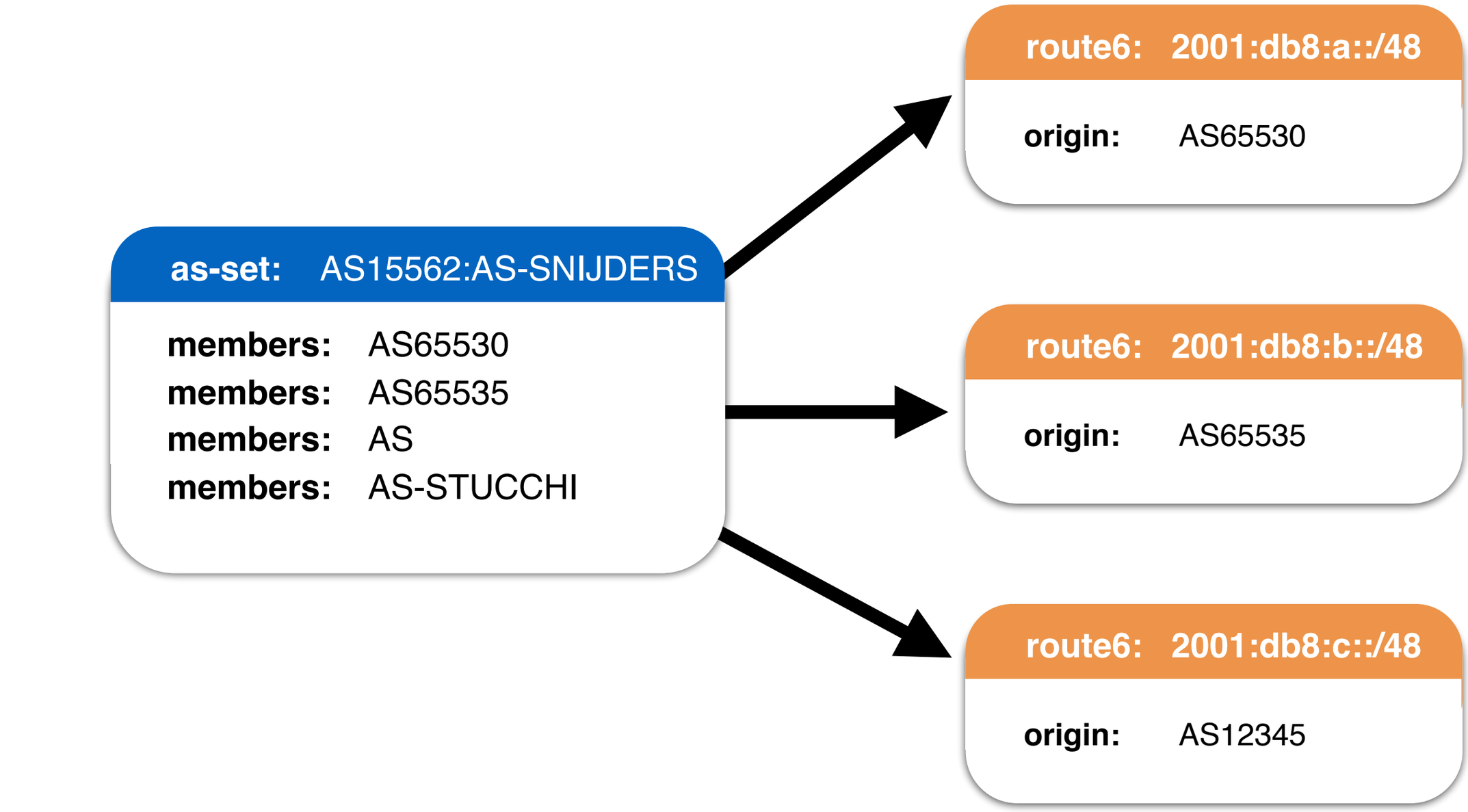
RIPE NCC
RIPE NETWORK COORDINATION CENTRE

