

RPKI for managers

5th NLNOG Day

7 September 2018

Niels Raijer, Fusix Networks BV

Contents

- RPKI refresher - why do we do this?
- What do you need to do to make it happen?
- Decide on your policy
- What will you encounter?
- Conclusion

RPKI refresher

RPKI refresher

- BGP announcements can be truly wild
- RPKI ties the correct prefix length together to the correct origin AS
- We need this since many Hurr^WPC^W networks do not properly filter their customers and allow hijacked networks to propagate freely, trusting whatever their customers announce

So why do we do this?

- Reduce the impact of BGP hijacks (most famous routing incidents could not have occurred with widely deployed RPKI with reject invalids)
- Honour other networks' intentions - operators don't publish ROAs for nothing, everyone wants someone to do something useful with their ROAs
- Rejecting invalids is mutually beneficial ("If I accept your misconfiguration, we both suffer")
- IXP peering LAN protection - if IXPs publish ROAs, then our network will protect itself and others automatically
- Marketing: Fusix focus on security

What do you need?

What do you need?

- Management decision that this is A Good Thing
- Of course, publish your own ROAs correctly
- Tiny bit of internal training for your first line support staff
- A (virtual) machine that can run the RPKI validator
- Half an hour of a network engineer's time (OK, that depends a bit on the number of routers you have)

Management decision

- Really the question is:
when will the first customer drag you to court for allowing a hijacked BGP prefix into your network, enabling bad guys to steal all your customer's virtual money?

Support training

- Question arriving to your support department will be of the form: “We can’t reach IP address x.x.x.x”
- Your support staff needs to know how to check for RPKI errors (can easily be built into your existing support systems)
- `whois -h whois.bgpmon.net x.x.x.x | grep RPKI`

```
niels@k4:~$ whois -h whois.bgpmon.net 94.142.240.1 | grep RPKI
RPKI status:          ROA validation successful
```

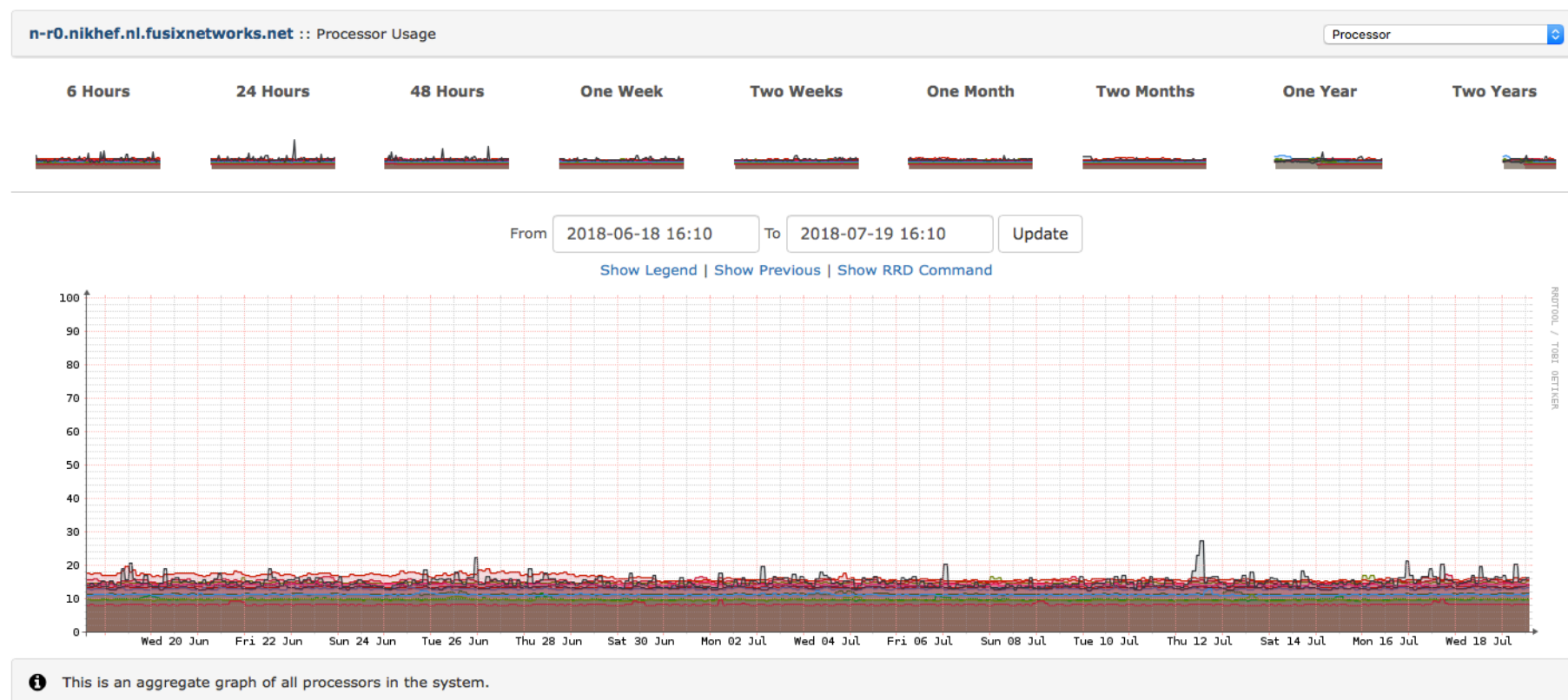
```
niels@k4:~$ whois -h whois.bgpmon.net 176.92.110.83 | grep RPKI
RPKI status:          ROA validation failed: Invalid Prefix-Length
```

Virtual machine

- A machine that can run the validator
- We run the RIPE NCC validator, but there are others
- Don't forget to import the ARIN TAL (Trust Anchor Locator)

Half an hour

- Yes, really, especially if you have automated your router configurations
- But but but this will chew up all the CPU on my router!!11



Decide on your policy

Your RPKI policy

- The validation database is separate from your RIB and FIB
- Your routing policies should be configured to take action on the RPKI status of every prefix (valid, invalid, unknown) as the route goes from RIB to FIB
- Common, useless approach: put lower local preference on invalid prefixes (DON'T DO THIS)
- The right thing to do is to **reject invalid prefixes**



What will you encounter?



What will you encounter?

- Customers will say that a certain prefix is not reachable
- If this is owing to RPKI with invalid rejects, then send an explanation to the customer, tell them how to get the other party to resolve it, and temporarily whitelist if it doesn't feel like a hijack
- We present RPKI with invalid rejects as an established fact
- This has been accepted by all customers so far

**Number of support cases since
enabling RPKI with invalid rejects**

2

**Number of these resolved within a day
or so after explaining to customer**

1

**Number of BGP hijacks that
did not affect our network**

521*

*** Sure. It's an estimate.**

Production networks in NL with RPKI invalid reject

- Amsio
- AMS-IX route servers
- Atom86
- Fusix
- True
- and you....!

Conclusion

Conclusion

- Using **RPKI with invalid rejects** is **viable for a production network**
- After enabling RPKI with invalid rejects, you can count on a small number of support cases (in our case, roughly 0.3 per week)
- However the number of BGP hijacks that your network has not been exposed to is much higher

