PIM WG
Internet-Draft
Intended status: Standards Track
Expires: October 26, 2016

Zheng. Zhang
BenChong. Xu
ZTE Corporation
April 24, 2016

MSDP YANG model
draft-zhang-pim-msdp-yang-00

Abstract

This document defines a YANG data model for MSDP protocol configuration and operation.

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 October 26, 2016.

Copyright Notice

Table of Contents

1. Introduction

[RFC3618] introduces protocol definition of MSDP. This document defines a YANG data model for MSDP. The content is in keeping with [RFC3618].

2. Design of the Data Model

The msdp peer and source-active content is important part of MSDP. The augment should be added below routing-protocol.

```
module: ietf-msdp
augment /rt:routing-state/rt:routing-instance/rt:routing-protocols/:
module: ietf-msdp
   +--rw msdp-configure
   |  +--rw peer-information
   |  |  +--rw peer-list* [peer-addr]
   |  |     +--rw peer-addr     inet:ip-address
   |  |     +--rw mesh-group?   string
   |  |     +--rw bfd
   |  |     |  +--rw enabled?   boolean
   |  |     +--rw peer-timer
   |  |        +--rw hold-timer?           uint16
   |  |        +--rw keepalive-timer?      uint16
   |  |        +--rw connectRetry-timer?   uint16
   |  +--rw source-active-cache
   |     +--rw adv-peer-list* [adv-peer-addr]
   |        +--rw adv-peer-addr    inet:ip-address
   |        +--rw rp-entry* [rp-addr]
   |           +--rw rp-addr     inet:ip-address
   |           +--rw sa-cache* [count]
   |              +--rw count               uint32
   |              +--rw sa-adv-timer?       uint16
   |              +--rw sa-cache-timeout?   uint16
   |              +--rw source-addr         inet:ip-address
   |              +--rw group-addr          inet:ip-address
   +--ro msdp
      +--ro peer-information
```

```
        |  +--ro peer-list* [peer-addr]
        |     +--ro peer-addr     inet:ip-address
        |     +--ro mesh-group?   string
        |     +--ro bfd
        |     |  +--ro enabled?   boolean
        |     +--ro peer-timer
        |     |  +--ro hold-timer?           uint16
        |     |  +--ro keepalive-timer?      uint16
        |     |  +--ro connectRetry-timer?   uint16
        |     +--ro (peer-state)
        |        +--:(disabled)
        |        +--:(inactive)
        |        +--:(connecting)
        |        +--:(listen)
        |        +--:(established)
        +--ro source-active-cache
           +--ro adv-peer-list* [adv-peer-addr]
              +--ro adv-peer-addr    inet:ip-address
              +--ro rp-entry* [rp-addr]
                 +--ro rp-addr     inet:ip-address
                 +--ro sa-cache* [count]
                    +--ro count               uint32
                    +--ro sa-adv-timer?       uint16
                    +--ro sa-cache-timeout?   uint16
                    +--ro source-addr         inet:ip-address
                    +--ro group-addr          inet:ip-address
  notifications:
     +---n msdp-notification
        +--ro peer-state-change
        |  +--ro peer-addr    inet:ip-address
        |  +--ro (peer-state)
        |     +--:(disabled)
        |     +--:(inactive)
        |     +--:(connecting)
        |     +--:(listen)
        |     +--:(established)
        +--ro tlv-format-error
           +--ro peer-addr           inet:ip-address
           +--ro sa-error-message
              +--ro adv-peer-list* [adv-peer-addr]
                 +--ro adv-peer-addr    inet:ip-address
                 +--ro rp-entry* [rp-addr]
                    +--ro rp-addr     inet:ip-address
                    +--ro sa-cache* [count]
                       +--ro count               uint32
                       +--ro sa-adv-timer?       uint16
                       +--ro sa-cache-timeout?   uint16
                       +--ro source-addr         inet:ip-address
```

```
                      +--ro group-addr         inet:ip-address
```

3. MSDP configuration

The msdp peers should be configured at first. And several peers may be in a mesh-group. The Source-Active information may be filtered for peers. So if the filter policy is needed, we will add it later.

The state of peer should not be configured except for disabled.

BFD should be enabled/disabled for a specific peer. There is something wrong that ietf-bfd is referenced, so we keep the basic configuration of BFD. The parameter should be referenced later.

4. Notifications

The peer state changing should be notified. And if there is error in TLV format, the notification should be sent.

