Internet Engineering Task Force
Internet Draft
Intended status: Standards Track
Expires: August 2017

H. Zhao
Ericsson

February 6, 2017

A Yang Data Model for IGMP and MLD Snooping
draft-zhao-pim-igmp-mld-snooping-yang-00.txt

Abstract

This document defines a YANG data model that can be used to configure and manage Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping devices.

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), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on August 6, 2017.

Table of Contents

1. Introduction

This document defines a YANG [RFC6020] data model for the management of Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping devices.

This data model includes configuration data and state data (status information and counters for the collection of statistics).

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119].

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

1.2. 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:

- Brackets "[" and "]" enclose list keys.
- Abbreviations before data node names: "rw" means configuration (read-write), and "ro" means state data (read-only).
- Symbols after data node names: "?" means an optional node, "!" means a presence container, and "*" denotes a list and leaf-list.
- Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").
- Ellipsis ("...") stands for contents of subtrees that are not shown.

2. Design of Data Model

The model covers Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches [RFC4541].

The goal of this document is to define a data model that provides a common user interface to IGMP and MLD Snooping. There is very information that is designated as "mandatory", providing freedom for vendors to adapt this data model to their respective product implementations.

2.1. Overview

The IGMP and MLD Snooping YANG module defined in this document has all the common building blocks for the IGMP and MLD Snooping protocol. The YANG module includes IGMP and MLD Snooping configuration data, Operational state data, and the rpc method for clearing the specified IGMP and MLD Snooping statistics.

2.2. IGMP and MLD Snooping Configuration

The IGMP and MLD Snooping modules define the configuration in a two-level hierarchy as listed below:

Global level: IGMP MLD Snooping configuration attributes for the entire routing instance

VLAN-level: IGMP MLD configuration attributes specific to the given VLAN.

```
module: ietf-igmp-mld-snooping

   +--rw configuration

   |  +--rw igmp-snooping

   |  |  +--rw global

   |  |  |  +--rw enable?   boolean {global-admin-enable}?

   |  |  +--rw vlans

   |  |     +--rw vlan* [vlan-id]

   |  |        +--rw vlan-id                       uint16

   |  |        +--rw enable?                       boolean {admin-enable}?

   |  |        +--rw forwarding-mode?              enumeration

   |  |        +--rw explicit-tracking?            string

   |  |        +--rw send-query?                   boolean

   |  |        +--rw mrouter-learning?             boolean

   |  |        +--rw mrouter-aging-time?           uint16

   |  |        +--rw fast-leave?                   empty {fast-leave}?

   |  |        +--rw last-member-query-interval?   uint16

   |  |        +--rw query-interval?               uint16

   |  |        +--rw max-response-time?            uint16

   |  |        +--rw require-router-alert?         boolean {require-router-
alert}?

   |  |        +--rw robustness-variable?          uint8
```

```
      |  |        +--rw source-policy?               string {source-policy}?
      |  |        +--rw verify-source-subnet?        empty {verify-source-
subnet}?
      |  |        +--rw version?                     uint8
      |  |        +--rw ssm-map* [source-addr group-addr-prefix] {ssm-map}?
      |  |        |  +--rw source-addr          ssm-map-ipv4-addr-type
      |  |        |  +--rw group-addr-prefix    inet:ipv4-prefix
      |  |        +--rw static-mrouter-interface*     if:interface-ref
      |  |        +--rw static-l2-multicast-group* [group]
      |  |           +--rw group        inet:ipv4-address
      |  |           +--rw interface?   if:interface-ref
      |  +--rw mld-snooping
      |     +--rw global
      |     |  +--rw enable?   boolean {global-admin-enable}?
      |     +--rw vlans
      |        +--rw vlan* [vlan-id]
      |           +--rw vlan-id                      uint16
      |           +--rw enable?                      boolean {admin-enable}?
      |           +--rw forwarding-mode?             enumeration
      |           +--rw explicit-tracking?           string
      |           +--rw send-query?                  boolean
      |           +--rw mrouter-learning?            boolean
      |           +--rw mrouter-aging-time?          uint16
      |           +--rw fast-leave?                  empty {fast-leave}?
      |           +--rw last-member-query-interval?  uint16
```

```
      |           +--rw query-interval?              uint16
      |           +--rw max-response-time?           uint16
      |           +--rw require-router-alert?        boolean {require-router-
alert}?
      |           +--rw robustness-variable?         uint8
      |           +--rw source-policy?               string {source-policy}?
      |           +--rw verify-source-subnet?        empty {verify-source-
subnet}?
      |           +--rw version?                     uint8
      |           +--rw ssm-map* [source-addr group-addr-prefix] {ssm-map}?
      |           |  +--rw source-addr          ssm-map-ipv6-addr-type
      |           |  +--rw group-addr-prefix    inet:ipv6-prefix
      |           +--rw static-mrouter-interface*     if:interface-ref
      |           +--rw static-l2-multicast-group* [group]
      |              +--rw group        inet:ipv4-address
      |              +--rw interface?   if:interface-ref
```

