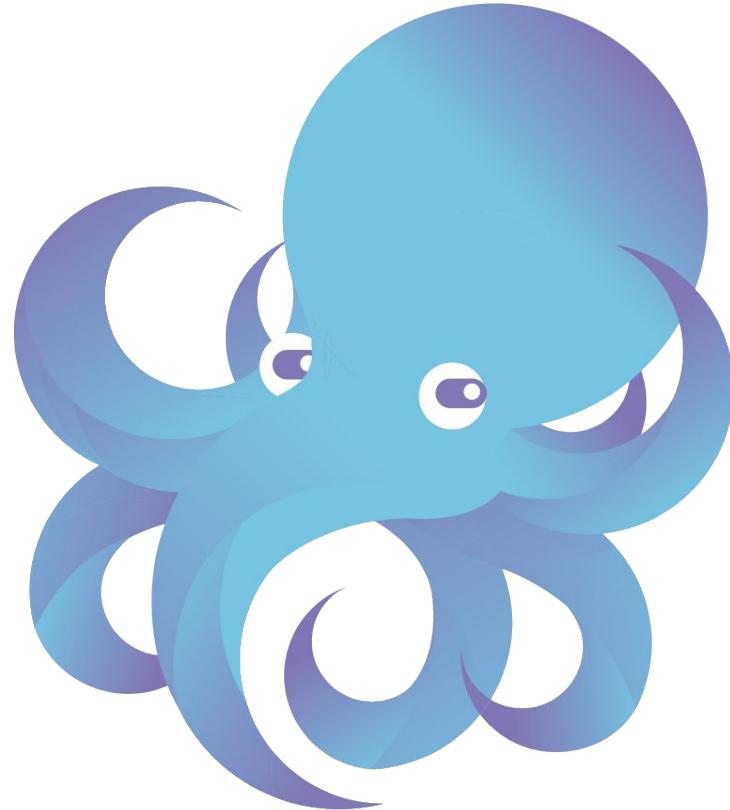




RPKI implementation Cloudflare

Louis Poinsignon



Introduction

Network Engineer at Cloudflare in San Francisco

Open-source projects including flows and RPKI

Network data collection (BGP, flows, peering-portal)

Story 1/4

The screenshot shows a web browser displaying a blog post on the Cloudflare website. The title of the post is "BGP leaks and cryptocurrencies". It was published on 24 April 2018 by Louis Poinsignon. The post discusses a recent attempt to steal cryptocurrencies using a BGP leak. Below the post is a stylized illustration of a burglar in a mask and hoodie, holding a device that looks like a safe or vault door.

BGP leaks and cryptocurrencies

24 Apr 2018 by Louis Poinsignon.

Share Like 739 Tweet

Over the few last hours, a dozen news stories have broken about how an attacker attempted (and perhaps managed) to steal cryptocurrencies using a BGP leak.



Categories

- Product News
- Security
- Performance
- Reliability
- Network
- Serverless
- International
- Cloudflare Apps

Enter your email address

Subscribe to this blog

US callers
1 (888) 99-FLARE
UK callers
+44 (0)20 3514 6970
Singapore callers
+65 3158 3954

Story 2/4

Authority DNS route hijack in April 2018.

This affected our DNS Resolver.

The route was sent to us on a Chicago peering session.

What should we do?

Story 3/4

At the time...

150+ PoPs, 26000 BGP sessions, IP space in 5 RIRs

Just the RIPE Validator^[1]

How to distribute a prefix list efficiently?

Story 4/4

July: started deploying internally GoRTR.

August: open-source release.

<https://github.com/cloudflare/gortr>

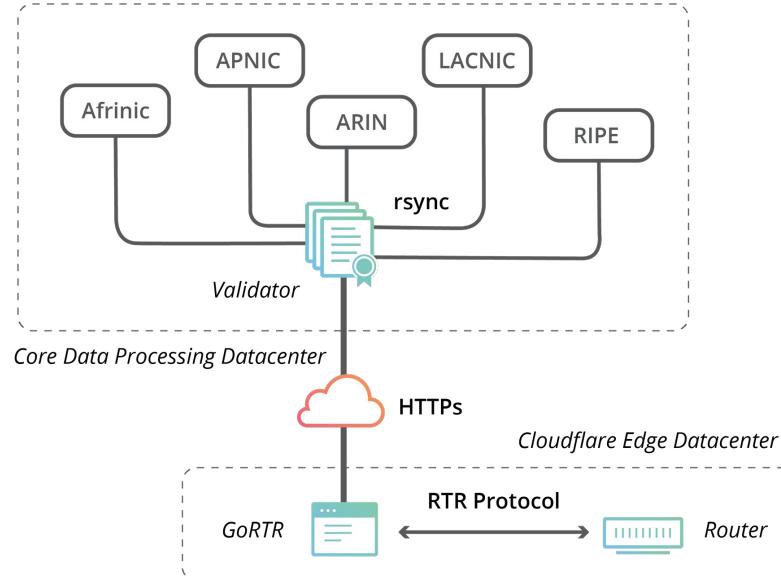
September → December:

- Turn up RTR sessions
- Signing prefixes

The screenshot shows the GitHub repository page for 'cloudflare/gortr'. The repository has 29 commits, 6 branches, 5 releases, and 5 contributors. The latest commit was made on Nov 12, 2018. The repository page includes sections for code, issues, pull requests, wiki, insights, and settings. It also features tabs for rpki, bgp, security, cloudflare, cryptography, prometheus, juniper, cisco, and Manage topics. A 'Clone or download' button is visible at the bottom right.

