

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: July 12, 2017

R. Gu
China Mobile
L. Xia
Z. Wang
Huawei Technologies
D. Kumar
Cisco Systems
M. Boucadair
France Telecom
January 8, 2017

YANG Data Model for SFC Operations, Administration, and Maintenance
(OAM)
draft-gu-sfc-yang-oam-00

Abstract

This document defines YANG data model for Service Function Chaining (SFC Operations, Administration, and Maintenance). It derives from the basic YANG data model for Layer independent OAM Management defined in [[I-D.ietf-lime-yang-connectionless-oam](#)] with SFC technology specifics. It includes SFC OAM related configuration and state data.

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 July 12, 2017.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

Internet-Draft

SFC OAM YANG Model

January 2017

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 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	3
2.1.	Terminologies	4
3.	Architecture of OAM YANG Model and Relationship with SFC OAM	5
4.	SFC OAM YANG DATA MODEL OVERVIEW	5
4.1.	SFC type extensions	6
4.2.	Configuration Model Extension	6
4.3.	RPC Extension	7
5.	SFC OAM YANG Data Hierarchy	8
6.	SFC OAM YANG Module	10
7.	Security Considerations	17
8.	IANA Considerations	17
9.	References	17
9.1.	Normative References	17
9.2.	Informative References	17
	Authors' Addresses	18

[1.](#) Introduction

YANG [[RFC6020](#)] is a data modeling language used to model configuration and state data manipulated by the Network Configuration Protocol (NETCONF) [[RFC6241](#)], NETCONF remote procedure calls (RPC), and NETCONF notifications. This document defines the YANG data model for Service Function Chaining (SFC) OAM [[I-D.ietf-sfc-oam-framework](#)]. The SFC OAM YANG module involves the OAM configuration, RPCs and notifications, etc.

Currently, [[I-D.ietf-lime-yang-connectionless-oam](#)] proposes a basic YANG data model for Layer independent OAM Management that can be applied to various OAM technologies. SFC OAM YANG data model can be defined by directly extending the basic model with SFC technology

specifics. It can bring some obvious benefits such as unified format, reusable parts, and correlation of defects, faults, network failure at the specific layer.

In addition, various components in the SFC technology specific YANG data model defined in [[I-D.penno-sfc-yang](#)] can be directly reused in this draft to define the SFC OAM YANG data model.

Note that SFC OAM mechanisms are not yet defined or standardized although some of the basic concepts and functions (e.g., fault detection, fault localization, performance measurement, etc) may be similar to traditional OAM mechanisms. This draft should get alignment with the latest development SFC OAM mechanisms.

[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 [\[RFC6020\]](#).

The following notations are used within the data tree and carry the meaning as noted below.

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

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying [RFC-2119](#) significance.

2.1. Terminologies

OAM Operations, Administration, and Maintenance [[RFC6291](#)]

LIME Layer Independent OAM Management [I-D.ietf-lime-yang-oam-model]

SF Service Function [[I-D.penno-sfc-yang](#)]

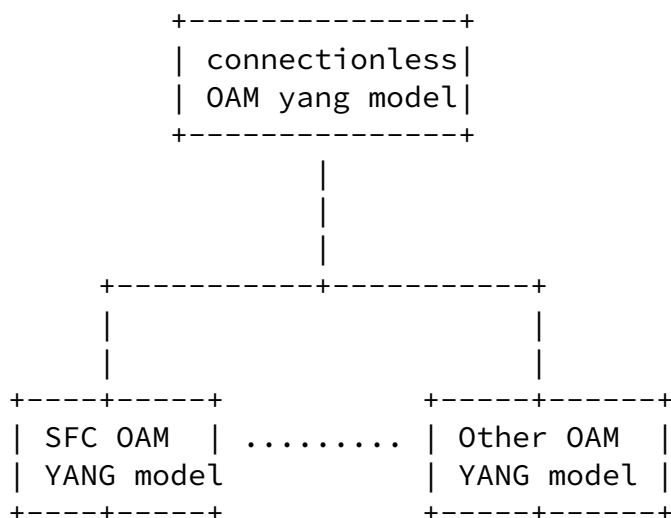
SFC Service Function Chaining [[I-D.penno-sfc-yang](#)]

