Filter-Based RIB Data Model
draft-hares-i2rs-fb-rib-data-model-00

Abstract

   This document defines a data model for the I2RS Filter-based Routing
   Information Base (RIB) Yang data model. A routing system uses the
   Filter-based RIBto program FIB entries that process incoming packets
   by matching on multiple fields within the packet and then performing
   a specified action on it. The FB-RIB can also specify an action to
   forward the packet according to the FIB entries programmed using the
   RIBs of its routing instance.

Table of Contents

1. Introduction

   The Interface to the Routing System (I2RS)
   [I-D.ietf-i2rs-architecture] architecture provides dynamic read and
   write access to the information and state within the routing
   elements. The I2RS client interacts with the I2RS agent in one or
   more network routing systems.

   This document provides a yang module for the I2RS filter Based
   Routing Information Base (FB-RIB) and describes the I2RS interaction
   with routing filters within a routing element. The informational
   model for the FB-RIB is in [I-D.kini-i2rs-fb-rib-info-model]

1.1. Definition of I2RS Filter Based RIB

   Filter-based routing is a technique used to make packet forwarding
   decisions based on a filter that is matched to the incoming packets
   and the specified action. It should be noted that that this is
   distinct from the static routes in the RIB
   [I-D.ietf-i2rs-rib-info-model] where the routing is destination
   ddress based.

   A Filter-Based RIB (Routing Information Base) is contained in a
   routing instance (defined in [I-D.ietf-i2rs-rib-info-model]). It
   contains a list of filters (match-action conditions), a list of
   interface the filter-based forwarding operates on. Filter-based RIBs
   (FB-RIBs) operate only on the interface the FB-RIB are configured on.

   A Filter Based RIB uses Event-Condition-Action policy. A Filter-
   based RIB entry specifies matches on fields in a packet (which may
   include layer 2 fields, IP header fields, transport or application
   fields) or size of the packet or interface received on. The matches
   are contained in an ordered list of filters which contain pairs of
   match condition-action (aka event-condition-action).

   If all matches fail, default action is to forward the packet using
   FIB entries that were programmed by the Routing Informational Base
   (RIB) manager described in [I-D.ietf-i2rs-rib-info-model].

   Actions in the condition-action pair may impact forwarding or set
   something in the packet that will impact forwarding. Policy actions
   are typically applied before applying QoS constraints since policy
   actions may override QoS constraint.

   The Filter-Based RIB resides in ephemeral state as does the I2RS RIB
   and I2RS topology models.

2. 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 [RFC2119].

   In this document, these words will appear with that interpretation
   only when in ALL CAPS. Lower case uses of these words are not to be
   interpreted as carrying RFC-2119 significance.

3. Definitions and Acronyms

     CLI

        Command Line Interface

     FB-RIB

        Filter-Based Routing Information Base

     FB-Route

        The policy rules in the filter-based RIB are prescriptive of the
        Event-Condition-Action form which is often represented by if
        Condition then action".

     Policy Group

        Policy Groups are groups of policy rules. The groups of policy in
        the basic network policy [I-D.hares-i2rs-bnp-info-model] allow
        grouping of policy by name. This name allow easier management of
        customer-based or provider based filters.

     RIB IM

        RIB Informational Model (RIB IM) [I-D.ietf-i2rs-rib-info-model]

     Routing instance

        A routing instance, in the context of the FB-FIB is a collection
        of RIBs, interfaces, and routing parameters. A routing instance
        creates a logical slice of the router and allows different logical
        slices; across a set of routers; to communicate with each other.