File	Description	Age
cmd/gortr	Fix ASN parsing on 32-bit platforms	4 months ago
lib	Fixes #99: segfault when incomplete packet	4 months ago
prefixfile	Refactoring:	6 months ago
.gitignore	various cleanups for distribution	6 months ago
.travis.yml	Bump Go versions	4 months ago
Dockerfile	Multi-stage dev dockerization	5 months ago

Diagram



Behind the scene (until January 2019)

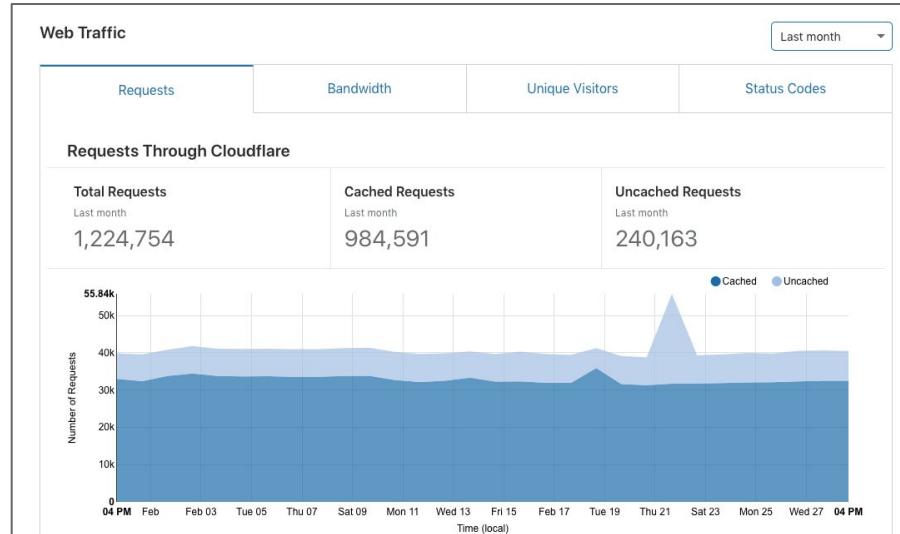
RIPE Validator providing list of prefixes.

Running in a Mesos cluster.

With a cronjob:

- Fetching the data
- Filtering it
(remove > /24 and > /48 and duplicates)
- Signing it
- Making it available to our edge.

<https://rpki.cloudflare.com/rpki.json> was born.



Effects

The question everyone asked us.

How much traffic was affected?

Many invalids. Little traffic in practice
(default or valid less specific).

Except in one place. Few gigabits per seconds displaced due to
geographical more specific.



<https://www.flickr.com/photos/thure/6287816628/>

Signing the routes

Signing the routes

IP space in 5 RIRs (*no twnic/jpnic/cnnic*).

Not a unified experience.

RIR	Features	Ease of use	API
AFRINIC	★	★	★
APNIC	★★	★★	★
ARIN	★★	★★	★★
LACNIC	★	★★★	★
RIPE	★★★	★★★	★★★

Rankings

Features: RRDP, 2 factors, extra info, CA.

Ease of use: steps to sign a ROA, multi user.

API: functional, complete and accessible.

Comparison - AFRINIC

Hard to set up: client TLS certificate to create (BPKI) in order to do RPKI.

Buggy.

No RRDP.

No API.

No auto-renew.

Hosted CA possible.

Extensive certificate informations.

Manage Your RPKI Resources

You can manage your RPKI resources from this page

RPKI Operations

- [List certificates](#)
- [View ROAs](#)
- [View Old ROAs](#)
- [Issue ROA](#)

Add ROA

* Name: Please enter a unique ROA name. Spaces will be replaced by '_'.

Your AS Numbers: Please select your ASN from this list or enter any other valid ASN in the field below.

* AS Number: ASN must be between 0 - 4294967295 in ASPLAIN format. "Reserved" and "Unallocated" ASNs will be rejected.

IPv4 address range: Please select your prefix in the drop down list and click the '+' button, then you can specify the details
 +

IPv6 address range: Please select your prefix in the drop down list and click the '+' button, then you can specify the details
 +

* Not Valid Before (YYYY-MM-DD):

* Not Valid After (YYYY-MM-DD):

Comparison - APNIC

Two factors or client certificate.

RRDP.

Auto-renew.

Allow BGP batch signing.

(slight bugs with big amount of prefixes).

Hosted CA possible.

Draft for API:

<https://www.apnic.net/manage-ip/apnic-services/services-roadmap/public-api-draft-for-members/>



The screenshot shows the APNIC Routes management interface. At the top, a blue header bar contains the title "Routes". Below it, a section titled "Routes" with a blue background and white text provides instructions: "Register your routes in MyAPNIC using the tool below. It will automatically create route objects and RPKI ROAs will also be created at the same time, if the ROA option is enabled (the ROA status will not be updated until then)." A yellow banner below this section says "Import routes" and "BGP announcements associated with your resources but not managed under this tool were found". It includes a "Review & Import from BGP" button and a "Dismiss" button. A modal window titled "Create route" is open in the foreground. It has fields for "Prefix" (Route's prefix. E.g. 203.10.0.0/20), "Origin AS" (Route's origin. E.g. AS123), and "Most specific announcement" (Route's most specific announcement. E.g. /22). Under "ROA" options, "Enabled" is checked. Under "Whois" options, "Enabled" is checked and "Define Whois route attributes" is unchecked. Under "Options", "Notify additional contacts" is unchecked. At the bottom right of the modal are "Cancel" and "Next" buttons.

Comparison - ARIN

Two factors. Separate signing key.

No RRDP.

No auto-renew.

Semi-functional API (add).

Dashboard not easy to find.

Hosted CA possible.

~~Slow rsync update (0000 and 1200 EST).~~

Edit: happens 4 times a day

Some certificate information.

Create a Route Origin Authorization

[Browser Signed](#) [Signed](#)

* denotes required field

*ROA Name:

Any name of your choosing.

*Origin AS:

The AS Number you are authorizing.

*Start Date:

The first date your ROA can be considered valid.

*End Date:

The last date your ROA can be considered valid.

*Prefixes: Address CIDR Max Length

The prefixes you authorize to originate from this AS.

*Private Key:

This key will not be uploaded to ARIN.

Comparison - LACNIC

No two factors. Single user.

No RRDP.

No API.

Auto-renew opt-in.

Allow BGP batch signing.

Based off RIPE.

No Hosted CA.

Some extra info (revoked, path).

Incorrect certificate encoding (BER). High turnover of certificate (few days).

