

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 23, 2015

Y. Liu
F. Guo
Huawei
March 22, 2015

Yang Model for Internet Group Management Protocol (IGMP) and Multicast
Listener Discovery (MLD)
draft-liu-pim-igmp-mld-yang-01

Abstract

This document defines a YANG data model that can be used to configure and manage IGMP and MLD.

Requirements Language

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 [RFC 2119](#) [[RFC2119](#)].

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 September 23, 2015.

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

Internet-Draft

GMP YANG model

March 2015

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.	Terminology	3
3.	Design of Data Model	3
3.1.	Overview	3
3.2.	GMP Per-instance Configuration	4
3.2.1.	Per-instance Parameters	4
3.2.2.	Per-SSM-Mapping Configuration of GMP Instance	5
3.2.3.	Per-interface Configuration of GMP Instance	5
3.3.	Operational State	7
4.	Notifications	10
5.	GMP Yang Module	11
6.	IANA Considerations	53
7.	Security Considerations	53
8.	Acknowledgements	53
9.	References	53
9.1.	Normative References	53
9.2.	Informative References	54
	Authors' Addresses	54

[1.](#) Introduction

YANG[RFC6020] is a data definition language that was introduced to define the contents of a conceptual data store that allows networked devices to be managed using NETCONF[RFC6241]. YANG is proving relevant beyond its initial confines, as bindings to other interfaces(e.g. REST) and encoding other than XML (e.g. JSON) are being defined. Furthermore, YANG data models can be used as the basis of implementation for other interface, such as CLI and Programmatic APIs.

This document defines a YANG data model that can be used to configure and manage IGMP and MLD. It includes IGMPv1[RFC1112], IGMPv2[RFC2236], IGMPv3[RFC3376] and MLDv1[RFC2710], MLDv2[RFC3810]. In addition, features described in IGMP and MLD standards other than mentioned above RFC are also supported. For convenience, IGMP and MLD are wholly called GMP below.

Internet-Draft

GMP YANG model

March 2015

[2.](#) Terminology

- o IGMP: Internet Group Management Protocol
- o MLD: Multicast Listener Discovery
- o GMP: Group Management Protocol
- o SSM: Source-Specific Multicast

[3.](#) Design of Data Model

[3.1.](#) Overview

The GMP Yang module has only one main container ::

- o gmp-instances : that contains per-instance writable configuration objects.
- o gmp-group-infos : provide IGMP or MLD group state for all of the dynamic group members.
- o gmp-static-group-infos : provide IGMP or MLD static group state for all of the static group members.
- o gmp-if-infos : provide IGMP or MLD interface state and running parameters for all of the IGMP or MLD interfaces.
- o gmp-packet-statistics : provide IGMP or MLD packet statistical information for all of the IGMP or MLD protocol packets.

The figure below describes the overall structure of the GMP Yang module :

Internet-Draft

GMP YANG model

March 2015

```
module: gmp
  +--rw gmp
    +--rw gmp-instances
      |   +--rw gmp-instance* [vrf-name addr-family]
      |   +--...
      |   +--rw gmp-ssm-mappings
      |   |   +--rw gmp-ssm-mapping
      |   |   +--...
      |   +--rw gmp-interfaces
      |   |   +--rw gmp-interface* [vrf-name if-name addr-family]
      |   |   +--...
      |   |   +--rw gmp-static-groups
      |   |   |   +--rw gmp-static-group
      |   |   |   +--...
      |   +--rw gmp-group-infos
      |   |   +--ro gmp-group-info* [vrf-name addr-family if-name]
      |   |   +--...
      |   |   +--ro gmp-source-infos
      |   |   |   +--ro gmp-source-info* [ipv4-source-addr ipv6-source-addr]
      |   |   |   +--...
      |   +--rw gmp-static-group-infos
      |   |   +--ro gmp-static-group-info* [vrf-name addr-family if-name]
      |   |   +--...
      |   +--rw gmp-if-infos
      |   |   +--ro gmp-if-info* [vrf-name if-name addr-family]
      |   |   +--...
      |   +--rw gmp-packet-statistics
      |   |   +--ro gmp-packet-statistic* [vrf-name addr-family if-name]
      |   |   +--...
```

Figure 1 The overview of GMP YANG data model

[3.2.](#) GMP Per-instance Configuration

GMP per-instance configuration container includes parameters of the public GMP instance or the GMP instance binding a specific VRF. GMP per-instance configuration container is divided into:

- o Per-instance parameters
- o Per-SSM-Mapping configuration of the GMP instance
- o Per-interface configuration of the GMP instance

[3.2.1.](#) Per-instance Parameters

The per-instance parameter includes the name of the VRF bound by the GMP instance, and timer parameters such as query interval etc.

```

+--rw gmp-instances
|   +--rw gmp-instance* [vrf-name addr-family]
|       +--rw vrf-name                string
|       +--rw addr-family             enumeration
|       +--rw query-interval?         uint16
|       +--rw query-rsp-interval?     uint8
|       +--rw robustness?             uint8
|       +--rw last-member-query-interval? uint8
|       +--rw require-router-alert?   boolean
|       +--rw send-router-alert?      boolean
|       +--rw other-querier-present-time? uint16
|       +--ro gmp-entry-limit?        uint32
|       +--rw ipsec-name?             string
|       +--rw ipsec-type?             enumeration

```

Figure 2 The YANG data model of GMP instance configuration

[3.2.2.](#) Per-SSM-Mapping Configuration of GMP Instance

Per-SSM-Mapping configuration of the GMP instance includes the SSM Mapping rules.IGMPv1/v2 and MLDv1 reports can use these rules to map SG state for PIM SSM[RFC4607].IGMPv3 and MLDv2 can default use PIM

SSM , which is described in [[RFC4604](#)].

```
|      +--rw gmp-ssm-mappings
|      |      +--rw gmp-ssm-mapping
|      |      |      +--rw ipv4-ssm-maping-group      inet:ipv4-address
|      |      |      +--rw ipv6-ssm-maping-group      inet:ipv6-address
|      |      |      +--rw is-ssmap-mask              boolean
|      |      |      +--rw ipv4-ssm-maping-mask?      inet:ipv4-address
|      |      |      +--rw ipv6-ssm-maping-mask?      uint8
|      |      |      +--rw is-masklen                 boolean
|      |      |      +--rw masklen?                   uint8
|      |      |      +--rw ipv4-source-addr           inet:ipv4-address
|      |      |      +--rw ipv6-source-addr           inet:ipv6-address
```

Figure 3 The YANG data model of GMP SSM-Mapping configuration

[3.2.3.](#) Per-interface Configuration of GMP Instance

Per-interface configuration of the GMP instance includes the interface name, timer parameters, policies, static groups etc. GMP per-instance configuration container is divided into two containers:

- o Per-interface parameters
- o Per-static-group configuration of the GMP interface