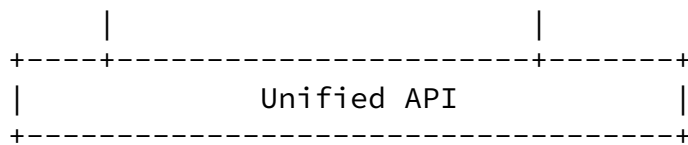
SFF Service Function Forwarder [[I-D.penno-sfc-yang](#)]

RPC Remote Process Call

3. Architecture of OAM YANG Model and Relationship with SFC OAM

Layer independent OAM YANG model [I-D.ietf-lime-yang-connectionless-oam] is 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 SFC OAM YANG model with the Connectionless OAM YANG Model.





Relationship of SFC OAM YANG model to Layer independent OAM YANG model

4. SFC OAM YANG DATA MODEL OVERVIEW

There are two main blocks in SFC-OAM YANG data model: Configuration and RPCs. The SFC configuration model is derived from "ietf-connectionless-oam" YANG data model [I-D.ietf-lime-yang-connectionless-oam]. The specific attributes of SFC OAM are defined and inserted into proper anchor point of "ietf-connectionless-oam". Therefore, the monitoring of SFC components, such as SF, SFF, SFP, classifier, etc, can be supported.

Meanwhile, the OAM function models, such as continuity check, trace route, etc, are provided by extending the "ietf-connectionless-oam-methods" [[draft-ietf-lime-yang-connectionless-oam-methods](#)].

4.1. SFC type extensions

A new Technology parameter of SFC is defined here for the purpose of identifying the SFC specific YANG model extension:

```

augment "/nd:networks/nd:network/nd:node/coam:location-type"
  +"/coam:ipv4-location-type/coam:test-point-ipv4-location-list"
  +"/coam:test-point-locations/coam:technology/coam:technology-string"{
    leaf sfc-oam{
      type string;
      description
        "sfc oam type";
    }
    description
      "augment the connectionless oam model to derive service

```

```

    functon oam yang model";
}

augment "/nd:networks/nd:network/nd:node/coam:location-type"
+ "/coam:ipv6-location-type/coam:test-point-ipv6-location-list"
+ "/coam:test-point-locations/coam:technology/coam:technology-string"{
    leaf sfc-oam{
        type string;
        description
            "sfc oam type";
    }
    description
        "augment the connectionless oam model to derive service
        functon oam yang model";
}

```

SFC type extension

Only when the Technology parameter is set to the "SFC" value, the SFC specific extensions are applied.

[4.2.](#) Configuration Model Extension

The following section demonstrates how the "ietf-connectionless-oam" model can be extended to support service function chain OAM. For this purpose, SFC OAM specific attributes are defined and inserted into the ipv4/ipv6-test point location.

```

module: ietf-sfc-oam
    augment /nd:networks/nd:network/nd:node/coam:location-type/coam:ipv4-location-
type/coam:test-point-ipv4-location-list/coam:test-point-locations/coam:technolo
y/coam:technology-string:
    +---rw sfc-oam?    string
augment /nd:networks/nd:network/nd:node/coam:location-type/coam:ipv4-location-t
    +---rw sf-name?           -> /sfc-sf:service-functions/service
    +---rw sf-type?    sfc-common:sft-type-name

```

```

+--rw sfc-name?                -> /sfc-sfc:service-function-chains/
+--rw sfp-name?                -> /sfc-sfp:service-function-paths/s
+--rw sf-data-plane-locator* [locator-name]
    +--rw locator-name          -> /sfc-sf:service-functions/service
    +--rw service-function-forwarder? sfc-common:sff-name