RPKI AS-Cones

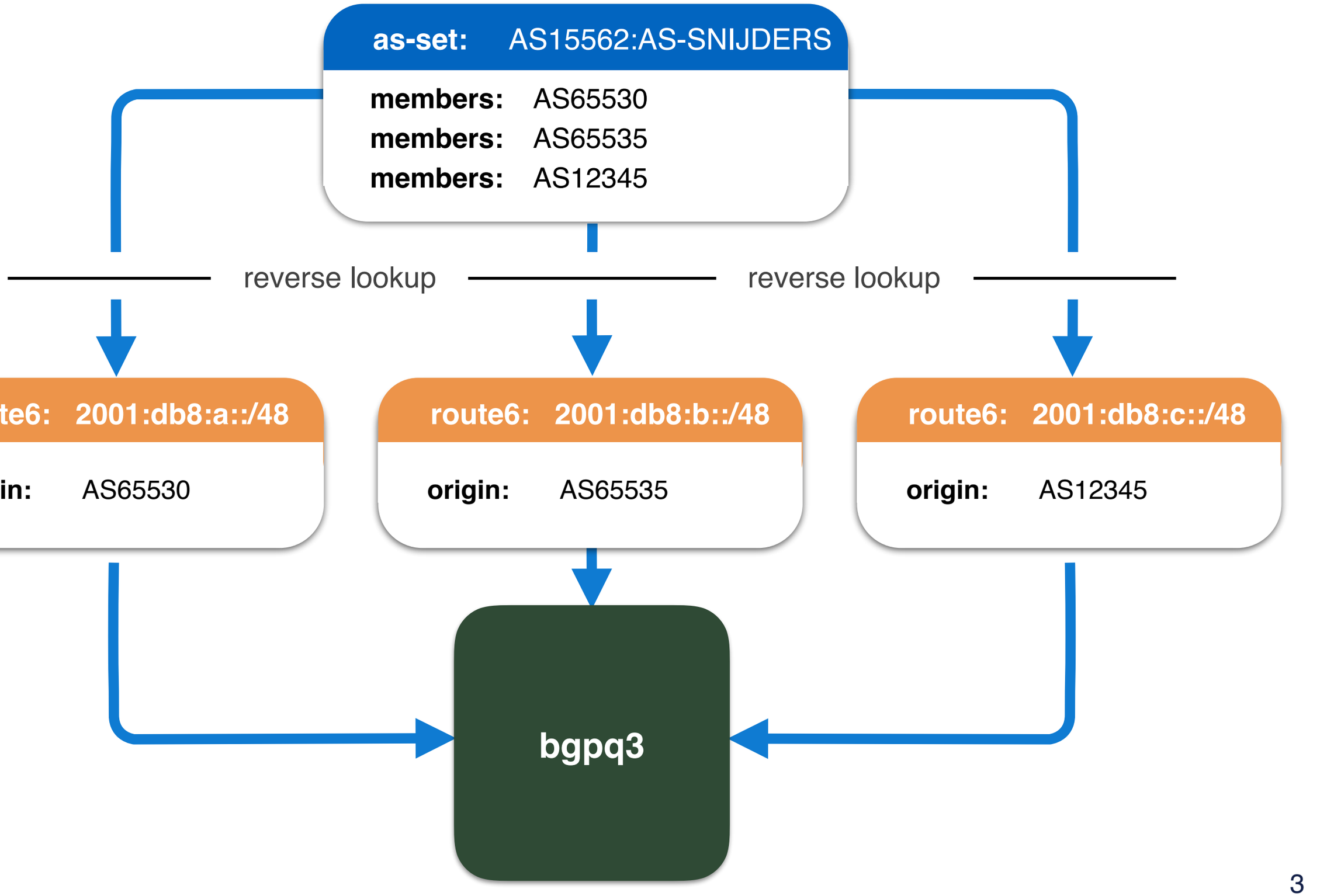
Bringing AS-Sets Functionality to RPKI

Massimiliano Stucchi | RONOG5 | October 2018

ROUTE Objects and AS-Sets



Generating a Prefix Filter

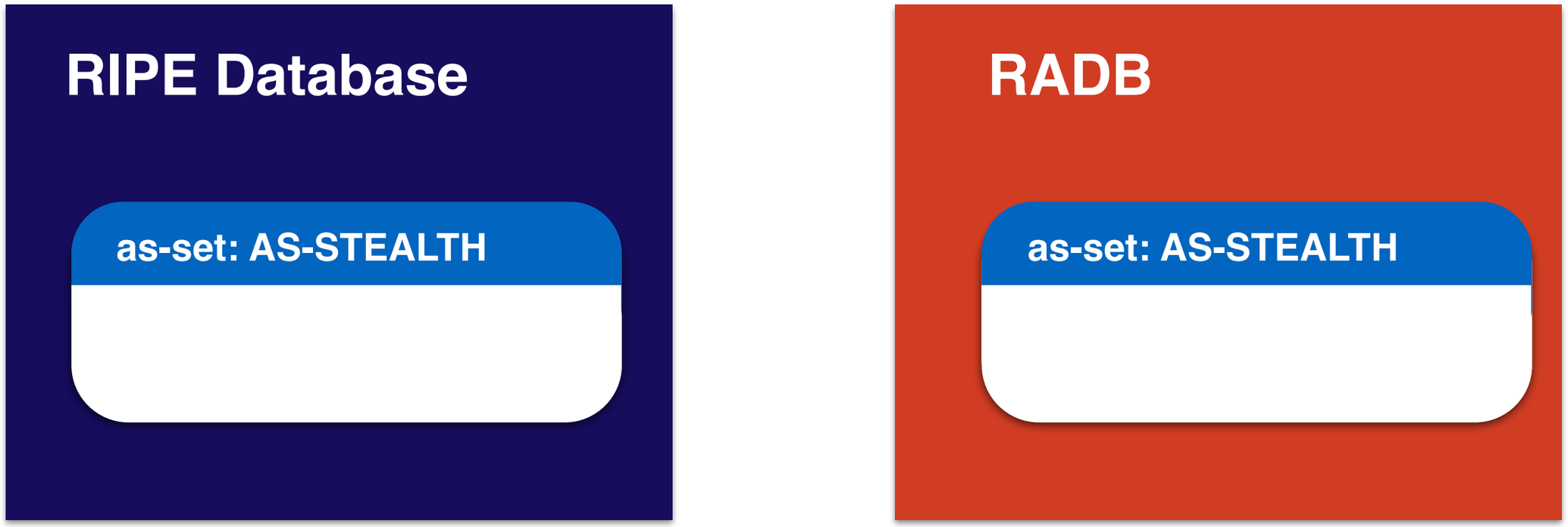


Generating a Prefix Filter

```
$ bgpq3 -h rr.ntt.net -l AS15562-in AS15562:AS-SNIJDERS
no ip prefix-list AS15562-in
ip prefix-list AS15562-in permit 67.221.245.0/24
ip prefix-list AS15562-in permit 165.254.255.0/24
ip prefix-list AS15562-in permit 165.254.255.0/26
ip prefix-list AS15562-in permit 165.254.255.26/32
ip prefix-list AS15562-in permit 165.254.255.64/26
ip prefix-list AS15562-in permit 165.254.255.132/32
ip prefix-list AS15562-in permit 165.254.255.133/32
ip prefix-list AS15562-in permit 165.254.255.144/28
ip prefix-list AS15562-in permit 165.254.255.149/32
ip prefix-list AS15562-in permit 165.254.255.160/28
ip prefix-list AS15562-in permit 192.147.168.0/24
ip prefix-list AS15562-in permit 204.2.30.0/23
ip prefix-list AS15562-in permit 204.42.254.192/26
ip prefix-list AS15562-in permit 209.24.0.0/16
```

Limitations of AS-Sets

- Can exist in multiple IRRs



- AS-STEALTH exists in both the RIPE Database and RADb
- The two are not managed by the same company

How to Discover What AS-Set to Use?

