opsawg
Internet-Draft
Intended status: Standards Track
Expires: November 20, 2021

S. Barguil
O. Gonzalez de Dios, Ed.
Telefonica
M. Boucadair, Ed.
Orange
Q. Wu
Huawei

May 19, 2021

A Layer 2/3 VPN Common YANG Model
draft-ietf-opsawg-vpn-common-08

Abstract

This document defines a common YANG module that is meant to be reused by various VPN-related modules such as Layer 3 VPN and Layer 2 VPN network models.

Editorial Note (To be removed by RFC Editor)

Please update these statements within the document with the RFC number to be assigned to this document:

- "This version of this YANG module is part of RFC XXXX;"
- "RFC XXXX: A Layer 2/3 VPN Common YANG Model";
- reference: RFC XXXX

Also, please update the "revision" date of the YANG module.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 20, 2021.

Table of Contents

1. Introduction 2
2. Terminology 3
3. Description of the VPN Common YANG Module 3
4. Layer 2/3 VPN Common Module 12
5. Security Considerations 57
6. IANA Considerations 57
7. Acknowledgements 58
8. Contributors 58
9. References 58
9.1. Normative References 58
9.2. Informative References 59
Appendix A. Example of Common Data Nodes in Early L2NM/L3NM Designs 65
Authors' Addresses 67

1. Introduction

The IETF has specified YANG data modules for VPN services, e.g., Layer 3 VPN Service Model (L3SM) [RFC8299] or Layer 2 VPN Service Model (L2SM) [RFC8466]. Other relevant YANG models are the Layer 3 VPN Network Model (L3NM) [I-D.ietf-opsawg-l3sm-l3nm] and the Layer 2 VPN Network Model (L2NM) [I-D.ietf-opsawg-l2nm]. There are common data nodes and structures that are present in all of these models or at least a subset of them.

This document defines a common YANG module that is meant to be reused by various VPN-related modules such as L3NM [I-D.ietf-opsawg-l3sm-l3nm] and L2NM [I-D.ietf-opsawg-l2nm]: "ietf-vpn-common" (Section 4).

The "ietf-vpn-common" module includes a set of identities, types, and groupings that are meant to be reused by other VPN-related YANG modules independently of their layer (e.g., Layer 2, Layer 3) and the type of the module (e.g., network model, service model) including possible future revisions of existing models (e.g., L3SM [RFC8299] or L2SM [RFC8466]).

2. Terminology

The terminology for describing YANG modules is defined in [RFC7950].

The meaning of the symbols in tree diagrams is defined in [RFC8340].

The reader may refer to [RFC4026] and [RFC4176] for VPN-related terms.

3. Description of the VPN Common YANG Module

The "ietf-vpn-common" module defines a set of common VPN-related features, including:

Encapsulation features such as Dot1q [IEEE802.1Q], QinQ [IEEE802.1ad], link aggregation [IEEE802.1AX], and Virtual eXtensible Local Area Network (VXLAN) [RFC7348].

Multicast [RFC6513].

Routing features such as BGP [RFC4271], OSPF [RFC4577][RFC6565], IS-IS [ISO10589], RIP [RFC2080][RFC2453], Bidirectional Forwarding Detection (BFD) [RFC5880], and Virtual Router Redundancy Protocol (VRRP) [RFC5798].

Also, the module defines a set of identities, including:

'service-type': Used to identify the VPN service type. Examples of supported service types are L3VPN, Virtual Private LAN Service (VPLS) using BGP [RFC4761], VPLS using Label Distribution Protocol (LDP) [RFC4762], Virtual Private Wire Service (VPWS) [RFC8214], BGP MPLS-Based Ethernet VPN [RFC7432], Ethernet VPN (EVPN) [RFC8365], and Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN) [RFC7623].

'vpn-signaling-type': Used to identify the signalling mode used for a given service type. Examples of supported VPN signaling types are L2VPNs using BGP [RFC6624], LDP signalling [RFC5036], and Layer Two Tunneling Protocol (L2TP) [RFC3931].

The module covers both IPv4 and IPv6 identities. It also includes multicast related identities such as Internet Group Management Protocol version 1 (IGMPv1) [RFC1112], IGMPv2 [RFC2236], IGMPv3 [RFC3376], Multicast Listener Discovery version 1 (MLDv1) [RFC2710], MLDv2 [RFC3810], and Protocol Independent Multicast (PIM) [RFC7761].

The reader should refer to Section 4 for the full list of supported identities (identities related to address families, VPN topologies, network access types, operational and administrative status, site or node roles, VPN service constraints, routing protocols, routes imports and exports, bandwidth and Quality of Service (QoS), etc.).

The "ietf-vpn-common" module also contains a set of reusable VPN-related groupings. The tree diagram of the "ietf-vpn-common" module that depicts the common groupings is provided in Figure 1.

```
module: ietf-vpn-common

  grouping vpn-description
    +-- vpn-id?            vpn-id
    +-- vpn-name?          string
    +-- vpn-description?   string
    +-- customer-name?     string
  grouping vpn-profile-cfg
    +-- valid-provider-identifiers
       +-- external-connectivity-identifier* [id]
       |       {external-connectivity}?
       |  +-- id?   string
       +-- encryption-profile-identifier* [id]
       |  +-- id?   string
       +-- qos-profile-identifier* [id]
       |  +-- id?   string
       +-- bfd-profile-identifier* [id]
       |  +-- id?   string
       +-- forwarding-profile-identifier* [id]
       |  +-- id?   string
       +-- routing-profile-identifier* [id]
          +-- id?   string
  grouping status-timestamp
    +--ro status?         identityref
    +--ro last-updated?   yang:date-and-time
  grouping service-status
    +-- status
       +-- admin-status
       |  +-- status?         identityref
       |  +-- last-updated?   yang:date-and-time
       +-- oper-status
          +--ro status?         identityref
```

```
            +--ro last-updated?   yang:date-and-time
        grouping underlay-transport
          +-- (type)?
             +--:(abstract)
             |  +-- transport-instance-id?   string
             +--:(protocol)
                +-- protocol*               identityref
        grouping vpn-route-targets
          +-- vpn-target* [id]
          |  +-- id?                  int8
          |  +-- route-targets* [route-target]
          |  |  +-- route-target?   rt-types:route-target
          |  +-- route-target-type    rt-types:route-target-type
          +-- vpn-policies
             +-- import-policy?   string
             +-- export-policy?   string
        grouping route-distinguisher
          ...
        grouping vpn-components-group
          +-- groups
             +-- group* [group-id]
                +-- group-id?   string
        grouping placement-constraints
          +-- constraint* [constraint-type]
             +-- constraint-type?   identityref
             +-- target
                +-- (target-flavor)?
                   +--:(id)
                   |  +-- group* [group-id]
                   |     +-- group-id?   string
                   +--:(all-accesses)
                   |  +-- all-other-accesses?   empty
                   +--:(all-groups)
                      +-- all-other-groups?     empty
        grouping ports
          ...
        grouping qos-classification-policy
          ...
```

Figure 1: VPN Common Tree

The description of the common groupings is provided below:

'vpn-description':

A YANG grouping that provides common administrative VPN information such as an identifier, a name, a textual description, and a customer name.

'vpn-profile-cfg':

A YANG grouping that defines a set of valid profiles (encryption, routing, forwarding, etc.) that can be bound to a Layer 2/3 VPN. This document does not make any assumption about the structure of such profiles, but allows "gluing" a VPN service with other parameters that can be required locally to provide added value features to requesting customers.

For example, a service provider may provide an external connectivity to a VPN customer (e.g., to a private or public cloud, Internet). Such service may involve tweaking both filtering and NAT rules (e.g., bind a Virtual Routing and Forwarding (VRF) interface with a NAT instance as discussed in Section 2.10 of [RFC8512]). These added value features may be bound to all or a subset of network accesses. Some of these added value features may be implemented in nodes other than PEs (e.g., a P node or even a dedicated node that hosts the NAT function).

It is out of the scope of this document to elaborate the structure of these profiles.

'status-timestamp':

A YANG grouping that defines the operational status updates of a VPN service or component.

'service-status':

A YANG grouping that defines the administrative and operational status of a component. The grouping can be applied to the whole service or an endpoint.

'underlay-transport':

A YANG grouping that defines the type of the underlay transport for a VPN service.

The underlay transport can be expressed as an abstract transport instance (e.g., an identifier of a VPN+ instance [I-D.ietf-teas-enhanced-vpn], a virtual network identifier [I-D.ietf-teas-actn-vn-yang][RFC8453], or a network slice name [I-D.ietf-teas-ietf-network-slice-framework]) or as an ordered list of the actual protocols to be enabled in the network.

The module supports a rich set of protocol identifiers that can be used, e.g., to refer to an underlay transport. Examples of

supported protocols are IP-in-IP [RFC2003][RFC2473], GRE [RFC1701][RFC1702][RFC7676], MPLS-in-UDP [RFC7510], Generic Network Virtualization Encapsulation (GENEVE) [RFC8926], Segment Routing (SR) [RFC8660][RFC8663][RFC8754], Resource ReSerVation Protocol (RSVP) with traffic engineering extensions [RFC3209], and BGP with labeled prefixes [RFC8277].

'vpn-route-targets':

A YANG grouping that defines Route Target (RT) import/export rules used in a BGP-enabled VPN (e.g., [RFC4364][RFC4664]).

'route-distinguisher':

A YANG grouping that defines Route Distinguishers (RDs).

As depicted in Figure 2, the module supports these RD assignment modes: direct assignment, automatic assignment from a given pool, automatic assignment, and no assignment.

Also, the module accommodates deployments where only the Assigned Number subfield of RDs (Section 4.2 of [RFC4364]) is assigned from a pool while the Administrator subfield is set to, e.g., the router-id that is assigned to a VPN node. The module supports these modes for managing the Assigned Number subfield: explicit assignment, auto-assignment from a pool, and full auto-assignment.

```
   grouping route-distinguisher
     +-- (rd-choice)?
        +--:(directly-assigned)
        |  +-- rd?               rt-types:route-distinguisher
        +--:(directly-assigned-suffix)
        |  +-- rd-suffix?        uint16
        +--:(auto-assigned)
        |  +-- rd-auto
        |     +-- (auto-mode)?
        |     |  +--:(from-pool)
        |     |  |  +-- rd-pool-name?   string
        |     |  +--:(full-auto)
        |     |     +-- auto?           empty
        |     +--ro auto-assigned-rd?   rt-types:route-distinguisher
        +--:(auto-assigned-suffix)
        |  +-- rd-auto-suffix
        |     +-- (auto-mode)?
        |     |  +--:(from-pool)
        |     |  |  +-- rd-pool-name?        string
        |     |  +--:(full-auto)
        |     |     +-- auto?                empty
        |     +--ro auto-assigned-rd-suffix?   uint16
        +--:(no-rd)
           +-- no-rd?           empty
```

Figure 2: Route Distinguisher Grouping Subtree

'vpn-components-group':

A YANG grouping that is used to group VPN nodes, VPN network accesses, or sites. For example, diversity or redundancy constraints can be applied on a per group basis.

