

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 4, 2017

D. Kumar
Cisco
M. Wang
Q. Wu
Huawei
R. Rahman
S. Raghavan
Cisco
October 31, 2016

**Retrieval Methods YANG Data Model for Connectionless Operations,
Administration, and Maintenance(OAM) protocols
draft-ietf-lime-yang-connectionless-oam-methods-00**

Abstract

This document presents a retrieval method YANG Data model for connectionless OAM protocols. It provides a technology-independent RPC commands for connectionless OAM protocols. The retrieval methods model presented here can be extended to include technology specific details. This is leading to uniformity between OAM protocols and support nested OAM workflows (i.e., performing OAM functions at different or same levels through a unified interface).

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 4, 2017.

Copyright Notice

Copyright (c) 2016 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 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 used in this document	3
2.1.	Terminology	3
2.2.	Tree Diagrams	3
3.	Overview of the Connectionless OAM retrieval methods Model .	4
3.1.	RPC definitions	4
3.2.	OAM Retrieval Methods Hierarchy	8
4.	OAM Retrieval Methods YANG Module	16
5.	Security Considerations	20
6.	IANA Considerations	20
7.	Normative References	20
	Authors' Addresses	21

[1.](#) Introduction

Operations, Administration, and Maintenance (OAM) are important networking functions that allow operators to:

1. Monitor networks connections (Reachability Verification, Continuity Check).
2. Troubleshoot failures (Fault verification and localization).
3. Monitor Performance

An overview of OAM tools is presented at [\[RFC7276\]](#).

Ping and Traceroute [\[RFC792\]](#), [\[RFC4443\]](#) are well-known fault verification and isolation tools, respectively, for IP networks. Over the years, different technologies have developed similar tools for similar purposes.

In this document, we present a retrieval method YANG Data model for connectionless OAM protocols. This module provides technology-independent RPC commands for connectionless OAM protocols. It is separated from the generic YANG model for connectionless OAM [lime

base model] and can avoid mixing the models for the retrieved-data from the retrieval procedures. It is expected that retrieval procedures would evolve faster than the data model [lime base model] and will allow new procedures to be defined for retrieval of the same data defined by the base data model.

2. Conventions used in this document

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. Terminology

TP - Test Point

MAC - Media Access Control

RPC - A Remote Procedure Call, as used within the NETCONF protocol

2.2. 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. Overview of the Connectionless OAM retrieval methods Model

In this document, we present a retrieval method YANG Data model for connectionless OAM protocols. This module provides technology-independent retrieval procedures (RPC commands) for connectionless OAM protocols. It will allow the user to flexibility to retrieve the retrieved-data which defined by the base data model[lime base model].

3.1. RPC definitions

The rpc model facilitates issuing commands to a NETCONF server (in this case to the device that need to execute the OAM command) and obtaining a response.

Under connectionless-oam-methods module, we summarize the common OAM functions and define the generic rpc commands: continuity-check and

path-discovery. In practice, these commands are supported by corresponding technology-specific OAM tools [[RFC7276](#)]. For example, for the IP OAM model, the continuity-check rpc corresponds to the IP Ping, while the path-discovery rpc command corresponds to IP Traceroute.

Note that the rpc command presented in this document is the base building block, which is used to derive a model for a technology-specific OAM(i.e., icmp ping, lsp ping), the base building block should be extended with corresponding technology specific parameters. To facilitate this and for future enhancements to data retrieval methods, the RPCs are captured under a separate module.

The generic path-discovery-data and continuity-check-data are used as data outputs from the different RPCs described in the document. Similar methods including other RPCs can retrieve the data using the same data model.

```
rpc continuity-check {
  if-feature coam:continuity-check;
  description
    "Generates continuity-check as per RFC7276";
  input {
    container destination-tp {
      uses coam:tp-address;
      description
        "destination test point.";
    }
    uses coam:session-type;
    leaf source-interface {
      type if:interface-ref;
      description
        "source interface.";
    }
    leaf outbound-interface {
      type if:interface-ref;

      description
        "outbound interface.";
    }
    leaf count {
      type uint32;
      default "5";
      description
        "Specifies the number of packets that will be sent.";
    }
    leaf vrf {
      type coam:routing-instance-ref;
```



```

        description
            "vrf instance.";
    }
    leaf ttl {
        type uint8;
        default "255";
        description
            "Time to live (TTL).";
    }
    leaf packet-size {
        type uint32 {
            range "64..10000";
        }
        default "64";
        description
            "Size of ping echo request packets, in octets";
    }
}
output {
    list error-code-list {
        key "response-index";
        leaf response-index {
            type uint32;
            description
                "response index.";
        }
        leaf status-code {
            type int32;
            description
                "error code is ";
        }
        leaf status-sub-code {
            type uint8;
            description
                "sub code.";
        }
        description
            "error code list.";
    }
}

uses coam:continuity-check-data;
}

rpc path-discovery {
    description
        "Generates path discovery as per RFC7276.";
    input {

