and referenced private source and group address etc.

Where fields are not genuinely essential to protocol operation, they are marked as optional. Some fields will be essential but have a default specified, so that they need not be configured explicitly.

We define the MVPN model as a network-instance-centric model, and the MVPN model will augment "/ni:network-instances/ni:network-instance:" in [I-D.ietf-rtgwg-ni-model] and will allow a single mvpn instance per VRF.

```
augment /ni:network-instances/ni:network-instance:
  +--rw mvpn
```

```
      +--rw mvpnv4
      |  +--rw signaling-mode?                enumeration
      |  +--rw auto-discovery-mode?           enumeration
      |  +--rw config-type?                   enumeration
      |  +--rw is-sender-site?                boolean
      |  +--rw rpt-spt-mode?                  enumeration
      |  +--rw mvpn-route-targets
      |  |  +--rw mvpn-route-target* [rt-type rt-value]
      |  |     +--rw rt-type     enumeration
      |  |     +--rw rt-value    string
      |  +--rw mvpn-ipmsi-tunnel
      |  |  +--rw tunnel-type?                enumeration
      |  |  +--rw (ipmsi-tunnel-attribute)?
      |  |     +--:(p2mp-te)
      |  |     |  +--rw te-p2mp-template?           string
      |  |     +--:(p2mp-mldp)
      |  |     +--:(pim-ssm)
      |  |     |  +--rw ssm-default-group-addr?     inet:ip-address
      |  |     +--:(pim-sm)
      |  |     |  +--rw sm-default-group-addr?      inet:ip-address
      |  |     +--:(bidir-pim)
      |  |     |  +--rw bidir-default-group-addr?   inet:ip-address
      |  |     +--:(ingress-replication)
      |  |     +--:(mp2mp-mldp)
      |  +--rw mvpn-spmsi-tunnels
      |  |  +--rw switch-delay-time?           uint8
      |  |  +--rw switch-back-holddown-time?   uint16
      |  |  +--rw tunnel-limit?                uint16
      |  |  +--rw mvpn-spmsi-tunnel* [tunnel-type]
      |  |     +--rw tunnel-type                     enumeration
      |  |     +--rw (spmsi-tunnel-attribute)?
      |  |     |  +--:(p2mp-te)
      |  |     |  |  +--rw te-p2mp-template?              string
      |  |     |  +--:(p2mp-mldp)
      |  |     |  +--:(pim-ssm)
      |  |     |  |  +--rw ssm-group-pool-addr?           inet:ip-address
      |  |     |  |  +--rw ssm-group-pool-masklength?     uint8
      |  |     |  +--:(pim-sm)
      |  |     |  |  +--rw sm-group-pool-addr?            inet:ip-address
      |  |     |  |  +--rw sm-group-pool-masklength?      uint8
      |  |     |  +--:(bidir-pim)
      |  |     |  |  +--rw bidir-group-pool-addr?         inet:ip-address
      |  |     |  |  +--rw bidir-group-pool-masklength?   uint8
      |  |     |  +--:(ingress-replication)
      |  |     |  +--:(mp2mp-mldp)
      |  |     +--rw switch-threshold?               uint32
      |  |     +--rw switch-wildcard-mode?           enumeration
      |  |     +--rw (address-mask-or-acl)?
      |  |        +--:(address-mask)
```

```
       |  |        |  +--rw ipv4-group-addr?               inet:ipv4-address
       |  |        |  +--rw ipv4-group-masklength?         uint8
       |  |        |  +--rw ipv4-source-addr?              inet:ipv4-address
       |  |        |  +--rw ipv4-source-masklength?        uint8
       |  |        +--:(acl)
       |  |           +--rw group-acl-ipv4?                string
       |  +--ro mvpn-ipmsi-tunnel-info
       |  |  +--ro tunnel-type?                      enumeration
       |  |  +--ro (pmsi-tunnel-attribute)?
       |  |  |  +--:(p2mp-te)
       |  |  |  |  +--ro te-p2mp-id?                      uint16
       |  |  |  |  +--ro te-tunnel-id?                    uint16
       |  |  |  |  +--ro te-extend-tunnel-id?             uint16
       |  |  |  +--:(p2mp-mldp)
       |  |  |  |  +--ro mldp-root-addr?                  inet:ip-address
       |  |  |  |  +--ro mldp-lsp-id?                     string
       |  |  |  +--:(pim-ssm)
       |  |  |  |  +--ro ssm-group-addr?                  inet:ip-address
       |  |  |  +--:(pim-sm)
       |  |  |  |  +--ro sm-group-addr?                   inet:ip-address
       |  |  |  +--:(bidir-pim)
       |  |  |  |  +--ro bidir-group-addr?                inet:ip-address
       |  |  |  +--:(ingress-replication)
       |  |  |  +--:(mp2mp-mldp)
       |  |  +--ro tunnel-role?                      enumeration
       |  |  +--ro mvpn-pmsi-ipv4-ref-sg-entries
       |  |     +--ro mvpn-pmsi-ipv4-ref-sg-entries* [ipv4-source-address ipv4-g
roup-address]
       |  |        +--ro ipv4-source-address    inet:ipv4-address
       |  |        +--ro ipv4-group-address     inet:ipv4-address
       |  +--ro mvpn-spmsi-tunnel-ipv4-info
       |     +--ro mvpn-spmsi-tunnel-ipv4-info* [tunnel-type]
       |        +--ro tunnel-type                       enumeration
       |        +--ro (pmsi-tunnel-attribute)?
       |        |  +--:(p2mp-te)
       |        |  |  +--ro te-p2mp-id?                      uint16
       |        |  |  +--ro te-tunnel-id?                    uint16
       |        |  |  +--ro te-extend-tunnel-id?             uint16
       |        |  +--:(p2mp-mldp)
       |        |  |  +--ro mldp-root-addr?                  inet:ip-address
       |        |  |  +--ro mldp-lsp-id?                     string
       |        |  +--:(pim-ssm)
       |        |  |  +--ro ssm-group-addr?                  inet:ip-address
       |        |  +--:(pim-sm)
       |        |  |  +--ro sm-group-addr?                   inet:ip-address
       |        |  +--:(bidir-pim)
       |        |  |  +--ro bidir-group-addr?                inet:ip-address
       |        |  +--:(ingress-replication)
       |        |  +--:(mp2mp-mldp)
       |        +--ro tunnel-role?                      enumeration
```

```
       |        +--ro mvpn-pmsi-ipv4-ref-sg-entries
