An opinionated review of RPKI validators

and the state of their Debian packaging

Marco d'Itri
<md@seeweb.it>
@md@linux.it

Seeweb s.r.l. | The DHH group

ANIX meeting 2024 - December 6, 2024

Content

seeweb

What is RPKI?

RPKI is the system used to cryptographically validate BGP announcements.

The data components of RPKI:

- Route Origin Authorizations (ROA): the certificates stating which ASN is authorized to announce certain IP networks. A bit like RPSL route objects.
- Autonomous System Provider Authorizations (ASPA): the certificates stating which upstreams are authorized for an ASN.

The software components of RPKI:

- Publishing infrastructure by RIRs and networks.
- Validation infrastructure by each network.

seeweb

RPKI in practice

Networks use RPKI to verify that the routes received from BGP peers, transits and customers are not spoofed.

BGP routers check if the state of a route is valid, invalid or unknown.

The software used by ISPs:

- Validators: collect the ROAs and ASPAs and verifies them.
- RPKI-to-Router (RTR) servers: make the result of validation available to the routers.

seeweb

Content

seeweb

The software (1)

Validators

- Routinator 3000
- OpenBSD's rpki-client
- ~~RIPE NCC RPKI Validator~~ (discontinued)
- ~~OctoRPKI~~ (discontinued)
- FORT Validator (development restarted in mid-2023)
- rpki-prover (niche software)
- ~~Dragon Research Labs RPKI toolkit~~ (not developed since 2018)

seeweb

The software (2)

OctoRPKI and rpki-client do not implement the RPKI-to-router (RTR) protocol themselves, but use an external daemon.

RTR servers

- ~~gortr~~ (abandoned)
- stayrtr

stayrtr is an actively maintained fork of gortr and has replaced it.

seeweb

Usage of validation software

	Oct 2021	May 2022	Apr 2023	Nov 2023	Dec 2024
Routinator	79%	69.9%	78.9%	79.2%	79.5%
rpki-client	8%	19.3%	9.3%	10.4%	10.1%
OctoRPKI	6%	3.5%	6.1%	4.5%	4.4%
FORT Validator	3%	3.2%	4.2%	3.9%	3.8%
RIPE NCC Validator	4%	4.4%	1.3%	1.9%	1.9%
rpki-prover	0%	0.5%	0.1%	0.1%	0.1%

This is dangerously close to becoming a *software monoculture*.

This data was gathered by NLNet Labs by counting the unique IPs accessing a RRDP web server.

seeweb

Routinator

Pros

- Actively developed, support contracts available.
- Well documented.

Cons

- Difficult to package by distributions.
- Too high adoption causes a lack of software diversity.

Developed in Rust by NLnet Labs.

seeweb

rpki-client

Pros

- Actively developed by network operators, support contracts available.
- Simple and essential.
- Separation of privileges in multiple processes.
- Quickly implements new protocol features.

Cons

- Needs a third party RTR daemon.

Developed in C by the OpenBSD project.

seeweb

RIPE NCC Validator

Pros

- Nothing else was available at the time?

Cons

- Written in Java.
- RIPE NCC stopped development.
- End of support in June 2021: **nobody should use it anymore!**

Developed in Java by RIPE NCC.

seeweb

OctoRPKI

Pros

- Simple and essential.

Cons

- Feels like a Cloudflare-specific project, the development roadmap is unclear.
- Needs a third party RTR daemon.
- Officially discontinued in March 2024: **nobody should use it anymore!**

Developed in Go by Cloudflare.

seeweb

FORT Validator

Pros

- Used to be actively developed.
- Well documented.
- Good middle ground of features and complexity.

Cons

- After a long pause development resumed in mid-2023, but it is still slow.

Developed in C by LACNIC and NIC.MX.

seeweb

rpki-prover

Pros

- Software diversity is good.

Cons

- Niche programming language.
- ~~**Very** low~~ No adoption.

Developed in Haskell by Mikhail Puzanov.

Should I package it?

My suggestions

Use two of:

- Routinator
- FORT Validator (?)
- rpki-client + stayrtr

They are all good and have different tradeoffs.

Using software packaged by a Linux distribution significantly reduces the system administration effort and allows to adopt diverse implementations.

Software diversity is important and needs to be encouraged!

seeweb

Features

	BGPSec	ASPA	RSC	signed TALs
Routinator	✓	✓		
rpki-client	✓	✓	✓	✓
FORT Validator				
rpki-prover	✓	✓	✓	

Content

seeweb

Why use packaged software

The great debate: packages from distributions[1] or the developers?

Why use distribution packages?

- Integration with the OS and high attention to details.
- Ready to use after the installation.
- Automatic security updates[2].
- Maintained by system administrators, not software developers.

Why use vendor packages?

- Freshness.

[1] Full disclosure: I develop a Linux distribution (Debian).

[2] Job Snijders estimated in 2022 that over 70% of the clients currently in use are insecure.

seeweb

Debian for network operators

Debian GNU/Linux is the one stop shop for all your RPKI validation needs.

My goals

- Packages with sane defaults which just work after being installed.
- Common management of TALs in the `rpki-trust-anchors` package.
- State of the art security with systemd sandboxing.

Issues

- The RPKI ecosystem is still young and fast-moving for a stable distribution.
- Routinator cannot be packaged (yet?).

seeweb

The issue with Routinator

The Rust development ecosystem is broken and hostile to distributions

- APIs are not stable (and there is no dynamic linking).
- Hence it is common for Rust software to depend on specific versions of libraries.
- General *vendoring* of dependencies is not acceptable to the Debian security team.
- Maintaining multiple versions of libraries in the distribution is too much time consuming (and not appreciated either...).
- Different Rust programs depend on different versions of the same library.
- **Packaging complex Rust projects is difficult.**

The Routinator developers publish a Debian package which is good enough, but it does not use `rpki-trust-anchors`.

The state of Debian RPKI packages

Package	Debian 11	Debian 12
routinator	✗	✗
rpki-client	✗	(✓)
~~octorpki~~	✗	✗
fort-validator	(✓)	(✓)
~~gortr~~	✓	✓
stayrtr	(✓)	(✓)
rpki-trust-anchors	✓	✓
OpenBGPD (bonus!)	✗	(✓)

I removed `gortr` from Debian 12, in favour of `stayrtr`.

All packages in Ubuntu 22.04 LTS are not up to date at this point and I do not recommend to use them for RPKI validation.

At this point I will not further update Debian 11.

seeweb

Backports to Debian/stable

Backported packages of RPKI-related software and OpenBGPD will be maintained in the official Debian backports archive at least until the release of Debian 13.

```
echo 'deb http://deb.debian.org/debian bookworm-backports main' \
  > /etc/apt/sources.list.d/bookworm-backports.list
apt update
apt install rpki-client/bookworm-backports stayrtr/bookworm-backports
```

I will do the same for Debian 13 after it will be released.

seeweb

Any questions?

https://www.linux.it/~md/text/rpki-validators-anix2024.pdf

(Google ... Marco d'Itri ... I'm feeling lucky)

seeweb