RIPE NCC
RIPE NETWORK COORDINATION CENTRE

RIS and RIPEstat

Analysing Operators' Routing with RIPEstat and RIS

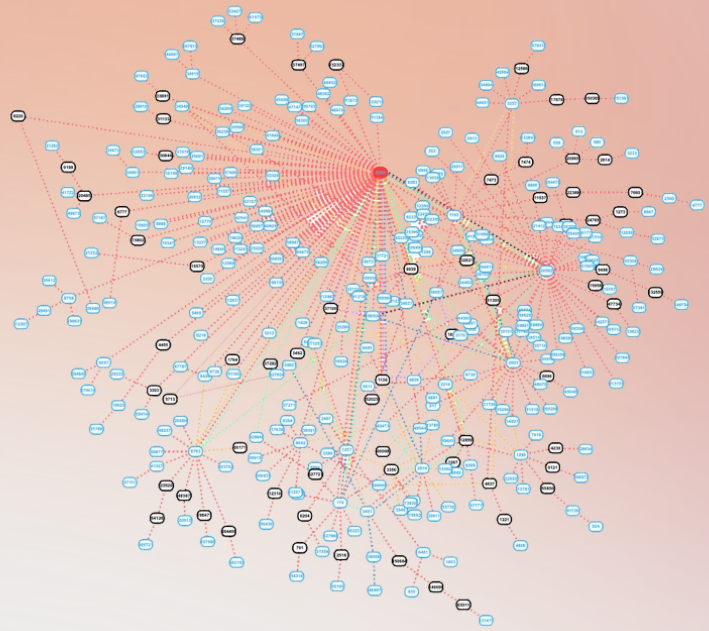
Agustín Formoso | RIPE NCC | 30 September 2026

What is RIPEstat?

Information service for Internet-related data and analytics

- **RIPEstat offers:**
  - Information on IP address space and Autonomous System Numbers (ASNs)
  - Statistics on specific hostnames and countries
  - Visualisations of Internet routing

- **All in a centralised place**



RIPEstat

How RIPEstat Works

RIPEstat pulls data from a variety of sources, including:

- The RIPE Database (for ownership and contact information)
- Routing registries (e.g., RIPE IRR)
- DNS and reverse DNS systems
- Live and historical BGP data (RIPE RIS)
- Regional Internet Registries and IANA

More details at:

stat.ripe.net/data-sources

What Can RIPEstat Do For You?



Analyse routing
and network health



Check routing security
and RPKI validation



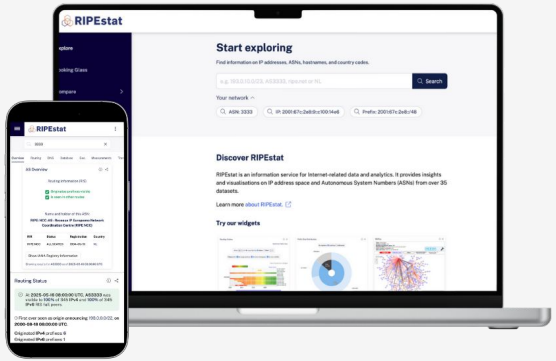
Analyse network outages
and anomalies



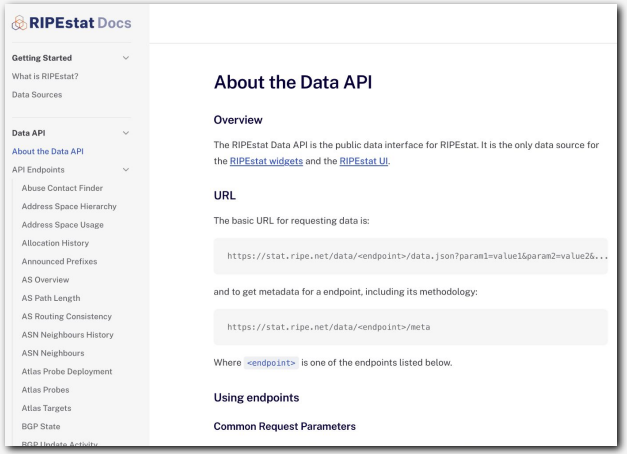
Have a real-time
Looking Glass

How to Access RIPEstat

User Interface (UI)



API ⚙️



More information at:

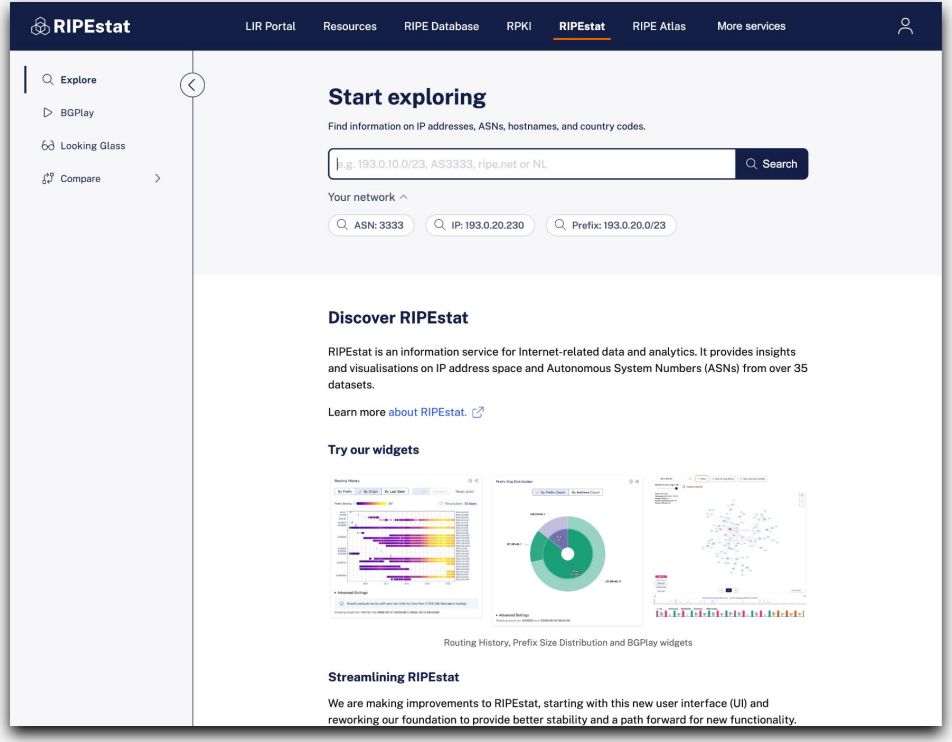
stat.ripe.net/docs

Get started with RIPEstat

Quick Intro

- Search by IP address/prefix (IPv4, IPv6), ASN, country code or fully qualified domain name (FQDN)