The screenshot shows a web-based form for managing a Resource Certificate (ROA). The form includes fields for 'Name' (empty), 'ASN' (0), 'Valid from' (04/03/2019), and 'Valid until' (04/03/2021). A checkbox labeled 'Automatically extend the validity of the ROA?' is checked. Below these fields is a large text area containing comments and ROA entries:

```
#esto es un comentario  
#ejemplo  
#10.0.0.0/28-30  
#2000::0000/32-34  
  
#recursos autorizados  
  
#131.0.72.0/22-22  
##190 93 240 0/20-20
```

A green 'Save' button is located at the bottom right of the form.

Comparison - RIPE

Two factors.

RRDP.

Auto-renew.

Nice API.

Allow BGP batch signing.

No Hosted CA (theoretically).

No extra information. But history.

Incorrect certificate encoding (BER).

AS number	Prefix	Most specific length allowed	Affects
<input type="text" value="AS Number"/>	<input type="text" value="Prefix"/>	<input type="text" value="Max length"/>	

Automation

We automated prefixes adding on ARIN with a **Salt state**.

Two secrets to store (API key and signing key).

Cannot delete or list via API: very prone to mistakes if user wants to reduce the amount of ROA files.

```
def _format_payload(roas, signature):
    template = """-----BEGIN ROA REQUEST-----
{roas}
-----END ROA REQUEST-----
-----BEGIN SIGNATURE-----
{signature}
-----END SIGNATURE-----
"""
    payload = template.format(
        roas=roas, signature="\n".join(textwrap.wrap(signature, width=64))
    )
    return payload

def _make_roa(name, asn, t, start_val, end_val, prefix, length, maxlenlength):
    template = (
        '1|{time}|{name}|{asn}|{start_val}|{end_val}|{prefix}|{length}|{maxlength}|'
    )
    time_str = calendar.timegm(t.timetuple())
    start_val_str = start_val.strftime(_TIME_FORMAT)
    end_val_str = end_val.strftime(_TIME_FORMAT)
    roa = template.format(
        time=time_str,
        name=name,
        asn=asn,
        start_val=start_val_str,
        end_val=end_val_str,
        prefix=prefix,
        length=length,
        maxlenlength=maxlenlength,
    )
    return roa

def _sign(pkey, roas):
    signature = pkey.sign(roas.encode('utf-8'), padding.PKCS1v15(), hashes.SHA256())
    return base64.b64encode(signature).decode('utf-8')
```

Validator

Why making a validator?

First release of Routinator in November 2018.

We were still using RIPE Validator.

We wanted something more custom: with monitoring and RRDP.

By building it in Go:

- Many APIs and easy for concurrency
- Community doing cryptography
- Cloudflare uses Go a lot (cfssl, sidh, etc.)

Challenges

Juniper bugs: Routing Validation disabled.

Difficulties: rsync, BER encoded instead of DER

3) a subjectPublicKeyInfo [RFC5280] in DER format [X.509],
encoded in Base64 (see Section 4 of [RFC4648]). To avoid long
lines, <CRLF> or <LF> line breaks MAY be inserted into the
Base64-encoded string.

where the URI section is comprised of one of more of the ordered
sequence of:

Cloudflare's RPKI Toolkit

Sets of libraries and tools written in Go.

Including *OctoRPKI* 🐦

Cloudflare's RPKI Toolkit

Libraries

- CER/ROA/MFT decoder
- PKI manager (exploring, validating)
- RRDP/Rsync fetcher



Cloudflare's RPKI Toolkit

Software

- Local validator (without RRDP/Rsync)
- API tools for a distributed version without filesystem
- OctoRPKI

OctoRPKI - Features (1/2)

- Decodes TAL/CER/ROA/MFT
- Explore via Manifest or directory.
- RRDP support (and failover to Rsync)
- Monitoring (Prometheus and JSON API which includes logs)
- Dockerizeable
- Handle stability (generate file when done)

OctoRPKI - Features (2/2)

- Full compatibility with GoRTR (including signing the JSON file)
- Server + caching options for generated file (CDN friendly)
- Configuration options
 - Disable/Enable components
 - Modes (server, one-off)
- ~5-15 minutes for a full cold-start sync