'placement-constraints':

A YANG grouping that is used to define the placement constraints of a VPN node, VPN network access, or site.

'ports':

A YANG grouping that defines ranges of source and destination port numbers and operators. The subtree of this grouping is depicted in Figure 3.

```
            grouping ports
              +-- (source-port)?
              |  +--:(source-port-range-or-operator)
              |     +-- source-port-range-or-operator
              |        +-- (port-range-or-operator)?
              |           +--:(range)
              |           |  +-- lower-port    inet:port-number
              |           |  +-- upper-port    inet:port-number
              |           +--:(operator)
              |              +-- operator?     operator
              |              +-- port          inet:port-number
              +-- (destination-port)?
                 +--:(destination-port-range-or-operator)
                    +-- destination-port-range-or-operator
                       +-- (port-range-or-operator)?
                          +--:(range)
                          |  +-- lower-port    inet:port-number
                          |  +-- upper-port    inet:port-number
                          +--:(operator)
                             +-- operator?     operator
                             +-- port          inet:port-number
```

Figure 3: Port Numbers Grouping Subtree

'qos-classification-policy':

A YANG grouping that defines a set of QoS classification policies based on various match Layer 3/4 and application criteria. The subtree of this grouping is depicted in Figure 4.

Any layer 4 protocol can be indicated in the 'protocol' data node under 'l3', but only TCP and UDP specific match criteria are elaborated in this version as these protocols are widely used in the context of VPN services. Augmentations can be considered in the future to add other Layer 4 specific data nodes (e.g., Stream Control Transmission Protocol [RFC4960]), if needed.

```
  grouping qos-classification-policy
    +-- rule* [id]
       +-- id?                        string
       +-- (match-type)?
       |  +--:(match-flow)
       |  |  +-- (l3)?
       |  |  |  +--:(ipv4)
       |  |  |  |  +-- ipv4
       |  |  |  |     +-- dscp?                          inet:dscp
```

```
|  |  |  |     +-- ecn?                              uint8
|  |  |  |     +-- length?                           uint16
|  |  |  |     +-- ttl?                              uint8
|  |  |  |     +-- protocol?                         uint8
|  |  |  |     +-- ihl?                              uint8
|  |  |  |     +-- flags?                            bits
|  |  |  |     +-- offset?                           uint16
|  |  |  |     +-- identification?                   uint16
|  |  |  |     +-- (destination-network)?
|  |  |  |     |  +--:(destination-ipv4-network)
|  |  |  |     |     +-- destination-ipv4-network?
|  |  |  |     |            inet:ipv4-prefix
|  |  |  |     +-- (source-network)?
|  |  |  |        +--:(source-ipv4-network)
|  |  |  |           +-- source-ipv4-network?
|  |  |  |                  inet:ipv4-prefix
|  |  |  +--:(ipv6)
|  |  |     +-- ipv6
|  |  |        +-- dscp?                           inet:dscp
|  |  |        +-- ecn?                            uint8
|  |  |        +-- length?                         uint16
|  |  |        +-- ttl?                            uint8
|  |  |        +-- protocol?                       uint8
|  |  |        +-- (destination-network)?
|  |  |        |  +--:(destination-ipv6-network)
|  |  |        |     +-- destination-ipv6-network?
|  |  |        |            inet:ipv6-prefix
|  |  |        +-- (source-network)?
|  |  |        |  +--:(source-ipv6-network)
|  |  |        |     +-- source-ipv6-network?
|  |  |        |            inet:ipv6-prefix
|  |  |        +-- flow-label?
|  |  |               inet:ipv6-flow-label
|  |  +-- (l4)?
|  |     +--:(tcp)
|  |     |  +-- tcp
|  |     |     +-- sequence-number?                uint32
|  |     |     +-- acknowledgement-number?         uint32
|  |     |     +-- data-offset?                    uint8
|  |     |     +-- reserved?                       uint8
|  |     |     +-- flags?                          bits
|  |     |     +-- window-size?                    uint16
|  |     |     +-- urgent-pointer?                 uint16
|  |     |     +-- options?                        binary
|  |     |     +-- (source-port)?
|  |     |     |  +--:(source-port-range-or-operator)
|  |     |     |     +-- source-port-range-or-operator
|  |     |     |        +-- (port-range-or-operator)?
```

```
|  |     |     |           +--:(range)
|  |     |     |           |  +-- lower-port
|  |     |     |           |  |       inet:port-number
|  |     |     |           |  +-- upper-port
|  |     |     |           |          inet:port-number
|  |     |     |           +--:(operator)
|  |     |     |              +-- operator?     operator
|  |     |     |              +-- port
|  |     |     |                      inet:port-number
|  |     |     +-- (destination-port)?
|  |     |        +--:(destination-port-range-or-operator)
|  |     |           +-- destination-port-range-or-operator
|  |     |              +-- (port-range-or-operator)?
|  |     |                 +--:(range)
|  |     |                 |  +-- lower-port
|  |     |                 |  |       inet:port-number
|  |     |                 |  +-- upper-port
|  |     |                 |          inet:port-number
|  |     |                 +--:(operator)
|  |     |                    +-- operator?     operator
|  |     |                    +-- port
|  |     |                            inet:port-number
|  |     +--:(udp)
|  |        +-- udp
|  |           +-- length?                        uint16
|  |           +-- (source-port)?
|  |           |  +--:(source-port-range-or-operator)
|  |           |     +-- source-port-range-or-operator
|  |           |        +-- (port-range-or-operator)?
|  |           |           +--:(range)
|  |           |           |  +-- lower-port
|  |           |           |  |       inet:port-number
|  |           |           |  +-- upper-port
|  |           |           |          inet:port-number
|  |           |           +--:(operator)
|  |           |              +-- operator?     operator
|  |           |              +-- port
|  |           |                      inet:port-number
|  |           +-- (destination-port)?
|  |              +--:(destination-port-range-or-operator)
|  |                 +-- destination-port-range-or-operator
|  |                    +-- (port-range-or-operator)?
|  |                       +--:(range)
|  |                       |  +-- lower-port
|  |                       |  |       inet:port-number
|  |                       |  +-- upper-port
|  |                       |          inet:port-number
|  |                       +--:(operator)
```

```
          |  |                          +-- operator?     operator
          |  |                          +-- port
          |  |                                  inet:port-number
          |  +--:(match-application)
          |     +-- match-application?   identityref
          +-- target-class-id?           string {qos}?
```

