(RPS) Routing Policy System Working Group

Chairs:

Cengiz Alaettinoglu <cengiz@isi.edu>

Daniel Karrenberg <Daniel.Karrenberg@ripe.net>

Mailing List: `rps@isi.edu`

To subscribe: `rps-request@isi.edu`

Ftp archive: `ftp://ftp.isi.edu/rps`

Motivation: Extensions to Ripe-181

Routing policy expression extensions:

- **as-path** regular expressions

  `accept AS1 AND NOT <.* AS2 AS3 .*>`

- **next-hop** attribute

  `accept AS1 AND (next-hop == 128.8.18.8)`

- other attributes (color, IDRP attributes, etc.)

Other extensions

History

PRDB	NSFNet centric
Ripe 60	
Ripe 81	Simple but need more functionality
Ripe 181	Still need more functionality
RPSL	

Charter

Routing Policy Specification Language

Distributed Registry Model

Tools

Routing Policy Specification Language RPSL

express **preference policies** (on import)

express **access restriction policies** (on export)

express **other policies** (e.g. multicast?)

Level of Detail in RPSL

Models the policy decision process

e.g. if decision depends on variable/attribute *X*

RPSL should model *X*

Abstract router configuration

e.g. capability *X* in config file

abstract capability *X* in RPSL

RPSL Goals: Simplicity and Expressibility

Most domains have simple policies

simple to use **for most domains**

Some domains have complicated policies

expressible but **not so simple** for them

RPSL Goals: Protocol Independent

Inter-domain routing protocols:

BGP, IDRP, ERP, ...

Other protocols

multicasting, resource reservation, ...

RPSL Goals: Router Vendor Independent

Not a router configuration language

Can be translated into router configuration languages

RPSL Goals: Support for Tools

Config file generation from RPSL specification

RtConfig

Analysis and **diagnosis** tools

prpath, prtraceroute, prcheck, peval, aggchk, ...

Tools are tools:

no restrictions on RPSL to support a tool

RPSL Goals: Extensibility

new policy mechanisms (e.g. colors)

new protocols

RPSL Goals: Compatibility with Ripe-181

Through **Ripe-181 to RPSL** converter

Why not Ripe-181?

Need more functionality today

Ripe-181 is not extensible

Will need even more functionality

Need an extensible language

RPSL Technical Details

Cengiz Alaettinoglu

RPSL Policy Terms

```
on: <policy-event-specifier>
    matching <policy-filter-specifier>
    do <policy-action>
```

```
as-in: from AS1 5 accept {128.8.0.0/16}
on: import from AS1
    matching {128.8.0.0/16}
    do pref = 5
```


Policy Events (on ...)

```
import from domain-id local-interface remote-interface
export to   domain-id local-interface remote-interface
```

domain-id	**local-/remote-interface**
AS number	ipv4 address
RDI	ipv6 address
*	*

Other events:

`aggregate, reserve, ...`

Interaction between Events

```
import *   *            *
import AS1 *            *
import AS1 128.8.128.8  *
import AS1 *            128.8.128.9
import AS1 128.8.128.8  128.8.128.9
```

Least specific

Most specific

Possible interactions:

- more specific **overrides** less specific
- more specific **complements** less specific
- **ripe-181** interaction

Interaction between Same Type Events (cont)

```
on import AS1 * * matching AS1 AS2 do ...
on import AS1 8.8.8.8 8.8.8.9 matching AS1 do ...
```

peering	overrides	complements	ripe-181
8.8.8.8-9	AS1	AS1 AS2	AS1 AS2
9.9.9.8-9	AS1 AS2	AS1 AS2	AS2

Interaction between Same Type Events (cont)

```
on import AS1 * * matching AS1 AS2 do ...
on import AS1 8.8.8.8 8.8.8.9 matching AS3 do ...
```

peering	overrides	complements	ripe-181
8.8.8.8-9	AS3	AS1 AS2 AS3	error
9.9.9.8-9	AS1 AS2	AS1 AS2	error

same expressibility power

overrides and **complements** are more intuitive

overrides is simplest to implement

Policy Filters (matching ...)