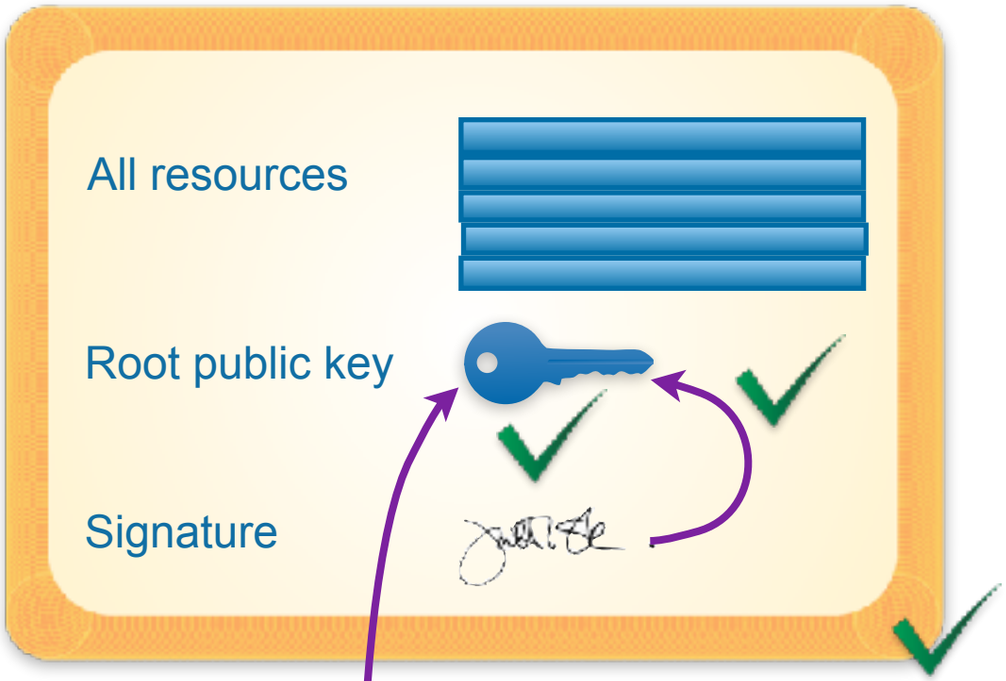
- Ask people in the service order form
- Look at PeeringDB
- Different trust levels based on the IRR

RPKI

- **A ROA is very similar to a ROUTE object**
  - With Prefix and Origin
  - With an additional field: maxLength

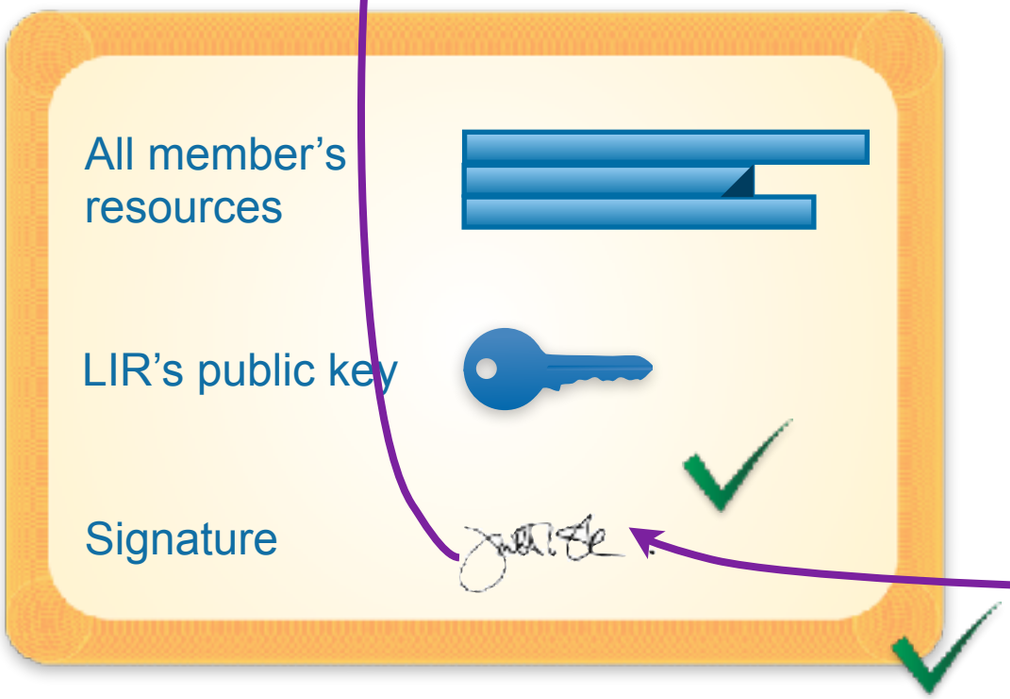
- **RPKI is data you can trust**
  - We know the ROAs are created by the holders of the address space