Figure 4: QoS Classification Subtree

4. Layer 2/3 VPN Common Module

This module uses types defined in [RFC6991], [RFC8294], and [RFC8519]. It also uses the extension defined in [RFC8341].

```
<CODE BEGINS>  file "ietf-vpn-common@2021-05-18.yang"
module ietf-vpn-common {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-vpn-common";
  prefix vpn-common;

  import ietf-netconf-acm {
    prefix nacm;
    reference
      "RFC 8341: Network Configuration Access Control Model";
  }
  import ietf-routing-types {
    prefix rt-types;
    reference
      "RFC 8294: Common YANG Data Types for the Routing Area";
  }
  import ietf-yang-types {
    prefix yang;
    reference
      "RFC 6991: Common YANG Data Types, Section 3";
  }
  import ietf-packet-fields {
    prefix packet-fields;
    reference
      "RFC 8519: YANG Data Model for Network Access
                 Control Lists (ACLs)";
  }

  organization
    "IETF OPSA (Operations and Management Area) Working Group";
  contact
    "WG Web:   <https://datatracker.ietf.org/wg/opsawg/>
     WG List:  <mailto:opsawg@ietf.org>
```

```
     Author:  Samier Barguil
              <mailto:samier.barguilgiraldo.ext@telefonica.com>
     Author:  Oscar Gonzalez de Dios
              <mailto:oscar.gonzalezdedios@telefonica.com>
     Editor:  Mohamed Boucadair
              <mailto:mohamed.boucadair@orange.com>
     Author:  Qin Wu
              <mailto:bill.wu@huawei.com>";
  description
    "This YANG module defines a common module that is meant
     to be reused by various VPN-related modules (e.g.,
     Layer 3 VPN Service Model (L3SM), Layer 2 VPN Service
     Model (L2SM), Layer 3 VPN Network Model (L3NM), Layer 2
     VPN Network Model (L2NM)).

     Copyright (c) 2021 IETF Trust and the persons identified as
     authors of the code.  All rights reserved.

     Redistribution and use in source and binary forms, with or
     without modification, is permitted pursuant to, and subject
     to the license terms contained in, the Simplified BSD License
     set forth in Section 4.c of the IETF Trust's Legal Provisions
     Relating to IETF Documents
     (http://trustee.ietf.org/license-info).

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

  revision 2021-05-18 {
    description
      "Initial revision.";
    reference
      "RFC XXXX: A Layer 2/3 VPN Common YANG Model";
  }

  /******** Collection of VPN-related Features ********/
  /*
   * Features related to encapsulation schemes
   */

  feature dot1q {
    description
      "Indicates the support of the Dot1q encapsulation.";
    reference
      "IEEE Std 802.1Q: Bridges and Bridged Networks";
  }

  feature qinq {
```

```
    description
      "Indicates the support of the QinQ encapsulation.";
    reference
      "IEEE Std 802.1ad: Provider Bridges";
  }

  feature vxlan {
    description
      "Indicates the support of the Virtual eXtensible
       Local Area Network (VXLAN) encapsulation.";
    reference
      "RFC 7348: Virtual eXtensible Local Area  Network (VXLAN):
                 A Framework for Overlaying Virtualized Layer 2
                 Networks over Layer 3 Networks";
  }

  feature qinany {
    description
      "Indicates the support of the QinAny encapsulation.";
  }

  feature lag-interface {
    description
      "Indicates the support of Link Aggregation Group (LAG)
       between VPN network accesses.";
    reference
      "IEEE Std. 802.1AX: Link Aggregation";
  }

  /*
   * Features related to multicast
   */

  feature multicast {
    description
      "Indicates multicast capabilities support in a VPN.";
    reference
      "RFC 6513: Multicast in MPLS/BGP IP VPNs";
  }

  feature igmp {
    description
      "Indicates support of Internet Group Management Protocol
       (IGMP).";
    reference
      "RFC 1112: Host Extensions for IP Multicasting
       RFC 2236: Internet Group Management Protocol, Version 2
       RFC 3376: Internet Group Management Protocol, Version 3";
```

```
  }

  feature mld {
    description
      "Indicates support of Multicast Listener Discovery (MLD).";
    reference
      "RFC 2710: Multicast Listener Discovery (MLD) for IPv6
       RFC 3810: Multicast Listener Discovery Version 2 (MLDv2)
                 for IPv6";
  }

  feature pim {
    description
      "Indicates support of Protocol Independent Multicast (PIM).";
    reference
      "RFC 7761: Protocol Independent Multicast - Sparse Mode
                (PIM-SM): Protocol Specification (Revised)";
  }

  /*
   * Features related to address family types
   */

  feature ipv4 {
    description
      "Indicates IPv4 support in a VPN. That is, IPv4 traffic
       can be conveyed in the VPN, IPv4 addresses/prefixes can
       be assigned to an access, IPv4 routes can be installed
       for the CE/PE link, etc.";
  }

  feature ipv6 {
    description
      "Indicates IPv6 support in a VPN. That is, IPv6 traffic
       can be conveyed in the VPN, IPv6 addresses/prefixes can
       be assigned to an access, IPv6 routes can be installed
       for the CE/PE link, etc.";
  }

  /*
   * Features related to routing protocols
   */

  feature rtg-ospf {
    description
      "Indicates support of the OSPF as the Provider Edge (PE)/
       Customer Edge (CE) routing protocol.";
    reference
```

```
      "RFC 4577: OSPF as the Provider/Customer Edge Protocol
                 for BGP/MPLS IP Virtual Private Networks (VPNs)
       RFC 6565: OSPFv3 as a Provider Edge to Customer Edge
                 (PE-CE) Routing Protocol";
  }

  feature rtg-ospf-sham-link {
    description
      "Indicates support of OSPF sham links.";
    reference
      "RFC 4577: OSPF as the Provider/Customer Edge Protocol
                 for BGP/MPLS IP Virtual Private Networks (VPNs),
                 Section 4.2.7
       RFC 6565: OSPFv3 as a Provider Edge to Customer Edge
                 (PE-CE) Routing Protocol, Section 5";
  }

  feature rtg-bgp {
    description
      "Indicates support of BGP as the PE/CE routing protocol.";
    reference
      "RFC 4271: A Border Gateway Protocol 4 (BGP-4)";
  }

  feature rtg-rip {
    description
      "Indicates support of RIP as the PE/CE routing protocol.";
    reference
      "RFC 2453: RIP Version 2
       RFC 2080: RIPng for IPv6";
  }

  feature rtg-isis {
    description
      "Indicates support of IS-IS as the PE/CE routing protocol.";
    reference
      "ISO10589: Intermediate System to Intermediate System intra-
                 domain routeing information exchange protocol for
                 use in conjunction with the protocol for providing
                 the connectionless-mode network service
                 (ISO 8473)";
  }

  feature rtg-vrrp {
    description
      "Indicates support of the Virtual Router Redundancy
       Protocol (VRRP) in CE/PE link.";
    reference
```

```
      "RFC 5798: Virtual Router Redundancy Protocol (VRRP) Version 3
                 for IPv4 and IPv6";
  }

  feature bfd {
    description
      "Indicates support of Bidirectional Forwarding Detection (BFD)
       between the CE and the PE.";
    reference
      "RFC 5880: Bidirectional Forwarding Detection (BFD)";
  }

  /*
   * Features related to VPN service constraints
   */

  feature bearer-reference {
    description
      "Indicates support of the bearer reference access constraint.
       That is, the reuse of a network connection that was already
       ordered to the service provider apart from the IP VPN site.";
  }

  feature placement-diversity {
    description
      "Indicates support of placement diversity constraints in the
       customer premises. An example of these constraints may be to
       avoid connecting a site network access to the same Provider
       Edge as a target site network access.";
  }

  /*
   * Features related to bandwidth and Quality of Service (QoS)
   */

  feature qos {
    description
      "Indicates support of Classes of Service (CoSes) in the VPN.";
  }

  feature input-bw {
    description
      "Indicates support of the input bandwidth in a VPN. That is,
       support of specifying the download bandwidth from the service
       provider network to the VPN site.";
  }

  feature output-bw {
```

```
    description
      "Indicates support of the output bandwidth in a VPN. That is,
       support of specifying the upload bandwidth from the VPN site
       to the service provider network.";
  }

  /*
   * Features related to security and resilience
   */

  feature encryption {
    description
      "Indicates support of encryption in the VPN.";
  }

  feature fast-reroute {
    description
      "Indicates support of Fast Reroute (FRR) capabilities for
       a VPN site.";
  }

  /*
   * Features related to advanced VPN options
   */

  feature external-connectivity {
    description
      "Indicates support of the VPN to provide external
       connectivity (e.g., Internet, private or public cloud).";
    reference
      "RFC 4364: BGP/MPLS IP Virtual Private Networks
                 (VPNs), Section 11";
  }

  feature extranet-vpn {
    description
      "Indicates support of extranet VPNs. That is, the capability of
       a VPN to access a list of other VPNs.";
    reference
      "RFC 4364: BGP/MPLS IP Virtual Private Networks
                 (VPNs), Section 1.1";
  }

  feature carrierscarrier {
    description
      "Indicates support of Carrier-of-Carrier VPNs.";
    reference
      "RFC 4364: BGP/MPLS IP Virtual Private Networks
```

```
                (VPNs), Section 9";
  }

  /*
   * Address family related identities
   */

  identity address-family {
    description
      "Defines a type for the address family.";
  }

  identity ipv4 {
    base address-family;
    description
      "Identity for IPv4 address family.";
  }

  identity ipv6 {
    base address-family;
    description
      "Identity for IPv6 address family.";
  }

  identity dual-stack {
    base address-family;
    description
      "Identity for IPv4 and IPv6 address family.";
  }

  /*
   * Identities related to VPN topology
   */

  identity vpn-topology {
    description
      "Base identity of the VPN topology.";
  }

  identity any-to-any {
    base vpn-topology;
    description
      "Identity for any-to-any VPN topology. All VPN sites
       can communicate with each other without any restrictions.";
  }

  identity hub-spoke {
    base vpn-topology;
```

```
    description
      "Identity for Hub-and-Spoke VPN topology. All Spokes can
       communicate only with Hubs but not with each other. Hubs
       can communicate with each other.";
  }

  identity hub-spoke-disjoint {
    base vpn-topology;
    description
      "Identity for Hub-and-Spoke VPN topology where Hubs cannot
       communicate with each other.";
  }

  identity custom {
    base vpn-topology;
    description
      "Identity for custom VPN topologies where the role of the nodes
       is not strictly Hub or Spoke. The VPN topology is controlled by
       the import/export policies. The custom topology reflects more
       complex VPN nodes such as VPN node that acts as Hub for certain
       nodes and Spoke to others.";
  }

  /*
   * Identities related to network access types
   */

  identity site-network-access-type {
    description
      "Base identity for site network access type.";
  }

  identity point-to-point {
    base site-network-access-type;
    description
      "Identity for point-to-point access type.";
  }

  identity multipoint {
    base site-network-access-type;
    description
      "Identity for multipoint access type.";
  }

  identity irb {
    base site-network-access-type;
    description
      "Integrated Routing Bridge (IRB).
```

```
      Identity for pseudowire connections.";
  }

  identity loopback {
    base site-network-access-type;
    description
      "Identity for loopback access type.";
  }

  /*
   * Identities related to operational and administrative status
   */

  identity operational-status {
    description
      "Base identity for the operational status.";
  }

  identity op-up {
    base operational-status;
    description
      "Operational status is UP/Enabled.";
  }

  identity op-down {
    base operational-status;
    description
      "Operational status is DOWN/Disabled.";
  }

  identity op-unknown {
    base operational-status;
    description
      "Operational status is UNKNOWN.";
  }

  identity administrative-status {
    description
      "Base identity for administrative status.";
  }

  identity admin-up {
    base administrative-status;
    description
      "Administrative status is UP/Enabled.";
  }

  identity admin-down {
```

```
    base administrative-status;
    description
      "Administrative status is DOWN/Disabled.";
  }

  identity admin-testing {
    base administrative-status;
    description
      "Administrative status is up for testing purposes.";
  }

  identity admin-pre-deployment {
    base administrative-status;
    description
      "Administrative status is pre-deployment phase. That is,
       prior to the actual deployment of a service.";
  }

  /*
   * Identities related to site or node role
   */

  identity role {
    description
      "Base identity of a site or a node role.";
  }

  identity any-to-any-role {
    base role;
    description
      "Identity of any-to-any role.";
  }

  identity spoke-role {
    base role;
    description
      "A node or a site is acting as a Spoke.";
  }

  identity hub-role {
    base role;
    description
      "A node or a site is acting as a Hub.";
  }

  identity custom-role {
    base role;
    description
```

```
      "VPN node with custom or complex role in the VPN. For some
       sources/destinations it can behave as a Hub, but for others it
       can act as a Spoke depending on the configured policy.";
  }

  /*
   * Identities related to VPN service constraints
   */

  identity placement-diversity {
    description