OctoRPKI - Compute footprint

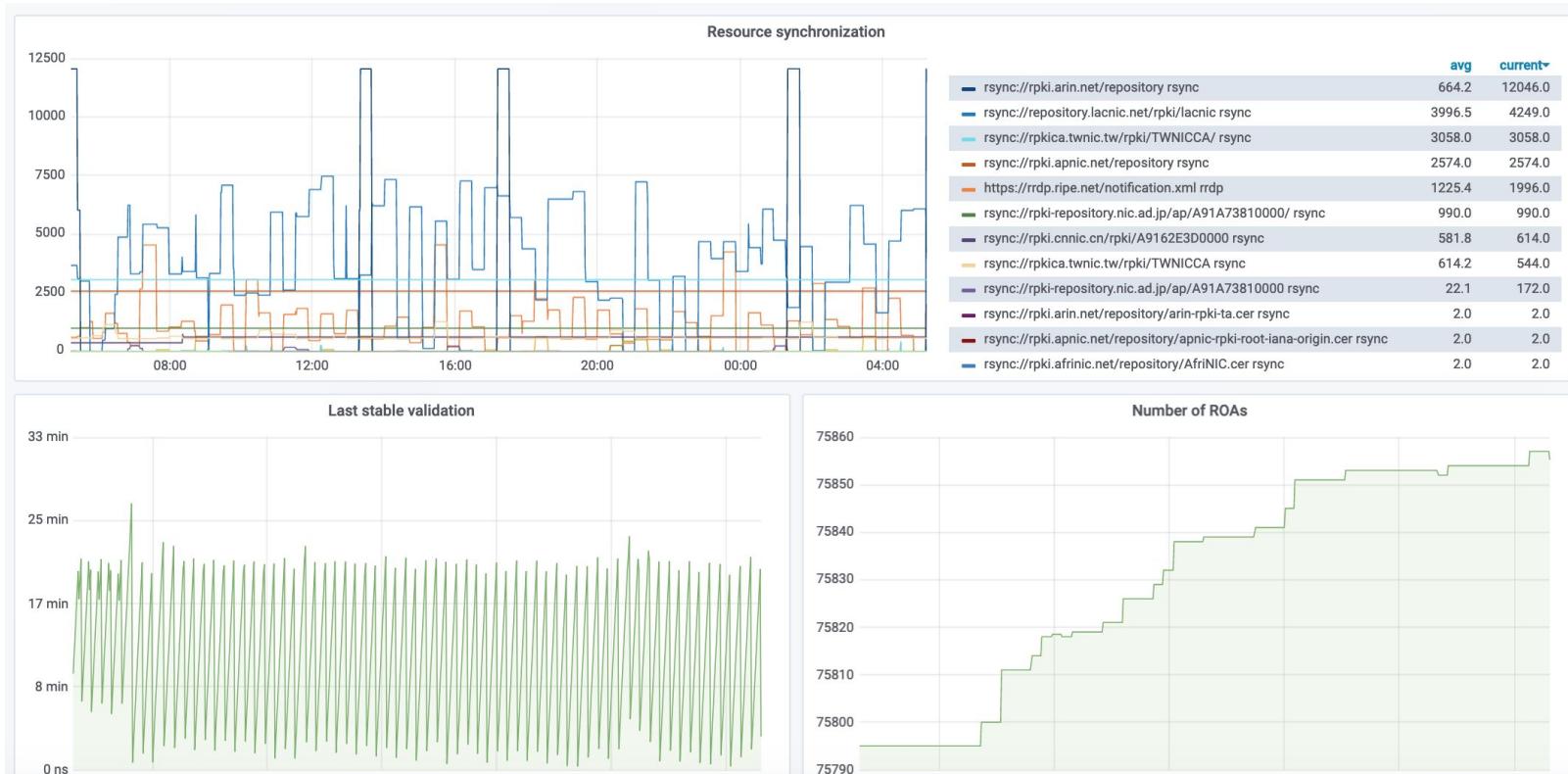


OctoRPKI - Compute footprint

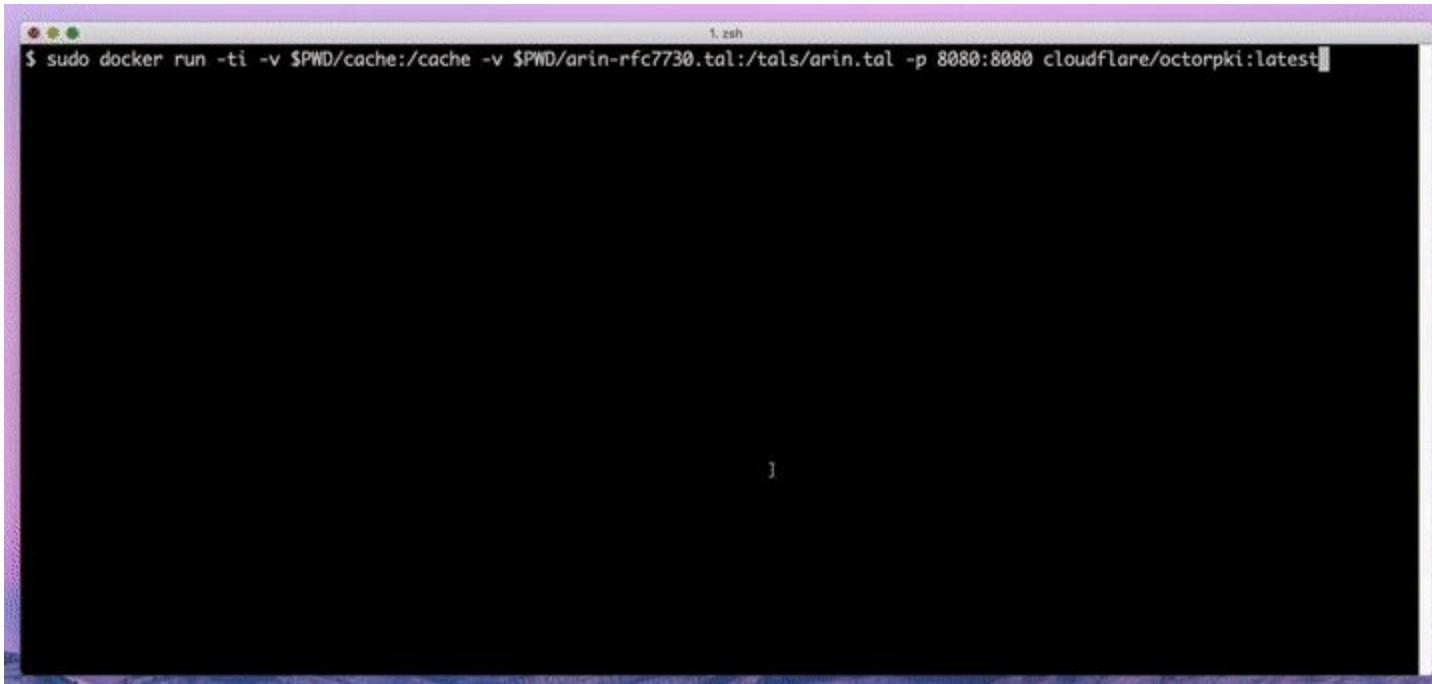
Why 800-1500MB of RAM?

- Buffer RRDP files. Not on disk
- Keeping every resource in memory
- Webserver + metrics
- *Possible Kubernetes overhead (file volume)*