[3.2.3.1.](#) Per-interface Parameters

The per-interface parameter includes the name of the interface, and the VRF name bound by the interface, and time parameters, policies etc.

```
|      +--rw gmp-interfaces
|      |      +--rw gmp-interface* [vrf-name if-name addr-family]
|      |      |      +--rw vrf-name                  string
|      |      |      +--rw if-name                    if-name
|      |      |      +--rw addr-family                enumeration
|      |      |      +--rw gmp-enable                 boolean
|      |      |      +--rw ip-source-ply?             boolean
|      |      |      +--rw ip-source-acl-name?        string
|      |      |      +--rw ip-source-acl-name-ipv6?   string
```

```

|      +--rw query-interval?                uint16
|      +--rw query-rsp-interval?            uint8
|      +--rw robustness?                    uint8
|      +--rw version?                      uint8
|      +--rw last-member-query-interval?    uint8
|      +--rw require-router-alert?         boolean
|      +--rw send-router-alert?            boolean
|      +--rw other-querier-present-time?    uint16
|      +--rw immediate-leave?              boolean
|      +--rw immediate-leave-acl-name?      string
|      +--rw immediate-leave-acl-name-ipv6? string
|      +--rw gmp-entry-limit?              uint16
|      +--rw except-acl-name?              string
|      +--rw except-acl-name-ipv6?         string
|      +--rw ssmap-enable?                 boolean
|      +--rw group-acl-name?               string
|      +--rw group-acl-name-ipv6?          string
|      +--rw group-acl-gmp-ver?            uint8
|      +--rw query-acl-name?               string
|      +--rw queryaclnameipv6?             string
|      +--rw ipsec-name?                   string
|      +--rw ipsec-type?                   enumeration

```

Figure 4 The YANG data model of GMP interface configuration

[3.2.3.2.](#) Per-static-group Configuration of GMP interface

Per-static-group configuration of the GMP interface includes the static group address, and as a option also includes source address, every static group step, and group numbers on the interface.

```

|      +--rw gmp-static-groups
|      |      +--rw gmp-static-group
|      |      |      +--rw vrf-name                string
|      |      |      +--rw addr-family            enumeration
|      |      |      +--rw if-name                if-name
|      |      |      +--rw ipv4-static-group       inet:ipv4-address
|      |      |      +--rw ipv6-static-group       inet:ipv6-address
|      |      |      +--rw is-source-addr         boolean

```

	+-rw	ipv4-source-addr?	inet:ipv4-address
	+-rw	ipv6-source-addr?	inet:ipv6-address
	+-rw	is-step-grp-mask	boolean
	+-rw	ipv4-inc-step-group-mask?	inet:ipv4-address
	+-rw	ipv6-inc-step-group-mask?	inet:ipv6-address
	+-rw	is-masklen	boolean
	+-rw	masklen?	uint8
	+-rw	total-number?	uint16

Figure 5 The YANG data model of GMP static group configuration

3.3. Operational State

"gmp" container provides operational states for IGMP or MLD. This container is divided in four components:

- o gmp-group-infos : provide IGMP or MLD group state for all of the dynamic group members.
- o gmp-static-group-infos : provide IGMP or MLD static group state for all of the static group members.
- o gmp-if-infos : provide IGMP or MLD interface state and running parameters for all of the IGMP or MLD interfaces.
- o gmp-packet-statistics : provide IGMP or MLD packet statistical information for all of the IGMP or MLD protocol packets.

+-rw gmp-group-infos


```

|  +---ro gmp-group-info* [vrf-name addr-family if-name]
|      +---ro vrf-name          string
|      +---ro addr-family       enumeration
|      +---ro if-name           if-name
|      +---ro ipv4-group-addr?   inet:ipv4-address
|      +---ro ipv6-group-addr?   inet:ipv6-address
|      +---ro ipv4-last-report-addr? inet:ipv4-address
|      +---ro ipv6-last-report-addr? inet:ipv6-address
|      +---ro up-time?           uint32
|      +---ro expire-time?       uint32
|      +---ro last-member-query-count? uint8
|      +---ro last-member-query-time? uint8
|      +---ro source-filter-mode? enumeration
|      +---ro v1-host-timer?     uint32
|      +---ro v2-host-timer?     uint32
|      +---ro is-ssm-mapping?    boolean
|      +---ro gmp-source-infos
|          +---ro gmp-source-info* [ipv4-source-addr ipv6-source-addr]
|              +---ro ipv4-source-addr      inet:ipv4-address
|              +---ro ipv6-source-addr      inet:ipv6-address
|              +---ro source-up-time?        uint32
|              +---ro source-expire-time?     uint32
|              +---ro source-last-member-query-count? uint8
|              +---ro source-last-member-query-time? uint8

```

The Figure 6 The YANG data model of GMP dynamic group and source information

```

+---rw gmp-static-group-infos
|  +---ro gmp-static-group-info* [vrf-name addr-family if-name]
|      +---ro vrf-name          string
|      +---ro addr-family       enumeration
|      +---ro if-name           if-name
|      +---ro ipv4-group-addr?   inet:ipv4-address
|      +---ro ipv6-group-addr?   inet:ipv6-address
|      +---ro ipv4-source-addr?   inet:ipv4-address
|      +---ro ipv6-source-addr?   inet:ipv6-address
|      +---ro multicast-boundary? boolean

```

The Figure 7 The YANG data model of GMP static group and source information

```
+--rw gmp-if-infos
|   +--ro gmp-if-info* [vrf-name if-name addr-family]
|       +--ro vrf-name          string
|       +--ro if-name          if-name
|       +--ro addr-family       enumeration
|       +--ro ipv4-if-addr?     inet:ipv4-address
|       +--ro ipv6-if-addr?     inet:ipv6-address
|       +--ro gmp-if-state?     enumeration
|       +--ro negotiate-query-time? uint16
|       +--ro configure-query-time? uint16
|       +--ro other-querier-timeout? uint16
|       +--ro max-query-response-time? uint8
|       +--ro last-member-query-time? uint8
|       +--ro startup-query-interval? uint16
|       +--ro startup-query-count? uint8
|       +--ro general-query-time? uint16
|       +--ro ipv4-querier-addr? inet:ipv4-address
|       +--ro ipv6-querier-addr? inet:ipv6-address
|       +--ro querier-local?    enumeration
|       +--ro gmp-joins?        uint32
|       +--ro gmp-leaves?       uint32
|       +--ro negotiate-robust? uint8
|       +--ro configure-robust? uint8
|       +--ro startup-query-timer-state? enumeration
|       +--ro other-querier-timer-state? enumeration
|       +--ro ipsec-name?       string
|       +--ro ipsec-type?       enumeration
```

The Figure 8 The YANG data model of GMP interface information

```
+--rw gmp-packet-statistics
|   +--ro gmp-packet-statistic* [vrf-name addr-family if-name]
|       +--ro vrf-name          string
|       +--ro if-name          if-name
|       +--ro addr-family       enumeration
|       +--ro ipv4-if-addr?     inet:ipv4-address
|       +--ro ipv6-if-addr?     inet:ipv6-address
|       +--ro query-recv-valid-num? uint32
|       +--ro query-recv-invalid-num? uint32
|       +--ro query-recv-ignore-num? uint32
|       +--ro report-v1v2-asm-recv-valid-num? uint32
|       +--ro report-v1v2-asm-recv-invalid-num? uint32
|       +--ro report-v1v2-asm-recv-ignore-num? uint32
|       +--ro report-v1v2-ssm-recv-valid-num? uint32
|       +--ro report-v1v2-ssm-recv-invalid-num? uint32
|       +--ro report-v1v2-ssm-recv-ignore-num? uint32
```

