

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 30, 2016

Z. Wang
L. Xia
Huawei
D. Kumar
Cisco
Q. Wu
Huawei

November 27, 2015

LIME base model extension for BFD Support
draft-wang-yang-bfd-oam-06

Abstract

This document discusses how LIME base model is applied to BFD and present an example of YANG Data model for BFD support. The YANG Model presented in this document extends the technology independent YANG model for OAM with BFD technology specifics.

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 <http://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 May 30, 2016.

Copyright Notice

Copyright (c) 2015 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 (<http://trustee.ietf.org/license-info>) in effect on the date of 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

Internet-Draft

YANG LIME BFD OAM

November 2015

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 |
| 2. | Conventions and Terminology | 2 |
| 2.1. | Tree Diagrams | 3 |
| 3. | Interaction between BFD OAM YANG model and LIME base model . | 4 |
| 4. | Generic YANG Model extension for BFD | 5 |
| 4.1. | MD Level configuration extension | 5 |
| 4.1.1. | Technology Type Extension | 5 |
| 4.1.2. | Sub Technology Type Extension | 5 |
| 4.2. | MA configuration extension | 6 |
| 4.2.1. | Connectivity-Context Extension | 6 |
| 4.3. | MEP configuration extension | 6 |
| 4.3.1. | Session Configuration Extension | 7 |
| 4.3.2. | Interface configuration extension | 7 |
| 4.3.3. | New Notification definition | 8 |
| 5. | Model structure of LIME model extension for BFD | 8 |
| 6. | OAM YANG Module | 10 |
| 7. | Security Considerations | 14 |
| 8. | IANA Considerations | 14 |
| 9. | Normative References | 14 |
| | Authors' Addresses | 15 |

[1.](#) Introduction

[[draft-ietf-bfd-yang](#)] defines a YANG data model that can be used to configure and manage Bidirectional Forwarding Detection (BFD). This document discusses how LIME base model is applied to BFD and present an example of YANG Data model for BFD support. The YANG Model example presented in this document extends the Generic YANG model for OAM defined in [I-D.ietf-lime-yang-oam-model]. The YANG model example uses the grouping defined in the BFD model [[draft-ietf-bfd-yang](#)]. The groupings contain the basic BFD session parameters for applications to use.

[2.](#) Conventions and 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](#)].

The following terms are defined in [[RFC6241](#)] and are not redefined here:

- o client
- o configuration data
- o server
- o state data

The following terms are defined in [[RFC6020](#)] and are not redefined here:

- o augment
- o data model
- o data node

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

[2.1](#). Tree Diagrams

A simplified graphical representation of the data model is used in this document. The meaning of the symbols in these diagrams is as follows:

Each node is printed as:

<status> <flags> <name> <opts> <type>

<status> is one of:

- + for current
- x for deprecated
- o for obsolete

<flags> is one of:

- rw for configuration data
- ro for non-configuration data
- x for rpcs
- n for notifications

<name> is the name of the node

If the node is augmented into the tree from another module, its name is printed as <prefix>:<name>.

<opts> is one of:

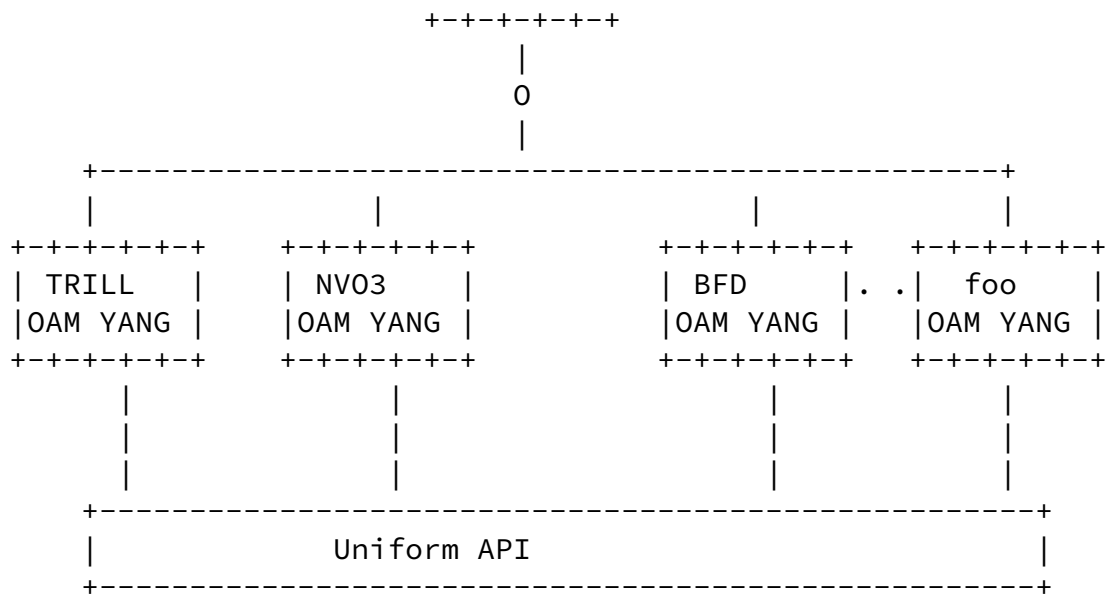
- ? for an optional leaf or choice
- ! for a presence container
- * for a leaf-list or list
- [<keys>] for a list's keys

<type> is the name of the type for leafs and leaf-lists

[3.](#) Interaction between BFD OAM YANG model and LIME base model

Both BFD model and LIME model can be developed as generic model. LIME model can be extended for BFD to provide consistent configuration, representation and reporting. Therefore LIME model extension for BFD can be viewed as a BFD application. The consistent configuration and representation can be met by extending LIME configuration structure while the consistent reporting and representation can be met by extending LIME RPC structure or Notification Structure. Generic YANG model for OAM defined in [I-D.ietf-lime-yang-oam-model] can also be used as the basis for all the other OAM YANG models. This allows users to span across OAM tools of different technologies through a uniform API. The following Figure depicts the relationship of BFD OAM YANG model to the Layer Independent OAM YANG Model.

```
+---+
| Layer |
|independent|
|OAM YANG |
```



Relationship of BFD OAM YANG model to technology independent OAM YANG model

[4. Generic YANG Model extension for BFD](#)

[4.1. MD Level configuration extension](#)

MD level configuration parameters are management information which can be inherited in the BFD model and set by LIME base model as default values. For example domain name can be set to area-ID in the BFD case. In addition, at the Maintenance Domain level, domain data node at root level can be augmented with technology type and sub-technology type.

[4.1.1. Technology Type Extension](#)

No BFD technology type has been defined in the LIME base model. Therefore a technology type extension is required in the BFD OAM model. The technology type "bfd" is defined as an identity that augments the base "technology-types" defined in the LIME base model:

[4.1.2. Sub Technology Type Extension](#)

In BFD, since different encapsulation types such as IP/UDP Encapsulation, PW-ACH encapsulation can be employed.

In lime-bfd-extension yang data model, we define an identity: "technology-sub-type" to further identify the encapsulation types within the BFD. And based on it, we also define four identity encapsulation types:

- o technology-sub-type-sh-udp: technology sub-type is single hop with IP/UDP encapsulation;
- o technology-sub-type-mh-udp: technology sub-type is multiple hop with IP/UDP encapsulation;
- o technology-sub-type-sh-ach: technology sub-type is single hop with PW-ACH encapsulation;
- o technology-sub-type-mh-ach: technology sub-type is multiple hop with PW-ACH encapsulation;

In MD level, we define a sub-technology leaf with an identityref type which base on the technology-sub-type:

```
augment "/goam:domains/goam:domain/" {  
    leaf sub-technology{  
        type identityref {  
            base technology-sub-type;  
        }  
    }  
}
```

[4.2.](#) MA configuration extension

MA level configuration parameters (e.g., MA Name) are management information which can be inherited in the BFD model and set by LIME base model as default values. One example of MA Name is Tunnel Name or LAG Name. In addition, at the Maintenance Association(MA) level, MA data node at the second level can be augmented with connectivity-

context extension.

