A YANG model for Multicast Protocol for Low power and lossy Networks (MPL)
draft-vanderstok-roll-mpl-yang-01

Abstract

   This document defines a YANG data model for management of Multicast
   Protocol for Low power and lossy Networks (MPL) implementations. The
   data model includes configuration data and state data.

Note

   Discussion and suggestions for improvement are requested, and should
   be sent to roll@ietf.org.

   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

   This document defines a YANG [RFC6020] data model for management of
   Multicast Protocol for Low power and lossy Networks (MPL) [RFC7731]
   implementations. The data model covers configuration of per-
   interface MPL parameters. It also provides information about which
   Multicast addresses are operationally used, and the seeds for which
   packets are forwarded

1.1. Terminology

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

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

   o client

   o configuration data

   o server

o   state data

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

o   data model

o   data node

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

Terms like domain, seed, I, k, c are defined in [RFC7731].

1.1.1. Tree Diagrams

A simplified graphical representation of the data model is used in
the YANG modules specified 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
   data (read-write) and "ro" 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. MPL model

This document defines the YANG module "ietf-yang-mpl", which
specifies a data model for MPL servers. The model consists of the
following parts: (1) a "mpl-domain" part that describes the MPL-
domains and associated Multicast addresses and the interfaces on
which the Multicast addresses are enabled, (2) a "mpl-op" part that
describes the parameters settings per seed, (3) a "mpl-seeds" part
that describes the MPL buffer contents and the Trickle timer values,
and (4) a "mpl-statistics" part that describes the number of lost and
correctly forwarded messages. The data model, divided in four
modules, has the following structure for MPL configuration per node:

```
module: ietf-yang-mpl-domain
   +--rw (single)?
      +--:(mpl-domain)
      |  +--rw mpl-domain
      |     +--rw domains* [domainID]
      |     |  +--rw domainID    uint16
      |     |  +--rw MClist*      inet:ipv6-address
      |     +--rw addresses* [MCaddress]
      |        +--rw MCaddress     inet:ipv6-address
      |        +--rw interfaces*   string
      +--:(mpl-single)
         +--rw mpl-single
            +--rw MCaddresses*   inet:ipv6-address

module: ietf-yang-mpl-ops
 +--rw mpl-ops
    +--rw SE_LIFETIME?              uint16
    +--rw PROACTIVE_FORWARDING?     boolean
    +--rw SEED_SET_ENTRY_LIFETIME?    uint64
    +--rw mpl-parameter* [domainID]
       +--rw domainID                            uint16
       +--rw DATA_MESSAGE_IMIN?                  uint16
       +--rw DATA_MESSAGE_IMAX?                  uint16
       +--rw DATA_MESSAGE_K?                     uint16
       +--rw DATA_MESSAGE_TIMER_EXPIRATIONS?     uint16
       +--rw CONTROL_MESSAGE_IMIN?               uint16
       +--rw CONTROL_MESSAGE_IMAX?               uint16
       +--rw CONTROL_MESSAGE_K?                  uint16
       +--rw CONTROL_MESSAGE_TIMER_EXPIRATIONS?    uint16

module: ietf-yang-mpl-seeds
 +--ro mpl-seeds* [seedID domainID]
    +--ro seedID              uint64
    +--ro domainID            uint16
    +--ro local?              boolean
    +--ro generate-seqno?     uint8
    +--ro life-time?          uint64
    +--ro min-seqno?          uint8
    +--ro data-number?        uint8
    +--ro control-number?     uint8
    +--ro buffered-messages* [seqno]
       +--ro seqno     uint8
       +--ro I?        uint8
       +--ro c?        uint8
       +--ro e?        uint8
       +--ro t?        uint8

module: ietf-yang-mpl-statistics
```

```
      +--ro mpl-statistics* [seedID domainID]
         +--ro seedID                         uint64
         +--ro domainID                       uint16
         +--ro c-too-high?                    uint64
         +--ro nr-forwarded?                  uint64
         +--ro nr-of-messages-received?       uint64
         +--ro nr-of-copies-received?         uint64
         +--ro nr-of-messages-forwarded?      uint64
         +--ro nr-of-copies-forwarded?        uint64
         +--ro nr-of-refused?                 uint64
         +--ro nr-of-missed?                  uint64
         +--ro nr-of-notreceived?             uint64
         +--ro nr-of-inconsistent-data?       uint64
         +--ro nr-of-consistent-data?         uint64
         +--ro nr-of-consistent-control?      uint64
         +--ro nr-of-inconsistent-control?    uint64
```

3. yang-mpl modules

   This section describes four yang modules. The model is based on the
   MPL specification published in [RFC7731] and the specification of
   [RFC6206]. The identification of the interfaces follows the
   specification of ietf-interfaces of [RFC7223].