+++ro leave-asm-recv-valid-num?	uint32
+++ro leave-asm-recv-invalid-num?	uint32

+++ro leave-asm-recv-ignore-num?	uint32
+++ro leave-ssm-recv-valid-num?	uint32
+++ro leave-ssm-recv-invalid-num?	uint32
+++ro leave-ssm-recv-ignore-num?	uint32
+++ro report-v3-isin-recv-valid-num?	uint32
+++ro report-v3-isin-recv-invalid-num?	uint32
+++ro report-v3-isin-recv-ignore-num?	uint32
+++ro report-v3-isex-recv-valid-num?	uint32
+++ro report-v3-isex-recv-invalid-num?	uint32
+++ro report-v3-isex-recv-ignore-num?	uint32
+++ro report-v3-toin-recv-valid-num?	uint32
+++ro report-v3-toin-recv-invalid-num?	uint32
+++ro report-v3-toin-recv-ignore-num?	uint32
+++ro report-v3-toex-recv-valid-num?	uint32
+++ro report-v3-toex-recv-invalid-num?	uint32
+++ro report-v3-toex-recv-ignore-num?	uint32
+++ro report-v3-allow-recv-valid-num?	uint32
+++ro report-v3-allow-recv-invalid-num?	uint32
+++ro report-v3-allow-recv-ignore-num?	uint32
+++ro report-v3-block-recv-valid-num?	uint32
+++ro report-v3-block-recv-invalid-num?	uint32
+++ro report-v3-block-recv-ignore-num?	uint32
+++ro source-recv-valid-num?	uint32
+++ro source-recv-invalid-num?	uint32
+++ro source-recv-ignore-num?	uint32
+++ro query-send-num?	uint32
+++ro report-v1v2-asm-send-num?	uint32
+++ro report-v1v2-ssm-send-num?	uint32
+++ro leave-asm-send-num?	uint32
+++ro leave-ssm-send-num?	uint32
+++ro query-g-send-num?	uint32
+++ro query-sg-send-num?	uint32
+++ro report-v3-isin-send-num?	uint32
+++ro report-v3-isex-send-num?	uint32
+++ro report-v3-toin-send-num?	uint32
+++ro report-v3-toex-send-num?	uint32
+++ro report-v3-allow-send-num?	uint32
+++ro report-v3-block-send-num?	uint32
+++ro source-send-num?	uint32

4. Notifications

The gmp yang data model introduces some notifications:

gmp-group-join : This notification is sent when the system receive a IGMP or MLD group membership report.

gmp-group-leave : This notification is sent when the system receive a IGMP or MLD group membership leave.

notifications:

```
+---n gmp-group-join
|  +--ro vrf-name?          string
|  +--ro if-name?           if-name
|  +--ro addr-family?       enumeration
|  +--ro ipv4-group-addr?    inet:ipv4-address
|  +--ro ipv6-group-addr?    inet:ipv6-address
|  +--ro ipv4-source-addr?   inet:ipv4-address
|  +--ro ipv6-source-addr?   inet:ipv6-address
|  +--ro ipv4-sender-addr?   inet:ipv4-address
|  +--ro ipv6-sender-addr?   inet:ipv6-address
+---n gmp-group-leave
|  +--ro vrf-name?          string
|  +--ro if-name?           if-name
|  +--ro addr-family?       enumeration
|  +--ro ipv4-group-addr?    inet:ipv4-address
|  +--ro ipv6-group-addr?    inet:ipv6-address
|  +--ro ipv4-source-addr?   inet:ipv4-address
|  +--ro ipv6-source-addr?   inet:ipv6-address
|  +--ro ipv4-sender-addr?   inet:ipv4-address
|  +--ro ipv6-sender-addr?   inet:ipv6-address
```

The Figure 10 The YANG data model of GMP notifications

5. GMP Yang Module

```
module gmp {
  namespace "urn:huawei:params:xml:ns:yang:gmp";
  // replace with IANA namespace when assigned -
```

```
// urn:ietf:params:xml:ns:yang:1
prefix "gmp";
import ietf-inet-types {
    prefix inet;
}
organization
    "IETF PIM(Protocol Independent Multicast) Working Group";
contact
    "liuyisong@huawei.com
    guofeng@huawei.com ";
description
    "This YANG module defines the generic configuration data for GMP,
    i.e. IGMP and MLD, which is common across all of the vendor
    implementations of the protocol. It is intended that the module
    will be extended by vendors to define vendor-specific
    GMP configuration parameters.";
```

Liu & Guo

Expires September 23, 2015

[Page 11]

Internet-Draft

GMP YANG model

March 2015

```
revision 2015-03-09 {
    description
        "Add operation and notification contents.";
}
revision 2014-10-21 {
    description
        "Initial revision.";
}

typedef if-name {
    description "if-name is like ethernet1/1/1/1";
    type string {
        length "1..63";
    }
}

container gmp {

    container gmp-instances {

        list gmp-instance {

            key "vrf-name addr-family";
            max-elements "unbounded";
```

```

min-elements "0";
description "Specifies a list of gmp instances.";

leaf vrf-name {
    description
        "Name of an gmp instance.If the name string is
        empty the instance means a public instance
        whose name is _public_.";
    config "true";
    //default "_public_";
    type string {
        length "0..32";
    }
}
leaf addr-family {
    description
        "Specify an address family, which determines
        whether an address is an ipv4 or ipv6 address.";
    config "true";
    mandatory "true";
    type enumeration {
        enum ipv4unicast {
            value "0";

```

```

        description
            "Specify an address family, which
            determines whether an address is an
            ipv4 or ipv6 address.";
    }
    enum ipv6unicast {
        value "1";
        description
            "Specify an address family, which
            determines whether an address is an
            ipv4 or ipv6 address.";
    }
}
leaf query-interval {
    description
        "Specify the interval at which the router sends
        general query messages. The value is an integer

```

```

        ranging from 1 to 18000, in seconds. The default
        value for ipv4 is 60, and 125 for ipv6.";
    config "true";
    default "60";
    type uint16 {
        range "1..18000";
    }
}
leaf query-rsp-interval {
    description
        "Specify the maximum response time for a query
        message. The value is an integer ranging from
        1 to 25, in seconds. The default value is 10.";
    config "true";
    default "10";
    type uint8 {
        range "1..25";
    }
}
leaf robustness {
    description
        "Specify the number of times for retransmitting
        a message to avoid the packet loss. The value is
        an integer ranging from 2 to 5. The default
        value is 2.";
    config "true";
    default "2";
    type uint8 {
        range "2..5";
    }
}

```

```

}
leaf last-member-query-interval {
    description
        "Specify the interval at which the querier sends
        last-member query messages. The value is an
        integer ranging from 1 to 5, in seconds. The
        default value is 1. This parameter makes sense
        only when the current querier runs igmpv2,
        igmpv3 or mld.";
    config "true";
    default "1";
}

```

```

        type uint8 {
            range "1..5";
        }
    }
    leaf require-router-alert {
        description
            "Configure the router to process only the
            messages whose ip headers contain router-alert
            options globally. By default, the router does
            not check whether the received igmp messages
            contain router-alert options.";
        config "true";
        type boolean;
    }
    leaf send-router-alert {
        description
            "Configure the router to send the messages with
            router-alert options in the ip headers globally.
            By default, the ip headers contain router-alert
            options. ";
        config "true";
        type boolean;
    }
    leaf other-querier-present-time {
        description
            "Set the global keepalive period for other
            queriers. ";
        config "true";
        type uint16 {
            range "60..300";
        }
    }
    leaf gmp-entry-limit {
        description
            "Set the maximum number of entries that can be
            created for the current instance. The value is
            an integer ranging from 1 to 65536. The default

