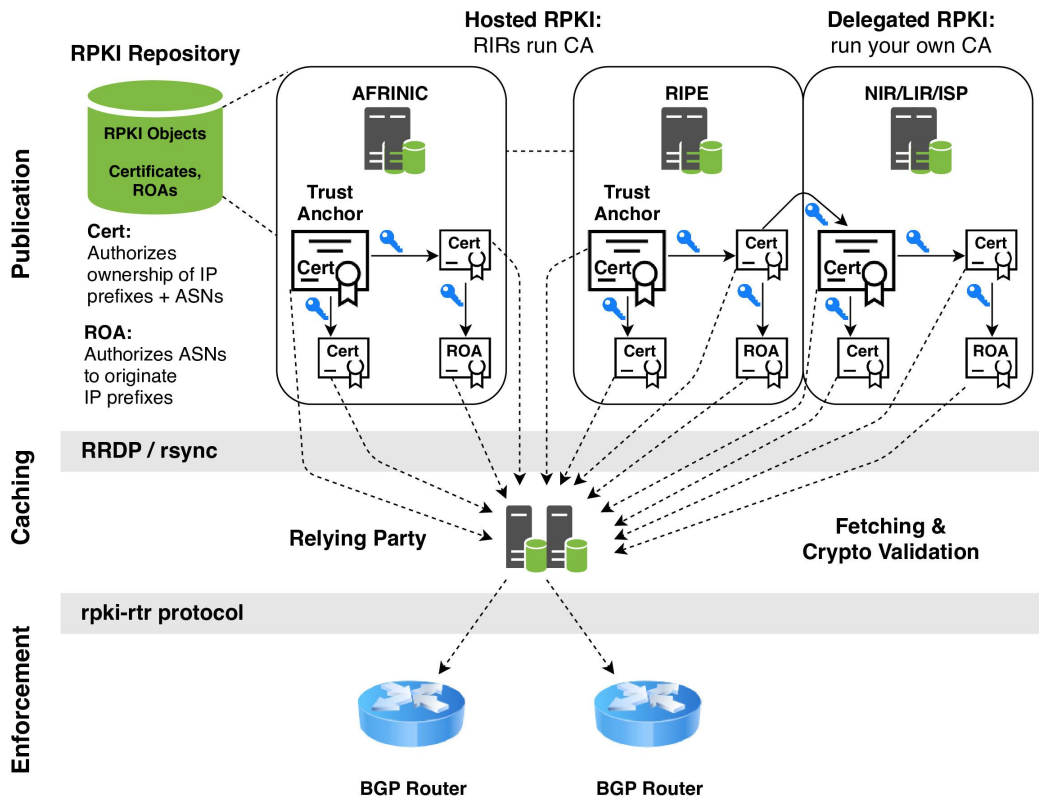


Measuring Relying Parties

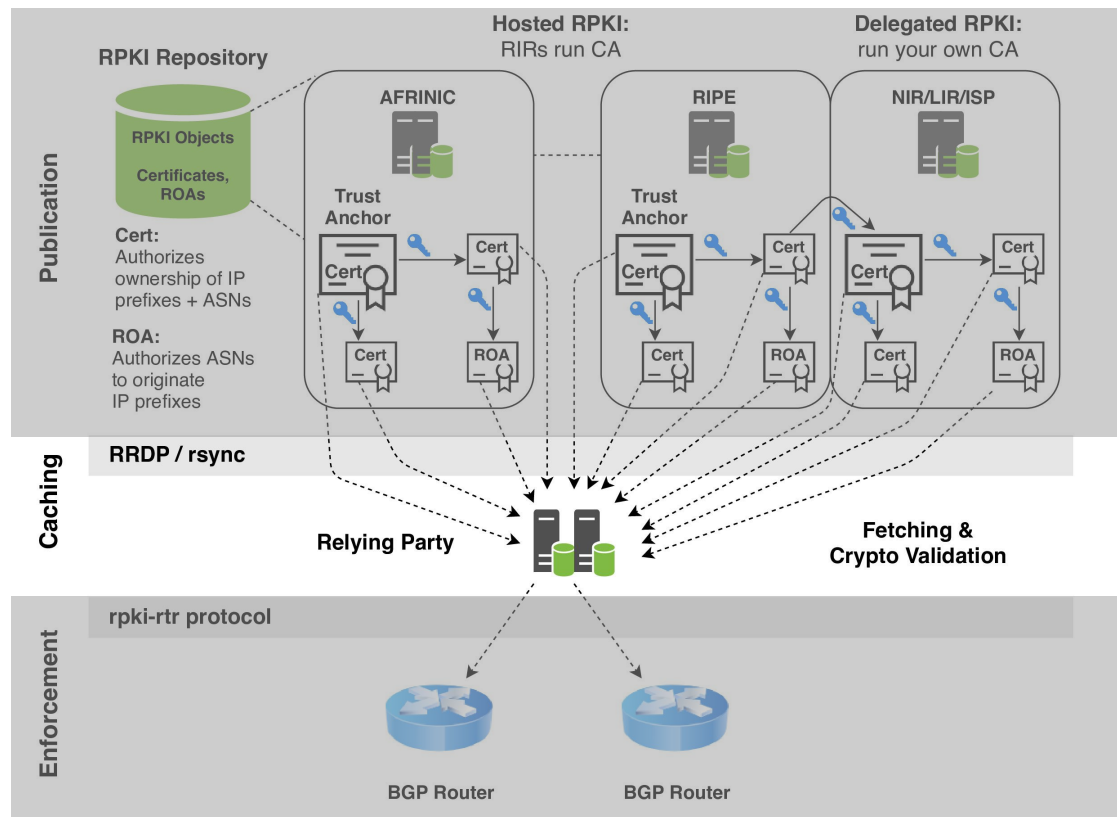
John Kristoff, Randy Bush, Chris Kanich, George Michaelson,
Amreesh Phokeer, Thomas C. Schmidt, Matthias Wählisch

Contact: jtk@depaul.edu

Overview: RPKI Ecosystem & Relying Parties (RPs)



Overview: RPKI Ecosystem & Relying Parties (RPs)



We are interested in monitoring and understanding this part.

Research and Operational Questions

Completeness

Do relying parties fetch data from all publication points?

