The RPKISPOOL format for materializing RPKI data as used in RPKIViews

Job Snijders
job@bsd.nl

Fedor Vompe
theodor-fedor.vompe@telekom.de

What is RPKIViews.org?

- The goal is to archive ***all*** the world's RPKI data
  - (and ideally, do this in a way so that the data is accessible)
- Inspired by routeviews.org (they make BGP data available as MRT)
- In contrast to BGP, for RPKI there is a *finite* amount of data! It is feasible to capture it **all**! (Or at least, 99.999% of it)
- Not many others seem to be doing this
  - E.g. RIPE NCC makes 1 snapshot per day, all changes during the day are missed!

RPKIViews “gen1” in a nutshell (2020-2025)

- run rpki-client
- `tar -zcf $(date +%Y%m%dT%H%M%ST).tgz /var/cache/rpki-client/*`
- Publish the resulting tarball via rsync & HTTPS

Example file listing:

```
-rw-r--r--  1 root  wheel  117M Dec 31  2020 2020/12/31/rpki-20201231T234230Z.tgz
-rw-r--r--  1 root  wheel  187M Dec 31  2021 2021/12/31/rpki-20211231T234655Z.tgz
-rw-r--r--  1 root  wheel  232M Dec 31  2022 2022/12/31/rpki-20221231T233537Z.tgz
-rw-r--r--  1 root  wheel  267M Dec 31  2023 2023/12/31/rpki-20231231T233407Z.tgz
-rw-r--r--  1 root  wheel  407M Dec 31  2024 2024/12/31/rpki-20241231T233650Z.tgz
-rw-r--r--  1 root  wheel  486M Dec 31  2025 2025/12/31/rpki-20251231T235614Z.tgz
```

RPKIViews “gen1” storage requirements (single node)

```
# du -csh josephine.sobornost.net/2026/*

1.6T    josephine.sobornost.net/2026/01
1.5T    josephine.sobornost.net/2026/02
574G    josephine.sobornost.net/2026/03
3.6T    total
```

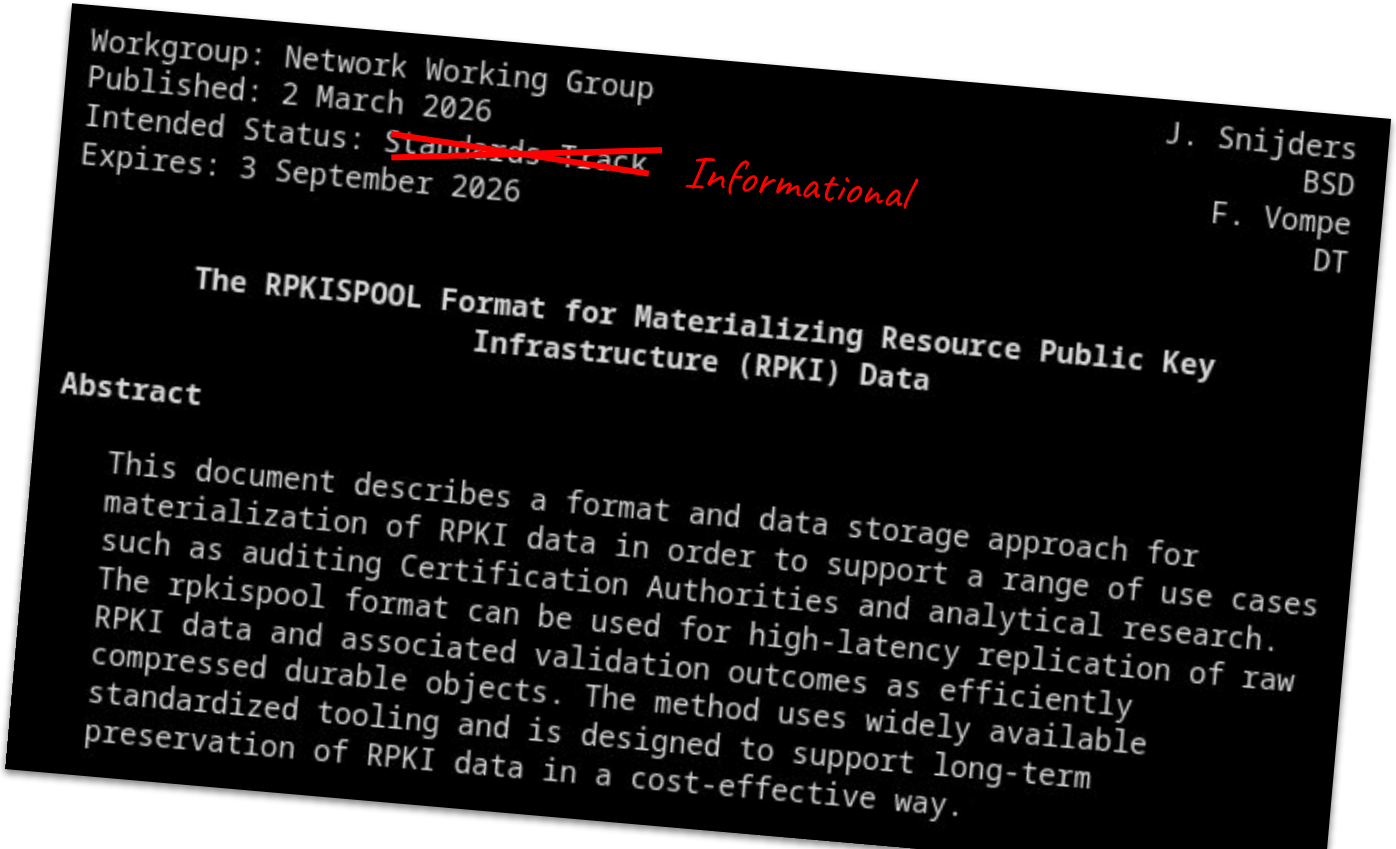
RPKIViews “gen1” storage requirements (multi node)

```
# du -csh */2026

3.6T    josephine.sobornost.net/2026
3.1T    rpkiviews.kerfuffle.net/2026
3.1T    dango.attn.jp/2026
9.7T    total
```