```

```

        value is 49152.";
        config "false";
        type uint32 {
            range "1..65536";

```



```

    }
}
leaf ipsec-name {
    description
        "Sa name. The value is a string of 1 to 15
        characters.";
    config "true";
    type string {
        length "1..15";
        pattern "^[^ ]+$";
    }
}
leaf ipsec-type {
    description
        "Ipsec type used as a query option.";
    config "true";
    type enumeration {
        enum forall {
            value "0";
            description
                "Ipsec type used as a query option.";
        }
        enum forquery {
            value "1";
            description
                "Ipsec type used as a query option.";
        }
    }
}

container gmp-ssm-mappings {

    container gmp-ssm-mapping {

        leaf ipv4-ssm-maping-group {
            description
                "Specify the address of a multicast
                group.";
            config "true";
            mandatory "true";
            type inet:ipv4-address;
        }
        leaf ipv6-ssm-maping-group {
            description

```

```
        "Specify the address of a multicast
        group.";
        config "true";
        mandatory "true";
        type inet:ipv6-address;
    }
    leaf is-ssmap-mask {
        description
            "Whether a mask is configured for a
            multicast group address.";
        config "true";
        mandatory "true";
        type boolean;
    }
    leaf ipv4-ssm-maping-mask {
        description
            "Specify the mask of a multicast group
            address.";
        config "true";
        type inet:ipv4-address;
    }
    leaf ipv6-ssm-maping-mask {
        description
            "Specify the mask of a multicast group
            address.";
        config "true";
        type uint8 {
            range "0..128";
        }
    }
    leaf is-masklen {
        description
            "Whether the length is set for the mask
            of a multicast group address.";
        config "true";
        mandatory "true";
        type boolean;
    }
    leaf masklen {
        description
            "Specify the mask length of a multicast
            group address. In the case of an ipv4
            address, the mask length ranges from 4
            to 32. In the case of an ipv6 address,
            the mask length is 16/32/64/128.";
        config "true";
        type uint8 {
```

range "4..128";

Internet-Draft

GMP YANG model

March 2015

```
    }
  }
  leaf ipv4-source-addr {
    description
      "Specify the address of a multicast
       source.";
    config "true";
    mandatory "true";
    type inet:ipv4-address;
  }
  leaf ipv6-source-addr {
    description
      "Specify the address of a multicast
       source.";
    config "true";
    mandatory "true";
    type inet:ipv6-address;
  }
}

container gmp-interfaces {

  list gmp-interface {

    key "vrf-name if-name addr-family";
    max-elements "unbounded";
    min-elements "0";
    description "Specifies an gmp interface.";

    leaf vrf-name {
      description
        "Name of an gmp instance. If the name
         string is empty the instance means a
         public instance whose name is _public_.";
      config "true";
      mandatory "true";
      //default "_public_";
      type string {
        length "0..32";
      }
    }
  }
}
```

```

    }
}
leaf if-name {
    description "Interface name.";
    config "true";
    type if-name;
}
leaf addr-family {

```

```

    description
        "Specify an address family, which
        determines whether an address is an
        ipv4 or ipv6 address.";
    config "true";
    mandatory "true";
    type enumeration {
        enum ipv4unicast {
            value "0";
            description
                "Specify an address family, which
                determines whether an address is
                an ipv4 or ipv6 address.";
        }
        enum ipv6unicast {
            value "1";
            description
                "Specify an address family, which
                determines whether an address is
                an ipv4 or ipv6 address.";
        }
    }
}
leaf gmp-enable {
    description
        "Enable protocols on an interface.";
    config "true";
    mandatory "true";
    type boolean;
}
leaf ip-source-ply {
    description
        "Configure a policy for filtering igmp

```

```

        report messages based on host addresses.
        By default, no policy is configured for
        filtering igmp report messages based on
        host addresses.";
    config "true";
    type boolean;
}
leaf ip-source-acl-name {
    description
        "Configure an acl that defines a host
        addresses range. The value is an integer
        ranging from 2000 to 2999, or a
        case-sensitive string with a maximum of
        32 characters. By default, the acl is
        not configured.";

```

```

        config "true";
        type string {
            length "1..32";
            pattern "^[^ ]+$";
        }
    }
    leaf ip-source-acl-name-ipv6 {
        description
            "Configure an acl that defines a host
            addresses range. The value is an integer
            ranging from 2000 to 2999, or a
            case-sensitive string with a maximum of
            32 characters. By default, the acl is
            not configured.";
        config "true";
        type string {
            length "1..32";
            pattern "^[^ ]+$";
        }
    }
}
leaf query-interval {
    description
        "Specify the interval at which the router
        sends general query messages. The value
        is an integer ranging from 1 to 18000,
        in seconds. The default value for ipv4

```

```

        is 60, and 125 for ipv6.";
    config "true";
    default "60";
    type uint16 {
        range "1..18000";
    }
}
leaf query-rsp-interval {
    description
        "Specify the maximum response time for a
        query message. The value is an integer
        ranging from 1 to 25, in seconds. The
        default value is 10. This time is used
        to control the deadline of mainframe
        feed back the relation-ship of group
        members.";
    config "true";
    default "10";
    type uint8 {
        range "1..25";
    }
}

```

```

leaf robustness {
    description
        "Specify the number of times for
        retransmitting messages to avoid packet
        loss. The value is an integer ranging
        from 2 to 5. The default value is 2.";
    config "true";
    default "2";
    type uint8 {
        range "2..5";
    }
}
leaf version {
    description
        "Specify the version of igmp or mld
        running on an interface. By default,
        igmpv2 or mldv2 is used.";
    config "true";
    default "2";
}

```

```

        type uint8 {
            range "1..3";
        }
    }
    leaf last-member-query-interval {
        description
            "Specify the interval at which the
            querier sends last-member query messages.
            The value is an integer ranging from 1
            to 5, in seconds. The default value is 1.
            This parameter makes sense only when the
            current querier runs igmpv2, igmpv3 or
            mld.";
        config "true";
        default "1";
        type uint8 {
            range "1..5";
        }
    }
    leaf require-router-alert {
        description
            "Configure an interface to process only
            messages whose ip headers contain
            router-alert options. By default, the
            interface does not check whether the
            received messages contain router-alert
            options.";
        config "true";
        type boolean;
    }

```

```

    }
    leaf send-router-alert {
        description
            "Configure an interface to send the
            messages with router-alert options in
            the ip headers. By default, the ip
            header contain router-alert options.";
        config "true";
        default "true";
        type boolean;
    }
    leaf other-querier-present-time {

```

```

        description
            "Set the keepalive period for other
            queriers on an interface. The value
            ranges from 60 to 300, in second.
            By default, no keepalive period is set
            for other queriers.";
        config "true";
        type uint16 {
            range "60..300";
        }
    }
    leaf immediate-leave {
        description
            "Configure an interface that receives a
            leave message of a certain group to
            immediately delete the corresponding
            group records, without sending a
            last-member query message.";
        config "true";
        type boolean;
    }
    leaf immediate-leave-acl-name {
        description
            "Configure an acl that defines a
            multicast group range. the basic acl
            number ranges from 2000 to 2999, and
            the advanced acl number ranges from 3000
            to 3999. The name is a string with a
            maximum of 32 case-sensitive characters.
            By default, the acl is not configured";
        config "true";
        type string {
            length "1..32";
            pattern "^[^ ]+$";
        }
    }
}

