Network Working Group L. Berger

Internet-Draft LabN Consulting, L.L.C.

Intended status: Standards Track C. Hopps

Expires: March 30, 2018 Deutsche Telekom

A. Lindem

Cisco Systems

D. Bogdanovic

X. Liu

Jabil

September 26, 2017

YANG Logical Network Elements

draft-ietf-rtgwg-lne-model-04

Abstract

This document defines a logical network element module. This module

can be used to manage the logical resource partitioning that may be

present on a network device. Examples of common industry terms for

logical resource partitioning are Logical Systems or Logical Routers.

Status of This Memo

This Internet-Draft is submitted in full conformance with the

provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering

Task Force (IETF). Note that other groups may also distribute

working documents as Internet-Drafts. The list of current Internet-

Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months

and may be updated, replaced, or obsoleted by other documents at any

time. It is inappropriate to use Internet-Drafts as reference

material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 30, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the

document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

Provisions Relating to IETF Documents

(https://trustee.ietf.org/license-info) in effect on the date of

Berger, et al. Expires March 30, 2018 [Page 1]

Internet-Draft YANG LNEs September 2017

publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as

described in the Simplified BSD License.

Table of Contents

1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . 2

1.1. Terminology . . . . . . . . . . . . . . . . . . . . . . . 3

2. Overview . . . . . . . . . . . . . . . . . . . . . . . . . . 3

3. Logical Network Elements . . . . . . . . . . . . . . . . . . 5

3.1. LNE Instantiation and Resource Assignment . . . . . . . . 6

3.2. LNE Management - LNE View . . . . . . . . . . . . . . . . 7

3.3. LNE Management - Host Network Device View . . . . . . . . 7

4. Security Considerations . . . . . . . . . . . . . . . . . . . 8

5. IANA Considerations . . . . . . . . . . . . . . . . . . . . . 9

6. Logical Network Element Model . . . . . . . . . . . . . . . . 9

7. References . . . . . . . . . . . . . . . . . . . . . . . . . 13

7.1. Normative References . . . . . . . . . . . . . . . . . . 13

7.2. Informative References . . . . . . . . . . . . . . . . . 14

Appendix A. Acknowledgments . . . . . . . . . . . . . . . . . . 14

Appendix B. Examples . . . . . . . . . . . . . . . . . . . . . . 15

B.1. Example: Host Device Managed LNE . . . . . . . . . . . . 15

B.1.1. Configuration Data . . . . . . . . . . . . . . . . . 19

B.1.2. State Data . . . . . . . . . . . . . . . . . . . . . 23

B.2. Example: Self Managed LNE . . . . . . . . . . . . . . . . 32

B.2.1. Configuration Data . . . . . . . . . . . . . . . . . 35

B.2.2. State Data . . . . . . . . . . . . . . . . . . . . . 38

Authors' Addresses . . . . . . . . . . . . . . . . . . . . . . . 47

1. Introduction

This document defines a YANG [RFC6020] module to support the creation

of logical network elements on a network device. A logical network

element (LNE) is an independently managed virtual device made up of

resources allocated to it from the host or parent network device. An

LNE running on a host network device conceptually parallels a virtual

machine running on a host system. Using host-virtualization

terminology one could refer to an LNE as a "Guest", and the

containing network-device as the "Host". While LNEs may be

implemented via host-virtualization technologies this is not a

requirement.

This document also defines the necessary augmentations for allocating

host resources to a given LNE. As the interface management model

[RFC7223] is the only a module that currently defines host resources,

Berger, et al. Expires March 30, 2018 [Page 2]

Internet-Draft YANG LNEs September 2017

this document currently defines only a single augmentation to cover

the assignment of interfaces to an LNE. Future modules that define

support for the control of host device resources are expected to,

where appropriate, provide parallel support for the assignment of

controlled resources to LNEs.

As each LNE is an independently managed device, each will have its

own set of YANG modeled data that is independent of the host device

and other LNEs. For example, multiple LNEs may all have their own

"Tunnel0" interface defined which will not conflict with each other

and will not exist in the host's interface model. An LNE will have

its own management interfaces possibly including independent

instances of netconf/restconf/etc servers to support configuration of

their YANG models. As an example of this independence, an

implementation may choose to completely rename assigned interfaces,

so on the host the assigned interface might be called "Ethernet0/1"

while within the LNE it might be called "eth1".

In addition to standard management interfaces, a host device

implementation may support accessing LNE configuration and

operational YANG models directly from the host system. When

supported, such access is accomplished through a yang-schema-mount

mount point [I-D.ietf-netmod-schema-mount] under which the root level

LNE YANG models may be accessed.

Examples of vendor terminology for an LNE include logical system or

logical router, and virtual switch, chassis, or fabric.

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",

"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this

document are to be interpreted as described in [RFC2119].

Readers are expected to be familiar with terms and concepts of YANG

[RFC7950] and YANG Schema Mount [I-D.ietf-netmod-schema-mount].

This document uses the graphical representation of data models

defined in [I-D.ietf-netmod-yang-tree-diagrams].

2. Overview

In this document, we consider network devices that support protocols

and functions defined within the IETF Routing Area, e.g, routers,

firewalls, and hosts. Such devices may be physical or virtual, e.g.,

a classic router with custom hardware or one residing within a

server-based virtual machine implementing a virtual network function

(VNF). Each device may sub-divide their resources into logical

Berger, et al. Expires March 30, 2018 [Page 3]

Internet-Draft YANG LNEs September 2017

network elements (LNEs), each of which provides a managed logical

device. Examples of vendor terminology for an LNE include logical

system or logical router, and virtual switch, chassis, or fabric.

Each LNE may also support virtual routing and forwarding (VRF) and

virtual switching instance (VSI) functions, which are referred to

below as a network instances (NIs). This breakdown is represented in

Figure 1.

,'''''''''''''''''''''''''''''''''''''''''''''''.

| Network Device (Physical or Virtual) |

| ..................... ..................... |

| : Logical Network : : Logical Network : |

| : Element : : Element : |

| :+-----+-----+-----+: :+-----+-----+-----+: |

| :| Net | Net | Net |: :| Net | Net | Net |: |

| :|Inst.|Inst.|Inst.|: :|Inst.|Inst.|Inst.|: |

| :+-----+-----+-----+: :+-----+-----+-----+: |

| : | | | | | | : : | | | | | | : |

| :..|.|...|.|...|.|..: :..|.|...|.|...|.|..: |

| | | | | | | | | | | | | |

''''|'|'''|'|'''|'|'''''''''|'|'''|'|'''|'|'''''

| | | | | | | | | | | |

Interfaces Interfaces

Figure 1: Module Element Relationships

A model for LNEs is described in Section 3 and the model for NIs is

covered in [I-D.ietf-rtgwg-ni-model].

