

Smart Data Filtering in the RIPE Atlas Measurement Platform

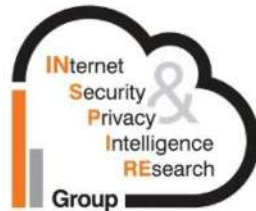
Spyridon-Andreas Siskos^{1,2}, Petros Gigis^{1,3}, Lefteris Manassakis¹ and Xenofontas Dimitropoulos^{1,2}

{asiskos, leftman, fontas} @ics.forth.gr {p.gkigkis} @cs.ucl.ac.uk

¹ICS-FORTH, Greece ² University of Crete, Greece ³ University College London



www.ics.forth.gr



www.inspire.edu.gr



www.uoc.gr

Background and Motivation

- Researchers and network operators, use data plane measurements for network monitoring and diagnostics.
- RIPE Atlas is an Internet measurement platform that allows users to make their own campaigns and access those data through a REST API.
- But it lacks more advanced filtering, which would help users to take advantage of those data more efficiently.

In this work

- Enable advanced filtering over RIPE Atlas API.
- Explore RIPE Atlas collected data more efficiently.
- Use filtering over traceroute data.
- Provide an open-source framework.

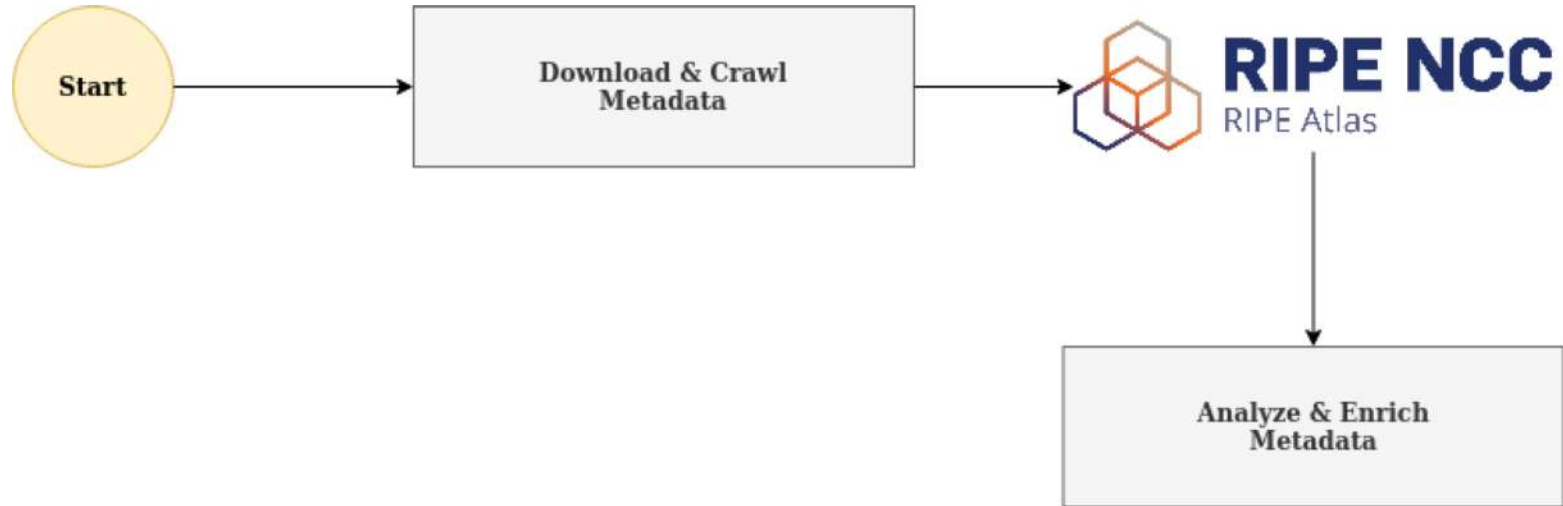
Prototype Framework Architecture

- Python 3
- Django web framework
- Relational PostgreSQL DB
- Docker

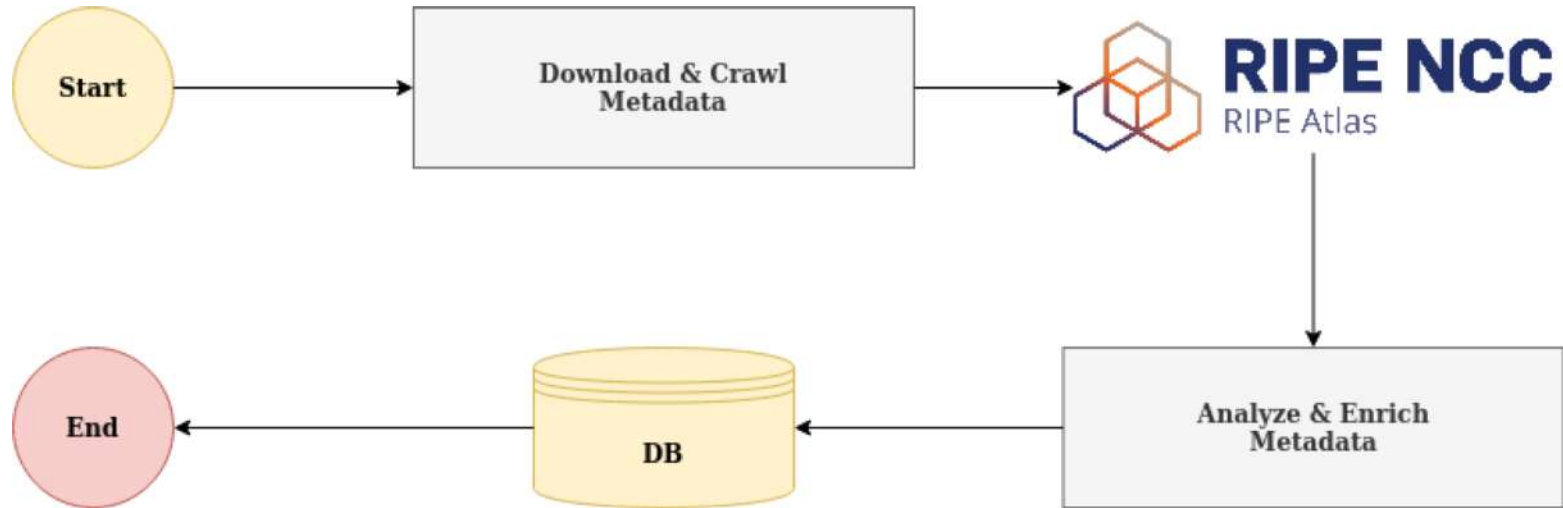
1. Download & Crawl Metadata from RIPE Atlas.



2. Analyze and Enrich Metadata



3. Insert Metadata into our Database.



Indexed Metadata

Traceroutes:

- msm_id
- prb_ids
- from_addr
- dst_addr
- src_asn
- dst_asn
- creation_time
- stop_time

Probes:

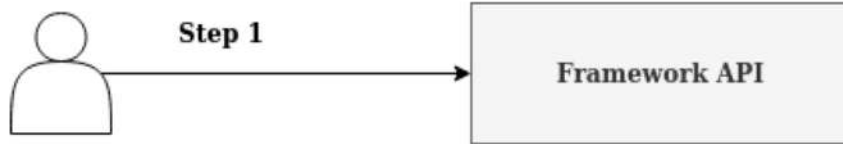
- prb_id
- latitude
- longitude
- creation_time
- stop_time
- asn_v4
- asn_v6
- country_code

Measurements:

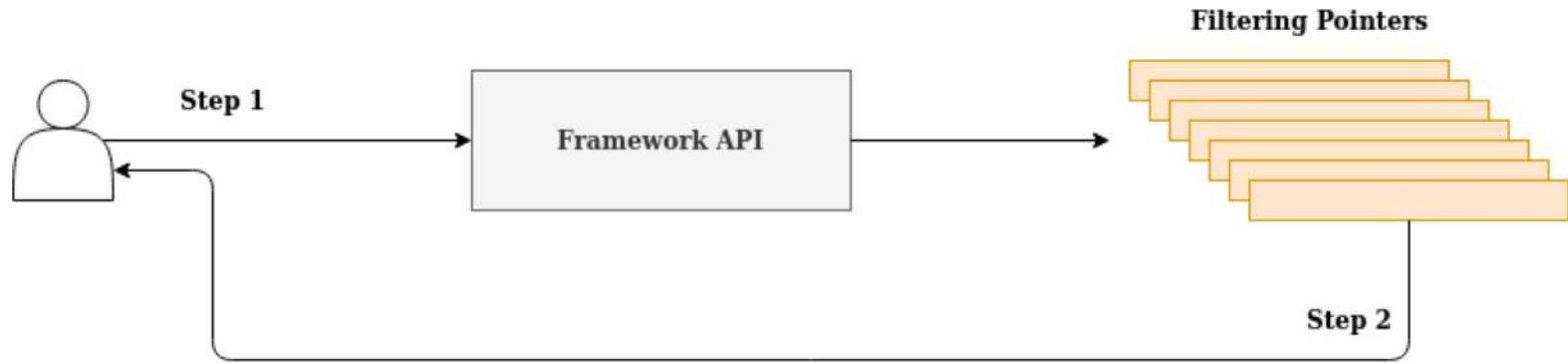
- msm_id
- address_fam
- description
- protocol
- creation_time
- stop_time
- target_asn
- target_ip
- source_asns
- prb_list

How can a user take advantage of
our API?

