

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 25, 2015

Y. Liu
F. Guo
Huawei
October 22, 2014

YANG Data Model for PIM
draft-liu-pim-yang-00

Abstract

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

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 April 25, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	Terminology	3
3.	Design of Data Model	3
3.1.	Overview	3
3.2.	PIM BSR Configuration	4
3.2.1.	Per-instance C-BSR Configuration of PIM BSR Instance	5
3.2.2.	Per-instance C-RP Configuration of PIM BSR Instance .	5
3.3.	PIM SM Configuration	6
3.3.1.	Per-instance Parameters	6
3.3.2.	Per-static-RP configuration of the PIM SM instance .	7
3.3.3.	Per-SPT-switch configuration of the PIM SM instance .	8
3.3.4.	Per-Anycast-RP configuration of the PIM SM instance .	8
3.3.5.	Per-interface configuration of the PIM SM instance .	9
4.	PIM Yang Module	10
5.	IANA Considerations	30
6.	Security Considerations	30
7.	Acknowledgements	30
8.	References	31
8.1.	Normative References	31
8.2.	Informative References	31
	Authors' Addresses	32

[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 PIM. It includes PIM SM[RFC4601], PIM SSM[RFC4607][[RFC4608](#)], PIM BSR[RFC5059]. In addition, It can extend PIM DM[RFC3973], BIDIR PIM[RFC5015] etc. features described in PIM standards other than mentioned above RFC in future version.

2. Terminology

- o PIM: Protocol Independent Multicast
- o SM: Sparse Mode
- o SSM: Source-specific Multicast
- o DM: Dense Mode
- o BSR: Bootstrap Router
- o RP: Rendezvous Point
- o SPT: shortest-path tree
- o RPT: Rendezvous Point Tree

3. Design of Data Model

3.1. Overview

Because PIM SSM is a subset of PIM SM, it is not necessary to assign a separate container for PIM SSM. The PIM Yang module is divided into two main containers :

- o PIM_BSR : that contains all pim bsr&rp writable configuration objects.
- o PIM_SM : that contains all pim sm writable configuration objects.

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


```

module: pim
  +--rw PIM
    +--rw PIM_BSR
      | +--rw PIM_BSR-Instances
      |   +--rw C-BSR-Instance* [vrfName]
      |   | ...
      |   +--rw C-RP-Instance* [vrfName]
      |   | ...
    +--rw PIM_SM
      +--rw PIM_SM-Instances
        +--rw PIM_SM-Instance* [vrfName]
          | ...
          +--rw PIM-Static-RPs
            | +--rw PIM-Static-RP
            |   | ...
          +--rw PIM-SPT-Switchs
            | +--rw PIM-SPT-Switch* [sptGrpPlyName]
            |   | ...
          +--rw PIM_SM-Anycast-RPs
            | +--rw PIM_SM-Anycast-RP
            |   | ...
            |   +--rw PIM_SM-RP-Peers
            |       +--rw PIM_SM-RP-Peer
            |       | ...
          +--rw PIM_SM-Interfaces
            +--rw PIM_SM-Interface* [ifName]
            | ...

```

Figure 1 The overview of PIM YANG data model

3.2. PIM BSR Configuration

PIM BSR configuration container has only one main container:

- o PIM_BSR-instances : that contains all pim c-bsr and c-rp writable configuration objects.

PIM BSR per-instance configuration container includes c-bsr and c-rp parameters of the public PIM instance or the PIM instance binding a specific VRF. PIM BSR per-instance configuration container is divided into:

- o Per-instance C-BSR configuration of the PIM BSR instance.
- o Per-instance C-RP configuration of the PIM BSR instance.

3.2.1. Per-instance C-BSR Configuration of PIM BSR Instance

The per-instance C-BSR configuration includes the name of the VRF bound by the pim bsr instance, and timer parameters , policies etc. it also includes administratively scoped BSR configuration.

```

+--rw C-BSR-Instance* [vrfName]
  |   | +--rw vrfName          string
  |   | +--rw addressFamily    enumeration
  |   | +--rw cBsrIfName       ifName
  |   | +--rw IPV4-cBsrIfAddr? inet:ipv4-address
  |   | +--rw IPV6-cBsrIfAddr  inet:ipv6-address
  |   | +--rw cBsrHoldTime?    uint32
  |   | +--rw cBsrInterval?    uint32
  |   | +--rw cBsrHashLen?     uint32
  |   | +--rw cBsrPriority?     uint32
  |   | +--rw cBsrPlyName?     string
  |   | +--rw cBsrPlyIpv6?     string
  |   | +--rw cBsrAdminScope?  boolean
  |   | +--rw cBsrGlobalEnable? boolean
  |   | +--rw cBsrGlobalHashLength? uint32
  |   | +--rw cBsrGlobalPriority? uint32
  |   | +--rw isFragable       boolean

```

Figure 2 The YANG data model of PIM C-BSR

3.2.2. Per-instance C-RP Configuration of PIM BSR Instance

The per-instance C-RP configuration includes the name of the VRF bound by the pim bsr instance, specific interface or address for c-rp and timer parameters , policies etc.

```

+--rw C-RP-Instance* [vrfName]
  |   +--rw vrfName          string
  |   +--rw cRpIfName        ifName
  |   +--rw cRpGrpPlyName?   string
  |   +--rw cRpPriority?      uint32
  |   +--rw cRpHoldTime?     uint32
  |   +--rw cRpAdvInterval?   uint32
  |   +--rw IPV4-cRpAddr?     inet:ipv4-address
  |   +--rw IPV6-cRpAddr      inet:ipv6-address

```

Figure 3 The YANG data model of PIM C-RP

3.3. PIM SM Configuration

PIM SM configuration container has only one main container:

- o PIM_SM-instances : that contains all pim sm writable configuration objects.