   The data model allows to set values to the parameters of the MPL
   algorithm. This approach requires an active manager process to set
   the values without use of DHCP as described in: [RFC7774].

   The names of the four modules are: yang-mpl-domain, yang-mpl-ops,
   yang-mpl-seeds, and yang-mpl-statistics, described in subsections
   with the same name.

3.1. yang-mpl-domain module

   This modules describes (1) the MPL domains and the associated
   multicast addresses, and (2) the interfaces and the multicast
   addresses for which they are enabled.

   The model features a choice such that for constrained devices with
   only one "single" interface and only one "single" domain, the model
   specifies a list of MC addresses for which the single interface is
   enabled.

<CODE BEGINS>

module ietf-yang-mpl-domain {
```

```
  yang-version 1;

  namespace
      "urn:ietf:params:xml:ns:yang:ietf-yang-mpl-domain";

  prefix mpl;

import ietf-inet-types{
    prefix inet;
}

  organization
    "IETF ROLL (Routing Over Low power and lossy networks) Working Group";

  contact
      "WG Web:    http://tools.ietf.org/wg/roll/
      WG List:  mailto:roll@ietf.org

      WG Chair: Peter van der Stok
                mailto:consultancy@vanderstok.org

      WG Chair: Ines Robles
                mailto:maria.ines.robles@ericsson.com

      Editor:   Peter van der Stok
                mailto:consultancy@vanderstok.org";

  description
    "This module contains information about the state of the MPL domain.

      Copyright (c) 2016 IETF Trust and the persons identified as
      authors of the code.  All rights reserved.

      Redistribution and use in source and binary forms, with or
      without modification, is permitted pursuant to, and subject
      to the license terms contained in, the Simplified BSD License
      set forth in Section 4.c of the IETF Trust's Legal Provisions
      Relating to IETF Documents
      (http://trustee.ietf.org/license-info).

      This version of this YANG module is part of RFC XXXX; see
      the RFC itself for full legal notices.";

revision "2016-05-18" {
      description "Initial revision.";
      reference
        "I-D:draft-vanderstok-roll-mpl-yang: A YANG model for Multicast
Protocol for Low power and lossy Networks (MPL)";
    }
```

```
    choice single {
          description
          "A choice between single domain/interface and multiple
          domains and interfaces.";
       container mpl-domain {
          description
          "The entries describe the MPL domains, the associated
           Multicast addresses and interfaces.";

          list domains {
           key domainID;
           description
           "The entries describe a given domain identified with domainID and the
associated Multicast addresses.";

          leaf domainID {
            type uint16;
            description
               "Entry uniquely identifies the domain in the
                forwarder.";
          }

          leaf-list MClist{
             type inet:ipv6-address;
             description
               "List of associated IPv6 Addresses.";
          }
        }   // domains list

        list addresses {
          key MCaddress;
          description
            "The entries describe the interfaces enabled with the specified MC
address.";

          leaf MCaddress {
             type inet:ipv6-address;
             description
                 "MC address belonging to a MPL domain.";
           }

          leaf-list interfaces {
              type string;
              description
              "List of names of interfaces enabled for this Multicast address.
Interface name is defined in [RFC6206].";
          }
        }   // addresses list
```

```
        }   // container mpl-domain
        container mpl-single {
            description
```

```
          "For small devices list of MC addresses for single
           interface and domain.";
          leaf-list MCaddresses{
              type inet:ipv6-address;
              description
                 "list of MC addresses belonging to one single domain and
interface.";
          }
      }   // container mpl-simple
    }   // choice simple
} //module ietf-yang-mpl-domain
```


<CODE ENDS>

3.2. yang-mpl-ops module

   This module models the operational aspects of MPL. Per domain MPL
   specifies four parameters I_MAX, I_MIN, K, and TIMER_EXPIRATIONs for
   data and control messages. The value of the MPL intervals are
   expressed in TUNIT. The entry SE_LIFETIME taken over from [RFC7774]
   fixes TUNIT to milliseconds. For very constrained devices with only
   one domain there can be only one instance of mpl-parameter list.

<CODE BEGINS>
```
module ietf-yang-mpl-ops {

  yang-version 1;

  namespace
      "urn:ietf:params:xml:ns:yang:ietf-yang-mpl-ops";

  prefix mpl;

  organization
    "IETF ROLL (Routing over Low power and lossy networks) Working Group";

  contact
      "WG Web:    http://tools.ietf.org/wg/roll/
      WG List:  mailto:roll@ietf.org