       |           +--ro mvpn-pmsi-ipv4-ref-sg-entries* [ipv4-source-address ipv
4-group-address]
       |              +--ro ipv4-source-address    inet:ipv4-address
       |              +--ro ipv4-group-address     inet:ipv4-address
       +--rw mvpnv6
          +--rw signaling-mode?                enumeration
          +--rw auto-discovery-mode?           enumeration
          +--rw config-type?                   enumeration
          +--rw is-sender-site?                boolean
          +--rw rpt-spt-mode?                  enumeration
          +--rw mvpn-route-targets
          |  +--rw mvpn-route-target* [rt-type rt-value]
          |     +--rw rt-type     enumeration
          |     +--rw rt-value    string
          +--rw mvpn-ipmsi-tunnel
          |  +--rw tunnel-type?                enumeration
          |  +--rw (ipmsi-tunnel-attribute)?
          |     +--:(p2mp-te)
          |     |  +--rw te-p2mp-template?           string
          |     +--:(p2mp-mldp)
          |     +--:(pim-ssm)
          |     |  +--rw ssm-default-group-addr?     inet:ip-address
          |     +--:(pim-sm)
          |     |  +--rw sm-default-group-addr?      inet:ip-address
          |     +--:(bidir-pim)
          |     |  +--rw bidir-default-group-addr?   inet:ip-address
          |     +--:(ingress-replication)
          |     +--:(mp2mp-mldp)
          +--rw mvpn-spmsi-tunnels
          |  +--rw switch-delay-time?           uint8
          |  +--rw switch-back-holddown-time?   uint16
          |  +--rw tunnel-limit?                uint16
          |  +--rw mvpn-spmsi-tunnel* [tunnel-type]
          |     +--rw tunnel-type                      enumeration
          |     +--rw (spmsi-tunnel-attribute)?
          |     |  +--:(p2mp-te)
          |     |  |  +--rw te-p2mp-template?             string
          |     |  +--:(p2mp-mldp)
          |     |  +--:(pim-ssm)
          |     |  |  +--rw ssm-group-pool-addr?           inet:ip-address
          |     |  |  +--rw ssm-group-pool-masklength?     uint8
          |     |  +--:(pim-sm)
          |     |  |  +--rw sm-group-pool-addr?            inet:ip-address
          |     |  |  +--rw sm-group-pool-masklength?      uint8
          |     |  +--:(bidir-pim)
          |     |  |  +--rw bidir-group-pool-addr?         inet:ip-address
          |     |  |  +--rw bidir-group-pool-masklength?   uint8
          |     |  +--:(ingress-replication)
          |     |  +--:(mp2mp-mldp)
```

```
          |     +--rw switch-threshold?              uint32
          |     +--rw switch-wildcard-mode?          enumeration
          |     +--rw (address-mask-or-acl)?
          |        +--:(address-mask)
          |        |  +--rw ipv6-group-addr?               inet:ipv6-address
          |        |  +--rw ipv6-groupmasklength?          uint8
          |        |  +--rw ipv6-source-addr?              inet:ipv6-address
          |        |  +--rw ipv6-source-masklength?        uint8
          |        +--:(acl)
          |           +--rw group-acl-ipv6?                string
          |  +--ro mvpn-ipmsi-tunnel-info
          |  |  +--ro tunnel-type?                      enumeration
          |  |  +--ro (pmsi-tunnel-attribute)?
          |  |  |  +--:(p2mp-te)
          |  |  |  |  +--ro te-p2mp-id?                     uint16
          |  |  |  |  +--ro te-tunnel-id?                   uint16
          |  |  |  |  +--ro te-extend-tunnel-id?            uint16
          |  |  |  +--:(p2mp-mldp)
          |  |  |  |  +--ro mldp-root-addr?                 inet:ip-address
          |  |  |  |  +--ro mldp-lsp-id?                    string
          |  |  |  +--:(pim-ssm)
          |  |  |  |  +--ro ssm-group-addr?                 inet:ip-address
          |  |  |  +--:(pim-sm)
          |  |  |  |  +--ro sm-group-addr?                  inet:ip-address
          |  |  |  +--:(bidir-pim)
          |  |  |  |  +--ro bidir-group-addr?               inet:ip-address
          |  |  |  +--:(ingress-replication)
          |  |  |  +--:(mp2mp-mldp)
          |  |  +--ro tunnel-role?                      enumeration
          |  |  +--ro mvpn-pmsi-ipv6-ref-sg-entries
          |  |     +--ro mvpn-pmsi-ipv6-ref-sg-entries* [ipv6-source-address ipv6-g
roup-address]
          |  |        +--ro ipv6-source-address    inet:ipv6-address
          |  |        +--ro ipv6-group-address     inet:ipv6-address
          |  +--ro mvpn-spmsi-tunnel-ipv6-info
          |     +--ro mvpn-spmsi-tunnel-ipv6-info* [tunnel-type]
          |        +--ro tunnel-type                        enumeration
          |        +--ro (pmsi-tunnel-attribute)?
          |        |  +--:(p2mp-te)
          |        |  |  +--ro te-p2mp-id?                     uint16
          |        |  |  +--ro te-tunnel-id?                   uint16
          |        |  |  +--ro te-extend-tunnel-id?            uint16
          |        |  +--:(p2mp-mldp)
          |        |  |  +--ro mldp-root-addr?                 inet:ip-address
          |        |  |  +--ro mldp-lsp-id?                    string
          |        |  +--:(pim-ssm)
          |        |  |  +--ro ssm-group-addr?                 inet:ip-address
          |        |  +--:(pim-sm)
          |        |  |  +--ro sm-group-addr?                  inet:ip-address
          |        |  +--:(bidir-pim)
```

```
               |  |  +--ro bidir-group-addr?               inet:ip-address
               |  +--:(ingress-replication)
               |  +--:(mp2mp-mldp)
               +--ro tunnel-role?                    enumeration
               +--ro mvpn-pmsi-ipv6-ref-sg-entries
                  +--ro mvpn-pmsi-ipv6-ref-sg-entries* [ipv6-source-address ipv
6-group-address]
                     +--ro ipv6-source-address    inet:ipv6-address
                     +--ro ipv6-group-address     inet:ipv6-address
```