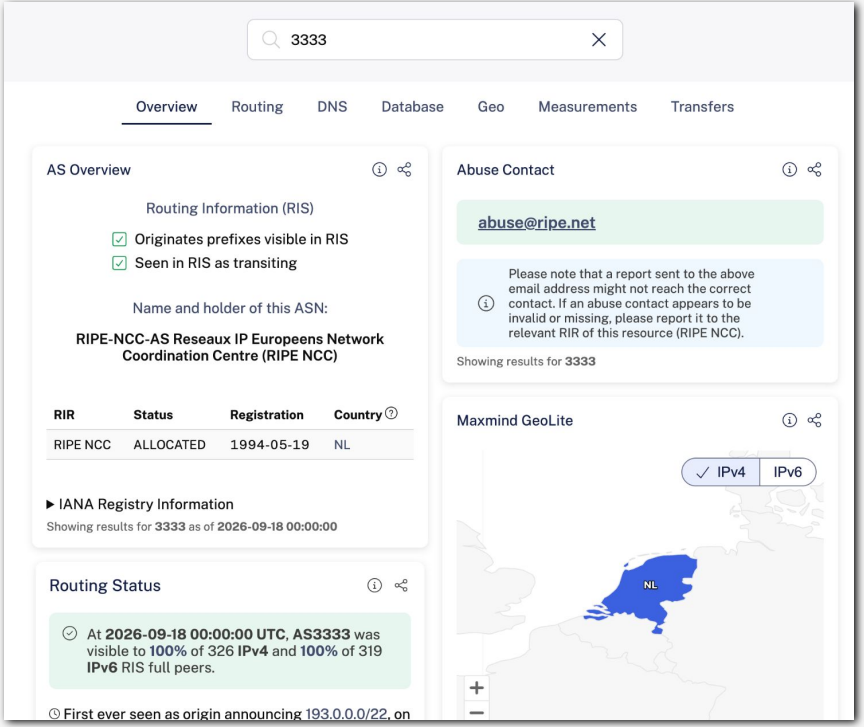
- Suggestions are pulled from your network or randomly



Quick Intro

Information grouped on the top through tabs

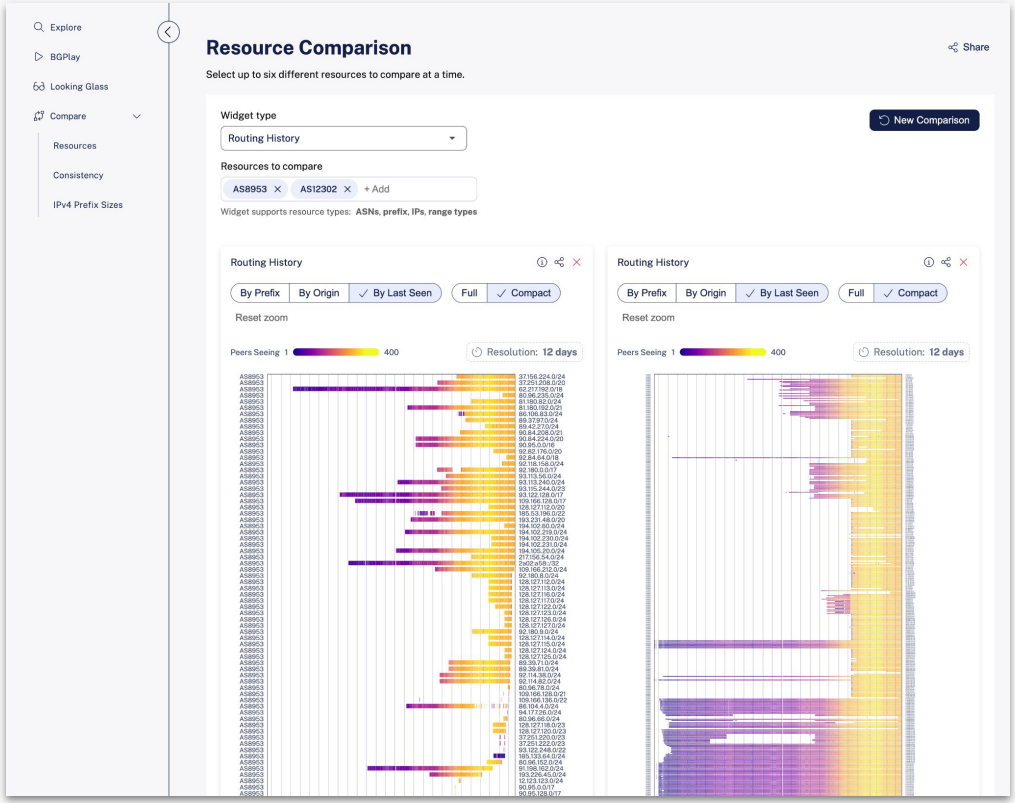
- Routing
- DNS
- Database
- Geolocation
- Measurements
- Transfers