PIM SM per-instance configuration container includes pim sm protocol parameters of the public PIM instance or the PIM instance binding a specific VRF. PIM SM per-instance configuration container is divided into:

- o Per-instance parameters
- o Per-static-RP configuration of the PIM SM instance
- o Per-SPT-switch configuration of the PIM SM instance
- o Per-Anycast-RP configuration of the PIM SM instance
- o Per-interface configuration of the PIM SM instance

3.3.1. Per-instance Parameters

The per-instance parameter includes the name of the VRF bound by the PIM SM instance, and timer parameters such as hello interval etc., and varied policies etc


```

+--rw PIM_SM-Instances
  +--rw PIM_SM-Instance* [vrfName]
    +--rw vrfName                string
    +--rw addressFamily          enumeration
    +--rw assertHoldTime?       uint32
    +--rw jpHoldTime?           uint32
    +--rw probeInterval?        uint32
    +--rw jpTimerInterval?      uint32
    +--rw drPriority?            uint32
    +--rw helloHoldtime?        uint16
    +--rw helloLandelay?        uint16
    +--rw helloInterval?        uint32
    +--rw helloOverride?        uint16
    +--rw regChecksum?          uint16
    +--rw regSuppInterval?      uint32
    +--rw embeddedRp?           boolean
    +--rw nbrCheckRecv?         boolean
    +--rw nbrCheckSend?         boolean
    +--rw regPlyName?           string
    +--rw regPlyIpv6?           string
    +--rw ssmPlyName?           string
    +--rw ssmPlyIpv6?           string
    +--rw srcPlyName?           string
    +--rw srcPlyIpv6?           string
    +--rw bsrPlyName?           string
    +--rw bsrPlyIpv6?           string
    +--rw embRpPlyName?         string
    +--rw sourceLifeTime?       uint32
    +--rw sptDetInterval?       uint32
    +--rw ipsecName?            string
    +--rw ipsecType?            enumeration
    +--rw uniIpsecName?         string

```

Figure 4 The YANG data model of PIM SM instance

3.3.2. Per-static-RP configuration of the PIM SM instance

Per-static-RP configuration of the PIM SM instance includes static RP address, preference and policy for group range. For simply deployment, sometimes it is not necessary to deploy dynamic BSR/RP mechanism, and static RP mechanism is also satisfied.


```

+--rw PIM-Static-RPs
|   +--rw PIM-Static-RP
|       +--rw IPV4-staticRpAddr      inet:ipv4-address
|       +--rw IPV6-staticRpAddr      inet:ipv6-address
|       +--rw staticRpPlyName?       string
|       +--rw staticRpPlyNameIpv6?   string
|       +--rw preference?            enumeration

```

Figure 5 The YANG data mode of PIM static-RP

3.3.3. Per-SPT-switch configuration of the PIM SM instance

Per-SPT-switch configuration of the PIM SM instance includes spt switch threshold, group range by policy. PIM SM SPT switch process as described in [\[RFC4601\]](#) makes multicast flows from RPT to SPT in order to optimizing the forwarding paths.

```

+--rw PIM-SPT-Switchs
|   +--rw PIM-SPT-Switch* [sptGrpPlyName]
|       +--rw infinity?      boolean
|       +--rw sptThreshHold? uint32
|       +--rw isACLEnable    boolean
|       +--rw sptGrpPlyName   string
|       +--rw sptGrpPlcyOrder? uint32

```

Figure 6 The YANG data model of PIM SPT Switch

3.3.4. Per-Anycast-RP configuration of the PIM SM instance

Anycast-RP as described in [\[RFC4610\]](#) is a mechanism that pim sm register packets have been used to exchange multicast source information and get fast convergence when a PIM Rendezvous Point (RP) router fails. Per-Anycast-RP configuration container is divided into :

- o Per-Anycast-RP parameters
- o Per-RP-peer configuration of PIM SM Anycast RP

3.3.4.1. Per-Anycast-RP Parameters

Per-Anycast-RP parameter includes the RP for anycast address, and the local address to establish RP peer link.


```

+--rw PIM_SM-Anycast-RPs
|   +--rw PIM_SM-Anycast-RP
|       +--rw IPV4-rpAddress      inet:ipv4-address
|       +--rw IPV6-rpAddress      inet:ipv6-address
|       +--rw local-IPV4-Address  inet:ipv4-address
|       +--rw local-IPV6-Address  inet:ipv6-address

```

Figure 7 The YANG data model of PIM SM Anycast-RP

3.3.4.2. Per-RP-peer Configuration of PIM SM Anycast RP

Per-RP-peer configuration of PIM SM anycast RP includes the peer address, source information forwarding policy etc.

```

+--rw PIM_SM-RP-Peers
|   +--rw PIM_SM-RP-Peer
|       +--rw IPV4-rpPeer-Address  inet:ipv4-address
|       +--rw IPV6-rpPeer-Address  inet:ipv6-address
|       +--rw fwdSaSwt?             boolean
|       +--rw fwdPolicy?            string
|       +--rw fwdPolicyIpv6?        string