```

```

leaf immediate-leave-acl-name-ipv6 {
    description
        "Configure an acl that defines a
        multicast group range. The basic acl
        number ranges from 2000 to 2999, and

```



```

        the advanced acl number ranges from 3000
        to 3999. The name is a string with a
        maximum of 32 case-sensitive characters.
        By default, the acl is not configured";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf gmp-entry-limit {
    description
        "Specify the maximum number of entries
        that the current interface can create.
        It is an integer ranging from 1 to
        16384. The default value is 16384.";
    config "true";
    type uint16 {
        range "1..16384";
    }
}
leaf except-acl-name {
    description
        "Specify the range of multicast groups,
        the number of igmp entries corresponding
        to which needs not be limited. The basic
        acl number ranges from 2000 to 2999. The
        basic acl filters group addresses only,
        without distinguishing (*, g) entries
        and (s, g) entries.";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf except-acl-name-ipv6 {
    description
        "Specify the range of multicast groups,
        the number of igmp entries corresponding
        to which needs not be limited. The basic
        acl number ranges from 2000 to 2999. The
        basic acl filters group addresses only,

```

```

        without distinguishing (*, g) entries
        and (s, g) entries";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf ssmmap-enable {
    description
        "Enable ssm mapping on an interface.";
    config "true";
    type boolean;
}
leaf group-acl-name {
    description
        "Set a filter for multicast groups on an
        interface to control the range of
        multicast groups that hosts can join.
        The basic acl number ranges from 2000
        to 2999, and the advanced acl number
        ranges from 3000 to 3999. The name is a
        string with a maximum of 32 ";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf group-acl-name-ipv6 {
    description
        "Set a filter for multicast groups on an
        interface to control the range of
        multicast groups that hosts can join.
        The basic acl number ranges from 2000
        to 2999, and the advanced acl number
        ranges from 3000 to 3999. The name is a
        string with a maximum of 32 ";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf group-acl-gmp-ver {
    description
        "Forbids hosts that use a specified
        version to join the multicast group.

```

Internet-Draft

GMP YANG model

March 2015

```

        The version can be specified in this
        command only after a multicast filtering
        policy is configured.";
    config "true";
    type uint8 {
        range "1..3";
    }
}
leaf query-acl-name {
    description
        "Configure an acl that defines a host
        addresses range. The value is an integer
        ranging from 2000 to 2999, or a
        case-sensitive string with a maximum of
        32 characters. By default, the acl is
        not configured.";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf queryaclnameipv6 {
    description
        "Configure an acl that defines a host
        addresses range. The value is an integer
        ranging from 2000 to 2999, or a
        case-sensitive string with a maximum of
        32 characters. By default, the acl is
        not configured.";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf ipsec-name {
    description
        "Sa name. The value is a string of 1 to
        15 characters.";
    config "true";
    type string {
```

```

        length "1..15";
        pattern "^[^ ]+$";
    }
}
leaf ipsec-type {
    description

```

```

        "Ipssec type used as a query option.";
    config "true";
    type enumeration {
        enum forall {
            value "0";
            description
                "Ipssec type used as a query option.";
        }
        enum forquery {
            value "1";
            description
                "Ipssec type used as a query option.";
        }
    }
}

container gmp-static-groups {

    container gmp-static-group {

        leaf vrf-name {
            description
                "Name of an gmp instance. If the
                 name string is empty the instance
                 means a public instance whose
                 name is _public_.";
            config "true";
            mandatory "true";
            //default "_public_";
            type string {
                length "0..32";
            }
        }
        leaf addr-family {
            description

```

```

        "Specify an address family, which
        determines whether an ipv4 or
        ipv6 address is used.";
    config "true";
    mandatory "true";
    type enumeration {
        enum ipv4unicast {
            value "0";
            description
                "Specify an address family,
                which determines whether
                an ipv4 or ipv6 address
                is used.";

```

```

    }
    enum ipv6unicast {
        value "1";
        description
            "Specify an address family,
            which determines whether
            an ipv4 or ipv6 address
            is used.";
    }
}
leaf if-name {
    description "Interface name.";
    config "true";
    mandatory "true";
    type if-name;
}
leaf ipv4-static-group {
    description
        "Specifies the address of a
        static group.";
    config "true";
    mandatory "true";
    type inet:ipv4-address;
}
leaf ipv6-static-group {
    description
        "Specifies the address of a

```

```

        static group.";
        config "true";
        mandatory "true";
        type inet:ipv6-address;
    }
    leaf is-source-addr {
        description
            "Whether an address is configured
            for the multicast source.";
        config "true";
        mandatory "true";
        type boolean;
    }
    leaf ipv4-source-addr {
        description
            "Specifies the ipv4 address of a
            remote neighbor.";
        config "true";
        type inet:ipv4-address;
    }
}

```

```

    leaf ipv6-source-addr {
        description
            "Specifies the ipv6 address of a
            remote neighbor.";
        config "true";
        type inet:ipv6-address;
    }
    leaf is-step-grp-mask {
        description
            "Determine whether to specify the
            step mask in batch configuration
            mode.";
        config "true";
        mandatory "true";
        type boolean;
    }
    leaf ipv4-inc-step-group-mask {
        description
            "Specify the step mask of a group
            address in batch configuration
            mode.";
    }
}

```

```

        config "true";
        type inet:ipv4-address;
    }
    leaf ipv6-inc-step-group-mask {
        description
            "Specify the step mask of a group
            address in batch configuration
            mode.";
        config "true";
        type inet:ipv6-address;
    }
    leaf is-masklen {
        description
            "Determine whether to set the
            length for the step mask of a
            multicast group address in batch
            configuration mode. ";
        config "true";
        mandatory "true";
        type boolean;
    }
    leaf masklen {
        description
            "Specify the mask length of a
            multicast group address. In the
            case of an ipv4 address, the
            mask length ranges from 4 to 32.