Use Cases: Resource Comparison

Analyse and compare statistics

View the same dataset for multiple resources at the same time

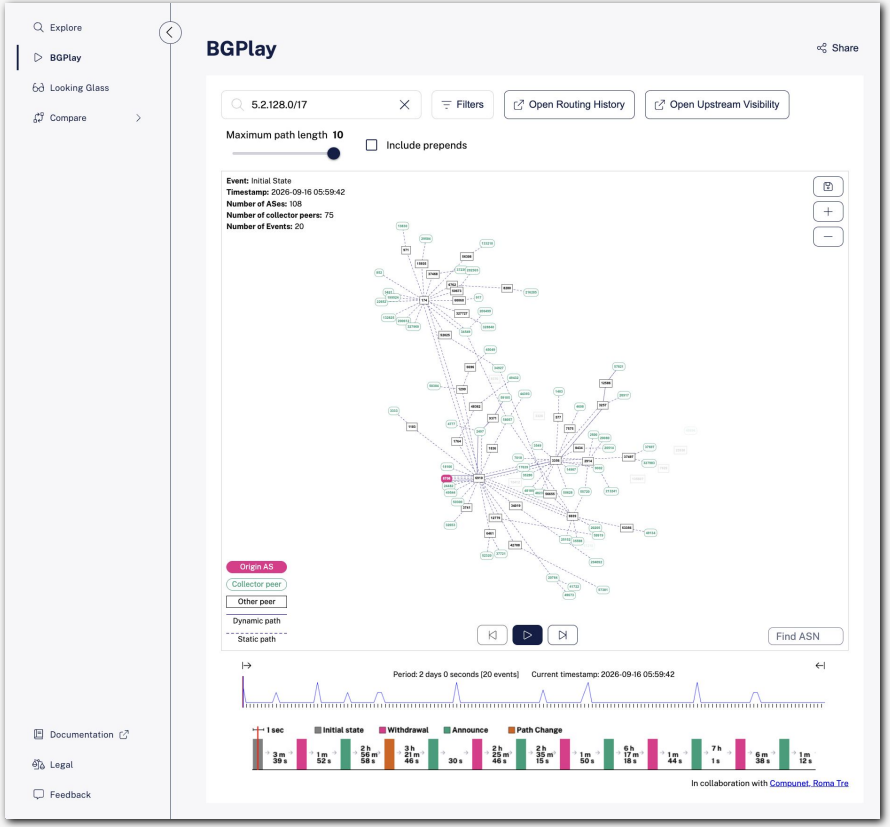


link

Use Cases: BGPlay

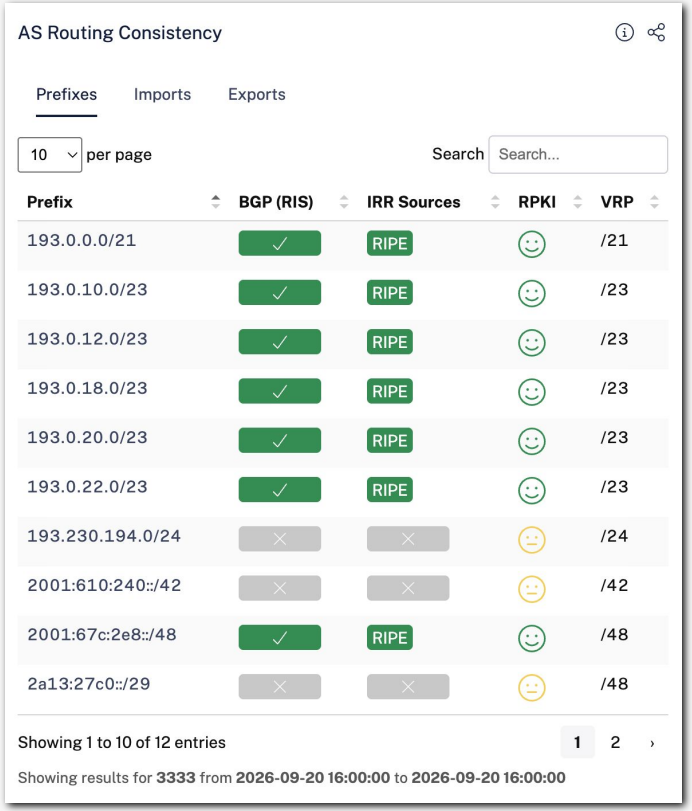
- BGPlay shows the routing history related to a specific set of resources (prefixes, Autonomous Systems, IPs)
- BGPlay visualises changes in BGP announcements for a specific destination over time
- As seen by RIS

link



Use Cases: Routing Consistency

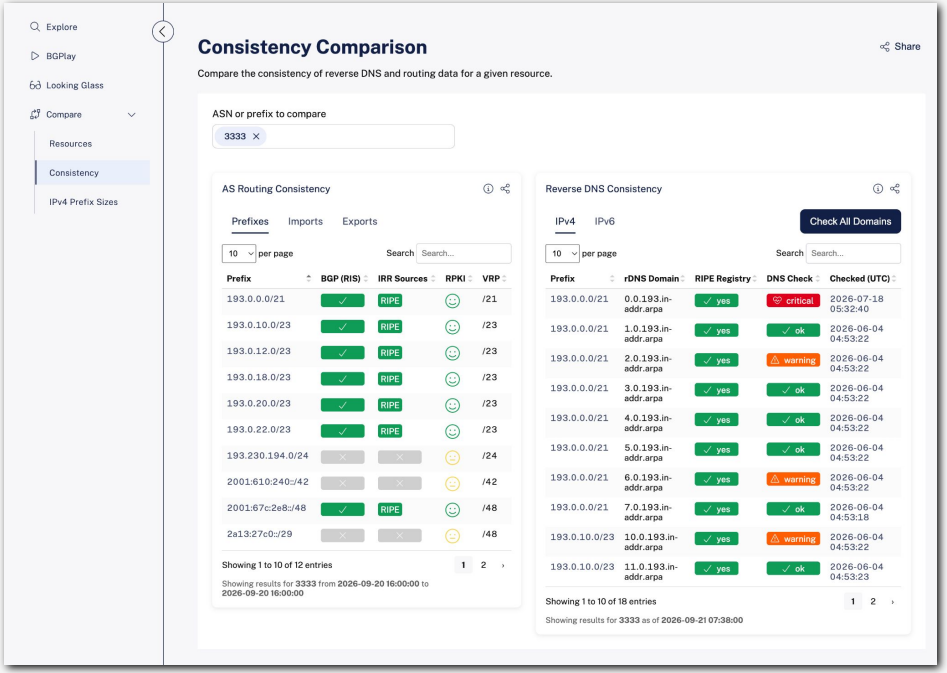
- Routing Consistency compares:
  - Objects in Routing Registries with observed real-world routing (seen by RIS)
  - RPKI validation status where applicable



link

Use Cases: Consistency Comparison

Compare the consistency of reverse DNS and routing data for a given resource

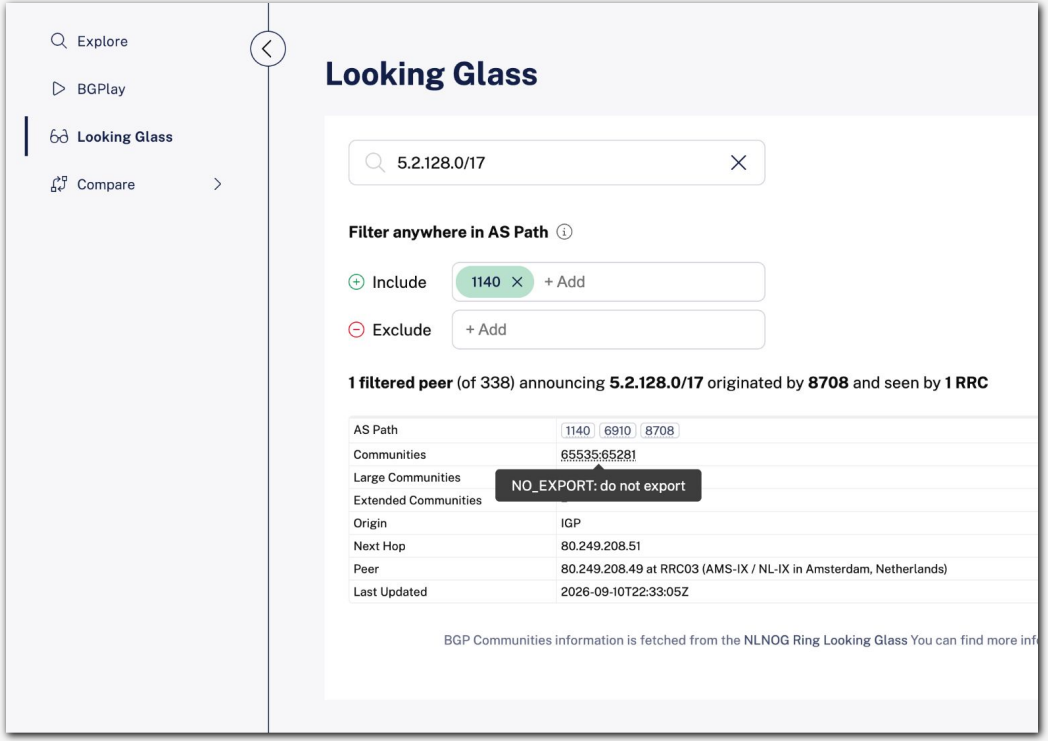


link

Looking Glass

- Real-time data as seen by RIS
- Filter anywhere in AS Path
- Include/exclude filters
- See BGP communities

stat.ripe.net/lg/5.2.128.0/17



How RIPEstat Works

RIPEstat pulls data from a variety of sources, including:

- The RIPE Database (for ownership and contact information)
- Routing registries (e.g., RIPE IRR)
- DNS and reverse DNS systems
- Live and historical BGP data (RIPE RIS)
- Regional Internet Registries and IANA

More details at:

stat.ripe.net/data-sources

How RIPEstat Works

RIPEstat pulls data from a variety of sources, including:

- The RIPE Database (for ownership and contact information)
- Routing registries (e.g., RIPE IRR)
- DNS and reverse DNS systems
- **Live and historical BGP data (RIPE RIS)**
- Regional Internet Registries and IANA

More details at:

stat.ripe.net/data-sources

RIPE RIS

The RIPE Routing Information Service

What is RIS?

- RIS is a route data collection platform
- Collaborative with community
- Collecting BGP data since 1999



The RIPE Routing Information Service

What is RIS?