Houston,
we have
a problem.
- Jim Lovell

RPKIViews “gen2” uses RPKISPOOL (since 2026/1/1)



Workgroup: Network Working Group
Published: 2 March 2026
Intended Status: ~~Standards Track~~ Informational
Expires: 3 September 2026

J. Snijders
BSD
F. Vompe
DT

The RPKISPOOL Format for Materializing Resource Public Key Infrastructure (RPKI) Data

Abstract

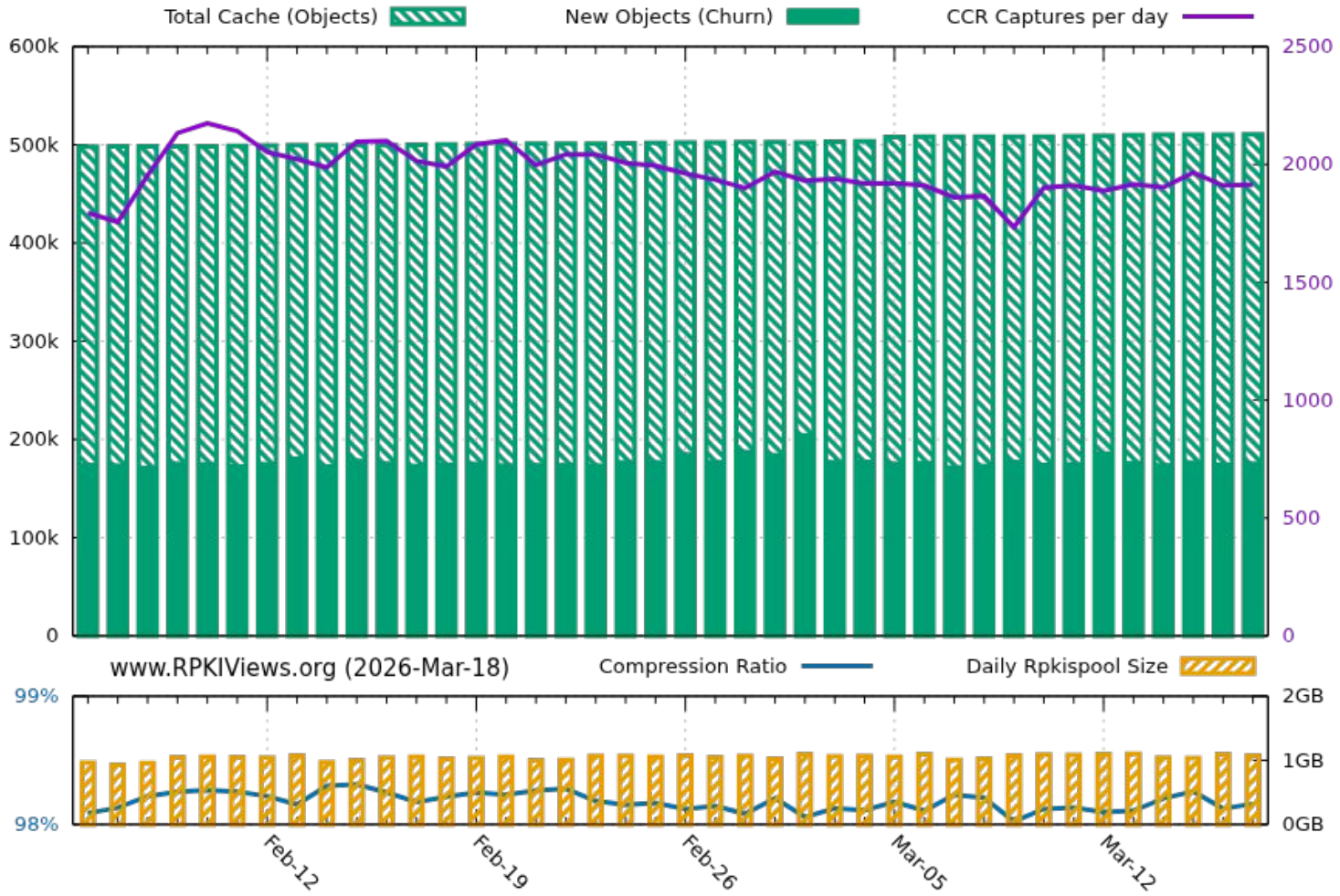
This document describes a format and data storage approach for materialization of RPKI data in order to support a range of use cases such as auditing Certification Authorities and analytical research. The rpkispool format can be used for high-latency replication of raw RPKI data and associated validation outcomes as efficiently compressed durable objects. The method uses widely available standardized tooling and is designed to support long-term preservation of RPKI data in a cost-effective way.