[4.2.1.](#) Connectivity-Context Extension

In BFD context-id is a 32bit local discriminator. The LIME base model defines a placeholder for context-id. This allows other technologies to easily augment that to include technology specific extensions. The snippet below depicts an example of augmenting context-id to include local discriminator.

```
augment "/goam:domains/goam:domain/goam:MA/goam:MA
/goam:connectivity-context"
{
    case connectivity-context-bfd {
        leaf local-discriminator {
            type local-discriminator;
        }
    }
}
```

[4.3.](#) MEP configuration extension

In BFD, the MEP address is either an IPv4 or IPV6 address. MEP-ID is either a 2 octet unsigned integer value or a variable length label value. In the LIME base model, MEP-ID is defined as a variable length label value and the same definition can be used for BFD with no further modification. In addition, at the Maintenance Association Endpoint(MEP) level, MEP data node at the third level can be augmented with Session extension and interface extension.

[4.3.1.](#) Session Configuration Extension

At the Session level, Session data node at the fourth level can be augmented with 3 interval parameters and 2 TTL parameters. The Session Configuration extension should reuse grouping defined in [\[draft-ietf-bfd-yang\]](#) for session related parameters. In [\[draft-ietf-bfd-yang\]](#), source and destination address in the bfd-session-cfg can be corresponding to Session configuration extension as source MEP

and destination MEP.

```
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session:
+--rw (interval-config-type)?
|  +--:(tx-rx-intervals)
|  |  +--rw desired-min-tx-interval      uint32
|  |  +--rw required-min-rx-interval     uint32
|  +--:(single-interval)
|      +--rw min-interval                 uint32

augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session:
+--rw tx-ttl?                            ttl
+--rw rx-ttl                             ttl
```

[4.3.2.](#) Interface configuration extension

At the Interface level, interface data node at the fifth level can be augmented with the same parameters defined in per-interface configuration of [[draft-ietf-bfd-yang](#)]. Interface configuration extension should reuse grouping defined in [[draft-ietf-bfd-yang](#)] for interface related parameters.

```
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session/goam:
+--rw local-multiplier?                  multiplier
+--rw (interval-config-type)?
|  +--:(tx-rx-intervals)
|  |  +--rw desired-min-tx-interval      uint32
|  |  +--rw required-min-rx-interval     uint32
|  +--:(single-interval)
|      +--rw min-interval                 uint32
+--rw demand-enabled?                   boolean
+--rw enable-authentication?            boolean
+--rw authentication-parms {bfd-authentication}?
|  +--rw key-chain-name?                string
|  +--rw algorithm?                    bfd-auth-algorithm
+--rw desired-min-echo-tx-interval?     uint32
+--rw required-min-echo-rx-interval?    uint32
```

[4.3.3.](#) New Notification definition

[GENYANGOAM] defines a notification model which abstracts defects notification in a technology independent manner. However what BFD is required is state change notification, therefore a new notification definition can be specified to meet BFD requirement. The new notification definition should reuse groupings defined in [[draft-ietf-bfd-yang](#)] for state change related parameters.

notifications:

```
+---n state-change-notification
  +--ro local-discriminator?      uint32
  +--ro remote-discriminator?    uint32
  +--ro new-state?                enumeration
  +--ro state-change-reason?     string
  +--ro time-in-previous-state?  string
  +--ro dest-addr?               inet:ip-address
  +--ro source-addr?             inet:ip-address
  +--ro session-cookie?          leafref
  +--ro technology-sub-type?     identityref
  +--ro interface?               leafref
  +--ro echo-enabled?            boolean
```

In this state-change-notification, technology-sub-type is used to identify whether the notification is for single hop or multi-hop or other types.

