Network Working Group B. Wu
Internet-Draft D. Dhody
Intended status: Experimental Huawei Technologies
Expires: December 8, 2021 June 6, 2021

A VTN Network YANG Module
draft-wd-teas-vtn-network-yang-00

Abstract

This document defines a virtual transport network (VTN) network YANG module for retrieving and manipulating VTN topology and resource allocation. The model can be used to implement the provisioning of IETF network slice services.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

[I-D.ietf-teas-ietf-network-slices] defines IETF network slice services that provide connectivity coupled with network resources commitment between a number of endpoints over a shared network infrastructure, and also defines the IETF Network Slice controller (NSC) to realize the network slice services by mapping it to a suitable underlying technology.

[I-D.ietf-teas-enhanced-vpn] describes that enhanced VPN (VPN+) services can be used to realize IETF network slice services. To improve service scalability, The virtual transport network (VTN), which has a customized network topology and a group of dedicated or shared nodes and links of the physical network, is introduced for multiple VPN+ services with similar connection and SLA requirements. For the control and management of these VTN resources, [I-D.dong-teas-enhanced-vpn-vtn-scalability] gives a detailed analysis and description.

This document defines VTN network model that the NSC can use to create and manage VTN instances to realize the network slicing services. According to the YANG model classification of [RFC8309], VTN network model is a network configuration model.

2. Conventions used in this document

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP14, [RFC2119], [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following terms are defined in [RFC6241] and are used in this specification:

- o configuration data
- o state data

The following terms are defined in [RFC7950] and are used in this specification:

- o augment
- o data model
- o data node

The terminology for describing YANG data models is found in [RFC7950].

2.1. Tree Diagrams

The tree diagram used in this document follows the notation defined in [RFC8340].

3. VTN Network Yang Module Consideration

To realize the IETF Network Slice based on the reference framework defined in [I-D.ietf-teas-ietf-network-slices] , the Figure 1 shows an approach with VPN network model and VTN network YANG module.

```
          +------------------------------------------+
          |                Customer                  |
          |                                          |
          +------------------------------------------+
                               A
                               | NSC NBI
                               V
          +------------------------------------------+
          |    IETF Network Slice Controller (NSC)   |
          +------------------------------------------+
                               A
                 LxNM model    | NSC SBI  VTN network module
                               V          VTN as VPN underlay
          +------------------------------------------+
          |           Network Controller(s)          |
          +------------------------------------------+
                               A
             Device model      |       VTN device model
                               V
       +--------------------------------------------------+
                             Network
```

Figure 1: Reference Module Use Case

The VTN network model can be used in the following ways:

- o Static VTN configuration: A VTN instance can be created before processing IETF Network Slice service request by a network controller.

- o Dynamic VTN configuration: A VTN instance can be initiated along with configuring IETF Network Slice service request by a network controller.

In the process of realizing an IETF network slice service, when creating a Layer 3 VPN or Layer 2 VPN instance, The NSC can use a static VTN instance or dynamically create one as the VPN underlay transport. Compared with existing VPN underlying full mesh tunneling mechanisms, the VTN could provide resource isolation, topology constraints, and simplified configuration. Additionally, specific service flows of a VPN can be further optimized using SR policies defined in [I-D.dong-idr-sr-policy-vtn].

And also in multi-domain network slicing cases, instead of mapping the overlay VPN to the intra-domain VTNs at the edge of each domain, an inter-domain VTN could be used directly for inter-domain interconnection, which is described in

[I-D.li-teas-e2e-ietf-network-slicing] . The network controller serving the transit domain can only manage the VTNs. A 5G end-to-end network slicing scenario is shown in the following figure.

```
  /----\        /----\         /----\          /----\         /----\
 /      \     //      \\     //      \\      //      \\      /      \
|  RAN   |---|   NW-1   |---|   NW-2   |----|   NW-3   |----|  Core  |
 \      /     \\      //     \\      //      \\      //      \      /
  \----/        \----/         \----/          \----/         \----/

  S-NSSAI
o-----------------------------------------------------------------------o
          IETF Network Slice (VPN+)
        o---------------------------------------------------o
             Global VTN
           o============================================o
            Domain VTN-1   Domain VTN-2    Domain VTN-3
           o************o o************o   o***********o
```

5G end-to-end network slicing scenario

In addition to providing VTN network configuration, VTN network model also provides monitoring details of the underlying resource created to meet the requirements of IETF network slice service.