4. MVPN YANG Modules

```
   <CODE BEGINS> file "ietf-mvpn@2017-09-15.yang"
   module ietf-mvpn {
      namespace "urn:ietf:params:xml:ns:yang:ietf-mvpn";
      prefix mvpn;

      import ietf-network-instance {
        prefix ni;
      }

      import ietf-inet-types {
        prefix inet;
      }


      organization
        "IETF BESS(BGP Enabled Services) Working Group";
      contact
        "
        Yisong Liu
        <mailto:liuyisong@huawei.com>
        Feng Guo
        <mailto:guofeng@huawei.com>
        Xufeng Liu
        <mailto:Xufeng_Liu@jabil.com>
        Robert Kebler
        <mailto:rkebler@juniper.net>
        Mahesh Sivakumar
        <mailto:masivaku@cisco.com>";
      description
        "This YANG module defines the generic configuration
         and operational state data for mvpn, which is common across
         all of the vendor implementations of the protocol. It is
         intended that the module will be extended by vendors to
         define vendor-specific mvpn parameters.";

      revision 2017-09-15 {
        description
```

```
          "Update for NMDA version and errata.";
        reference
          "RFC XXXX: A YANG Data Model for MVPN";
      }
      revision 2017-07-03 {
        description
          "Update S-PMSI configuration and errata.";
        reference
          "RFC XXXX: A YANG Data Model for MVPN";
      }
      revision 2016-10-28 {
        description
          "Initial revision.";
        reference
          "RFC XXXX: A YANG Data Model for MVPN";
      }

      grouping mvpn-instance-config {
        description "Mvpn basic configuration per instance.";

        leaf signaling-mode {
          type enumeration {
            enum invalid {
              value "0";
              description "invalid";
            }
            enum bgp {
              value "1";
              description "bgp";
            }
            enum pim {
              value "2";
              description "pim";
            }
            enum mldp {
              value "3";
              description "mldp";
            }
          }
          default "invalid";
          description "Signaling mode for C-multicast route.";
        }
        leaf auto-discovery-mode {
          type enumeration {
            enum none {
              value "0";
              description "none";
            }
            enum ad {
```

```
            value "1";
            description "auto-discovery by BGP";
          }
        }
        default "none";
        description "Auto discovery mode.";
      }
      leaf config-type {
        type enumeration {
          enum md {
            value "0";
            description "md(rosen)";
          }
          enum ng {
            value "1";
            description "ng";
          }
        }
        default "md";
        description "Mvpn type, which can be md(rosen) mvpn or ng mvpn.";
      }
      leaf is-sender-site {
        type boolean;
        default "false";
        description "Configure the current PE as a sender PE.";
      }
      leaf rpt-spt-mode {
        type enumeration {
          enum spt-only {
            value "0";
            description
              "Only spt mode for crossing public net.";
          }
          enum rpt-spt {
            value "1";
            description
              "Both rpt and spt mode for corssing public net.";
          }
        }
        default "spt-only";
        description
          "ASM mode in multicast private net for crossing public net.";
      }

    }

    grouping mvpn-vpn-targets {
      description "May be different from l3vpn unicast route-targets";
      container mvpn-route-targets{
```

```
          description "Multicast vpn route-targets";
          list mvpn-route-target {
            key "rt-type rt-value" ;
            description
              "List of multicast route-targets" ;
            leaf rt-type {
               type enumeration {
                 enum export-extcommunity {
                   value "0";
                   description "export-extcommunity";
                 }
                 enum import-extcommunity {
                   value "1";
                   description "import-extcommunity";
                 }
               }
               mandatory "true";
               description
                 "rt types are as follows:
                  export-extcommunity: specifies the value of
                  the extended community attribute of the
                  route from an outbound interface to the
                  destination vpn.
                  import-extcommunity: receives routes that
                  carry the specified extended community
                  attribute";
             }
             leaf rt-value {
               type string {
                 length "3..21";
               }
               description
                 "the available mvpn target formats are as
                  follows:
                  - 16-bit as number:32-bit user-defined
                  number, for example, 1:3. an as number
                  ranges from 0 to 65535, and a user-defined
                  number ranges from 0 to 4294967295. The as
                  number and user-defined number cannot be
                  both 0s. That is, a vpn target cannot be 0:0.
                  - 32-bit ip address:16-bit user-defined
                  number, for example, 192.168.122.15:1.
                  The ip address ranges from 0.0.0.0 to
                  255.255.255.255, and the user-defined
                  number ranges from 0 to 65535.";
             }