2.3. IGMP and MLD Snooping Operational State

The IGMP and MLD Snooping module contains operational state information also in a two-level hierarchy as mentioned earlier.

Global level: IGMP MLD Snooping operational state attributes for the entire routing instance

VLAN-level: IGMP MLD Snooping operational state attributes specific to the given VLAN.

```
module: ietf-igmp-mld-snooping
   +--rw configuration
   ...
   +--ro state
```

```
         +--ro igmp-snooping
         |  +--ro global
         |  |  +--ro enable?          boolean {global-admin-enable}?
         |  |  +--ro entries-count?   uint32
         |  +--ro vlans
         |     +--ro vlan* [vlan-id]
         |        +--ro vlan-id                       uint16
         |        +--ro enable?                       boolean {admin-enable}?
         |        +--ro forwarding-mode?              enumeration
         |        +--ro explicit-tracking?            string
         |        +--ro send-query?                   boolean
         |        +--ro mrouter-learning?             boolean
         |        +--ro mrouter-aging-time?           uint16
         |        +--ro fast-leave?                   empty {fast-leave}?
         |        +--ro last-member-query-interval?   uint16
         |        +--ro query-interval?               uint16
         |        +--ro max-response-time?            uint16
         |        +--ro require-router-alert?         boolean {require-router-
alert}?
         |        +--ro robustness-variable?          uint8
         |        +--ro source-policy?                string {source-policy}?
         |        +--ro verify-source-subnet?         empty {verify-source-
subnet}?
         |        +--ro version?                      uint8
         |        +--ro ssm-map* [source-addr group-addr-prefix] {ssm-map}?
         |        |  +--ro source-addr          ssm-map-ipv4-addr-type
```

```
|       |  +--ro group-addr-prefix    inet:ipv4-prefix
|       +--ro mrouter*                      if:interface-ref
|       +--ro port-info*                    if:interface-ref
|       +--ro mac-address*                  yang:phys-address
|       +--ro querier?                      inet:ipv4-address
|       +--ro group* [address]
|          +--ro address    inet:ipv4-address
|          +--ro source* [address]
|             +--ro address        inet:ipv4-address
|             +--ro reporter?      inet:ipv4-address
|             +--ro filter-mode?   enumeration
|             +--ro interface?     if:interface-ref
|             +--ro up-time?       uint32
|             +--ro last-join?     uint32
|             +--ro last-leave?    uint32
+--ro mld-snooping
   +--ro global
   |  +--ro enable?          boolean {global-admin-enable}?
   |  +--ro entries-count?   uint32
   +--ro vlans
      +--ro vlan* [vlan-id]
         +--ro vlan-id                     uint16
         +--ro enable?                     boolean {admin-enable}?
         +--ro forwarding-mode?            enumeration
```

```
                +--ro explicit-tracking?           string
                +--ro send-query?                  boolean
                +--ro mrouter-learning?            boolean
                +--ro mrouter-aging-time?          uint16
                +--ro fast-leave?                  empty {fast-leave}?
                +--ro last-member-query-interval?  uint16
                +--ro query-interval?              uint16
                +--ro max-response-time?           uint16
                +--ro require-router-alert?        boolean {require-router-
alert}?
                +--ro robustness-variable?         uint8
                +--ro source-policy?               string {source-policy}?
                +--ro verify-source-subnet?        empty {verify-source-
subnet}?
                +--ro version?                     uint8
                +--ro ssm-map* [source-addr group-addr-prefix] {ssm-map}?
                |  +--ro source-addr          ssm-map-ipv6-addr-type
                |  +--ro group-addr-prefix    inet:ipv6-prefix
                +--ro mrouter*                     if:interface-ref
                +--ro port-info*                   if:interface-ref
                +--ro mac-address*                 yang:phys-address
                +--ro querier?                     inet:ipv6-address
                +--ro group* [address]
                   +--ro address    inet:ipv6-address
                   +--ro source* [address]
                      +--ro address        inet:ipv6-address
```

```
                        +--ro reporter?      inet:ipv6-address
                        +--ro filter-mode?   enumeration
                        +--ro interface?     if:interface-ref
                        +--ro up-time?       uint32
                        +--ro last-join?     uint32
                        +--ro last-leave?    uint32
```