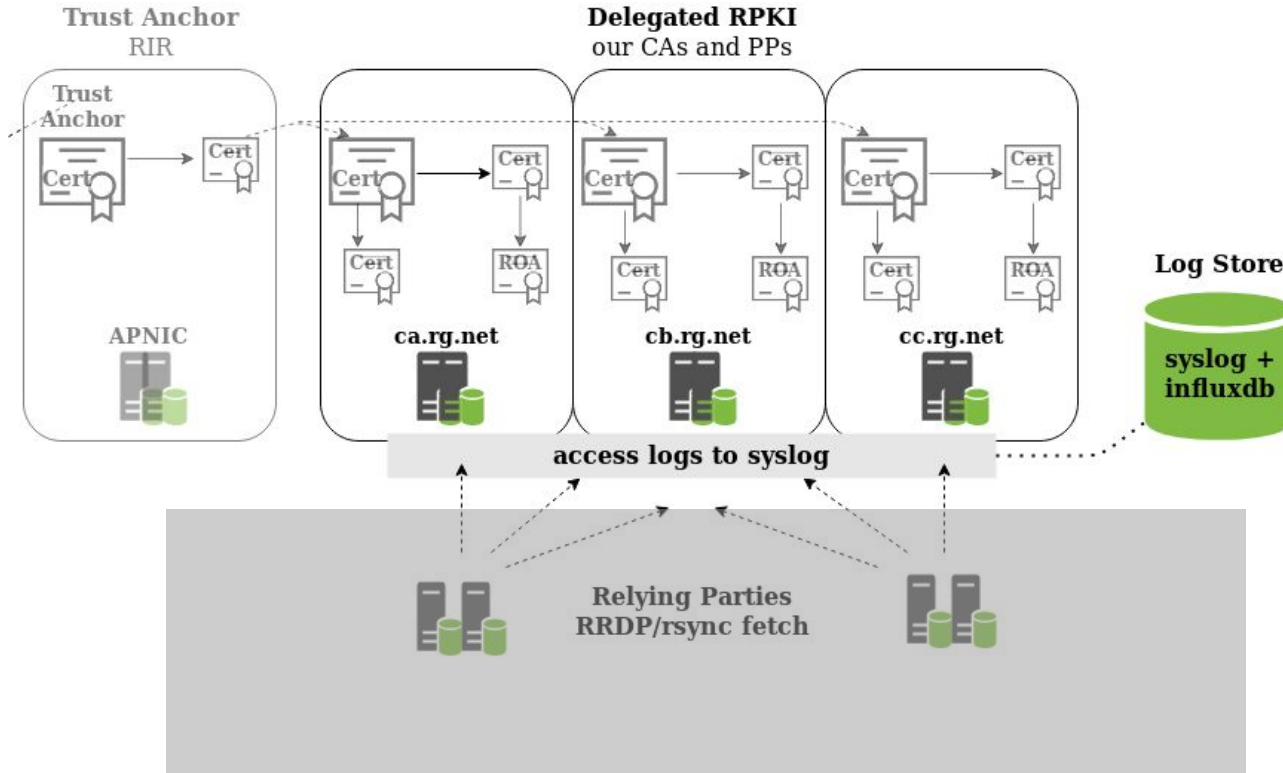
Timeliness

How quickly do relying parties fetch data in the wild?

Spec compliance

Do relying parties implement the specs as expected?

Measurement Framework Synopsis



We run

- (1) a delegated CA + compare access logs w/ logs from other publication points.
- (2) Controlled RPs
- (3) ROA Beacons

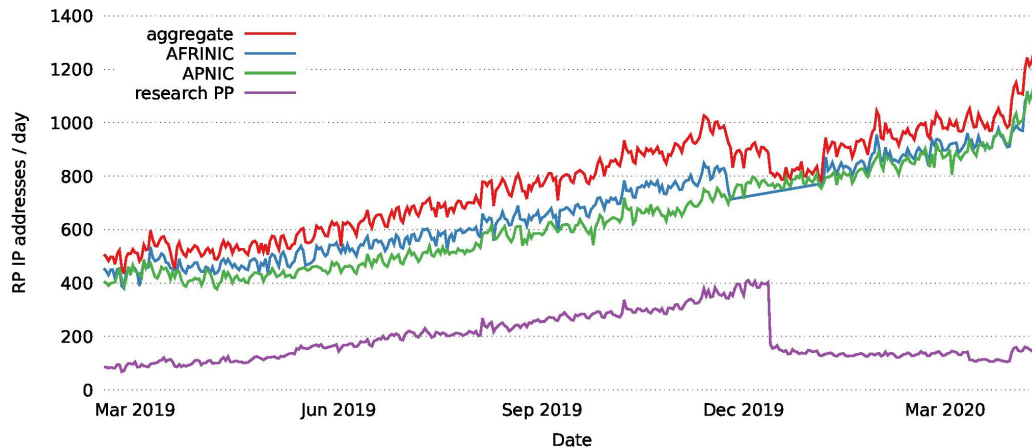
RPs do not fetch from all PPs equally.

Child PPs that support RRDP see rsync only from RPs that do not support RRDP.

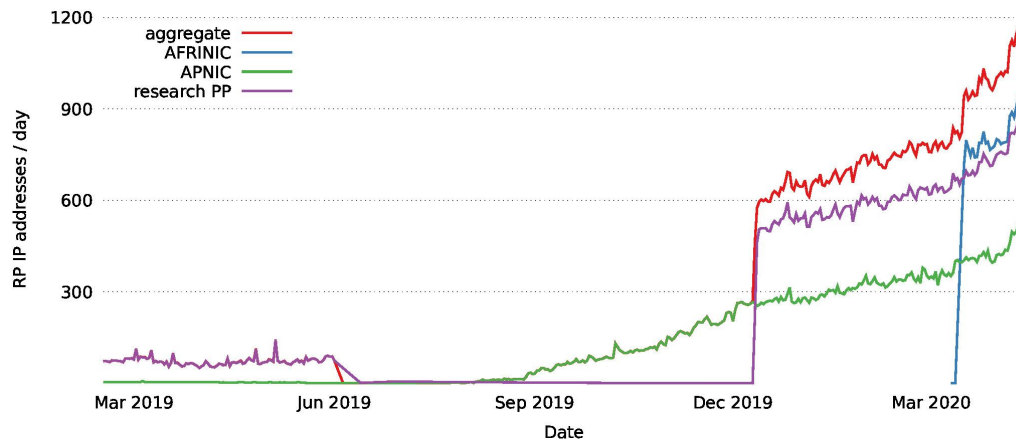
December 2019, research CA/PP “fixes” RRDP.