           }
         }
       }
```

```
      grouping mvpn-ipmsi-tunnel-config {
        description "Default mdt for rosen mvpn and I-PMSI for ng mvpn";

        container mvpn-ipmsi-tunnel {
          description "I-PMSI tunnel configuraton";
          leaf tunnel-type {
            type enumeration {
              enum invalid {
                value "0";
                description "invalid";
              }
              enum p2mp-te {
                value "1";
                description "p2mp-te";
              }
              enum p2mp-mldp {
                value "2";
                description "p2mp-mldp";
              }
              enum pim-ssm {
                value "3";
                description "pim-ssm";
              }
              enum pim-sm {
                value "4";
                description "pim-sm";
              }
              enum bidir-pim {
                value "5";
                description "bidir-pim";
              }
              enum ingress-replication {
                value "6";
                description "ingress-replication";
              }
              enum mp2mp-mldp {
                value "7";
                description "mp2mp-mldp";
              }
            }
            description "I-PMSI tunnel type.";
          }
          choice ipmsi-tunnel-attribute {
            description "I-PMSI tunnel attributes configuration";
            case p2mp-te {
              description "P2mp TE tunnel";
              leaf te-p2mp-template {
                type string {
```

```
                length "1..31";
              }
              description "P2mp te tunnel template";
            }
          }
          case p2mp-mldp {
            description "Mldp tunnel";
          }
          case pim-ssm {
            description "Pim ssm tunnel";
            leaf ssm-default-group-addr {
              type inet:ip-address;
              description "Default mdt or I-PMSI group address.";
            }
          }
          case pim-sm {
            description "Pim sm tunnel";
            leaf sm-default-group-addr {
              type inet:ip-address;
              description "Default mdt or I-PMSI group address.";
            }
          }
          case bidir-pim {
            description "Bidir pim tunnel";
            leaf bidir-default-group-addr {
              type inet:ip-address;
              description "Default mdt or I-PMSI group address.";
            }
          }
          case ingress-replication {
            description "Ingress replication p2p tunnel";
          }
          case mp2mp-mldp {
            description "Mp2mp mldp tunnel";
          }
        }
      }
    }

    grouping mvpn-spmsi-tunnel-basic-config {
      description "S-PMSI tunnel basic configuration";
      leaf tunnel-type {
        type enumeration {
          enum invalid {
            value "0";
            description "invalid";
          }
          enum p2mp-te {
            value "1";
```

```
            description "p2mp-te";
          }
          enum p2mp-mldp {
            value "2";
            description "p2mp-mldp";
          }
          enum pim-ssm {
            value "3";
            description "pim-ssm";
          }
          enum pim-sm {
            value "4";
            description "pim-sm";
          }
          enum bidir-pim {
            value "5";
            description "bidir-pim";
          }
          enum ingress-replication {
            value "6";
            description "ingress-replication";
          }
          enum mp2mp-mldp {
            value "7";
            description "mp2mp-mldp";
          }
        }
        description "S-PMSI tunnel type.";
      }
      choice spmsi-tunnel-attribute {
        description "S-PMSI tunnel attributes configuration";
        case p2mp-te {
          description "P2mp te tunnel";
          leaf te-p2mp-template {
            type string {
              length "1..31";
            }
            description "P2mp te tunnel template";
          }
        }
        case p2mp-mldp {
          description "Mldp tunnel";
        }
        case pim-ssm {
          description "Pim ssm tunnel";
          leaf ssm-group-pool-addr {
            type inet:ip-address;
            description "Group pool address for data mdt or pim s-pmsi.";
          }
```

```
            leaf ssm-group-pool-masklength {
              type uint8 {
                range "8..128";
              }
              description "Group pool mask for data mdt or pim s-pmsi";
            }
          }
          case pim-sm {
            description "Pim sm tunnel";
            leaf sm-group-pool-addr {
              type inet:ip-address;
              description "Group pool address for data mdt or pim s-pmsi.";
            }
            leaf sm-group-pool-masklength {
              type uint8 {
                range "8..128";
              }
              description "Group pool mask for data mdt or pim s-pmsi";
            }
          }
          case bidir-pim {
            description "Bidir pim tunnel";
            leaf bidir-group-pool-addr {
              type inet:ip-address;
              description "Group pool address for data mdt or pim s-pmsi.";
            }
            leaf bidir-group-pool-masklength {
              type uint8 {
                range "8..128";
              }
              description "Group pool mask for data mdt or pim s-pmsi";
            }
          }
          case ingress-replication {
            description "Ingress replication p2p tunnel";
          }
          case mp2mp-mldp {