RIPE NCC's Root Certificate



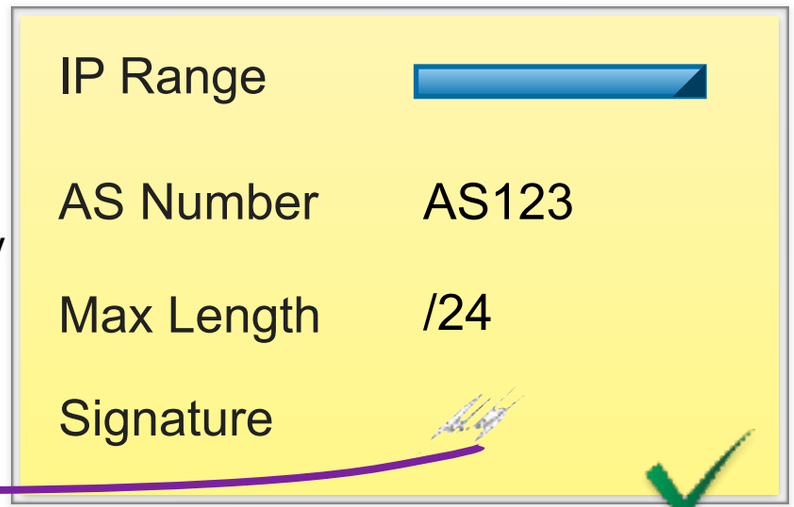
Root's (RIPE NCC) private key

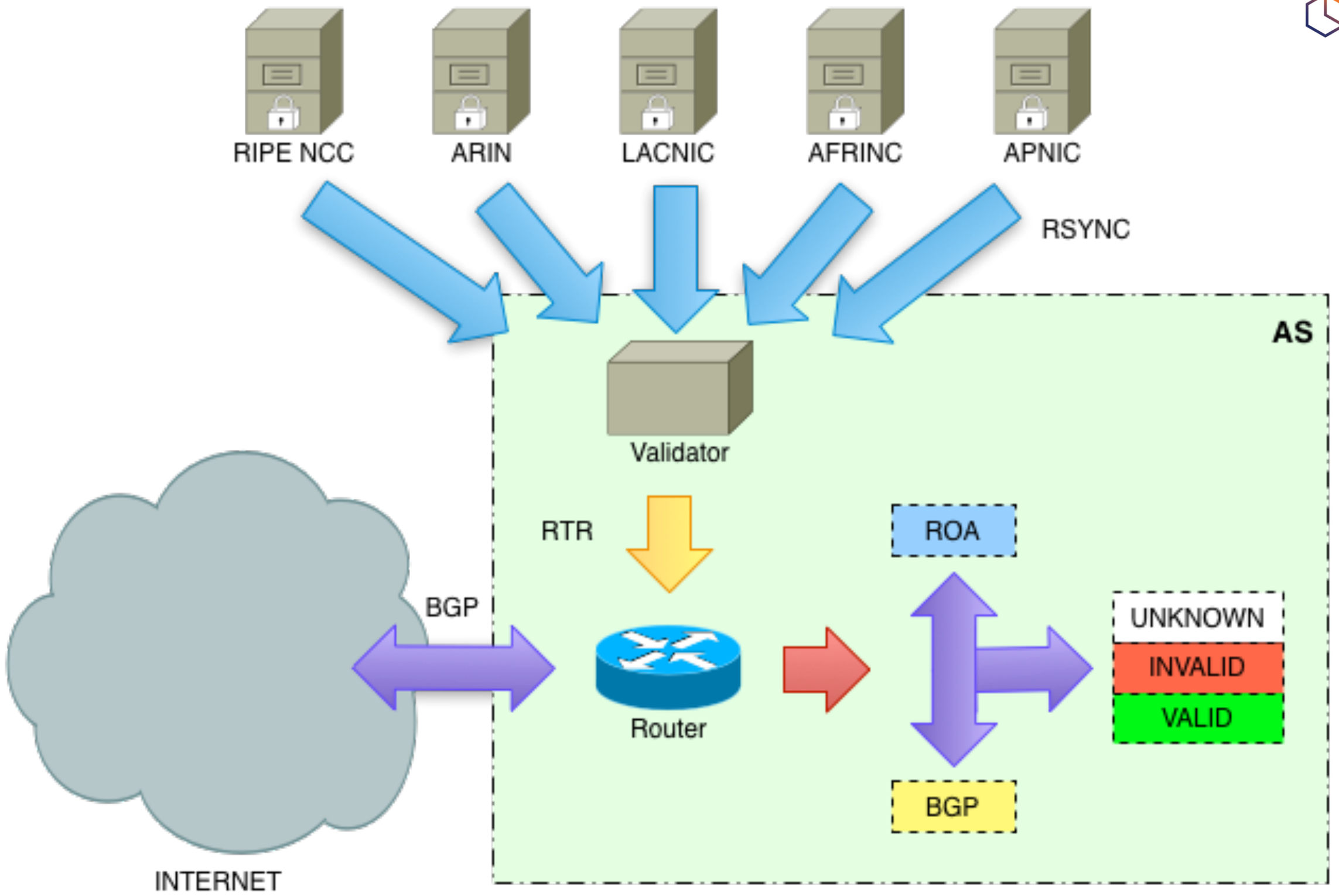
LIR's Certificate



LIR's private key

ROA




RIPE NCC
ARIN
LACNIC
AFRINC
APNIC