An example of VTN instances and a physical network is illustrated in Figure 2.

```
               ++++   ++++   ++++
               +--+===+--+===+--+
               +--+===+--+===+--+
               ++++   +++\\  ++++
                ||     || \\  ||            Physical
                ||     ||  \\ ||            Network
        ++++   ++++   ++++  \\+++   ++++
        +  +===+--+===+--+===+--+===+  +
        +  +===+--+===+--+===+--+===+  +
        ++++   ++++   ++++   ++++   ++++
         PE1                         PE2
                         |
                        \|/

                o----o-----o
               /          /               VTN-1
        o-----o-----o----o----o


                o----o
               /    / \                   VTN-2
        o-----o----o---o------o

                                           ...

                     o----o
                    /    /                VTN-n
        o-----o----o----o-----o

           o   is a virtual node
           --- is a virtual link
```

Figure 2: A VTN example

In the example, each VTN instance has a customized network topology comprised of a set of links and nodes in the physical network. In control plane, each VTN is associated with a multi-topology or a Flex-Algo. And it also has its own forwarding plane resources and identifiers which provide VTN-specific packet processing.

3.1. VTN Operation

There are multiple modes of VTN operations to be supported as follows.

o New VTN Binding: In realization, a NSC could request a set of underlay resources that are unaffected by other slice services. A

new VTN could be created and bound to a VPN per the network slice service and not used for any other VPNs.

- VTN Sharing: A NSC could decide to use allocated underlay resources to meet the requirements of an IETF network slice. Therefore, an existing VTN instance can be reused and multiple VPNs in the VTN instance can share same VTN resources. In some cases, the properties of the existing VTN (e.g., link bandwidth) need modification.

- VTN Deletion: If the NSC determines that no VPN service is using a VTN, the NSC can delete the VTN instance.

- VTN Monitoring: The NSC could also use the VTN network model to track and monitor VTN resource status and usage.

3.2. VTN Network Modeling Design

A VTN network is modeled as network topology defined in [RFC8345] with augmentations. A new network type "vtn" is defined in this document. When a network topology data instance contains the vtn network type, it represents an instance of a VTN.

Each VTN consists of a set of nodes and a set of links. Each node and link have different attributes that represent the allocated resources or the operational status of the VTN network. VTN supports several resource partition methods, which are defined by 'interface-partition-capability'' under a link, which can further be supported by FlexE and independent queue techniques.

The container "vtn" under 'network' of [RFC8345] defines global parameters for a VTN, which defines the specific control plane technique of the VTN and a unique "vtn-data-plane identifier" for data plane. And also, a color attribute for steering traffic, such as VPN traffic, into a VTN is also defined.

4. Description of the VTN Network YANG Module

The description of the VTN data nodes are as follows:

- "vtn-id": Is an identifier that is used to uniquely identify the VTN instance within the network scope.

- VTN allocation resources: The nodes and links represent the network resource allocated for a VTN instance. 'bandwidth-reservation' specifies the bandwidth allocated to a VTN network, or is overridden by the configuration of the VTN link.

'interface-partition-capability' specifies the resource partition capability of the physical interfaces associated with a VTN link.

o VTN control plane: Based on the existing work in IETF, control plane mechanism of VTN could be implemented by Multi-Topology Routing (MTR) which defined in [RFC4915], [RFC5120], and [I-D.ietf-lsr-isis-sr-vtn-mt] or Flex-algo which is defined in [I-D.ietf-lsr-flex-algo]. With these control plane technologies, VTN nodes of each VTN instance will create their own VTN-specific forwarding tables.

o VTN data plane: Defines the data plane mechanism and the VTN identifier of the network domain managed by the network controller. The data plane mechanism could be baed on MPLS or IPv6 forwarding. "vtn-domain-identifier" is used to identify network resource of data plane that has been allocated for the VTN. In the case of IPv6 based forwarding, VTN data plane identifier is defined in [I-D.dong-6man-enhanced-vpn-vtn-id]. If a network slice service traverses multiple network domains, a global VTN identifier across the domains may be defined. For example, [I-D.li-6man-e2e-ietf-network-slicing] defines a IPv6 extension header to carry the global VTN identifier.

o VTN steering policy: "vtn-color-id" is the color attribute of VTN for traffic steering.

