I2RS working group   S. Hares
Internet-Draft   Huawei
Intended status: Standards Track   R. White
Expires: September 22, 2016   LinkedIn
                                                      March 21, 2016

                      Filter-Based RIB Data Model
                      draft-hares-rtgwg-fb-rib-00

Abstract

   This document defines a yang data model for a Filter-based Routing
   Information Base (RIB) Yang data model. A routing system uses the
   Filter-based RIB to program FIB entries that process incoming packets
   by matching on multiple fields (n-tuple) 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.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

    This document provides a yang module for flow filter n-tuple policy
    that is locally configured. This flow filter policy has also been
    called Policy routing in some implementations.

    This document defines a yang data model for a Filter-based Routing
    Information Base (RIB) Yang data model. A routing system uses the
    Filter-based RIB to 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.

1.1. Definition of I2RS Filter Based RIB

    Filter-based routing is a technique used to make packet forwarding
    decisions based on a n-tuple 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 following RIBS:

o configured RIB created using static routes in
   [I-D.ietf-netmod-routing-cfg]

o Extended static RIB defined in [I-D.acee-rtgwg-yang-rib-extend],

o Ephmeral Protocol Independent RIB defined in
   [I-D.ietf-i2rs-rib-info-model], or

A Filter-Based RIB (Routing Information Base) is contained in a
routing instance. 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 packet forwarding policy. If packet
reception is considered an event, then the I2RS Filter-based RIB uses
a minimalistic Event-Condition-Action policy. A Filter-based RIB
entry specifies matche filters for the fields in a packet (which may
include layer 1 to layer 3 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, the default action is to forward the packet
using FIB entries that were programmed by the default Routing
Informational Base (RIB) manager configured in the Filter-Based RIB
(FB-RB)

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.

1.2. 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.

1.3. 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. All policy in the filter-based RIB are in
      a ordered list, ordered by "order-number". Order number is
      similar to some CLI concepts of line number.

   Policy Group

      Policy Groups are groups of policy rules that are set-up for the
      convenience of operators who wish to link the rules connected to a
      particular client.

      * Groups do not affect the order of policy rulies.

      * The policy groups in the basic network policy
         [I-D.hares-i2rs-pkt-eca-data-model] allow grouping of policy by
         name. This name allow easier management of customer-based or
         provider based filters. This policy group is a second way to
         access certain policy rules on the policy rule list.

    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.

1.4. Yang High Level (YHL) graphical form

   The High-level Yang graphical representation uses the following
   symbols:

Brackets "[" and "]" enclose list keys.

Curly braces "{" and "}" contain names of optional features that
make the corresponding node conditional.

Abbreviations before data node names: "rw" means configuration
(read-write), "ro" state data (read-only), "-x" RPC operations,
and "-n" notifications.

Symbols after data node names: "?" means an optional node, "!" a
container with presence, and "*" denotes a "list" or "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. Where Filter-Based RIB Fits in Global RIBs

   The Top-level Yang structure for a global FB-RIB types (similar to
   acl) is not defined. The Filter-Based RIB should be defined under
   this structure under a routing instance. The two things under this
   RIB would be: configured Filter-Based RIB (aka Policy routing), I2RS
   reboot Ephemeral Filter-Based RIB. ACLs [I-D.ietf-netmod-acl-model]
   have the potential to be augmented to be included, but this version
   of this document does address that issue.

   The purpose of this section is illustrate why the flow specification
   policy installed in yang modules loaded into intended configuration
   needs to be able to be compared. After demonstrating why this is
   needed, this section suggests a structure for filter-based RIBS.

   BGP's Flow Specification (BGP-FS) configures filter-based policy in
   the local BGP configuration, and passes this information in BGP
   packets (in NLRI and Extended Communities). The BGP-FS YANG model
   [I-D.wu-idr-flowspec-yang-cfg] specifies the locally configuration,
   and the derived state that includes the BGP Flow Specifications
   received. BGP-FS processing may install the locally configured BGP
   Flow specification in the local FB-RIB. If it does, this policy is
   like any other locally configured policy.