```



```

    container destination-tp {
        uses coam:tp-address;
        description
            "destination test point.";
    }
    uses coam:session-type;
    leaf source-interface {
        type if:interface-ref;
        description
            "source interface.";
    }
    leaf outbound-interface {
        type if:interface-ref;
        description
            "outbound interface.";
    }
    leaf vrf {
        type coam:routing-instance-ref;
        description
            "vrf";
    }
    leaf max-ttl {
        type uint8;
        default "255";
        description
            "max ttl.";
    }
}

output {
    list response-list {
        key "response-index";
        description
            "path discovery response list.";
        leaf response-index {
            type uint32;
            description
                "response index.";
        }
        leaf status-code {
            type int32;
            description
                "error code is ";
        }
        leaf status-sub-code {
            type uint8;

            description
                "sub code is ";
        }
    }
}

```



```

    }
  }

  uses coam:path-discovery-data;
}
}

```

Snippet of data hierarchy related to rpc calls

3.2. OAM Retrieval Methods Hierarchy

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

module: ietf-connectionless-oam-methods

rpcs:

```

+---x continuity-check {coam:continuity-check}?
| +---w input
| | +---w destination-tp
| | | +---w tp-address-type-value?  identityref
| | | +---w (tp-address)?
| | |   +--:(mac-address)
| | |   | +---w mac-address?          yang:mac-address
| | |   +--:(ipv4-address)
| | |   | +---w ipv4-address?         inet:ipv4-address
| | |   +--:(ipv6-address)
| | |   | +---w ipv6-address?         inet:ipv6-address
| | |   +--:(src-dst-address)
| | |   | +---w src-ip-address?       inet:ip-address
| | |   | +---w dst-ip-address?      inet:ip-address
| | |   | +---w Interface?           if:interface-ref
| | |   +--:(fec)
| | |   | +---w fec-type?             fec-type
| | |   | +---w (fec-value)?
| | |   |   +--:(ip-prefix)
| | |   |   | +---w ip-prefix?        inet:ip-prefix
| | |   |   +--:(bgp)
| | |   |   | +---w bgp?              inet:ip-prefix
| | |   |   +--:(tunnel)
| | |   |   | +---w tunnel-interface? uint32
| | |   |   +--:(pw)
| | |   |   | +---w remote-pe-address? inet:ip-address
| | |   |   | +---w pw-id?            uint32
| | |   |   +--:(vpls)
| | |   |   | +---w route-distinguisher? uint32
| | |   |   | +---w sender-ve-id?      uint32
| | |   |   | +---w receiver-ve-id?   uint32

```



```

| | | | +--:(mpls-mldp)
| | | | +---w (root-address)?
| | | | +--:(ip-address)
| | | | | +---w source-address? inet:ip-address
| | | | | +---w group-ip-address? IP-Multicast-
Group-Address
| | | | +--:(vpn)
| | | | | +---w as-number? inet:as-number
| | | | +--:(global-id)
| | | | | +---w lsp-id? string
| | | +--:(tlv-address)
| | | | +---w tlv-type? int16
| | | | +---w tlv-len? int16
| | | | +---w tlv-value? binary
| | | +--:(system-info)
| | | | +---w system-id? inet:uri
| | +---w session-type-enum? enumeration
| | +---w source-interface? if:interface-ref
| | +---w outbound-interface? if:interface-ref
| | +---w count? uint32
| | +---w vrf? coam:routing-instance-ref
| | +---w ttl? uint8
| | +---w packet-size? uint32
| +--ro output
| | +--ro error-code-list* [response-index]
| | | +--ro response-index uint32
| | | +--ro status-code? int32
| | | +--ro status-sub-code? uint8
| | +--ro src-test-point
| | | +--ro vrf? routing-instance-ref
| | | +--ro tp-address-type-value? identityref
| | | +--ro (tp-address)?
| | | | +--:(mac-address)
| | | | | +--ro mac-address? yang:mac-address
| | | | +--:(ipv4-address)
| | | | | +--ro ipv4-address? inet:ipv4-address
| | | | +--:(ipv6-address)
| | | | | +--ro ipv6-address? inet:ipv6-address
| | | | +--:(src-dst-address)
| | | | | +--ro src-ip-address? inet:ip-address
| | | | | +--ro dst-ip-address? inet:ip-address
| | | | | +--ro Interface? if:interface-ref
| | | +--:(fec)
| | | | +--ro fec-type? fec-type
| | | | +--ro (fec-value)?
| | | | +--:(ip-prefix)
| | | | | +--ro ip-prefix? inet:ip-prefix
| | | +--:(bgp)