March 2020 AFRINIC enables RRDP.

rsync



RRDP



Overlap of RP IP Addresses

Surprising number of RPs do not fetch from all PPs, resulting in an incomplete view

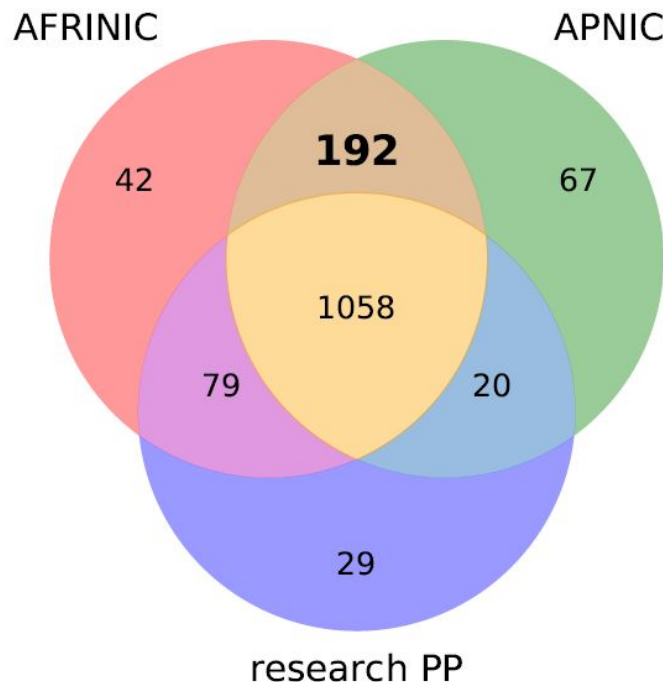
Possible reasons for divergence:

- IPv4 versus IPv6

- Firewalls

- Fetching data via an overlay

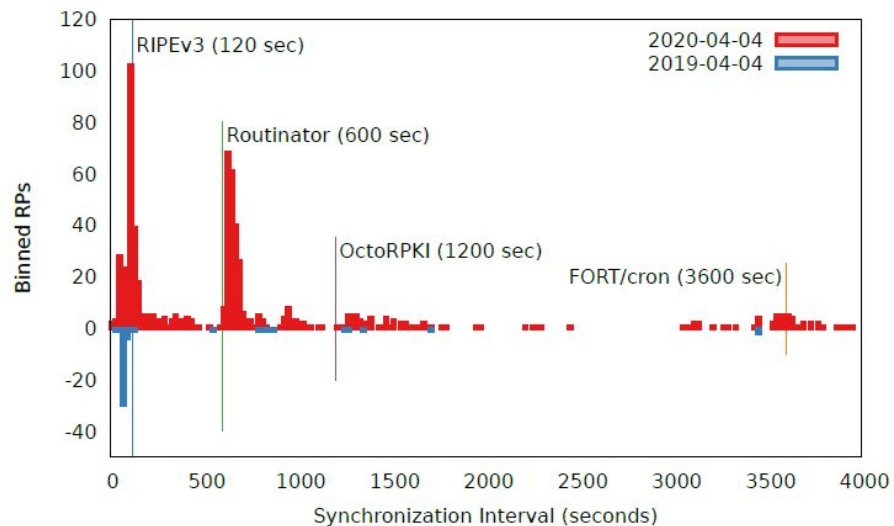
- Inability to follow delegations



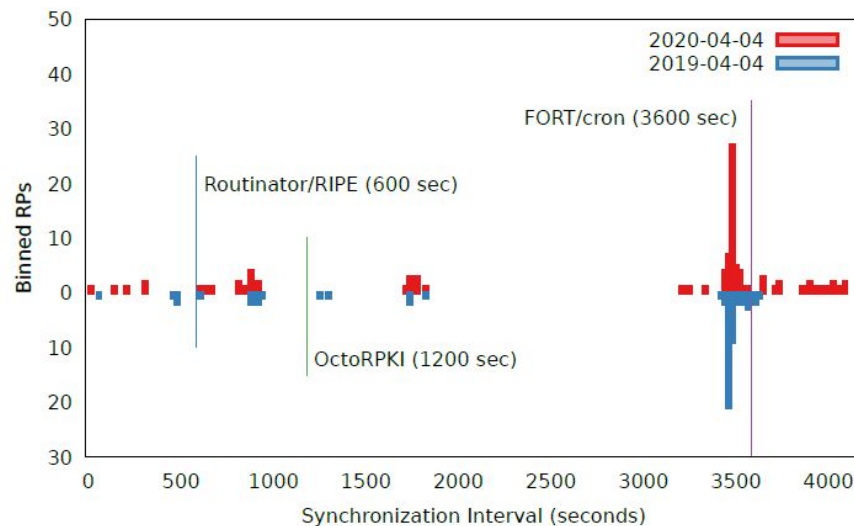
Default Synchronization Interval Time

RP Software	RRDP	rsync
FORT Validator	1 hour	1 hour
OctoRPKI	20 minutes	20 minutes
Routinator	10 minutes	10 minutes
Dragon Labs	1 hour	1 hour
OpenBSD rpki-client	not implemented	1 hour
RIPE NCC Validator 2	1 minute	10 minutes
RIPE NCC Validator 3	2 minutes	10 minutes