   The BGP-FS may installed the flow policy received from a remote BGP
   peer and stored in derived state. This policy has a different
   characteristics as it will disappear if the peer connection between
   the two peers drops, or if the peer changes the BGP-FS policy. Due
   to the ephemeral nature of the BGP-FS, it should be installed unique.
   Otherwise, If the local configuration state changes, it cannot

differentiate between the true configured state and the ephemeral
states (I2RS ephemeral and BGP-session ephemeral). Both I2RS
ephemeral and BGP-session ephemeral policy will disappear upon a
reboot.

```
ietf-fb-rib module
+--rw routing-instance
   +--rw ietf-fb-rib
      +--rw default-instance-name string
      +--rw default-router-id rt:router-id
         +--rw config-fb-rib // config state
            uses fb-ribs
         +--rw I2rs-fb-rib  // ephemeral state
            uses fb-ribs
         +--rw BGP-FB-RIB   // Install derived
            uses fb-ribs    // BGP-FS policy state
```

     Figure 6: Global FB RIB Yang Structure

I2RS architecture [I-D.ietf-i2rs-architecture] specifies that by
default the Local configuration will win if the local configuration
changes. In the NETCONF/NETMOD language, the "last write wins".

An example will help illustrate this:

    local configuration installs filter for IP-Dest=128.2/16, IP-
    SRC=192.5.7/24 DPORT=ALL drop in the running configuration, and
    then synchronously loads it to the intended configuration and
    applied configuration.

    I2RS installs an ephemeral filter for IP-Dest=128.2/16, IP-
    SRC=192.5.7/24 DPORT=125 forward intended configuration
    synchronously.

    BGP-FS processing installs BGP-FS policy for IP-Dest=128.2/16, IP-
    SRC=192.5.7/24 DPORT=125 forward, traffic-rate by bytes.

    local configuration install a filter for IP-Dest=128.2/16, IP-
    SRC=192.5.7/24, DPort=125, drop. This local configuration policy
    would win over the I2RS policy and the BGP-FS. The I2RS process
    is required to receive an event indicating the overwrite. The
    BGP-FS process should also receive an event indicating an
    overwrite.

The I2RS [I-D.ietf-i2rs-architecture] also allows that the preference
between local-configuration and I2RS ephemeral state can be
determined by operator-applied policy. However, illustrations of
this are out of scope for this version of this document.

3. Proposed Structure for Filter-Based RIBs

    There are three levels in the Filter-Based RIBs (FB-RIB) structure:

    o   a global FB-RIB structures,

    o   the common structure of the FB-RIB, and

    o   the groupings that make up the FB-RIB

    All structures have two types: configuration/ephemeral state and
    operational state.

    This yang model describes three types of FB-RIBS: configuration,
    I2RS, and BGP Flow Specification. The configuration FB-RIB yang
    module is config state ("config true" and "ephemeral false") and
    survives a reboot. The I2RS FB-RB yang model is reboot ephemeral
    ("config true" and "ephemeral true"). The BGP Flow Specification
    Filter-Based RIB stores policy which is received by the BGP peers,
    and can be considered policy configured as part of BGP infrastructure
    ("config true" and "peer-ephemeral true;")