            description "Mp2mp mldp tunnel";
          }
        }
        leaf switch-threshold {
          type uint32 {
            range "0..4194304";
          }
          default "0";
          description
            "Multicast packet rate threshold for
             triggering the switching from the
             I-PMSI to the S-PMSI. The value is
```

```
             an integer ranging from 0 to 4194304, in
             kbit/s. The default value is 0.";
        }
        leaf switch-wildcard-mode {
          type enumeration {
            enum source-group {
              value "0";
              description
                "Wildcard neither for source or group address.";
            }
            enum star-star {
              value "1";
              description
                "Wildcard for both source and group address.";
            }
            enum star-group {
              value "2";
              description
                "Wildcard only for source address.";
            }
            enum source-star {
              value "3";
              description
                "Wildcard only for group address.";
            }
          }
          default "source-group";
          description
            "I-PMSI switching to S-PMSI mode for private net
            wildcard mode, which including (*,*), (*,G), (S,*),
            (S,G) four modes.";
        }
      }

      grouping mvpn-spmsi-tunnel-config-ipv4 {
        description
          "Data mdt for rosen mvpn or S-PMSI for ng mvpn in
           IPv4 private network";

        container mvpn-spmsi-tunnels {
          description "S-PMSI tunnel configuration";
          leaf switch-delay-time {
            type uint8 {
              range "3..60";
            }
            units seconds;
            default "5";
            description
            "Delay for switching from the I-PMSI to
```

```
             the S-PMSI. The value is an integer
             ranging from 3 to 60, in seconds. ";
          }
          leaf switch-back-holddown-time {
            type uint16 {
              range "0..512";
            }
            units seconds;
            default "60";
            description
              "Delay for switching back from the S-PMSI
               to the I-PMSI. The value is an integer
               ranging from 0 to 512, in seconds. ";
          }
          leaf tunnel-limit {
            type uint16 {
              range "1..1024";
            }
            description
              "Maximum number of s-pmsi tunnels allowed.";
          }

          list mvpn-spmsi-tunnel {
            key "tunnel-type";
            description "S-PMSI tunnel attributes configuration";

            uses mvpn-spmsi-tunnel-basic-config;

            choice address-mask-or-acl {
              description
                "Type of definition of private net multicast address range";
              case address-mask {
                description "Use the type of address and mask";
                leaf ipv4-group-addr {
                  type inet:ipv4-address;
                  description
                    "Start and end ipv4 addresses of the group
                     address in private net. ";
                }
                leaf ipv4-group-masklength {
                  type uint8 {
                    range "4..32";
                  }
                  description
                    "Group mask length for ipv4 addresses in
                     the group address pool in private net.";
                }
                leaf ipv4-source-addr {
                  type inet:ipv4-address;
```

```
                description
                  "Start and end ipv4 addresses of the source
                   address in private net.";
              }
              leaf ipv4-source-masklength {
                type uint8 {
                  range "0..32";
                }
                description
                  "Source mask length for ipv4 addresses in
                   the group address pool in private net.";
              }
            }
            case acl {
              description "Use the type of acl";
              leaf group-acl-ipv4 {
                type string {
                  length "1..32";
                }
                description
                  "Specify the (s, g) entry on which the
                   S-PMSI tunnel takes effect.
                   The value is an integer ranging from 3000
                   to 3999 or a string of 32 case-sensitive
                   characters. If no value is specified, the
                   switch-group address pool takes effect on
                   all (s, g).";
              }
            }
          }
        }
      }
    }

    grouping mvpn-spmsi-tunnel-config-ipv6 {
      description
        "Data mdt for rosen mvpn or S-PMSI for ng mvpn in
         IPv6 private network";

      container mvpn-spmsi-tunnels {
        description "S-PMSI tunnel configuration";
        leaf switch-delay-time {
          type uint8 {
            range "3..60";
          }
          units seconds;
          default "5";
          description
          "Delay for switching from the I-PMSI to
```

```
             the S-PMSI. The value is an integer
             ranging from 3 to 60, in seconds. ";
          }
          leaf switch-back-holddown-time {
            type uint16 {
              range "0..512";
            }
            units seconds;
            default "60";
            description
              "Delay for switching back from the S-PMSI
               to the I-PMSI. The value is an integer