Monitoring



OctoRPKI - What it looks like

A screenshot of a terminal window titled "1. zsh". The window has a pink border and a black background. At the top, there is a command line prompt: "\$ sudo docker run -ti -v \$PWD/cache:/cache -v \$PWD/arin-rfc7730.tal:/tals/arin.tal -p 8080:8080 cloudflare/octorpk1:latest". Below the prompt, there is a large black rectangular area representing the output of the command, which is currently empty.

```
$ sudo docker run -ti -v $PWD/cache:/cache -v $PWD/arin-rfc7730.tal:/tals/arin.tal -p 8080:8080 cloudflare/octorpk1:latest
```

```
1.docker
INFO[0002] Rsync sync rsync://rpki.arin.net/repository/arin-rpki-ta.cer
INFO[0002] Rsync sync rsync://rpki.ripe.net/ta/ripe-ncc-ta.cer
INFO[0004] Rsync sync rsync://repository.lacnic.net/rpki/lacnic/rta-lacnic-rpki.cer
INFO[0007] Rsync sync rsync://rpki.afrinic.net/repository/AfriNIC.cer
ERROR[0012] Error exploring file: open cache/rpki.afrinic.net/repository/04E8B0D80F4D11E0B657D8931367AE7D/62gPOPXWxxu0sQa4vQZYUBLaMbY.mft: no such file or directory
ERROR[0012] Error exploring file: open cache/rpki.apnic.net/repository/838DB214166511E2B3BC286172FD1FF2/C5zKkN0Neo03ZmsZIX_g2EA3t6I.mft: no such file or directory
ERROR[0012] Error exploring file: open cache/rpki.arin.net/repository/arin-rpki-ta.mft: no such file or directory
ERROR[0012] Error exploring file: open cache/repository.lacnic.net/rpki/lacnic/rta-lacnic-rpki.mft: no such file or directory
ERROR[0012] Error exploring file: open cache/rpki.ripe.net/repository/ripe-ncc-ta.mft: no such file or directory
INFO[0012] Still exploring. Revalidating now
INFO[0012] RRDP sync https://rrdp.apnic.net/notification.xml
INFO[0012] RRDP: Downloading root notification https://rrdp.apnic.net/notification.xml
INFO[0014] RRDP: https://rrdp.apnic.net/notification.xml Downloading snapshot at: https://rrdp.apnic.net/fa64523b-7381-4fd4-9eb9-b1233b30f503/83968/snapshot.xml
INFO[0064] RRDP sync https://rrdp.ripe.net/notification.xml
INFO[0064] RRDP: Downloading root notification https://rrdp.ripe.net/notification.xml
INFO[0064] RRDP: https://rrdp.ripe.net/notification.xml Downloading snapshot at: https://rrdp.ripe.net/8ab7553b-b124-4717-b20c-105ada07476c/866/snapshot.xml
INFO[0177] Rsync sync rsync://repository.lacnic.net/rpki/lacnic
INFO[0241] Rsync sync rsync://rpki.arin.net/repository
INFO[0298] Rsync sync rsync://rpki.afrinic.net/repository
INFO[0309] Rsync sync rsync://rpki.apnic.net/repository
```

Validator

```
127.0.0.1:8080/infos x 127.0.0.1:8080/output.json x +
<--> C ① 127.0.0.1:8080/infos
{
  "uri": "rsync://repository.lacnic.net/rpki/lacnic/rta-lacnic-rpki.cer",
  "file-count": 1,
  "iteration": 1,
  "errors": 0,
  "duration": 2.9613296,
  "last-fetch": 1550342002
},
{
  "uri": "rsync://rpki.afrinic.net/repository/AfriNIC.cer",
  "file-count": 1,
  "iteration": 1,
  "errors": 0,
  "duration": 4.3905323,
  "last-fetch": 1550342006
},
{
  "uri": "https://rrdp.apnic.net/notification.xml",
  "file-count": 6734,
  "iteration": 1,
  "errors": 0,
  "duration": 52.0518939,
  "last-fetch": 15503420590,
  "rrdp-serial": 83968,
  "rrdp-sessionid": "fa64523b-7381-4fd4-9eb9-b1233b30f503",
  "rrdp-last-file": "https://rrdp.apnic.net/fa64523b-7381-4fd4-9eb9-b1233b30f503/83968/snapshot.xml"
},
{
  "uri": "file:///var/lib/docker/containers/127.0.0.1/127.0.0.1:8080/infos"
}
```

