opsawg
Internet-Draft
Intended status: Standards Track
Expires: May 2, 2021

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

October 29, 2020

A Layer 2/3 VPN Common YANG Model
draft-ietf-opsawg-vpn-common-02

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

Table of Contents

1. Introduction

Various VPN-related YANG data modules were specified by the IETF (e.g., Layer 3 VPN Service Model (L3SM) [RFC8299] or Layer 2 VPN Service Model (L2SM) [RFC8466]). Others are also being specified (e.g., Layer 3 VPN Network Model (L3NM) [I-D.ietf-opsawg-l3sm-l3nm] or Layer 2 VPN Network Model (L2NM) [I-D.ietf-opsawg-l2nm]). These modules have data nodes and structures that are present in almost all these models or a subset of them. An example of such data nodes is depicted in Figure 1.

```
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 1: Example of Common Data Nodes in Both L2NM/L3NM

In order to avoid data nodes duplication and to ease passing data among layers (service layer to network layer and vice versa), early versions of the L3NM reused many of the data nodes that are defined in the L3SM [RFC8299]. 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.

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 L3NM is shown in Figure 2. This data nodes reuse was

interpreted as if the deployment of L2NM requires both 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 2: Excerpt from the L2NM YANG Module

To avoid the issues discussed above, this document defines a common YANG module that is meant to be reused by various VPN-related modules such as Layer 3 VPN Network Model (L3NM) [I-D.ietf-opsawg-l3sm-l3nm] and Layer 2 VPN Network Model (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 future revisions (if any) of existing models (e.g., Layer 3 VPN Service Model (L3SM) [RFC8299] or Layer 2 VPN Service Model (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].

3. Description of the VPN Common YANG Module

The "ietf-vpn-common" module defines a set of common identities. It also contains the following reusable groupings:

- 'ports':

  A YANG grouping that defines ranges of source and destination port numbers and operators.

- 'qos-classification-policy':

  A YANG grouping that defines a set of QoS classification policies based on various match Layer 3/4 criteria.

- 'vpn-description':

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

- 'vpn-profile-cfg':

  A YANG grouping that defines a set of profiles (encryption, routing, forwarding) valid for any Layer 2/3 VPN.

- 'status-timestamp':

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

- 'svc-transport-encapsulation':

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

- 'rt-rd':

  A YANG grouping that defines the set of route targets, defined as Route targets (RTs) and Route Distinguishers (RDs), to match for import and export routes to/from a Virtual Routing and Forwarding (VRF).

- 'group':

  A YANG grouping that is used to group VPN nodes, VPN network accesses, or sites.

- 'placement-constraints':

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

The tree diagram of the "ietf-vpn-common" module that depicts the common groupings is provided in Figure 3.

```
module: ietf-vpn-common

  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
  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
     grouping vpn-description
       +-- vpn-id?            vpn-common:vpn-id
       +-- vpn-name?          string
       +-- vpn-description?   string
       +-- customer-name?     string
     grouping vpn-profile-cfg
       +-- valid-provider-identifiers
          +-- cloud-identifier* [id] {cloud-access}?
          |  +-- 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
       +-- status?         identityref
       +-- last-updated?   yang:date-and-time
     grouping service-status
       +-- status
          +-- admin-status
          |  +-- status?         identityref
          |  +-- last-updated?   yang:date-and-time
          +--ro oper-status
             +--ro status?         identityref
             +--ro last-updated?   yang:date-and-time
     grouping svc-transport-encapsulation
       +-- underlay-transport
          +-- type*   identityref
     grouping rt-rd
       +-- (rd-choice)?
       |  +--:(directly-assigned)
```

```
      |  |  +-- rd?             rt-types:route-distinguisher
      |  +--:(pool-assigned)
      |  |  +-- rd-pool-name?   string
      |  |  +--ro rd-assign?      rt-types:route-distinguisher
      |  +--:(full-autoasigned)
      |  |  +-- auto?           empty
      |  |  +--ro rd-assigned?    rt-types:route-distinguisher
      |  +--:(no-rd)
      |     +-- no-rd?          empty
      +-- vpn-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 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 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
```

Figure 3: VPN Common Tree

4. Layer 2/3 VPN Common Module

This module uses types defined in [RFC6991], [RFC8294], and [RFC8519].

Editor's Note: Check that RFCs cited in the reference statements are included in the References Section and called out in the core text.