```

Configuration Model Extension

[4.3.](#) RPC Extension

The rpc model facilitates issuing commands to a NETCONF server (in this case to the device that needs to execute the OAM command) and obtaining a response. The Continuity-check, continuity-verification, and traceroute are defined in "ietf-connectionless-oam-methods" [[draft-ietf-lime-yang-connectionless-oam-methods](#)]. The following section demonstrates the input extensions for SFC-OAM.

```
augment /coam-methods:continuity-check/coam-methods:input/coam-methods:destina
```



```

tion-tp/coam-methods:tp-address:
    +---:(sfc-address)
        +----- sf-name?                -> /sfc-sf:service-functions/service
function/name
        +----- sfc-name?                -> /sfc-sfc:service-function-chains/
ervice-function-chain/name
        +----- sfp-name?                -> /sfc-sfp:service-function-paths/s
ervice-function-path/name
        +----- classifier?              -> /sfc-sfp:service-function-paths/s
ervice-function-path/classifier
        +----- service-function-forwarder? -> /sfc-sff:service-function-forward
rs/service-function-forwarder/name
augment /coam-methods:path-discovery/coam-methods:input/coam-methods:destinati
on-tp/coam-methods:tp-address:
    +---:(sfc-address)
        +----- sf-name?                -> /sfc-sf:service-functions/service
function/name
        +----- sfc-name?                -> /sfc-sfc:service-function-chains/
ervice-function-chain/name
        +----- sfp-name?                -> /sfc-sfp:service-function-paths/s
ervice-function-path/name
        +----- classifier?              -> /sfc-sfp:service-function-paths/s
ervice-function-path/classifier
        +----- service-function-forwarder? -> /sfc-sff:service-function-forward
rs/service-function-forwarder/name

```

RPC Extension

5. SFC OAM YANG Data Hierarchy

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

```

module: ietf-sfc-oam
augment /nd:networks/nd:network/nd:node/coam:location-type/coam:ipv4-location-
type/coam:test-point-ipv4-location-list/coam:test-point-locations/coam:technolo
y/coam:technology-string:
    +---rw sfc-oam?    string
    augment /nd:networks/nd:network/nd:node/coam:location-type/coam:ipv4-location
type/coam:test-point-ipv4-location-list/coam:test-point-locations:
        +---rw sf-name?                -> /sfc-sf:service-functions/service-functio
/name
        +---rw sf-type?                sfc-common:sft-type-name
        +---rw sfc-name?                -> /sfc-sfc:service-function-chains/service-
unction-chain/name
        +---rw sfp-name?                -> /sfc-sfp:service-function-paths/service-f
nction-path/name

```

```

    +---rw sf-data-plane-locator* [locator-name]
      +---rw locator-name                -> /sfc-sf:service-functions/service-
function/sf-data-plane-locator/name
      +---rw service-function-forwarder? sfc-common:sff-name
    augment /nd:networks/nd:network/nd:node/coam:location-type/coam:ipv6-location
type/coam:test-point-ipv6-location-list/coam:test-point-locations/coam:technolo
y/coam:technology-string:
      +---rw sfc-oam? string
    augment /nd:networks/nd:network/nd:node/coam:location-type/coam:ipv6-location
type/coam:test-point-ipv6-location-list/coam:test-point-locations:
      +---rw sf-name?                    -> /sfc-sf:service-functions/service-functio
/name
      +---rw sf-type?                    sfc-common:sft-type-name
      +---rw sfc-name?                    -> /sfc-sfc:service-function-chains/service-
unction-chain/name
      +---rw sfp-name?                    -> /sfc-sfp:service-function-paths/service-f
nction-path/name
      +---rw classifier?                  -> /sfc-sfp:service-function-paths/service-f
nction-path/classifier
      +---rw sf-data-plane-locator* [locator-name]
        +---rw locator-name                -> /sfc-sf:service-functions/service-
function/sf-data-plane-locator/name
        +---rw service-function-forwarder? -> /sfc-sff:service-function-forward
rs/service-function-forwarder/name

```

RPCs:

```

augment /coam-methods:continuity-check/coam-methods:input/coam-methods:destina
tion-tp/coam-methods:tp-address:
  +---:(sfc-address)
    +---- sf-name?                        -> /sfc-sf:service-functions/service-
function/name
    +---- sfc-name?                      -> /sfc-sfc:service-function-chains/
ervice-function-chain/name
    +---- sfp-name?                      -> /sfc-sfp:service-function-paths/s
rvice-function-path/name
    +---- classifier?                    -> /sfc-sfp:service-function-paths/s
rvice-function-path/classifier
    +---- service-function-forwarder?    -> /sfc-sff:service-function-forward
rs/service-function-forwarder/name
  augment /coam-methods:path-discovery/coam-methods:input/coam-methods:destinat
on-tp/coam-methods:tp-address:
    +---:(sfc-address)
      +---- sf-name?                      -> /sfc-sf:service-functions/service-
function/name
      +---- sfc-name?                      -> /sfc-sfc:service-function-chains/

```

ervice-function-chain/name

+---- sfp-name?