RPKISPOOL
Radio Shack
HI OUT
SPEED
LO IN
SELECTOR
OFF
TAPE
INPUT
STEREO
ADD
MONO
LISTENING VOLUME
FAST WINDING
PLAY OR RECORD
STOP
RECORDING VOLUME
RADIO/PHONO
MICROPHONE
NORMAL

What is the RPKISPOOL method?

- Uses standardized tooling: tar (POSIX), Zstandard (RFC), CCR (draft)
- Every day starts with a checkpoint (`YYYYMMDD-initstate.tar.zst`)
- Throughout the day, multiple nodes apply *Change Data Capture*
- Every day, a compactor process gathers, deduplicates, and compresses all yesterday's data: (YYYYMMDD-rpkispool.tar.zst)

… and the results are awesome …


Total Cache (Objects)
New Objects (Churn)
CCR Captures per day
600k
500k
400k
300k
200k
100k
0
2500
2000
1500
1000
500
0
www.RPKIViews.org (2026-Mar-18)
Compression Ratio
Daily Rpkispool Size
99%
98%
2GB
1GB
0GB
Feb-12
Feb-19
Feb-26
Mar-05
Mar-12

Statistics

- Currently 12 gatherer nodes contributing to the daily rpkispool
  - Some gatherer use RRDP + Rsync
  - Some gatherers use only Rsync
  - Some gatherers reset their RRDP state before each run
  - Some are IPv4/IPv6 dual-stack, some are not
  - Diverse geographies: Sydney, Bangalore, Amsterdam, Zurich, and more…
- All data from all nodes is losslessly aggregated/compressed
  - It is possible to reconstruct the exact state of the cache of a particular node
- Total storage requirement: ***only ~ 1 gigabyte per day!!!***

What the daily output files look like

```
$ find ./2026/03/08/ ./2026/03/09/ ./2026/03/10/ -type f | xargs ls -lah

-rw-r--r-- 1 nobody nogroup 496M Mar  8 00:00 2026/03/08/20260308-initstate.tar.zst
-rw-r--r-- 1 nobody nogroup 552M Mar  8 23:58 2026/03/08/20260308-rpkispool.tar.zst

-rw-r--r-- 1 nobody nogroup 495M Mar  9 00:00 2026/03/09/20260309-initstate.tar.zst
-rw-r--r-- 1 nobody nogroup 602M Mar  9 23:58 2026/03/09/20260309-rpkispool.tar.zst

-rw-r--r-- 1 nobody nogroup 495M Mar 10 00:00 2026/03/10/20260310-initstate.tar.zst
-rw-r--r-- 1 nobody nogroup 632M Mar 10 23:59 2026/03/10/20260310-rpkispool.tar.zst
```