4. The Top-level Yang structure for the FB-RIB

```
    module: FB-RIB
    +--FB-RIB-module
       +--rw FB-RIB-instance-name
       +--rw RB-RIB-router-id
          uses rt:router-id
       +--rw FB-RIB*  [rib-name]
             |  +--rw rib-Name
             |  +--rw rib-afi
        |  +--rw fb-rib-intf* if:inteface-ref
             |  +--rw I2RS-default-RIB
             |  |  +--rw RIB-name
             |  |    uses i2rs-rib:name
             |  +--rw fb-rib-status-info
             |  |  +--rw fb-rib-update-ref uint64
             |  +--rw fb-rib-Group*
            +-rw filter-type  // for group
            +-rw order-number // for group
              + choice (filter-type)
                +-case: acl
                 uses: acl: access_lists: access-list-entries
                            // operational status augment to group
                 augments: access_lists: access-list-entries
                 uses fb-rib-group-order_status;

           // operational status augment to individual ACL
                augments: access_lists:access-list-entries:
                                      access-list-entry
                             uses fb-rib-rule-order-status;

             +--case: bnp-eca Rules
                uses bnp-eca: bnp-policy-set
                   augments bnp-eca:bnp-policy-set:group-list:group
                        uses fb-rib-group-order_status
                   augment bnp-eca:bnp-policy-set:group-list:group:rule
                        uses fb-rib-rule-order_status
```

                    Figure 4: FB RIB Yang Structure

5. yang models

5.1. Filter-Based RIB types

```
        //<CODE BEGINS> file "i2rs fb-rib-types@2015-10-17.yang"
    module i2rs-fb-rib-types {

      yang-version "1";

        // namespace
```

```
      namespace "urn:TBD1:params:xml:ns:yang:rt:i2rs:fb-rib-types";
        // replace with iana namespace when assigned
        prefix "i2rs-fbrib-t";

      // meta
      organization
        "IETF";

      contact
         "email: sriganesh.kini@ericsson.com
          email: cengiz@packetdesign.com
          email: anoop@ieee.duke.edu
          email: ivandean@gmal.org
          email: shares@ndzh.com;
          email: linda.dunbar@huawei.com;
          email: russ@riw.com;
          email: Jeff.Tantsura@ericsson.com;
             ";

      description
        "This module describes a YANG model for the I2RS
        Filter-based RIB Types.  These types
        specify types for the Filter-Based RIB.

            Copyright (c) 2015 IETF Trust and the persons identified as
        the document authors.  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).";


      revision "2015-06-20" {
          description
            "I2RS Filter-Based RIB protocol ";
          reference "draft-hares-i2rs-fb-rib-data-model-00";
      }

      typedef fb-rib-policy-type-def {
            type identityref {
              base "fb-rib-policy-type";
                  }
                  description
                  "This type is used to refer to FB-RIB type";
        }
```

```
             identity fb-rib-policy-type {
               description
               "Types of filter-based policies:
                    acl and eca";
             }

             identity fb-rib-acl {
                    base fb-rib-policy-type;
                 description
              "filter based policy based on access-lists";
               }

             identity fb-bnp-eca-rules {
                    base fb-rib-policy-type;
                 description
              "filter based policy based on qos forwarding rules";
               }

        typedef fb-rules-status  {
           type identityref {
             base "fb-rule-opstat";
                 }
               description
               "This type is used to refer to FB-RIB type";
         }

           identity fb-rule-opstat {
                 description
                 "operational statues for filter rules
                  inactive and active";
                  }

           identity fb-rule-inactive {
                 base fb-rule-opstat;
                 description
                 "policy rule is inactive";
            }

           identity fb-rule-active {
                 base fb-rule-opstat;
                 description
                 "policy rule is active";
            }

           grouping fb-rib-rule-order-status {
           leaf statement-order {
                 description "order identifier";
                 type uint16;
```

```
               }
             leaf statement-oper_status {
                  description "status of rule";
                  type fb-rules-status;
                 }
          }

       grouping fb-rib-group-order-status {
            leaf group-order{
              type uint16;
              description "default group order";
            }
            leaf group-refcnt {
              type uint16;
              description "refcnt for this group";
                 }
             leaf group-installed {
              type uint16;
              description "number of rules installed";
                 }
           }

           grouping fb-rib-status-info {
            leaf fb-rib-update-ref {
              description
                  "number of updates to this FB RIB
                   since last reboot";
                  type uint64;
           }
          }
        }
     }

   // <CODE ENDS>
```