-> /sfc-sfp:service-function-paths/s

Gu, et al.

Expires July 12, 2017

[Page 9]

Internet-Draft

SFC OAM YANG Model

January 2017

ervice-function-path/name

+---- classifier?

-> /sfc-sfp:service-function-paths/s

ervice-function-path/classifier

+---- service-function-forwarder?

-> /sfc-sff:service-function-forward

rs/service-function-forwarder/name

Data hierarchy of SFC OAM

[6.](#) SFC OAM YANG Module

```
<CODE BEGINS> file "ietf-sfc-oam@2016-11-21.yang"
module ietf-sfc-oam {
  namespace "urn:ietf:params:xml:ns:yang:ietf-sfc-oam";
  prefix sfc-oam;

  import ietf-network{
    prefix nd;
  }

  import ietf-connectionless-oam{
    prefix "coam";
  }

  import ietf-connectionless-oam-methods{
    prefix "coam-methods";
  }

  import service-function-chain{
    prefix "sfc-sfc";
  }
  import sfc-common{
    prefix sfc-common;
  }

  import service-function {
    prefix sfc-sf;
  }
}
```

```

import service-function-forwarder {
    prefix sfc-sff;
}

import service-function-path{
    prefix sfc-sfp;
}

organization "IETF SFC Working Group";

```

Gu, et al.

Expires July 12, 2017

[Page 10]

Internet-Draft

SFC OAM YANG Model

January 2017

```

contact
    "Gu Rong ";

description
    "The YANG module defines a SFC OAM configuration model.";

revision 2016-11-21 {
    description
        "Initial vision. - 00 version";
    reference "";
}

/*service function oam blocks*/

/*test point ipv4 extension*/

/*oam type extension*/

augment "/nd:networks/nd:network/nd:node/coam:location-type"
    +"/coam:ipv4-location-type/coam:test-point-ipv4-location-list"
    +"/coam:test-point-locations/coam:technology/coam:technology-string"{
    leaf sfc-oam{
        type string;
        description
            "sfc oam type";
    }
    description
        "augment the connectionless oam model to derive service
        functon oam yang model";
}

```

```

/*service function*/

augment "/nd:networks/nd:network/nd:node/coam:location-type"
+ "/coam:ipv4-location-type/coam:test-point-ipv4-location-list"
+ "/coam:test-point-locations"{
  leaf sf-name {
    type leafref{
      path "/sfc-sf:service-functions/sfc-sf:service-function"
      + "/sfc-sf:name";
    }
  }
  description
    "The name of the service function.";
}

  leaf sf-type{
type sfc-common:sft-type-name;

```

```

  description
    "The type of the service function";
}
leaf sfc-name {
  type leafref{
    path "/sfc-sfc:service-function-chains"
    + "/sfc-sfc:service-function-chain/sfc-sfc:name";
  }
  description
    "service function chain name";
}
leaf sfp-name {
  type leafref{
    path "/sfc-sfp:service-function-paths"
    + "/sfc-sfp:service-function-path/sfc-sfp:name";
  }
  description
    "Service function path name";
}
list sf-data-plane-locator {
  key "locator-name";
  leaf locator-name{
    type leafref{
      path "/sfc-sf:service-functions/sfc-sf:service-function"
      + "/sfc-sf:sf-data-plane-locator/sfc-sf:name";
    }
  }
}

```

```

    }
    description
    "The sf data plane locator name";
    }

    leaf service-function-forwarder{
        type sfc-common:sff-name;
        description
        "The name of teh service function forwarder";
    }

    description
    "list for service function data plane locator";
    }
    description
    "augment connectionless oam model to derive service
    function oam yang model";
    }

/*test point ipv6 extension*/

/*oam type extension*/