               ranging from 0 to 512, in seconds. ";
          }
          leaf tunnel-limit {
            type uint16 {
              range "1..1024";
            }
            description
              "Maximum number of s-pmsi tunnels allowed.";
          }

          list mvpn-spmsi-tunnel {
            key "tunnel-type";
            description "S-PMSI tunnel parameter configuration";

            uses mvpn-spmsi-tunnel-basic-config;

            choice address-mask-or-acl {
              description
                "Type of definition of private net multicast address range";
              case address-mask {
                description "Use the type of address and mask";
                leaf ipv6-group-addr {
                  type inet:ipv6-address;
                  description
                    "Start and end ipv6 addresses of the group
                     address in private net.";
                }
                leaf ipv6-groupmasklength {
                  type uint8 {
                    range "8..128";
                  }
                  description
                    "Group mask length for ipv6 addresses in
                     the group address pool in private net.";
                }
                leaf ipv6-source-addr {
                  type inet:ipv6-address;
```

```
                description
                  "Start and end ipv6 addresses of the source
                   address in private net.";
              }
              leaf ipv6-source-masklength {
                type uint8 {
                  range "0..128";
                }
                description
                  "Source mask length for ipv6 addresses in
                   the group address pool in private net.";
              }
            }
            case acl {
              description "Use the type of acl";
              leaf group-acl-ipv6 {
                type string {
                  length "1..32";
                }
                description
                  "Specify the (s, g) entry on which the
                   S-PMSI tunnel takes effect.
                   The value is an integer ranging from 3000
                   to 3999 or a string of 32 case-sensitive
                   characters. If no value is specified, the
                   switch-group address pool takes effect on
                   all (s, g).";
              }
            }
          }
        }
      }

      grouping mvpn-pmsi-state {
        description "PMSI tunnel operational state information";
        leaf tunnel-type {
          type enumeration {
            enum invalid {
              value "0";
              description "invalid";
            }
            enum p2mp-te {
              value "1";
              description "p2mp-te";
            }
            enum p2mp-mldp {
              value "2";
              description "p2mp-mldp";
```

```
            }
            enum pim-ssm {
              value "3";
              description "pim-ssm";
            }
            enum pim-sm {
              value "4";
              description "pim-sm";
            }
            enum bidir-pim {
              value "5";
              description "bidir-pim";
            }
            enum ingress-replication {
              value "6";
              description "ingress-replication";
            }
            enum mp2mp-mldp {
              value "7";
              description "mp2mp-mldp";
            }
          }
          description "PMSI tunnel type.";
        }
        choice pmsi-tunnel-attribute {
          description "PMSI tunnel operational state information for each type";
          case p2mp-te {
            description "P2mp te tunnel";
            leaf te-p2mp-id {
              type uint16 {
                range "0..65535";
              }
              default "0";
              description "P2mp id of the p2mp tunnel.";
            }
            leaf te-tunnel-id {
              type uint16 {
                range "1..65535";
              }
              description "Id of the p2mp tunnel.";
            }
            leaf te-extend-tunnel-id {
              type uint16 {
                range "1..65535";
              }
              description "P2mp extended tunnel interface id.";
            }
          }
          case p2mp-mldp {
```

```
            description "P2mp mldp tunnel";
            leaf mldp-root-addr {
              type inet:ip-address;
              description "Ip address of the root of a p2mp ldp lsp.";
            }
            leaf mldp-lsp-id {
              type string {
                length "1..256";
              }
              description "P2mp ldp lsp id.";
            }
          }
          case pim-ssm {
            description "Pim ssm tunnel";
            leaf ssm-group-addr {
              type inet:ip-address;
              description "Group address for pim ssm";
            }
          }
          case pim-sm {
            description "Pim sm tunnel";
            leaf sm-group-addr {
              type inet:ip-address;
              description "Group address for pim sm";
            }
          }
          case bidir-pim {
            description "Bidir pim tunnel";
            leaf bidir-group-addr {
              type inet:ip-address;
              description "Group address for bidir-pim";
            }
          }
          case ingress-replication {
            description "Ingress replication p2p tunnel";
          }
          case mp2mp-mldp {
            description "mp2mp mldp tunnel";
          }
        }
        leaf tunnel-role {