```

Figure 8 The YANG data model of PIM SM Anycast-RP peer

3.3.5. Per-interface configuration of the PIM SM instance

Per-interface configuration of the PIM SM instance includes the interface name, and the VRF name bound by the interface, and time parameters, policies etc.


```

+--rw PIM_SM-Interfaces
  +--rw PIM_SM-Interface* [ifName]
    +--rw vrfName          string
    +--rw addressFamily    enumeration
    +--rw ifName           ifName
    +--rw pimsmEnable       boolean
    +--rw drPriority?       uint32
    +--rw helloInterval?   uint32
    +--rw helloHoldtime?   uint16
    +--rw helloOverride?   uint16
    +--rw helloLanDelay?   uint16
    +--rw jpTimerInterval? uint32
    +--rw jpHoldtime?      uint32
    +--rw jpPlyName?       string
    +--rw jpPlyIpv6?       string
    +--rw jpAsmPlyName?    string
    +--rw jpAsmPlyIpv6?    string
    +--rw jpSsmPlyName?    string
    +--rw jpSsmPlyIpv6?    string
    +--rw nbrPlyName?      string
    +--rw nbrPlyIpv6?      string
    +--rw assertHoldtime?  uint32
    +--rw requireGenId?    boolean
    +--rw pimBsrBoundary?  enumeration
    +--rw bfdEnable?       boolean
    +--rw bfdMinTx?        uint32
    +--rw bfdMinRx?        uint32
    +--rw bfdMultiplier?  uint16
    +--rw isSilent?         boolean
    +--rw isDrSwtDelay?     boolean
    +--rw drSwtDelayInterval? uint32
    +--rw ipsecName?       string
    +--rw ipsecType?       enumeration

```

Figure 9 The YANG data model of PIM SM interface

4. PIM Yang Module

```

module pim {
  namespace "urn:huawei:params:xml:ns:yang:pim";
  // replace with IANA namespace when assigned - urn:ietf:params:xml:ns:yang:
1  prefix "pim";
  import ietf-inet-types {
    prefix inet;
  }
  organization
    "Huawei Technologies Co., Ltd.";

```

contact

Liu & Guo

Expires April 25, 2015

[Page 10]

```
"liuyisong@huawei.com
  guofeng@huawei.com";
description
  "This YANG module defines the generic configuration
  data for PIM, 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
  PIM configuration parameters.";
  revision 2014-10-21 {
    description
      "Initial revision.";
  }

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

container PIM {
  container PIM_BSR {

    container PIM_BSR-Instances {

      list C-BSR-Instance {

        key "vrfName";
        max-elements "unbounded";
        min-elements "0";
        description "Specifies an PIM BSR instance.";

        leaf vrfName {
          description "Name of a VPN instance.";
          config "true";
          //default "_public_";
          type string {
            length "0..32";
          }
        }
        leaf addressFamily {
          config "true";
          mandatory "true";
          type enumeration {
            enum ipv4unicast {
              value "0";
              description "Address family, which determines
whether an address belongs to IPv4 or IPv6.";
```

}

```
        enum ipv6unicast {
            value "1";
            description "Address family, which determines
whether an address belongs to IPv4 or IPv6.";
        }
    }
    leaf cBsrIfName {
        description "Interface name.";
        config "true";
        mandatory "true";
        type ifName;
    }

    leaf IPV4-cBsrIfAddr {
        description "Global IPv4 unicast address of the C-
BSR.";
        config "true";
        type inet:ipv4-address;
    }
    leaf IPV6-cBsrIfAddr {
        description "Global IPv6 unicast address of the C-
BSR.";
        config "true";
        mandatory "true";
        type inet:ipv6-address;
    }
    leaf cBsrHoldTime {
        description "Timeout period (called holdtime) during
which C-BSRs wait to receive bootstrap messages from the BSR. The value is an
integer ranging from 1 to 214748364, in seconds. The default value is 130. To
prevent frequent BSR elections, set the same holdtime for all C-BSRs in the
same domain. If an interval (called BS_interval) at which bootstrap messages
are sent has been set for C-BSRs, ensure that the specified holdtime is larger
than the BS_interval.";
        config "true";
        default "130";
        type uint32 {
            range "1..214748364";
        }
    }
    leaf cBsrInterval {
        description "Interval (called BS_interval) at which a
BSR continuously sends bootstrap messages. The value is an integer ranging from
1 to 107374177, in seconds. The default value is 60. To prevent frequent BSR
elections, set the same BS_interval for all C-BSRs in the same domain. If a
timeout period (called holdtime) during which C-BSRs wait to receive bootstrap
messages from the BSR has been set for C-BSRs, ensure that the specified
```

```

BS_interval is smaller than the holdtime.";
    config "true";
    default "60";
    type uint32 {
        range "1..107374177";
    }
}
leaf cBsrHashLen {
    description "Global hash mask length for a C-BSR. The
value is an integer ranging from 0 to 32. The default value is 30. In IPv4, the
value is an integer ranging from 0 to 32, and the default value is 30. In IPv6,
the value is an integer ranging from 0 to 128, and the default value is 126.";
    config "true";
    default "126";
    type uint32 {
        range "0..128";
    }
}