To see the directory hierarchy structure yourself:

```
$ rsync -rt rsync://josephine.sobornost.net/rpki/rpkispools/
```

Inside the daily checkpoint (the “initstate”)

```
# tar vtf 20260310-initstate.tar.zst | head -n 20

-r--r--r-- 0/0     1914 2025-11-16 07:49
ams1/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
ams3/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
blr1/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
blr2/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
dus1/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
miso/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
nyc1/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
sng1/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
syd1/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
syd2/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
yyz1/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
zur1/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0     1914 2025-11-16 07:49
zur2/akane.maru.co.jp/repo/1073c6/1/3134302e3233352e3139392e302f32342d3234203d3e20323134363735.roa
-r--r--r-- 0/0      524 2026-03-09 22:37  ams1/akane.maru.co.jp/repo/1073c6/1/F03205B3993400CC3FC657EFE68E0696DA332458.crl
-r--r--r-- 0/0      524 2026-03-09 22:37  ams3/akane.maru.co.jp/repo/1073c6/1/F03205B3993400CC3FC657EFE68E0696DA332458.crl
-r--r--r-- 0/0      524 2026-03-09 22:37  blr1/akane.maru.co.jp/repo/1073c6/1/F03205B3993400CC3FC657EFE68E0696DA332458.crl
-r--r--r-- 0/0      524 2026-03-09 22:37  blr2/akane.maru.co.jp/repo/1073c6/1/F03205B3993400CC3FC657EFE68E0696DA332458.crl
-r--r--r-- 0/0      524 2026-03-09 22:37  dus1/akane.maru.co.jp/repo/1073c6/1/F03205B3993400CC3FC657EFE68E0696DA332458.crl
```

Node Identifier (4 letters)

FQDN + full Signed Object path (SIA)

Inside the log (rpkispool)

```
# tar vtf 20260301-rpkispool.tar.zst | fgrep -m1 -A5 -B5 .ccr

-r--r--r-- 0/0         548 2026-03-01 00:00 static/0K/LU/HvnvCbGHIE42NpG6Fpx8NK_94LLNHHbZsuh1Q7l0KLU
-r--r--r-- 0/0         548 2026-03-01 00:00 static/0s/8Y/eQmMs0N8T2FObu-k7HorubQqUrVQd3lkM7Mm_kZ0s8Y
-r--r--r-- 0/0         434 2026-03-01 00:00 static/1A/HE/QWFu2t5XTsuMGhaVowKVMrKyRmLlHJlmqL7uf0M1AHE
-r--r--r-- 0/0         510 2026-03-01 00:00 static/1D/XQ/sKeIyf9Fj71hW0R0SPcFViBchkZvcNhi6VL45V-1DXQ
-r--r--r-- 0/0         510 2026-03-01 00:00 static/1Z/jU/2SMhqOUgK5NYtq3L06ZqFHvZdKbhDi9HYAbCPlc1ZjU
-r--r--r-- 0/0    21295191 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.ccr
-r--r--r-- 0/0      125194 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.log
-r--r--r-- 0/0      710263 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.metrics
-r--r--r-- 0/0         548 2026-03-01 00:00 static/2C/M8/gRAUD18G1a4Sr5A0jrH7kTpX4Yj1Zfrywjt5yeD2CM8
-r--r--r-- 0/0        1997 2026-03-01 00:00 static/2N/j4/TVkptGmN3prJANdeWxQVS1Bt-UpnjtDX6RzIG-B2Nj4
-r--r--r-- 0/0         660 2026-03-01 00:00 static/2k/7U/vKNQKfG--QClvWrie6lq_LZSPiXhkcxHK_Thfkf2k7U
```