- RIS is a route data collection platform
- Collaborative with community
- Collecting BGP data since 1999
  - We *permanently* store the data (BGP tables + BGP updates)
  - Long-term archive is *publicly and freely available*
- Used by many tools and researchers
  - RIPE NCC: RIPEstat, riswhois (IP to origin AS + prefix), ris-live, …
  - Others: Academic research, open source tools, commercial tools, …

What is RIPE RIS?

System overview

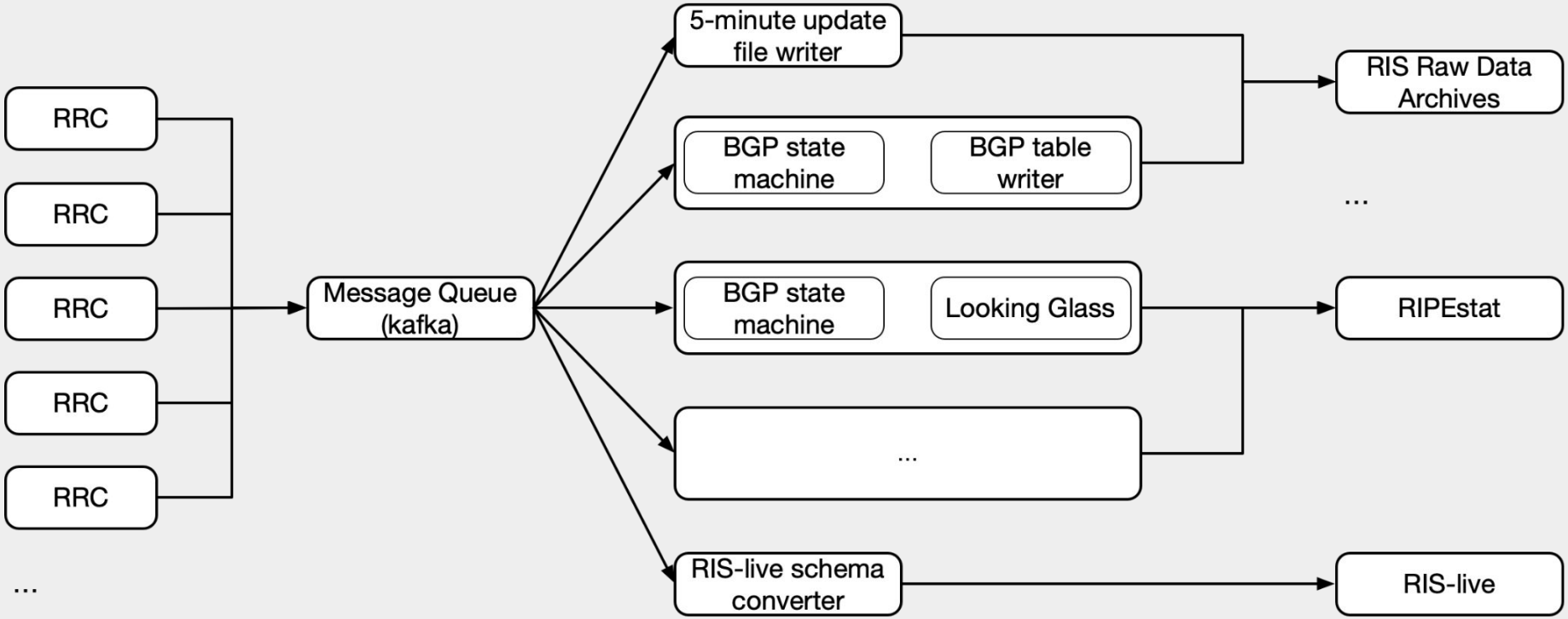
- A world-wide network of route collectors
  - RRC22 @ InterLAN
- *Many* BGP peers that have BGP sessions with those collectors
- Standard components
  - Linux servers, ExaBGP, Kafka message broker
- + Custom data processing and plumbing

What is RIPE RIS?

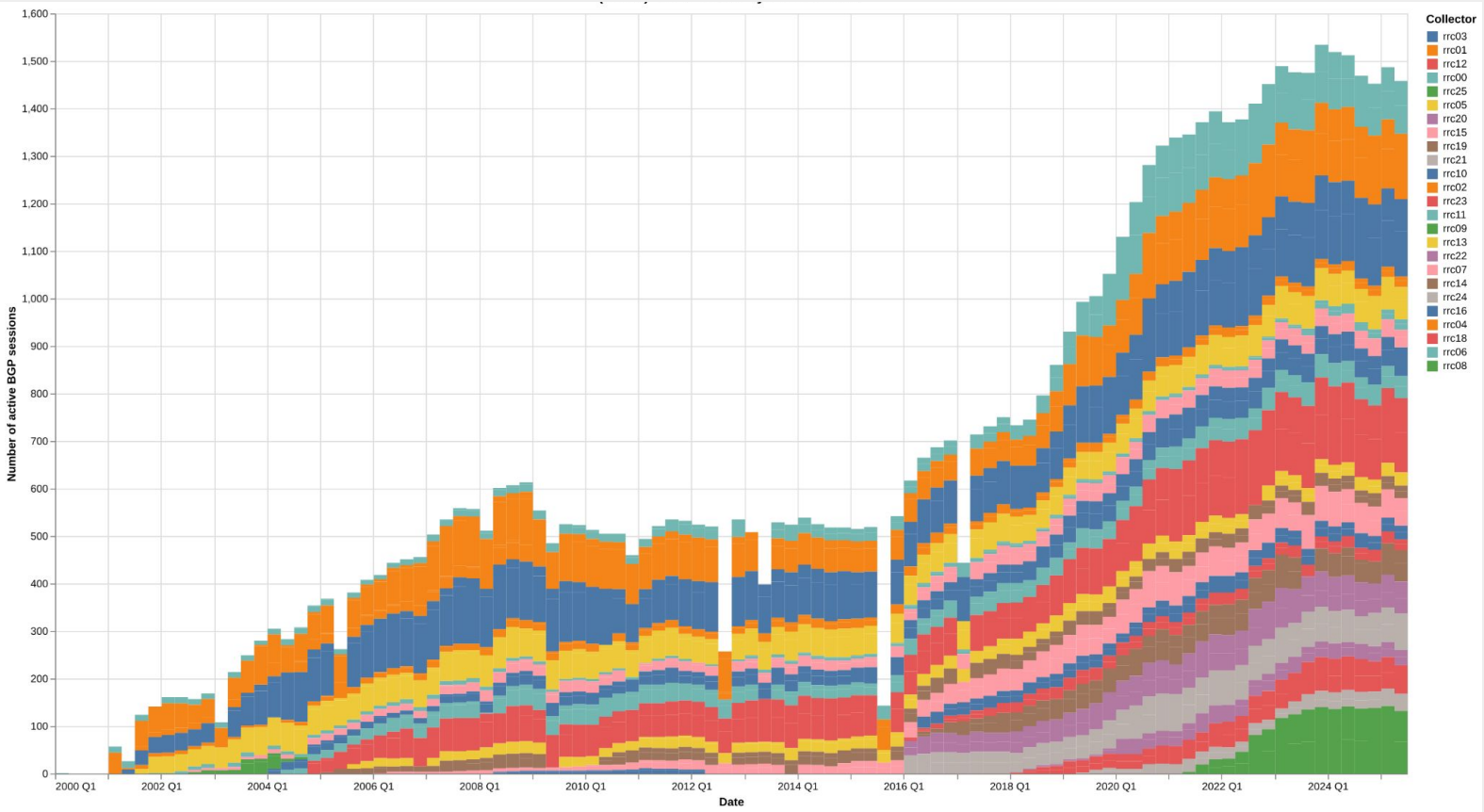
System overview

- A world-wide network of route collectors
  - RRC22 @ InterLAN
- *Many* BGP peers that have BGP sessions with those collectors
- Standard components
  - Linux servers, ExaBGP, Kafka message broker
- + Custom data processing and plumbing
- *Many* BGP peers: A lot of routes and updates.
- 21 May 2026:
  - ~366M IPv4 routes, ~84M IPv6 routes
  - 17.5K bgp announcements/withdraws per second on average,
  - 46k A/W per second peak one-minute average (across five days)

