

IRR Working Group--Notes on Routing Registries

04-Feb-2020

This note was produced by the Internet2 IRR working group with specific input from Ventsi Gotov and Theo Lavis and editing by Michael Lambert.

Motivation for registering in IRR

There are several factors that may influence the motivations to register in the IRR:

- Prevention from getting traffic filtered by peers that are using IRR data to generate filters. This has already started to occur with several carriers including Hurricane Electric, and has been announced to be forthcoming from Google.
- Providing a documented reference for one's own network routing policies that may be used to generate one's own network prefix-lists automatically, or as a check against active routing configurations for compliance reviews. In principle, it is possible for one to include in the IRR all information needed for the generation of one's routing configurations. Security implications are not discussed here...
- Documenting a clear routing policy as a component of participation in MANRS
- Greater participation overall enables traffic filtering via IRR records to be a more viable option for all. Wider use of filtering should aid in overall Internet stability, helping to prevent accidental and intentional route hijacking.

Best practices on what to register

While there are several object types that can be registered in the IRR system, there are six that are most critical: *Mntner*, *Aut-num*, *AS-Set*, *Route*, *Route6*, and *Route-Set*. These six items are the minimum that should be registered to prevent issues with carriers filtering traffic.

Authoritative references include RFCs 2622, 2650, 2725 and 4012.

Maintainer (mntner)

Before creating any other object entries, a Maintainer object must be defined. This Maintainer will be used to create every other object in the IRR. Controlling the Maintainer object allows one to control subordinate objects.

Suggested minimum content:

mntner: [mandatory] [single] [primary/lookup key]
descr: [mandatory] [multiple] []
admin-c: [mandatory] [multiple] [inverse key]
auth: [mandatory] [multiple] [inverse key]
mnt-by: [mandatory] [multiple] [inverse key]
referral-by: [mandatory] [single] [inverse key]
changed: [mandatory] [multiple] []
source: [mandatory] [single] []

Aut-num

An Aut-num object should be created for every Autonomous System that you control. ISPs may sometimes (perhaps more often than one would really like) proxy register a record for the Autonomous System of a downstream customer with their consent (and, frankly, often without it), in order to prevent traffic issues due to IRR based filtering.

Suggested minimum content:

aut-num: [mandatory] [single] [primary/lookup key]
as-name: [mandatory] [single] []
descr: [mandatory] [multiple] []
admin-c: [mandatory] [multiple] [inverse key]
tech-c: [mandatory] [multiple] [inverse key]
mnt-by: [mandatory] [multiple] [inverse key]
changed: [mandatory] [multiple] []
source: [mandatory] [single] []

AS-set

An AS-set is just a list of aut-num objects. Generally you should have at least one overall AS-set object that lists all the AS numbers (including your own) that will originate route advertisements which you will advertise to your BGP peers. This can be done as a single list, or through a list that includes references to other AS-set records.

Many providers and peering networks require that the overall, inclusive AS-set be listed in your PeeringDB profile, which makes it easy to find and use for their filtering needs. It has also been suggested that the AS-set should be listed in an *export*: line on the Aut-num record, also to make it easier for others to find.

Naming conventions:

1. All AS-SET names must start with **AS-**
2. Because uniqueness is enforced only within individual IRRs, there is potential for multiple entities to register AS-SET objects that happen to have the same name. This could potentially cause confusion for anyone querying for the AS-SET information.

- a. One emerging best practice to assist in global uniqueness is to scope the AS-SET name by prepending one's AS to the AS_SET name. Many advocate using a hierarchical approach by using a colon as specified in [RFC 2622 section 5 'Set Classes'](#). *Example: **AS1234:AS-CUSTOMERS***
- b. **HOWEVER**, hierarchical naming is not universally supported amongst IRRs. In particular, ARIN's IRR does not support hierarchical naming. To prepend an AS onto ARIN AS-SET record names, an approach must use dashes (*Example: AS-AS1234-NAME*)

Suggested minimum content:

as-set: [mandatory] [single] [primary/lookup key]
descr: [mandatory] [multiple] []
members: [optional] [multiple] []
tech-c: [mandatory] [multiple] [inverse key]
admin-c: [mandatory] [multiple] [inverse key]
mnt-by: [mandatory] [multiple] [inverse key]
changed: [mandatory] [multiple] []
source: [mandatory] [single] []

Route

Route objects should be created for each specific IPv4 prefix that will be advertised to your peers.