```

| | | | | |--ro bgp?

inet:ip-prefix

Kumar, et al.

Expires May 4, 2017

[Page 9]

				+--:(tunnel)	
				+--ro tunnel-interface?	uint32
				+--:(pw)	
				+--ro remote-pe-address?	inet:ip-address
				+--ro pw-id?	uint32
				+--:(vpls)	
				+--ro route-distinguisher?	uint32
				+--ro sender-ve-id?	uint32
				+--ro receiver-ve-id?	uint32
				+--:(mpls-mldp)	
				+--ro (root-address)?	
				+--:(ip-address)	
				+--ro source-address?	inet:ip-address
				+--ro group-ip-address?	IP-Multicast-
Group-Address					
				+--:(vpn)	
				+--ro as-number?	inet:as-number
				+--:(global-id)	
				+--ro lsp-id?	string
				+--:(tlv-address)	
				+--ro tlv-type?	int16
				+--ro tlv-len?	int16
				+--ro tlv-value?	binary
				+--:(system-info)	
				+--ro system-id?	inet:uri
				+--ro egress-intf-name?	if:interface-ref
				+--ro dest-test-point	
				+--ro vrf?	routing-instance-ref
				+--ro tp-address-type-value?	identityref
				+--ro (tp-address)?	
				+--:(mac-address)	
				+--ro mac-address?	yang:mac-address
				+--:(ipv4-address)	
				+--ro ipv4-address?	inet:ipv4-address
				+--:(ipv6-address)	
				+--ro ipv6-address?	inet:ipv6-address
				+--:(src-dst-address)	
				+--ro src-ip-address?	inet:ip-address
				+--ro dst-ip-address?	inet:ip-address
				+--ro Interface?	if:interface-ref
				+--:(fec)	
				+--ro fec-type?	fec-type
				+--ro (fec-value)?	
				+--:(ip-prefix)	
				+--ro ip-prefix?	inet:ip-prefix
				+--:(bgp)	
				+--ro bgp?	inet:ip-prefix
				+--:(tunnel)	

| | | | | +-ro tunnel-interface? uint32

```

| | | | +--:(pw)
| | | | | +--ro remote-pe-address? inet:ip-address
| | | | | +--ro pw-id? uint32
| | | | +--:(vpls)
| | | | | +--ro route-distinguisher? uint32
| | | | | +--ro sender-ve-id? uint32
| | | | | +--ro receiver-ve-id? uint32
| | | | +--:(mpls-mldp)
| | | | | +--ro (root-address)?
| | | | | +--:(ip-address)
| | | | | | +--ro source-address? inet:ip-address
| | | | | | +--ro group-ip-address? IP-Multicast-
Group-Address
| | | | +--:(vpn)
| | | | | +--ro as-number? inet:as-number
| | | | +--:(global-id)
| | | | | +--ro lsp-id? string
| | | +--:(tlv-address)
| | | | +--ro tlv-type? int16
| | | | +--ro tlv-len? int16
| | | | +--ro tlv-value? binary
| | | +--:(system-info)
| | | | +--ro system-id? inet:uri
| | | | +--ro ingress-intf-name? if:interface-ref
| | +--ro sequence-number? uint64
| | +--ro hop-cnt? uint8
| | +--ro session-packet-statistics
| | | +--ro rx-packet-count? uint32
| | | +--ro tx-packet-count? uint32
| | | +--ro rx-bad-packet? uint32
| | | +--ro tx-packet-failed? uint32
| | +--ro session-error-statistics
| | | +--ro packet-drops-count? uint32
| | | +--ro packet-reorder-count? uint32
| | | +--ro packets-out-of-seq-count? uint32
| | | +--ro packets-dup-count? uint32
| | +--ro session-delay-statistics
| | | +--ro time-resolution-value? identityref
| | | +--ro min-delay-value? uint32
| | | +--ro max-delay-value? uint32
| | | +--ro average-delay-value? uint32
| | +--ro session-jitter-statistics
| | | +--ro time-resolution-value? identityref
| | | +--ro min-jitter-value? uint32
| | | +--ro max-jitter-value? uint32
| | | +--ro average-jitter-value? uint32
+---x path-discovery
+---w input

```

| +---w destination-tp

Kumar, et al.