      "Base identity for access placement constraints.";
  }

  identity bearer-diverse {
    base placement-diversity;
    description
      "Identity for bearer diversity.

       The bearers should not use common elements.";
  }

  identity pe-diverse {
    base placement-diversity;
    description
      "Identity for PE diversity.";
  }

  identity pop-diverse {
    base placement-diversity;
    description
      "Identity for Point Of Presence (POP) diversity.";
  }

  identity linecard-diverse {
    base placement-diversity;
    description
      "Identity for linecard diversity.";
  }

  identity same-pe {
    base placement-diversity;
    description
      "Identity for having sites connected on the same PE.";
  }

  identity same-bearer {
    base placement-diversity;
```

```
    description
      "Identity for having sites connected using the same bearer.";
  }

  /*
   * Identities related to service types
   */

  identity service-type {
    description
      "Identity of service type.";
  }

  identity l3vpn {
    base service-type;
    description
      "Identity for L3VPN service.";
    reference
      "RFC 4364: BGP/MPLS IP Virtual Private Networks (VPNs)";
  }

  identity vpls {
    base service-type;
    description
      "Identity for the VPLS service type.";
    reference
      "RFC 4761: Virtual Private LAN Service (VPLS) Using BGP for
                 Auto-Discovery and Signaling
       RFC 4762: Virtual Private LAN Service (VPLS) Using Label
                 Distribution Protocol (LDP) Signaling";
  }

  identity vpws-evpn {
    base service-type;
    description
      "Identity for the Point-to-point Virtual Private Wire Service
       (VPWS) service type.";
    reference
      "RFC 8214: Virtual Private Wire Service Support in Ethernet VPN";
  }

  identity pbb-evpn {
    base service-type;
    description
      "Identity for Provider Backbone Bridging (PBB) EVPNs.";
    reference
      "RFC 7623: Provider Backbone Bridging Combined with Ethernet VPN
                (PBB-EVPN)";
```

```
  }

  identity mpls-evpn {
    base service-type;
    description
      "Identity for MPLS based EVPNs.";
    reference
      "RFC 7432: BGP MPLS-Based Ethernet VPN";
  }

  identity vxlan-evpn {
    base service-type;
    description
      "Identity for VXLAN based EVPNs.";
    reference
      "RFC 8365: A Network Virtualization Overlay Solution Using
                 Ethernet VPN (EVPN)";
  }

  /*
   * Identities related to VPN signaling type
   */

  identity vpn-signaling-type {
    description
      "Identity for VPN signaling types";
  }

  identity bgp-signaling {
    base vpn-signaling-type;
    description
      "Identity for Layer 2 VPNs using BGP";
    reference
      "RFC 6624: Layer 2 Virtual Private Networks Using BGP for
                 Auto-Discovery and Signaling
       RFC 7432: BGP MPLS-Based Ethernet VPN";
  }

  identity ldp-signaling {
    base vpn-signaling-type;
    description
      "Identity for Targeted Label Distribution Protocol.";
    reference
      "RFC 5036: LDP Specification";
  }

  identity l2tp-signaling {
    base vpn-signaling-type;
```

```
    description
      "Identity for Layer Two Tunneling Protocol (L2TP).";
    reference
      "RFC 3931: Layer Two Tunneling Protocol - Version 3 (L2TPv3)";
  }

  /*
   * Identities related to routing protocols
   */

  identity routing-protocol-type {
    description
      "Base identity for routing protocol type.";
  }

  identity static {
    base routing-protocol-type;
    description
      "Identity for static routing protocol type.";
  }

  identity bgp {
    if-feature "rtg-bgp";
    base routing-protocol-type;
    description
      "Identity for BGP routing protocol type.";
    reference
      "RFC 4271: A Border Gateway Protocol 4 (BGP-4)";
  }

  identity ospf {
    if-feature "rtg-ospf";
    base routing-protocol-type;
    description
      "Identity for OSPF routing protocol type.";
    reference
      "RFC 4577: OSPF as the Provider/Customer Edge Protocol
                 for BGP/MPLS IP Virtual Private Networks(VPNs)
       RFC 6565: OSPFv3 as a Provider Edge to Customer Edge
                 (PE-CE) Routing Protocol";
  }

  identity rip {
    if-feature "rtg-rip";
    base routing-protocol-type;
    description
      "Identity for RIP routing protocol type.";
    reference
```

```
      "RFC 2453: RIP Version 2
       RFC 2080: RIPng for IPv6";
  }

  identity isis {
    if-feature "rtg-isis";
    base routing-protocol-type;
    description
      "Identity for IS-IS routing protocol type.";
    reference
      "ISO10589: Intermediate System to Intermediate System intra-
                 domain routeing information exchange protocol for
                 use in conjunction with the protocol for providing
                 the connectionless-mode network service
                 (ISO 8473)";
  }

  identity vrrp {
    if-feature "rtg-vrrp";
    base routing-protocol-type;
    description
      "Identity for VRRP protocol type.

       This is to be used when LANs are directly connected to PEs.";
    reference
      "RFC 5798: Virtual Router Redundancy Protocol (VRRP) Version 3
                 for IPv4 and IPv6";
  }

  identity direct {
    base routing-protocol-type;
    description
      "Identity for direct routing protocol type.

       This is to be used when LANs are directly connected to PEs
       and must be advertised in the VPN.";
  }

  identity any {
    base routing-protocol-type;
    description
      "Identity for any routing protocol type.

       This can be, e.g., used to set policies that apply to any
       routing protocol in place.";
  }

  identity isis-level {
```

```
    if-feature "rtg-isis";
    description
      "Identity for the IS-IS level.";
    reference
      "ISO10589: Intermediate System to Intermediate System intra-
                 domain routeing information exchange protocol for
                 use in conjunction with the protocol for providing
                 the connectionless-mode network service
                 (ISO 8473)";
  }

  identity level-1 {
    base isis-level;
    description
      "Identity for IS-IS level 1.";
  }

  identity level-2 {
    base isis-level;
    description
      "Identity for IS-IS level 2.";
  }

  identity level-1-2 {
    base isis-level;
    description
      "Identity for IS-IS levels 1 and 2.";
  }

  /*
   * Identities related to Routes Import and Export
   */

  identity ie-type {
    description
      "Identity for 'import/export' routing profiles. These profiles
       can be reused between VPN nodes.";
  }

  identity import {
    base ie-type;
    description
      "Identity for 'import' routing profile.";
    reference
      "RFC 4364: BGP/MPLS IP Virtual Private Networks
                 (VPNs), Section 4.3.1";
  }
```

```
  identity export {
    base ie-type;
    description
      "Identity for 'export' routing profile.";
    reference
      "RFC 4364: BGP/MPLS IP Virtual Private Networks
                 (VPNs), Section 4.3.1";
  }

  identity import-export {
    base ie-type;
    description
      "Identity for 'import/export' routing profile.";
  }

  /*
   * Identities related to bandwidth and QoS
   */

  identity bw-direction {
    description
      "Identity for the bandwidth direction.";
  }

  identity input-bw {
    if-feature "input-bw";
    base bw-direction;
    description
      "Identity for the input bandwidth.";
  }

  identity output-bw {
    if-feature "output-bw";
    base bw-direction;
    description
      "Identity for the output bandwidth.";
  }

  identity bw-type {
    description
      "Identity of the bandwidth type.";
  }

  identity bw-per-cos {
    if-feature "qos";
    base bw-type;
    description
      "The bandwidth is per CoS.";
```

```
  }

  identity bw-per-port {
    base bw-type;
    description
      "The bandwidth is per site network access.";
  }

  identity bw-per-site {
    base bw-type;
    description
      "The bandwidth is per site. It is applicable to all the site
       network accesses within a site.";
  }

  identity bw-per-service {
    base bw-type;
    description
      "The bandwidth is per VPN service.";
  }

  identity qos-profile-direction {
    if-feature "qos";
    description
      "Base identity for the QoS profile direction.";
  }

  identity site-to-wan {
    base qos-profile-direction;
    description
      "Identity for customer site to provider's network
       direction. This is typically the CE-to-PE direction.";
  }

  identity wan-to-site {
    base qos-profile-direction;
    description
      "Identity for provider's network to customer site
       direction. This is typically the PE-to-CE direction.";
  }

  identity both {
    base qos-profile-direction;
    description
      "Identity for both WAN-to-Site and Site-to-WAN directions.";
  }

  /*
```

```
   *  Identities related to underlay transport instances
   */

  identity transport-instance-type {
    description
      "Base identity for underlay transport instance type.";
  }

  identity virtual-network {
    base transport-instance-type;
    description
      "Identity for the virtual network.";
    reference
      "RFC 8453: Framework for Abstraction and Control of TE
                 Networks (ACTN)";
  }

  identity enhanced-vpn {
    base transport-instance-type;
    description
      "Identity for the Enhanced VPN (VPN+). VPN+ is an
       approach that is based on existing VPN and Traffic
       Engineering (TE) technologies but adds characteristics
       that specific services require over and above traditional
       VPNs.";
  }

  identity ietf-network-slice {
    base transport-instance-type;
    description
      "Identity for the IETF network slice. An IETF network slice
       is a logical network topology connecting a number of
       endpoints using a set of shared or dedicated network
       resources that are used to satisfy specific service
       objectives.";
  }

  /*
   *  Identities related to protocol types. These types are typically
   *  used to identify the underlay transport.
   */

  identity protocol-type {
    description
      "Base identity for Protocol Type.";
  }

  identity ip-in-ip {
```

```
    base protocol-type;
    description
      "Transport is based on IP-in-IP.";
    reference
      "RFC 2003: IP Encapsulation within IP
       RFC 2473: Generic Packet Tunneling in IPv6 Specification";
  }

  identity ip-in-ipv4 {
    base ip-in-ip;
    description
      "Transport is based on IP over IPv4.";
    reference
      "RFC 2003: IP Encapsulation within IP";
  }

  identity ip-in-ipv6 {
    base ip-in-ip;
    description
      "Transport is based on IP over IPv6.";
    reference
      "RFC 2473: Generic Packet Tunneling in IPv6 Specification";
  }

  identity gre {
    base protocol-type;
    description
      "Transport is based on Generic Routing Encapsulation (GRE).";
    reference
      "RFC 1701: Generic Routing Encapsulation (GRE)
       RFC 1702: Generic Routing Encapsulation over IPv4 networks
       RFC 7676: IPv6 Support for Generic Routing Encapsulation (GRE)";
  }

  identity gre-v4 {
    base gre;
    description
      "Transport is based on GRE over IPv4.";
    reference
      "RFC 1702: Generic Routing Encapsulation over IPv4 networks";
  }

  identity gre-v6 {
    base gre;
    description
      "Transport is based on GRE over IPv6.";
    reference
      "RFC 7676: IPv6 Support for Generic Routing Encapsulation (GRE)";
```

```
  }

  identity vxlan-trans {
    base protocol-type;
    description
      "Transport is based on VXLAN.";
    reference
      "RFC 7348: Virtual eXtensible Local Area  Network (VXLAN):
                 A Framework for Overlaying Virtualized Layer 2
                 Networks over Layer 3 Networks";
  }

  identity geneve {
    base protocol-type;
    description
      "Transport is based on Generic Network Virtualization
       Encapsulation (GENEVE).";
    reference
      "RFC 8926: Geneve: Generic Network Virtualization Encapsulation";
  }

  identity ldp {
    base protocol-type;
    description
      "Transport is based on LDP.";
    reference
      "RFC 5036: LDP Specification";
  }

  identity mpls-in-udp {
    base protocol-type;
    description
      "Transport is MPLS in UDP.";
    reference
      "RFC 7510: Encapsulating MPLS in UDP";
  }

  identity sr {
    base protocol-type;
    description
      "Transport is based on Segment Routing (SR).";
    reference
      "RFC 8660: Segment Routing with the MPLS Data Plane
       RFC 8663: MPLS Segment Routing over IP
       RFC 8754: IPv6 Segment Routing Header (SRH)";
  }

  identity sr-mpls {
```

```
    base sr;
    description
      "Transport is based on SR with MPLS.";
    reference
      "RFC 8660: Segment Routing with the MPLS Data Plane";
  }

  identity srv6 {
    base sr;
    description
      "Transport is based on SR over IPv6.";
    reference
      "RFC 8663: MPLS Segment Routing over IP
       RFC 8754: IPv6 Segment Routing Header (SRH)";
  }

  identity rsvp-te {
    base protocol-type;
    description
      "Transport is based on RSVP-TE.";
    reference
      "RFC 3209: RSVP-TE: Extensions to RSVP for LSP Tunnels";
  }

  identity bgp-lu {
    base protocol-type;
    description
      "Transport is based on BGP-LU.";
    reference
      "RFC 8277: Using BGP to Bind MPLS Labels to Address Prefixes";
  }

  identity unknown {
    base protocol-type;
    description
      "Not known protocol type.";
  }

  /*
   * Identities related to encapsulations
   */

  identity encapsulation-type {
    description
      "Base identity for the encapsulation type.";
  }

  identity priority-tagged {
```

```
    base encapsulation-type;
    description
      "Identity for the priority-tagged interface.";
  }

  identity dot1q {
    if-feature "dot1q";
    base encapsulation-type;
    description
      "Identity for the support of the Dot1q encapsulation.";
