PeeringDB and Internet Routing Registry

Michael Lambert
Internet2 Routing Security Workshop
5 March 2019

Outline

- Importance of publishing contact and routing information
- PeeringDB
  - Overview
  - Walkthrough
- Internet Routing Registry
  - Overview
  - Diversion: Background on RPSL
  - Using it (well, at least one registry)
  - Walkthrough

Why is this important?

- Need to be reachable by other operators
  - Operational issues
    - Security (scans, DDoS, etc)
    - Routing (leakage, unexpected paths, etc)
  - Peering opportunities
    - See who else is at/near your PoPs
- Campus: most benefit from IRR
- Regionals: benefit from both PeeringDB and IRR
- [Ignore additional exposure surface for salescritters]

PeeringDB

What is PeeringDB?

- According to tagline, “The Interconnection Database”
- Freely-available, community-maintained database of
  - Networks (organizations and ASNs)
  - Exchanges (public peering points)
  - Facilities (including private peering points)
- Sponsored
  - Microsoft, Amazon @ $25,000/yr
  - Down to $2,500/yr
- Source code on GitHub

Using PeeringDB

- Create a user account
- Associate with (or create) one or more networks
- Add contacts and policies
- Associate with (or create) facilities and exchanges
- https://www.peeringdb.com/

Internet Routing Registry

What is the IRR?

- Distributed database of routing information
- Implements RPSL (Routing Policy Specification Language)
- Used by many networks for generating router configurations

How is the IRR “distributed”?

- Many RPSL-based databases worldwide
  - Regional Internet Registries and country-based registries
    - ARIN, RIPE, APNIC, AFRINIC, LACNIC, CANARIE, JPIRR, etc
  - Provider-based registries
    - Level3/CenturyLink, Rogers, etc
  - General-purpose registries
    - RADB, ALTDB, maybe others
- Each registry mirrors zero or more of the others
  - Generally, very frequent updates
- Each can be queried by whois or specialized tools

What is RPSL?

- Object-oriented language for specifying routing policy
- Defined in several RFCs; most important ones follow
  - RFC 2622: Routing Policy Specification Language
    - Core document; to quote www.irr.net: “This document should only require a couple of read-throughs for the average user.” Only 69 pages.
  - RFC 2650: Using RPSL in Practice
    - Tutorial with examples
  - RFC 4012: Routing Policy Specification Language next generation
    - RPSLng; adds support for IPv6, multicast

Types of RPSL objects

- mntner
  - The maintainer object
  - Top-level object for an entity; basically its owner
  - Created by registry when account is set up
  - All other objects reference it
- person, role
  - A contact with phone numbers, addresses, etc
  - Can be individual or group account (*eg*, NOC)

Types of RPSL objects

- route, route6
  - Object describing an IPv4 or IPv6 route
  - Has associated origin AS
- aut-num
  - Describes an autonomous system
  - May contain routing policy for the AS
- as-set, route-set
  - Collections of autonomous systems or routes
- Many more...

Using RADB

- Why RADB?
  - Because it's the one I have experience with (we've been using it for the better part of 20 years)
  - Would personally steer away from provider-based registries
  - Others (ARIN, for example) perfectly usable
- Why not RADB?
  - Annual fee (discount for non-profits)

RADB, an aside…

Originally stood for Routing Arbiter Database, part of the NSF-funded Route Arbiter project at Merit; once funding went away (and charging for sustainability became necessary), rebranded Routing Assets Database

RADB account creation

- First step, create an account
  - Visit www.radb.net; click on Register button
  - Walk through brief forms for user, organization, maintainer
  - Last page is confirmation
  - Click, then: Stop. Pay Bill.
- Maintainer record created
  - Receive/create password
  - Can start adding other objects

Manipulating RADB data

- Web interface
  - Point and shoot
  - Relies on crypt-pw password for authentication
  - Perhaps not every record type supported
- Email templates
  - Easily scriptable
  - Can use OpenPGP signing for authentication
  - Full support for all RPSL object types

My recommendation

- YMMV
- Hashed password formerly visible in whois
  - Seems to have been addressed (at least for RADB)
- Use password authentication to add two or more OpenPGP keys
  - Yes, at least two
- Delete CRYPT-PW authentication
- Just use signed email for updates

Walkthrough of RADB web interface

www.radb.net