Expires May 4, 2017

[Page 11]

```

| | +---w tp-address-type-value? identityref
| | +---w (tp-address)?
| |   +---:(mac-address)
| |     | +---w mac-address?          yang:mac-address
| |   +---:(ipv4-address)
| |     | +---w ipv4-address?        inet:ipv4-address
| |   +---:(ipv6-address)
| |     | +---w ipv6-address?        inet:ipv6-address
| |   +---:(src-dst-address)
| |     | +---w src-ip-address?      inet:ip-address
| |     | +---w dst-ip-address?      inet:ip-address
| |     | +---w Interface?          if:interface-ref
| |   +---:(fec)
| |     | +---w fec-type?            fec-type
| |     | +---w (fec-value)?
| |       +---:(ip-prefix)
| |         | +---w ip-prefix?        inet:ip-prefix
| |       +---:(bgp)
| |         | +---w bgp?              inet:ip-prefix
| |       +---:(tunnel)
| |         | +---w tunnel-interface? uint32
| |       +---:(pw)
| |         | +---w remote-pe-address? inet:ip-address
| |         | +---w pw-id?            uint32
| |       +---:(vpls)
| |         | +---w route-distinguisher? uint32
| |         | +---w sender-ve-id?      uint32
| |         | +---w receiver-ve-id?    uint32
| |       +---:(mpls-mldp)
| |         +---w (root-address)?
| |           +---:(ip-address)
| |             | +---w source-address? inet:ip-address
| |             | +---w group-ip-address? IP-Multicast-
Group-Address
| |           +---:(vpn)
| |             | +---w as-number?      inet:as-number
| |           +---:(global-id)
| |             | +---w lsp-id?          string
| |         +---:(tlv-address)
| |           | +---w tlv-type?          int16
| |           | +---w tlv-len?           int16
| |           | +---w tlv-value?         binary
| |         +---:(system-info)
| |           | +---w system-id?          inet:uri
| +---w session-type-enum? enumeration
| +---w source-interface? if:interface-ref
| +---w outbound-interface? if:interface-ref
| +---w vrf? coam:routing-instance-ref

```

| +---w max-ttl? uint8

Kumar, et al.

Expires May 4, 2017

[Page 12]

```

+--ro output
  +--ro response-list* [response-index]
    | +--ro response-index      uint32
    | +--ro status-code?       int32
    | +--ro status-sub-code?    uint8
  +--ro src-test-point
    | +--ro vrf?                routing-instance-ref
    | +--ro tp-address-type-value? identityref
    | +--ro (tp-address)?
    |   +--:(mac-address)
    |     | +--ro mac-address?      yang:mac-address
    |   +--:(ipv4-address)
    |     | +--ro ipv4-address?      inet:ipv4-address
    |   +--:(ipv6-address)
    |     | +--ro ipv6-address?      inet:ipv6-address
    |   +--:(src-dst-address)
    |     | +--ro src-ip-address?     inet:ip-address
    |     | +--ro dst-ip-address?     inet:ip-address
    |     | +--ro Interface?          if:interface-ref
    |   +--:(fec)
    |     | +--ro fec-type?           fec-type
    |     | +--ro (fec-value)?
    |     |   +--:(ip-prefix)
    |     |     | +--ro ip-prefix?      inet:ip-prefix
    |     |   +--:(bgp)
    |     |     | +--ro bgp?            inet:ip-prefix
    |     |   +--:(tunnel)
    |     |     | +--ro tunnel-interface? uint32
    |     |   +--:(pw)
    |     |     | +--ro remote-pe-address? inet:ip-address
    |     |     | +--ro pw-id?          uint32
    |     |   +--:(vpls)
    |     |     | +--ro route-distinguisher? uint32
    |     |     | +--ro sender-ve-id?      uint32
    |     |     | +--ro receiver-ve-id?    uint32
    |     |   +--:(mpls-mldp)
    |     |     | +--ro (root-address)?
    |     |     |   +--:(ip-address)
    |     |     |     | +--ro source-address?      inet:ip-address
    |     |     |     | +--ro group-ip-address?    IP-Multicast-
    |     |     |
    |     |   +--:(vpn)
    |     |     | +--ro as-number?            inet:as-number
    |     |   +--:(global-id)
    |     |     | +--ro lsp-id?              string
    |   +--:(tlv-address)
    |     | +--ro tlv-type?                  int16
    |     | +--ro tlv-len?                  int16