5. VTN Yang Module Tree

```
module: ietf-vtn-ntw
  augment /nw:networks/nw:network/nw:network-types:
    +--rw vtn!
  augment /nw:networks/nw:network:
    +--rw vtn
       +--rw vtn-id?                  uint32
       +--rw vtn-name?                string
       +--rw bandwidth-reservation
       |  +--rw (bandwidth-type)?
       |     +--:(bandwidth-value)
       |     |  +--rw bandwidth-value?     uint64
       |     +--:(bandwidth-percentage)
       |        +--rw bandwidth-percent?   rt-types:percentage
       +--rw control-plane
       |  +--rw (vtn-cp-type)?
       |     +--:(flex-algo)
       |     |  +--rw flex-algo
       |     |     +--rw flex-algo-id?   uint32
       |     +--:(multi-topology)
       |        +--rw multi-topology-id?   uint32
       +--rw data-plane
       |  +--rw vtn-global-identifier?   uint32
       |  +--rw domain-data-plane
       |     +--rw data-plane-type?         identityref
       |     +--rw vtn-domain-identifier?   uint32
       +--rw steering-policy
          +--rw vtn-color-id?   uint32
  augment /nw:networks/nw:network/nw:node:
    +--rw vtn
  augment /nw:networks/nw:network/nt:link:
    +--rw vtn
    |  +--rw interface-partition-capability?   identityref
    |  +--rw bandwidth-reservation
    |     +--rw (bandwidth-type)?
    |        +--:(bandwidth-value)
    |        |  +--rw bandwidth-value?     uint64
    |        +--:(bandwidth-percentage)
    |           +--rw bandwidth-percent?   rt-types:percentage
    +--ro statistics
       +--ro admin-status?                  te-types:te-admin-status
       +--ro oper-status?                   te-types:te-oper-status
       +--ro one-way-available-bandwidth?   rt-types:bandwidth-ieee-float32
       +--ro one-way-utilized-bandwidth?    rt-types:bandwidth-ieee-float32
       +--ro one-way-min-delay?             uint32
       +--ro one-way-max-delay?             uint32
       +--ro one-way-delay-variation?       uint32
       +--ro one-way-packet-loss?           decimal64
```