Configuration RIBS

bgp-fs-fb-rib - is the BGP processes installation of
     the BGP Flow Specification (BGP-FS) policy rules
          from remote peers. Locally configured
          BGP-FS rules are configured in the BGP peer
          structure.

```
      +-----------------------------------------+
      |      routing instance                   |
      +-------|------------|---------------|--+
              |            |               |
              |            |               |
              |            |               |
  +---------|----+  +-----|-----+ +--------|-----+
  |config-fb-rib |  |i2rs-fb-rib| |bgp-fs-fb-rib |
  +------|-------+  +-----|------+ +------|------+
         |.............:...|...............|
                      :  (uses common structures
                      :   in separate lists of FB-RIBs)
              +--------|----+
              |fb-ribs*     |
              |            |
              +--|----------+
                 |
```
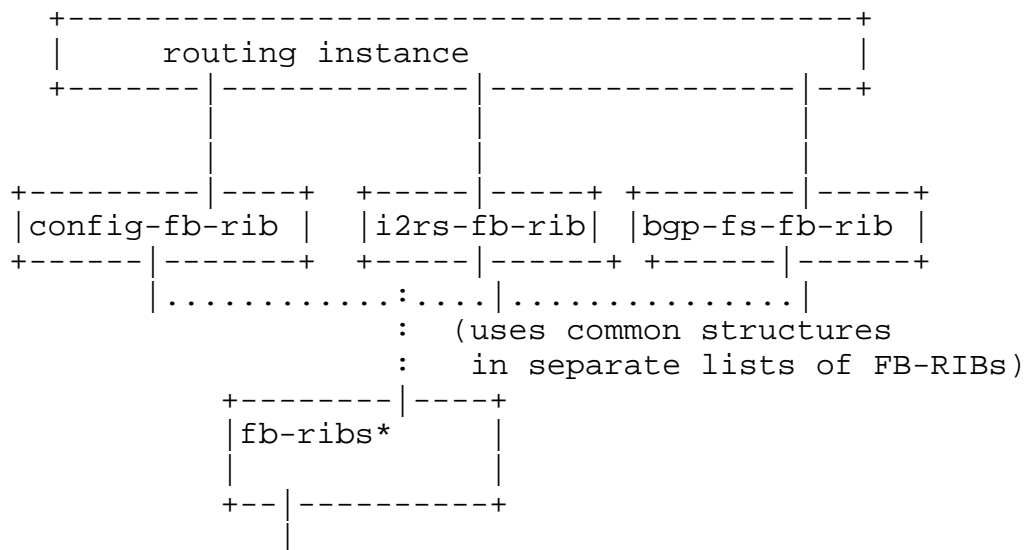
        Figure 3: Routing instance with three types of
                  Filter-FIB lists

4. Yang High Level Structure for FB-RIBs

   The following section provides the high level yang structure diagrams
   for the following levels of structures for both config/ephemeral
   state and operationa.

   o ietf-fb-rib - contains filter-based RIBS for config, I2RS FB-RIB,
      and BGP Flow Specification.

   o fb-rib - that contains the structures for the filter-based
      grouping

   o fb-rib-types - that contains the structures for groupings within
      the filter-based RIBS

   These structures are contained within the yang section in this draft.

   The packet-reception ECA policy yang module is contained in the draft
   [I-D.hares-i2rs-pkt-eca-data-model].

For those who desire more information regarding the logic behind the
I2RS Filter-Based RIB, please see the Informational Model at:
[I-D.kini-i2rs-fb-rib-info-model].

4.1. Top Level Yang Structure for ietf-fb-rib

The Top-level Yang structure for a global FB-RIB types (similar to
acl) is not defined for filter-based RIBS. The I2RS Filter-Based RIB
should be defined under this structure under a routing instance. The
three things under this RIB would be: configured Filter-Based RIB
(aka Policy routing), I2RS reboot Ephemeral Filter-Based RIB, and BGP
Flow Specification's Filter-Based RIB. All of these RIBs have
similar actions.

There are two types top-level structures for ietf-fb-ribs: config and
operational state.