RSYNC
AS
Validator
RTR
ROA
BGP
UNKNOWN
INVALID
VALID
Router
BGP
INTERNET

Enter AS-Cones

AS-Cones

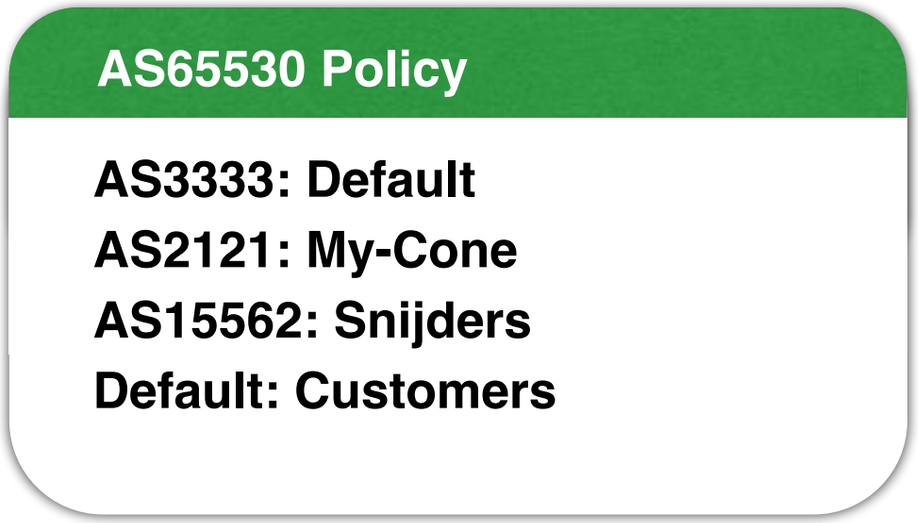
- **New IETF Draft**
  - https://datatracker.ietf.org/doc/draft-ss-grow-rpki-as-cones/