```

```
    leaf cBsrPriority {
      description "Globally specify a priority for all C-BSRs
on the router. The greater the value, the higher the priority. The value is an
integer ranging from 0 to 255. The default value is 0. ";
      config "true";
      default "0";
      type uint32 {
        range "0..255";
      }
    }
    leaf cBsrPlyName {
      description "Policy for limiting the range of valid BSR
addresses so that a router discards the messages received from the BSRs not in
the specified address range. The value is an integer ranging from 2000 to 2999,
or a string of 1 to 32 case-sensitive characters. By default, the range of
valid BSR addresses is not limited.";
      config "true";
      type string {
        length "1..255";
        pattern "^[^ ]+$";
      }
    }
    leaf cBsrPlyIpv6 {
      description "Policy for limiting the range of valid BSR
addresses so that a router discards the messages received from the BSRs not in
the specified address range. The value is an integer ranging from 2000 to 2999,
or a string of 1 to 32 case-sensitive characters. By default, the range of
valid BSR addresses is not limited.";
      config "true";
      type string {
        length "1..255";
        pattern "^[^ ]+$";
      }
    }
    leaf cBsrAdminScope {
      description "One PIM-SM domain is divided into multiple
BSR administrative domains to implement RP-Set advertisement. By default, there
is only one BSR in the entire PIM-SM domain.";
      config "true";
      type boolean
      {
      }
    }
    leaf cBsrGlobalEnable {
      description "The router is a C-BSR in the global
domain. By default, no C-BSR is configured in the global domain.";
      config "true";
      type boolean
```



```

    {
    }
}
leaf cBsrGlobalHashLength {
    description "Hash mask length for the C-BSR in the
global domain. The value is an integer ranging from 0 to 32. The default value
is 30.";
    config "true";
    default "30";
    type uint32 {
        range "0..32";
    }
}
leaf cBsrGlobalPriority {
    description "Priority for the C-BSR in the global
domain. The greater the value, the higher the priority. The value is an integer
ranging from 0 to 255. The default value is 0.";

```

```
        config "true";
        default "0";
        type uint32 {
            range "0..255";
        }
    }
    leaf isFragable {
        description "Enable the C-BSR message fragmentation
function. By default, this function is disabled.";
        config "true";
        mandatory "true";
        type boolean
        {
        }
    }
}

list C-RP-Instance {

    key "vrfName";
    max-elements "unbounded";
    min-elements "0";
    description "Specifies a C-RP instance.";

    leaf vrfName {
        description "Name of a VPN name.";
        config "true";
        //default "_public_";
        type string {
            length "0..32";
        }
    }
    leaf cRpIfName {
        description "Interface name.";
        config "true";
        mandatory "true";
        type ifName;
    }
    leaf cRpGrpPlyName {
        description "Policy for limiting the range of valid
group addresses. With this policy, a router discards messages received from the
addresses not in the specified range. The value is an integer ranging from 2000
to 2999, or a string of 1 to 32 case-sensitive characters.";
        config "true";
        type string {
            length "1..255";
            pattern "^[^ ]+$";
        }
    }
}
```

```
    }  
  }  
  leaf cRpPriority {  
    description "Priority of a C-RP. The greater the value,  
the lower the priority. The value is an integer ranging from 0 to 255. The  
default value is 0.";
```

```
        config "true";
        default "0";
        type uint32 {
            range "0..255";
        }
    }
    leaf cRpHoldTime {
        description "Timeout period during which a BSR waits to
receive Advertisement messages from a C-RP. The value is an integer ranging
from 1 to 65535, in seconds. The default value is 150.";
        config "true";
        default "150";
        type uint32 {
            range "1..65535";
        }
    }
    leaf cRpAdvInterval {
        description "Interval at which a C-RP sends
Advertisement messages. The value is an integer ranging from 1 to 65535, in
seconds. The default value is 60.";
        config "true";
        default "60";
        type uint32 {
            range "1..65535";
        }
    }
    leaf IPV4-cRpAddr {
        description "Specifies a C-RP address.";
        config "true";
        type inet:ipv4-address;
    }
    leaf IPV6-cRpAddr {
        description "Specifies a C-RP address.";
        config "true";
        mandatory "true";
        type inet:ipv6-address;
    }
}

}

}

container PIM_SM {
    container PIM_SM-Instances {
        list PIM_SM-Instance {

            key "vrfName";
            max-elements "unbounded";
```

```
min-elements "0";  
description "Specifies an PIM-SM instance."  
  
leaf vrfName {
```

```
        description "Name of a VPN instance.";
        config "true";
        //default "_public_";
        type string {
            length "0..32";
        }
    }

    leaf addressFamily {
        config "true";
        mandatory "true";
        type enumeration {
            enum ipv4unicast {
                value "0";
                description "Address family, which determines
whether an address belongs to IPv4 or IPv6.";
            }
            enum ipv6unicast {
                value "1";
                description "Address family, which determines
whether an address belongs to IPv4 or IPv6.";
            }
        }
    }

    leaf assertHoldTime {
        description "Timeout period during which PIM interfaces
wait to receive Assert messages from the forwarder. The value is an integer
ranging from 7 to 65535, in seconds. The default value is 180. ";
        config "true";
        default "180";
        type uint32 {
            range "7..65535";
        }
    }

    leaf jpHoldTime {
        description "Holdtime for a Join/Prune message sent by
a PIM interface. The value is an integer ranging from 1 to 65535, in seconds.
The default value is 210. Commonly, the holdtime is 3.5 times longer than the
interval for all the interfaces to send Join/Prune messages.";
        config "true";
        default "210";
        type uint32 {
            range "1..65535";
        }
    }

    leaf probeInterval {
        description "Interval at which Probe messages are sent
to an RP. The value is an integer ranging from 1 to 1799, in seconds. The
```

```
default value is 5.";
    config "true";
    default "5";
    type uint32 {
        range "1..1799";
    }
}
leaf jpTimerInterval {
    description "Interval at which Join/Prune messages are
sent. The value is an integer ranging from 1 to 2147483647, in seconds. It must
be shorter than the holdtime of Join/Prune messages. The default value is 60.
";
```

```
        config "true";
        default "60";
        type uint32 {
            range "1..18000";
        }
    }
    leaf drPriority {
        description "DR election priority for a router. The
greater the value, the higher the priority. The value is an integer ranging
from 0 to 4294967295. The default value is 1.";
        config "true";
        default "1";
        type uint32 {
            range "0..4294967295";
        }
    }
    leaf helloHolddtime {
        description "Timeout period during which a PIM
interface waits to receive Hello messages from its neighbors. The value is an
integer that ranging from 1 to 65535, in seconds. It must be longer than the
interval for PIM neighbors to send Hello messages. The default value is 105. ";
        config "true";
        default "105";
        type uint16 {
            range "1..65535";
        }
    }
    leaf helloLdelay {
        description "Delay for transmitting Prune messages on a
shared network segment. The value is an integer ranging from 1 to 32767, in
milliseconds. The default value is 500.";
        config "true";
        default "500";
        type uint16 {
            range "1..32767";
        }
    }
    leaf helloInterval {
        description "Specifies the interval at which Hello
messages are sent. The value is an integer ranging from 1 to 2147483647, in
seconds. It must be shorter than the timeout period of PIM neighbors. The
default value is 30.";
        config "true";
        default "30";
        type uint32 {
            range "1..18000";
        }
    }
}
```



```
leaf helloOverride {
    description "Interval at which the prune action in a
Hello message is overridden. The value is an integer ranging from 1 to 65535,
in milliseconds. The default value is 2500.";
    config "true";
    default "2500";
    type uint16 {
        range "1..65535";
    }
}
leaf regChecksum {
    description "Configure a router to calculate the
checksum based on all contents of a Register message. By default, the checksum
is calculated based on the header of a Register message only.";
```

```
        config "true";
        type uint16 {
            range "1..65535";
        }
    }
    leaf regSuppInterval {
        description "Timeout period during which a router
remains in the registration suppression state. The value is an integer ranging
from 11 to 3600, in seconds. The default value is 60.";
        config "true";
        default "60";
        type uint32 {
            range "11..3600";
        }
    }
    leaf embeddedRp {
        description "Enable the embedded RP function. By
default, the embedded RP function is enabled.";
        config "true";
        default "true";
        type boolean {
        }
    }
    leaf nbrCheckRecv {
        description "Enable the PIM neighbor check function to
check whether received Join/Prune and Assert messages are sent from a PIM
neighbor. If not, these messages are discarded. By default, the PIM neighbor
check function is disabled.";
        config "true";
        type boolean {
        }
    }
    leaf nbrCheckSend {
        description "Enable the PIM neighbor check function to
check whether Join/Prune and Assert messages are to be sent to an IPv4/v6 PIM
neighbor. If not, these messages are not sent. By default, the PIM neighbor
check function is disabled for Join/Prune and Assert messages to be sent.";
        config "true";
        type boolean {
        }
    }
    leaf regPlyName {
        description "Policy for filtering Register messages.
The value is an integer ranging from 3000 to 3999, or a string of 1 to 32 case-
sensitive characters.";
        config "true";
        type string {
            length "1..255";
        }
    }
}
```

```

        pattern "^[^ ]+$";
    }
}
leaf regPlyIpv6 {
    description "Policy for filtering Register messages.
The value is an integer ranging from 3000 to 3999, or a string of 1 to 32 case-
sensitive characters.";
    config "true";
    type string {
        length "1..255";
        pattern "^[^ ]+$";
    }
}