RIS Architecture



RIS Over Time: Number of Active BGP Sessions



RIS Data Over Time

Remember: We store all data permanently

- Data size continuously increases
- Selecting relevant data is hard
  - For researchers: data volume is too big
  - For us as well: peering is selecting data for long-term observation

RIS Data Over Time

Remember: We store all data permanently

- Data size continuously increases
- Selecting relevant data is hard
  - For researchers: data volume is too big
  - For us as well: peering is selecting data for long-term observation
- We have a selective peering policy
  - Under change
  - Important part: validation
- We *really* want to learn about the most relevant interconnection in a region. Please talk to us!

How is RIS Data Used?

By us:

- In our tools
- In our internet analysis

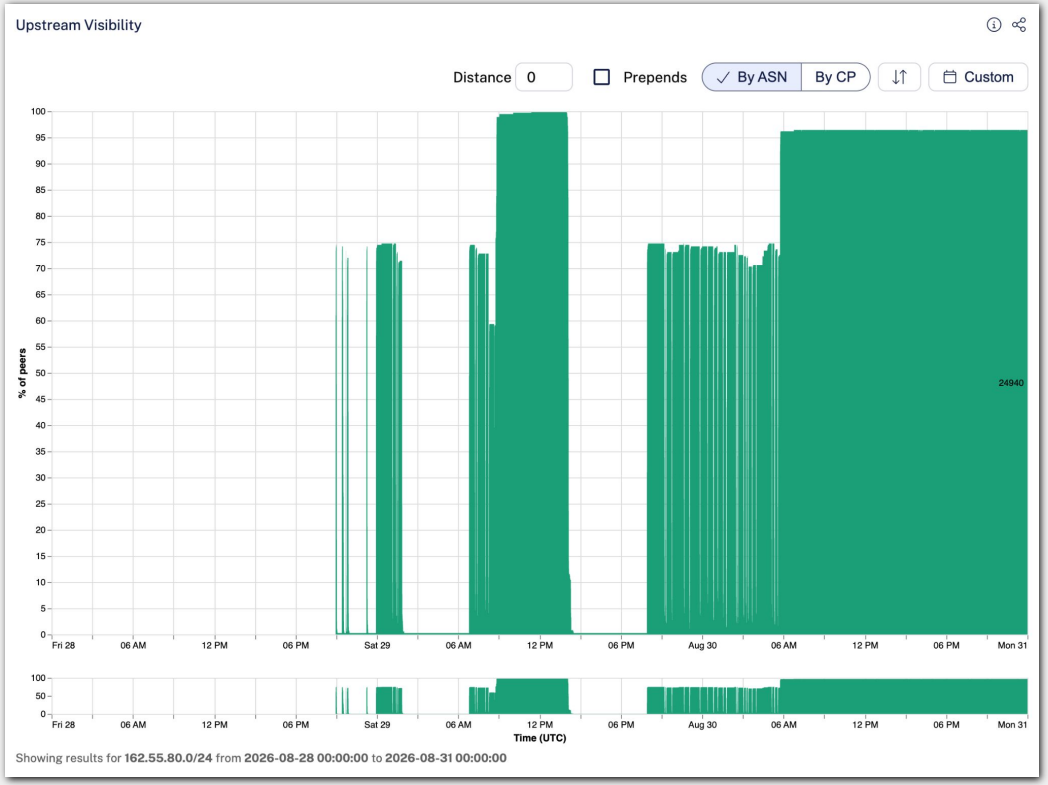


Web Application Example: 162.55.80.0/24

Upstream Visibility

- 28-31 August 2026
- 162.55.80.0/24
- Distance from origin: 0
  - Origin itself
  - Same over time

link

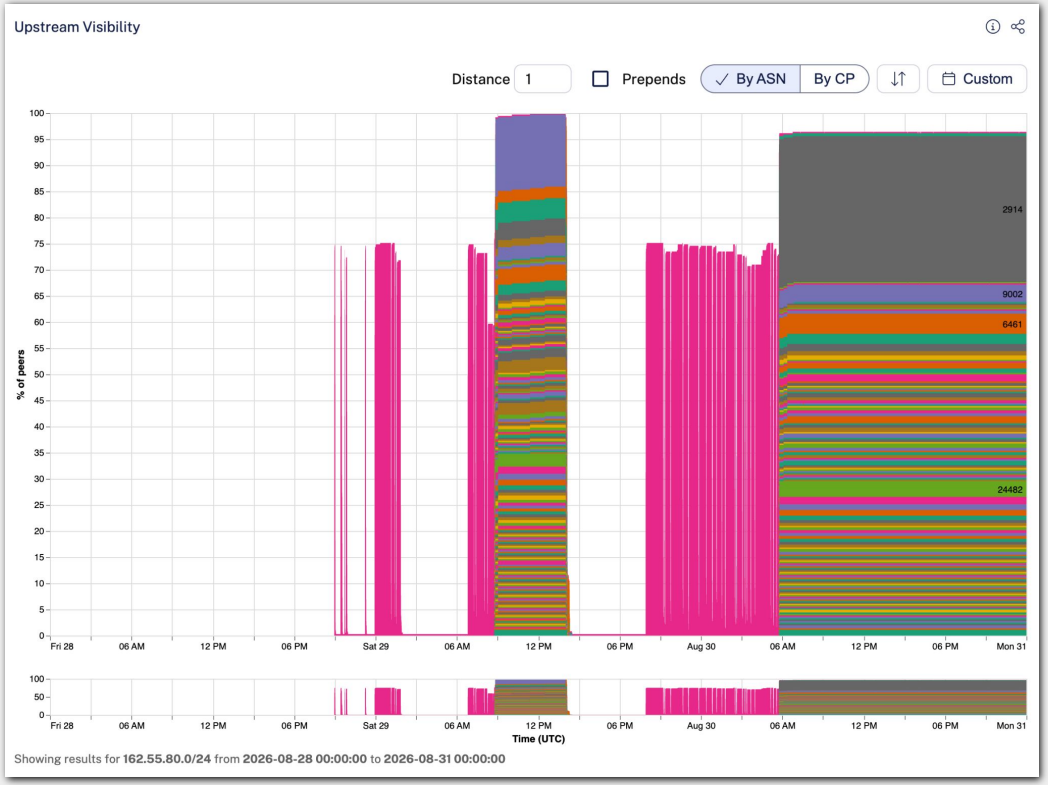


RIPEstat's Upstream Visibility showing the visibility of 162.55.80.0/24 as seen by RIPE RIS peers.