Observed RP Synchronization Intervals @ Research PP



(a) RRDP



(b) rsync

Details depend on PP, but RRDP usually more predictable

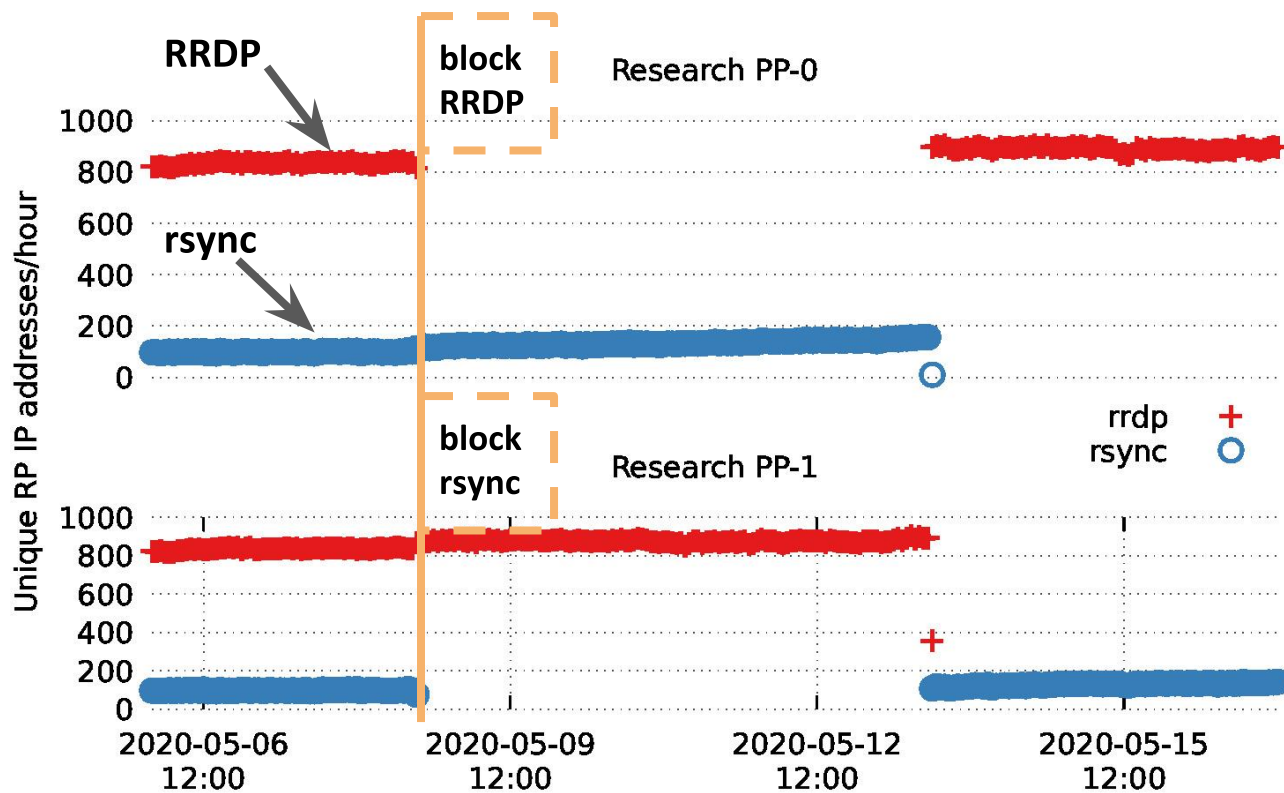
Filtering RRDP vs rsync Experiment