2.4. IGMP and MLD Snooping RPC

IGMP and MLD Snooping RPC clears the specified IGMP and MLD Snooping statistics.

```
rpcs:
   +---x clear-igmp-mld-snooping-statistics     {rpc-clear-statistics}?
      +---w input
         +---w vlan-id?   uint16
```

3. IGMP and MLD Snooping YANG Module

```
<CODE BEGINS> file "ietf-igmp-mld-snooping@2017-02-05.yang"
module ietf-igmp-mld-snooping {
      namespace "urn:ietf:params:xml:ns:yang:ietf-igmp-mld-snooping";
      // replace with IANA namespace when assigned
      prefix igmp-mld-snooping;

      import ietf-inet-types {
        prefix "inet";
      }

      import ietf-yang-types {
        prefix "yang";
      }

      import ietf-interfaces {
        prefix "if";
```

```
        }

        organization
          "IETF PIM Working Group";

        contact
          "WG Web:   <http://tools.ietf.org/wg/pim/>
           WG List:  <mailto:pim@ietf.org>

           WG Chair: Stig Venaas
                     <mailto:stig@venaas.com>

           WG Chair: Mike McBride
                     <mailto:mmcbride7@gmail.com>

           Editors:  Hongji Zhao
                 <mailto:hongji.zhao@ericsson.com> ";

        description
          "The module defines a collection of YANG definitions common
   for
          IGMP and MLD Snooping.";

        revision 2017-02-05 {
          description
            "Initial revision.";
          reference
            "RFC XXXX: A YANG Data Model for IGMP and MLD Snooping";
        }

        /*
         * Features
         */
        feature global-admin-enable {
          description
            "Support global configuration to enable or disable IGMP and
   MLD Snooping.";
        }

        feature admin-enable {
          description
            "Support configuration to enable or disable protocol under
   VLAN instance.";
        }

        feature fast-leave {
          description
```

```
            "Support configuration of fast-leave.";
        }

        feature join-group {
          description
            "Support configuration of join-group.";
        }

        feature require-router-alert {
          description
            "Support configuration of require-router-alert.";
        }

        feature source-policy {
          description
            "Support configuration of source policy.";
        }

        feature ssm-map {
          description
            "Support configuration of ssm-map.";
        }

        feature static-l2-multicast-group {
          description
            "Support configuration of L2 multicast static-group.";
        }

        feature verify-source-subnet {
          description
            "Support configuration of verify-source-subnet.";
        }

        feature per-vlan-config {
          description
            "Support configuration of each VLAN.";
        }

       feature rpc-clear-statistics {
          description
            "Support to clear statistics by RPC for IGMP and MLD
   Snooping.";
        }


        /*
         * Typedefs
```

```
 */
typedef ssm-map-ipv4-addr-type {
  type union {
    type enumeration {
      enum 'policy' {
        description
        "Source address in SSM map policy.";
      }
    }
    type inet:ipv4-address;
  }
  description
    "Source IPV4 address type for SSM map.";
} // ssm-map-ipv4-addr-type

typedef ssm-map-ipv6-addr-type {
  type union {
    type enumeration {
      enum 'policy' {
        description
        "Source address in SSM map policy.";
      }
    }
    type inet:ipv6-address;
  }
  description
    "Source IPV6 address type for SSM map.";
} // ssm-map-ipv6-addr-type



/*
 * Identities
 */

/*
 * Groupings
 */
grouping global-config-attributes {
  description "Global configuration of IGMP and MLD Snooping.";

  leaf enable {
    if-feature global-admin-enable;
    type boolean;

    description
```

```
      "Set the value to true to enable IGMP or MLD Snooping in the
   global view";
          }


        } // global-config-attributes

        grouping global-state-attributes {
          description  "Global attributes of IGMP and MLD Snooping
   state";
          leaf entries-count {
            type uint32;
            description
              "The number of L2 multicast entries in IGMP and MLD
   Snooping.";
          }


        } // global-state-attributes

        grouping vlan-config-attributes-igmp-snooping {
          description "IGMP snooping configuration of each VLAN.";

          uses vlan-config-attributes-igmp-mld-snooping;

          list ssm-map {

            if-feature ssm-map;
            key "source-addr group-addr-prefix";
          description "SSM mapping list for igmp snooping";

            leaf source-addr {
              type ssm-map-ipv4-addr-type;
              description
                "Source IP address.";
            }
            leaf group-addr-prefix {
              type inet:ipv4-prefix;
              description
                "Multicast group IP address for ssm-map in the format
   of x.x.x.x/x ";
            }
          }


        } // vlan-config-attributes-igmp-snooping
```

```
        grouping vlan-config-attributes-igmp-mld-snooping {
          description
            "IGMP and MLD Snooping configuration of each VLAN.";

          leaf enable {
            if-feature admin-enable;
            type boolean;
            description
