A YANG Data Model for Routing Management

draft-ietf-netmod-routing-cfg-11

Ladislav Lhotka
⟨lhotka@nic.cz⟩

6 November 2013
Changes since -10

Minutes from IETF 87: “Lada will work with authors of the I2RS working group to harmonize things with the information model defined by the I2RS working group ...”

- terminology changes,
- address family info,
- keys for state lists, relationship between system-controlled and user controlled list entries,
- new options for nexthop specification,
- feature advanced-router replaced user-defined-routing-tables.
Terminology

- router → routing instance,
- routing table → RIB.
Identities for Address Families

address-family
  ipv4
    ipv4-unicast
  ipv6
    ipv6-unicast

Two leafs with *enumeration* type were replaced with a single *identityref* leaf:

```
<address-family>ipv4</address-family>
<safi>nlri-unicast</safi>

<address-family>v4ur:ipv4-unicast</address-family>
```

The module *iana-afn-safi* is no more imported.
System- versus User-Controlled Entries

Entries in some operational state lists (config false):

**system-controlled entry**

Created by the system and assigned a unique numerical id (list key); cannot be deleted.

User may provide additional configuration for such an entry by creating an entry in the config list with an arbitrary key \(\text{name}\) and a reference to the system-controlled entry’s id.

**user-controlled entry**

Created and deleted as a direct consequence of creating/deleting an entry in the config list.

The system also assigns an id but the user needn’t use it.
Example

1. A system-controlled entry is created automatically:

```xml
<rt:routing-state>
  <rt:routing-instance>
    <rt:id>1415926535</rt:id>
    <rt:router-id>192.0.2.1</rt:router-id>
    ...
  </rt:routing-instance>
</rt:routing-state>
```

2. User adds some configuration (changes router-id):

```xml
<rt:routing>
  <rt:routing-instance>
    <rt:name>rtr0</rt:name>
    <rt:routing-instance-id>1415926535</rt:routing-instance-id>
    <rt:description>Router A</rt:description>
    <rt:router-id>192.0.4.2</rt:router-id>
    ...
  </rt:routing-instance>
</rt:routing>
```
Final result:

```xml
<rt:routing-state>
  <rt:routing-instance>
    <rt:id>1415926535</rt:id>
    <rt:name>rtr0</rt:name>
    <rt:router-id>192.0.4.2</rt:router-id>
  </rt:routing-instance>
  ...
</rt:routing-state>

<rt:routing>
  <rt:routing-instance>
    <rt:name>rtr0</rt:name>
    <rt:routing-instance-id>1415926535</rt:routing-instance-id>
    <rt:description>Router A</rt:description>
  </rt:routing-instance>
  ...
</rt:routing>
```
Nexthop Options

- special nexthops

```cpp
    case special-nexthop {
        leaf special-nexthop {
            type enumeration {
                enum blackhole; silently discard
                enum unreachable; discard & notify
                enum prohibit;
                enum receive; receive locally
            }
        }
    }
```
• simple nexthop (for IPv4 routes)

case simple-nexthop {
    leaf outgoing-interface {
        type leafref {
            path "/routing-state/routing-instance/"
            + "interfaces/interface/name";
        }
    }
    leaf gateway {
        type inet:ipv4-address;
    }
}
- **nexthop list (for IPv6 routes)**

  ```
  +--:(nexthop-list) {advanced-router}?
    +--ro nexthop* [id]
    +--ro id uint64
    +--ro outgoing-interface? leafref
    +--ro priority? enumeration
    +--ro weight? uint8
    +--ro v6ur:address? inet:ipv6-address
  ```
Feature advanced-router

Replaces and extends feature user-defined-routing-tables.
Includes:

- user-defined routing tables + related framework,
- multi-path routes (nexthop list).
Open Issues

- Instead of advanced-router, more specific features may be useful.

  Proposal: two features
  - multiple-ribs,
  - multipath-routes.
Conclusions

- Minor update (features), then WGLC.
- I-D draft-ietf-netmod-iana-afn-safi-00 can be dropped.
- I2RS WG can take this data model as a starting point and augment it with additional parameters.
- Individual submissions defining data models for OSPF and BGP are not compatible with the core routing data model:
  - draft-yeung-netmod-ospf-00,
  - draft-zhdankin-netmod-bgp-cfg-00.