```

Group-Address

| | +--ro tlv-value? binary

```

|      +---:(system-info)
|      |      +---ro system-id?                  inet:uri
+---ro dest-test-point
|  +---ro vrf?                      routing-instance-ref
|  +---ro tp-address-type-value?    identityref
|  +---ro (tp-address)?
|      +---:(mac-address)
|      |  +---ro mac-address?          yang:mac-address
|      +---:(ipv4-address)
|      |  +---ro ipv4-address?        inet:ipv4-address
|      +---:(ipv6-address)
|      |  +---ro ipv6-address?        inet:ipv6-address
|      +---:(src-dst-address)
|      |  +---ro src-ip-address?      inet:ip-address
|      |  +---ro dst-ip-address?      inet:ip-address
|      |  +---ro Interface?          if:interface-ref
|      +---:(fec)
|      |  +---ro fec-type?            fec-type
|      |  +---ro (fec-value)?
|      |      +---:(ip-prefix)
|      |      |  +---ro ip-prefix?      inet:ip-prefix
|      |      +---:(bgp)
|      |      |  +---ro bgp?            inet:ip-prefix
|      |      +---:(tunnel)
|      |      |  +---ro tunnel-interface? uint32
|      |      +---:(pw)
|      |      |  +---ro remote-pe-address? inet:ip-address
|      |      |  +---ro pw-id?          uint32
|      |      +---:(vpls)
|      |      |  +---ro route-distinguisher? uint32
|      |      |  +---ro sender-ve-id?      uint32
|      |      |  +---ro receiver-ve-id?    uint32
|      |      +---:(mpls-mldp)
|      |      |  +---ro (root-address)?
|      |      |      +---:(ip-address)
|      |      |      |  +---ro source-address?      inet:ip-address
|      |      |      |  +---ro group-ip-address?    IP-Multicast-
Group-Address
|      |      |      +---:(vpn)
|      |      |      |  +---ro as-number?          inet:as-number
|      |      |      +---:(global-id)
|      |      |      |  +---ro lsp-id?              string
|      +---:(tlv-address)
|      |  +---ro tlv-type?          int16
|      |  +---ro tlv-len?           int16
|      |  +---ro tlv-value?         binary
|      +---:(system-info)
|      |  +---ro system-id?         inet:uri

```

+--ro sequence-number?

uint64

Kumar, et al.

Expires May 4, 2017

[Page 14]

				+--ro bgp?	inet:ip-prefix
				+--:(tunnel)	
				+--ro tunnel-interface?	uint32
				+--:(pw)	
				+--ro remote-pe-address?	inet:ip-address
				+--ro pw-id?	uint32
				+--:(vpls)	
				+--ro route-distinguisher?	uint32
				+--ro sender-ve-id?	uint32
				+--ro receiver-ve-id?	uint32
				+--:(mpls-mldp)	
				+--ro (root-address)?	
				+--:(ip-address)	
				+--ro source-address?	inet:ip-address
Group-Address				+--ro group-ip-address?	IP-Multicast-
				+--:(vpn)	
				+--ro as-number?	inet:as-number
				+--:(global-id)	
				+--ro lsp-id?	string
				+--:(tlv-address)	
				+--ro tlv-type?	int16
				+--ro tlv-len?	int16
				+--ro tlv-value?	binary
				+--:(system-info)	
				+--ro system-id?	inet:uri
				+--ro timestamp-val?	yang:date-and-time
				+--ro ingress-intf-name?	if:interface-ref
				+--ro egress-intf-name?	if:interface-ref
				+--ro app-meta-data?	uint32

data hierarchy of OAM Retrieval Methods

4. OAM Retrieval Methods YANG Module

<CODE BEGINS> file "ietf-connectionless-oam-methods.yang"

```

module ietf-connectionless-oam-methods {
  namespace "urn:ietf:params:xml:ns:yang:ietf-connectionless-oam-methods";
  prefix coam-methods;

  import ietf-interfaces {
    prefix if;
  }
  import ietf-connectionless-oam {
    prefix coam;
  }
}

```

organization "IETF LIME Working Group";

Kumar, et al.