The interface management model [RFC7223] is an existing model that is

impacted by the definition of LNEs and network instances. This

document and [I-D.ietf-rtgwg-ni-model] define augmentations to the

interface module to support LNEs and NIs. Similar elements, although

perhaps only for LNEs, may also need to be included as part of the

definition of the future hardware and QoS modules.

Interfaces are a crucial part of any network device's configuration

and operational state. They generally include a combination of raw

physical interfaces, link-layer interfaces, addressing configuration,

and logical interfaces that may not be tied to any physical

interface. Several system services, and layer 2 and layer 3

protocols may also associate configuration or operational state data

with different types of interfaces (these relationships are not shown

for simplicity). The interface management model is defined by

[RFC7223]. The logical-network-element module augments existing

interface management model by adding an identifier which is used on

physical interface types to identify an associated LNE.

Berger, et al. Expires March 30, 2018 [Page 4]

Internet-Draft YANG LNEs September 2017

The interface related augmentation is as follows:

module: ietf-logical-network-element

augment /if:interfaces/if:interface:

+--rw bind-lne-name? ->

/logical-network-elements/logical-network-element/name

The interface model defined in [RFC7223] is structured to include all

interfaces in a flat list, without regard to logical assignment of

resources supported on the device. The bind-lne-name and leaf

provides the association between an interface and its associated LNE.

Note that as currently defined, to assign an interface to both an LNE

and NI, the interface would first be assigned to the LNE and then

within that LNE's interface module, the LNE's representation of that

interface would be assigned to an NI using the mechanisms defined in

[I-D.ietf-rtgwg-ni-model].

3. Logical Network Elements

Logical network elements support the ability of some devices to

partition resources into independent logical routers and/or switches.

Device support for multiple logical network elements is

implementation specific. Systems without such capabilities need not

include support for the logical-network-element module. In physical

devices, some hardware features are shared across partitions, but

control plane (e.g., routing) protocol instances, tables, and

configuration are managed separately. For example, in logical

routers or VNFs, this may correspond to establishing multiple logical

instances using a single software installation. The model supports

configuration of multiple instances on a single device by creating a

list of logical network elements, each with their own configuration

and operational state related to routing and switching protocols.

The LNE model can be represented using the tree format defined in

[I-D.ietf-netmod-yang-tree-diagrams] as:

Berger, et al. Expires March 30, 2018 [Page 5]

Internet-Draft YANG LNEs September 2017

module: ietf-logical-network-element

+--rw logical-network-elements

+--rw logical-network-element\* [name]

+--rw name string

+--rw managed? boolean

+--rw description? string

+--mp root

augment /if:interfaces/if:interface:

+--rw bind-lne-name?

-> /logical-network-elements/logical-network-element/name

notifications:

+---n bind-lne-name-failed

+--ro name -> /if:interfaces/interface/name

+--ro bind-lne-name

| -> /if:interfaces/interface/lne:bind-lne-name

+--ro error-info? string

'name' identifies the logical network element. 'managed' indicates

if the server providing the host network device will provide the

client LNE information via the 'root' structure. The root of an

LNE's specific data is the schema mount point 'root'. bind-lne-name

is used to associated an interface with an LNE and bind-lne-name-

failed is used in certain failure cases.

An LNE root MUST contain at least the YANG library [RFC7895] and

Interfaces [RFC7223] modules.

3.1. LNE Instantiation and Resource Assignment

Logical network elements may be controlled by clients using existing

list operations. When list entries are created, a new LNE is

instantiated. The models mounted under an LNE root are expected to

be dependent on the server implementation. When a list entry is

deleted, an existing LNE is destroyed. For more information, see

[RFC7950] Section 7.8.6.

Once instantiated, host network device resources can be associated

with the new LNE. As previously mentioned, this document augments

ietf-interfaces with the bind-lne-name leaf to support such

associations for interfaces. When a bind-lne-name is set to a valid

LNE name, an implementation MUST take whatever steps are internally

necessary to assign the interface to the LNE or provide an error

message (defined below) with an indication of why the assignment

failed. It is possible for the assignment to fail while processing

the set, or after asynchronous processing. Error notification in the

latter case is supported via a notification.

Berger, et al. Expires March 30, 2018 [Page 6]

Internet-Draft YANG LNEs September 2017

On a successful interface assignment to an LNE, an implementation

MUST also make the resource available to the LNE by providing a

system created interface to the LNE. The name of the system created

interface is a local matter and may be identical or completely

different, and mapped from and to, the name used in the context of

the host device. The system created interface SHOULD be exposed via

the LNE-specific instance of the interfaces module [RFC7223].

3.2. LNE Management - LNE View

Each LNE instance is expected to support management functions from

within the context of the LNE root, via a server that provides

information with the LNE's root exposed as device root. Management

functions operating within the context of an LNE are accessed through

the LNE's standard management interfaces, e.g., NETCONF and SNMP.

Initial configuration, much like the initial configuration of the

host device, is a local implementation matter.

When accessing an LNE via the LNE's management interface, a network-

device representation will be presented, but its scope will be

limited to the specific LNE. Normal YANG/NETCONF mechanisms,

together with the required YANG library [RFC7895] instance, can be

used to identify the available modules. Each supported module will

be presented as a top level module. Only LNE associated resources

will be reflected in resource related modules, e.g., interfaces,

hardware, and perhaps QoS. From the management perspective, there

will be no difference between the available LNE view (information)

and a physical network device.

3.3. LNE Management - Host Network Device View

There are multiple implementation approaches possible to enable a

network device to support the logical-network-element module and

multiple LNEs. Some approaches will allow the management functions

operating at network device level to access LNE configuration and

operational information, while others will not. Similarly, even when

LNE management from the network device is supported by the

implementation, it may be prohibited by user policy.

Independent of the method selected by an implementation, the

'managed' boolean mentioned above is used to indicate when LNE

management from the network device context is possible. When the

'managed' boolean is 'false', the LNE cannot be managed by the host

system and can only be managed from within the context of the LNE as

described in the previous section, Section 3.2. Attempts to access

information below a root node whose associated 'managed' boolean is

set to 'false' MUST result in the error message indicated below. In

some implementations, it may not be possible to change this value.

Berger, et al. Expires March 30, 2018 [Page 7]

Internet-Draft YANG LNEs September 2017

For example, when an LNE is implemented using virtual machine and

traditional hypervisor technologies, it is likely that this value

will be restricted to a 'false' value.

It is an implementation choice if the information can be accessed and

modified from within the context of the LNE, or even the context of

the host device. When the 'managed' boolean is 'true', LNE

information SHALL be accessible from the context of the host device.

When the associated schema-mount definition has the 'config' leaf set

to 'true', then LNE information SHALL also be modifiable from the

context of the host device. When LNE information is available from

both the host device and from within the context of the LNE, the same

information MUST be made available via the 'root' element, with paths

modified as described in [I-D.ietf-netmod-schema-mount].