Inside the *log* (rpkispool)

```
# tar vtf 20260301-rpkispool.tar.zst | fgrep -m1 -A5 -B5 .ccr

-r--r--r-- 0/0          548 2026-03-01 00:00 static/0K/LU/HvnvCbGHIE42NpG6Fpx8NK_94LLNHHbZsuh1Q7l0KLU
-r--r--r-- 0/0          548 2026-03-01 00:00 static/0s/8Y/eQmMs0N8T2FObu-k7HorubQqUrVQd3lkM7Mm_kZ0s8Y
-r--r--r-- 0/0          434 2026-03-01 00:00 static/1A/HE/QWFu2t5XTsuMGhaVowKVMrKyRmLlHJlmqL7uf0M1AHE
-r--r--r-- 0/0          510 2026-03-01 00:00 static/1D/XQ/sKeIyf9Fj71hW0R0SPcFViBchkZvcNhi6VL45V-1DXQ
-r--r--r-- 0/0          510 2026-03-01 00:00 static/1Z/jU/2SMhqOUgK5NYtq3L06ZqFHvZdKbhDi9HYAbCPlc1ZjU
-r--r--r-- 0/0     21295191 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.ccr
-r--r--r-- 0/0       125194 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.log
-r--r--r-- 0/0       710263 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.metrics
-r--r--r-- 0/0          548 2026-03-01 00:00 static/2C/M8/gRAUD18G1a4Sr5A0jrH7kTpX4Yj1Zfrywjt5yeD2CM8
-r--r--r-- 0/0         1997 2026-03-01 00:00 static/2N/j4/TVkptGmN3prJANdeWxQVS1Bt-UpnjtDX6RzIG-B2Nj4
-r--r--r-- 0/0          660 2026-03-01 00:00 static/2k/7U/vKNQKfG--QClvWrie6lq_LZSPiXhkcxHK_Thfkf2k7U
```

RFC 9589-style timestamps
(CMS signing-time, notBefore, thisUpdate, producedAt)

Inside the log (rpkispool)

```
# tar vtf 20260301-rpkispool.tar.zst | fgrep -m1 -A5 -B5 .ccr

-r--r--r-- 0/0         548 2026-03-01 00:00  static/0K/LU/HvnvCbGHIE42NpG6Fpx8NK_94LLNHHbZsuh1Q7l0KLU
-r--r--r-- 0/0         548 2026-03-01 00:00 static/0s/8Y/eQmMs0N8T2FObu-k7HorubQqUrVQd3lkM7Mm_kZ0s8Y
-r--r--r-- 0/0         434 2026-03-01 00:00 static/1A/HE/QWFu2t5XTsuMGhaVowKVMrKyRmLlHJlmqL7uf0M1AHE
-r--r--r-- 0/0         510 2026-03-01 00:00 static/1D/XQ/sKeIyf9Fj71hW0R0SPcFViBchkZvcNhi6VL45V-1DXQ
-r--r--r-- 0/0         510 2026-03-01 00:00 static/1Z/jU/2SMhqOUgK5NYtq3L06ZqFHvZdKbhDi9HYAbCPlc1ZjU
-r--r--r-- 0/0    21295191 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.ccr
-r--r--r-- 0/0      125194 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.log
-r--r--r-- 0/0      710263 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.metrics
-r--r--r-- 0/0         548 2026-03-01 00:00 static/2C/M8/gRAUD18G1a4Sr5A0jrH7kTpX4Yj1Zfrywjt5yeD2CM8
-r--r--r-- 0/0        1997 2026-03-01 00:00 static/2N/j4/TVkptGmN3prJANdeWxQVS1Bt-UpnjtDX6RzIG-B2Nj4
-r--r--r-- 0/0         660 2026-03-01 00:00 static/2k/7U/vKNQKfG--QClvWrie6lq_LZSPiXhkcxHK_Thfkf2k7U
```