```

```

        In the case of an ipv6 address,
        the mask length is 16/32/64/128.";
        config "true";
        type uint8 {
            range "4..128";
        }
    }
    leaf total-number {
        description
            "Specify the number of multicast
            group addresses in batch
            configuration mode. It is an
            integer ranging from 2 to 512.";
        config "true";

```

```

                                default "2";
                                type uint16 {
                                    range "2..512";
                                }
                            }
                        }
                    }
                }
            }
        }
    }

    container gmp-group-infos {

        list gmp-group-info {

            key "vrf-name addr-family if-name";
            config "false";

            leaf vrf-name {
                description "Specify the name of a vpn instance.";
                config "false";
                //default "_public_";
                type string {
                    length "0..32";
                }
            }
            leaf addr-family {
                description
                    "Specify an address family, which determines
                    whether an address is an ipv4 or ipv6 address.";
            }
        }
    }

```

```

        config "false";
        type enumeration {
            enum ipv4unicast {
                value "0";
                description "ipv4unicast:";
            }
            enum ipv6unicast {

```



```

        value "1";
        description "ipv6unicast:";
    }
}
leaf if-name {
    description "Specify the name of an interface.";
    config "false";
    type if-name;
}
leaf ipv4-group-addr {
    description
        "Multicast group address. In ipv4 address, the
        value ranges from 224.0.0.0 to 239.255.255.255. ";
    config "false";
    type inet:ipv4-address;
}
leaf ipv6-group-addr {
    description
        "Multicast group address. In ipv6, the value is
        in hexadecimal notation and in the format of
        ffx:xxxx:xxxx::xxxx ";
    config "false";
    type inet:ipv6-address;
}
leaf ipv4-last-report-addr {
    description
        "Specify the host that sends the last report
        message.";
    config "false";
    type inet:ipv4-address;
}
leaf ipv6-last-report-addr {
    description
        "Specify the host that sends the last report
        message.";
    config "false";
    type inet:ipv6-address;
}
leaf up-time {
    description

```

```

        "Specify the time when a multicast group is
        discovered, in seconds.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf expire-time {
    description
        "Specify the scheduled time to remove a group
        from the group table. The value is expressed in
        seconds.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf last-member-query-count {
    description
        "Specify the number of times that the last member
        query message is transmitted.";
    config "false";
    type uint8 {
        range "1..5";
    }
}
leaf last-member-query-time {
    description
        "Specify the timeout period for sending last member
        query messages.";
    config "false";
    type uint8 {
        range "1..25";
    }
}
leaf source-filter-mode {
    description
        "Specify the group mode, exclude or include.";
    config "false";
    type enumeration {
        enum include {
            value "0";
            description "include:include";
        }
        enum exclude {
            value "1";
            description "exclude:exclude";
        }
    }
}

```

Internet-Draft

GMP YANG model

March 2015

```
        enum na {
            value "2";
            description "na:na";
        }
    }
}
leaf v1-host-timer {
    description
        "Specify the timeout period of an v1 host.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf v2-host-timer {
    description
        "Specify the timeout period of an v2 host.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf is-ssm-mapping {
    description
        "Identifies an ssm mapping-enabled group.";
    config "false";
    type boolean;
}
container gmp-source-infos {

    list gmp-source-info {

        description
            "Specify dynamic multicast source information.";
        key "ipv4-source-addr ipv6-source-addr";
        config "false";

        leaf ipv4-source-addr {
            description
                "Specify the address of a multicast source.";
            config "false";
            type inet:ipv4-address;
        }
    }
}
```

```

leaf ipv6-source-addr {
  description
    "Specify the address of a multicast source.";
  config "false";
  type inet:ipv6-address;
}

```

```

}
leaf source-up-time {
  description
    "Specify the time when a multicast source
    is discovered.";
  config "false";
  type uint32 {
    range "0..4294967295";
  }
}
leaf source-expire-time {
  description
    "Specify the time left to delete records
    from the source table.";
  config "false";
  type uint32 {
    range "0..4294967295";
  }
}
leaf source-last-member-query-count {
  description
    "Specify the number of times that the
    last member query message is transmitted.";
  config "false";
  type uint8 {
    range "1..5";
  }
}
leaf source-last-member-query-time {
  description
    "Specify the timeout period for sending
    last member query messages.";
  config "false";
  type uint8 {
    range "1..25";
  }
}

```

```

        }
    }
}

}

container gmp-static-group-infos {

```

```

list gmp-static-group-info {
    description
        "Specify static multicast group information.";
    key "vrf-name addr-family if-name";
    config "false";

    leaf vrf-name {
        description "Specify the name of a vpn instance.";
        config "false";
        //default "_public_";
        type string {
            length "0..32";
        }
    }
    leaf addr-family {
        description
            "Specify an address family, which determines
            whether an address is an ipv4 or ipv6 address.";
        config "false";
        type enumeration {
            enum ipv4unicast {
                value "0";
                description "ipv4unicast:";
            }
            enum ipv6unicast {
                value "1";
                description "ipv6unicast:";
            }
        }
    }
}

```

```

    }
}
leaf if-name {
    description "Specify the name of an interface.";
    config "false";
    type if-name;
}
leaf ipv4-group-addr {
    description
        "Multicast group address. In ipv4 address, the
        value ranges from 239.0.0.0 to 239.255.255.255. ";
    config "false";
    type inet:ipv4-address;
}
leaf ipv6-group-addr {
    description
        "Multicast group address. In ipv6, the value is
        in hexadecimal notation and in the format of
        ffxx:xxxx:xxxx::xxxx ";

```

```

    config "false";
    type inet:ipv6-address;
}
leaf ipv4-source-addr {
    description
        "Specify the address of a multicast source.";
    config "false";
    type inet:ipv4-address;
}
leaf ipv6-source-addr {
    description
        "Specify the address of a multicast source.";
    config "false";
    type inet:ipv6-address;
}
leaf multicast-boundary {
    description
        "Whether the multicast forwarding boundary is
        configured.";
    config "false";
    type boolean;
}

```

```

    }

}

container gmp-if-infos {

    list gmp-if-info {

        description "Specify interface information.";
        key "vrf-name if-name addr-family";
        config "false";

        leaf vrf-name {
            description "Specify the name of a vpn instance.";
            config "false";
            //default "_public_";
            type string {
                length "0..32";
            }
        }
        leaf if-name {
            description "Specify the name of an interface.";
            config "false";
            type if-name;
        }
        leaf addr-family {

```

```

        description
            "Specify an address family, which determines
            whether an address is an ipv4 or ipv6 address.";
        config "false";
        type enumeration {
            enum ipv4unicast {
                value "0";
                description "ipv4unicast:";
            }
            enum ipv6unicast {
                value "1";
                description "ipv6unicast:";
            }
        }
    }
}

```

```

leaf ipv4-if-addr {
    description
        "Specify the ip address of an interface.";
    config "false";
    type inet:ipv4-address;
}
leaf ipv6-if-addr {
    description
        "Specify the ip address of an interface.";
    config "false";
    type inet:ipv6-address;
}
leaf gmp-if-state {
    description
        "Specify the status of an interface enabled with
        a protocol. The value can be up or down.";
    config "false";
    type enumeration {
        enum down {
            value "0";
            description "down:down";
        }
        enum up {
            value "1";
            description "up:up";
        }
    }
}
leaf negotiate-query-time {
    description
        "Interval negotiated by non-queriers for sending
        query messages, in seconds. The negotiated value
        is supported by igmpv3 or mldv2 only.";

```

```

        config "false";
        type uint16 {
            range "1..18000";
        }
    }
leaf configure-query-time {
    description
        "Interval configured by non-queriers for sending

```



```

        query messages, in seconds. ";
    config "false";
    type uint16 {
        range "1..18000";
    }
}
leaf other-querier-timeout {
    description
        "Specify the timeout period of other queriers.
        The value decreases by one per second. The value
        is 0 on the interface that acts as the querier.";
    config "false";
    type uint16 {
        range "1..65535";
    }
}
leaf max-query-response-time {
    description
        "Specify the maximum response time carried in a
        query message.";
    config "false";
    type uint8 {
        range "1..25";
    }
}
leaf last-member-query-time {
    description
        "Specify the timeout period for sending the
        last-member query message.";
    config "false";
    type uint8 {
        range "1..25";
    }
}
leaf startup-query-interval {
    description
        "Specify the interval at which an interface
        functioning as the querier sends query messages
        when it is started. ";
    config "false";