5.2. fb-fib yang module

```
 //<CODE BEGINS > file "i2rs fb-rib@2015-10-17.yang"
 module i2rs-fb-rib {

   yang-version "1";

   // namespace
   namespace "urn:TBD1:params:xml:ns:yang:rt:i2rs:fbrib";
     // replace with iana namespace when assigned
     prefix "i2rs-fbrib";
```

```
    // import some basic inet types
    import ietf-interfaces { prefix "if";}
    import ietf-routing { prefix "rt"; }
    import i2rs-rib { prefix "i2rs-rib"; }
    import i2rs-fb-rib-types { prefix "i2rs-fbrib-t";}
    import i2rs-eca-policy { prefix "i2rs-eca";}

    // meta
    organization
      "IETF";

    contact
       "email: sriganesh.kini@ericsson.com
            email: cengiz@packetdesign.com
        email: anoop@ieee.duke.edu
        email: ivandean@gmal.org
        email: shares@ndzh.com;
        email: linda.dunbar@huawei.com;
        email: russ@riw.com;
        email: Jeff.Tantsura@ericsson.com;
            ";

    description
      "This module describes a YANG model for the I2RS
          Filter-based RIB which is a protocol independent I2RS module.";


      revision "2015-10-18" {
        description "initial revision";
        reference "draft-kini-i2rs-fb-rib-info-model-03";
      }

            grouping fb-rib-group-list-entry  {
                  description "lists of groups of ordered lists";
                      leaf group-name {
                          description "policy-group name";
                          type string;
                          }
                          uses i2rs-fbrib-t:fb-rib-group-order-status;
                          leaf fb-group-policy-type {
                            description "Policy type (acl/eca)";
                            type i2rs-fbrib-t:fb-rib-policy-type-def;
                          }
                          choice fb-rib-policy-type {
                      case fb-rib-acl {
                // re-add when ietf-access-control-list works
                // uses acl:access_lists;
                        uses i2rs-fbrib-t:fb-rib-group-order-status;
```

```
                              }
                           case fb-eca-rules {
            /// re-add i2rs-eca:bnp-ecap-policy-set:policy groups
             /// linked to this
                uses i2rs-fbrib-t:fb-rib-group-order-status;
                            }
                            }

                  container fb-rib_status {
                    leaf fb-rib-update-ref {
               type uint64;
               description "reference count for fb-rib";
                    }
                        }
                 }

                 grouping I2RS-Default-RIB {
                        description "I2RS RIB which will be used
                           even if none of the policy match";
              // problem with my use format - add back after fix
                  //   uses i2rs-rib:rib-list:name;
                           uses i2rs-fbrib-t:fb-rib-status-info;
                  }

    /// top level FB-RIB structure
     container FB-RIB-routing-instance {
        description
        "Configuration of an 'i2rs' pseudo-protocol instance
         consists of a list of ribs.";

         leaf FB-RIB-instance-name {
           description
             "A routing instance is identified by its name,
              INSTANCE_name.  This MUST be unique across all routing
              instances in a given network device.";
              type string ;
              mandatory true;
            }
           uses rt:router-id;

                   list FB-RIB {
                     key rib-Name;
                     leaf rib-Name {
                          description "RIB name";
                          type string;
                              mandatory true;
                 }
                         // fix the afi list to key point
```

```
             leaf rib-afi {
                        description "RIB AFI";
                        type uint32;
                      }
              list fb-rib-intf {
                      description
                       "This represents the list of interfaces
                       associated with this routing instance.
                       The interface list helps constrain the
                       boundaries of packet forwarding.
                       Packets coming on these interfaces are
                       directly associated with the given routing
                       instance. The interface list contains a
                       list of identifiers, with each identifier
                       uniquely identifying an interface.";
                       key "name";
                       leaf name {
                             type if:interface-ref;
                         description
                          "A reference to the name of a
                             configured network layer
                          interface.";
                       }
               }
                      uses I2RS-Default-RIB;

                      list fb-rib-group {
                         description "list of policy groups ";
                         key grp-name;
                         uses i2rs-eca:rule-group;
                       }
                  }
              }
     }
  // <CODE ENDS>
```

6. IANA Considerations

    TBD.