```
<CODE BEGINS>  file "ietf-vpn-common@2020-10-26.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
      "Section 3 of RFC 6991";
  }
  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>

     Editor:  Samier Barguil
              <mailto:samier.barguilgiraldo.ext@telefonica.com>

     Editor:  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) 2020 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
      (https://trustee.ietf.org/license-info).

      This version of this YANG module is part of RFC XXXX
      (https://www.rfc-editor.org/info/rfcXXXX); see the RFC itself
      for full legal notices.";

   revision 2020-10-26 {
     description
       "Initial revision.";
     reference
       "RFC XXXX: A Layer 2/3 VPN Common YANG Model";
   }

   /* Features */

   feature cloud-access {
     description
       "Indicates support of the VPN to connect to a Cloud
        Service Provider (CSP).";
   }

   feature lag-interface {
     description
       "Indicates the support of link aggregation between
        VPN site network accesses.";
   }

   feature placement-diversity {
```

```
    description
      "Indicates the 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.";
  }

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

  feature ipv4 {
```

```
    description
      "Indicates IPv4 support in a VPN.";
  }

  feature ipv6 {
    description
      "Indicates IPv6 support in a VPN.";
  }

  feature carrierscarrier {
    description
      "Indicates support of Carrier-of-Carrier VPNs.";
    reference
      "Section 9 of RFC 4364";
  }

  feature extranet-vpn {
    description
      "Indicates support of extranet VPNs. That is,
       the capability of a VPN to access a list of
       other VPNs.";
  }

  feature fast-reroute {
    description
      "Indicates support of Fast Reroute (FRR).";
  }

  feature qos {
    description
      "Indicates support of Classes of Services (CoSes).";
  }

  feature encryption {
    description
      "Indicates support of encryption.";
  }

  feature rtg-ospf {
    description
      "Indicates support of the OSPF routing protocol.";
    reference
      "RFC 4577: OSPF as the Provider/Customer Edge Protocol
                 for BGP/MPLS IP Virtual Private Networks
                 (VPNs)";
  }

  feature rtg-ospf-sham-link {
```

```
    description
      "This feature indicates the support of OSPF sham links.";
    reference
      "Section 4.2.7 of RFC 4577";
  }

  feature rtg-bgp {
    description
      "Indicates support of BGP as the Provider/Customer
       Edge protocol.";
  }

  feature rtg-rip {
    description
      "Indicates support of RIP as the Provider/Customer
       Edge protocol.";
  }

  feature rtg-vrrp {
    description
      "Indicates support of the Virtual Router Redundancy
       Protocol (VRRP) between a cutsomer LAN and the PE.";
  }

  feature rtg-isis {
    description
      "Indicates the support of IS-IS as the Provider/Customer
       Edge protocol.";
  }

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

  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 SP apart from the IP VPN
       site.";
  }

  feature input-bw {
    description
```

```
      "Indicates the support of the input bandwidth in a VPN.";
  }

  /* Typedef */

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

  typedef address-family {
    type enumeration {
      enum ipv4 {
        description
          "IPv4 address family.";
      }
      enum ipv6 {
        description
          "IPv6 address family.";
      }
    }
    description
      "Defines a type for the address family.";
  }

  //L2xMs

  typedef ccm-priority-type {
    type uint8 {
      range "0..7";
    }
    description
      "A 3-bit priority value to be used in the VLAN tag,
       if present in the transmitted frame.";
  }

  typedef control-mode {
    type enumeration {
      enum peer {
        description
          "'peer' mode, i.e., participate in the protocol towards
           the CE.  Peering is common for LACP and the Ethernet
           Local Management Interface (E-LMI) and, occasionally,
           for LLDP.  For VPLSs and VPWSs, the subscriber can also
           request that the SP peer enable spanning tree.";
      }
```