          type enumeration {
            enum none {
              value "0";
              description "none";
            }
            enum root {
              value "1";
              description "root";
```

```
            }
            enum leaf {
              value "2";
              description "leaf";
            }
            enum root-and-leaf {
              value "3";
              description "root-and-leaf";
            }
          }
          description "Role of a tunnel node.";
        }
      }

      grouping mvpn-pmsi-ipv4-entry {
        description
          "Multicast entries in ipv4 mvpn referenced the pmsi tunnel";
        container mvpn-pmsi-ipv4-ref-sg-entries {
          description
            "Multicast entries in ipv4 mvpn referenced the pmsi tunnel";
          list mvpn-pmsi-ipv4-ref-sg-entries {
            key "ipv4-source-address ipv4-group-address";
            description
              "IPv4 source and group address of private network entry";
            leaf ipv4-source-address {
              type inet:ipv4-address;
              description
                "IPv4 source address of private network entry
                 in I-PMSI or S-PMSI.";
            }
            leaf ipv4-group-address {
              type inet:ipv4-address;
              description
                "IPv4 group address of private network entry
                 in I-PMSI or S-PMSI.";
            }
          }
        }
      }


      grouping mvpn-pmsi-ipv6-entry {
        description
          "Multicast entries in ipv6 mvpn referenced the pmsi tunnel";
        container mvpn-pmsi-ipv6-ref-sg-entries {
          description
            "Multicast entries in ipv6 mvpn referenced the pmsi tunnel";
          list mvpn-pmsi-ipv6-ref-sg-entries {
            key "ipv6-source-address ipv6-group-address";
```

```
            description
              "IPv6 source and group address of private network entry";
            leaf ipv6-source-address {
              type inet:ipv6-address;
              description
                "IPv6 source address of private network entry
                 in I-PMSI or S-PMSI.";
            }
            leaf ipv6-group-address {
              type inet:ipv6-address;
              description
                "IPv6 group address of private network entry
                 in I-PMSI or S-PMSI.";
            }
          }
        }
      }

      grouping mvpn-ipmsi-tunnel-state-ipv4 {
        description
          "Default mdt or I-PMSI operational state information";
        container mvpn-ipmsi-tunnel-info {
          config false;
          description
            "Default mdt or I-PMSI operational state information";
          uses mvpn-pmsi-state;
          uses mvpn-pmsi-ipv4-entry;
        }
      }

      grouping mvpn-ipmsi-tunnel-state-ipv6 {
        description
          "Default mdt or I-PMSI operational state information";
        container mvpn-ipmsi-tunnel-info {
          config false;
          description
            "Default mdt or I-PMSI operational state information";
          uses mvpn-pmsi-state;
          uses mvpn-pmsi-ipv6-entry;
        }
      }

      grouping mvpn-spmsi-tunnel-state-ipv4 {
        description
          "Data mdt or S-PMSI operational state information";
        container mvpn-spmsi-tunnel-ipv4-info {
          config false;
          description
            "Data mdt or S-PMSI operational state information";
```

```
          list mvpn-spmsi-tunnel-ipv4-info {
            key "tunnel-type";
            description
              "Data mdt or S-PMSI operational state information";
            uses mvpn-pmsi-state;
            uses mvpn-pmsi-ipv4-entry;
          }
        }
      }

      grouping mvpn-spmsi-tunnel-state-ipv6 {
        description
          "Data mdt or S-PMSI operational state information";
        container mvpn-spmsi-tunnel-ipv6-info {
          config false;
          description
            "Data mdt or S-PMSI operational state information";
          list mvpn-spmsi-tunnel-ipv6-info {
            key "tunnel-type";
            description
              "Data mdt or S-PMSI operational state information";
            uses mvpn-pmsi-state;
            uses mvpn-pmsi-ipv6-entry;
          }
        }
      }

      augment "/ni:network-instances/ni:network-instance" {
        description
          "Augment network instance container for per multicast VRF
           configuration and operational state.";
        container mvpn {
          description
            "Mvpn configuration and operational state information.";
          container mvpnv4 {
            description
              "Configuration of multicast IPv4 vpn specific parameters and
               operational state of multicast IPv4 vpn specific parameters";
            uses mvpn-instance-config;
            uses mvpn-vpn-targets;
            uses mvpn-ipmsi-tunnel-config;
            uses mvpn-spmsi-tunnel-config-ipv4;
            uses mvpn-ipmsi-tunnel-state-ipv4;
            uses mvpn-spmsi-tunnel-state-ipv4;
          }
          container mvpnv6 {
            description
              "Configuration of multicast IPv6 vpn specific parameters and
               operational state of multicast IPv6 vpn specific parameters";
```

```
            uses mvpn-instance-config;
            uses mvpn-vpn-targets;
            uses mvpn-ipmsi-tunnel-config;
            uses mvpn-spmsi-tunnel-config-ipv6;
            uses mvpn-ipmsi-tunnel-state-ipv6;
            uses mvpn-spmsi-tunnel-state-ipv6;
          }
        }
      }
   }
   <CODE ENDS>
```

