

Beyond your network, why Routing Security requires global collaborative effort

MANRS

Kunal Raj

E: kunalkrishnilraj@gmail.com

Telecom Fiji Limited

Acknowledgement

- This paper is made taking notes, diagram, configurations from MANRS, APNIC training materials, NLnet Labs & Dr. Philip Smith along with the operational experience of the author.

MANRS

- Mutually Agreed Norms for Routing Security (MANRS) is a global initiative, supported by the [Internet Society](#), that provides crucial fixes to reduce the most common routing threats.

MANRS

- MANRS outlines four simple but concrete actions that network operators should take:
 - Filtering
 - Anti-spoofing
 - Coordination
 - Global Validation

Global Validation

- Publish your data, so others can validate routing information on a global scale
- Automated information validation needs arrangements
- Securing global routing information is done by RPKI

Resource Public Key Infrastructure (RPKI)

- RPKI allows holders of Internet number resources to make verifiable statements about how they intend to use their resources.
- RPKI is a way to define data in an out-of-band system such that the information that are exchanged by BGP can be validated to be correct.
- RPKI is used to make Internet routing more secure.

Importance of RPKI

- Secured Routing Table
- Dynamic LOA checking
- Maintaining a Dynamic Chain of Trust
- Digitally Signed Resources Certificate (X.509 Certificates-RFC5280)
- Helps to Stop Route Hijack

Route Origin Authorizations (ROA)

- Using the RPKI system, the legitimate holder of a block of IP addresses can use their resource certificate to make an authoritative, signed statement about which autonomous system is authorized to originate their prefix in BGP.
- These statements are called Route Origin Authorizations (ROAs).

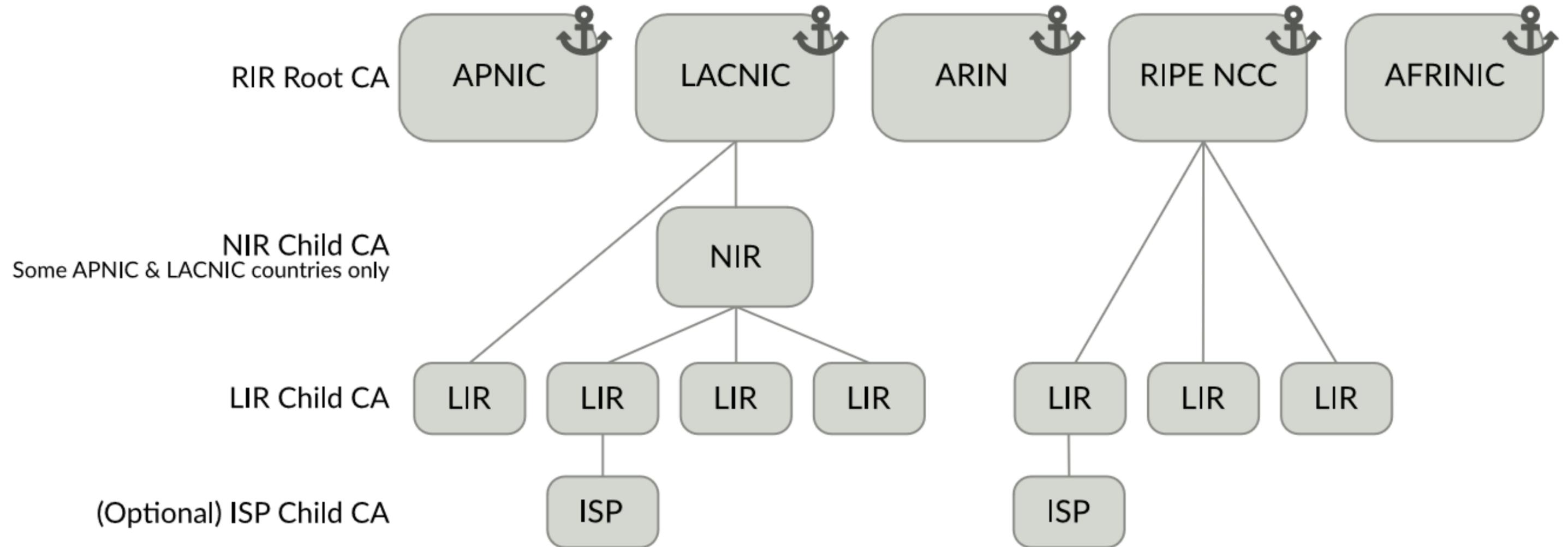
Route Origin Validation (ROV)

- RPKI system tries to closely mimic what route objects in the IRR intend to do, but then in a more trustworthy manner.
- This process is called route origin validation (ROV)

Validity

- Valid - Resources found in database which is called Validated ROA Payload (VRP).
- Invalid – Resources found but partial/whole information doesn't match with database.
- Not Found - The prefix in this announcement is not covered by a VRP.