```
      enum tunnel {
        description
          "'tunnel' mode, i.e., pass to the egress or destination
           site.  For EPLs, the expectation is that L2CP frames are
           tunneled.";
      }
      enum discard {
        description
          "'discard' mode, i.e., discard the frame.";
      }
    }
    description
      "Defines the type of control mode on L2CP protocols.";
  }

  typedef neg-mode {
    type enumeration {
      enum full-duplex {
        description
          "Defines full-duplex mode.";
      }
      enum auto-neg {
        description
          "Defines auto-negotiation mode.";
      }
    }
    description
      "Defines the type of negotiation mode.";
  }

  /* Identities */

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

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

  identity bgp {
    base routing-protocol-type;
    description
      "Identity for BGP protocol type.";
  }
```

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

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

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

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

        This is to be used when LANs are directly connected
        to PE routers.";
   }

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

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

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

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

   identity output-bw {
```

```
    base bw-direction;
    description
      "Identity for the output bandwidth.";
  }

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

  identity bw-per-cos {
    base bw-type;
    description
      "Bandwidth is per CoS.";
  }

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

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

  identity bw-per-svc {
    base bw-type;
    description
      "Bandwidth is per VPN service.";
  }

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

  identity site-to-wan {
    base qos-profile-direction;
    description
      "Identity for Site-to-WAN direction.";
  }

  identity wan-to-site {
    base qos-profile-direction;
```

```
    description
      "Identity for WAN-to-Site direction.";
  }

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

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

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

  identity ie-type {
    description
      "Defines Import-Export routing profiles.
       Those profiles can be reused between VPN nodes.";
  }

  identity import {
    base ie-type;
    description
      "Import a routing profile.";
  }

  identity export {
    base ie-type;
    description
      "Export a routing profile.";
  }

  identity import-export {
    base ie-type;
    description
      "Import/Export a routing profile.";
  }

  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 connections.";
  }
```

```
   identity multipoint {
     base site-network-access-type;
     description
       "Identity for multipoint connections.
        Example: Ethernet broadcast segment.";
   }

   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 connections.";
   }

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

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

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

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

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

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

   identity administrative-state-down {
     base administrative-status;
     description
       "Administrative status is DOWN/Disabled.";
   }

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

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

   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 {
     base encapsulation-type;
     description
       "Identity for the support of the 'dot1q'
        encapsulation.";
   }

   identity qinq {
     base encapsulation-type;
     description
       "Identity for the support of the 'qinq'
        encapsulation.";
```

```
  }

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

  identity 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 {
    base encapsulation-type;
    description
      "Identity of the Link Aggregation Group (LAG)
       interface type.";
    reference
      "IEEE Std. 802.1AX: Link Aggregation";
  }
```

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

   identity c-vlan {
     base tag-type;
     description
       "A CVLAN tag, normally using the 0x8100 Ethertype.";
   }

   identity s-vlan {
     base tag-type;
     description
       "An SVLAN tag.";
   }

   identity c-s-vlan {
     base tag-type;
     description
       "Uses both a CVLAN tag and an SVLAN tag.";
   }

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

   identity gre {
     base protocol-type;
     description
       "GRE encapsulation.";
     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 ldp {
     base protocol-type;
     description
       "Transport based on LDP.";
     reference
       "RFC 3086: LDP Specification";
   }

   identity sr {
```

```
    base protocol-type;
    description
      "Transport based on 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-te {
    base protocol-type;
    description
      "Transport based on SR-TE.";
    reference
      "RFC 8426: Recommendations for RSVP-TE and Segment Routing (SR)
                 Label Switched Path (LSP) Coexistence";
  }

  identity rsvp-te {
    base protocol-type;
    description
      "Transport based on RSVP-TE.";
    reference
      "RFC 2205: Resource ReSerVation Protocol (RSVP) --
                 Version 1 Functional Specification";
  }

  identity bgp-lu {
    base protocol-type;
    description
      "Transport 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.";
  }

  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.";
  }

  identity hub-spoke {
    base vpn-topology;
    description
      "Identity for Hub-and-Spoke VPN topology.";
  }