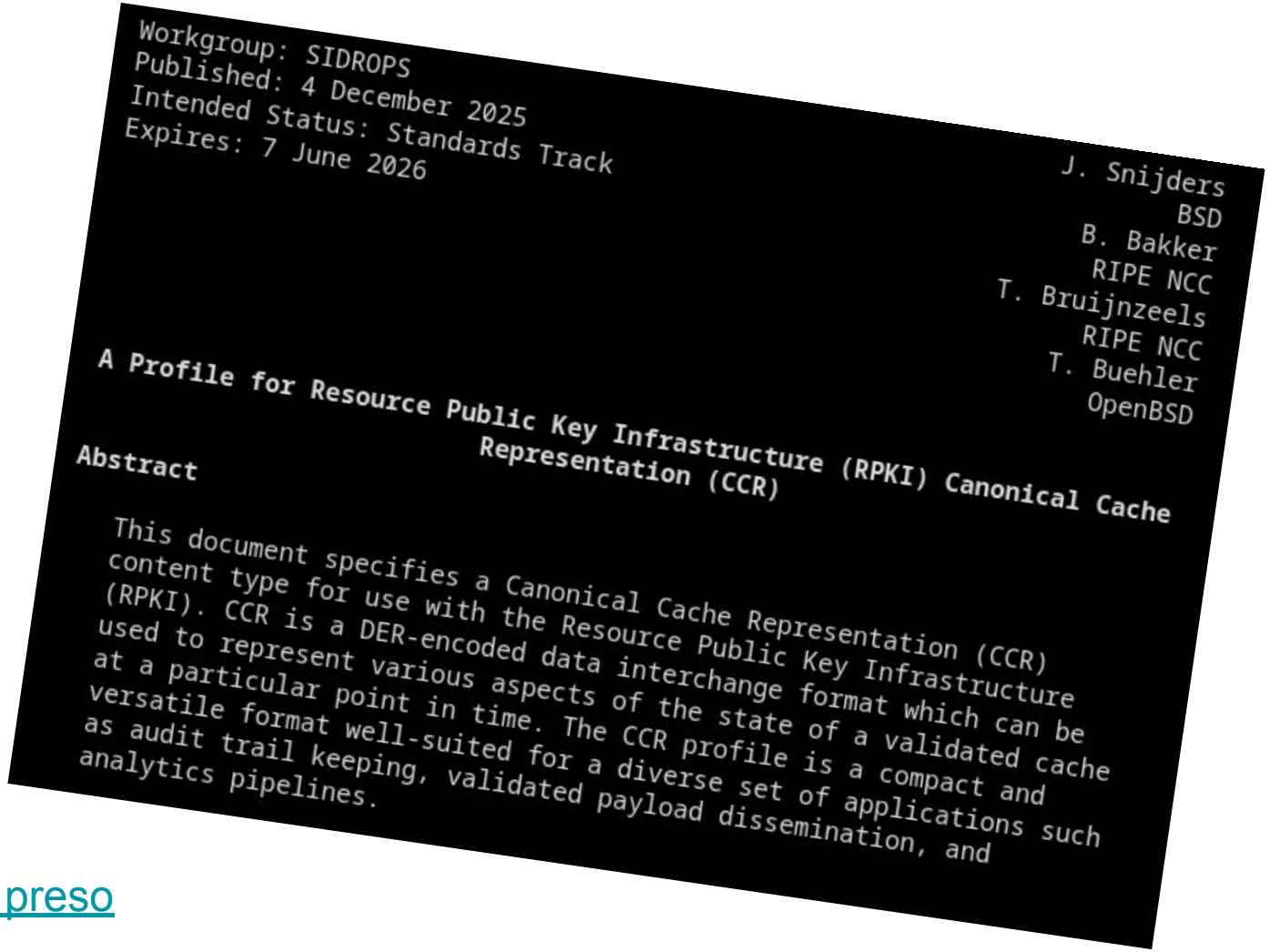
Inside the log (rpkispool)

```
# tar vtf 20260301-rpkispool.tar.zst | fgrep -m1 -A5 -B5 .ccr

-r--r--r-- 0/0         548 2026-03-01 00:00 static/0K/LU/HvnvCbGHIE42NpG6Fpx8NK_94LLNHHbZsuh1Q7l0KLU
-r--r--r-- 0/0         548 2026-03-01 00:00 static/0s/8Y/eQmMs0N8T2FObu-k7HorubQqUrVQd3lkM7Mm_kZ0s8Y
-r--r--r-- 0/0         434 2026-03-01 00:00 static/1A/HE/QWFu2t5XTsuMGhaVowKVMrKyRmLlHJlmqL7uf0M1AHE
-r--r--r-- 0/0         510 2026-03-01 00:00 static/1D/XQ/sKeIyf9Fj71hW0R0SPcFViBchkZvcNhi6VL45V-1DXQ
-r--r--r-- 0/0         510 2026-03-01 00:00 static/1Z/jU/2SMhqOUgK5NYtq3L06ZqFHvZdKbhDi9HYAbCPlc1ZjU
-r--r--r-- 0/0    21295191 2026-03-01 00:00  2026/03/01/20260301T000035Z-miso.ccr
-r--r--r-- 0/0      125194 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.log
-r--r--r-- 0/0      710263 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.metrics
-r--r--r-- 0/0         548 2026-03-01 00:00 static/2C/M8/gRAUD18G1a4Sr5A0jrH7kTpX4Yj1Zfrywjt5yeD2CM8
-r--r--r-- 0/0        1997 2026-03-01 00:00 static/2N/j4/TVkptGmN3prJANdeWxQVS1Bt-UpnjtDX6RzIG-B2Nj4
-r--r--r-- 0/0         660 2026-03-01 00:00 static/2k/7U/vKNQKfG--QClvWrie6lq_LZSPiXhkcxHK_Thfkf2k7U
```

CCR???!