  }

  identity qinq {
    if-feature "qinq";
    base encapsulation-type;
    description
      "Identity for the support of the QinQ encapsulation.";
  }

  identity qinany {
    if-feature "qinany";
    base encapsulation-type;
    description
      "Identity for the support of the QinAny encapsulation.";
  }

  identity vxlan {
    if-feature "vxlan";
    base encapsulation-type;
    description
      "Identity for the support of the VxLAN encapsulation.";
  }

  identity ethernet-type {
    base encapsulation-type;
    description
      "Identity of the Ethernet encapsulation type.";
  }

  identity vlan-type {
    base encapsulation-type;
    description
      "Identity of the VLAN encapsulation.";
  }

  identity untagged-int {
    base encapsulation-type;
    description
```

```
      "Identity of the untagged interface type.";
  }

  identity tagged-int {
    base encapsulation-type;
    description
      "Identity of the tagged interface type.";
  }

  identity lag-int {
    if-feature "lag-interface";
    base encapsulation-type;
    description
      "Identity of the LAG interface type.";
    reference
      "IEEE Std. 802.1AX: Link Aggregation";
  }

  /*
   * Identities related to VLAN Tag
   */

  identity tag-type {
    description
      "Base identity of the tag types.";
  }

  identity c-vlan {
    base tag-type;
    description
      "Indicates Customer VLAN (C-VLAN) tag, normally using
       the 0x8100 Ethertype.";
  }

  identity s-vlan {
    base tag-type;
    description
      "Indicates Service VLAN (S-VLAN) tag.";
  }

  identity c-s-vlan {
    base tag-type;
    description
      "Uses both a C-VLAN tag and a S-VLAN tag.";
  }

  /*
   * Identities related to VXLAN
```

```
   */

  identity vxlan-peer-mode {
    if-feature "vxlan";
    description
      "Base identity for the VXLAN peer mode.";
  }

  identity static-mode {
    base vxlan-peer-mode;
    description
      "Identity for VXLAN access in the static mode.";
  }

  identity bgp-mode {
    base vxlan-peer-mode;
    description
      "Identity for VXLAN access by BGP EVPN learning.";
  }

  /*
   * Identities related to multicast
   */

  identity multicast-gp-address-mapping {
    if-feature "multicast";
    description
      "Identity for multicast group mapping type.";
  }

  identity static-mapping {
    base multicast-gp-address-mapping;
    description
      "Identity for static mapping, i.e., attach the interface to the
       multicast group as a static member.";
  }

  identity dynamic-mapping {
    base multicast-gp-address-mapping;
    description
      "Identity for dynamic mapping, i.e., an interface is added to the
       multicast group as a result of snooping.";
  }

  identity multicast-tree-type {
    if-feature "multicast";
    description
      "Base identity for multicast tree type.";
```

```
  }

  identity ssm-tree-type {
    base multicast-tree-type;
    description
      "Identity for Source-Specific Multicast (SSM) tree type.";
  }

  identity asm-tree-type {
    base multicast-tree-type;
    description
      "Identity for Any-Source Multicast (ASM) tree type.";
  }

  identity bidir-tree-type {
    base multicast-tree-type;
    description
      "Identity for bidirectional tree type.";
  }

  identity multicast-rp-discovery-type {
    if-feature "multicast";
    description
      "Base identity for Rendezvous Point (RP) discovery type.";
  }

  identity auto-rp {
    base multicast-rp-discovery-type;
    description
      "Base identity for Auto-RP discovery type.";
  }

  identity static-rp {
    base multicast-rp-discovery-type;
    description
      "Base identity for static type.";
  }

  identity bsr-rp {
    base multicast-rp-discovery-type;
    description
      "Base identity for Bootstrap Router (BSR) discovery type.";
  }

  identity group-management-protocol {
    if-feature "multicast";
    description
      "Identity for multicast group management protocol.";
```

```
  }

  identity igmp-proto {
    base group-management-protocol;
    description
      "Identity for IGMP.";
    reference
      "RFC 1112: Host Extensions for IP Multicasting
       RFC 2236: Internet Group Management Protocol, Version 2
       RFC 3376: Internet Group Management Protocol, Version 3";
  }

  identity mld-proto {
    base group-management-protocol;
    description
      "Identity for MLD.";
    reference
      "RFC 2710: Multicast Listener Discovery (MLD) for IPv6
       RFC 3810: Multicast Listener Discovery Version 2 (MLDv2)
                 for IPv6";
  }

  identity pim-proto {
    if-feature "pim";
    base routing-protocol-type;
    description
      "Identity for PIM.";
    reference
      "RFC 7761: Protocol Independent Multicast - Sparse Mode
                (PIM-SM): Protocol Specification (Revised)";
  }

  identity igmp-version {
    if-feature "igmp";
    description
      "Base identity for IGMP version.";
  }

  identity igmpv1 {
    base igmp-version;
    description
      "Identity for IGMPv1.";
    reference
      "RFC 1112: Host Extensions for IP Multicasting";
  }

  identity igmpv2 {
    base igmp-version;
```

```
    description
      "Identity for IGMPv2.";
    reference
      "RFC 2236: Internet Group Management Protocol, Version 2";
  }

  identity igmpv3 {
    base igmp-version;
    description
      "Identity for IGMPv2.";
    reference
      "RFC 3376: Internet Group Management Protocol, Version 3";
  }

  identity mld-version {
    if-feature "mld";
    description
      "Base identity for MLD version.";
  }

  identity mldv1 {
    base mld-version;
    description
      "Identity for MLDv1.";
    reference
      "RFC 2710: Multicast Listener Discovery (MLD) for IPv6";
  }

  identity mldv2 {
    base mld-version;
    description
      "Identity for MLDv2.";
    reference
      "RFC 3810: Multicast Listener Discovery Version 2 (MLDv2)
                 for IPv6";
  }

  /*
   * Identities related to traffic types
   */

  identity tf-type {
    description
      "Identity for the traffic type.";
  }

  identity multicast-traffic {
    base tf-type;
```

```
    description
      "Identity for multicast traffic.";
  }

  identity broadcast-traffic {
    base tf-type;
    description
      "Identity for broadcast traffic.";
  }

  identity unknown-unicast-traffic {
    base tf-type;
    description
      "Identity for unknown unicast traffic.";
  }

  /*
   * Identities related to customer applications
   */

  identity customer-application {
    description
      "Base identity for customer applications.";
  }

  identity web {
    base customer-application;
    description
      "Identity for a Web application (e.g., HTTP, HTTPS).";
  }

  identity mail {
    base customer-application;
    description
      "Identity for a mail application.";
  }

  identity file-transfer {
    base customer-application;
    description
      "Identity for a file transfer application (e.g., FTP, SFTP).";
  }

  identity database {
    base customer-application;
    description
      "Identity for a database application.";
  }
```

```
  identity social {
    base customer-application;
    description
      "Identity for a social-network application.";
  }

  identity games {
    base customer-application;
    description
      "Identity for a gaming application.";
  }

  identity p2p {
    base customer-application;
    description
      "Identity for a peer-to-peer application.";
  }

  identity network-management {
    base customer-application;
    description
      "Identity for a management application (e.g., Telnet, syslog,
       SNMP).";
  }

  identity voice {
    base customer-application;
    description
      "Identity for a voice application.";
  }

  identity video {
    base customer-application;
    description
      "Identity for a video conference application.";
  }

  identity embb {
    base customer-application;
    description
      "Identity for an enhanced Mobile Broadband (eMBB) application.
       Note that an eMBB application demands network performance with a
       wide variety of characteristics, such as data rate, latency,
       loss rate, reliability, and many other parameters.";
  }

  identity urllc {
    base customer-application;
```

```
    description
      "Identity for an Ultra-Reliable and Low Latency Communications
       (URLLC) application.  Note that an URLLC application demands
       network performance with a wide variety of characteristics, such
       as latency, reliability, and many other parameters.";
  }

  identity mmtc {
    base customer-application;
    description
      "Identity for a massive Machine Type Communications (mMTC)
       application.  Note that an mMTC application demands network
       performance with a wide variety of characteristics, such as data
       rate, latency, loss rate, reliability, and many other
       parameters.";
  }

  /*
   * Identities related to service bundling
   */

  identity bundling-type {
    description
      "The base identity for the bundling type. It supports a subset or
       all CE-VLANs associated with an L2VPN service.";
  }

  identity multi-svc-bundling {
    base bundling-type;
    description
      "Identity for multi-service bundling, i.e., multiple C-VLAN IDs
       can be associated with an L2VPN service at a site.";
  }

  identity one2one-bundling {
    base bundling-type;
    description
      "Identity for one-to-one service bundling, i.e., each L2VPN can
       be associated with only one C-VLAN ID at a site.";
  }

  identity all2one-bundling {
    base bundling-type;
    description
      "Identity for all-to-one bundling, i.e., all C-VLAN IDs are mapped
       to one L2VPN service.";
  }
```

```
  /*
   * Identities related to Ethernet Services
   */

  identity control-mode {
    description
      "Defines the type of control mode on Layer 2 Control Protocol
       (L2CP).";
  }

  identity peer {
    base control-mode;
    description
      "'peer' mode, i.e., participate in the protocol towards the CE.
       Peering is common for Link Aggregation Control Protocol (LACP)
       and the Ethernet Local Management Interface (E-LMI) and,
       occasionally, for Link Layer Discovery Protocol (LLDP).
       For VPLSs and VPWSs, the subscriber can also request that the
       peer service provider enables spanning tree.";
  }

  identity tunnel {
    base control-mode;
    description
      "'tunnel' mode, i.e., pass to the egress or destination site. For
       Ethernet Private Lines (EPLs), the expectation is that L2CP
       frames are tunnelled.";
  }

  identity discard {
    base control-mode;
    description
      "Identity for 'discard' mode, i.e., discard the frame.";
  }

  identity neg-mode {
    description
      "Identity for the negotiation mode.";
  }

  identity full-duplex {
    base neg-mode;
    description
      "Identity for the full-duplex mode.";
  }

  identity auto-neg {
    base neg-mode;
```

```
    description
      "Identity for auto-negotiation mode.";
  }

  /******** Collection of VPN-related Types ********/

  typedef vpn-id {
    type string;
    description
      "Defines an identifier that is used as a service identifier,
       for example.";
  }

  /******* VPN-related reusable groupings *******/

  grouping vpn-description {
    description
      "Provides common VPN information.";
    leaf vpn-id {
      type vpn-id;
      description
        "VPN identifier.
         This identifier has a local meaning.";
    }
    leaf vpn-name {
      type string;
      description
        "A name used to refer to the VPN.";
    }
    leaf vpn-description {
      type string;
      description
        "Textual description of a VPN.";
    }
    leaf customer-name {
      type string;
      description
        "Name of the customer that actually uses the VPN.";
    }
  }

  grouping vpn-profile-cfg {
    description
      "Grouping for VPN Profile configuration.";
    container valid-provider-identifiers {
      description
        "Container for valid provider profile identifiers.";
      list external-connectivity-identifier {
```

```
        if-feature "external-connectivity";
        key "id";
        description
          "List for profile identifiers that uniquely identify profiles
           governing how external connectivity is provided to a VPN.
           A profile indicates the type of external connectivity
           (Internet, cloud, etc.), the sites/nodes that are associated
           with a connectivity profile, etc. A profile can also indicate
           filtering rules and/or address translation rules. Such
           features may involve PE, P, or dedicated nodes as a function
           of the deployment.";
        leaf id {
          type string;
          description
            "Identification of an external connectivity profile. It has
             a local administration meaning.";
        }
      }
      list encryption-profile-identifier {
        key "id";
        description
          "List for encryption profile identifiers.";
        leaf id {
          type string;
          description
            "Identification of the encryption profile to be used. It
             has a local administration meaning.";
        }
      }
      list qos-profile-identifier {
        key "id";
        description
          "List for QoS Profile Identifiers.";
        leaf id {
          type string;
          description
            "Identification of the QoS profile to be used. It has
             a local administration meaning.";
        }
      }
      list bfd-profile-identifier {
        key "id";
        description
          "List for BFD profile identifiers.";
        leaf id {
          type string;
          description
            "Identification of the BFD profile to be used.
```

```
            This identifier has a local administration meaning.";
        }
      }
      list forwarding-profile-identifier {