  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.
       VPN topology 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.";
  }

  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 IP VPN.";
  }

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

  identity hub-role {
    base role;
```

```
    description
      "A node or a site isacting as a Hub IP VPN.";
  }

  identity custom-role {
    base role;
    description
      "VPN-Node with custom or complex role in the VPN.
       For certain sources/destinations,  it can behave
       as a hub but for others it can act as a spoke
       depending on the configured policy.";
  }

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

  identity l2vpn-bgp {
    base vpn-signaling-type;
    description
      "Identity of Layer 2 VPNs using BGP";
    reference
      "RFC 6624: Layer 2 Virtual Private Networks Using BGP for
                 Auto-Discovery and Signaling";
  }

  identity evpn-bgp {
    base vpn-signaling-type;
    description
      "Identity of BGP Ethernet VPNs.";
    reference
      "RFC 7432: BGP MPLS-Based Ethernet VPN";
  }

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

  identity h-vpls {
    base vpn-signaling-type;
    description
      "Identity of hierarchical VPLS.";
    reference
```

```
       "RFC 4762: Virtual Private LAN Service (VPLS) Using
                  Label Distribution Protocol (LDP)
                  Signaling";
   }

   identity l2tp {
     base vpn-signaling-type;
     description
       "Identity of l2tp.";
   }

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

   identity l3vpn {
     base service-type;
     description
       "Identity of L3VPN service.";
   }

   identity vpws {
     base service-type;
     description
       "Identity of the Point-to-point Virtual Private
        Wire Service (VPWS) service type.";
   }

   identity pwe3 {
     base service-type;
     description
       "Identity of the Pseudowire Emulation Edge to Edge
        (PWE3) service type.";
   }

   identity ldp-l2tp-vpls {
     base service-type;
     description
       "Identity of the LDP-based or L2TP-based multipoint
        Virtual Private LAN Service (VPLS) service type.
        This VPLS uses LDP-signaled Pseudowires
        or L2TP-signaled Pseudowires.";
   }

   identity bgp-vpls {
     base service-type;
     description
```

```
      "Identity of the BGP-based multipoint VPLS service type.
       This VPLS uses a BGP control plane.";
    reference
      "RFC 4761:  Virtual Private LAN Service (VPLS) Using
                  BGP for Auto-Discovery and Signaling
       RFC 6624: Layer 2 Virtual Private Networks Using BGP for
                  Auto-Discovery and Signaling";
  }

  identity vpws-evpn {
    base service-type;
    description
      "Identity of the VPWS service type using EVPNs.";
    reference
      "RFC 8214: Virtual Private Wire Service Support
                  in Ethernet VPN";
  }

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

  identity vxlan-peer-mode {
    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.";
  }

  identity multicast-gp-address-mapping {
    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 was added
       to the multicast group as a result of snooping.";
  }

  identity multicast-tree-type {
    description
      "Base identity for multicast tree type.";
  }

  identity ssm-tree-type {
    base multicast-tree-type;
    description
      "Identity for SSM tree type.";
  }

  identity asm-tree-type {
    base multicast-tree-type;
    description
      "Identity for ASM tree type.";
  }

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

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

   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.";
   }

   identity bundling-type {
     description
       "The base identity for the bundling type.  It supports
        multiple CE-VLANs associated with an L2VPN service 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 CE-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 CE-VLAN ID
       at a site.";
  }

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

  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 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.";
   }

   /* Grouping */

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

   grouping qos-classification-policy {
     list rule {
       key "id";
       ordered-by user;
       leaf id {
         type string;
         description
           "A description identifying the
            qos-classification-policy rule.";
```