(think MRT!)



Workgroup: SIDROPS
Published: 4 December 2025
Intended Status: Standards Track
Expires: 7 June 2026

J. Snijders
BSD
B. Bakker
RIPE NCC
T. Bruijnzeels
RIPE NCC
T. Buehler
OpenBSD

A Profile for Resource Public Key Infrastructure (RPKI) Canonical Cache Representation (CCR)

Abstract

This document specifies a Canonical Cache Representation (CCR) content type for use with the Resource Public Key Infrastructure (RPKI). CCR is a DER-encoded data interchange format which can be used to represent various aspects of the state of a validated cache at a particular point in time. The CCR profile is a compact and versatile format well-suited for a diverse set of applications such as audit trail keeping, validated payload dissemination, and analytics pipelines.

[IETF 124 CCR preso]

Inside the log (rpkispool)

```
# tar vtf 20260301-rpkispool.tar.zst | fgrep -m1 -A5 -B5 .ccr

-r--r--r-- 0/0         548 2026-03-01 00:00 static/0K/LU/HvnvCbGHIE42NpG6Fpx8NK_94LLNHHbZsuh1Q7l0KLU
-r--r--r-- 0/0         548 2026-03-01 00:00 static/0s/8Y/eQmMs0N8T2FObu-k7HorubQqUrVQd3lkM7Mm_kZ0s8Y
-r--r--r-- 0/0         434 2026-03-01 00:00 static/1A/HE/QWFu2t5XTsuMGhaVowKVMrKyRmLlHJlmqL7uf0M1AHE
-r--r--r-- 0/0         510 2026-03-01 00:00 static/1D/XQ/sKeIyf9Fj71hW0R0SPcFViBchkZvcNhi6VL45V-1DXQ
-r--r--r-- 0/0         510 2026-03-01 00:00 static/1Z/jU/2SMhqOUgK5NYtq3L06ZqFHvZdKbhDi9HYAbCPlc1ZjU
-r--r--r-- 0/0    21295191 2026-03-01 00:00 2026/03/01/20260301T000035Z-miso.ccr
-r--r--r-- 0/0      125194 2026-03-01 00:00  2026/03/01/20260301T000035Z-miso.log
-r--r--r-- 0/0      710263 2026-03-01 00:00  2026/03/01/20260301T000035Z-miso.metrics
-r--r--r-- 0/0         548 2026-03-01 00:00 static/2C/M8/gRAUD18G1a4Sr5A0jrH7kTpX4Yj1Zfrywjt5yeD2CM8
-r--r--r-- 0/0        1997 2026-03-01 00:00 static/2N/j4/TVkptGmN3prJANdeWxQVS1Bt-UpnjtDX6RzIG-B2Nj4
-r--r--r-- 0/0         660 2026-03-01 00:00 static/2k/7U/vKNQKfG--QClvWrie6lq_LZSPiXhkcxHK_Thfkf2k7U
```

How to interact with the data in the .tar.zst

```
# rpki-client -j -f static/0K/LU/HvnvCbGHIE42NpG6Fpx8NK_94LLNHHbZsuh1Q7l0KLU … … … …
{
        "file": "static/0K/LU/HvnvCbGHIE42NpG6Fpx8NK_94LLNHHbZsuh1Q7l0KLU",
        "hash_id": "HvnvCbGHIE42NpG6Fpx8NK/94LLNHHbZsuh1Q7l0KLU=",
        "type": "crl",
        "aki": "C1EEE2CAB86B5EE40FC2770CA8CB5C50C62B755B",
        "crl_issuer": "/CN=c1eee2cab86b5ee40fc2770ca8cb5c50c62b755b",
        "crl_serial": "14BC",
        "valid_since": 1772323235,
        "valid_until": 1772409635,
        "revoked_certs": [
            { "serial": "0194258FA7B79703FD990F039DC8EA9A55BD", "date": "Fri 02 Jan 2026 06:18:33 +0000" },
            { "serial": "019CA2D5F6B2361B39BE0622373D8125F02D", "date": "Sat 28 Feb 2026 15:01:05 +0000" },
            { "serial": "019CA4C479C2050A6E6452B318D6A5E8A7C9", "date": "Sun 01 Mar 2026 00:00:35 +0000" }
        ],
        "validation": "N/A"
}
```

RPKISPOOL archive contents in summary:

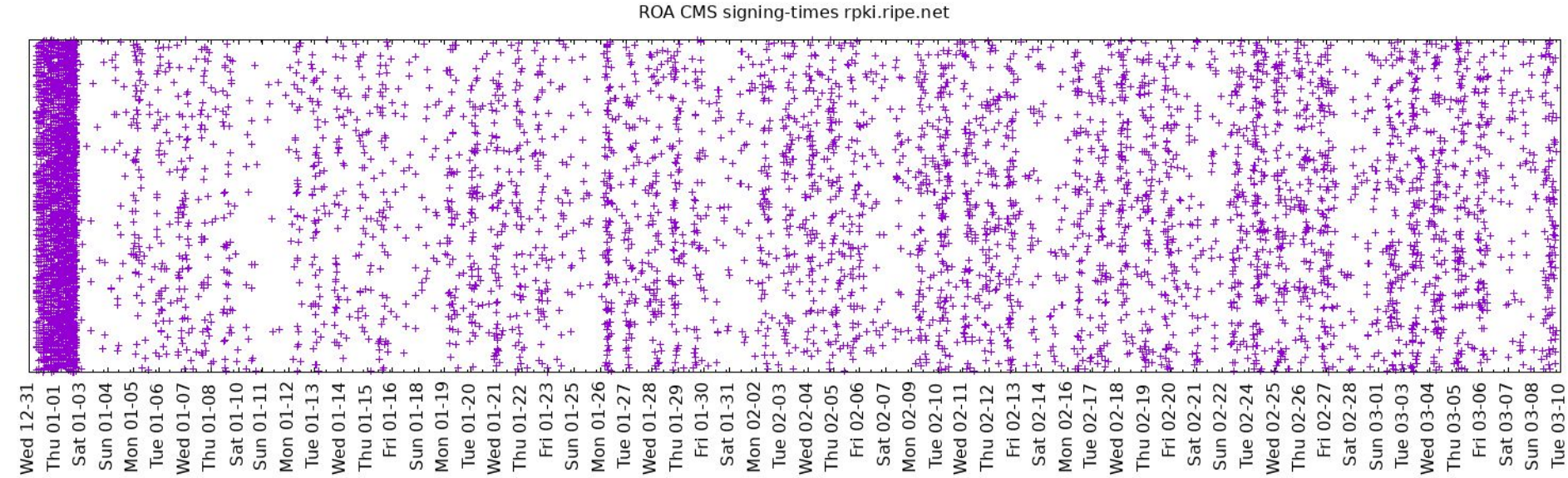
- All “yesterday’s” newly discovered raw data
- Each RPKI object appears only once: the data is deduplicated
- Tar archive member timestamps are “RFC 9589” derived timestamps
- Tar archive member filenames are the SHA-256 of the object content
- Tar members are sorted by their internal timestamp (to optimize for Zstd)
- All objects can be inspected with either ‘`rpki-client -f`’ or ‘`cat`’
- Each CCR represents what *that* RP saw at *that* moment in time
- From a given CCR the full validated cache can be reconstructed
- RPKISPOOL also contains OpenMetrics data and auxiliary files
- The file naming scheme was designed so that many rpkispool archives can be extracted within the same working directory: no filename clashes

Cool things to do with RPKISPOOL archives

- Discover backdating CAs!
  - Not-yet-resolved bug was found in ARIN's Hosted CA implementation where the system was re-using timestamps from previous ROA issuances!?!?!
- Discover mis-issuances!
  - Not-yet-resolved bug was found in APNIC's Hosted CA system where super old objects re-appear!??!?!
- Discover lack of HTTP compression!
  - Analysing the rpki-client logs in the rpkispools, e.g. it was found that TWNIC didn't have support for HTTP compression enabled in their RRDP nginx. (now solved)
- Discover RRDP consistency issues!
  - Analysing the rpki-client logs in the rpkispools, a race condition was discovered in TWNIC's RRDP snapshot retention approach (now solved)
- …???

Other cool things to do with RPKISPOOL archives

- Analyse discrepancies between Rsync & RRDP?
- Search for “two-faced CA” attacks?
- Detect patterns: it seems people avoid doing ROA work on the weekend?



FAQ & Questions?

Q: is all this data available for free?
A: yes! See the mirrors list on https://www.rpkiviews.org/#mirrors

Q: tooling for RPKISPOOL?
A: yes! work in progress: https://github.com/fvompe/rpkispool-cli

Q: did you also make RPKI data for the years 2020-2025 accessible?
A: yes! http://rpkiviews.org/#amalgamations