	Filter term	Routing protocol attribute
Ripe-181	address-prefix	NLRI
	AS number	NLRI ~~or AS_PATH (ambiguous)~~
	communities	NLRI
	AS macros	NLRI ~~or AS_PATH~~
Extensions	AS path regexp	AS_PATH
	next-hop = address	NEXT_HOP
	color = integer	COLOR
	confederation	RD_PATH
	...	any attribute

Policy Action (do ...)

```
do pref = 5
do MED = 5
do MED = IGP

do color = 6
do MED = 4 + IGP, AS_EXCLUDE_LIST = {AS1, AS2}
do pref = AS_PATH_LENGTH, pref = pref + 1
do pref = 1, pref2 = 5 + MED
```


Support for Arbitrary Attributes/Events

Needs to be **protocol independent**

Needs to be router **configuration format independent**

Needs to be **extensible**

dictionary object

for specifying rpsl-attributes

and events

Dictionary Object

```
rpsl-attribute: <name> <type> <default-value> <comment>

dictionary: IRR
rpsl-attribute: MED  int 0 "multi-exit discrimination"
rpsl-attribute: color int 0 "color attribute"
rpsl-attribute: pref  int 0 "preference"
rpsl-attribute: AS_PATH_LENGTH int 1 "length of
                                       as_path"
rpsl-attribute: ...
rpsl-event: import ...
rpsl-event: export ...
rpsl-event: ...
```


Dictionary Object for Ripe-181 Attributes

```
dictionary: IRR
rpsl-attribute: cost       int 0 "cost"
rpsl-attribute: pref       int 0 "preference"
rpsl-attribute: metric-out int 0 "multi-exit disc."
rpsl-attribute: IGP        int 0 "IGP cost"
```

Demonstration Using `color` Attribute

```
dictionary: IRR
rpsl-attribute: pref int 0 "preference"
rpsl-attribute: color int 0 "color"

aut-num: AS4
on: import AS1 * * matching AS2 do pref=1, color=6
on: export AS3 * * matching color == 6 do ;
```


Macros

Global level:

```
macro-object: name
expands: arbitrary-string
```

Object level:

```
aut-num: AS1
macro: name arbitrary-string
```

Macro use:

```
aut-num: AS-690
macro: ENSS28 128.8.8.8 128.8.8.9
macro: GOOD-GUYS AS1 AS2 AS3 OR {7.7.7.0/24}
on: import AS1 $ENSS28 matching $GOOD-GUYS do ...
```


Macros vs Communities

```
community: COMA
descr: Community A
mnt-by: MAINT-1
```

```
route: 8.8.8.0/24
comm-list: COMA
mnt-by: MAINT-2
```

```
route: 9.9.9.0/24
comm-list: COMA
mnt-by: MAINT-3
```

```
macro-object: COMA
descr: Community A
expands: {8.8.8.0/24, 9.9.9.0/24}
mnt-by: MAINT-1
```

Communities are still needed.

Scope Rules

```
macro-object: foo
expands: global definition

aut-num: AS1
macro: foo AS1's definition

aut-num: AS2
macro: foo AS2's definition
on:... $foo ...
```

`$foo`	this object's definition, if none then global definition
`$AS2::foo`	AS2's definition
`$::foo`	global definition

More specifics in network lists

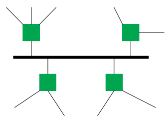
8.8.0.0/16	matches only 8.8.0.0/16
8.8.0.0/16+	matches 8.8.0.0/16 and more specifics

IDRP Confederations

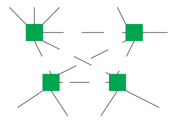
Confederation object:

```
confederation: confederation-id
sub-confederations: confederation-id, ...
parent-confederations: confederation-id, ...
```