API

```
[INFO@0842] RRDP sync https://rrdp.apnic.net/notification.xml
[INFO@0842] RRDP: Downloading root notification https://rrdp.apnic.net/notification.xml
[INFO@0844] RRDP: https://rrdp.apnic.net/notification.xml has 0 deltas to parse (cur: 83980, last: 83980)
[INFO@0844] RRDP: finished downloading https://rrdp.apnic.net/notification.xml. Last serial 83981
[INFO@0844] RRDP sync https://rrdp.apnic.net/notification.xml
[INFO@0844] RRDP: Downloading root notification https://rrdp.apnic.net/notification.xml
[INFO@0845] RRDP: https://rrdp.apnic.net/notification.xml has -1 deltas to parse (cur: 83980, last: 83981)
[INFO@0845] RRDP: finished downloading https://rrdp.apnic.net/notification.xml. Last serial 83981
[INFO@0845] RRDP sync https://rrdp.ripe.net/notification.xml
[INFO@0845] RRDP: Downloading root notification https://rrdp.ripe.net/notification.xml
[INFO@0845] RRDP: https://rrdp.ripe.net/notification.xml has -1 deltas to parse (cur: 867, last: 868)
[INFO@0845] RRDP: finished downloading https://rrdp.ripe.net/notification.xml. Last serial 868
[INFO@0845] RRDP sync https://rpki.cnnic.cn/rrdp/notify.xml
[INFO@0845] RRDP: Downloading root notification https://rpki.cnnic.cn/rrdp/notify.xml
[INFO@0847] RRDP: https://rpki.cnnic.cn/rrdp/notify.xml has 0 deltas to parse (cur: 253274, last: 253274)
[INFO@0847] RRDP: finished downloading https://rpki.cnnic.cn/rrdp/notify.xml. Last serial 253275
[INFO@0847] Rsync sync rsync://rpki.arin.net/repository
[INFO@0861] Rsync sync rsync://rpki.arininc.net/repository
[INFO@0867] Rsync sync rsync://rpki.ripe.net/ta/ripe-ncc-ta.cer
[INFO@0869] Rsync sync rsync://repository.lacnic.net/rpki/lacnic
[INFO@0883] Rsync sync rsync://rpki-repository.mic.ad.jp/ap/A91A73810000
[INFO@0885] Rsync sync rsync://rpkica.twnic.tw/rpki/TWNICCA
[WARN@0885] Rsync sync rsync://rpkica.twnic.tw/rpki/TWNICCA: port 12345 is invalid: <nil>
[INFO@0885] Rsync sync rsync://rpkica.twnic.tw/rpki/TWNICCA: port 12345 is invalid: <nil>
```

```
[{"metaData": {"connId": "7e7658", "generated": "1550343058", "valid": "1550346658", "signature": "3a045022100b5dbf4fbdb878b41bd6f5687744d2c3bc65cfce05fb56b100b00220a7b6aeac99b8db81d817faa0ef2fd2ad5f54b7fe709954296c938c2b", "signatureDate": "2019-05-10T00:36:09Z", "id": "10L55a283877a87d45636f970c86bf5ea0f65c2511c77e8cc0202fcfd4692b166b9e4c7cb2fb41e0cad40f51cd03d0422de2ee17b17d57ed", "coas": [{"prefix": "21.106.62.0/24", "maxLength": "24", "asn": "AS116177", "ta": ":"}, {"prefix": "192.168.1.0/24", "maxLength": "24", "asn": "AS22755", "ta": ":"}, {"prefix": "192.168.216.0/22", "maxLength": "24", "asn": "AS265751", "ta": ":"}, {"prefix": "186.179.112.0/20", "maxLength": "24", "asn": "AS22755", "ta": ":"}, {"prefix": "190.192.224.0/20", "maxLength": "24", "asn": "AS927755", "ta": ":"}, {"prefix": "190.192.224.0/24", "maxLength": "24", "asn": "AS927755", "ta": ":"}, {"prefix": "172.249.84.0/24", "maxLength": "24", "asn": "AS113353", "ta": ":"}, {"prefix": "77.74.76.0/24", "maxLength": "24", "asn": "AS206850", "ta": ":"}, {"prefix": "37.130.198.0/24", "maxLength": "24", "asn": "AS199386", "ta": ":"}, {"prefix": "37.130.199.0/24", "maxLength": "24", "asn": "AS199386", "ta": ":"}, {"prefix": "37.130.199.0/25", "maxLength": "24", "asn": "AS199386", "ta": ":"}, {"prefix": "37.130.199.0/27", "maxLength": "24", "asn": "AS199386", "ta": ":"}, {"prefix": "20a04dc1/32", "maxLength": "32", "asn": "AS199386", "ta": ":"}, {"prefix": "20a04dc1/32", "maxLength": "32", "asn": "AS199386", "ta": ":"}, {"prefix": "192.168.224.0/23", "maxLength": "24", "asn": "AS203489", "ta": ":"}, {"prefix": "183.212.68.0/22", "maxLength": "22", "asn": "AS203489", "ta": ":"}, {"prefix": "185.214.0.0/22", "maxLength": "22", "asn": "AS203489", "ta": ":"}, {"prefix": "2a0b19c0/21", "maxLength": "22", "asn": "AS203489", "ta": ":"}, {"prefix": "132.147.56.0/24", "maxLength": "24", "asn": "AS2787", "ta": ":"}, {"prefix": "192.151.50.0/24", "maxLength": "24", "asn": "AS34864", "ta": ":"}, {"prefix": "122.129.83.0/24", "maxLength": "24", "asn": "AS38264", "ta": ":"}, {"prefix": "203.128.15.0/24", "maxLength": "24", "asn": "AS38264", "ta": ":"}, {"prefix": "203.128.16.0/24", "maxLength": "24", "asn": "AS38264", "ta": ":"}, {"prefix": "185.154.128.0/24", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "185.154.128.0/25", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "185.154.128.0/27", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "185.154.128.0/28", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "185.252.128.0/22", "maxLength": "22", "asn": "AS58059", "ta": ":"}, {"prefix": "185.252.129.0/24", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "26000.9000120d51/48", "maxLength": "48", "asn": "AS58059", "ta": ":"}], "coas": [{"prefix": "192.168.1.0/24", "maxLength": "24", "asn": "AS116177", "ta": ":"}, {"prefix": "192.168.216.0/22", "maxLength": "24", "asn": "AS265751", "ta": ":"}, {"prefix": "186.179.112.0/20", "maxLength": "24", "asn": "AS22755", "ta": ":"}, {"prefix": "190.192.224.0/20", "maxLength": "24", "asn": "AS927755", "ta": ":"}, {"prefix": "190.192.224.0/24", "maxLength": "24", "asn": "AS927755", "ta": ":"}, {"prefix": "172.249.84.0/24", "maxLength": "24", "asn": "AS113353", "ta": ":"}, {"prefix": "77.74.76.0/24", "maxLength": "24", "asn": "AS206850", "ta": ":"}, {"prefix": "37.130.198.0/24", "maxLength": "24", "asn": "AS199386", "ta": ":"}, {"prefix": "37.130.199.0/24", "maxLength": "24", "asn": "AS199386", "ta": ":"}, {"prefix": "37.130.199.0/25", "maxLength": "24", "asn": "AS199386", "ta": ":"}, {"prefix": "37.130.199.0/27", "maxLength": "24", "asn": "AS199386", "ta": ":"}, {"prefix": "20a04dc1/32", "maxLength": "32", "asn": "AS199386", "ta": ":"}, {"prefix": "20a04dc1/32", "maxLength": "32", "asn": "AS199386", "ta": ":"}, {"prefix": "192.168.224.0/23", "maxLength": "24", "asn": "AS203489", "ta": ":"}, {"prefix": "183.212.68.0/22", "maxLength": "22", "asn": "AS203489", "ta": ":"}, {"prefix": "185.214.0.0/22", "maxLength": "22", "asn": "AS203489", "ta": ":"}, {"prefix": "2a0b19c0/21", "maxLength": "22", "asn": "AS203489", "ta": ":"}, {"prefix": "132.147.56.0/24", "maxLength": "24", "asn": "AS2787", "ta": ":"}, {"prefix": "192.151.50.0/24", "maxLength": "24", "asn": "AS34864", "ta": ":"}, {"prefix": "122.129.83.0/24", "maxLength": "24", "asn": "AS38264", "ta": ":"}, {"prefix": "203.128.15.0/24", "maxLength": "24", "asn": "AS38264", "ta": ":"}, {"prefix": "203.128.16.0/24", "maxLength": "24", "asn": "AS38264", "ta": ":"}, {"prefix": "185.154.128.0/24", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "185.154.128.0/25", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "185.154.128.0/27", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "185.154.128.0/28", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "185.252.128.0/22", "maxLength": "22", "asn": "AS58059", "ta": ":"}, {"prefix": "185.252.129.0/24", "maxLength": "24", "asn": "AS58059", "ta": ":"}, {"prefix": "26000.9000120d51/48", "maxLength": "48", "asn": "AS58059", "ta": ":"}]}]
```

```
"counts": 76058,
"generated": "1550343058",
"valid": "1550346658",
"signature": "3045022100b5db8f44bdb78b418d6ef6ec55f687794ed723c63c63cef0705fb
e65ba10db1002202a7b6aeec89bb8bd845b481f7aa20ebf2fd2a0df54b1bcf70995429f3e39c82e",
"signatureDate": "3045022100f0368e9cdf510c55ae28387a87fbdcc415636f97c86fbe5af0
65c29514cf77e8c02202fcfd496a9b2c166b9e4c7cb2f410ecad406f1c03d05024dce2ee71b9177d
e5"
},
"roas": [
{
  "prefix": "216.59.62.0/24",
  "maxLength": 24,
  "asn": "AS36167",
  "ta": ""
},
{
  "prefix": "91.204.44.0/22"
}
]
$ cat output.json | jq '.roas | length'
76058
$
```