        key "id";
        description
          "List for forwarding profile identifiers.";
        leaf id {
          type string;
          description
            "Identification of the Forwrding Profile Filter to be used.
             Local administration meaning.";
        }
      }
      list routing-profile-identifier {
        key "id";
        description
          "List for Routing Profile Identifiers.";
        leaf id {
          type string;
          description
            "Identification of the routing profile to be used by the
             routing protocols within sites, vpn-network-accesses, or
             vpn-nodes for refering VRF's import/export policies.

             This identifier has a local meaning.";
        }
      }
      nacm:default-deny-write;
    }
  }

  grouping status-timestamp {
    description
      "This grouping defines some operational parameters for the
       service.";
    leaf status {
      type identityref {
        base operational-status;
      }
      config false;
      description
        "Operations status.";
    }
    leaf last-updated {
      type yang:date-and-time;
      config false;
      description
```

```
        "Indicates the actual date and time of the service status
         change.";
    }
  }

  grouping service-status {
    description
      "Service status grouping.";
    container status {
      description
        "Service status.";
      container admin-status {
        description
          "Administrative service status.";
        leaf status {
          type identityref {
            base administrative-status;
          }
          description
            "Administrative service status.";
        }
        leaf last-updated {
          type yang:date-and-time;
          description
            "Indicates the actual date and time of the service status
             change.";
        }
      }
      container oper-status {
        description
          "Operational service status.";
        uses status-timestamp;
      }
    }
  }

  grouping underlay-transport {
    description
      "This grouping defines the type of underlay transport for the
       VPN service. It can include an identifier to an abstract
       transport instance to which the VPN is grafted or indicate a
       technical implementation that is expressed as an ordered list
       of protocols.";
    choice type {
      description
        "A choice based on the type of underlay transport
         constraints.";
      case abstract {
```

```
        description
          "Indicates that the transport constraint is an abstract
           concept.";
        leaf transport-instance-id {
          type string;
          description
            "Includes an identifier of an abstract transport instance.";
        }
        leaf instance-type {
          type identityref {
            base transport-instance-type;
          }
          description
            "Indicates a transport instance type. For example, it can
             be a VPN+, an IETF network slice, a virtual network, etc.";
        }
      }
      case protocol {
        description
          "Indicates a list of protocols.";
        leaf-list protocol {
          type identityref {
            base protocol-type;
          }
          ordered-by user;
          description
            "Indicates an ordered-by user list of transport protocols.";
        }
      }
    }
  }

  grouping vpn-route-targets {
    description
      "A grouping that specifies Route Target (RT) import-export rules
       used in a BGP-enabled VPN.";
    reference
      "RFC 4364: BGP/MPLS IP Virtual Private Networks (VPNs)
       RFC 4664: Framework for Layer 2 Virtual Private Networks
                 (L2VPNs)";
    list vpn-target {
      key "id";
      description
        "Route targets. AND/OR operations are available
         based on the RTs assigment.";
      leaf id {
        type int8;
        description
```

```
          "Identifies each VPN Target.";
      }
      list route-targets {
        key "route-target";
        description
          "List of RTs.";
        leaf route-target {
          type rt-types:route-target;
          description
            "Conveys an RT value.";
        }
      }
      leaf route-target-type {
        type rt-types:route-target-type;
        mandatory true;
        description
          "Import/export type of the RT.";
      }
    }
    container vpn-policies {
      description
        "VPN service policies. It contains references to the
         import and export policies to be associated with the
         VPN service.";
      leaf import-policy {
        type string;
        description
          "Defines the 'import' policy.";
      }
      leaf export-policy {
        type string;
        description
          "Defines the 'export' policy.";
      }
    }
  }

  grouping route-distinguisher {
    description
      "Grouping for route distinguisher (RD).";
    choice rd-choice {
      description
        "Route distinguisher choice between several options
         on providing the route distinguisher value.";
      case directly-assigned {
        description
          "Explicitly assign an RD value.";
        leaf rd {
```

```
          type rt-types:route-distinguisher;
          description
            "Indicates an RD value that is explicitly
             assigned.";
        }
      }
      case directly-assigned-suffix {
        description
          "Explicitly the value of the Assigned Number subfield
           of the RD. The Administrator subfield of the RD will
           be based on other configuration information such as
           router-id or ASN.";
        leaf rd-suffix {
          type uint16;
          description
            "Indicates the value of the Assigned Number
             subfield that is explicitly assigned.";
        }
      }
      case auto-assigned {
        description
          "The RD is auto-assigned.";
        container rd-auto {
          description
            "The RD is auto-assigned.";
          choice auto-mode {
            description
              "Indicates the auto-assignment mode. RD can be
               automatically assigned either with or without
               indicating a pool from which the RD should be
               taken.

               For both cases, the server will auto-assign an RD
               value 'auto-assigned-rd' and use that value
               operationally.";
            case from-pool {
              leaf rd-pool-name {
                type string;
                description
                  "The auto-assignment will be made from the pool
                   identified by the rd-pool-name.";
              }
            }
            case full-auto {
              leaf auto {
                type empty;
                description
                  "Indicates an RD is fully auto-assigned.";
```

```
            }
          }
        }
        leaf auto-assigned-rd {
          type rt-types:route-distinguisher;
          config false;
          description
            "The value of the auto-assigned RD.";
        }
      }
    }
    case auto-assigned-suffix {
      description
        "The value of the Assigned Number subfield will
         be auto-assigned. The Administrator subfield
         will be based on other configuration information such as
         router-id or ASN.";
      container rd-auto-suffix {
        description
          "The Assigned Number subfield is auto-assigned.";
        choice auto-mode {
          description
            "Indicates the auto-assignment mode of the Assigned Number
             subfield. This number can be automatically assigned
             either with or without indicating a pool from which
             the value should be taken.

             For both cases, the server will auto-assign
             'auto-assigned-rd-suffix' and use that value to build
             the RD that will be used operationally.";
          case from-pool {
            leaf rd-pool-name {
              type string;
              description
                "The assignment will be made from the pool identified
                 by the rd-pool-name.";
            }
          }
          case full-auto {
            leaf auto {
              type empty;
              description
                "Indicates that the Assigned Number is fully auto
                 assigned.";
            }
          }
        }
        leaf auto-assigned-rd-suffix {
```

```
            type uint16;
            config false;
            description
              "Includes the value of the Assigned Number subfield that
               is auto-assigned .";
          }
        }
      }
      case no-rd {
        description
          "Use the empty type to indicate RD has no value and is not to
           be auto-assigned.";
        leaf no-rd {
          type empty;
          description
            "No RD is assigned.";
        }
      }
    }
  }

  grouping vpn-components-group {
    description
      "Grouping definition to assign group-ids to associate VPN nodes,
       sites, or network accesses.";
    container groups {
      description
        "Lists the groups to which a VPN node,a site, or a network
         access belongs to.";
      list group {
        key "group-id";
        description
          "List of group-ids.";
        leaf group-id {
          type string;
          description
            "Is the group-id to which a VPN node, a site, or a network
             access belongs to.";
        }
      }
    }
  }

  grouping placement-constraints {
    description
      "Constraints for placing a network access.";
    list constraint {
      key "constraint-type";
```

```
      description
        "List of constraints.";
      leaf constraint-type {
        type identityref {
          base placement-diversity;
        }
        description
          "Diversity constraint type.";
      }
      container target {
        description
          "The constraint will apply against this list of groups.";
        choice target-flavor {
          description
            "Choice for the group definition.";
          case id {
            list group {
              key "group-id";
              description
                "List of groups.";
              leaf group-id {
                type string;
                description
                  "The constraint will apply against this particular
                   group-id.";
              }
            }
          }
          case all-accesses {
            leaf all-other-accesses {
              type empty;
              description
                "The constraint will apply against all other network
                 accesses of a site.";
            }
          }
          case all-groups {
            leaf all-other-groups {
              type empty;
              description
                "The constraint will apply against all other groups that
                 the customer is managing.";
            }
          }
        }
      }
    }
  }
```

```
  grouping ports {
    description
      "Choice of specifying a source or destination port numbers.";
    choice source-port {
      description
        "Choice of specifying the source port or referring to a group
         of source port numbers.";
      container source-port-range-or-operator {
        description
          "Source port definition.";
        uses packet-fields:port-range-or-operator;
      }
    }
    choice destination-port {
      description
        "Choice of specifying a destination port or referring to a group
         of destination port numbers.";
      container destination-port-range-or-operator {
        description
          "Destination port definition.";
        uses packet-fields:port-range-or-operator;
      }
    }
  }

  grouping qos-classification-policy {
    description
      "Configuration of the traffic classification policy.";
    list rule {
      key "id";
      ordered-by user;
      description
        "List of marking rules.";
      leaf id {
        type string;
        description
          "An identifier of the QoS classification policy rule.";
      }
      choice match-type {
        default "match-flow";
        description
          "Choice for classification.";
        case match-flow {
          choice l3 {
            description
              "Either IPv4 or IPv6.";
            container ipv4 {
              description
```

```
              "Rule set that matches IPv4 header.";
            uses packet-fields:acl-ip-header-fields;
            uses packet-fields:acl-ipv4-header-fields;
          }
          container ipv6 {
            description
              "Rule set that matches IPv6 header.";
            uses packet-fields:acl-ip-header-fields;
            uses packet-fields:acl-ipv6-header-fields;
          }
        }
        choice l4 {
          description
            "Includes Layer 4 specific information.
             This version focuses on TCP and UDP.";
          container tcp {
            description
              "Rule set that matches TCP header.";
            uses packet-fields:acl-tcp-header-fields;
            uses ports;
          }
          container udp {
            description
              "Rule set that matches UDP header.";
            uses packet-fields:acl-udp-header-fields;
            uses ports;
          }
        }
      }
      case match-application {
        leaf match-application {
          type identityref {
            base customer-application;
          }
          description
            "Defines the application to match.";
        }
      }
    }
    leaf target-class-id {
      if-feature "qos";
      type string;
      description
        "Identification of the class of service. This identifier is
         internal to the administration.";
    }
  }
}
```

```
}
<CODE ENDS>
```