```
      }
      choice match-type {
        default "match-flow";
        case match-flow {
          choice l3 {
            container ipv4 {
              uses packet-fields:acl-ip-header-fields;
              uses packet-fields:acl-ipv4-header-fields;
              description
                "Rule set that matches IPv4 header.";
            }
            container ipv6 {
              uses packet-fields:acl-ip-header-fields;
              uses packet-fields:acl-ipv6-header-fields;
              description
                "Rule set that matches IPv6 header.";
            }
            description
              "Either IPv4 or IPv6.";
          }
          choice l4 {
            container tcp {
              uses packet-fields:acl-tcp-header-fields;
              uses ports;
              description
                "Rule set that matches TCP header.";
            }
            container udp {
              uses packet-fields:acl-udp-header-fields;
              uses ports;
              description
                "Rule set that matches UDP header.";
            }
            description
              "Can be TCP or UDP";
          }
        }
        case match-application {
          leaf match-application {
            type identityref {
              base customer-application;
            }
            description
              "Defines the application to match.";
          }
        }
        description
          "Choice for classification.";
```

```
      }
      leaf target-class-id {
        type string;
        description
          "Identification of the class of service.
           This identifier is internal to the
           administration.";
      }
      description
        "List of marking rules.";
    }
    description
      "Configuration of the traffic classification
       policy.";
  }

  grouping vpn-description {
    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 service.";
    }
    leaf customer-name {
      type string;
      description
        "Name of the customer that actually uses the VPN service.";
    }
    description
      "Provides common VPN information.";
  }

  grouping vpn-profile-cfg {
    container valid-provider-identifiers {
      list cloud-identifier {
        if-feature "cloud-access";
        key "id";
        leaf id {
```

```
          type string;
          description
            "Identification of cloud service.
             Local administration meaning.";
        }
        description
          "List for Cloud Identifiers.";
      }
      list encryption-profile-identifier {
        key "id";
        leaf id {
          type string;
          description
            "Identification of the SP encryption profile
             to be used.  Local administration meaning.";
        }
        description
          "List for encryption profile identifiers.";
      }
      list qos-profile-identifier {
        key "id";
        leaf id {
          type string;
          description
            "Identification of the QoS Profile to be used.
             Local administration meaning.";
        }
        description
          "List for QoS Profile Identifiers.";
      }
      list bfd-profile-identifier {
        key "id";
        leaf id {
          type string;
          description
            "Identification of the SP BFD Profile to be used.
             Local administration meaning.";
        }
        description
          "List for BFD Profile identifiers.";
      }
      list forwarding-profile-identifier {
        key "id";
        leaf id {
          type string;
          description
            "Identification of the Forwrding Profile Filter to be used.
             Local administration meaning.";
```

```
        }
        description
          "List for Forwrding Profile identifiers.";
      }
      list routing-profile-identifier {
        key "id";
        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-import/export policies.

             This identifier has a local meaning.";
        }
        description
          "List for Routing Profile Identifiers.";
      }
      nacm:default-deny-write;
      description
        "Container for Valid Provider Identifies.";
    }
    description
      "Grouping for VPN Profile configuration.";
  }

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

  grouping service-status {
    container status {
```

```
      container admin-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.";
        }
        description
          "Administrative service status.";
      }
      container oper-status {
        config false;
        uses status-timestamp;
        description
          "Operational service status.";
      }
      description
        "Service status.";
    }
    description
      "Service status grouping.";
  }

  grouping svc-transport-encapsulation {
    container underlay-transport {
      leaf-list type {
        type identityref {
          base protocol-type;
        }
        ordered-by user;
        description
          "Protocols used to deliver a VPN service.";
      }
      description
        "Container for the Transport underlay.";
    }
    description
      "This grouping defines the type of underlay transport
       for VPN service.";
  }
```

```
   grouping rt-rd {
     choice rd-choice {
       case directly-assigned {
         leaf rd {
           type rt-types:route-distinguisher;
           description
             "Explicitly assign a route distinguisher (RD) value.";
         }
         description
           "Explicitly assign a RD value";
       }
       case pool-assigned {
         leaf rd-pool-name {
           type string;
           description
             "The server will auto-assign a route
              distinguisher value and use that value operationally.
              The assignment will be selected from the pool
              identified by the rd-pool-name.";
         }
         leaf rd-assign {
           type rt-types:route-distinguisher;
           config false;
           description
             "Route distinguisher is assigned.";
         }
       }
       case full-autoasigned {
         leaf auto {
           type empty;
           description
             "Indicates an RD is fully auto assigned.";
         }
         leaf rd-assigned {
           type rt-types:route-distinguisher;
           config false;
           description
             "Route distinguisher is assigned.";
         }
       }
       case no-rd {
         leaf no-rd {
           type empty;
           description
             "No RD is assigned.";
         }
         description
           "Use the empty type to indicate RD has no value and
```

