

YANG Data Model for RIP

draft-liu-rtgwg-yang-rip-01

Xufeng Liu (Ericsson)

Prateek Sarda (Ericsson)

Vikram Choudhary (Huawei)

Scope

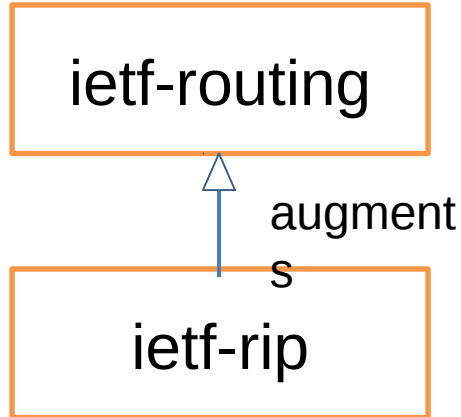
- YANG Data Model for configuring and monitoring RIP (Routing Information Protocol)
 - RIP version 2 (RFC2453) for IPv4
 - RIPng (RFC2080) for IPv6
- Covers configuration, operational states, and actions.

Goals

- Keep simple
- Avoid vendor specific features
- Support vendor extensions
- Be consistent with other protocol models

Relationship with Other Modules

- Augments ietf-routing (draft-ietf-netmod-routing-cfg)



Augmentation

- Each RIP instance is modeled as an instance of routing-protocol with type “ripv2” or “ripng”.
- Model supports multiple RIP instances in a routing-instance.

```
module: ietf-routing
  +--ro routing-state
  |   +--ro routing-instance* [name]
  |       +--ro routing-protocols
  |           +--ro routing-protocol* [name]
  |               +--ro name                string
  |               +--ro type                identityref
  |               +--ro rip:rip
  +--rw routing
      +--rw routing-instance* [name]
          +--rw routing-protocols
              +--rw routing-protocol* [name]
                  +--rw name                string
                  +--rw description?       string
                  +--rw enabled?           boolean
                  +--rw type                identityref
                  +--rw rip:rip
```

Instance Configuration

- Model covers RIP instance level attributes
 - Timers, route characteristic parameters, and redistribution option.

```
augment /rt:routing/rt:routing-instance/rt:routing-protocols/rt:routing-protocol:
```

```
  +--rw rip
    +--rw originate-default-route!
      | +--rw route-map?  string
    +--rw default-metric?          uint8
    +--rw distance?               uint8
    +--rw flash-update-threshold? uint8
    +--rw maximum-paths?          uint8
    +--rw output-delay?           uint8
    +--rw distribute-list* [prefix-list-name direction]
      | +--rw prefix-list-name  string
      | +--rw direction         enumeration
      | +--rw if-name?          if:interface-ref
    +--rw timers
      | +--rw update-interval?   uint16
      | +--rw invalid-interval?  uint16
      | +--rw holddown-interval? uint16
      | +--rw flush-interval?    uint16
```

Interface Configuration

- Model covers RIP interface level attributes
- Timers, route characteristic parameters, neighbor specification, split-horizon option, and authentication reference.

```
+--rw rip
  +--rw interface* [interface]
    +--rw interface          if:interface-ref
    +--rw authentication
    +--rw bfd?               boolean {bfd}?
    +--rw cost?             uint8
    +--rw neighbors {neighbor-configuration}?
      | +--rw neighbor* [address]
      |   +--rw address      inet:ip-address
    +--rw no-listen?        empty
    +--rw no-supply?        empty
    +--rw originate-default-route!
      | +--rw route-map?    string
    +--rw split-horizon?    enumeration
    +--rw summary-address
      | +--rw address?      inet:ip-prefix
      | +--rw metric?       uint8
    +--rw timers
```

Instance States

- Model covers RIP instance level state attributes

```
augment /rt:routing-state/rt:routing-instance/rt:routing-protocols/  
rt:routing-protocol:
```

```
  +--ro rip
```

```
    +--ro originate-default-route!
```

```
      | +--ro route-map?  string
```

```
    +--ro default-metric?          uint8
```

```
    +--ro distance?                uint8
```

```
    +--ro flash-update-threshold?  uint8
```

```
    +--ro maximum-paths?           uint8
```

```
    +--ro output-delay?            uint8
```

```
    +--ro distribute-list* [prefix-list-name direction]
```

```
      | +--ro prefix-list-name  string
```

```
      | +--ro direction         enumeration
```

```
      | +--ro if-name?          if:interface-ref
```

```
    +--ro timers
```

```
      ... ..
```

```
    +--ro next-flash-update?        uint32
```

```
    +--ro num-of-routes?            uint32
```

```
      ... ..
```