5. Security Considerations

The YANG modules specified in this document define schemas for data that is designed to be accessed via network management protocols such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446].

The Network Configuration Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

The "ietf-vpn-common" module defines a set of identities, types, and groupings. These nodes are intended to be reused by other YANG modules. The module does not expose by itself any data nodes which are writable, contain read-only state, or RPCs. As such, there are no additional security issues to be considered relating to the "ietf-vpn-common" module.

6. IANA Considerations

This document requests IANA to register the following URI in the "ns" subregistry within the "IETF XML Registry" [RFC3688]:

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

This document requests IANA to register the following YANG module in the "YANG Module Names" subregistry [RFC6020] within the "YANG Parameters" registry.

```
name: ietf-vpn-common
namespace: urn:ietf:params:xml:ns:yang:ietf-vpn-common
maintained by IANA: N
prefix: vpn-common
reference: RFC XXXX
```

7. Acknowledgements

During the discussions of this work, helpful comments and reviews were received from (listed alphabetically): Alejandro Aguado, Raul Arco, Miguel Cros Cecilia, Joe Clarke, Dhruv Dhody, Adrian Farrel, Roque Gagliano, Christian Jacquenet, Kireeti Kompella, Julian Lucek, Tom Petch, Erez Segev, and Paul Sherratt. Many thanks to them.

This work is partially supported by the European Commission under Horizon 2020 grant agreement number 101015857 Secured autonomic traffic management for a Tera of SDN flows (Teraflow).

Many thanks to Radek Krejci for the yangdoctors review, Wesley Eddy for the tsvart review, and Ron Bonica for the Rtgdir review.

8. Contributors

Italo Busi
Huawei Technologies
Email: Italo.Busi@huawei.com

Luis Angel Munoz
Vodafone
Email: luis-angel.munoz@vodafone.com

Victor Lopez Alvarez
Telefonica
Email: victor.lopezalvarez@telefonica.com

9. References

9.1. Normative References

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

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/info/rfc6242>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8294] Liu, X., Qu, Y., Lindem, A., Hopps, C., and L. Berger, "Common YANG Data Types for the Routing Area", RFC 8294, DOI 10.17487/RFC8294, December 2017, <https://www.rfc-editor.org/info/rfc8294>.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <https://www.rfc-editor.org/info/rfc8341>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8519] Jethanandani, M., Agarwal, S., Huang, L., and D. Blair, "YANG Data Model for Network Access Control Lists (ACLs)", RFC 8519, DOI 10.17487/RFC8519, March 2019, <https://www.rfc-editor.org/info/rfc8519>.

9.2. Informative References

[I-D.ietf-opsawg-l2nm]
Barguil, S., Dios, O. G. D., Boucadair, M., and L. A. Munoz, "A Layer 2 VPN Network YANG Model", draft-ietf-opsawg-l2nm-02 (work in progress), April 2021.

[I-D.ietf-opsawg-l3sm-l3nm]
Barguil, S., Dios, O. G. D., Boucadair, M., Munoz, L. A., and A. Aguado, "A Layer 3 VPN Network YANG Model", draft-ietf-opsawg-l3sm-l3nm-08 (work in progress), April 2021.

[I-D.ietf-teas-actn-vn-yang]
Lee, Y., Dhody, D., Ceccarelli, D., Bryskin, I., and B. Y. Yoon, "A YANG Data Model for VN Operation", draft-ietf-teas-actn-vn-yang-11 (work in progress), February 2021.

[I-D.ietf-teas-enhanced-vpn]
Dong, J., Bryant, S., Li, Z., Miyasaka, T., and Y. Lee, "A Framework for Enhanced Virtual Private Network (VPN+) Services", draft-ietf-teas-enhanced-vpn-07 (work in progress), February 2021.

[I-D.ietf-teas-ietf-network-slice-framework]
Gray, E. and J. Drake, "Framework for IETF Network Slices", March 2021, <https://datatracker.ietf.org/doc/draft-ietf-teas-ietf-network-slice-framework/>.

[IEEE802.1ad]
"Virtual Bridged Local Area Networks Amendment 4: Provider Bridges", IEEE Std 802.1ad-2005, 2006.

[IEEE802.1AX]
"Link Aggregation", IEEE Std 802.1AX-2020, 2020.

[IEEE802.1Q]
"Bridges and Bridged Networks", IEEE Std 802.1Q-2018, July 2018.

[ISO10589]
ISO, "Intermediate System to Intermediate System intra-domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode network service (ISO 8473)", 2002, <International Standard 10589:2002, Second Edition>.

[RFC1112] Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112, DOI 10.17487/RFC1112, August 1989, <https://www.rfc-editor.org/info/rfc1112>.

[RFC1701] Hanks, S., Li, T., Farinacci, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 1701, DOI 10.17487/RFC1701, October 1994, <https://www.rfc-editor.org/info/rfc1701>.

[RFC1702] Hanks, S., Li, T., Farinacci, D., and P. Traina, "Generic Routing Encapsulation over IPv4 networks", RFC 1702, DOI 10.17487/RFC1702, October 1994, <https://www.rfc-editor.org/info/rfc1702>.

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, DOI 10.17487/RFC2003, October 1996, <https://www.rfc-editor.org/info/rfc2003>.

[RFC2080] Malkin, G. and R. Minnear, "RIPng for IPv6", RFC 2080, DOI 10.17487/RFC2080, January 1997, <https://www.rfc-editor.org/info/rfc2080>.

[RFC2236] Fenner, W., "Internet Group Management Protocol, Version 2", RFC 2236, DOI 10.17487/RFC2236, November 1997, <https://www.rfc-editor.org/info/rfc2236>.

[RFC2453] Malkin, G., "RIP Version 2", STD 56, RFC 2453, DOI 10.17487/RFC2453, November 1998, <https://www.rfc-editor.org/info/rfc2453>.

[RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", RFC 2473, DOI 10.17487/RFC2473, December 1998, <https://www.rfc-editor.org/info/rfc2473>.

[RFC2710] Deering, S., Fenner, W., and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710, DOI 10.17487/RFC2710, October 1999, <https://www.rfc-editor.org/info/rfc2710>.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <https://www.rfc-editor.org/info/rfc3209>.

[RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376, DOI 10.17487/RFC3376, October 2002, <https://www.rfc-editor.org/info/rfc3376>.

[RFC3810] Vida, R., Ed. and L. Costa, Ed., "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, DOI 10.17487/RFC3810, June 2004, <https://www.rfc-editor.org/info/rfc3810>.

[RFC3931] Lau, J., Ed., Townsley, M., Ed., and I. Goyret, Ed., "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, DOI 10.17487/RFC3931, March 2005, <https://www.rfc-editor.org/info/rfc3931>.

[RFC4026] Andersson, L. and T. Madsen, "Provider Provisioned Virtual Private Network (VPN) Terminology", RFC 4026, DOI 10.17487/RFC4026, March 2005, <https://www.rfc-editor.org/info/rfc4026>.

[RFC4176] El Mghazli, Y., Ed., Nadeau, T., Boucadair, M., Chan, K., and A. Gonguet, "Framework for Layer 3 Virtual Private Networks (L3VPN) Operations and Management", RFC 4176, DOI 10.17487/RFC4176, October 2005, <https://www.rfc-editor.org/info/rfc4176>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4577] Rosen, E., Psenak, P., and P. Pillay-Esnault, "OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4577, DOI 10.17487/RFC4577, June 2006, <https://www.rfc-editor.org/info/rfc4577>.

[RFC4664] Andersson, L., Ed. and E. Rosen, Ed., "Framework for Layer 2 Virtual Private Networks (L2VPNs)", RFC 4664, DOI 10.17487/RFC4664, September 2006, <https://www.rfc-editor.org/info/rfc4664>.

[RFC4761] Kompella, K., Ed. and Y. Rekhter, Ed., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, DOI 10.17487/RFC4761, January 2007, <https://www.rfc-editor.org/info/rfc4761>.

[RFC4762] Lasserre, M., Ed. and V. Kompella, Ed., "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, DOI 10.17487/RFC4762, January 2007, <https://www.rfc-editor.org/info/rfc4762>.

[RFC4960] Stewart, R., Ed., "Stream Control Transmission Protocol", RFC 4960, DOI 10.17487/RFC4960, September 2007, <https://www.rfc-editor.org/info/rfc4960>.

[RFC5036] Andersson, L., Ed., Minei, I., Ed., and B. Thomas, Ed., "LDP Specification", RFC 5036, DOI 10.17487/RFC5036, October 2007, <https://www.rfc-editor.org/info/rfc5036>.

[RFC5798] Nadas, S., Ed., "Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6", RFC 5798, DOI 10.17487/RFC5798, March 2010, <https://www.rfc-editor.org/info/rfc5798>.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <https://www.rfc-editor.org/info/rfc5880>.

[RFC6513] Rosen, E., Ed. and R. Aggarwal, Ed., "Multicast in MPLS/ BGP IP VPNs", RFC 6513, DOI 10.17487/RFC6513, February 2012, <https://www.rfc-editor.org/info/rfc6513>.

[RFC6565] Pillay-Esnault, P., Moyer, P., Doyle, J., Ertekin, E., and M. Lundberg, "OSPFv3 as a Provider Edge to Customer Edge (PE-CE) Routing Protocol", RFC 6565, DOI 10.17487/RFC6565, June 2012, <https://www.rfc-editor.org/info/rfc6565>.

[RFC6624] Kompella, K., Kothari, B., and R. Cherukuri, "Layer 2 Virtual Private Networks Using BGP for Auto-Discovery and Signaling", RFC 6624, DOI 10.17487/RFC6624, May 2012, <https://www.rfc-editor.org/info/rfc6624>.

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC7510] Xu, X., Sheth, N., Yong, L., Callon, R., and D. Black, "Encapsulating MPLS in UDP", RFC 7510, DOI 10.17487/RFC7510, April 2015, <https://www.rfc-editor.org/info/rfc7510>.

[RFC7623] Sajassi, A., Ed., Salam, S., Bitar, N., Isaac, A., and W. Henderickx, "Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)", RFC 7623, DOI 10.17487/RFC7623, September 2015, <https://www.rfc-editor.org/info/rfc7623>.

[RFC7676] Pignataro, C., Bonica, R., and S. Krishnan, "IPv6 Support for Generic Routing Encapsulation (GRE)", RFC 7676, DOI 10.17487/RFC7676, October 2015, <https://www.rfc-editor.org/info/rfc7676>.

[RFC7761] Fenner, B., Handley, M., Holbrook, H., Kouvelas, I., Parekh, R., Zhang, Z., and L. Zheng, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", STD 83, RFC 7761, DOI 10.17487/RFC7761, March 2016, <https://www.rfc-editor.org/info/rfc7761>.

[RFC8214] Boutros, S., Sajassi, A., Salam, S., Drake, J., and J. Rabadan, "Virtual Private Wire Service Support in Ethernet VPN", RFC 8214, DOI 10.17487/RFC8214, August 2017, <https://www.rfc-editor.org/info/rfc8214>.

[RFC8277] Rosen, E., "Using BGP to Bind MPLS Labels to Address Prefixes", RFC 8277, DOI 10.17487/RFC8277, October 2017, <https://www.rfc-editor.org/info/rfc8277>.

[RFC8299] Wu, Q., Ed., Litkowski, S., Tomotaki, L., and K. Ogaki, "YANG Data Model for L3VPN Service Delivery", RFC 8299, DOI 10.17487/RFC8299, January 2018, <https://www.rfc-editor.org/info/rfc8299>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <https://www.rfc-editor.org/info/rfc8365>.

[RFC8453] Ceccarelli, D., Ed. and Y. Lee, Ed., "Framework for Abstraction and Control of TE Networks (ACTN)", RFC 8453, DOI 10.17487/RFC8453, August 2018, <https://www.rfc-editor.org/info/rfc8453>.

[RFC8466] Wen, B., Fioccola, G., Ed., Xie, C., and L. Jalil, "A YANG Data Model for Layer 2 Virtual Private Network (L2VPN) Service Delivery", RFC 8466, DOI 10.17487/RFC8466, October 2018, <https://www.rfc-editor.org/info/rfc8466>.

[RFC8512] Boucadair, M., Ed., Sivakumar, S., Jacquenet, C., Vinapamula, S., and Q. Wu, "A YANG Module for Network Address Translation (NAT) and Network Prefix Translation (NPT)", RFC 8512, DOI 10.17487/RFC8512, January 2019, <https://www.rfc-editor.org/info/rfc8512>.

[RFC8660] Bashandy, A., Ed., Filsfils, C., Ed., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with the MPLS Data Plane", RFC 8660, DOI 10.17487/RFC8660, December 2019, <https://www.rfc-editor.org/info/rfc8660>.

[RFC8663] Xu, X., Bryant, S., Farrel, A., Hassan, S., Henderickx, W., and Z. Li, "MPLS Segment Routing over IP", RFC 8663, DOI 10.17487/RFC8663, December 2019, <https://www.rfc-editor.org/info/rfc8663>.

[RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <https://www.rfc-editor.org/info/rfc8754>.

[RFC8926] Gross, J., Ed., Ganga, I., Ed., and T. Sridhar, Ed., "Geneve: Generic Network Virtualization Encapsulation", RFC 8926, DOI 10.17487/RFC8926, November 2020, <https://www.rfc-editor.org/info/rfc8926>.

Appendix A. Example of Common Data Nodes in Early L2NM/L3NM Designs

Subtrees of early versions of the L3NM and L2NM are shown in Figure 5.

```
module: ietf-l2vpn-ntw
   +--rw vpn-services
       +--rw vpn-service* [vpn-id]
             +--rw vpn-id                     svc-id
             +--rw vpn-svc-type?              identityref
             +--rw customer-name?             string
             +--rw svc-topo?                  identityref
             +-rw service-status
             |  +-rw admin
             |  |  +-rw status?      operational-type
             |  |  +-rw timestamp?   yang:date-and-time
             |  +-ro ops
             |     +-ro status?      operational-type
             |     +-ro timestamp?   yang:date-and-time
             |  ...

module: ietf-l3vpn-ntw
   +--rw vpn-services
       +--rw vpn-service* [vpn-id]
             +-rw service-status
             |  +-rw admin
             |  |  +-rw status?      operational-type
             |  |  +-rw timestamp?   yang:date-and-time
             |  +-ro ops
             |     +-ro status?      operational-type
             |     +-ro timestamp?   yang:date-and-time
             +--rw vpn-id                  l3vpn-svc:svc-id
             +--rw l3sm-vpn-id?            l3vpn-svc:svc-id
             +--rw customer-name?          string
             +--rw vpn-service-topology?   identityref
             +--rw description?            string
             |  ...
```

Figure 5: Example of Common Data Nodes in Both L2NM/L3NM

In order to avoid data nodes duplication and to ease passing data among layers (i.e., from the service layer to the network layer and vice versa), early versions of the L3NM reused many of the data nodes that are defined in the L3SM. Nevertheless, that approach was abandoned because that design was interpreted as if the deployment of L3NM depends on L3SM, while this is not required. For example, a service provider may decide to use the L3NM to build its L3VPN services without exposing the L3SM to customers.

Likewise, early versions of the L2NM reused many of the data nodes that are defined in both L2SM and L3NM. An example of L3NM groupings reused in L2NM is shown in Figure 6. Such data nodes reuse was

interpreted as if the deployment of the L2NM requires the support of the L3NM; which is not required.

```
module ietf-l2vpn-ntw {
 ...
  import ietf-l3vpn-ntw {
    prefix l3vpn-ntw;
    reference
      "RFC NNNN: A Layer 3 VPN Network YANG Model";
  }
  ...
  container l2vpn-ntw {
    ...
    container vpn-services {
      list vpn-service {
        ...
        uses l3vpn-ntw:service-status;
        uses l3vpn-ntw:svc-transport-encapsulation;
        ...
      }
    }
    ...
  }
}
```

Figure 6: Excerpt from the L2NM YANG Module

Authors' Addresses

Samier Barguil
Telefonica
Madrid
Spain

Email: samier.barguilgiraldo.ext@telefonica.com

Oscar Gonzalez de Dios (editor)
Telefonica
Madrid
Spain

Email: oscar.gonzalezdedios@telefonica.com

Mohamed Boucadair (editor)
Orange
France

Email: mohamed.boucadair@orange.com

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: bill.wu@huawei.com