Objective: Verify mandatory support of rsync

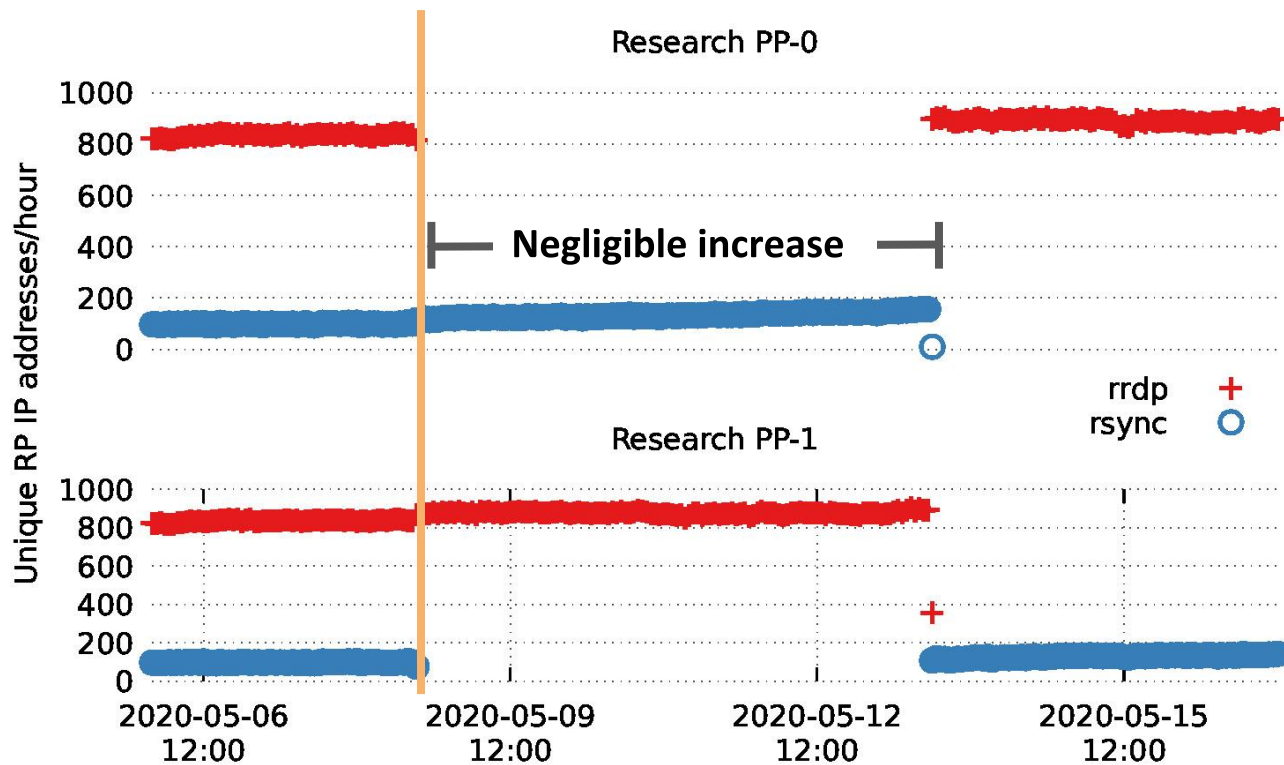
Expectation: When availability of RRDP stops, increase of rsync access because of rsync fall back

