RPKI Validation

Attacks against the routing system are increasing, and it's not uncommon in today's Internet world to experience prefix hijacking. The IETF has for a while, been woking on an Internet Resource Public Key Infrastructure, to help validate routing (BGP) announcements.

Details on RPKI and how this works is best followed up through your RIR. The RIPE-NCC in particular has excellent resources for you to peruse, and another excellent set of guidelines is available at https://rpki.readthedocs.io. INX runs separate workshops on IRR and RPKI usage, so look out for our announcements, and join the classes.

At INX-ZA, we operate a few RPKI relying party caches that we use in production, and which, in true community spirit, we make available to the general public for use. These are spread across South Africa, and are freely available for use for prefix validation. We place no restriction on reasonable use.

We stongly recommend that each network implements their own set of validators. We provide these for use as backup and/or failover validators primarily for peers at the INXes, who are typically one network hop away from us

Validators you may use are (note the IATA city code if you want geographic diversity)

- vc1-jnb.inx.net.za (Routinator 3000)
- vc2-jnb.inx.net.za (GoRTR)
- vc1-cpt.inx.net.za (Routinator 3000)
- vc2-cpt.inx.net.za (GoRTR)
- vc1-dur.inx.net.za (Routinator 3000)
- vc2-dur.inxnet.za (GoRTR)

All of the hosts are dual-stacked; please remember to use the v6 addresses, if your router supports IPv6.

The above are meant to be used by your routers, and don't have pretty front-ends, we also operate an easy to use RPKI relying party cache/validator that you can use to lookup/debug manually. This is available at https://rpki-validator.inx.net.za and yes, we use the ARIN TAL as well!

Of course the point of RPKI validation is for your network equipment to do this automatically, so we suggest the following configuration:

Cisco RPKI Config

```
router bgp 65001
 bgp rpki server tcp <<host>> port 3323 refresh 900
```

That's really all you need to do, to get your router speaking the RTR protocol to a relying party cache /validator! Of course, you still need to apply some kind of policy from this, once you have verified the RPKI "status" of the prefix.

Recommendations

We recommend that you

- assign a higher local-pref to prefixes that have a Valid ROA
- leave prefixes with Not-Found ROAs untouched (for now)
- drop prefixes that are marked as Invalid ROA

You can do this through any regular means of route policy manipulation on your routing equipment (eg. route-maps on a Cisco). Again, the resources mention above have more in-depth examples.

Dealing with Invalids

Operators may be tempted to choose an approach where they set the local-pref of RPKI Invalids to something really low (ie. least preferred). This really isn't a good idea. The simple problem you're still likely to see is that a more-specific (ie. longer match) route for this, will **still** win in the BGP route selection process, and therefore still leave you vulnerable to attack.

RPKI deployment at INX

From June 2019, INX drops RPKI invalids on our BGP Route server service. We also tag these for easy debugging and alert peering networks that they are propagating these paths. Details for these, and other filtered routes, can be seen directly, from the view into the INX BGP-RS services.

Should you need assistance with this, please feel free to send a mail to ops [at] inx.net.za