              "Set the value to true to enable IGMP and MLD Snooping in
   the VLAN instance.";
          }

        leaf forwarding-mode {
           type enumeration {
              enum "mac" {
               description
                 "";
              }
              enum "ip"  {
               description
                 "";
              }
           }
           description "The default forwarding mode for IGMP and MLD
   Snooping is ip.";
        }

        leaf explicit-tracking {
           when "../version = 3";
           type string;
           description "Tracks IGMPv3 membership reports from
   individual hosts for each port of each VLAN. ";
        }

        leaf send-query {
           type boolean;
           default true;
           description "Enable quick response for topo changes";
        }

        leaf mrouter-learning {
           type boolean;
           default true;
           description
              "Enable router port learning function";
        }
```

```
        leaf mrouter-aging-time {
           type uint16 ;
           default 180;
           description "Aging time for mrouter port";
        }


          leaf fast-leave {
            if-feature fast-leave;
            type empty;
            description
              "When fast leave is enabled, the IGMP software assumes
   that no more than one host is present on each VLAN port.";
          }
          leaf last-member-query-interval {
            type uint16 {
              range "1..65535";
            }
            description
              " Modify the value to adjust the leave latency of the
   network.";
          }

          leaf query-interval {
            type uint16;
            units seconds;
            default 125;
            description
              "Interval between general queries sent by the querier.";
          }
          leaf max-response-time {
            type uint16;
            units seconds;
            description
              "Maximum time for the host to respond the query packet.";

          }
          leaf require-router-alert {
            if-feature require-router-alert;
            type boolean;
            description
              "When the value is true, router alert exists in the IP
   head of IGMP or MLD packet.";
          }
          leaf robustness-variable {
            type uint8 {
              range "2..7";
```

```
            }
            default 2;
            description
              "Adjust the value for the expected packet loss on the
   network.";

          }
          leaf source-policy {
            if-feature source-policy;
            type string;
            description
              "Name of the access policy used to filter sources.";
          }
          leaf verify-source-subnet {
            if-feature verify-source-subnet;
            type empty;
            description
              "If the value is present, the interface accepts packets
   with matching source IP subnet only.";
          }
          leaf version {
            type uint8 {
              range "1..3";
            }
            description "IGMP and MLD Snooping version.";
          }
        } // vlan-config-attributes-igmp-mld-snooping

        grouping vlan-config-attributes-mld-snooping {
          description "MLD snooping configuration of each VLAN.";

          uses vlan-config-attributes-igmp-mld-snooping;


          list ssm-map {
            if-feature ssm-map;
            key "source-addr group-addr-prefix";
            description "SSM mapping list for MLD snooping";
            leaf source-addr {
              type ssm-map-ipv6-addr-type;
              description
                "Source IP address.";
            }
            leaf group-addr-prefix {
              type inet:ipv6-prefix;
              description
                "Multicast group IP address for ssm-map";
```

```
            }
          }

        } // vlan-config-attributes-mld-snooping

        grouping vlan-state-attributes-igmp-snooping {
          description
            "State attributes for IGMP snooping of each VLAN.";

          uses vlan-state-attributes-igmp-mld-snooping;

          leaf querier {
            type inet:ipv4-address;
            description "Configures a snooping querier when PIM is
   disabled because multicast traffic does not need to be routed. The IP
   address is used as the source address in messages.";
          }


          list group {
            key "address";
            description "IGMP snooping information";

            leaf address {
              type inet:ipv4-address;
              description
                "Multicast group IP address";
            }


            list source {
              key "address";
             description "Source IP address for multicast stream";
              leaf address {
                type inet:ipv4-address;
                description "Source IP address for multicast stream";
              }

             leaf reporter {
                type inet:ipv4-address;
                description "The reporter IP address behind the port";
             }

                uses vlan-state-group-attributes-igmp-mld-snooping;

            } // list source
          } // list group
```

```
        } // vlan-state-attributes-igmp-snooping

        grouping vlan-state-attributes-igmp-mld-snooping {
          description