```

```
    leaf ssmPlyName {
      description "Policy for limiting the range of valid SSM
group addresses. The value is an integer ranging from 2000 to 2999, or a string
of 1 to 32 case-sensitive characters.";
      config "true";
      type string {
        length "1..255";
        pattern "^[^ ]+$";
      }
    }
    leaf ssmPlyIpv6 {
      description "Range of SSM group addresses. The value is
an integer ranging from 2000 to 2999, or a string of 1 to 32 case-sensitive
characters.";
      config "true";
      type string {
        length "1..255";
        pattern "^[^ ]+$";
      }
    }
    leaf srcPlyName {
      description "Policy for filtering multicast entries
based on source addresses or based on both source and group addresses. The
value is an integer ranging from 2000 to 3999, or a string of 1 to 32 case-
sensitive characters.";
      config "true";
      type string {
        length "1..255";
        pattern "^[^ ]+$";
      }
    }
    leaf srcPlyIpv6 {
      description "Policy for filtering multicast entries
based on source addresses or based on both source and group addresses. The
value is an integer ranging from 2000 to 3999, or a string of 1 to 32 case-
sensitive characters.";
      config "true";
      type string {
        length "1..255";
        pattern "^[^ ]+$";
      }
    }
    leaf bsrPlyName {
      description "Policy for limiting the range of valid BSR
addresses so that a router discards the messages received from the BSRs not in
the specified address range. The value is an integer ranging from 2000 to 2999,
or a string of 1 to 32 case-sensitive characters. By default, the range of
valid BSR addresses is not limited.";
```

```

        config "true";
        type string {
            length "1..255";
            pattern "^[^ ]+$";
        }
    }
    leaf bsrPlyIpv6 {
        description "Policy for limiting the range of valid BSR
addresses so that a router discards the messages received from the BSRs not in
the specified address range. The value is an integer ranging from 2000 to 2999,
or a string of 1 to 32 case-sensitive characters. By default, the range of
valid BSR addresses is not limited.";
        config "true";
        type string {
            length "1..255";
            pattern "^[^ ]+$";
        }
    }
}