RPKI - Chain Of Trust



Working Steps

- Creating ROA for owned resources for RPKI
- Implementing Validator relying software for ROV
- Enforcing policies for based on Validation

Creating ROA

- Go to Resources > Route Management and select Create route

MyAPNIC

Home Resources Admin Contact Tools Events My Profile

Home / Resources

Resources

Internet Resources

Summary
View all of your resource holdings.

IPv4
View your IPv4 resource holdings.

IPv6
View your IPv6 resource holdings.

AS Numbers
View your ASN resource holdings.

Reverse DNS Delegations

Add Reverse Delegations
Add new reverse delegations.

Reverse Delegation Summary
View and manage reverse delegations

Whois Updates

Whois Updates
Add, update, and delete individual Whois objects.

Bulk Whois Updates
Add, update, and delete multiple Whois objects.

Contact Details Update
Update contact details of the internet resources associated with your account.

Maintainers
View your registered maintainers, and register new maintainers.

IRTs
View your registered IRT objects, and register new IRT objects.

Resource certification

RPKI
Set up your RPKI engine, and manage your Route Origin Authorization (ROA) objects.

Route management
Routes
Add, update, delete and view routes. Create Route Origin Authorisation (ROA) for routes.

Home / Resources / Routes

Routes

Register your routes in MyAPNIC using the tool below. It will automatically create route objects in the APNIC Whois Database with any AS number you have authorized. RPKI ROAs will also be created at the same time, if the ROA option is enabled (changes to RPKI may take around ten minutes to propagate so the ROA status will not be updated until then).

Create route **Delete selected**

Show 10 entries Search:

Select all Deselect all

	Route	Origin AS	ROA status	Whois status	Actions
<input type="checkbox"/>	2001:df0:a::/48	AS45192	✔	✔	Edit Delete
<input type="checkbox"/>	2001:df2:ee00::/48	AS131107	✔	✔	Edit Delete
<input type="checkbox"/>	2001:df2:ee01::/48	AS45192	✔	✔	Edit Delete
<input type="checkbox"/>	202.125.96.0/24	AS131107	✔	✔	Edit Delete
<input type="checkbox"/>	202.125.97.0/24	AS45192	✔	✔	Edit Delete

https://www.apnic.net/wp-content/uploads/2017/12/ROUTE_MANAGEMENT_GUIDE.pdf

Creating ROA

- Mention your prefix with ASN & desired subnet & Submit

Example for IPv4

Create route

Prefix: 61.45.248.0/21

Origin AS: 45192

MSA: /24

ROA: Enabled

Whois: Enabled

Define Whois route attributes

Options: Notify additional contacts

Cancel Next

Confirm route creation

ROA: Enabled

Whois: Enabled

Prefix: 61.45.248.0/21

Origin AS: 45192

Most specific announcement: /24 (distance from prefix length: 3)

Select the sub-routes to be enabled:

Show 10 entries Search:

Select all Deselect all

Route
<input checked="" type="checkbox"/> 61.45.248.0/21
<input checked="" type="checkbox"/> 61.45.248.0/22
<input checked="" type="checkbox"/> 61.45.248.0/23
<input checked="" type="checkbox"/> 61.45.248.0/24
<input checked="" type="checkbox"/> 61.45.249.0/24
<input checked="" type="checkbox"/> 61.45.250.0/23
<input checked="" type="checkbox"/> 61.45.250.0/24
<input checked="" type="checkbox"/> 61.45.251.0/24
<input checked="" type="checkbox"/> 61.45.252.0/22
<input checked="" type="checkbox"/> 61.45.252.0/23