OctoRPKI - Demo (videos)

Real-time:

<https://www.youtube.com/watch?v=yykkoZxz6gE>



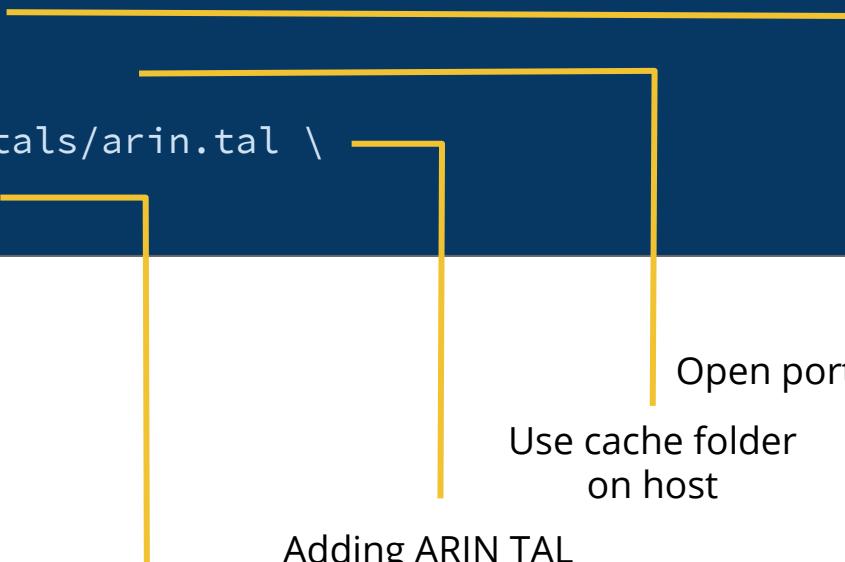
Speed-up:

<https://www.youtube.com/watch?v=OQwtf2m2hxU>



OctoRPKI - Run it yourself

```
$ docker run -ti \  
  -p 8080:8080 \  
  -v $PWD/cache:/cache \  
  -v $PWD/tals/arin.tal:/tals/arin.tal \  
  cloudfare/octorpk
```



GoRTR

OctoRPKI does not embed a RTR server. Modular and independence!

Fully compatible with **GoRTR** <https://github.com/cloudflare/gortr>

Signs the prefix list to ensure a safe distribution of the file.

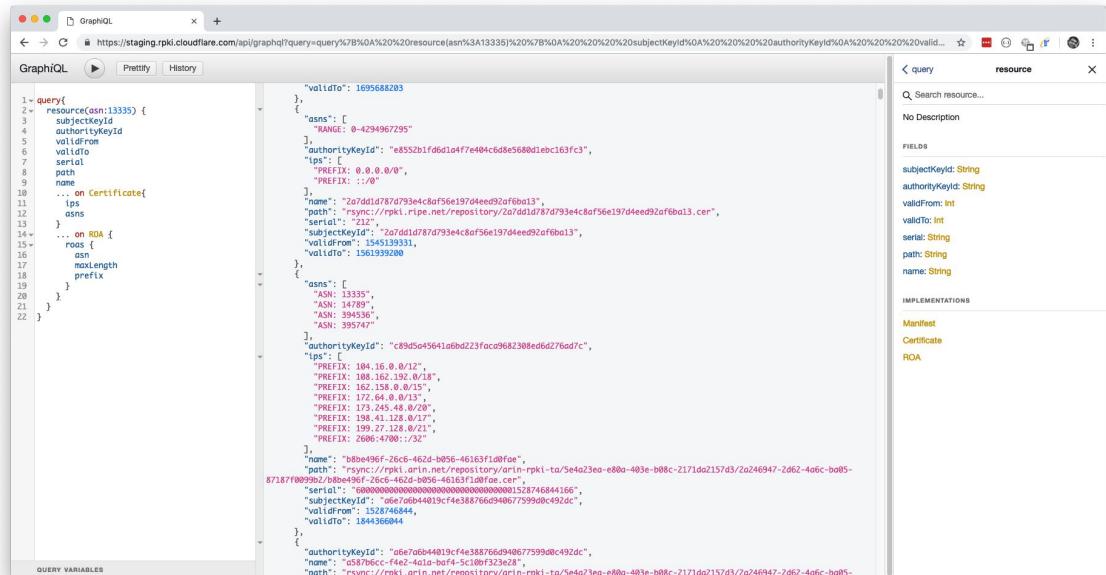
Can run natively on Juniper!

```
$ docker run -ti \
-p 8082:8082 \
-v $PWD/example.pub:/example.pub \
cloudflare/gortr \
--verify.key /example.pub \
--cache https://YOUR_ROA_URL
```

Cloudflare's Internal Version

Is providing:
<https://rpki.cloudflare.com/rpki.json>

But also a **GraphQL** API



Dashboard preview

The screenshot displays the Cloudflare RPKI Portal dashboard in two separate browser windows.