```

```
    leaf embRpPlyName {
      description "Policy for limiting the range of multicast
groups to which an embedded-RP applies. The value is an integer ranging from
2000 to 2999, or a string of 1 to 32 case-sensitive characters.";
      config "true";
      type string {
        length "1..255";
        pattern "^[^ ]+$";
      }
    }
    leaf sourceLifeTime {
      description "Timeout period for (S, G) entries on a
router. The value is an integer ranging from 60 to 65535, in seconds. The
default value is 210.";
      config "true";
      default "210";
      type uint32 {
        range "60..65535";
      }
    }
    leaf sptDetInterval {
      description "Interval for checking the multicast
message rate is configured. When the multicast message rate reaches the
specified threshold, a switchover from the RPT to the SPT is performed. The
value is an integer ranging from 15 to 65535, in seconds.";
      config "true";
      default "15";
      type uint32 {
        range "15..65535";
      }
    }
    leaf ipsecName {
      description "Description of an SA. The value is a
string of 1 to 15 characters.";
      config "true";
      type string {
        length "1..15";
        pattern "^[^ ]+$";
      }
    }
    leaf ipsecType {
      description "Hello options based on the IPsec type.";
      config "true";
      type enumeration {
        enum forAll {
          value "0";
          description "Hello options based on the IPsec
type.";
```

```

    }
    enum forHello {
        value "1";
        description "Hello options based on the IPsec
type.";
    }
}
leaf uniIpsecName {
    description "Description of an SA. The value is a
string of 1 to 15 characters.";

```

```
    config "true";
    type string {
        length "1..15";
        pattern "^[^ ]+$";
    }
}
container PIM-Static-RPs {

    container PIM-Static-RP {

        leaf IPV4-staticRpAddr {
            description "Specifies a static RP address.";
            config "true";
            mandatory "true";
            type inet:ipv4-address;
        }
        leaf IPV6-staticRpAddr {
            description "Specifies a static RP address.";
            config "true";
            mandatory "true";
            type inet:ipv6-address;
        }
        leaf staticRpPlyName {
            description "Static RP policy. The value is an
integer ranging from 2000 to 2999, or a string of 1 to 32 case-sensitive
characters.";
            config "true";
            type string {
                length "1..32";
                pattern "^[^ ]+$";
            }
        }
        leaf staticRpPlyNameIpv6 {
            description "Static RP policy. The value is an
integer ranging from 2000 to 2999, or a string of 1 to 32 case-sensitive
characters.";
            config "true";
            type string {
                length "1..32";
                pattern "^[^ ]+$";
            }
        }
        leaf preference {
            description "Whether the static RP is
preferred. The value can be: 0: the static RP is not preferred. 1: the static
RP is preferred. The default value is 0.";
            config "true";
            default "NotPrefer";
        }
    }
}
```



```
type enumeration {
    enum NotPrefer {
        value "0";
        description "Whether the static RP is
preferred. The value can be: 0: the static RP is not preferred. 1: the static
RP is preferred. The default value is 0.";
    }
    enum Prefer {
```

```

        value "1";
        description "Whether the static RP is
preferred. The value can be: 0: the static RP is not preferred. 1: the static
RP is preferred. The default value is 0.";
    }
}
}
}

container PIM-SPT-Switchs {

    list PIM-SPT-Switch {

        key "sptGrpPlyName";
        max-elements "unbounded";
        min-elements "0";
        description "Rate threshold of multicast packets
that can trigger a receiver's DR to send a Join message to a multicast source
to set up an SPT. By default, the receiver's DR performs the switchover from
the RPT to the SPT after receiving the first multicast data packet.";

        leaf infinity {
            description "The receiver's DR never triggers
the switchover from the RPT to the SPT.";
            config "true";
            type boolean {
            }
        }
        leaf sptThreshHold {
            description "Multicast packet rate threshold
that triggers the switchover from the RPT to the SPT. The value is an integer
ranging from 1 to 4194304, in kbit/s. ";
            config "true";
            default "1";
            type uint32 {
                range "1..4194304";
            }
        }
        leaf isACLEnable {
            description "Whether to enable a group policy.

";

            type boolean {
            }
            config "true";
            mandatory "true";

```

```

    }
    leaf sptGrpPlyName {
        description "Name of a named ACL. The value is
an integer ranging from 2000 to 2999, or a string of 1 to 32 case-sensitive
characters. If the parameter is not set, the threshold is applied to all
multicast groups. \"0\" means that no ACL is specified.";
        config "true";
        type string {
            length "1..32";
            pattern "^[^ ]+$";
        }
    }
}

```

```

        leaf sptGrpPlcyOrder {
            description "Adjust the order of the ACLs in
the group-policy list. If a group matches multiple ACLs, the threshold is
selected in the order specified by the order-value parameter. order-value
specifies the updated number. It is an integer. The value is any value other
than original one in the current group-policy list. If the parameter is not
set, the order of the ACLs in the group-policy list does not change.";
            config "true";
            default "4294967295";
            type uint32 {
                range "1..4294967295";
            }
        }
    }
}

```

```

container PIM_SM-Anycast-RPs {

```

```

    container PIM_SM-Anycast-RP {

```

```

        leaf IPV4-rpAddress {
            description "Address of an IPV4 Anycast-RP
peer";

            config "true";
            mandatory "true";
            type inet:ipv4-address;
        }

```

```

        leaf IPV6-rpAddress {
            description "Address of an IPV6 Anycast-RP
peer";

            config "true";
            mandatory "true";
            type inet:ipv6-address;
        }

```

```

        leaf local-IPV4-Address {
            description "Address of an IPV4 Anycast-RP
peer";

            config "true";
            mandatory "true";
            type inet:ipv4-address;
        }

```

```

        leaf local-IPV6-Address {
            description "Address of an IPV4 Anycast-RP
peer";