[5.](#) Model structure of LIME model extension for BFD

The complete data hierarchy related to the OAM YANG model is presented below.

```
module: lime-bfd-extension
augment /goam:domains/goam:domain/goam:MA/goam:MA:
  +--rw technology-sub-type?  identityref
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session:
  +--rw on-demand-enable?      boolean
  +--rw local-multiplier?      uint8
  +--rw bfd-tx-rx-interval
  | +--rw desired-min-tx-interval?  uint32
  | +--rw required-min-rx-interval?  uint32
  +--rw enable-authentication?  boolean
  +--rw bfd-authentication {bfd-authentication}?
    +--rw key-chain-name?  string
    +--rw algorithm?      enumeration
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session:
  +--rw tx-ttl?  uint8
  +--rw rx-ttl?  uint8
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session/goam:connectivity-context-bfd:
  +--:(connectivity-context-bfd)
    +--rw local-discriminator?  uint32
    +--rw remote-discriminator?  uint32
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session/goam:connectivity-context-bfd:
  +--rw on-demand-enable?      boolean
  +--rw local-multiplier?      uint8
  +--rw bfd-tx-rx-interval
  | +--rw desired-min-tx-interval?  uint32
  | +--rw required-min-rx-interval?  uint32
  +--rw enable-authentication?  boolean
  +--rw bfd-authentication {bfd-authentication}?
  | +--rw key-chain-name?  string
  | +--rw algorithm?      enumeration
  +--rw desired-min-echo-tx-interval?  uint32
  +--rw required-min-echo-rx-interval?  uint32
notifications:
  +---n state-change-notification
    +--ro local-discriminator?  uint32
    +--ro remote-discriminator?  uint32
    +--ro new-state?            enumeration
    +--ro state-change-reason?  string
    +--ro time-in-previous-state?  string
    +--ro dest-addr?            inet:ip-address
    +--ro source-addr?          inet:ip-address
    +--ro session-cookie?       leafref
    +--ro technology-sub-type?  identityref
    +--ro interface?           leafref
    +--ro echo-enabled?         boolean
```

6. OAM YANG Module

```
<CODE BEGINS> file "ietf-lime-bfd-extension.yang"
module iETF-lime-bfd-extension{
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-lime-bfd-extension";
  prefix limebfd;

  import iETF-gen-oam {
    prefix goam;
  }

  import iETF-bfd {
    prefix bfd;
  }

  import iETF-interfaces {
    prefix if;
  }

  organization
    "IETF BFD Working Group";
  contact
    "WG List: <mailto:bfd@ietf.org>
    Editor:";

  description
    "This YANG Model extends the technology independent
    YANG model for OAM with BFD technology specifics.";

  revision 2014-08-30 {
    description
      "Initial revision.";
    reference "";
  }

  identity bfd{
    base goam:technology-types;
    description
```

```

        "bfd type";
    }

    identity technology-sub-type {
        description
        "certain implementations such as bfd can have different
        encapsulation types such as ip/udp, pw-ach and so on.
        Instead of defining separate models for each
        encapsulation, we define a technology sub-type to

```

```

        further identify different encapsulations. Technology
        sub-type is associated at the MA level";
    }

    identity technology-sub-type-sh-udp {
        base technology-sub-type;
        description
        "technology sub-type is single
        hop with IP/UDP encapsulation";
    }

    identity technology-sub-type-mh-udp {
        base technology-sub-type;
        description
        "technology sub-type is multiple
        hop with IP/UDP encapsulation";
    }

    identity technology-sub-type-sh-ach {
        base technology-sub-type;
        description
        "technology sub-type is single
        hop with PW-ACH encapsulation";
    }

    identity technology-sub-type-mh-ach {
        base technology-sub-type;
        description
        "technology sub-type is multiple hop
        with PW-ACH encapsulation";
    }

```

```

grouping tx-rx-ttl{
  description
    "bfd tx ttl";

  leaf tx-ttl{
    type uint8;
    description
      "tx ttl.";
  }
  leaf rx-ttl{
    type uint8;
    description
      "rx ttl.";
  }
}

```

```

feature bfd-authentication {
  description "BFD authentication supported";
}

augment "/goam:domains/goam:domain/goam:MA/goam:MA"{
  when "goam:technology = 'bfd'"
  {description
    "when goam:technology = bfd.";}
    leaf technology-sub-type {
      type identityref {
        base technology-sub-type;
      }
      description
        "technology sub-type such as
        single hop udp, multiple hop udp,
        single hop ach, multiple hop ach.";
    }
    description
      "augment the MA with bfd parameters.";
  }

  augment "/goam:domains/goam:domain/goam:MA/goam:MA"
  +"/goam:MEP/goam:session"{
    when "goam:technology = 'bfd'"