7. Security Considerations

    A I2RS RIB is ephemeral data store that will dyanamically change
    traffic paths set by the routing configuration. An I2RS FB-RIB
    provides dynamic Event-Condition-Action policy that will further
    change the operation of forwarding by allow dyanmic policy and

ephemeral RIBs to alter the traffic paths set by routing
configuration. Care must be taken in deployments to use the
appropriate security and operational control to make use of the tools
the I2RS RIB and I2RS FB-RIB provide.

8. References

8.1. Normative References:

   [I-D.hares-i2rs-bnp-info-model]
            Hares, S., Wu, Q., Tantsura, J., and R. White, "An
            Information Model for Basic Network Policy and Filter
            Rules", draft-hares-i2rs-bnp-info-model-02 (work in
            progress), March 2015.

   [I-D.ietf-i2rs-architecture]
            Atlas, A., Halpern, J., Hares, S., Ward, D., and T.
            Nadeau, "An Architecture for the Interface to the Routing
            System", draft-ietf-i2rs-architecture-09 (work in
            progress), March 2015.

   [I-D.ietf-i2rs-rib-data-model]
            Wang, L., Ananthakrishnan, H., Chen, M.,
            amit.dass@ericsson.com, a., Kini, S., and N. Bahadur, "A
            YANG Data Model for Routing Information Base (RIB)",
            draft-ietf-i2rs-rib-data-model-01 (work in progress),
            September 2015.

   [I-D.ietf-i2rs-rib-info-model]
            Bahadur, N., Kini, S., and J. Medved, "Routing Information
            Base Info Model", draft-ietf-i2rs-rib-info-model-07 (work
            in progress), September 2015.

   [I-D.ietf-netmod-acl-model]
            Bogdanovic, D., Sreenivasa, K., Huang, L., and D. Blair,
            "Network Access Control List (ACL) YANG Data Model",
            draft-ietf-netmod-acl-model-03 (work in progress), June
            2015.

   [I-D.kini-i2rs-fb-rib-info-model]
            Kini, S., Hares, S., Dunbar, L., Ghanwani, A., Krishnan,
            R., Bogdanovic, D., Tantsura, J., and R. White, "Filter-
            Based RIB Information Model", draft-kini-i2rs-fb-rib-info-
            model-01 (work in progress), July 2015.

8.2. Informative References

   [I-D.ietf-i2rs-usecase-reqs-summary]
             Hares, S. and M. Chen, "Summary of I2RS Use Case
             Requirements", draft-ietf-i2rs-usecase-reqs-summary-01
             (work in progress), May 2015.

   [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>.

Authors' Addresses

   Susan Hares
   Huawei
   7453 Hickory Hill
   Saline, MI 48176
   USA

   Email: shares@ndzh.com

   Sriganesh Kini
   Ericsson

   Email: sriganesh.kini@ericsson.com

   Linda Dunbar
   Huawei
   USA

   Email: linda.dunbar@huawei.com

   Anoop Ghanwani
   Dell

   Email: anoop@alumni.duke.edu

   Ram Krishnan
   Dell

   Email: Ramkri123@gmail.com

Dean Bogdanovic
Juniper Networks
Westford, MA

Email: deanb@juniper.net

Jeff Tantsura
Ericsson

Email: jeff.tantsura@ericsson.com

Russ White
Ericsson

Email: russ@riw.us