```

```
    config "true";  
    mandatory "true";  
    type inet:ipv6-address;  
}  
  
container PIM_SM-RP-Peers {  
  
    container PIM_SM-RP-Peer {
```

```

    leaf IPv4-rpPeer-Address {
        description "Address of an IPV4";

        config "true";
        mandatory "true";
        type inet:ipv4-address;
    }
    leaf IPV6-rpPeer-Address {
        description "Address of an IPV6";

        config "true";
        mandatory "true";
        type inet:ipv6-address;
    }
    leaf fwdSaSwt {
        description "Whether to forward to
Anycast-RP peers source information learned through SA messages.";
        config "true";
        type boolean {
        }
    }
    leaf fwdPolicy {
        description "Policy for filtering SA
messages to be forwarded to Anycast-RP peers. The value is an integer ranging
from 2000 to 2999, or a string of 1 to 32 case-sensitive characters.";
        config "true";
        type string {
            length "1..32";
            pattern "^[^ ]+$";
        }
    }
    leaf fwdPolicyIpv6 {
        description "Policy for filtering SA
messages to be forwarded to Anycast-RP peers. The value is an integer ranging
from 2000 to 2999, or a string of 1 to 32 case-sensitive characters.";
        config "true";
        type string {
            length "1..32";
            pattern "^[^ ]+$";
        }
    }
}
}
}

```

```
container PIM_SM-Interfaces {  
  list PIM_SM-Interface {  
    key "ifName";  
    max-elements "unbounded";  
    min-elements "0";  
    description "Specifies an PIM_SM interface.";
```

```
    leaf vrfName {
        description "Name of an PIM SM 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 addressFamily {
        config "true";
        mandatory "true";
        type enumeration {
            enum ipv4unicast {
                value "0";
                description "Address family, which
determines whether an address belongs to IPv4 or IPv6.";
            }
            enum ipv6unicast {
                value "1";
                description "Address family, which
determines whether an address belongs to IPv4 or IPv6.";
            }
        }
    }
    leaf ifName {
        description "Interface name.";
        config "true";
        mandatory "true";
        type ifName;
    }
    leaf pimsmEnable {
        description "Enable PIM-SM on an interface.";
        config "true";
        mandatory "true";
        type boolean {
        }
    }
    leaf drPriority {
        description "DR priority. The greater the
value, the higher the priority. The value is an integer ranging from 0 to
4294967295. The default value is 1.";
        config "true";
        default "1";
        type uint32 {
            range "0..4294967295";
        }
    }
}
```



```
    }  
  }  
  leaf helloInterval {  
    description "Interval at which Hello messages  
are sent. The value is an integer ranging from 1 to 2147483647, in seconds. It  
must be shorter than the timeout period of PIM neighbors. The default value is  
30.";  
    config "true";  
    default "30";  
  }  
}
```

```
        type uint32 {
            range "1..18000";
        }
    }
    leaf helloHoldtime {
        description "Timeout period during which a
router waits for Hello messages sent from its PIM neighbors. The value is an
integer ranging from 1 to 65535, in seconds. It must be longer than the
interval for PIM neighbors to send Hello messages. The default value is 105.";
        config "true";
        default "105";
        type uint16 {
            range "1..65535";
        }
    }
    leaf helloOverride {
        description "Interval at which the prune action
in a Hello message is overridden. The value is an integer ranging from 1 to
65535, in milliseconds. The default value is 2500.";
        config "true";
        default "2500";
        type uint16 {
            range "1..65535";
        }
    }
    leaf helloLanDelay {
        description "Period from the time when a router
receives a Prune message from a downstream device to the time when the router
performs the prune action. The value is an integer ranging from 1 to 32767, in
milliseconds. The default value is 500.";
        config "true";
        default "500";
        type uint16 {
            range "1..32767";
        }
    }
    leaf jpTimerInterval {
        description "Interval at which Join/Prune
messages are sent. The value is an integer ranging from 1 to 2147483647, in
seconds. The default value is 60. The interval must be shorter than the
holdtime of Join/Prune messages.";
        config "true";
        default "60";
        type uint32 {
            range "1..18000";
        }
    }
    leaf jpHoldtime {
```

description "Holdtime for Join/Prune messages sent by a router. The value is an integer ranging from 1 to 65535, in seconds. The default value is 210. The holdtime must be longer than the interval at which Join/Prune messages are sent.";

```
    config "true";
    default "210";
    type uint32 {
        range "1..65535";
    }
```

```
    }
    leaf jpPlyName {
```

description "Policy for filtering Join/Prune messages. The value is an integer ranging from 2000 to 2999, or a string of 1 to 32 case-sensitive characters. By default, no policy is configured to filter Join/Prune messages.";

```
    config "true";
    type string {
```

```
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf jpPlyIpv6 {
    description "Policy for filtering Join/Prune
messages. The value is an integer ranging from 2000 to 2999, or a string of 1
to 32 case-sensitive characters. By default, no policy is configured to filter
Join/Prune messages.";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf jpAsmPlyName {
    description "Policy for filtering ASM Join/
Prune messages. The value is an integer ranging from 2000 to 2999, or a string
of 1 to 32 case-sensitive characters. By default, no policy is configured to
filter ASM Join/Prune messages.";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf jpAsmPlyIpv6 {
    description "Policy for filtering ASM Join/
Prune messages. The value is an integer ranging from 2000 to 2999, or a string
of 1 to 32 case-sensitive characters. By default, no policy is configured to
filter ASM Join/Prune messages.";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf jpSsmPlyName {
    description "Policy for filtering SSM Join/
Prune messages. The value is an integer ranging from 3000 to 3999, or a string
of 1 to 32 case-sensitive characters. By default, no policy is configured to
filter SSM Join/Prune messages.";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
```

```

        leaf jpSsmPlyIpv6 {
            description "Policy for filtering SSM Join/
Prune messages. The value is an integer ranging from 3000 to 3999, or a string
of 1 to 32 case-sensitive characters. By default, no policy is configured to
filter SSM Join/Prune messages.";
            config "true";
            type string {
                length "1..32";
                pattern "^[^ ]+$";
            }
        }
        leaf nbrPlyName {
            description "Policy for filtering PIM
neighbors. The value is an integer ranging from 2000 to 2999, or a string of 1
to 32 case-sensitive characters. By default, no policy is configured to filter
PIM neighbors.";
            config "true";
            type string {