Showing 1 to 10 of 15 entries 15 rows selected

Previous 1 2 Next

Cancel Go back Submit

https://www.apnic.net/wp-content/uploads/2017/12/ROUTE_MANAGEMENT_GUIDE.pdf

Relying Party Software

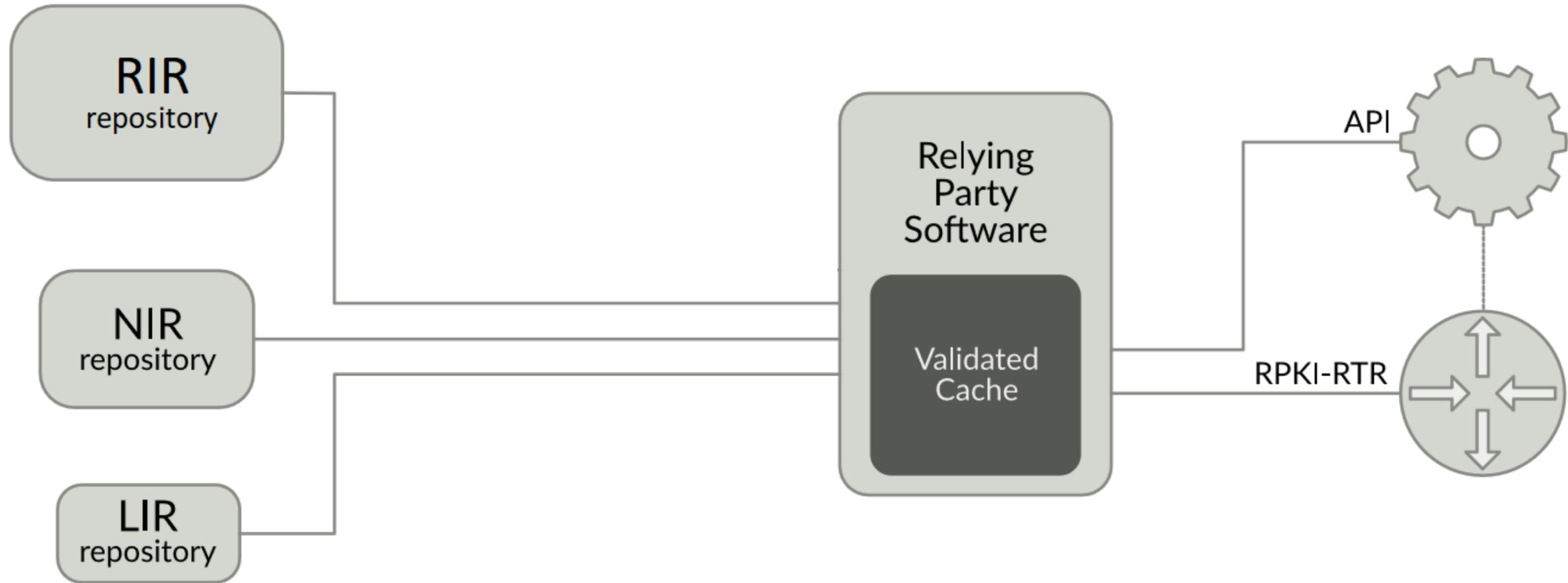
- ❑ NLnet Labs Routinator
 - <https://www.nlnetlabs.nl/projects/rpki/routinator/>
 - <https://github.com/NLnetLabs/routinator>
- ❑ Dragon Research validator
 - <https://rpki.net>
 - <https://github.com/dragonresearch/rpki.net/>
- ❑ RIPE NCC validator
 - <https://github.com/RIPE-NCC/rpki-validator-3/wiki>
- ❑ LACNIC/NIC Mexico validator (FORT)
 - <https://github.com/NICMx/FORT-validator>
- ❑ Cloudflare validator (OctoRPKI)
 - <https://github.com/cloudflare/cfrpki>

Routinator

- Routinator is free, open source RPKI Relying Party software written by NLnet Labs in the Rust programming language.
- Routinator connects to the Trust Anchors of the five Regional Internet Registries (RIRs) — APNIC, AFRINIC, ARIN, LACNIC and RIPE NCC — downloads all of the certificates and ROAs in the various repositories, verifies the signatures and makes the result available for use in the BGP workflow.
- The validated cache can be fed directly into RPKI-capable routers via the RPKI to Router Protocol (RPKI-RTR), described in RFC 8210.

<https://rpkireadthedocs.io/en/latest/routinator/index.html>

Ecosystem



Why Routinator?