      WG Chair: Peter van der Stok
                mailto:consultancy@vanderstok.org

      WG Chair: Ines Robles
                mailto:maria.ines.robles@ericsson.com
```

      Editor: Peter van der Stok
                mailto:consultancy@vanderstok.org";

  description
    "This module contains information about the operation of the MPL protocol.

      This version of this YANG module is part of RFC XXXX; see
      the RFC itself for full legal notices.";

revision "2016-05-18" {
      description "Initial revision.";
      reference
        "I-D:draft-vanderstok-roll-mpl-yang: A YANG model for Multicast
Protocol for Low power and lossy Networks (MPL)";
    }

    container mpl-ops {
      description
        "Parameter settings for each MPL server and for each individual domain
of the server.";

      leaf SE_LIFETIME {
        type uint16;
        description
           "lifetime in milliseconds/(mpl timer units),
            equivalent to SEED_SET_ENTRY_LIFETIME/TUNIT as
           specified in RFC7774.";
      }

      leaf PROACTIVE_FORWARDING {
        type boolean;
        description
          "The boolean value indicates whether the MPL forwarder
           schedules MPL data message transmission after
            receiving them for the first time.";
      }

```
   leaf SEED_SET_ENTRY_LIFETIME {
```

```
   type uint64;
    description
      "The value indicates the minimum lifetime for an entry
       in the Seed set expressed in seconds. Default value
       is 30 minutes.";
}

list mpl-parameter{
key  domainID;
description
   "Each domain has a set of MPL forwarding parameters
    which regulate the forwarding operation.";

leaf  domainID{
type uint16;
description
    "Each domainID must be present in  mpl-parameter list.";
}

leaf DATA_MESSAGE_IMIN{
type uint16;
description
     "The minimum Trickle timer interval, as defined in
      [RFC6206], for MPL Data Message transmissions.";
 }

 leaf DATA_MESSAGE_IMAX{
 type uint16;
 description
      "The maximum Trickle timer interval, as defined in
      [RFC6206], for MPL Data Message transmissions.";
 }

 leaf DATA_MESSAGE_K{
 type uint16;
 default 1;
 description
    "The redundancy constant, as defined in [RFC6206], for
     MPL Data Message transmissions.";
 }

 leaf DATA_MESSAGE_TIMER_EXPIRATIONS{
 type uint16;
 default 3;
 description
    "The number of Trickle timer expirations that occur
     before terminating the Trickle algorithm's
     retransmission of a given MPL Data Message.";
```

```
        }

        leaf CONTROL_MESSAGE_IMIN{
        type uint16;
        description
           "The minimum Trickle timer interval, as defined
            in [RFC6206], for MPL Control Message
            transmissions.";
        }

        leaf CONTROL_MESSAGE_IMAX{
        type uint16;
        description
           "The maximum Trickle timer interval, as defined
            in [RFC6206], for MPL Control Message
             transmissions.";
        }

        leaf CONTROL_MESSAGE_K{
        type uint16;
        default 1;
        description
           "The redundancy constant, as defined in [RFC6206],
             for MPL Control Message transmissions.";
        }

        leaf CONTROL_MESSAGE_TIMER_EXPIRATIONS{
        type uint16;
        default 10;
        description
           "The number of Trickle time expirations that occur
              before terminating the Trickle algorithm
              for MPL Control Message transmissions.";
         }

      }  // list MPL-parameter
    }  // container MPL-ops
} // module ietf-yang-mpl-ops
```

<CODE ENDS>

3.3. yang-mpl-seeds module

   This module specifies the current values of the operation of the MPL
   forwarder. The values are acquired by the client and set by the
   server. The module specifies a set of message buffers, with a buffer

   per seed and domain. In constrained devices there will be only one
   domain, but probably multiple seeds.

   The message buffer contains a set of messages where each message is
   uniquely identified by its sequence number and seed. The associated
   I, c, e, and t values indicate the progress of MPL with respect to
   this message, as specified in [RFC7731]. A forwarder sends and
   receives multiple copies of a message. When a forwarder has sent
   (received) a copy of a message, the forwarder has sent (received)
   that message.

   For forwarders which are seeds, local has value true and seqno is the
   sequence number of the next message to send.

<CODE BEGINS>
module ietf-yang-mpl-seeds {

   yang-version 1;

   namespace
       "urn:ietf:params:xml:ns:yang:ietf-yang-mpl-seeds";

   prefix mpl;

   organization
     "IETF ROLL (Routing over Low power and lossy networks) Working Group";

   contact
      "WG Web: http://tools.ietf.org/wg/roll/
      WG List: mailto:roll@ietf.org

      WG Chair: Peter van der Stok
               mailto:consultancy@vanderstok.org

      WG Chair: Ines Robles
               mailto:maria.ines.robles@ericsson.com

      Editor: Peter van der Stok
               mailto:consultancy@vanderstok.org";

   description
     "This module contains information about the operation of the MPL protocol.

      Copyright (c) 2016 IETF Trust and the persons identified as
      authors of the code. All rights reserved.

      Redistribution and use in source and binary forms, with or