```

```

        {description
        "when goam:technology = bfd.";}
    leaf admin-down {
        type boolean;
        default false;
        description
        "Is the BFD session administratively down";
    }

    uses bfd:bfd-grouping-common-cfg-parms;

    description
    "augment the session with bfd parameters.";
}

augment "/goam:domains/goam:domain/goam:MA/goam:MA"
+ "/goam:MEP/goam:session"{
    when "technology-sub-type = 'technology-sub-type-mh-udp'
    or 'technology-sub-type-mh-ach'"{
        description
        "when technology sub type = techonlogy sub type mh udp
        or technology sub type = technology-sub-type-mh-ach.";
    }
}

```

```

    uses tx-rx-ttl;
    description
    "augment the session with bfd parameters.";
}

augment "/goam:domains/goam:domain/goam:MA/goam:MA"
+ "/goam:MEP/goam:session/goam:connectivity-context"{
    when "goam:technology = 'bfd'"{
        description
        "when goam:techolgoy =bfd.";}
        case connectivity-context-bfd {
            leaf local-discriminator {
                type uint32;
            }
            description
            "local discriminator";
        }
    }
    leaf remote-discriminator{

```

```

    type uint32;

    description
    "remote-discriminator";
}
    }
        description
        "augment the connectivity-context with
        bfd parameters.";
    }

augment "/goam:domains/goam:domain/goam:MA/goam:MA"
+ "/goam:MEP/goam:session/goam:outgoing-interface"{
    when "technology-sub-type = 'technology-sub-type-sh-udp'
    or 'technology-sub-type-sh-ach'"{
        description
        "when technology-sub-type = 'technology-sub-type-sh-udp'
        or 'technology-sub-type-sh-ach.'";
    }
uses bfd:bfd-grouping-common-cfg-parms;
uses bfd:bfd-grouping-echo-cfg-parms;
    description
    "augment the outgoing interface with bfd parameters.";
}
notification state-change-notification
{
    uses bfd:bfd-notification-parms;
    leaf interface {
        type if:interface-ref;
        description "Interface to which this BFD session belongs to";
    }
}

```

```

    }
    leaf echo-enabled {
        type boolean;
        description "Was echo enabled for BFD";
    }
    description
    "state change notification.";
}
}

```

<CODE ENDS>

7. Security Considerations

TBD.

8. IANA Considerations

TBD.

9. Normative References

- [I-D.tissa-lime-yang-oam-model]
Senevirathne, T., Finn, N., Kumar, D., Salam, S., Wu, Q.,
and Z. Wang, "Generic YANG Data Model for Operations,
Administration, and Maintenance (OAM)", [draft-tissa-lime-
yang-oam-model-06](#) (work in progress), August 2015.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", March 1997.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection
(BFD)", [RFC 5880](#), DOI 10.17487/RFC5880, June 2010,
<<http://www.rfc-editor.org/info/rfc5880>>.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for
the Network Configuration Protocol (NETCONF)", [RFC 6020](#),
DOI 10.17487/RFC6020, October 2010,
<<http://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,
<<http://www.rfc-editor.org/info/rfc6241>>.

- [RFC7331] Nadeau, T., Ali, Z., and N. Akiya, "Bidirectional
Forwarding Detection (BFD) Management Information Base",
[RFC 7331](#), DOI 10.17487/RFC7331, August 2014,
<<http://www.rfc-editor.org/info/rfc7331>>.

Authors' Addresses

Zitao Wang
Huawei Technologies, Co., Ltd
101 Software Avenue, Yuhua District
Nanjing 210012
China

Email: wangzitao@huawei.com

Liang Xia
Huawei Technologies, Co., Ltd
101 Software Avenue, Yuhua District
Nanjing 210012
China

Email: frank.xialiang@huawei.com

Deepak Kumar
Cisco Systems
510 McCarthy Blvd Milpitas,
CA 95035
USA

Email: dekumar@cisco.com

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

Email: bill.wu@huawei.com