- Designed to have a small footprint and great portability
- Can run on any Unix-like operating system, but also works on Microsoft Windows via API
- Have a mailing list for general discussion and exchanging operational experiences (<https://nlnetlabs.nl/mailman/listinfo/rpki>)
- Problem report & feature request is possible (<https://github.com/NLnetLabs/routinator/issues>)
- Used in production by AT&T, NTT, AMS-IX, DECIX and many more

<https://rpki.readthedocs.io/en/latest/routinator/index.html>

Installation

- `curl https://sh.rustup.rs -sSf | sh`
- `sudo apt install cargo`
- `source ~/.cargo/env`
- `cargo install routinator`
- `routinator init --accept-arin-rpa`
- `routinator server --rtr [SERVER IP]:3323 --http [SERVER IP]:9556 -d`
- `routinator -v vrps`

<https://rpki.readthedocs.io/en/latest/routinator/installation.html>

Adding Into Crontab

- `nano /etc/rovscript.sh`

```
#!/bin/bash
```

```
/home/[USER]/.cargo/bin/routinator init -f --accept-arin-rpa &
```

```
/home/[USER]/.cargo/bin/routinator server --rtr [SERVER IP]:3323 --http [SERVER IP]:9556 -d &
```

- `sudo chmod +x rovscript.sh`

- `crontab -e`

```
@reboot /etc/rovscript.sh
```

```
5 13 * * * /home/nano/.cargo/bin/routinator -v vrps &
```

```
5 0 * * * /home/nano/.cargo/bin/routinator -v vrps &
```

Allow In Iptables

- `-A INPUT -i ens18 -p tcp -m tcp --dport 873 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-A INPUT -i ens18 -p tcp -m tcp --sport 873 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-A INPUT -i ens18 -p tcp -m tcp --dport 9556 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-A INPUT -i ens18 -p tcp -m tcp --sport 9556 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-A INPUT -i ens18 -p tcp -m tcp --dport 3323 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-A INPUT -i ens18 -p tcp -m tcp --sport 3323 -m state --state NEW,ESTABLISHED -j ACCEPT`

- `-A INPUT -i ens18 -p tcp -m tcp --dport 9100 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-A INPUT -i ens18 -p tcp -m tcp --sport 9100 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-A INPUT -i ens18 -p tcp -m tcp --dport 9090 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-A INPUT -i ens18 -p tcp -m tcp --sport 9090 -m state --state NEW,ESTABLISHED -j ACCEPT`

Router Configuration

```
router bgp [ASN]
```

```
rpki server [SERVER IP]
```

```
transport tcp port 3323
```

```
refresh-time 120
```

```
address-family ipv4 unicast
```

```
bgp origin-as validation signal ibgp
```

```
address-family ipv6 unicast
```

```
bgp origin-as validation signal ibgp
```

Configuration of IOS-XR

Router Configuration

- routing-options {
 autonomous-system [ASN];
 validation {
 group rpki-validator {
 Session [Server IP] {
 refresh-time 120;
 Port 3323;
 local-address X.X.X.253;
 }
 }
 }
}

Configuration of Junos

Checking

- `ps ax| grep routinator`

```
1369 ?    SI 124:05 /home/[USER]/.cargo/bin/routinator server --rtr [SERVER IP]:3323 --http [SERVER IP]:9556 -d
7487 pts/0  S+  0:00 grep --color=auto routinator
```

- `sh bgp rpki server summary`

Sun Nov 3 12:27:37.333 UTC

Hostname/Address	Transport	State	Time	ROAs (IPv4/IPv6)
[SERVER IP]	TCP:3323	ESTAB	6d19h	97473/16440

Decision

- Since now the validation states are visible to you, you can decide what to do with invalids
- You can –
 - Use them with low preference
 - Or drop them

Policy

```
route-policy RPKI
  if validation-state is invalid then
    set local-preference 50
  else
    if validation-state is valid then
      set local-preference 200
    else
      pass
    endif
  endif
end-policy
```

```
route-policy RPKI
  if validation-state is invalid then
    drop
  else
    if validation-state is valid then
      set local-preference 200
    else
      pass
    endif
  endif
end-policy
```

Configuration of IOS-XR

Closure

- <https://sg-pub.ripe.net/jasper/rpki-web-test/>



Reference

- <https://www.manrs.org/about/>
- <https://blog.apnic.net/2019/10/28/how-to-installing-an-rpki-validator/>
- <http://www.bgp4all.com.au/pfs/training/apnic48/agenda>
- <https://www.nlnetlabs.nl/projects/rpki/routinator/>
- <https://www.ripe.net/manage-ips-and-asns/resource-management/certification/tools-and-resources>
- <https://github.com/cloudflare/cfrpki#octorpki>
- [RPKI Validator - Quick Overview of BGP Origin Validation \(apnic.net\)](#)

Query !!

Thanks ...