```
            is not to be auto-assigned.";
      }
      description
        "Route distinguisher choice between several options
         on providing the route distiniguisher value.";
    }
    container vpn-targets {
      description
        "Set of route-targets to match for import and export routes
         to/from VRF";
      uses vpn-route-targets;
    }
    description
      "Grouping for RT and RD.";
  }

  grouping vpn-route-targets {
    description
      "A grouping that specifies Route Target import-export rules
       used in a BGP-enabled VPN.";
    list vpn-target {
      key "id";
      leaf id {
        type int8;
        description
          "Identifies each VPN Target";
      }
      list route-targets {
        key "route-target";
        leaf route-target {
          type rt-types:route-target;
          description
            "Route Target value";
        }
        description
          "List of Route Targets.";
      }
      leaf route-target-type {
        type rt-types:route-target-type;
        mandatory true;
        description
          "Import/export type of the Route Target.";
      }
      description
        "L3VPN route targets. AND/OR Operations are available
         based on the RTs assigment.";
    }
    reference
```

```
       "RFC 4364: BGP/MPLS IP Virtual Private Networks (VPNs)
        RFC 4664: Framework for Layer 2 Virtual Private Networks
                  (L2VPNs)";
     container vpn-policies {
       description
         "VPN policies";
       leaf import-policy {
         type string;
         description
           "Defines the import policy.";
       }
       leaf export-policy {
         type string;
         description
           "Defines the export policy.";
       }
     }
   }

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

   grouping placement-constraints {
     list constraint {
       key "constraint-type";
       leaf constraint-type {
         type identityref {
           base placement-diversity;
```

```
        }
        description
          "Diversity constraint type.";
      }
      container target {
        choice target-flavor {
          case id {
            list group {
              key "group-id";
              leaf group-id {
                type string;
                description
                  "The constraint will apply
                   against this particular
                   group-id.";
              }
              description
                "List of groups";
            }
          }
          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 the
                 customer is managing.";
            }
          }
          description
            "Choice for the group definition";
        }
        description
          "The constraint will apply against
           this list of groups.";
      }
      description
        "List of constraints.";
    }
```

```
    description
      "Constraints for placing a network
       access.";
  }
}
<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. As such, 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. 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

8. References

8.1. 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>.

[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>.

8.2. Informative References

[I-D.ietf-opsawg-l2nm]
barguil, s., Dios, O., Boucadair, M., Munoz, L., Jalil, L., and J. Ma, "A Layer 2 VPN Network YANG Model", draft-ietf-opsawg-l2nm-00 (work in progress), July 2020.

[I-D.ietf-opsawg-l3sm-l3nm]
barguil, s., Dios, O., Boucadair, M., Munoz, L., and A. Aguado, "A Layer 3 VPN Network YANG Model", draft-ietf-opsawg-l3sm-l3nm-05 (work in progress), October 2020.

[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>.

[RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <https://www.rfc-editor.org/info/rfc2205>.

[RFC3086] Nichols, K. and B. Carpenter, "Definition of Differentiated Services Per Domain Behaviors and Rules for their Specification", RFC 3086, DOI 10.17487/RFC3086, April 2001, <https://www.rfc-editor.org/info/rfc3086>.

[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>.

[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>.

[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>.

[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>.

[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>.

[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>.

[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>.

[RFC8426] Sitaraman, H., Ed., Beeram, V., Minei, I., and S. Sivabalan, "Recommendations for RSVP-TE and Segment Routing (SR) Label Switched Path (LSP) Coexistence", RFC 8426, DOI 10.17487/RFC8426, July 2018, <https://www.rfc-editor.org/info/rfc8426>.

[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>.

[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>.

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