An implementation MAY represent an LNE's schema using either the

'inline' or 'use-schema' approaches defined in

[I-D.ietf-netmod-schema-mount]. The choice of which to use is

completely an implementation choice. The inline case is anticipated

to be generally used in the cases where the 'managed' will always be

'false'. The 'use-schema' approach is expected to be most useful

in the case where all LNEs share the same schema. When 'use-schema'

is used with an LNE mount point, the YANG library rooted in the LNE's

mount point MUST match the associated schema defined within the ietf-

yang-schema-mount module.

Beyond the two modules that will always be present for an LNE, as an

LNE is a network device itself, all modules that may be present at

the top level network device MAY also be present for the LNE. The

list of available modules is expected to be implementation dependent.

As is the method used by an implementation to support LNEs.

Appendix B provide example uses of LNEs.

4. Security Considerations

LNE information represents device and network configuration

information. As such, the security of this information is important,

but it is fundamentally no different than any other interface or

device configuration information that has already been covered in

other documents such as [RFC7223], [RFC7317] and [RFC8022].

The vulnerable "config true" parameters and subtrees are the

following:

/logical-network-elements/logical-network-element: This list

specifies the logical network element and the related logical

device configuration.

Berger, et al. Expires March 30, 2018 [Page 8]

Internet-Draft YANG LNEs September 2017

/logical-network-elements/logical-network-element/managed: While

this leaf is contained in the previous list, it is worth

particular attention as it controls whether information under the

LNE mount point is accessible by both the host device and within

the LNE context. There may be extra sensitivity to this leaf in

environments where an LNE is managed by a different party than the

host device, and that party does not wish to share LNE information

with the operator of the host device.

/if:interfaces/if:interface/bind-lne-name: This leaf indicates the

LNE instance to which an interface is assigned.

Unauthorized access to any of these lists can adversely affect the

security of both the local device and the network. This may lead to

network malfunctions, delivery of packets to inappropriate

destinations, and other problems.

5. IANA Considerations

This document registers a URI in the IETF XML registry [RFC3688].

Following the format in RFC 3688, the following registration is

requested to be made.

URI: urn:ietf:params:xml:ns:yang:ietf-logical-network-element

Registrant Contact: The IESG.

XML: N/A, the requested URI is an XML namespace.

This document registers a YANG module in the YANG Module Names

registry [RFC6020].

name: ietf-logical-network-element

namespace: urn:ietf:params:xml:ns:yang:ietf-logical-network-element

prefix: lne

reference: RFC XXXX

6. Logical Network Element Model

The structure of the model defined in this document is described by

the YANG module below.

<CODE BEGINS> file "ietf-logical-network-element@2017-09-27.yang"

module ietf-logical-network-element {

yang-version 1.1;

// namespace

Berger, et al. Expires March 30, 2018 [Page 9]

Internet-Draft YANG LNEs September 2017

namespace "urn:ietf:params:xml:ns:yang:ietf-logical-network-element";

prefix lne;

// import some basic types

import ietf-interfaces {

prefix if;

reference "RFC 7223: A YANG Data Model for Interface Management";

}

import ietf-yang-schema-mount {

prefix yangmnt;

reference "draft-ietf-netmod-schema-mount: YANG Schema Mount";

// RFC Ed.: Please replace this draft name with the corresponding

// RFC number

}

organization

"IETF Routing Area (rtgwg) Working Group";

contact

"WG Web: <http://tools.ietf.org/wg/rtgwg/>

WG List: <mailto:rtgwg@ietf.org>

Author: Lou Berger

<mailto:lberger@labn.net>

Author: Christan Hopps

<mailto:chopps@chopps.org>

Author: Acee Lindem

<mailto:acee@cisco.com>

Author: Dean Bogdanovic

<mailto:ivandean@gmail.com>";

description

"This module is used to support multiple logical network

elements on a single physical or virtual system.

Copyright (c) 2017 IETF Trust and the persons

identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or

without modification, is permitted pursuant to, and subject

to the license terms contained in, the Simplified BSD License

set forth in Section 4.c of the IETF Trust's Legal Provisions

Relating to IETF Documents

(http://trustee.ietf.org/license-info).

This version of this YANG module is part of RFC XXXX; see

the RFC itself for full legal notices.";

// RFC Ed.: replace XXXX with actual RFC number and remove

Berger, et al. Expires March 30, 2018 [Page 10]

Internet-Draft YANG LNEs September 2017

// this note

// RFC Ed.: please update TBD

revision 2017-09-27 {

description

"Initial revision.";

reference "RFC TBD";

}

// top level device definition statements

container logical-network-elements {

description

"Allows a network device to support multiple logical

network element (device) instances.";

list logical-network-element {

key "name";

description

"List of logical network elements.";

leaf name {

type string;

description

"Device-wide unique identifier for the

logical network element.";

}

leaf managed {

type boolean;

default "true";

description

"True if the host can access LNE information

using the root mount point. This value

my not be modifiable in all implementations.";

}

leaf description {

type string;

description

"Description of the logical network element.";

}

container "root" {

description

"Container for mount point.";

yangmnt:mount-point "root" {

description

"Root for models supported per logical

network element. This mount point

may or may not be inline based on the server

implementation. It SHALL always contain a YANG

library and interfaces instance.

Berger, et al. Expires March 30, 2018 [Page 11]

Internet-Draft YANG LNEs September 2017

When the associated 'managed' leaf is 'false' any

operation that attempts to access information below

the root SHALL fail with an error-tag of

'access-denied' and an error-app-tag of

'lne-not-managed'.";

}

}

}

}

// augment statements

augment "/if:interfaces/if:interface" {

description

"Add a node for the identification of the logical network

element associated with an interface. Applies to interfaces

that can be assigned on a per logical network element basis.

Note that a standard error will be returned if the

identified leafref isn't present. If an interfaces cannot

be assigned for any other reason, the operation SHALL fail

with an error-tag of 'operation-failed' and an error-app-tag

of 'lne-assignment-failed'. A meaningful error-info that

indicates the source of the assignment failure SHOULD also

be provided.";

leaf bind-lne-name {

type leafref {

path "/logical-network-elements/logical-network-element/name";

}

description

"Logical network element ID to which interface is bound.";

}

}

// notification statements

notification bind-lne-name-failed {

description

"Indicates an error in the association of an interface to an

LNE. Only generated after success is initially returned when

bind-lne-name is set.";

leaf name {

type leafref {

path "/if:interfaces/if:interface/if:name";

}

mandatory true;

description

"Contains the interface name associated with the

Berger, et al. Expires March 30, 2018 [Page 12]

Internet-Draft YANG LNEs September 2017

failure.";

}

leaf bind-lne-name {

type leafref {

path "/if:interfaces/if:interface/lne:bind-lne-name";

}

mandatory true;

description

"Contains the bind-lne-name associated with the

failure.";

}

leaf error-info {

type string;

description

"Optionally, indicates the source of the assignment

failure.";

}

}

}

<CODE ENDS>

7. References

7.1. Normative References

[I-D.ietf-netmod-schema-mount]

Bjorklund, M. and L. Lhotka, "YANG Schema Mount", draft-

ietf-netmod-schema-mount-06 (work in progress), July 2017.