Specifying Physical Topology



Physical topology

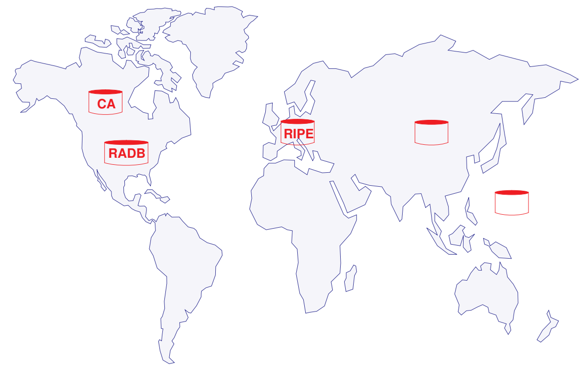


Peering topology

Some tools require the knowledge of physical topology.

E.g. **prpath**

Distributed Registry Model
Internet Routing Registry (IRR)



Internet Routing Registry

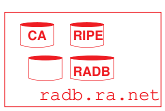
Distributed Registry

An object is registered to one or more sites

```
aut-num: AS1          aut-num: AS1
...                   ...
source: RADB          source: RIPE
```

Centralized Access

IRR sites keep local copies of each other's data



Synchronization and Atomicity

True synchronization and atomicity

- **two-phase commit** protocol
- **quorum** protocols for internetwork partitioning
- probably not necessary

Synchronization at time instances

- a copy of all database as of 2:00pm today
- often desired for router re-configurations

Synchronization

Ftp'able copies are not adequate and do not scale

History mechanism

source	serial-no	timestamp	operation	object
radb	1	040295090000	delete	*an=AS1
radb	2	040295090100	add	...

Ftp'able history diff files

Email based automatic updates

Atomicity
via Master Registry

If an object is in **one** database

- atomic update to **one master registry** is easy

If an object is in **two or more** databases

- treat the objects as separate
- **two separate updates** to two registries
- two copies may be different
  - need **scope rules**

Scope Rules

```
aut-num: AS1            aut-num: AS1
...                     ...
source: RADB            source: RIPE
```

Tools supply a **search order: RADB,RIPE**

Scope mechanisms:

`RIPE::AS1`

`RADB::NSF_AUP`

```
route: 8.8.8.0/24
comm-list: RADB::NSF_AUP
source: RIPE
```


Predefined Macros

`$ROUTES`

set of routes defined in this database

`RADB::$ROUTES, RIPE::$ROUTES, IRR::$ROUTES`

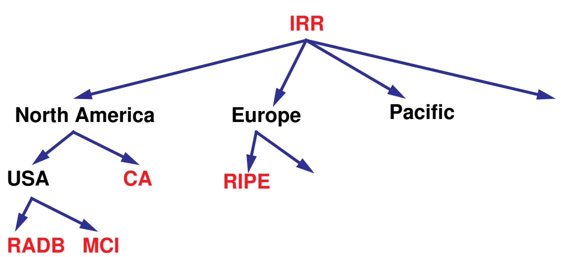
`$ASNS`

set of AS numbers defined in this database

`RADB::$ASNS, IRR::$ASNS`

Hierarchical Registry Organization

Flat organization does not scale up



Tools

Merit Tools

Pride Tools

RAToolSet

Tools and RPSL

Merit Tools

`aggis`

classful to classless converter

classless to classful converter

`aggchk`

list potential aggregates

report generators

Pride Tools

`prpath`

list all potential paths between two places

`prtraceroute`

compare actual routing to the registered policies

`prcheck`

consistency-check of policies of neighboring domains

RAToolSet

radbserver

extended whois server with optimized queries

peval

policy evaluator

inputs a policy expression

outputs another with possible expansions & evaluations

RtConfig

gated configuration generator

cisco support by CANet

Tools and RPSL

Attributes

RPSL does not attach any semantics to attributes

Tools need the semantics to operate

more intelligence into tools

more protocol knowledge into tools

Events

Tools already know the semantics of events

RAToolSet Framework