The Top-level Yang structure for a global configuration of Filter-
Based RIBs are:

Augments rt:logical-network-elements:\
        :logical-network-element:network-instances: \
            network-instance

```
ietf-fb-rib module
  +--rw ietf-fb-rib
     +--rw default-instance-name string
     +--rw default-router-id rt:router-id
     +--rw config-fb-ribs
           if-feature "config-filter-based-RIB";
        uses fb-ribs;
     +--rw i2rs-fb-ribs
                 if-feature "I2RS-filter-based-RIB";
                 uses fb-rib-t:fb-ribs;
     +--rw bgp-fs-fb-ribs
                 if-feature "BGP-FS-filter-based-RIB";
                 uses fb-rib-t:fb-ribs;
```

        Figure 5: configuration state

The Top-level Yang structure for a global operational state of
Filter-Based RIBs are:

```
   Augments rt:logical-network-elements:\
           :logical-network-element:network-instances: \
              network-instance

   ietf-fb-rib module
     +--rw ietf-fb-rib-opstate
       +--rw default-instance-name string
       +--rw default-router-id rt:router-id
           +--rw config-fb-rib-opstate
                   if-feature "config-filter-based-RIB";
                   uses fb-rib-t:fb-ribs-oper-status;
           +--rw i2rs-fb-rib-opstate {
                   if-feature "I2RS-filter-based-RIB";
                   uses fb-rib-t:fb-ribs-oper-status;
           +--rw bgp-fs-fb-rib-opstate
                   if-feature "BGP-FS-filter-based-RIB";
                   uses fb-rib-t:fb-ribs-oper-status;
```

              Figure 5: operational state

4.2. Filter-Based RIB structures

   The Top-level yang structures at the Filter-Based RIB level have two
   types: configuration and operational state.

   The Top-level Yang structure for the FB-RIB types is:

```
   module: fb-rib-types:
   +--rw fb-ribs
      +--rw fb-rib* [rib-name]
      |  +--rw rib-name string
      |  |  rw fb-type identityref / ephemeral or not
      |  +--rw rib-afi rt:address-family
      |  +--rw fb-rib-intf* [name]
      |  |  +--rw name string
      |  |  +--rw intf if:interface
      |  +--rw default-rib
      |  |  +--rw rt-rib rt:routing:routing-instance:name
      |  |  +--rw config-rib string;  // config rib name
      |  |  +--rw i2rs-rib:routing-instance:name
      |  |  +--rw i2rs-rib string;   //ephemeral rib name
      |  |  +--rw bgp-instance-name string
      |  |  +--rw bgp-rib  string    //session ephemeral
      |  +--rw fb-rib-refs
      |  |  +--rw fb-rib-update-ref uint32 /count of writes
      |  +--rw instance-using*
      |  |   device:networking-instance:networking-instance-name
      |  +--use pkt-eca:pkt-eca-policy-set
```

                Figure 6: FB RIB Type Structure

   HIgh Level Yang

```
   +--rw fb-ribs-oper-status
      +--rw fb-rib-oper-status* [fb-rib-name]
            uses pkt-eca:pkt-eca-opstate
```