Left Window (Resource List View):

- Header:** RPKI Portal, https://staging.rpki.cloudflare.com/?ta=
- Search Bar:** Enter an IP prefix
- Filter Buttons:** TRUST ANCHOR: All, KEY ID: (dropdown), PREFIX: (dropdown)
- Views:** Resource List (selected), Hierarchical View, Address Space View
- Message:** Showing 5 certificates and 0 ROAs
- Certificates Table:**

Key	Name	Trust Anchor	ASNs	IPs
e8552b1fd6d1a4f7e404c6d8e5680d1ebc163fc3	ripe-ncc-ta	RIPE	0-4294967295	0.0.0::/0
fc8a9cb3ed184e17d30eea1e0fa7615ce4b1af47	root trust anchor O=lacnic	LACNIC	0-4294967295	0.0.0::/0
0b9cca90dd0d7a8a37666b19217fe0d84037b7a2	apnic-rpki-root-iana-origin	APNIC	1-4294967295	0.0.0::/0
eb680f38f5d6c71bb4b106b8bd06585012da31b6	Afrinic Root Certificate	Afrinic	0-4294967295	0.0.0::/0

Right Window (Hierarchical View):

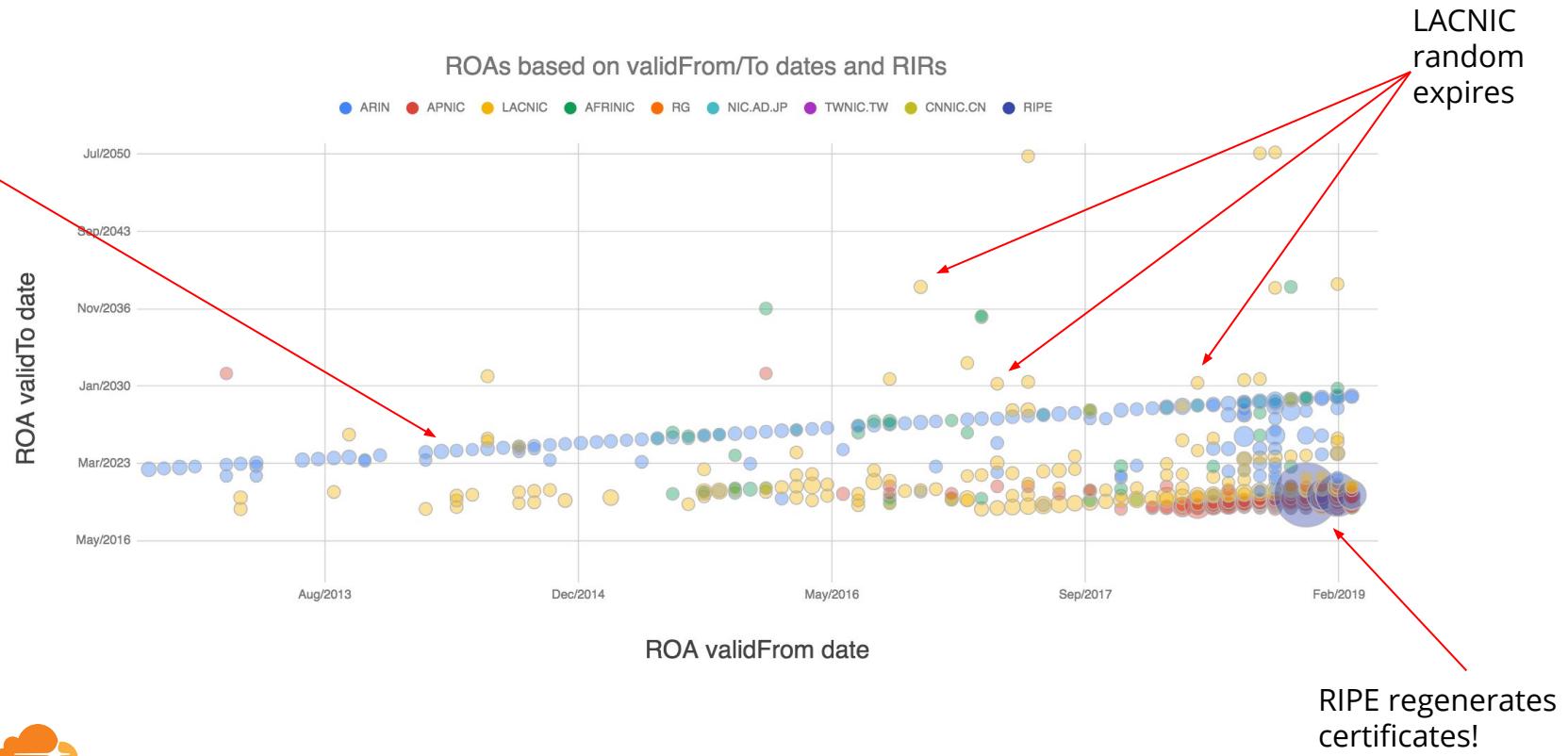
- Header:** RPKI Portal, https://staging.rpki.cloudflare.com/?ta=&prefix=1.1.1.1%2F24
- Search Bar:** Enter an AS number
- Filter Buttons:** TRUST ANCHOR: All, KEY ID: (dropdown), PREFIX: 1.1.1.1/24, ASN: (dropdown)
- Views:** Resource List, Hierarchical View, Address Space View
- Address Space View (IPv4):**

Range	Description
0/0	1.0.0.0/8 (AS173, AS681, AS1221, AS1233, AS1237, AS1250, AS1)
1.1.0/24	1.1.1.0/24 (AS9838, AS24021, AS538610, AS131072, AS131074)
1.1.1.0/24	1.1.1.0/24 (AS1335)

- Switch:** SCALE IP SPACE: Off

Other data - so how fresh are those ROAs?

ARIN uses
ten year
expire



Future projects or ideas

RPKI validation tester using our CDN:

- Using a /23 (/47 IPv4) valid and a /24 (/48 IPv6) invalid

Certificate encoder

ASPA

More toolings and visualizations around RPKI (BGP collection)

Thank you

Questions?

louis@cloudflare.com
@lpoinsig (twitter)