```

```

type uint16 {

```

```

        range "1..18000";
    }
}
leaf startup-query-count {
    description
        "Specify the number of times that an interface
        functioning as the querier sends query messages
        when it is started. ";
    config "false";
    type uint8 {
        range "2..5";
    }
}
leaf general-query-time {
    description
        "Specify the timeout period of a general query
        timer.";
    config "false";
    type uint16 {
        range "1..18000";
    }
}
leaf ipv4-querier-addr {
    description
        "Specify the address of a querier.";
    config "false";
    type inet:ipv4-address;
}
leaf ipv6-querier-addr {
    description
        "Specify the address of a querier.";
    config "false";
    type inet:ipv6-address;
}
leaf querier-local {
    description "Whether a query is a local query.";
    config "false";
    type enumeration {
        enum na {
            value "0";
            description "na:na";
        }
        enum no_querier {
            value "1";
            description "no_querier:no_querier";
        }
        enum this_router {

```

```
        value "2";
        description "this_router:this_router";
    }
    enum other_router {
        value "3";
        description "other_router:other_router";
    }
}
leaf gmp-joins {
    description
        "Specify the number of groups that an interface
        has joined. When the interface joins a new
        group, the value increases by one; when the
        interface leaves a group, the value remains
        unchanged";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf gmp-leaves {
    description
        "Specify the number of groups that the interface
        leaves. Each time when the interface leaves a
        group, the value increases by one.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf negotiate-robust {
    description
        "Specify the robustness variable negotiated by
        non-queriers. The value is supported by igmpv3
        or mldv2 only.";
    config "false";
    type uint8 {
        range "2..5";
    }
}
leaf configure-robust {
    description
        "Specify the robustness variable configured by
        non-queriers. ";
    config "false";
```

```
type uint8 {  
    range "2..5";
```

```
    }  
}  
leaf startup-query-timer-state {  
    description  
        "Specify the status of the timer for the interface  
        functioning as the querier after being started.  
        off: the interface immediately sends a query  
            message after being started.  
        on: the interface does not send a query message  
            after being started.";  
    config "false";  
    type enumeration {  
        enum off {  
            value "0";  
            description "off:off";  
        }  
        enum on {  
            value "1";  
            description "on:on";  
        }  
        enum na {  
            value "2";  
            description "na:na";  
        }  
    }  
}  
leaf other-querier-timer-state {  
    description  
        "Specify the status of the timer identifying  
        whether another querier is present.  
        off: the interface considers itself as a querier  
            and that no other queriers exist. ";  
    config "false";  
    type enumeration {  
        enum off {  
            value "0";  
            description "off:off";  
        }  
        enum on {
```

```

        value "1";
        description "on:on";
    }
}
leaf ipsec-name {
    description
        "Sa name. the value is a string of 1 to 15
        characters.";
}

```

```

        config "false";
        type string {
            length "1..15";
        }
    }
    leaf ipsec-type {
        description
            "Ipsec type used as a query option.";
        config "false";
        type enumeration {
            enum forall {
                value "0";
                description "ipsec sa for all:";
            }
            enum forquery {
                value "1";
                description "ipsec sa for query:";
            }
        }
    }
}

container gmp-packet-statistics {

    list gmp-packet-statistic {

        key "vrf-name addr-family if-name";
        config "false";

        leaf vrf-name {

```

```

        description "Specify the name of a vpn instance.";
        config "false";
        //default "_public_";
        type string {
            length "0..32";
        }
    }
    leaf if-name {
        description "Specify the name of an interface.";
        config "false";
        type if-name;
    }
    leaf addr-family {
        config "false";
        type enumeration {
            enum ipv4unicast {

```

```

            value "0";
            description "ipv4unicast:";
        }
        enum ipv6unicast {
            value "1";
            description "ipv6unicast:";
        }
    }
}
leaf ipv4-if-addr {
    description
        "Specify the ip address of an interface.";
    config "false";
    type inet:ipv4-address;
}
leaf ipv6-if-addr {
    description
        "Specify the ip address of an interface.";
    config "false";
    type inet:ipv6-address;
}
leaf query-recv-valid-num {
    description
        "Number of valid general query message received
        on the interface.";

```

```

        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf query-recv-invalid-num {
        description
            "Number of invalid general query message received
            on the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf query-recv-ignore-num {
        description
            "Number of ignored general query message received
            on the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
}

```

```

leaf report-v1v2-asm-recv-valid-num {
    description
        "Number of valid asm v1v2 report message received
        on the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v1v2-asm-recv-invalid-num {
    description
        "Number of invalid asm v1v2 report message received
        on the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}

```

```

leaf report-v1v2-asm-recv-ignore-num {
    description
        "Number of ignored asm v1v2 report message received
        on the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v1v2-ssm-recv-valid-num {
    description
        "Number of valid ssm v1v2 report message received
        on the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v1v2-ssm-recv-invalid-num {
    description
        "Number of invalid ssm v1v2 report message received
        on the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v1v2-ssm-recv-ignore-num {
    description
        "Number of ignored ssm v1v2 report message received

```

```

        on the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf leave-asm-recv-valid-num {
    description
        "Number of valid asm leave message received on
        the interface.";
    config "false";

```



```

        type uint32 {
            range "0..4294967295";
        }
    }
    leaf leave-asm-recv-invalid-num {
        description
            "Number of invalid asm leave message received on
            the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf leave-asm-recv-ignore-num {
        description
            "Number of ignored asm leave message received on
            the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf leave-ssm-recv-valid-num {
        description
            "Number of valid ssm leave message received on
            the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf leave-ssm-recv-invalid-num {
        description
            "Number of invalid ssm leave message received on
            the interface.";
        config "false";
        type uint32 {

```

```

            range "0..4294967295";
        }
    }
    leaf leave-ssm-recv-ignore-num {

```

```

        description
            "Number of ignored ssm leave message received on
            the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-isin-recv-valid-num {
        description
            "Number of valid isin message received on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-isin-recv-invalid-num {
        description
            "Number of invalid isin message received on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-isin-recv-ignore-num {
        description
            "Number of ignored isin message received on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-isex-recv-valid-num {
        description
            "Number of valid isex message received on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
}

```

```
leaf report-v3-isex-recv-invalid-num {
  description
    "Number of invalid isex message received on the
    interface.";
  config "false";
  type uint32 {
    range "0..4294967295";
  }
}
leaf report-v3-isex-recv-ignore-num {
  description
    "Number of ignored isex message received on the
    interface.";
  config "false";
  type uint32 {
    range "0..4294967295";
  }
}
leaf report-v3-toin-recv-valid-num {
  description
    "Number of valid toin message received on the
    interface.";
  config "false";
  type uint32 {
    range "0..4294967295";
  }
}
leaf report-v3-toin-recv-invalid-num {
  description
    "Number of invalid toin message received on the
    interface.";
  config "false";
  type uint32 {
    range "0..4294967295";
  }
}
leaf report-v3-toin-recv-ignore-num {
  description
    "Number of ignored toin message received on the
    interface.";
  config "false";
  type uint32 {
    range "0..4294967295";
  }
}
leaf report-v3-toex-recv-valid-num {
  description
```

```
        interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-toex-recv-invalid-num {
        description
            "Number of invalid toex message received on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-toex-recv-ignore-num {
        description
            "Number of ignored toex message received on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-allow-recv-valid-num {
        description
            "Number of valid allow message received on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-allow-recv-invalid-num {
        description
            "Number of invalid allow message received on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
```

```

    }
}
leaf report-v3-allow-recv-ignore-num {
    description
        "Number of ignored allow message received on the
        interface.";
    config "false";
    type uint32 {