5. yang models

5.1. Filter-Based RIB types

```
<CODE BEGINS> file "ietf-fb-rib-types@2016-01-26.yang"
 module ietf-fb-rib-types {

   yang-version "1";

   // namespace
     namespace "urn:ietf:params:xml:ns:yang:ietf-fb-rib-types";
     prefix "fb-rib-t";
         import ietf-interfaces {prefix "if";}
         import ietf-access-control-list {prefix "acl";}
         import ietf-routing {prefix "rt";}
         import ietf-pkt-eca-policy {prefix "pkt-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 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 "2016-01-26" {
     description
        "Filter-Based RIB protocol ";
        reference "draft-hares-i2rs-fb-rib-data-model-01";
  }

    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 {
                   type uint16;
                   description "order identifier";
            }
            leaf statement-oper_status {
                    type fb-rules-status;
                    description "status of rule";
```

```
                }
               description "filter-rib
                 policy rule order and status";
         }

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

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

     grouping default-fb-rib {

           leaf default-rib {
                 type string;
                 description "default ribs for
                   normal and ephemeral filter-based rib
                   should use rt:routing:routing-instance:name";
           }

           leaf i2rs-instance {
                 type string;
              description "default I2RS RIB
                   should use
            i2rs-rib:routing-instance:name";
            }
            leaf rib-name {
```

```
                    type string;
               description "name of RIB";
               }

               leaf fb-rib-update-ref {
                     type uint64;
                     description " number of
                       updates to this FB RIB
                    since last reboot";
                  }
           description "I2RS RIB which will be used
                 even if none of the policy match";
     }

     grouping fb-ribs {
         list fib-rib {
                key fb-rib-name;
                leaf fb-rib-name {
                     type string;
                             mandatory true;
                     description "RIB name";
          }
               uses rt:address-family;
            list fb-rib-intf {
                     key "name";
                     leaf name {
                          type if:interface-ref;
                       description
                         "A reference to the name of a
                            configured network layer
                         interface.";
                      }
                     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.";
               }
                     uses default-fb-rib;
               list instance-using {
                     key instance-name;
                     leaf instance-name {
                       type string;
```

```
                        description
                            " name of instance using this fb-rib
                             rt:routing-instance";
                       }
                  description "instances using
                   this fb-rib";
                }
                list fb-rib-group {
                   key group-name;
               leaf group-name {
                    type string;
                    description "policy-group name";
                }
                 leaf fb-group-policy-type {
                  type fb-rib-policy-type-def;
                  description "Policy type (acl/eca)";
                 }
               list acl-group {
                   key name;
         leaf name {
                        type string;
                        description "name of access list group";
                      }
                     list acls {
                       key fb-acl-name;
                       leaf fb-acl-name {
                         type acl:access-control-list-ref;
                         description "acl list name
                                associated with FB-RIB list";
                       }
                       leaf fb-acl-type {
                             type acl:acl-type;
                             description "acl type";
                       }
          description "list of acls";
                      }
         description "acl group entry";
                    }
                    list pkt-eca-rules {
                      key name;
                      leaf eca-group-name {
                         type pkt-eca:pkt-eca-rule-ref;
                         description "ECA rule";
                          }
                           description "list of eca groups";
                   }
                   uses fb-rib-group-order-status;
                   description "list of ordered policy groups ";
```

```
                   }
                 description  "Configuration of
                     an filter-based rib list";
             }
         description "fb-rib group";
     }
  }
}
<CODE ENDS>
```