Method: Filter RRDP dynamically

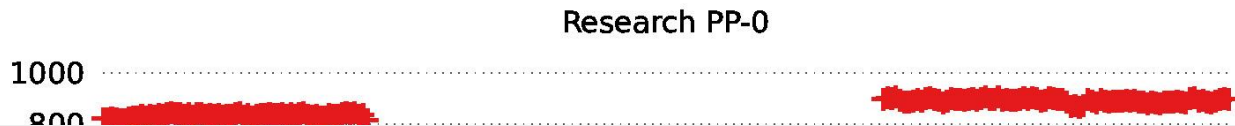
Filtering RRDP vs rsync Experiment



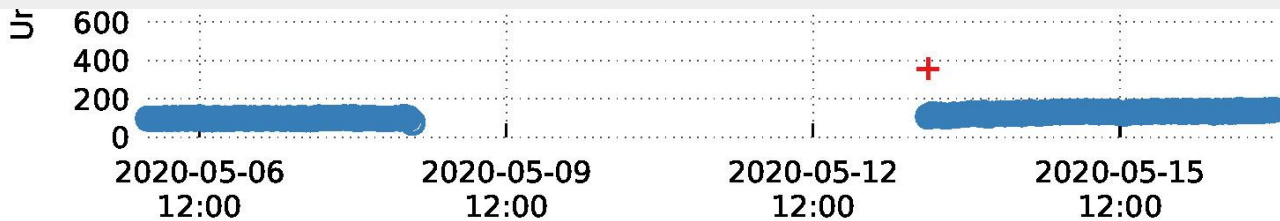
Filtering RRDP vs rsync Experiment



Filtering RRDP vs rsync Experiment



At least Routinator and OctoRPKI
did not fall back to rsync.

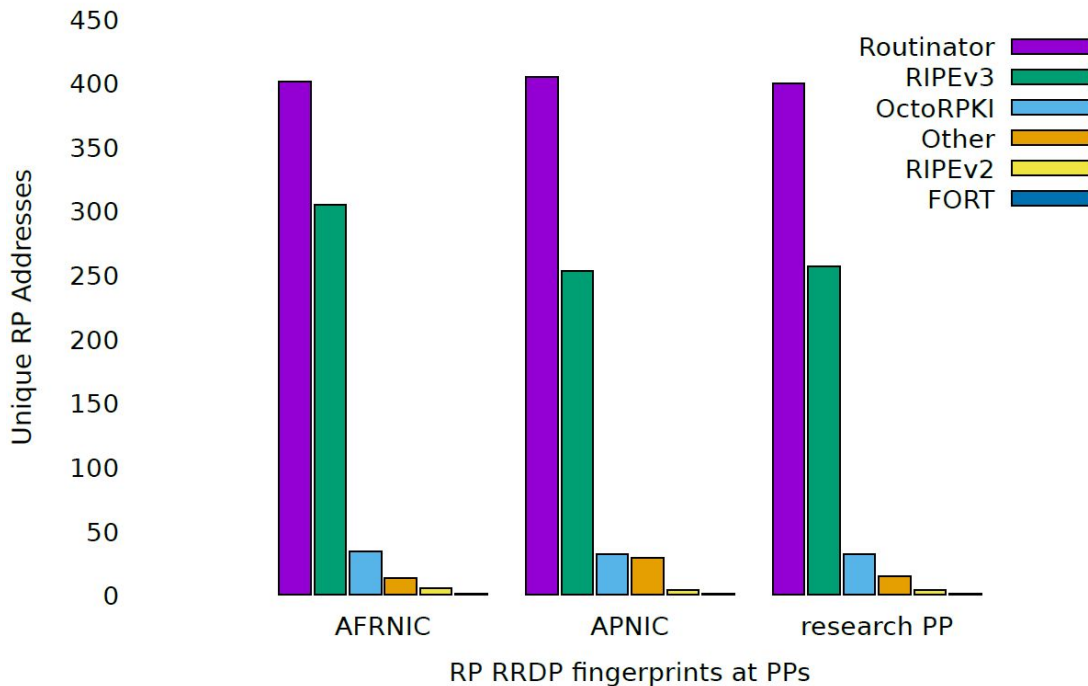


RP Software Popularity, March 2020

We leverage the RRDP
HTTP User-Agent string

Most RPs use both RRDP
and rsync

We see RRDP fingerprints
for ~95% of all RPs that
connect w/ rsync this way



Conclusions

Significant portion of RPs do not obtain a complete or timely view

Migration between RRDP and rsync does not work as defined

Surprisingly hard to map access on multiple PPs to a single RP

Future Work

Deploy delegated CA/PP infrastructure in each RIR region

Evaluate RP software object fetch and validation consistency

Evaluate data on RP cache servers

Evaluate timeliness between BGP updates and ROA updates

Backup

Our Contributions

Reproducible measurement framework to evaluate RPKI data dissemination completeness

Survey RP software, access protocols, and synchronization patterns

Characterize shortcomings in RP synchronization behavior

Identify mismatch in RP software behavior and protocol design expectations