Web Application Example: 162.55.80.0/24

Upstream Visibility

- 28-31 August 2026
- 162.55.80.0/24
- Distance from origin: 0
  - Origin itself
  - Same over time
- Distance from origin: 1
  - *Upstreams*
  - Significant change in upstreams



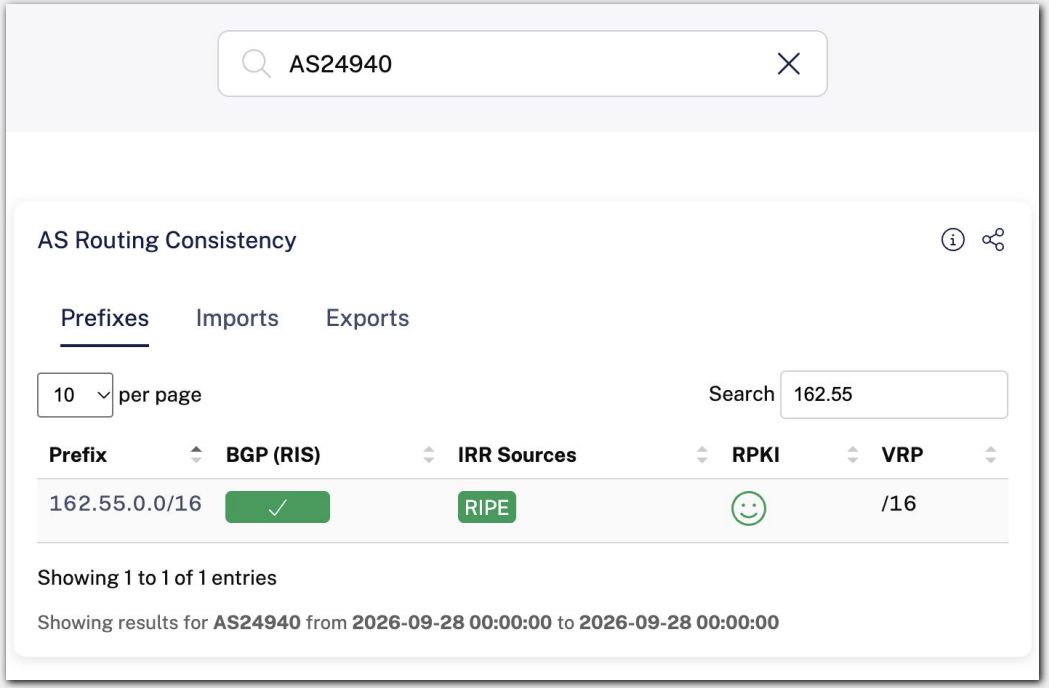
RIPEstat's Upstream Visibility showing the visibility of 162.55.80.0/24 as seen by RIPE RIS peers.

Incident covered by BGPKIT, Kentik, and others

Web Application Example: 162.55.80.0/24

AS Routing Consistency

- 28-31 August 2026
- 162.55.80.0/24
- ROA was tightened



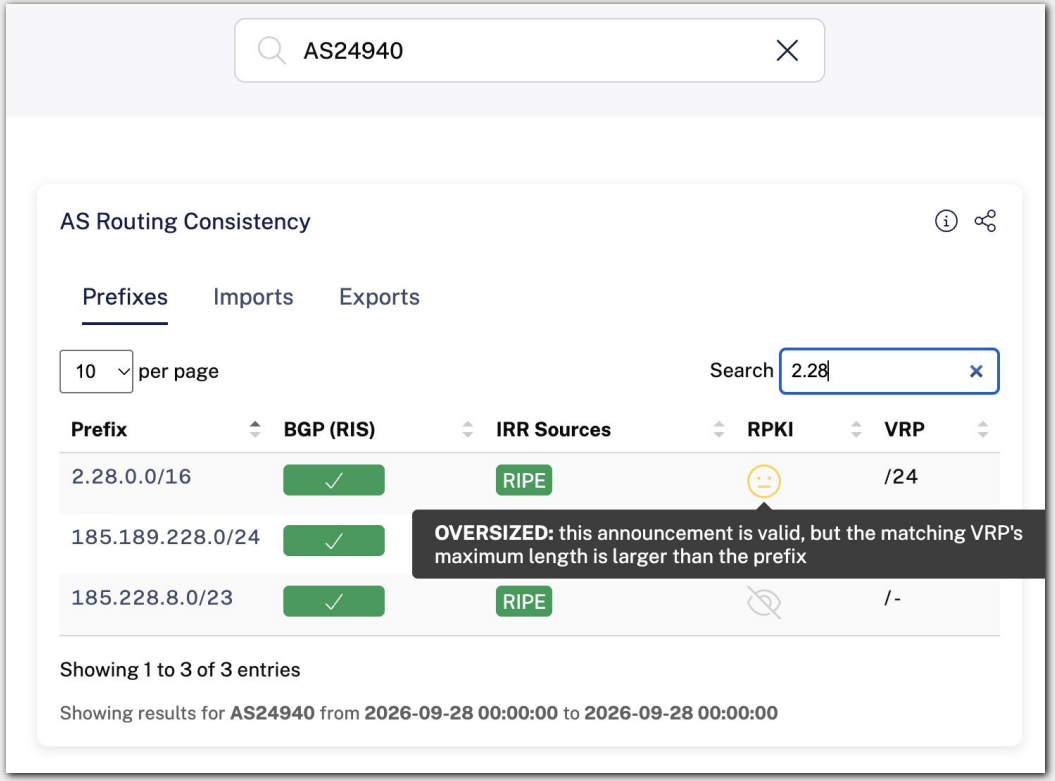
RIPEstat's *AS Routing Consistency* showing information about prefixes of AS24940.
This view is filtering for rows matching *"162.55"*

Incident covered by BGPKIT, Kentik, and others

Web Application Example: 162.55.80.0/24

AS Routing Consistency

- 28-31 August 2026
- 162.55.80.0/24
- ROA was tightened
- Other ROAs still oversized



RIPEstat's *AS Routing Consistency* showing information about prefixes of AS24940. This view is filtering for rows matching *"2.28" and showing an example of an oversized ROA for 2.28/16 => AS24940.*

Incident covered by BGPKIT, Kentik, and others

How is RIS Data Used?

By us:

- In our tools
- In our internet analysis

```
$ whois -h riswhois.ripe.net 162.55.80.0/16
% This is RIPE NCC's Routing Information Service
% whois gateway to collected BGP Routing Tables, version2.0
% IPv4 or IPv6 address to origin prefix match
%
% For more information visit http://www.ripe.net/ris/riswhois.html
%
% Connected to backend ris-whois10.ripe.net

route:        162.55.0.0/16
origin:       AS24940
descr:        HETZNER-AS Hetzner Online GmbH, DE
lastupd-frst: 2026-05-12 11:42Z  194.28.97.2@rrc00
lastupd-last: 2026-09-18 07:58Z  5.188.4.211@rrc24
seen-at:
rrc00,rrc01,rrc03,rrc04,rrc05,rrc06,rrc07,rrc10,rrc11,rrc12,rrc13,rrc14,r
rc15,rrc16,rrc18,rrc19,rrc20,rrc21,rrc22,rrc23,rrc24,rrc25,rrc26
num-rispeers: 363
source:       RISWHOIS
```

RIPE RISwhois, running on standard whois port 43, showing the visibility of a prefix.