[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>.

[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>.

[RFC7223] Bjorklund, M., "A YANG Data Model for Interface

Management", RFC 7223, DOI 10.17487/RFC7223, May 2014,

<https://www.rfc-editor.org/info/rfc7223>.

Berger, et al. Expires March 30, 2018 [Page 13]

Internet-Draft YANG LNEs September 2017

7.2. Informative References

[I-D.ietf-netmod-yang-tree-diagrams]

Bjorklund, M. and L. Berger, "YANG Tree Diagrams", draft-

ietf-netmod-yang-tree-diagrams-01 (work in progress), June

2017.

[I-D.ietf-rtgwg-device-model]

Lindem, A., Berger, L., Bogdanovic, D., and C. Hopps,

"Network Device YANG Logical Organization", draft-ietf-

rtgwg-device-model-02 (work in progress), March 2017.

[I-D.ietf-rtgwg-ni-model]

Berger, L., Hopps, C., Lindem, A., Bogdanovic, D., and X.

Liu, "YANG Network Instances", draft-ietf-rtgwg-ni-

model-03 (work in progress), July 2017.

[RFC7317] Bierman, A. and M. Bjorklund, "A YANG Data Model for

System Management", RFC 7317, DOI 10.17487/RFC7317, August

2014, <https://www.rfc-editor.org/info/rfc7317>.

[RFC7895] Bierman, A., Bjorklund, M., and K. Watsen, "YANG Module

Library", RFC 7895, DOI 10.17487/RFC7895, June 2016,

<https://www.rfc-editor.org/info/rfc7895>.

[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>.

[RFC8022] Lhotka, L. and A. Lindem, "A YANG Data Model for Routing

Management", RFC 8022, DOI 10.17487/RFC8022, November

2016, <https://www.rfc-editor.org/info/rfc8022>.

Appendix A. Acknowledgments

The Routing Area Yang Architecture design team members included Acee

Lindem, Anees Shaikh, Christian Hopps, Dean Bogdanovic, Lou Berger,

Qin Wu, Rob Shakir, Stephane Litkowski, and Yan Gang. Useful review

comments were also received by Martin Bjorklund and John Scudder.

This document was motivated by, and derived from,

[I-D.ietf-rtgwg-device-model].

The RFC text was produced using Marshall Rose's xml2rfc tool.

Berger, et al. Expires March 30, 2018 [Page 14]

Internet-Draft YANG LNEs September 2017

Appendix B. Examples

The following subsections provide example uses of LNEs.

B.1. Example: Host Device Managed LNE

This section describes an example of the LNE model using schema mount

to achieve the parent management. An example device supports

multiple instances of LNEs (logical routers), each of which supports

features of layer 2 and layer 3 interfaces [RFC7223], routing

information base [RFC8022], and OSPF protocol. Each of these

features is specified by a YANG model, and they are combined using

YANG Schema Mount as follows:

module: ietf-logical-network-element

+--rw logical-network-elements

+--rw logical-network-element\* [name]

+--rw name string

+--mp root

+--ro yanglib:modules-state/

| +--ro module-set-id string

| +--ro module\* [name revision]

| +--ro name yang:yang-identifier

+--rw sys:system/

| +--rw contact? string

| +--rw hostname? inet:domain-name

| +--rw authentication {authentication}?

| +--rw user-authentication-order\* identityref

| +--rw user\* [name] {local-users}?

| +--rw name string

| +--rw password? ianach:crypt-hash

| +--rw authorized-key\* [name]

| +--rw name string

| +--rw algorithm string

| +--rw key-data binary

+--ro sys:system-state/

| ...

+--ro rt:routing-state/

| +--ro router-id? yang:dotted-quad

| +--ro control-plane-protocols

| +--ro control-plane-protocol\* [type name]

| +--ro ospf:ospf/

| +--ro instance\* [af]

| ...

+--rw rt:routing/

| +--rw router-id? yang:dotted-quad

| +--rw control-plane-protocols

| +--rw control-plane-protocol\* [type name]

Berger, et al. Expires March 30, 2018 [Page 15]

Internet-Draft YANG LNEs September 2017

| +--rw ospf:ospf/

| +--rw instance\* [af]

| +--rw areas

| +--rw area\* [area-id]

| +--rw interfaces

| +--rw interface\* [name]

| +--rw name if:interface-ref

| +--rw cost? uint16

+--rw if:interfaces/

| +--rw interface\* [name]

| +--rw name string

| +--rw ip:ipv4!/

| | +--rw address\* [ip]

| | ...

+--ro if:interfaces-state/

+--ro interface\* [name]

+--ro name string

+--ro ip:ipv4!/

| +--ro address\* [ip]

| ...

module: ietf-interfaces

+--rw interfaces

| +--rw interface\* [name]

| +--rw name string

| +--rw lne:bind-lne-name? string

+--ro interfaces-state

+--ro interface\* [name]

+--ro name string

+--ro oper-status enumeration

module: ietf-yang-library

+--ro modules-state

+--ro module-set-id string

+--ro module\* [name revision]

+--ro name yang:yang-identifier

module: ietf-system

+--rw system

| +--rw contact? string

| +--rw hostname? inet:domain-name

| +--rw authentication {authentication}?

| +--rw user-authentication-order\* identityref

| +--rw user\* [name] {local-users}?

| +--rw name string

| +--rw password? ianach:crypt-hash

| +--rw authorized-key\* [name]

| +--rw name string

Berger, et al. Expires March 30, 2018 [Page 16]

Internet-Draft YANG LNEs September 2017

| +--rw algorithm string

| +--rw key-data binary

+--ro system-state

+--ro platform

| +--ro os-name? string

| +--ro os-release? string

To realize the above schema, the example device implements the

following schema mount instance:

"ietf-yang-schema-mount:schema-mounts": {

"mount-point": [

{

"module": "ietf-logical-network-element",

"name": "root",

"use-schema": [

{

"name": "lne-schema"

}

]

}

],

"schema": [

{

"name": "lne-schema",

"module": [

{

"name": "ietf-yang-library",

"revision": "2016-06-21",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-library",

"conformance-type": "implement"

},

{

"name": "ietf-system",

"revision": "2014-08-06",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-system",

"conformance-type": "implement"

},

{

"name": "ietf-routing",

"revision": "2016-11-04",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-routing",

"conformance-type": "implement"

},

{

Berger, et al. Expires March 30, 2018 [Page 17]

Internet-Draft YANG LNEs September 2017

"name": "ietf-ospf",

"revision": "2017-03-12",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ospf",

"conformance-type": "implement"

},

{

"name": "ietf-interfaces",

"revision": "2014-05-08",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-interfaces",

"conformance-type": "implement"

},

{

"name": "ietf-ip",

"revision": "2014-06-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ip",

"conformance-type": "implement"

}

]

}

]

}

By using the implementation of the YANG schema mount, an operator can

create instances of logical routers. An interface can be assigned to

a logical router, so that the logical router has the permission to

access this interface. The OSPF protocol can then be enabled on this

assigned interface.