```

```
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf nbrPlyIpv6 {
    description "Policy for filtering PIM
neighbors. The value is an integer ranging from 2000 to 2999, or a string of 1
to 32 case-sensitive characters. By default, no policy is configured to filter
PIM neighbors.";
    config "true";
    type string {
        length "1..32";
        pattern "^[^ ]+$";
    }
}
leaf assertHoldtime {
    description "Timeout period during which PIM
interfaces wait to receive Assert messages from the forwarder. The value is an
integer ranging from 7 to 65535, in seconds. The default value is 180.";
    config "true";
    default "180";
    type uint32 {
        range "7..65535";
    }
}
leaf requireGenId {
    description "Configure a PIM interface to deny
Hello messages that do not carry generation IDs. By default, a PIM interface
receives Hello messages that do not carry generation IDs.";
    config "true";
    type boolean {
    }
}
leaf pimBsrBoundary {
    description "Boundary for a PIM domain. The
value can be: 0: no boundary; 1: bidirectional domain boundary; 2: inbound
unidirectional domain boundary.";
    config "true";
    default "None";
    type enumeration {
        enum None {
            value "0";
            description "Boundary for a PIM domain.
The value can be: 0: no boundary; 1: bidirectional domain boundary; 2: inbound
unidirectional domain boundary.";
        }
        enum Both {
            value "1";
        }
    }
}
```

```

        description "Boundary for a PIM domain.
The value can be: 0: no boundary; 1: bidirectional domain boundary; 2: inbound
unidirectional domain boundary.";
    }
    enum Incoming {
        value "2";
        description "Boundary for a PIM domain.
The value can be: 0: no boundary; 1: bidirectional domain boundary; 2: inbound
unidirectional domain boundary.";
    }
}
}
leaf bfdEnable {
    description "Enable PIM BFD on an interface. By
default, PIM BFD is not enabled on an interface.";
    config "true";

```

```
        type boolean {
        }
    }
    leaf bfdMinTx {
        description "Minimum interval at which BFD
messages are sent. The value is an integer ranging from 3 to 1000, in
milliseconds. ";
        config "true";
        default "0";
        type uint32 {
            range "0..1000";
        }
    }
    leaf bfdMinRx {
        description "Minimum interval at which BFD
messages are received. The value is an integer ranging from 3 to 1000, in
milliseconds. ";
        config "true";
        default "0";
        type uint32 {
            range "0..1000";
        }
    }
    leaf bfdMultiplier {
        description "Local detect multiplier for BFD
messages. The value is an integer ranging from 3 to 50. The default value is
3.";
        config "true";
        default "3";
        type uint16 {
            range "3..50";
        }
    }
    leaf isSilent {
        description "Enable the PIM silent function on
an interface. By default, PIM silent is disabled on an interface. ";
        config "true";
        default "false";
        type boolean {
        }
    }
    leaf isDrSwtDelay {
        description "Enable DR switchover delay on an
interface. By default, this function is disabled on an interface.";
        config "true";
        default "false";
        type boolean {
        }
    }
}
```



```
    }  
    leaf drSwtDelayInterval {  
        description "Delay for a DR switchover. The  
value is an integer ranging from 10 to 3600. The default value is 10. The  
setting is ineffective if the DR switchover delay function is not enabled.";  
        config "true";  
        default "10";  
        type uint32 {  
            range "10..3600";  
        }  
    }
```

```

    }
    leaf ipsecName {
      description "Description of an SA. The value is
a string of 1 to 15 characters.";
      config "true";
      type string {
        length "1..15";
        pattern "^[^ ]+$";
      }
    }
    leaf ipsecType {
      description "Hello options based on the IPsec
type.";
      config "true";
      type enumeration {
        enum forAll {
          value "0";
          description "Hello options based on the
IPsec type.";
        }
        enum forHello {
          value "1";
          description "Hello options based on the
IPsec type.";
        }
      }
    }
  }
}

```

5. IANA Considerations

This draft includes no request to IANA.

6. Security Considerations

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

7. Acknowledgements

TBD

Liu & Guo

Expires April 25, 2015

[Page 30]

8. References

8.1. Normative References

- [I-D.ietf-netmod-routing-cfg] Lhotka, L., "A YANG Data Model for Routing Management", [draft-ietf-netmod-routing-cfg-15](#) (work in progress), May 2014.
- [RFC3973] Adams, A., Nicholas, J., and W. Siadak, "Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised)", [RFC 3973](#), January 2005.
- [RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", [RFC 4601](#), August 2006.
- [RFC4607] Holbrook, H. and B. Cain, "Source-Specific Multicast for IP", [RFC 4607](#), August 2006.
- [RFC4608] Meyer, D., Rockell, R., and G. Shepherd, "Source-Specific Protocol Independent Multicast in 232/8", [BCP 120](#), [RFC 4608](#), August 2006.
- [RFC4610] Farinacci, D. and Y. Cai, "Anycast-RP Using Protocol Independent Multicast (PIM)", [RFC 4610](#), August 2006.
- [RFC5015] Handley, M., Kouvelas, I., Speakman, T., and L. Vicisano, "Bidirectional Protocol Independent Multicast (BIDIR-PIM)", [RFC 5015](#), October 2007.
- [RFC5059] Bhaskar, N., Gall, A., Lingard, J., and S. Venaas, "Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)", [RFC 5059](#), January 2008.
- [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.

8.2. 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

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

Email: guofeng@huawei.com