5. Security Considerations

The data model defined does not introduce any security implications. This draft does not change any underlying security issues inherent in [RFC8022].

6. IANA Considerations

TBD

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

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

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

[I-D.ietf-netmod-rfc6087bis] Bierman, A., "Guidelines for Authors and Reviewers of YANG Data Model Documents", draft-ietf-netmod-rfc6087bis-14, September 2017.

[I-D.dsdt-nmda-guidelines] M. Bjorklund, J. Schoenwaelder, P. Shafer, K. Watsen, R. Wilton, "Guidelines for YANG Module Authors (NMDA)", draft-dsdt-nmda-guidelines-01, May 2017

7.2. Informative References

[RFC6037] Rosen, E., Cai, Y., and IJ. Wijnands, "Cisco Systems' Solution for Multicast in BGP/MPLS IP VPNs", RFC 6037, October 2010.

[RFC6513] Rosen, E. and R. Aggarwal, "Multicast in MPLS/BGP IP VPNs", RFC 6513, February 2012.

[RFC6514] Aggarwal, R., Rosen, E., Morin, T., and Y. Rekhter, "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, February 2012.

[RFC7246] IJ. Wijnands, P. Hitchen, N. Leymann, W. Henderickx, A. Gulko and J. Tantsura, " Multipoint Label Distribution Protocol In-Band Signaling in a Virtual Routing and Forwarding (VRF) Table Context ", RFC 7246, June 2014.

[RFC7900] Y. Rekhter, E. Rosen, R. Aggarwal, Arktan, Y. Cai and T. Morin, " Extranet Multicast in BGP/IP MPLS VPNs ", RFC 7900, June 2016.

[I-D.ietf-rtgwg-ni-model] Berger, L., Hopps, C., Lindem, A., and D. Bogdanovic, X. Liu, "Network Instance Model", draft-ietf-rtgwg-ni-model-03, July 2017.

[I-D.ietf-bess-l3vpn-yang] D. Jain, K. Patel, P. Brissette, Z. Li, S. Zhuang, X. Liu, J. Haas, S. Esale and B. Wen, "Yang Data Model for BGP/MPLS L3 VPNs", draft-ietf-bess-l3vpn-yang-01, April 2017.

8. Acknowledgments

The authors would like to thank Anish Peter, Stig Venaas for their valuable contributions.

Authors' Addresses

Yisong Liu
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: liuyisong@huawei.com

Feng Guo
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: guofeng@huawei.com

Xufeng Liu
Jabil
8281 Greensboro Drive, Suite 200
McLean VA 22102
USA

Email: Xufeng_Liu@jabil.com

Robert Kebler
Juniper Networks
10 Technology Park Drive
Westford, MA 01886
USA

Email: rkebler@juniper.net

Mahesh Sivakumar
Cisco Systems, Inc
510 McCarthy Blvd
Milpitas, California 95035
USA

Email: masivaku@cisco.com