For this implementation, a parent management session has access to

the schemas of both the parent hosting system and the child logical

routers. In addition, each child logical router can grant its own

management sessions, which have the following schema:

module: ietf-yang-library

+--ro modules-state

+--ro module-set-id string

+--ro module\* [name revision]

+--ro name yang:yang-identifier

module: ietf-system

+--rw system

| +--rw contact? string

| +--rw hostname? inet:domain-name

| +--rw authentication {authentication}?

| +--rw user-authentication-order\* identityref

Berger, et al. Expires March 30, 2018 [Page 18]

Internet-Draft YANG LNEs September 2017

| +--rw user\* [name] {local-users}?

| +--rw name string

| +--rw password? ianach:crypt-hash

| +--rw authorized-key\* [name]

| +--rw name string

| +--rw algorithm string

| +--rw key-data binary

+--ro system-state

+--ro platform

| +--ro os-name? string

| +--ro os-release? string

module: ietf-routing

+--ro routing-state

| +--ro router-id? yang:dotted-quad

| +--ro control-plane-protocols

| | +--ro control-plane-protocol\* [type name]

| | +--ro ospf:ospf/

| | +--ro instance\* [af]

+--rw routing

+--rw router-id? yang:dotted-quad

+--rw control-plane-protocols

+--rw control-plane-protocol\* [type name]

+--rw ospf:ospf/

+--rw instance\* [af]

+--rw areas

+--rw area\* [area-id]

+--rw interfaces

+--rw interface\* [name]

+--rw name if:interface-ref

+--rw cost? uint16

module: ietf-interfaces

+--rw interfaces

| +--rw interface\* [name]

| +--rw name string

+--ro interfaces-state

+--ro interface\* [name]

+--ro name string

+--ro oper-status enumeration

B.1.1. Configuration Data

The following shows an example where two customer specific LNEs are

configured:

{

"ietf-logical-network-element:logical-network-elements": {

Berger, et al. Expires March 30, 2018 [Page 19]

Internet-Draft YANG LNEs September 2017

"logical-network-element": [

{

"name": "cust1",

"root": {

"ietf-system:system": {

"authentication": {

"user": [

{

"name": "john",

"password": "$0$password"

}

]

}

},

"ietf-routing:routing": {

"router-id": "192.0.2.1",

"control-plane-protocols": {

"control-plane-protocol": [

{

"type": "ietf-routing:ospf",

"name": "1",

"ietf-ospf:ospf": {

"instance": [

{

"af": "ipv4",

"areas": {

"area": [

{

"area-id": "203.0.113.1",

"interfaces": {

"interface": [

{

"name": "eth1",

"cost": 10

}

]

}

}

]

}

}

]

}

}

]

}

},

"ietf-interfaces:interfaces": {

Berger, et al. Expires March 30, 2018 [Page 20]

Internet-Draft YANG LNEs September 2017

"interfaces": {

"interface": [

{

"name": "eth1",

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.11",

"prefix-length": 24,

}

]

}

}

]

}

}

}

},

{

"name": "cust2",

"root": {

"ietf-system:system": {

"authentication": {

"user": [

{

"name": "john",

"password": "$0$password"

}

]

}

}

"ietf-routing:routing": {

"router-id": "192.0.2.2",

"control-plane-protocols": {

"control-plane-protocol": [

{

"type": "ietf-routing:ospf",

"name": "1",

"ietf-ospf:ospf": {

"instance": [

{

"af": "ipv4",

"areas": {

"area": [

{

"area-id": "203.0.113.1",

"interfaces": {

"interface": [

Berger, et al. Expires March 30, 2018 [Page 21]

Internet-Draft YANG LNEs September 2017

{

"name": "eth1",

"cost": 10

}

]

}

}

]

}

}

]

}

}

]

}

}

"ietf-interfaces:interfaces": {

"interfaces": {

{

"name": "eth1",

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.11",

"prefix-length": 24,

}

]

}

}

]

}

}

}

]

},

"ietf-interfaces:interfaces": {

"interfaces": {

"interface": [

{

"name": "eth0",

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.10",

"prefix-length": 24,

}

]

Berger, et al. Expires March 30, 2018 [Page 22]

Internet-Draft YANG LNEs September 2017

}

},

{

"name": "cust1:eth1",

"lne:bind-lne-name": "cust1"

},

{

"name": "cust2:eth1",

"lne:bind-lne-name": "cust2"

}

]

}

},

"ietf-system:system": {

"authentication": {

"user": [

{

"name": "root",

"password": "$0$password"

}

]

}

}

}

B.1.2. State Data

The following shows possible state data associated the above

configuration data:

{

"ietf-logical-network-element:logical-network-elements": {

"logical-network-element": [

{

"name": "cust1",

"root": {

"ietf-yang-library:modules-state": {

"module-set-id": "123e4567-e89b-12d3-a456-426655440000",

"module": [

{

"name": "iana-if-type",

"revision": "2014-05-08",

"namespace":

"urn:ietf:params:xml:ns:yang:iana-if-type",

"conformance-type": "import"

},

{

Berger, et al. Expires March 30, 2018 [Page 23]

Internet-Draft YANG LNEs September 2017

"name": "ietf-inet-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-inet-types",

"conformance-type": "import"

},

{

"name": "ietf-interfaces",

"revision": "2014-05-08",

"feature": [

"arbitrary-names",

"pre-provisioning"

],

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-interfaces",

"conformance-type": "implement"

},

{

"name": "ietf-ip",

"revision": "2014-06-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ip",

"conformance-type": "implement"

},

{

"name": "ietf-ospf",

"revision": "2017-03-12",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ospf",

"conformance-type": "implement"

},

{

"name": "ietf-routing",

"revision": "2016-11-04",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-routing",

"conformance-type": "implement"

},

{

"name": "ietf-system",

"revision": "2014-08-06",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-system",

"conformance-type": "implement"

},

{

"name": "ietf-yang-library",

"revision": "2016-06-21",

Berger, et al. Expires March 30, 2018 [Page 24]

Internet-Draft YANG LNEs September 2017

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-library",

"conformance-type": "implement"

},

{

"name": "ietf-yang-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-types",

"conformance-type": "import"

}

]