            "State attributes for both IGMP and MLD Snooping of each
   VLAN.";

          leaf-list mrouter {
           type if:interface-ref;
           description "The interfaces that connect to the multicast
   router";
          }

          leaf-list port-info {
           type if:interface-ref;
           description "The port-info shows the interfaces that attach
   the hosts, which join in the multicast group";
          }

          leaf-list mac-address {
           type yang:phys-address;
           description "Destination mac address for L2 multicast
   forwarding.";
          }

        } // vlan-config-attributes-igmp-mld-snooping

        grouping vlan-state-attributes-mld-snooping {
          description
            "State attributes for MLD snooping of each VLAN.";

          uses vlan-state-attributes-igmp-mld-snooping;

          leaf querier {
            type inet:ipv6-address;
            description
           "Configures a snooping querier when PIM is disabled because
   multicast traffic does not need to be routed.";
          }

          list group {
            key "address";
            description "MLD snooping statistics information";

            leaf address {
              type inet:ipv6-address;
              description
```

```
                "Multicast group IP address";
             }

             list source {
               key "address";
             description "Source IP address for multicast stream";

               leaf address {
                 type inet:ipv6-address;
                 description "Source IP address for multicast stream";
               }

             leaf reporter {
               type inet:ipv6-address;
               description "The reporter IP address behind the port";
             }

               uses vlan-state-group-attributes-igmp-mld-snooping;

              } // list source
            } // list group
          } // vlan-state-attributes-mld-snooping

          grouping vlan-state-group-attributes-igmp-mld-snooping {
            description
              "State attributes for both IGMP and MLD Snooping of each
     VLAN.";

            leaf filter-mode {
              type enumeration {
                enum "include" {
                  description
                  "";
                }
                enum "exclude" {
                  description
                  "";
                }
              }
              description "Allow (include) or block (exclude) filtering
     ";
            }

          leaf interface {
             type if:interface-ref;
             description
             "The outgoing interface for L2 multicast stream.";
```

```
        }

          leaf up-time {
            type uint32;
            units seconds;
            description "The up time for the interface.";
          }

        leaf last-join {
           type uint32;
           units seconds;
           description "The last-join information collected since a
   clear command was entered last time.";
        }

        leaf last-leave {
           type uint32;
           units seconds;
           description "The last-leave information collected since a
   clear command was entered last time.";
        }

        } // vlan-state-group-attributes-igmp-mld-snooping



      grouping vlan-config-static-l2-multicast-group {
         description
           "Configures a L2 static connection to a multicast
   receiver.";

        list static-l2-multicast-group {

           key group;
           description "Multicast group IP address";

           leaf group {
              type inet:ipv4-address;
              description
              "Multicast group IP address";
           }
           leaf interface {
              type if:interface-ref;
              description
              "Interface for static multicast group";
           }
        }
```

```
        } //vlan-config-static-l2-multicast-group

         /*
          * Configuration data nodes
          */
         container configuration
         {
           description
             "IGMP and MLD Snooping configuration data.";

           container igmp-snooping {
             description
               "IGMP Snooping configuration data.";

             container global {
               description
                 "Global attributes.";
               uses global-config-attributes;
             }

             container vlans {
               description
                 "List of VLANs that enable IGMP Snooping.";


               list vlan {
                 key "vlan-id";
                 description
                   "IGMP Snooping config-attributes for each VLAN.";
                 leaf vlan-id {
                   type uint16 {
                  range "1 .. 4094";
               }
                     description
                       "VLAN ID, the range of which is 1 to 4094.";
                 }
                 uses vlan-config-attributes-igmp-snooping {
                   if-feature per-vlan-config;
                 }

              leaf-list static-mrouter-interface {
                type if:interface-ref;
                description "Configures a static connection to a
   multicast router. The interface to the router must be in the selected
   VLAN. The interface is specified by type and number.";
            }
```

```
             uses vlan-config-static-l2-multicast-group {