```
revision "2016-05-18" {
     description "Initial revision.";
      reference
        "I-D:draft-vanderstok-roll-mpl-yang: A YANG model for Multicast
Protocol for Low power and lossy Networks (MPL)";
    }


    list mpl-seeds{
    key "seedID domainID";
    config false;
    description
       "List describes all seeds that are active in the server. Seed
information contains the message buffer contents and the operational values of
I, c, sequence number and the life-times per message.";

       leaf seedID{
       type uint64;
       description
           "value uniquely identifies the MPL Seed within a MPL
            domain.";
       }

       leaf domainID{
       type uint16;
       description
          "together with seedID uniquely identifies buffer set.";
       }

       leaf local {
       type boolean;
       description
          "When local == TRUE, seed is located in this forwarder.
           WHEN local == false, seed is located in different
           forwarder.";
       }

    leaf generate-seqno {
    type uint8;
    description
```

```
        "Sequence number of next message to be generated by this local
seed.";
      }
```

```
      leaf life-time {
      type uint64;
      description
          " Minimum remaining lifetime of the seed entry in
           SE_LIFETIME units.";
      }

      leaf min-seqno{
      type uint8;
      description
         "Lower bound sequence number in the buffer of the seed.";
      }

      leaf data-number{
      type uint8;
      description
          "Number of currently buffered data messages.";
      }

      leaf control-number{
      type uint8;
      description
          "Number of currently buffered control messages.";
      }

       list buffered-messages{
       key seqno;
       description
          " status of trickle intervals of the buffered message identified by
seqno. and seed/domain";

       leaf seqno{
       type uint8;
       description
          "Sequence number of message.";
      }

       leaf I{
       type uint8;
       description
          "Current Trickle timer interval size in SE-LIFETIME units.";
      }

       leaf c{
       type uint8;
       description
          "number of times that copy of this message has been
           received in this interval.";
```

```
      }
```

```
    leaf e{
    type uint8;
    description
            "number of Trickle time expirations since last
             Trickle timer reset.";
    }

    leaf t{
    type uint8;
    description
            " Time expressed in SE-LIFETIME units
             that message will be (is) forwarded";
    }
    } // list seed-timers
  } // list MPL-seeds
} // module ietf-yang-mpl-seeds
```

<CODE ENDS>

3.4. yang-mpl-statistics module

This module specifies the operation of the MPL forwarder expressed in
number of messages and copies. The values are acquired by the client
and set by the server. Statistics are specified per seed and domain.
In constrained devices there will be only one domain, but probably
multiple seeds.

The parameter k determines how many copies of a message can be
forwarded. The counters c-too-high, nr-forwarded, and nr-not-
forwarded give insight in the consequences of the current value of k.

The other counters give insight in the loss of messages caused by the
medium or forwarding delays. The inconsistent/consistent counters
indicate when consistent or inconsistent messages were received
according to the definition of consistent in [RFC7731].

<CODE BEGINS>

```
module ietf-yang-mpl-statistics {

  yang-version 1;

  namespace
      "urn:ietf:params:xml:ns:yang:ietf-yang-mpl-statistics";
```

   prefix mpl;

   organization
     "IETF ROLL (Routing over Low power and lossy networks) Working Group";

   contact
      "WG Web: http://tools.ietf.org/wg/roll/
      WG List: mailto:roll@ietf.org

      WG Chair: Peter van der Stok
                mailto:consultancy@vanderstok.org

      WG Chair: Ines Robles
                mailto:maria.ines.robles@ericsson.com

      Editor: Peter van der Stok
                mailto:consultancy@vanderstok.org";

   description
     "This module contains information about the operation of the MPL protocol.

      Copyright (c) 2016 IETF Trust and the persons identified as
      authors of the code. All rights reserved.

      Redistribution and use in source and binary forms, with or

      without modification, is permitted pursuant to, and subject
      to the license terms contained in, the Simplified BSD License
      set forth in Section 4.c of the IETF Trust's Legal Provisions
      Relating to IETF Documents
      (http://trustee.ietf.org/license-info).

      This version of this YANG module is part of RFC XXXX; see
      the RFC itself for full legal notices.";

revision "2016-05-18" {
      description "Initial revision.";
      reference
        "I-D:draft-vanderstok-roll-mpl-yang: A YANG model for Multicast
Protocol for Low power and lossy Networks (MPL)";
    }

     list mpl-statistics{
     key "seedID domainID";

     config false;