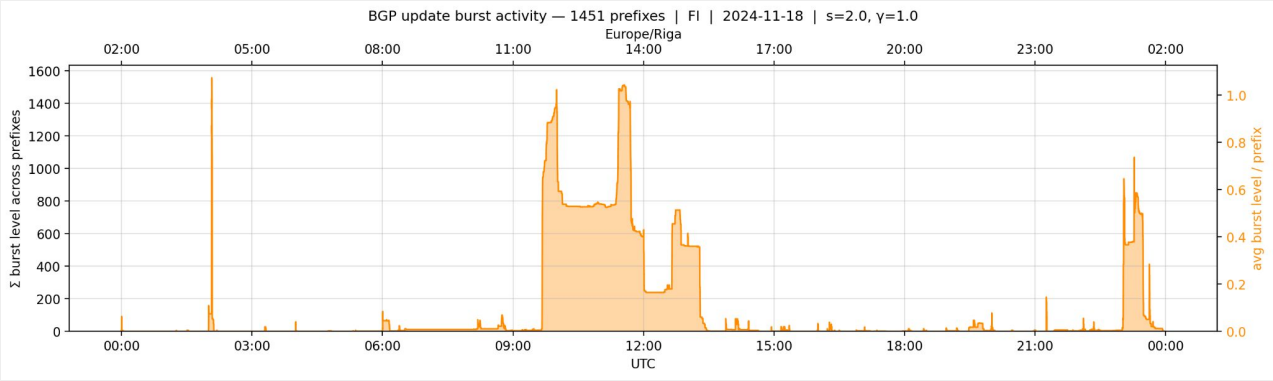
How is RIS Data Used?

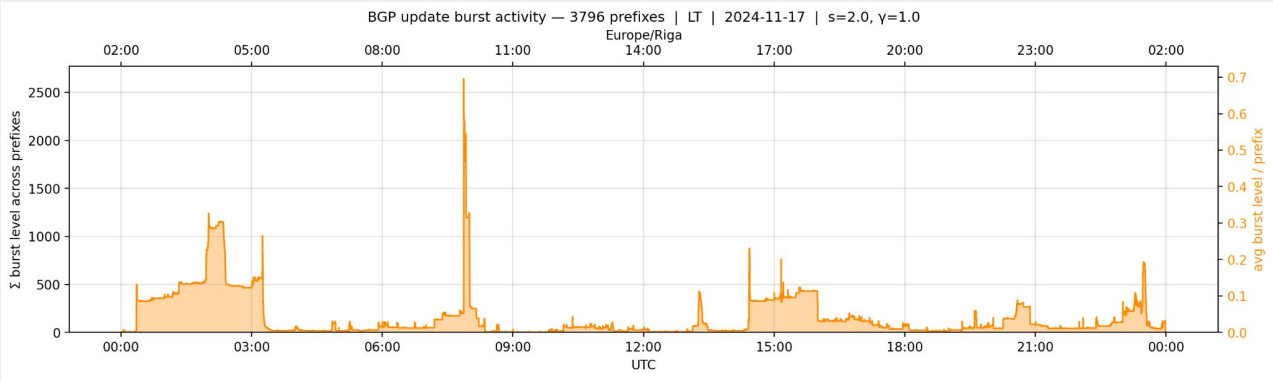
By us: routing analysis

- In our tools
- In our internet analysis

This example:

- Cable damage caused a spike in BGP updates for affected prefix
- Hard to find the needle in the haystack!

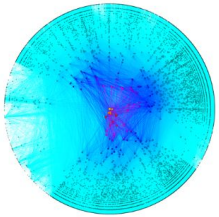




How is RIS Data Used?

By others:

- Often used in research
  - Example: CAIDA ASRank
- In open source software
- In other products



ASRank is CAIDA's ranking of Autonomous Systems (AS) (which approximately map to Internet Service Providers) and organizations (Orgs) (which are a collection of one or more ASes). This ranking is derived from topological data collected by CAIDA's Archipelago Measurement Infrastructure and Border Gateway Protocol (BGP) routing data collected by the Route Views Project and RIPE NCC.

ROUTE VIEWS 6447 RIPE NCC RIS

ASes and Orgs are ranked by their customer cone size, which is the number of their direct and indirect customers. Note: We do *not* have data to rank ASes (ISPs) by traffic, revenue, users, or any other non-topological metric.

ASN name or number search

1 2 3 4 .. 1987

ASes are sorted by their relationship, which is inferred from observed BGP paths. Note: we do not have data to infer financial arrangements between ASes.

AS Rank ▲	AS Number ▽	Organization		cone size (ASes) ▽
1	3356	Level 3 Parent, LLC		54887
2	1299	Arelion Sweden AB		42167
3	3257	GTT Communications Inc.		41285
4	174	Cogent Communications, LLC		41127
5	2914	NTT America, Inc.		25338
6	6939	Hurricane Electric LLC		22380
7	6453	TATA COMMUNICATIONS (AMERI...		21929
8	6461	Zayo Bandwidth		20242
9	6762	Telecom Italia S.p.A.		19766

How is RIS Data Used?

By others: Routing Analysis

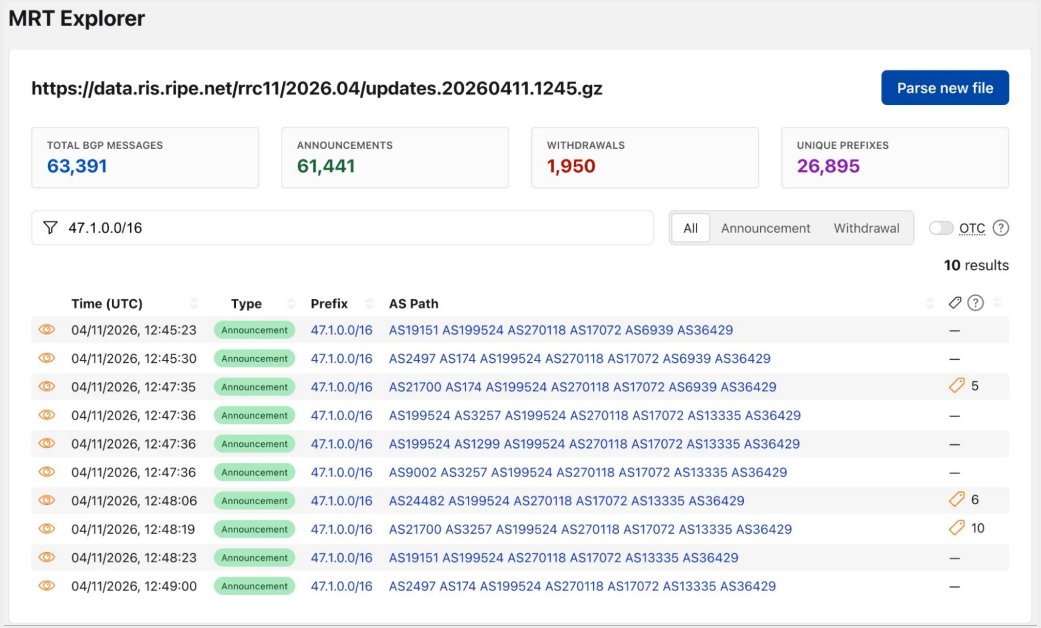
- Cloudflare blog: Enforcing the First AS in BGP AS_PATHs
- Blog shows multiple ways to query RIS data
- `monocle` - search for event
- MRT explorer - view raw data

```
→  ~ monocle search --start-ts 2026-04-13T00:20:00Z --end-ts 2026-04-13T00:23:59Z --prefix 90.98.0.0/15 --collector rrc26 --json
{
  "aggr_asn": null,
  "aggr_ip": null,
  "as_path": "48237 1299 199524 270118 17072 41128",
  "atomic": false,
  "collector": "rrc26",
  "communities": null,
  "local_pref": 0,
  "med": 0,
  "next_hop": "185.1.8.3",
  "origin": "IGP",
  "peer_asn": 48237,
  "peer_ip": "185.1.8.3",
  "prefix": "90.98.0.0/15",
  "timestamp": 1776039612.0,
  "type": "ANNOUNCE"
}
```

How is RIS Data Used?