}

"ietf-system:system-state": {

"ietf-system:system-state": {

"platform": {

"os-name": "NetworkOS"

}

}

},

"ietf-routing:routing-state": {

"router-id": "192.0.2.1",

"control-plane-protocols": {

"control-plane-protocol": [

{

"type": "ietf-routing:ospf",

"name": "1",

"ietf-ospf:ospf": {

"instance": [

{

"af": "ipv4",

"areas": {

"area": [

{

"area-id": "203.0.113.1",

"interfaces": {

"interface": [

{

"name": "eth1",

"cost": 10

}

]

}

}

]

}

}

]

Berger, et al. Expires March 30, 2018 [Page 25]

Internet-Draft YANG LNEs September 2017

}

}

]

}

},

"ietf-interfaces:interfaces-state": {

"interfaces": {

"interface": [

{

"name": "eth1",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C1",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

},

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.11",

"prefix-length": 24,

}

]

}

}

]

}

}

}

},

{

"name": "cust2",

"root": {

"ietf-yang-library:modules-state": {

"module-set-id": "123e4567-e89b-12d3-a456-426655440000",

"module": [

{

"name": "iana-if-type",

"revision": "2014-05-08",

"namespace":

"urn:ietf:params:xml:ns:yang:iana-if-type",

"conformance-type": "import"

},

{

"name": "ietf-inet-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-inet-types",

Berger, et al. Expires March 30, 2018 [Page 26]

Internet-Draft YANG LNEs September 2017

"conformance-type": "import"

},

{

"name": "ietf-interfaces",

"revision": "2014-05-08",

"feature": [

"arbitrary-names",

"pre-provisioning"

],

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-interfaces",

"conformance-type": "implement"

},

{

"name": "ietf-ip",

"revision": "2014-06-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ip",

"conformance-type": "implement"

},

{

"name": "ietf-ospf",

"revision": "2017-03-12",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ospf",

"conformance-type": "implement"

},

{

"name": "ietf-routing",

"revision": "2016-11-04",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-routing",

"conformance-type": "implement"

},

{

"name": "ietf-system",

"revision": "2014-08-06",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-system",

"conformance-type": "implement"

},

{

"name": "ietf-yang-library",

"revision": "2016-06-21",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-library",

"conformance-type": "implement"

},

Berger, et al. Expires March 30, 2018 [Page 27]

Internet-Draft YANG LNEs September 2017

{

"name": "ietf-yang-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-types",

"conformance-type": "import"

}

]

}

"ietf-system:system-state": {

"ietf-system:system-state": {

"platform": {

"os-name": "NetworkOS"

}

}

},

"ietf-routing:routing-state": {

"router-id": "192.0.2.2",

"control-plane-protocols": {

"control-plane-protocol": [

{

"type": "ietf-routing:ospf",

"name": "1",

"ietf-ospf:ospf": {

"instance": [

{

"af": "ipv4",

"areas": {

"area": [

{

"area-id": "203.0.113.1",

"interfaces": {

"interface": [

{

"name": "eth1",

"cost": 10

}

]

}

}

]

}

}

]

}

}

]

}

Berger, et al. Expires March 30, 2018 [Page 28]

Internet-Draft YANG LNEs September 2017

}

"ietf-interfaces:interfaces-state": {

"interfaces": {

{

"name": "eth1",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C2",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

},

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.11",

"prefix-length": 24,

}

]

}

}

]

}

}

}

]

},

"ietf-interfaces:interfaces-state": {

"interfaces": {

"interface": [

{

"name": "eth0",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C0",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

},

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.10",

"prefix-length": 24,

}

]

}

},

{

Berger, et al. Expires March 30, 2018 [Page 29]

Internet-Draft YANG LNEs September 2017

"name": "cust1:eth1",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C1",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

}

},

{

"name": "cust2:eth1",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C2",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

}

}

]

}

},

"ietf-yang-library:modules-state": {

"module-set-id": "123e4567-e89b-12d3-a456-426655440000",

"module": [

{

"name": "iana-if-type",

"revision": "2014-05-08",

"namespace":

"urn:ietf:params:xml:ns:yang:iana-if-type",

"conformance-type": "import"

},

{

"name": "ietf-inet-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-inet-types",

"conformance-type": "import"

},

{

"name": "ietf-interfaces",

"revision": "2014-05-08",

"feature": [

"arbitrary-names",

"pre-provisioning"

],

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-interfaces",

"conformance-type": "implement"

Berger, et al. Expires March 30, 2018 [Page 30]

Internet-Draft YANG LNEs September 2017

},

{

"name": "ietf-ip",

"revision": "2014-06-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ip",

"conformance-type": "implement"

},

{

"name": "ietf-logical-network-element",

"revision": "2017-03-13",

"feature": [

"bind-lne-name"

],

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-logical-network-element",

"conformance-type": "implement"

},

{

"name": "ietf-ospf",

"revision": "2017-03-12",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ospf",

"conformance-type": "implement"

},

{

"name": "ietf-routing",

"revision": "2016-11-04",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-routing",

"conformance-type": "implement"

},

{

"name": "ietf-system",

"revision": "2014-08-06",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-system",

"conformance-type": "implement"

},

{

"name": "ietf-yang-library",

"revision": "2016-06-21",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-library",

"conformance-type": "implement"

},

{

"name": "ietf-yang-schema-mount",

Berger, et al. Expires March 30, 2018 [Page 31]

Internet-Draft YANG LNEs September 2017

"revision": "2017-05-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-schema-mount",

"conformance-type": "implement"

},

{

"name": "ietf-yang-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-types",

"conformance-type": "import"

}

]

},

"ietf-system:system-state": {

"platform": {

"os-name": "NetworkOS"

}

}

}

B.2. Example: Self Managed LNE

This section describes an example of the LNE model using schema mount

to achieve child independent management. An example device supports

multiple instances of LNEs (logical routers), each of them has the

features of layer 2 and layer 3 interfaces [RFC7223], routing

information base [RFC8022], and the OSPF protocol. Each of these

features is specified by a YANG model, and they are put together by

the YANG Schema Mount as following:

Berger, et al. Expires March 30, 2018 [Page 32]

Internet-Draft YANG LNEs September 2017

module: ietf-logical-network-element

+--rw logical-network-elements

+--rw logical-network-element\* [name]

+--rw name string

+--mp root

// The internal modules of the LNE are not visible to

// the parament management.