Expires May 4, 2017

[Page 16]

contact

"Deepak Kumar dekkumar@cisco.com
Qin Wu bill.wu@huawei.com
S Raghavan srihari@cisco.com
Zitao Wang wangzitao@huawei.com
R Rahman rrahman@cisco.com";

description

"This YANG module defines the RPCs for ,
connectionless OAM to be used within IETF
in a protocol Independent manner.
Functional level abstraction is independent with
YANG modeling. It is assumed that each protocol maps
corresponding abstracts to its native format.
Each protocol may extend the YANG model defined
here to include protocol specific extensions";

revision 2016-06-23 {

description

"06 version";

reference "";

}

rpc continuity-check {

if-feature coam:continuity-check;

description

"Generates continuity-check as per [RFC7276](#).";

input {

container destination-tp {

uses coam:tp-address;

description

"destination test point.";

}

uses coam:session-type;

leaf source-interface {

type if:interface-ref;

description

"source interface.";

}

leaf outbound-interface {

type if:interface-ref;

description

"outbound interface.";

}

leaf count {

type uint32;

default "5";

description


```

        "Specifies the number of packets that will be sent.";
    }
    leaf vrf {
        type coam:routing-instance-ref;
        description
            "vrf instance.";
    }

    leaf ttl {
        type uint8;
        default "255";
        description
            "Time to live (TTL).";
    }
    leaf packet-size {
        type uint32 {
            range "64..10000";
        }
        default "64";
        description
            "Size of ping echo request packets, in octets";
    }
}
output {
    list error-code-list {
        key "response-index";
        leaf response-index {
            type uint32;
            description
                "response index.";
        }
        leaf status-code {
            type int32;
            description
                "error code is ";
        }
        leaf status-sub-code {
            type uint8;
            description
                "sub code.";
        }
        description
            "error code list.";
    }
}
uses coam:continuity-check-data;
}

```



```
rpc path-discovery {
  description
    "Generates path discovery as per RFC7276.";
  input {
    container destination-tp {
      uses coam:tp-address;
      description
        "destination test point.";
    }
    uses coam:session-type;
    leaf source-interface {
      type if:interface-ref;
      description
        "source interface.";
    }
    leaf outbound-interface {
      type if:interface-ref;
      description
        "outbound interface.";
    }
    leaf vrf {
      type coam:routing-instance-ref;
      description
        "vrf";
    }
    leaf max-ttl {
      type uint8;
      default "255";
      description
        "max ttl.";
    }
  }
  output {
    list response-list {
      key "response-index";
      description
        "path discovery response list.";
      leaf response-index {
        type uint32;
        description
          "response index.";
      }
      leaf status-code {
        type int32;
        description
          "error code is ";
      }
    }
  }
}
```



```
    leaf status-sub-code {  
        type uint8;  
  
        description  
            "sub code is ";  
    }  
}  
  
uses coam:path-discovery-data;  
  
}  
}
```

YANG module of OAM

<CODE ENDS>

5. Security Considerations

TBD.

6. IANA Considerations

This document registers a URI in the IETF XML registry [[RFC3688](#)] [[RFC3688](#)]. Following the format in [RFC 3688](#), the following registration is requested to be made:

URI: urn:ietf:params:xml:ns:yang:ietf-connectionless-oam

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-connectionless-oam namespace: urn:ietf:params:xml:ns:yang:ietf-connectionless-oam
prefix: goam reference: RFC XXXX

7. Normative References

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

- [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>>.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", [RFC 6242](#), DOI 10.17487/RFC6242, June 2011, <<http://www.rfc-editor.org/info/rfc6242>>.
- [RFC792] Postel, J., "Internet Control Message Protocol", [RFC 792](#), September 1981.

Authors' Addresses

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

Email: dekumar@cisco.com

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

Email: wangzitao@huawei.com

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

Email: bill.wu@huawei.com

Reshad Rahman
CISCO Systems
2000 Innovation Drive
KANATA, ONTARIO K2K 3E8
CANADA

Email: rrahman@cisco.com

Srihari Raghavan
CISCO Systems
TRIL INFOPARK SEZ, Ramanujan IT City
NEVILLE BLOCK, 2nd floor, Old Mahabalipuram Road
CHENNAI, TAMIL NADU 600113
INDIA

Email: srihari@cisco.com