6. VTN Yang Module

```
   <CODE BEGINS> file "ietf-vtn-ntw@2021-06-04.yang"

module ietf-vtn-ntw {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-vtn-ntw";
  prefix vtn-ntw;

  import ietf-network {
    prefix nw;
    reference
      "RFC 8345: A YANG Data Model for Network Topologies";
  }
  import ietf-network-topology {
    prefix nt;
    reference
      "RFC 8345: A YANG Data Model for Network Topologies";
  }
  import ietf-routing-types {
    prefix rt-types;
    reference
      "RFC 8294: Common YANG Data Types for the Routing Area";
  }
  import ietf-te-types {
    prefix te-types;
    reference
      "RFC 8776: Traffic Engineering Common YANG Types";
  }
  import ietf-te-packet-types {
    prefix te-packet-types;
    reference
      "RFC 8776: Traffic Engineering Common YANG Types";
  }

  organization
    "IETF TEAS Working Group";
  contact
    "
     WG Web: <http://tools.ietf.org/wg/teas/>
     WG List:<mailto:teas@ietf.org>

     Editor: Bo Wu <lana.wubo@huawei.com>
           : Dhruv Dhody <dhruv.ietf@gmail.com>";
  description
    "This YANG module defines a network data module for
     VTN(Virtual Transport Network)";
```

```
  revision 2021-06-04 {
    description
      "This is the initial version of VTN network yang module";
    reference
      "RFC XXX: YANG Data module for VTN network";
  }

  identity interface-partition-capability {
    description
      "Base identity for interface partition capability.";
  }

  identity flexe-partition {
    base interface-partition-capability;
    description
      "Identity for FlexE partition capability.";
  }

  identity queue-partition {
    base interface-partition-capability;
    description
      "Identity for queue partition capability.";
  }

  identity vtn-data-plane-type {
    description
      "Base identity for VTN data plane type.";
  }

  identity vtn-data-plane-vtn-ipv6 {
    base vtn-data-plane-type;
    description
      "Identity for VTN based packet forwarding of IPv6.";
  }

  identity vtn-data-plane-vtn-mpls {
    base vtn-data-plane-type;
    description
      "Identity for VTN based packet forwarding of MPLS.";
  }

  identity vtn-data-plane-sr-mpls {
    base vtn-data-plane-type;
    description
      "Identity for SR MPLS forwarding mechanism.";
  }

  identity vtn-data-plane-srv6 {
```

```
    base vtn-data-plane-type;
    description
      "Identity for SRv6 forwarding mechanism.";
  }

  /*
   * Groupings
   */

  grouping traffic-steering-policy {
    description
      "Configuration of the traffic mapping policy.";
    container steering-policy {
      description
        "Policy set that matches to a VTN.";
      leaf vtn-color-id {
        type uint32;
        description
          "VTN color ID for VTN traffic steering";
      }
    }
  }

  grouping vtn-bandwidth-reservation {
    description
      "Grouping for VTN bandwidth reservation.";
    container bandwidth-reservation {
      description
        "Container for VTN bandwidth reservation.";
      choice bandwidth-type {
        description
          "Choice of bandwidth reservation type.";
        case bandwidth-value {
          leaf bandwidth-value {
            type uint64;
            units "bps";
            description
              "Bandwidth allocation for the VTN as absolute value.";
          }
        }
        case bandwidth-percentage {
          leaf bandwidth-percent {
            type rt-types:percentage;
            description
              "Bandwidth allocation for the VTN as a percentage of a link.";
          }
        }
      }
```

```
    }
  }

  grouping vtn-control-plane-attributes {
    description
      "VTN topology control plane attributes.";
    container control-plane {
      description
        "vtn control plane mechanism.";
      choice vtn-cp-type {
        description
          "Choice of vtn control plane.";
        case flex-algo {
          container flex-algo {
            description
              "A VTN could use flex-algo as a control plane
               mechanism.";
            leaf flex-algo-id {
              type uint32;
              description
                "flex-algo-id for VTN";
            }
          }
        }
        case multi-topology {
          description
            "A VTN could use MT (Multi-Topology) as a control
             plane mechanism.";
          leaf multi-topology-id {
            type uint32;
            description
              "MT-id for VTN";
          }
        }
      }
    }
  }

  grouping vtn-data-plane-attributes {
    description
      "Grouping for VTN topology data plane attributes.";
    container data-plane {
      description
        "VTN data plane mechanism.";
      leaf vtn-global-identifier {
        type uint32;
        description
          "The global VTN identifier for multi-domain is specified.";
```

```
      }
      container domain-data-plane {
        description
          "VTN data plane mechanism per network domain.";
        leaf data-plane-type {
          type identityref {