// The child manages its modules, including ietf-routing

// and ietf-interfaces

module: ietf-interfaces

+--rw interfaces

| +--rw interface\* [name]

| +--rw name string

| +--rw lne:bind-lne-name? string

+--ro interfaces-state

+--ro interface\* [name]

+--ro name string

+--ro oper-status enumeration

module: ietf-yang-library

+--ro modules-state

+--ro module-set-id string

+--ro module\* [name revision]

+--ro name yang:yang-identifier

module: ietf-system

+--rw system

| +--rw contact? string

| +--rw hostname? inet:domain-name

| +--rw authentication {authentication}?

| +--rw user-authentication-order\* identityref

| +--rw user\* [name] {local-users}?

| +--rw name string

| +--rw password? ianach:crypt-hash

| +--rw authorized-key\* [name]

| +--rw name string

| +--rw algorithm string

| +--rw key-data binary

+--ro system-state

+--ro platform

| +--ro os-name? string

| +--ro os-release? string

To realize the above schema, the device implements the following

schema mount instance:

Berger, et al. Expires March 30, 2018 [Page 33]

Internet-Draft YANG LNEs September 2017

"ietf-yang-schema-mount:schema-mounts": {

"mount-point": [

{

"module": "ietf-logical-network-element",

"name": "root",

"inline": [null]

}

]

}

By using the implementation of the YANG schema mount, an operator can

create instances of logical routers, each with their logical router

specific in-line modules. An interface can be assigned to a logical

router, so that the logical router has the permission to access this

interface. The OSPF protocol can then be enabled on this assigned

interface. Each logical router independently manages its own set of

modules, which may or may not be the same as other logical routers.

The following is an example of schema set implemented on one

particular logical router:

module: ietf-yang-library

+--ro modules-state

+--ro module-set-id string

+--ro module\* [name revision]

+--ro name yang:yang-identifier

module: ietf-system

+--rw system

| +--rw contact? string

| +--rw hostname? inet:domain-name

| +--rw authentication {authentication}?

| +--rw user-authentication-order\* identityref

| +--rw user\* [name] {local-users}?

| +--rw name string

| +--rw password? ianach:crypt-hash

| +--rw authorized-key\* [name]

| +--rw name string

| +--rw algorithm string

| +--rw key-data binary

+--ro system-state

+--ro platform

| +--ro os-name? string

| +--ro os-release? string

module: ietf-routing

+--ro routing-state

| +--ro router-id? yang:dotted-quad

| +--ro control-plane-protocols

Berger, et al. Expires March 30, 2018 [Page 34]

Internet-Draft YANG LNEs September 2017

| | +--ro control-plane-protocol\* [type name]

| | +--ro ospf:ospf/

| | +--ro instance\* [af]

+--rw routing

+--rw router-id? yang:dotted-quad

+--rw control-plane-protocols

+--rw control-plane-protocol\* [type name]

+--rw ospf:ospf/

+--rw instance\* [af]

+--rw areas

+--rw area\* [area-id]

+--rw interfaces

+--rw interface\* [name]

+--rw name if:interface-ref

+--rw cost? uint16

module: ietf-interfaces

+--rw interfaces

| +--rw interface\* [name]

| +--rw name string

+--ro interfaces-state

+--ro interface\* [name]

+--ro name string

+--ro oper-status enumeration

B.2.1. Configuration Data

Each of the child virtual routers is managed through its own sessions

and configuration data.

B.2.1.1. Logical Network Element 'vnf1'

The following shows an example configuration data for the LNE name

"vnf1":

{

"ietf-system:system": {

"authentication": {

"user": [

{

"name": "john",

"password": "$0$password"

}

]

}

},

"ietf-routing:routing": {

"router-id": "192.0.2.1",

Berger, et al. Expires March 30, 2018 [Page 35]

Internet-Draft YANG LNEs September 2017

"control-plane-protocols": {

"control-plane-protocol": [

{

"type": "ietf-routing:ospf",

"name": "1",

"ietf-ospf:ospf": {

"instance": [

{

"af": "ipv4",

"areas": {

"area": [

{

"area-id": "203.0.113.1",

"interfaces": {

"interface": [

{

"name": "eth1",

"cost": 10

}

]

}

}

]

}

}

]

}

}

]

}

},

"ietf-interfaces:interfaces": {

"interfaces": {

"interface": [

{

"name": "eth1",

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.11",

"prefix-length": 24,

}

]

}

}

]

}

}

Berger, et al. Expires March 30, 2018 [Page 36]

Internet-Draft YANG LNEs September 2017

}

B.2.1.2. Logical Network Element 'vnf2'

The following shows an example configuration data for the LNE name

"vnf2":

{

"ietf-system:system": {

"authentication": {

"user": [

{

"name": "john",

"password": "$0$password"

}

]

}

},

"ietf-routing:routing": {

"router-id": "192.0.2.2",

"control-plane-protocols": {

"control-plane-protocol": [

{

"type": "ietf-routing:ospf",

"name": "1",

"ietf-ospf:ospf": {

"instance": [

{

"af": "ipv4",

"areas": {

"area": [

{

"area-id": "203.0.113.1",

"interfaces": {

"interface": [

{

"name": "eth1",

"cost": 10

}

]

}

}

]

}

}

]

}

}

Berger, et al. Expires March 30, 2018 [Page 37]

Internet-Draft YANG LNEs September 2017

]

}

},

"ietf-interfaces:interfaces": {

"interfaces": {

"interface": [

{

"name": "eth1",

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.11",

"prefix-length": 24,

}

]

}

}

]

}

}

}

B.2.2. State Data

The following sections shows possible state data associated the above

configuration data. Note that there are three views: the host

device's, and each LNE's.

B.2.2.1. Host Device

The following shows state data for the device hosting the example

LNEs:

{

"ietf-logical-network-element:logical-network-elements": {

"logical-network-element": [

{

"name": "vnf1",

"root": {

}

},

{

"name": "vnf2",

"root": {

}

}

]

},

Berger, et al. Expires March 30, 2018 [Page 38]

Internet-Draft YANG LNEs September 2017

"ietf-interfaces:interfaces-state": {

"interfaces": {

"interface": [

{

"name": "eth0",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C0",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

},

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.10",

"prefix-length": 24,

}

]

}

},

{

"name": "vnf1:eth1",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C1",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

}

},

{

"name": "vnf2:eth2",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C2",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

}

}

]

}

},

"ietf-yang-library:modules-state": {

"module-set-id": "123e4567-e89b-12d3-a456-426655440000",

"module": [

{

"name": "iana-if-type",

"revision": "2014-05-08",

Berger, et al. Expires March 30, 2018 [Page 39]

Internet-Draft YANG LNEs September 2017

"namespace":

"urn:ietf:params:xml:ns:yang:iana-if-type",

"conformance-type": "import"

},

{

"name": "ietf-inet-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-inet-types",

"conformance-type": "import"

},

{

"name": "ietf-interfaces",

"revision": "2014-05-08",

"feature": [

"arbitrary-names",

"pre-provisioning"

],

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-interfaces",

"conformance-type": "implement"

},

{

"name": "ietf-ip",

"revision": "2014-06-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ip",

"conformance-type": "implement"

},

{

"name": "ietf-logical-network-element",

"revision": "2017-03-13",

"feature": [

"bind-lne-name"