Suggested minimum content:

route: [mandatory] [single] [primary/lookup key]
descr: [mandatory] [multiple] []
origin: [mandatory] [single] [primary/inverse key]
mnt-by: [mandatory] [multiple] [inverse key]
changed: [mandatory] [multiple] []
source: [mandatory] [single] []

Route6

Route6 objects should be created for each specific IPv6 prefix that will be advertised to your peers.

Suggested minimum content:

route6: [mandatory] [single] [primary/lookup key]
descr: [mandatory] [multiple] []
origin: [mandatory] [single] [primary/inverse key]
mnt-by: [mandatory] [multiple] [inverse key]
changed: [mandatory] [multiple] []
source: [mandatory] [single] []

Route-set

A route-set is just a list of route objects. Route objects (prefixes) are listed on a *members:* line, and route6 objects are listed on a *mp-members:* line.

Suggested minimum content:

```
route-set: [mandatory] [single] [primary/lookup key]
descr: [mandatory] [multiple] []
members: [optional] [multiple] []
mp-members: [optional] [multiple] []
tech-c: [mandatory] [multiple] [inverse key]
admin-c: [mandatory] [multiple] [inverse key]
mnt-by: [mandatory] [multiple] [inverse key]
changed: [mandatory] [multiple] []
source: [mandatory] [single] []
```

How to use registries

Signaling of IRR entries

There are two common methods that have been identified as ways to notify other entities of key IRR entries relating to a network.

1) PeeringDB

- In addition to the network AS number, PeeringDB provides a field to list the AS-Set and Route-Set names for a network. When Listing these values in PeeringDB, it is recommended that the authoritative IRR be identified. (This is because IRRs do not enforce uniqueness globally, and the AS-Set name may be used by multiple records in various IRRs). There are two methods that have been used to accomplish this.
 - Appending the IRR name to the end of the AS or RS record name using the @ symbol. In the form of **AS-SET@SOURCE**
Examples: AS-CONE@ARIN , AS-CONE@RADB , AS-CONE@Level3
 - Prepending the IRR name to the beginning using two colons (::) in the form of **SOURCE::AS-SET**
Examples: ARIN::AS-CONE , RADB::AS-CONE , Level3::AS-CONE

2) Export lines within the Aut-num object

- Listing the AS-Set in a generic export line in the aut-num object.
Example: Export: to AS-ANY announce AS-CONE

The two major, publicly available registries are RADB, operated by MERIT, and the ARIN registry. There are pros and cons to each. Participation in the ARIN registry requires signing an ARIN RSA (Registration Services Agreement). Some organizations have had difficulty accepting this agreement. RADB requires signing what many view to be a lighter-weight user

agreement. RADB has an annual fee for each maintainer object (in general, an organization, which can be a regional with a number of members behind it, needs only one) of, as of late 2019, \$395 for non-profit organizations. ARIN levies no additional fees for its registry, but any maintenance fees for number objects still apply. Another public registry is ALTDB, which does not require an explicit agreement or payment of fees. However, because of its somewhat nebulous support from the community, its use is difficult to recommend unless the agreements and/or fees of the other registries are insurmountable obstacles.

Tools for working with IRR

bgpq3

- e.g. on Juniper:
 - `bgpq3 -J -I as14325_in AS14325`
- e.g. IOS-XR
 - `bgpq3 -X -I as14325_in AS14325`
 -

IRRToolSet has a long history. It originated as part of the NSFnet Routing Arbiter project at USC ISI, then moved to RIPE NCC and now is hosted by ISC. It originally had some graphical tools for editing objects, but the three most strongly supported tools (at least from the FreeBSD package distribution) are

- `rtconfig`: routing policy analyzer and router configuration tool,
- `peval`: low-level policy evaluation tool and
- `rpslcheck`: provides syntax checking of the aut-num policy for an AS