            base vtn-data-plane-type;
          }
          description
            "Specifies the data plane forwarding mechanism of the VTN.
             The mechanism consists of VTN based Packet Forwarding or
             existing Segment Routing with MPLS data plane or IPv6 data
             plane.";
        }
        leaf vtn-domain-identifier {
          type uint32;
          description
            "The domain VTN identifier is specified for
             VTN based Packet Forwarding of a network domain.
             The forwarding plane could be with
             the MPLS Data Plane or IPv6";
          reference
            "draft-li-mpls-enhanced-vpn-vtn-id?
             Carrying Virtual Transport Network identifier
             in MPLS Packet
             draft-dong-6man-enhanced-vpn-vtn-id
             Carrying Virtual Transport Network Identifier
             in IPv6 Extension Header";
        }
      }
    }
  }

  grouping vtn-topology-attributes {
    description
      "VTN topology scope attributes.";
    container vtn {
      description
        "Containing VTN topology attributes.";
      leaf vtn-id {
        type uint32;
        description
          "VTN identifier";
      }
      leaf vtn-name {
        type string;
        description
          "VTN Name";
```

```
      }
      uses vtn-bandwidth-reservation;
      uses vtn-control-plane-attributes;
      uses vtn-data-plane-attributes;
      uses traffic-steering-policy;
    }
    // vtn
  }

  // vtn-node-attributes

  grouping vtn-node-attributes {
    description
      "VTN node scope attributes.";
    container vtn {
      description
        "Containing VTN attributes.";
    }
  }

  // vtn-node-attributes

  grouping vtn-link-attributes {
    description
      "VTN link scope attributes";
    container vtn {
      description
        "Containing VTN attributes.";
      leaf interface-partition-capability {
        type identityref {
          base interface-partition-capability;
        }
        description
          "Describes different resource partition type of a link.";
      }
      uses vtn-bandwidth-reservation;
    }
  }

  // vtn-statistics

  grouping statistics-per-vtn {
    description
      "Statistics attributes per VTN.";
  }

  // vtn-node-statistics
```

```
  grouping statistics-per-node {
    description
      "Statistics attributes per VTN node.";
  }

  // one-way-performance-metrics

  grouping one-way-performance-bandwidth {
    description
      "Grouping for one-way performance bandwidth .";
    leaf one-way-available-bandwidth {
      type rt-types:bandwidth-ieee-float32;
      units "bytes per second";
      default "0x0p0";
      description
        "Available bandwidth that is defined to be VTN link
         bandwidth minus bandwidth utilization.  For a
         bundled link, available bandwidth is defined to be the
         sum of the component link available bandwidths.";
    }
    leaf one-way-utilized-bandwidth {
      type rt-types:bandwidth-ieee-float32;
      units "bytes per second";
      default "0x0p0";
      description
        "Bandwidth utilization that represents the actual
         utilization of the link (i.e. as measured in the router).
         For a bundled link, bandwidth utilization is defined to
         be the sum of the component link bandwidth
         utilizations.";
    }
  }

  // vtn-link-statistics

  grouping vtn-statistics-per-link {
    description
      "Statistics attributes per VTN link.";
    container statistics {
      config false;
      description
        "Statistics for VTN link.";
      leaf admin-status {
        type te-types:te-admin-status;
        description
          "The administrative state of the link.";
      }
      leaf oper-status {
```

```
        type te-types:te-oper-status;
        description
          "The current operational state of the link.";
      }
      uses one-way-performance-bandwidth;
      uses te-packet-types:one-way-performance-metrics-packet;
    }
  }

  augment "/nw:networks/nw:network/nw:network-types" {
    description
      "Defines the VTN topology type.";
    container vtn {
      presence "Indicates VTN topology";
      description
        "Its presence identifies the VTN type.";
    }
  }

  augment "/nw:networks/nw:network" {
    when 'nw:network-types/vtn-ntw:vtn' {
      description
        "Augment only for VTN topology.";
    }
    description
      "Augment VTN configuration and state.";
    uses vtn-topology-attributes;
  }

  augment "/nw:networks/nw:network/nw:node" {
    when '../nw:network-types/vtn-ntw:vtn' {
      description
        "Augment only for VTN topology.";
    }
    description
      "Augment node configuration and state.";
    uses vtn-node-attributes;
  }

  augment "/nw:networks/nw:network/nt:link" {
    when '../nw:network-types/vtn-ntw:vtn' {
      description
        "Augment only for VTN topology.";
    }
    description
      "Augment link configuration and state.";
    uses vtn-link-attributes;
    uses vtn-statistics-per-link;
```

```
  }
}
```