],

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-logical-network-element",

"conformance-type": "implement"

},

{

"name": "ietf-system",

"revision": "2014-08-06",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-system",

"conformance-type": "implement"

},

{

"name": "ietf-yang-library",

Berger, et al. Expires March 30, 2018 [Page 40]

Internet-Draft YANG LNEs September 2017

"revision": "2016-06-21",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-library",

"conformance-type": "implement"

},

{

"name": "ietf-yang-schema-mount",

"revision": "2017-05-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-schema-mount",

"conformance-type": "implement"

},

{

"name": "ietf-yang-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-types",

"conformance-type": "import"

}

]

},

"ietf-system:system-state": {

"platform": {

"os-name": "NetworkOS"

}

}

}

B.2.2.2. Logical Network Element 'vnf1'

The following shows state data for the example LNE with name "vnf1":

{

"ietf-yang-library:modules-state": {

"module-set-id": "123e4567-e89b-12d3-a456-426655440000",

"module": [

{

"name": "iana-if-type",

"revision": "2014-05-08",

"namespace":

"urn:ietf:params:xml:ns:yang:iana-if-type",

"conformance-type": "import"

},

{

"name": "ietf-inet-types",

"revision": "2013-07-15",

"namespace":

Berger, et al. Expires March 30, 2018 [Page 41]

Internet-Draft YANG LNEs September 2017

"urn:ietf:params:xml:ns:yang:ietf-inet-types",

"conformance-type": "import"

},

{

"name": "ietf-interfaces",

"revision": "2014-05-08",

"feature": [

"arbitrary-names",

"pre-provisioning"

],

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-interfaces",

"conformance-type": "implement"

},

{

"name": "ietf-ip",

"revision": "2014-06-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ip",

"conformance-type": "implement"

},

{

"name": "ietf-ospf",

"revision": "2017-03-12",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ospf",

"conformance-type": "implement"

},

{

"name": "ietf-routing",

"revision": "2016-11-04",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-routing",

"conformance-type": "implement"

},

{

"name": "ietf-system",

"revision": "2014-08-06",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-system",

"conformance-type": "implement"

},

{

"name": "ietf-yang-library",

"revision": "2016-06-21",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-library",

"conformance-type": "implement"

Berger, et al. Expires March 30, 2018 [Page 42]

Internet-Draft YANG LNEs September 2017

},

{

"name": "ietf-yang-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-types",

"conformance-type": "import"

}

]

},

"ietf-system:system-state": {

"platform": {

"os-name": "NetworkOS"

}

},

"ietf-routing:routing-state": {

"router-id": "192.0.2.1",

"control-plane-protocols": {

"control-plane-protocol": [

{

"type": "ietf-routing:ospf",

"name": "1",

"ietf-ospf:ospf": {

"instance": [

{

"af": "ipv4",

"areas": {

"area": [

{

"area-id": "203.0.113.1",

"interfaces": {

"interface": [

{

"name": "eth1",

"cost": 10

}

]

}

}

]

}

}

]

}

}

]

Berger, et al. Expires March 30, 2018 [Page 43]

Internet-Draft YANG LNEs September 2017

}

},

"ietf-interfaces:interfaces-state": {

"interfaces": {

"interface": [

{

"name": "eth1",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C1",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

},

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.11",

"prefix-length": 24,

}

]

}

}

]

}

}

}

B.2.2.3. Logical Network Element 'vnf2'

The following shows state data for the example LNE with name "vnf2":

{

"ietf-yang-library:modules-state": {

"module-set-id": "123e4567-e89b-12d3-a456-426655440000",

"module": [

{

"name": "iana-if-type",

"revision": "2014-05-08",

"namespace":

"urn:ietf:params:xml:ns:yang:iana-if-type",

"conformance-type": "import"

},

{

"name": "ietf-inet-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-inet-types",

Berger, et al. Expires March 30, 2018 [Page 44]

Internet-Draft YANG LNEs September 2017

"conformance-type": "import"

},

{

"name": "ietf-interfaces",

"revision": "2014-05-08",

"feature": [

"arbitrary-names",

"pre-provisioning"

],

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-interfaces",

"conformance-type": "implement"

},

{

"name": "ietf-ip",

"revision": "2014-06-16",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ip",

"conformance-type": "implement"

},

{

"name": "ietf-ospf",

"revision": "2017-03-12",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-ospf",

"conformance-type": "implement"

},

{

"name": "ietf-routing",

"revision": "2016-11-04",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-routing",

"conformance-type": "implement"

},

{

"name": "ietf-system",

"revision": "2014-08-06",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-system",

"conformance-type": "implement"

},

{

"name": "ietf-yang-library",

"revision": "2016-06-21",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-library",

"conformance-type": "implement"

},

Berger, et al. Expires March 30, 2018 [Page 45]

Internet-Draft YANG LNEs September 2017

{

"name": "ietf-yang-types",

"revision": "2013-07-15",

"namespace":

"urn:ietf:params:xml:ns:yang:ietf-yang-types",

"conformance-type": "import"

}

]

},

"ietf-system:system-state": {

"platform": {

"os-name": "NetworkOS"

}

},

"ietf-routing:routing-state": {

"router-id": "192.0.2.2",

"control-plane-protocols": {

"control-plane-protocol": [

{

"type": "ietf-routing:ospf",

"name": "1",

"ietf-ospf:ospf": {

"instance": [

{

"af": "ipv4",

"areas": {

"area": [

{

"area-id": "203.0.113.1",

"interfaces": {

"interface": [

{

"name": "eth1",

"cost": 10

}

]

}

}

]

}

}

]

}

}

]

}

Berger, et al. Expires March 30, 2018 [Page 46]

Internet-Draft YANG LNEs September 2017

},

"ietf-interfaces:interfaces-state": {

"interfaces": {

"interface": [

{

"name": "eth1",

"type": "iana-if-type:ethernetCsmacd",

"oper-status": "up",

"phys-address": "00:01:02:A1:B1:C2",

"statistics": {

"discontinuity-time": "2017-06-26T12:34:56-05:00"

},

"ip:ipv4": {

"address": [

{

"ip": "192.0.2.11",

"prefix-length": 24,

}

]

}

}

]

}

}

}

Authors' Addresses

Lou Berger

LabN Consulting, L.L.C.

Email: lberger@labn.net

Christan Hopps

Deutsche Telekom

Email: chopps@chopps.org

Acee Lindem

Cisco Systems

301 Midenhall Way

Cary, NC 27513

USA

Email: acee@cisco.com

Berger, et al. Expires March 30, 2018 [Page 47]

Internet-Draft YANG LNEs September 2017

Dean Bogdanovic

Email: ivandean@gmail.com

Xufeng Liu

Jabil

Email: Xufeng\_Liu@jabil.com

Berger, et al. Expires March 30, 2018 [Page 48]