By others: Routing Analysis

- Cloudflare blog: [Enforcing the First AS in BGP AS_PATHs]
- Blog shows multiple ways to query RIS data
- `monocle` - search for event
- MRT explorer - view raw data



How to use RIS Data Yourself?

Example: RIPEstat Looking Glass API

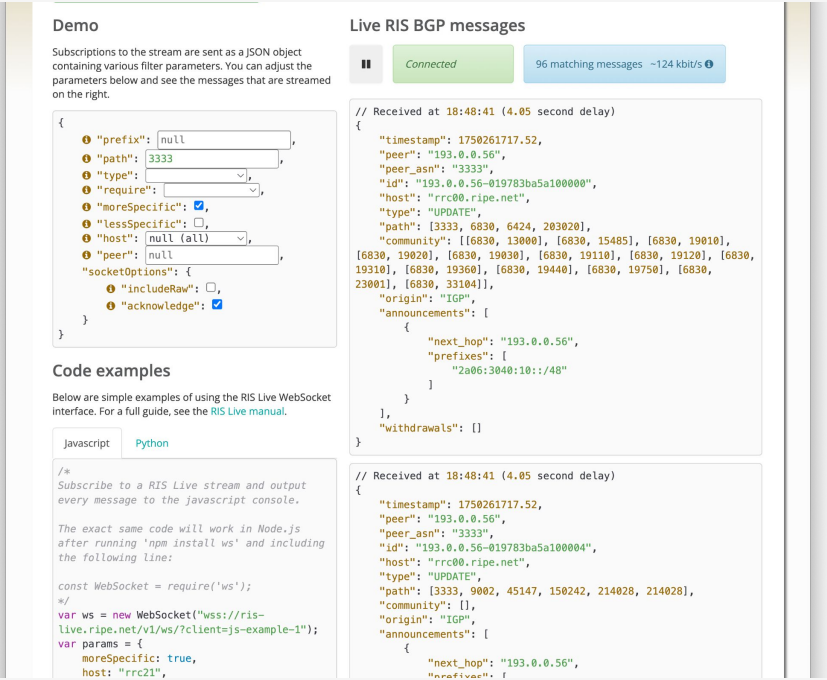
- Use it like any REST API
- Many ways to process this data

```
❯ curl -s "https://stat.ripe.net/data/looking-glass/data.json?resource=193.0.14.0/24"
 | jq '.data.rrcs[]'
{
  "rrc": "RRC00",
  "location": "Amsterdam, Netherlands",
  "peers": [
    {
      "asn_origin": "25152",
      "as_path": "34854 6939 25152",
      "community": "34854:1000",
      "largeCommunity": "",
      "extendedCommunity": "",
      "last_updated": "2025-06-18T10:55:20",
      "prefix": "193.0.14.0/24",
      "peer": "2.56.11.1",
      "origin": "IGP",
      "next_hop": "2.56.11.1",
      "latest_time": "2025-06-18T21:44:55"
    },
    {
      "asn_origin": "25152",
      "as_path": "59919 25152",
      "community": "25152:1028 59919:65001 64512:11 64512:21 64512:31",
      "largeCommunity": "25152:1028:61968",
      "extendedCommunity": "64512:11",
```

How to use RIS Data Yourself?

Example: RIS-live

- ris-live.ripe.net
- A websocket with BGP update messages in JSON format
- Most suitable for getting a *filtered stream* of BGP updates.



"Easy" to use from your own code.

Also used in open source projects and commercial projects.

Conclusion

You *are* probably using RIS data

- RIS is one of the key tools for capturing the development of the Internet
- You *are* probably using RIS data indirectly
- You *can* use this data yourself
  - Curious to hear your use case

Questions & Comments



aformoso@ripe.net,
ris@ripe.net, ris-peering@ripe.net
stat@ripe.net

Prototype: Querying recent bview files

Sample query

```
D SELECT
    prefix, origin_as, as_path,  peer_ip, peer_asn, collector
  FROM
'https://data.ris.ripe.net/derived/prototypes/parquet/v0/bview/year=2026/month=09/day=21/20260921.0000-bview-all-collectors.parquet'
  LIMIT 10;
```

prefix varchar	origin_as varchar	as_path varchar[]	peer_ip varchar	peer_asn uint32	collector varchar
0.0.0.0/0	174	[263237, 174]	187.16.222.45	263237	rrc15
0.0.0.0/0	3356	[271253, 3356]	187.16.209.34	271253	rrc15
0.0.0.0/0	61568	[28624, 61568]	187.16.216.103	28624	rrc15
0.0.0.0/0	6762	[13786, 6762]	187.16.217.161	13786	rrc15
1.0.128.0/17	23969	[24482, 58453, 38040, 23969]	187.16.208.144	24482	rrc15
1.0.128.0/17	23969	[271253, 6939, 38040, 38040, 38040, 38040, 23969]	187.16.209.243	271253	rrc15
1.0.128.0/17	23969	[271253, 6939, 38040, 38040, 38040, 38040, 23969]	187.16.209.34	271253	rrc15
1.0.128.0/17	23969	[60503, 6939, 38040, 38040, 38040, 38040, 23969]	187.16.214.32	60503	rrc15
1.0.128.0/17	23969	[28624, 6939, 38040, 38040, 38040, 38040, 23969]	187.16.216.103	28624	rrc15
1.0.128.0/17	23969	[28634, 273478, 3356, 38040, 23969]	187.16.216.209	28634	rrc15

10 rows 6 columns

Note: only the last 14d of data are kept, you might need to adapt this URL

Prototype: Querying recent bview files

There’s more data… (also BGP *updates*)

```
D DESCRIBE 'https://data.ris.ripe.net/derived/prototypes/parquet/v0/bview/year%3D2026/month%3D09/day%3D21/20260921.0000-bview-all-collectors.parquet';
┌──────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────┐
│ https://data.ris.ripe.net/derived/prototypes/parquet/v0/bview/year%3D2026/month%3D09/day%3D21/20260921.0000-bview-all-collectors.parquet │
│                                                                                                                                              │
│ ts                                                                                                                                timestamp  │
│ operation                                                                                                                         varchar    │
│ prefix                                                                                                                            varchar    │
│ origin_as                                                                                                                         varchar    │
│ as_path                                                                                                                           varchar[]  │
│ peer_ip                                                                                                                           varchar    │
│ peer_asn                                                                                                                          uinteger   │
│ nexthop                                                                                                                           varchar    │
│ origin                                                                                                                            varchar    │
│ localpref                                                                                                                         uinteger   │
│ med                                                                                                                               uinteger   │
│ aggregator_ip                                                                                                                     varchar    │
│ aggregator_asn                                                                                                                    uinteger   │
│ only_to_customer_asn                                                                                                              uinteger   │
│ communities                                                                                                                       varchar[]  │
│ collector                                                                                                                         varchar    │
│ unknown_attributes                                                                                                                uinteger[] │
│ deprecated_attributes                                                                                                             uinteger[] │
└──────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────┘
```


Note: besides the daily *bview*, there’s also hourly *updates*