              if-feature static-l2-multicast-group;
             }

              } // vlan
            } // vlans
          } // igmp-snooping

          container mld-snooping {
            description
              "MLD Snooping configuration data.";

            container global {
              description
                "Global attributes.";
              uses global-config-attributes;
            }

            container vlans {
              description
                "List of VLANs that enable IGMP Snooping.";


              list vlan {
                key "vlan-id";
                description
                  "IGMP Snooping configuration attributes for each
   VLAN.";

                leaf vlan-id {
                  type uint16 {
                 range "1 .. 4094";
              }
                  description
                    "VLAN ID, the range of which is 1 to 4094";
                }

                uses vlan-config-attributes-mld-snooping {
                  if-feature per-vlan-config;
                }

              leaf-list static-mrouter-interface {
               type if:interface-ref;
               description "Configures a static connection to a
   multicast router. The interface to the router must be in the selected
   VLAN. The interface is specified by type and number.";
```

```
            }

            uses vlan-config-static-l2-multicast-group {
             if-feature static-l2-multicast-group;
            }


             } // vlan
           } // vlans
         } // mld-snooping
       } //

       /*
        * Operational state data nodes
        */
       container state
       {
       config false;

         description
           "IGMP and MLD Snooping state data.";

         container igmp-snooping {
           description
             "IGMP Snooping operational state data.";

           container global {
             description
               "Global attributes.";
             uses global-config-attributes;
             uses global-state-attributes;
           }

           container vlans{
             description
               " List of VLANs that enable IGMP Snooping.";

             list vlan {
               key "vlan-id";
               description
                 "IGMP Snooping state-attributes for each VLAN.";
               leaf vlan-id {
                 type uint16 {
                range "1 .. 4094";
             }
                 description
                   "VLAN ID, the range of which is 1 to 4094 ";
```

```
            }
            uses vlan-config-attributes-igmp-snooping {
              if-feature per-vlan-config;
            }
            uses vlan-state-attributes-igmp-snooping;
          } // vlan
        } // vlans
      } // igmp-snooping

      container mld-snooping {
        description
          "MLD Snooping state data.";

        container global {
          description
            "Global attributes.";
          uses global-config-attributes;
          uses global-state-attributes;
        }

        container vlans {
          description
            "List of VLANs that enable MLD Snooping.";

          list vlan {
            key "vlan-id";
            description
              "MLD Snooping state-attributes for each VLAN.";

            leaf vlan-id {
              type uint16 {
               range "1 .. 4094";
          }
              description
                "VLAN ID, the range of which is 1 to 4094 ";
            }
            uses vlan-config-attributes-mld-snooping {
              if-feature per-vlan-config;
            }
            uses vlan-state-attributes-mld-snooping;
          } // vlan
        } // vlans
      } // mld-snooping
    } //

    /*
     * RPCs
```

```
          */
         //clear ip igmp snooping statistics interface vlan-id

        rpc clear-igmp-mld-snooping-statistics {
           if-feature rpc-clear-statistics;
           description
             "Clears the specified IGMP and MLD Snooping statistics.";

           input {
             leaf vlan-id {
                   type uint16 {
                  range "1 .. 4094";
               }
                   description
                     "VLAN ID, the range of which is 1 to 4094 ";
             }

           }
         } // rpc clear-igmp-mld-snooping-statistics


        /*
         * Notifications
         */
      }
<CODE ENDS>
```

4. Security Considerations

The data model defined does not create any security implications.

5. IANA Considerations

This draft does not request any IANA action.

6. Normative References

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6021] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6021, October 2010.

[RFC4541] M. Christensen, K. Kimball, F. Solensky, "Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches", RFC 4541, May 2006.

[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 InternetGroup 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.

[draft-ietf-pim-igmp-mld-yang-01] X. Liu, F. Guo, M. Sivakumar, P. McAllister, A. Peter, "A YANG data model for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD)", draft-ietf-pim-igmp-mld-yang-01, October 28, 2016.

Authors' Addresses

Hongji Zhao
Ericsson (China) Communications Company Ltd.
Ericsson Tower, No. 5 Lize East Street,
Chaoyang District Beijing 100102, P.R. China

Email: hongji.zhao@ericsson.com