```

```

augment "/nd:networks/nd:network/nd:node/coam:location-type"
+ "/coam:ipv6-location-type/coam:test-point-ipv6-location-list"
+ "/coam:test-point-locations/coam:technology/coam:technology-string"{
    leaf sfc-oam{
        type string;
        description
        "sfc oam type";
    }
    description
    "augment the connectionless oam model to derive service
    functon oam yang model";
    }

/*service function*/

augment "/nd:networks/nd:network/nd:node/coam:location-type"
+ "/coam:ipv6-location-type/coam:test-point-ipv6-location-list"
+ "/coam:test-point-locations"{

```

```

    leaf sf-name {
      type leafref{
        path "/sfc-sf:service-functions/sfc-sf:service-function"
          +"/sfc-sf:name";
      }
    }
    description
      "The name of the service function.";
  }
  leaf sf-type{
type sfc-common:sft-type-name;
    description
      "The type of the service function";
  }
  leaf sfc-name {
    type leafref{
      path "/sfc-sfc:service-function-chains"
        +"/sfc-sfc:service-function-chain/sfc-sfc:name";
    }
    description
      "service function chain name";
  }
  leaf sfp-name {
    type leafref{
      path "/sfc-sfp:service-function-paths"
        +"/sfc-sfp:service-function-path/sfc-sfp:name";
    }
    description
      "Service function path name";
  }
}

```

```

    leaf classifier{
      type leafref{
        path "/sfc-sfp:service-function-paths"
          +"/sfc-sfp:service-function-path/sfc-sfp:classifier";
      }
      description
        "SFC classifier.";
    }
    list sf-data-plane-locator {
      key "locator-name";
      leaf locator-name{

```

```

    type leafref{
      path "/sfc-sf:service-functions/sfc-sf:service-function"
        +"/sfc-sf:sf-data-plane-locator/sfc-sf:name";
    }
    description
    "The sf data plane locator name";
  }
  leaf service-function-forwarder{
    type leafref{
      path "/sfc-sff:service-function-forwarders"
        +"/sfc-sff:service-function-forwarder/sfc-sff:name";
    }
    description
    "The name of teh service function forwarder";
  }
  description
  "List for service function data plane locator";
}
description
"augment connectionless oam model to derive service
function oam yang model";
}

augment "/coam-methods:continuity-check/coam-methods:input"
+ "/coam-methods:destination-tp/coam-methods:tp-address"{
  case sfc-address{

    leaf sf-name {
      type leafref{
        path "/sfc-sf:service-functions/sfc-sf:service-function"
          +"/sfc-sf:name";
      }
      description
      "The name of the service function.";
    }

    leaf sfc-name {

```

```

type leafref{

  path "/sfc-sfc:service-function-chains"
    +"/sfc-sfc:service-function-chain/sfc-sfc:name";

```



```

    }
    description
    "service function chain name";
}

leaf sfp-name {
  type leafref{
    path "/sfc-sfp:service-function-paths"
    +"/sfc-sfp:service-function-path/sfc-sfp:name";
  }
  description
  "Service function path name";
}

leaf classifier{
  type leafref{
    path "/sfc-sfp:service-function-paths"
    +"/sfc-sfp:service-function-path/sfc-sfp:classifier";
  }
  description
  "SFC classifier";
}

leaf service-function-forwarder{
  type leafref{
    path "/sfc-sff:service-function-forwarders"
    +"/sfc-sff:service-function-forwarder/sfc-sff:name";
  }
  description
  "service function forwarder";
}
description
"SFC address";
}

description
"Augment base with SFC address";
}

augment "/coam-methods:path-discovery/coam-methods:input"
+ "/coam-methods:destination-tp/coam-methods:tp-address"{
  case sfc-address{

    leaf sf-name {
      type leafref{
        path "/sfc-sf:service-functions/sfc-sf:service-function"