- **Goals**
  - Create more feature parity between IRR and RPKI
  - Make provisioning operations easier
  - Go global, independent from IRR

Features of AS-Cones

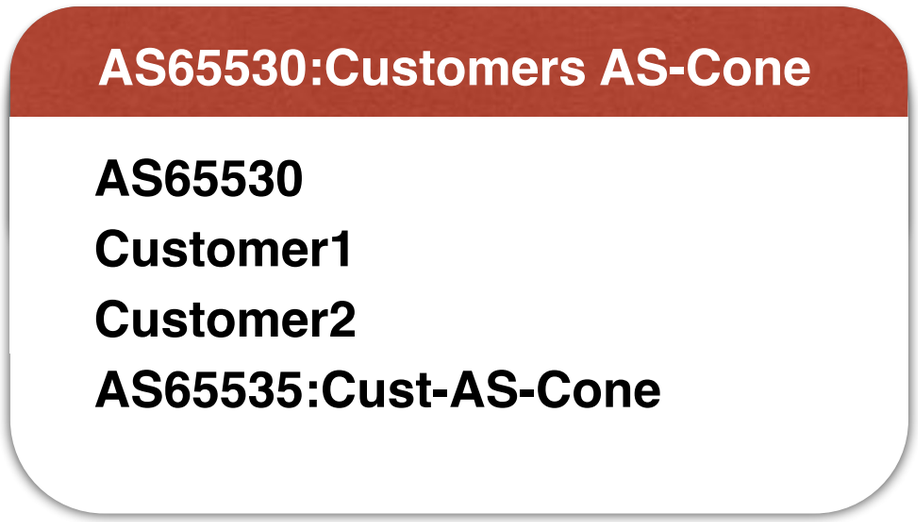
- Granularity of declarations
- Default namespace
- Simple validation process
- Stub networks don’t need to do anything

Policy Object



- **Must contain a “Default” policy**
  - Which, by default, contains only the ASN
- **Every relationship can point only to an AS-Cone**
  - No reference to only one ASN (for this, create an AS-Cone)

AS-Cone Object



- **Contains a list of ASNs or AS-Cones from customer networks**

- **Referenced as ASXXXX:Cone_name**
  - Name must be unique only per ASN

Finding Policies and AS-Cones

- Policies and AS-Cones should be distributed by your favourite Validator
- To generate prefix filters, access the validated cache via an API

Generating Prefix Filters with AS-Cones

- **As an upstream, read the policy definition for your customer network. Check:**
  - If it contains a specific policy declaration, otherwise Default
- **Take the AS-Cone referenced**
- **Walk the AS-Cone, create a list of all the ASN included**
  - If you find circular AS-Cones declaration, discard them
- **For every ASN, pick all the ROAs where it's listed as originator**

AS65530 Policy

AS3333: Default
AS2121: My-Cone
Default: Customers

AS65530:Customers

AS65531
Customer1
Customer2

AS65531:Customers

AS65534
Customer10
Customer20

AS65532:Customers

AS65535
Customer30
Customer40

AS65533:Customers

AS65536
Customer21
Customer22

AS65534:Customers

ASXXXXX

AS65556:Customers

AS6553YY

AS65530:Customers

AS6553X
Customer1XX
Customer2XX

AS65542:Customers

AS655XX

AS65543:Customers

AS655YY

AS65544:Customers

AS655YY

AS65546:Customers

AS65563
Customer1
Customer2

References

- **Material on Github**
  - https://github.com/bgp/draft-ss-grow-rpki-as-cones

- **Discussion welcome in the Grow IETF WG**

Other work

- **BGPSEC**
  - RFC
  - Not implemented, not usable at the moment
- **ASPA**
  - ASN Policy Attestation
  - Draft, soon to be adopted by GROW

Questions

mstucchi@ripe.net