<CODE ENDS>

7. Security Considerations

The YANG module defined in this document 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 NETCONF access control model [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.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations.

vtn-link: A malicious client could attempt to remove a link from a topology, add a new link. In each case, the structure of the topology would be sabotaged, and this scenario could, for example, result in an VTN topology that is less than optimal.

The entries in the nodes above include the whole network configurations corresponding with the VTN, and indirectly create or modify the PE or P device configurations. Unexpected changes to these entries could lead to service disruption and/or network misbehavior.

8. IANA Considerations

This document registers a URI in the IETF XML registry [RFC3688]. Following the format in [RFC3688], the following registration is requested to be made:

URI: urn:ietf:params:xml:ns:yang:ietf-vtn-ntw
Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.

This document requests to register a YANG module in the YANG Module Names registry [RFC7950].

Name: ietf-vtn-ntw
Namespace: urn:ietf:params:xml:ns:yang:ietf-vtn-ntw
Prefix: vtn-ntw
Reference: RFC XXXX

9. Contributor

Zhenbin Li
Huawei

Email: lizhenbin@huawei.com

Jie Dong
Huawei

Email: jie.dong@huawei.com

10. References

10.1. Normative References

[I-D.dong-6man-enhanced-vpn-vtn-id]
Dong, J., Li, Z., Xie, C., and C. Ma, "Carrying Virtual Transport Network Identifier in IPv6 Extension Header", draft-dong-6man-enhanced-vpn-vtn-id-03 (work in progress), February 2021.

[I-D.dong-idr-sr-policy-vtn]
Dong, J., Hu, Z., and R. Pang, "BGP SR Policy Extensions for Virtual Transport Network", draft-dong-idr-sr-policy-vtn-00 (work in progress), October 2020.

[I-D.ietf-lsr-flex-algo]
Psenak, P., Hegde, S., Filsfils, C., Talaulikar, K., and A. Gulko, "IGP Flexible Algorithm", draft-ietf-lsr-flex-algo-15 (work in progress), April 2021.

[I-D.ietf-lsr-isis-sr-vtn-mt]
Xie, C., Ma, C., Dong, J., and Z. Li, "Using IS-IS Multi-Topology (MT) for Segment Routing based Virtual Transport Network", draft-ietf-lsr-isis-sr-vtn-mt-00 (work in progress), March 2021.

[I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-11 (work in progress), April 2021.

[I-D.li-6man-e2e-ietf-network-slicing]
Li, Z. and J. Dong, "Encapsulation of End-to-End IETF Network Slice Information in IPv6", draft-li-6man-e2e-ietf-network-slicing-00 (work in progress), April 2021.

[I-D.zhu-lsr-isis-sr-vtn-flexalgo]
Zhu, Y., Dong, J., and Z. Hu, "Using Flex-Algo for Segment Routing based VTN", draft-zhu-lsr-isis-sr-vtn-flexalgo-02 (work in progress), February 2021.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

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

[RFC4915] Psenak, P., Mirtorabi, S., Roy, A., Nguyen, L., and P. Pillay-Esnault, "Multi-Topology (MT) Routing in OSPF", RFC 4915, DOI 10.17487/RFC4915, June 2007, <https://www.rfc-editor.org/info/rfc4915>.

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

[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>.

[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>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8309] Wu, Q., Liu, W., and A. Farrel, "Service Models Explained", RFC 8309, DOI 10.17487/RFC8309, January 2018, <https://www.rfc-editor.org/info/rfc8309>.

[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>.

[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>.

[RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <https://www.rfc-editor.org/info/rfc8345>.

[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>.

10.2. Informative References

[I-D.dong-teas-enhanced-vpn-vtn-scalability] Dong, J., Li, Z., Qin, F., Yang, G., and J. N. Guichard, "Scalability Considerations for Enhanced VPN (VPN+)", draft-dong-teas-enhanced-vpn-vtn-scalability-02 (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-slices]
Farrel, A., Gray, E., Drake, J., Rokui, R., Homma, S., Makhijani, K., Contreras, L. M., and J. Tantsura, "Framework for IETF Network Slices", draft-ietf-teas-ietf-network-slices-00 (work in progress), April 2021.

[I-D.li-teas-e2e-ietf-network-slicing]
Li, Z. and J. Dong, "Framework for End-to-End IETF Network Slicing", draft-li-teas-e2e-ietf-network-slicing-00 (work in progress), April 2021.

Appendix A. Example VTN Network Model

Device could map

Authors' Addresses

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

Email: lana.wubo@huawei.com

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park
Bangalore, Karnataka 560066
India

Email: dhruv.ietf@gmail.com