5. MSDP YANG model

<CODE BEGINS> file "ietf-msdp.yang"

```
module ietf-msdp {

 namespace "urn:ietf:params:xml:ns:yang:ietf-msdp";

 prefix msdp;

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

 /*import ietf-bfd {
    prefix "bfd";
 }*/

 organization " IETF PIM( Protocols for IP Multicast ) Working Group";
 contact
     "WG List:  <mailto:pim@ietf.org>
      WG Chair: Stig Venaas
                      <stig@venaas.com>
      WG Chair: Mike McBride
                      <mmcbride7@gmail.com>

      Editor:   Zheng Zhang
                <mailto:zhang.zheng@zte.com.cn>
      Editor:   Benchong Xu
```

```
                  <mailto:xu.benchong@zte.com.cn>
        ";

    description
        "This module contains a collection of YANG definitions for
         managing msdp protocol.";

    revision 2016-04-23 {
        description
        "Initial version.";
        reference "https://tools.ietf.org/html/draft-zhang-pim-msdp-yang";
    }

    /*feature bfd-protocol-parms {
       description "BFD protocol specific parameters support.";
    }*/

/*SA information*/
    grouping sa-message {
        description "The Source-Active message information.";

        list adv-peer-list {
            key "adv-peer-addr";
            description "The advertisement information of one peer.";
            leaf adv-peer-addr {
                type inet:ip-address;
                mandatory true;
                description "The address of peer that advertises the SA message.
                  When this value is set to zero mearns that the sa message is
                  originated by this node itself.";
            }

            list rp-entry {
                key "rp-addr";
                description "The SA entry information of one RP.";
                leaf rp-addr {
                    type inet:ip-address;
                    mandatory true;
                    description "The address of RP.";
                }
                list sa-cache {
                    key "count";
                    description "The SA cache information.";
                    leaf count {
                        type uint32;
                        mandatory true;
                        description "The sequence number of this SA cache.";
                    }
```

```
                    leaf sa-adv-timer {
                        type uint16 {
                            range "1 .. 65535";
                        }
                        units "seconds";
                        default 60;
                        description "The SA-Advertisement-Period of this SA cache.";
                    }
                    leaf sa-cache-timeout {
                        type uint16 {
                            range "1 .. 65535";
                        }
                        units "seconds";
                        description "The timeout interval of this SA cache. The
                          value of this timer MUST NOT be less than
                          [SA-Advertisement-Period] + [SA-Hold-Down-Period].";
                    }
                    leaf source-addr {
                        type inet:ip-address;
                        mandatory true;
                        description "The address of source.";
                    }
                    leaf group-addr {
                        type inet:ip-address;
                        mandatory true;
                        description "The address of group.";
                    }
                }
            }
        }
    }

/*peer information*/
        grouping peer-state {
        description "The collection of msdp peer state.";
        choice peer-state {
            mandatory true;
            case disabled {
                description "The state of this msdp peer is disabled.";
            }
            case inactive {
                description "The tcp connection of this peer is not established yet.";
            }
            case connecting {
                description "The tcp connection of this peer is in process.";
            }
            case listen {
                description "The tcp connection of this peer is in process, and the ip address of this n
```

```
            }
            case established {
                description "The tcp connection of this peer is established.";
            }
            description "The collection of all possible peer state.";
        }
    }

        grouping peer-timer {
            description "The timer of msdp peer.";
            leaf hold-timer {
            type uint16 {
                range "1 .. 65535";
            }
            units "seconds";
            default 75;
            description "The SA-Hold-Down-Period of this msdp peer.";
        }
        leaf keepalive-timer {
            type uint16 {
                range "1 .. 65535";
            }
            units "seconds";
            default 60;
            description "The keepalive timer of this msdp peer.";
        }
        leaf connectRetry-timer {
            type uint16 {
                range "1 .. 65535";
            }
            units "seconds";
            default 30;
            description "The connect retry timer of this msdp peer.";
        }
    }

    container msdp-configure {
        description "The information of msdp protocol.";
        container peer-information {
            description "The information of msdp peer.";

            list peer-list {
                key "peer-addr";
                description "The information of this msdp peer.";
                leaf peer-addr {
                    type inet:ip-address;
                    mandatory true;
                    description "The address of this msdp peer.";
```

```
                }
                leaf mesh-group {
                    type string;
                    description "The mesh-group that the msdp peer belongs to.";
                }
                container bfd {
                    description "BFD operation.";
                    leaf enabled {
                        type boolean;
                        description "True if BFD is enabled for the peer.";
                    }
                    /*uses bfd:bfd-grouping-base-cfg-parms {
                        if-feature bfd-protocol-parms;
                    }*/
                }
                container peer-timer {
                    description "The timer information of this peer.";
                    uses peer-timer;
                }
            }
        }

        container source-active-cache {
            description "The source active cache information.";
            uses sa-message;
        }
    }

    container msdp {
        config false;
        description "The information of msdp protocol.";
        container peer-information {
            description "The information of msdp peer.";

            list peer-list {
                key "peer-addr";
                description "The information of this msdp peer.";
                leaf peer-addr {
                    type inet:ip-address;
                    mandatory true;
                    description "The address of this msdp peer.";
                }
                leaf mesh-group {
                    type string;
                    description "The mesh-group that the msdp peer belongs to.";
                }
                container bfd {
                    description "BFD operation.";
```

```
                    leaf enabled {
                        type boolean;
                        description "True if BFD is enabled for the peer.";
                    }
                    /*uses bfd:bfd-grouping-base-cfg-parms {
                        if-feature bfd-protocol-parms;
                    }*/
                }
                container peer-timer {
                    description "The timer information of this peer.";
                    uses peer-timer;
                }
                uses peer-state;
            }
        }

        container source-active-cache {
            description "The source active cache information.";
            uses sa-message;
        }
    }

/*notification*/
        notification msdp-notification {
        description "This notification is sent when a condition changes in msdp.";

        container peer-state-change {
            description "The msdp peer state changes.";
            leaf peer-addr {
                type inet:ip-address;
                mandatory true;
                description "The address of this msdp peer.";
            }
            uses peer-state;
        }

        container tlv-format-error {
            description "The tlv format is error from this peer.";
            leaf peer-addr {
                type inet:ip-address;
                mandatory true;
                description "The address of this msdp peer.";
            }
            container sa-error-message {
                description "The source active message information.";
                uses sa-message;
            }
        }
```

```
    }
}
<CODE ENDS>
```

6. Normative References

[RFC3618] Fenner, B., Ed. and D. Meyer, Ed., "Multicast Source Discovery Protocol (MSDP)", RFC 3618, DOI 10.17487/RFC3618, October 2003, <http://www.rfc-editor.org/info/rfc3618>.

[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>.

Authors' Addresses

Zheng(Sandy) Zhang
ZTE Corporation
No. 50 Software Ave, Yuhuatai Distinct
Nanjing
China

Email: zhang.zheng@zte.com.cn

BenChong Xu
ZTE Corporation
No. 50 Software Ave, Yuhuatai Distinct
Nanjing
China

Email: xu.benchong@zte.com.cn