```

```

        range "0..4294967295";
    }
}
leaf report-v3-block-recv-valid-num {
    description
        "Number of valid block message received on the
        interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v3-block-recv-invalid-num {
    description
        "Number of invalid block message received on the
        interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v3-block-recv-ignore-num {
    description
        "Number of ignored block message received on the
        interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf source-recv-valid-num {
    description
        "Number of valid source received on the interface.";

```

```

        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf source-recv-invalid-num {
        description
            "Number of invalid source received on the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf source-recv-ignore-num {
        description

```

```

        "Number of ignored source received on the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf query-send-num {
        description
            "Specifies count of send query message on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v1v2-asm-send-num {
        description
            "Specifies count of send asm v1v2 report message
            on the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v1v2-ssm-send-num {
        description

```

```

        "Specifies count of send ssm vlv2 report message
        on the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf leave-asm-send-num {
    description
        "Specifies count of send asm leave message on
        the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf leave-ssm-send-num {
    description
        "Specifies count of send ssm leave message on
        the interface.";
    config "false";
    type uint32 {

```

```

        range "0..4294967295";
    }
}
leaf query-g-send-num {
    description
        "Specifies count of send (g) query message on
        the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf query-sg-send-num {
    description
        "Specifies count of send (sg) query message on
        the interface.";
    config "false";
    type uint32 {
        range "0..4294967295";

```

```

    }
}
leaf report-v3-isin-send-num {
    description
        "Specifies count of send isin message on the
        interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v3-isex-send-num {
    description
        "Specifies count of send isex message on the
        interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v3-toin-send-num {
    description
        "Specifies count of send toin message on the
        interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
}

```

```

leaf report-v3-toex-send-num {
    description
        "Specifies count of send toex message on the
        interface.";
    config "false";
    type uint32 {
        range "0..4294967295";
    }
}
leaf report-v3-allow-send-num {
    description
        "Specifies count of send allow message on the

```



```

        interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf report-v3-block-send-num {
        description
            "Specifies count of send block message on the
            interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf source-send-num {
        description
            "Specifies count of send source on the interface.";
        config "false";
        type uint32 {
            range "0..4294967295";
        }
    }
}

}

notification gmp-group-join {

    description "Receive a group membership report";
    leaf vrf-name {
        description "Specify the name of a vpn instance.";
        //default "_public_";
    }
}

```

```

        type string {
            length "0..32";
        }
    }
    leaf if-name {

```

```

        description "Specify the name of an interface.";
        type if-name;
    }
    leaf addr-family {
        type enumeration {
            enum ipv4unicast {
                value "0";
                description "ipv4unicast:";
            }
            enum ipv6unicast {
                value "1";
                description "ipv6unicast:";
            }
        }
    }
}
leaf ipv4-group-addr {
    description
        "Multicast group address. In ipv4 address, the value
        ranges from 239.0.0.0 to 239.255.255.255. ";
    type inet:ipv4-address;
}
leaf ipv6-group-addr {
    description
        "Multicast group address. In ipv6, the value is in
        hexadecimal notation and in the format of
        ffx:xxxx:xxxx::xxxx ";
    type inet:ipv6-address;
}
leaf ipv4-source-addr {
    description
        "Specify the address of a multicast source.";
    type inet:ipv4-address;
}
leaf ipv6-source-addr {
    description
        "Specify the address of a multicast source.";
    type inet:ipv6-address;
}
leaf ipv4-sender-addr {
    description "Specify the address of sender.";
    type inet:ipv4-address;
}
leaf ipv6-sender-addr {

```

```

        description "Specify the address of sender.";
        type inet:ipv6-address;
    }

}

notification gmp-group-leave {

    description "Receive a group membership leave";
    leaf vrf-name {
        description "specify the name of a vpn instance.";
        //default "_public_";
        type string {
            length "0..32";
        }
    }
    leaf if-name {
        description "Specify the name of an interface.";
        type if-name;
    }
    leaf addr-family {
        type enumeration {
            enum ipv4unicast {
                value "0";
                description "ipv4unicast:";
            }
            enum ipv6unicast {
                value "1";
                description "ipv6unicast:";
            }
        }
    }
}
leaf ipv4-group-addr {
    description
        "Multicast group address. In ipv4 address, the value
        ranges from 239.0.0.0 to 239.255.255.255. ";
    type inet:ipv4-address;
}
leaf ipv6-group-addr {
    description
        "Multicast group address. In ipv6, the value is in
        hexadecimal notation and in the format of
        ffx:xxxx:xxxx::xxxx ";
    type inet:ipv6-address;
}
leaf ipv4-source-addr {
    description
        "Specify the address of a multicast source.";

```

Internet-Draft

GMP YANG model

March 2015

```
        type inet:ipv4-address;
    }
    leaf ipv6-source-addr {
        description
            "Specify the address of a multicast source.";
        type inet:ipv6-address;
    }
    leaf ipv4-sender-addr {
        description "Specify the address of sender.";
        type inet:ipv4-address;
    }
    leaf ipv6-sender-addr {
        description "Specify the address of sender.";
        type inet:ipv6-address;
    }
}
}
```

[6.](#) IANA Considerations

This draft includes no request to IANA.

[7.](#) Security Considerations

The data model defined does not create any security implications. This draft does not change any underlying security issues inherent in [\[I-D.ietf-netmod-routing-cfg\]](#).

[8.](#) Acknowledgements

TBD

[9.](#) References

[9.1.](#) Normative References

[I-D.ietf-netmod-routing-cfg]

Lhotka, L. and A. Lindem, "A YANG Data Model for Routing Management", [draft-ietf-netmod-routing-cfg-17](#) (work in progress), March 2015.

- [RFC1112] Deering, S., "Host extensions for IP multicasting", STD 5, [RFC 1112](#), August 1989.
- [RFC2236] Fenner, W., "Internet Group Management Protocol, Version 2", [RFC 2236](#), November 1997.

- [RFC2710] Deering, S., Fenner, W., and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", [RFC 2710](#), October 1999.
- [RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", [RFC 3376](#), October 2002.
- [RFC3810] Vida, R. and L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", [RFC 3810](#), June 2004.
- [RFC4604] Holbrook, H., Cain, B., and B. Haberman, "Using Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Protocol Version 2 (MLDv2) for Source-Specific Multicast", [RFC 4604](#), August 2006.
- [RFC4607] Holbrook, H. and B. Cain, "Source-Specific Multicast for IP", [RFC 4607](#), August 2006.
- [RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](#), October 2010.
- [RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", [RFC 6241](#), June 2011.

[9.2](#). Informative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

Authors' Addresses

Yisong Liu

Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: liuyisong@huawei.com

Liu & Guo

Expires September 23, 2015

[Page 54]

Internet-Draft

GMP YANG model

March 2015

Feng Guo
Huawei
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: guofeng@huawei.com