```

```
        +"/sfc-sf:name";
    }
    description
    "The name of the service function.";
}

leaf sfc-name {
    type leafref{

        path "/sfc-sfc:service-function-chains"
        +"/sfc-sfc:service-function-chain/sfc-sfc:name";
    }
    description
    "service function chain name";
}

leaf sfp-name {
    type leafref{
        path "/sfc-sfp:service-function-paths"
        +"/sfc-sfp:service-function-path/sfc-sfp:name";
    }
    description
    "Service function path name";
}

leaf classifier{
    type leafref{
        path "/sfc-sfp:service-function-paths"
        +"/sfc-sfp:service-function-path/sfc-sfp:classifier";
    }
    description
    "SFC classifier";
}

leaf service-function-forwarder{
    type leafref{
        path "/sfc-sff:service-function-forwarders"
        +"/sfc-sff:service-function-forwarder/sfc-sff:name";
    }
    description
    "Service function forwarder";
}
description
"SFC address";
}

description
"Augment base model with SFC address";
}
```

}

Internet-Draft

SFC OAM YANG Model

January 2017

<CODE ENDS>

[7.](#) Security Considerations

TBD.

[8.](#) IANA Considerations

TBD.

[9.](#) References

[9.1.](#) Normative References

[IEEE.802.1Q-2011]

Institute of Electrical and Electronics Engineers, "Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", IEEE Standard 802.1Q, August 2011.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.

[RFC2234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", [RFC 2234](#), DOI 10.17487/RFC2234, November 1997, <<http://www.rfc-editor.org/info/rfc2234>>.

[9.2.](#) Informative References

[I-D.ietf-lime-yang-connectionless-oam]

Kumar, D., Wang, Z., Wu, Q., Rahman, R., and S. Raghavan, "Generic YANG Data Model for Connectionless Operations, Administration, and Maintenance(OAM) protocols", [draft-ietf-lime-yang-connectionless-oam-03](#) (work in progress), December 2016.

[I-D.ietf-lime-yang-connectionless-oam-methods]

Kumar, D., Wang, Z., Wu, Q., Rahman, R., and S. Raghavan, "Retrieval Methods YANG Data Model for Connectionless Operations, Administration, and Maintenance(OAM) protocols", [draft-ietf-lime-yang-connectionless-oam-](#)

[methods-00](#) (work in progress), October 2016.

[I-D.ietf-sfc-oam-framework]

Aldrin, S., Krishnan, R., Akiya, N., Pignataro, C., and A. Ghanwani, "Service Function Chaining Operation, Administration and Maintenance Framework", [draft-ietf-sfc-oam-framework-01](#) (work in progress), February 2016.

Gu, et al.

Expires July 12, 2017

[Page 17]

Internet-Draft

SFC OAM YANG Model

January 2017

[I-D.penno-sfc-yang]

Penno, R., Quinn, P., Zhou, D., and J. Li, "Yang Data Model for Service Function Chaining", [draft-penno-sfc-yang-15](#) (work in progress), June 2016.

[RFC6291] Andersson, L., van Helvoort, H., Bonica, R., Romascanu, D., and S. Mansfield, "Guidelines for the Use of the "OAM" Acronym in the IETF", [BCP 161](#), [RFC 6291](#), DOI 10.17487/RFC6291, June 2011, <<http://www.rfc-editor.org/info/rfc6291>>.

[Y.1731] "OAM functions and mechanisms for Ethernet based networks", ITU G.8013/Y.1731, July 2011.

Authors' Addresses

Rong Gu
China Mobile
32 Xuanwumen West Ave, Xincheng District
Beijing 100053
China

Email: gurong@chinamobile.com

Liang Xia
Huawei Technologies
101 Software Avenue, Yuhua District
Nanjing
China

Email: frank.xialiang@huawei.com

Zitao Wang
Huawei Technologies
101 Software Avenue, Yuhua District
Nanjing
China

Email: wangzitao@huawei.com

Gu, et al.

Expires July 12, 2017

[Page 18]

Internet-Draft

SFC OAM YANG Model

January 2017

Deepak Kumar
Cisco Systems
510 McCarthy Blvd Milpitas
CA
USA

Email: dekumar@cisco.com

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange.com