Instance Statistics

- Model covers RIP instance level statistic data

```
augment /rt:routing-state/rt:routing-instance/rt:routing-protocols
/rt:routing-protocol:
  +--ro rip
    +--ro statistics {global-statistics}?
      +--ro discontinuity-time?   yang:date-and-time
      +--ro requests-rcvd?       yang:counter32
      +--ro requests-sent?       yang:counter32
      +--ro responses-rcvd?      yang:counter32
      +--ro responses-sent?      yang:counter32
```

Neighbor States

- Model captures neighbor state attributes

```
augment /rt:routing-state/rt:routing-instance/rt:routing-protocols  
/rt:routing-protocol:
```

```
  +--ro rip
```

```
    +--ro ipv4
```

```
      | +--ro neighbors
```

```
      |   +--ro neighbor* [ipv4-address]
```

```
      |     +--ro ipv4-address      inet:ipv4-address
```

```
      |     +--ro last-update?     yang:date-and-time
```

```
      |     +--ro bad-packets-rcvd? yang:counter32
```

```
      |     +--ro bad-routes-rcvd? yang:counter32
```

```
    +--ro ipv6
```

```
      +--ro neighbors
```

```
        +--ro neighbor* [ipv6-address]
```

```
          +--ro ipv6-address      inet:ipv6-address
```

```
          +--ro last-update?     yang:date-and-time
```

```
          +--ro bad-packets-rcvd? yang:counter32
```

```
          +--ro bad-routes-rcvd? yang:counter32
```

Route States

- Model captures route state attributes

```
augment /rt:routing-state/rt:routing-instance/rt:routing-protocols
/rt:routing-protocol:
```

```
  +--ro rip
```

```
    +--ro ipv4
```

```
      | +--ro routes
```

```
      |   +--ro route* [ipv4-prefix]
```

```
      |     +--ro ipv4-prefix
```

```
inet:ipv4-prefix
```

```
      |     +--ro next-hop?
```

```
inet:ipv4-address
```

```
      |     +--ro interface?
```

```
if:interface-ref
```

```
      |     +--ro redistributed?
```

```
boolean
```

```
      |     +--ro route-type?
```

```
enumeration
```

```
      |     +--ro metric?
```

```
uint8
```

```
      |     +--ro expire-time?
```

```
uint16
```

```
      |     +--ro deleted?
```

```
boolean
```

```
      |     +--ro holddown?
```

```
boolean
```

```
      |     +--ro need-flash?
```

```
boolean
```

```
      |     +--ro need-download-to-rib?
```

```
boolean
```

```
      |     +--ro inactive?
```

```
boolean
```

```
      |     +--ro next-hop-flags?
```

```
bits
```

Interface States

- Model covers interface state data, including interface level statistics.

```
+--ro interface* [interface]
  | +--ro interface                if:interface-ref
  | +--ro oper-status?             enumeration
  | +--ro cost?                    uint8
  | +--ro listen?                  boolean
  | +--ro next-full-update?        uint32
  | +--ro originate-default-route? boolean
  | +--ro poison-reverse?         boolean
  | +--ro split-horizon?          boolean
  | +--ro supply?                  boolean
  | +--ro valid-address?           boolean
  | +--ro timers
  |     ... ..
  | +--ro statistics {interface-statistics}?
  |   +--ro discontinuity-time?    yang:date-and-time
  |   +--ro bad-packets-rcvd?     yang:counter32
  |   +--ro bad-routes-rcvd?     yang:counter32
  |   +--ro updates-sent?         yang:counter32
```

Actions

- Model specifies actions that can be applied to a RIP instance.

rpcs:

+---x clear-rip-route

+--ro input

+--ro instance-name? leafref

Next Steps

- Align with other models, including BFD, routing policy, and authentication.
- Solicit comments
- WG adoption