5.2. FB-RIB

```
   <CODE BEGINS> file "ietf-fb-rib@2016-02-09.yang"
   module ietf-fb-rib {
     yang-version "1";

     // namespace
     namespace "urn:ietf:params:xml:ns:yang:ietf-fb-rib";
     // replace with iana namespace when assigned
       prefix "fb-rib";


     // import some basic inet types
     import ietf-yang-types {prefix "yang";}
     import ietf-fb-rib-types { prefix "fb-rib-t";}

     // meta
     organization
       "IETF";

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

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

        revision "2016-02-09" {
          description "initial revision";
          reference "draft-hares-i2rs-fb-rib-data-model-01";
```

```
      }

         feature config-filter-based-RIB {
      description
        "This feature means that a node support
         config filter-based rib.";
      }
         feature I2RS-filter-based-RIB {
      description
        "This feature means that a node support
         I2RS filter-based rib.";
      }
         feature BGP-FS-filter-based-RIB {
      description
        "This feature means that a node support
        BGP FS filter-based rib.";
      }

         container ietf-fb-rib {
           presence "top-level structure for
             configuration";
        leaf default-instance-name {
             type string;
                 mandatory true;
          description
            "A routing instance is identified by its name,
             INSTANCE_name.  This MUST be unique across all routing
             instances in a given network device.";
          }
             leaf default-router-id {
                   type yang:dotted-quad;
                   description "Default router id";
                 }
               container config-fb-rib {
                if-feature config-filter-based-RIB;
                 uses fb-rib-t:fb-ribs;
                description "config filter-based RIB";
               }

               container i2rs-fb-rib {
                 if-feature I2RS-filter-based-RIB;
                 uses fb-rib-t:fb-ribs;
                description "bgp-fs filter-based RIB";
               }
               container bgp-fs-fb-rib {
                 if-feature BGP-FS-filter-based-RIB;
                 uses fb-rib-t:fb-ribs;
                 description "bgp fs filter-based RIB
```

```
                          of filter-based routes recieved from peers.";
                      }
                 description "fb-rib augments routing instance";
              }

            container ietf-fb-rib-opstate {
               presence "top-level structure for
                op-state";
               config "false";
            leaf default-instance-name {
                 type string;
                    mandatory true;
             description
               "A routing instance is identified by its name,
                INSTANCE_name.  This MUST be unique across all routing
                instances in a given network device.";
              }
                  leaf default-router-id {
                       type yang:dotted-quad;
                       description "Default router id";
                  }
                  container config-fb-rib-opstate {
                    if-feature config-filter-based-RIB;
                    uses fb-rib-t:fb-ribs-oper-status;
                   description "config filter-based RIB";
                  }
                  container i2rs-fb-rib-opstate {
                    if-feature I2RS-filter-based-RIB;
                    uses fb-rib-t:fb-ribs-oper-status;
                   description "i2rs filter-based RIB";
                  }
                  container bgp-fs-fb-rib-opstate {
                    if-feature BGP-FS-filter-based-RIB;
                     uses fb-rib-t:fb-ribs-oper-status;
                   description "bgp fs filter-based RIB";
                  }
              description "fb-rib augments routing instance";
            }
      }

      <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.acee-rtgwg-yang-rib-extend]
              Lindem, A. and Y. Qu, "YANG Data Model for RIB
              Extensions", draft-acee-rtgwg-yang-rib-extend-00 (work in
              progress), October 2015.

   [I-D.hares-i2rs-fb-rib-data-model]
              Hares, S., Kini, S., Dunbar, L., Krishnan, R., Bogdanovic,
              D., and R. White, "Filter-Based RIB Data Model", draft-
              hares-i2rs-fb-rib-data-model-02 (work in progress),
              February 2016.

   [I-D.hares-i2rs-pkt-eca-data-model]
              Hares, S., Wu, Q., and R. White, "Filter-Based Packet
              Forwarding ECA Policy", draft-hares-i2rs-pkt-eca-data-
              model-02 (work in progress), February 2016.

   [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-05 (work in progress),
              March 2016.

   [I-D.ietf-netmod-routing-cfg]
              Lhotka, L. and A. Lindem, "A YANG Data Model for Routing
              Management", draft-ietf-netmod-routing-cfg-21 (work in
              progress), March 2016.

   [I-D.wu-idr-flowspec-yang-cfg]
              Wu, N., Zhuang, S., and A. Choudhary, "A YANG Data Model
              for Flow Specification", draft-wu-idr-flowspec-yang-cfg-02
              (work in progress), October 2015.

8.2. Informative References

   [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-13 (work in
              progress), February 2016.

   [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-08 (work
              in progress), October 2015.

   [I-D.ietf-i2rs-usecase-reqs-summary]
              Hares, S. and M. Chen, "Summary of I2RS Use Case
              Requirements", draft-ietf-i2rs-usecase-reqs-summary-02
              (work in progress), March 2016.

   [I-D.ietf-netmod-acl-model]
              Bogdanovic, D., Koushik, K., Huang, L., and D. Blair,
              "Network Access Control List (ACL) YANG Data Model",
              draft-ietf-netmod-acl-model-07 (work in progress), March
              2016.

   [I-D.kini-i2rs-fb-rib-info-model]
              Kini, S., Hares, S., Dunbar, L., Ghanwani, A., Krishnan,
              R., Bogdanovic, D., and R. White, "Filter-Based RIB
              Information Model", draft-kini-i2rs-fb-rib-info-model-03
              (work in progress), February 2016.

   [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

Russ White
LinkedIn

Email: russ@riw.us