```
     description
        "List describes performance statistics integrated over the messages
identified by seed and domain identifiers. A forwarder can receive and forward
multiple copies of a message uniquely identified by seqno, domain, and seed.";

     leaf seedID{
     type uint64;
     description
         "value uniquely identifies the MPL Seed within a MPL
          domain.";
     }

     leaf domainID{
     type uint16;
     description
         "together with seed-ID uniquely identifies buffer set.";
     }

     leaf c-too-high {
     type uint64;
     description
         "Number of times that a copy was not forwarded
             because c > k.";
     }

     leaf nr-forwarded {
     type uint64;
     description
         "number of times copies are forwarded, while c <= k.";
     }

     leaf nr-of-messages-received{
     type uint64;
     description
         "number of messages (one or more copies) received, must be smaller
than or equal to seqno.";
     }

     leaf nr-of-copies-received{
     type uint64;
     description
         "total number of message copies received.";
     }

     leaf nr-of-messages-forwarded{
     type uint64;
     description
         "number of forwarded messages, must be smaller than or equal to nr-
```

```
of-messages-received.";
    }

    leaf nr-of-copies-forwarded{
```

```
     type uint64;
     description
         "number of forwarded copies, can be larger than number-of-copies-
received.";
     }

     leaf nr-of-refused{
     type uint64;
     description
         "number of refused copies because seqno too small.";
     }

     leaf nr-of-missed{
     type uint64;
     description
         "number of messages that were not received (derived from gaps in
received seqno's.)";
     }

     leaf nr-of-notreceived{
     type uint64;
     description
          "number of messages that were not received
           according to control message.";
     }

     leaf nr-of-inconsistent-data{
     type uint64;
     description
         "number of inconsistent data messages.";
     }

     leaf nr-of-consistent-data{
     type uint64;
     description
         "number of consistent data messages.";
     }

     leaf nr-of-consistent-control{
     type uint64;
     description
         "number of consistent control messages.";
     }

     leaf nr-of-inconsistent-control{
     type uint64;
     description
         "number of inconsistent control messages.";
```

```
      }
    }  // list mpl statistics
```

```
} // module  ietf-yang-mpl-statistics
```

<CODE ENDS>

4. IANA Considerations

   This specification has no consequences for IANA.

5. Acknowledgements

   Andy Bierman has commented on the use of YANG for mpl. YANG doctors
   pointed out a wrong use of config.

6. Changelog

   Changes from version 00 to version 01

   o config false in "statistics" and "seeds" modules

   o separated into 4 modules

   o inserted choice in domain modules

   o renamed some parameters

   o Introduced section per module

7. References

7.1. Normative References

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

   [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for
              the Network Configuration Protocol (NETCONF)", RFC 6020,
              DOI 10.17487/RFC6020, October 2010,
              <http://www.rfc-editor.org/info/rfc6020>.

   [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed.,
              and A. Bierman, Ed., "Network Configuration Protocol
              (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011,
              <http://www.rfc-editor.org/info/rfc6241>.

   [RFC7223] Bjorklund, M., "A YANG Data Model for Interface
              Management", RFC 7223, DOI 10.17487/RFC7223, May 2014,
              <http://www.rfc-editor.org/info/rfc7223>.

   [RFC7731] Hui, J. and R. Kelsey, "Multicast Protocol for Low-Power
              and Lossy Networks (MPL)", RFC 7731, DOI 10.17487/RFC7731,
              February 2016, <http://www.rfc-editor.org/info/rfc7731>.

7.2. Informative References

   [RFC6206] Levis, P., Clausen, T., Hui, J., Gnawali, O., and J. Ko,
              "The Trickle Algorithm", RFC 6206, DOI 10.17487/RFC6206,
              March 2011, <http://www.rfc-editor.org/info/rfc6206>.

   [RFC7774] Doi, Y. and M. Gillmore, "Multicast Protocol for Low-Power
              and Lossy Networks (MPL) Parameter Configuration Option
              for DHCPv6", RFC 7774, DOI 10.17487/RFC7774, March 2016,
              <http://www.rfc-editor.org/info/rfc7774>.

Author's Address

   Peter van der Stok (editor)
   consultant

   Phone: +31-492474673 (Netherlands), +33-966015248 (France)
   Email: consultancy@vanderstok.org
   URI: www.vanderstok.org