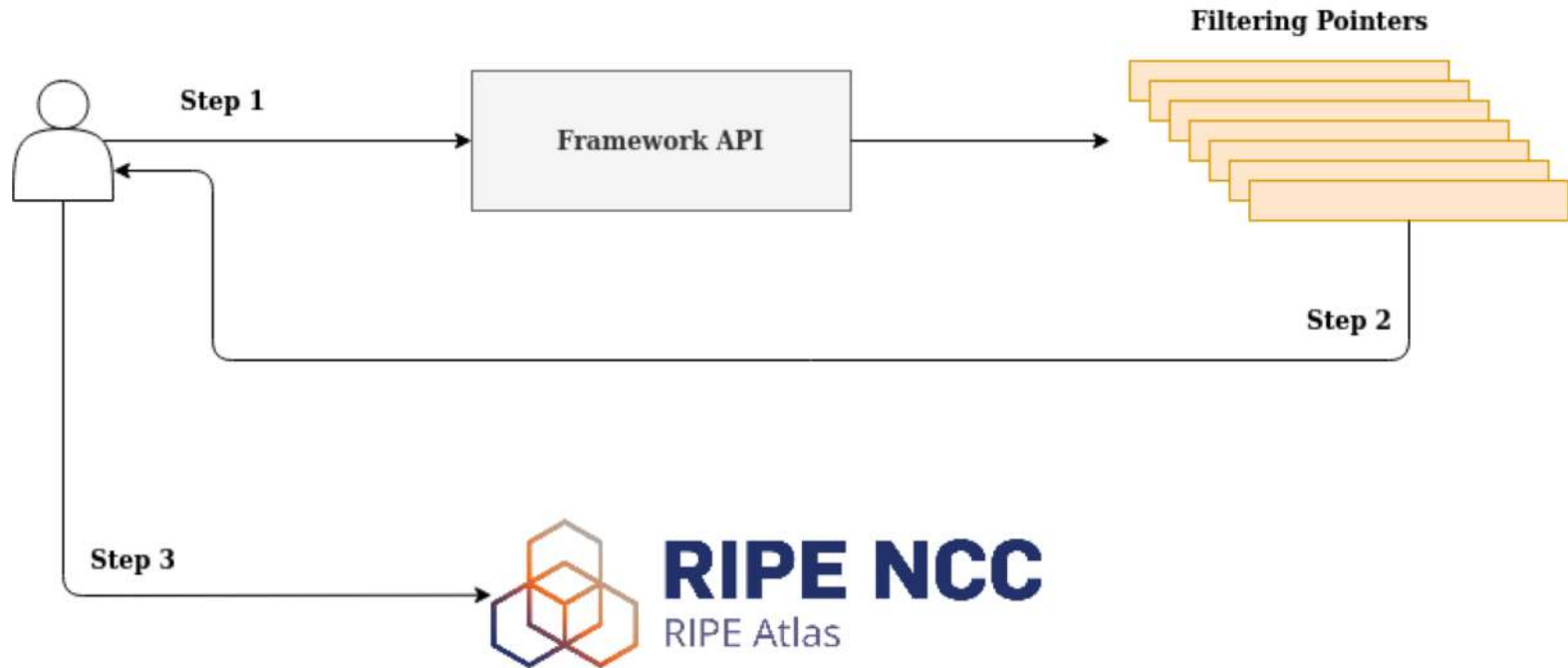
1. User queries our API.



2. API returns set of URL's pointing to RIPE Atlas.



3. User retrieves results based on initial query.



Real example:

A User wants to fetch all traceroutes

from

Vodafone (AS1273)

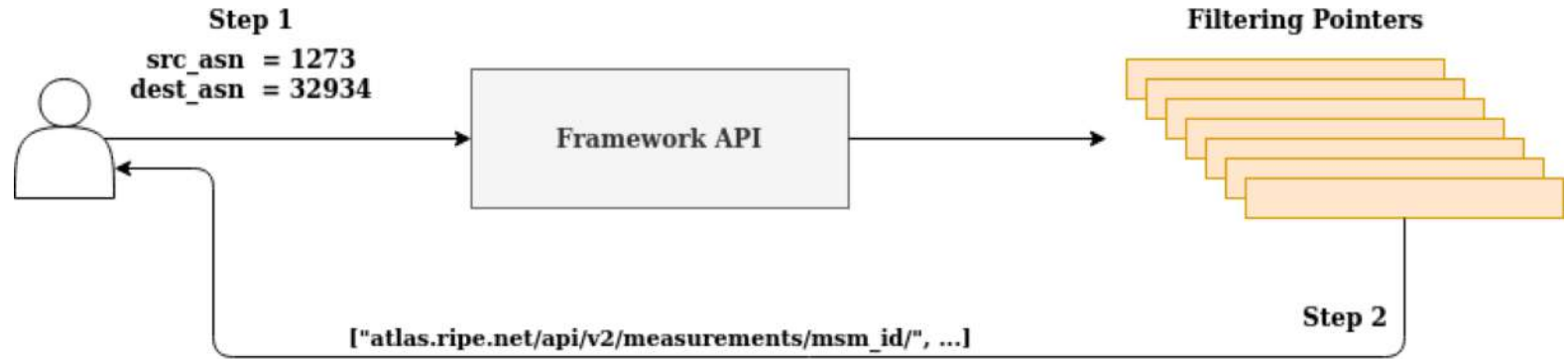
Towards

Facebook (AS32934)

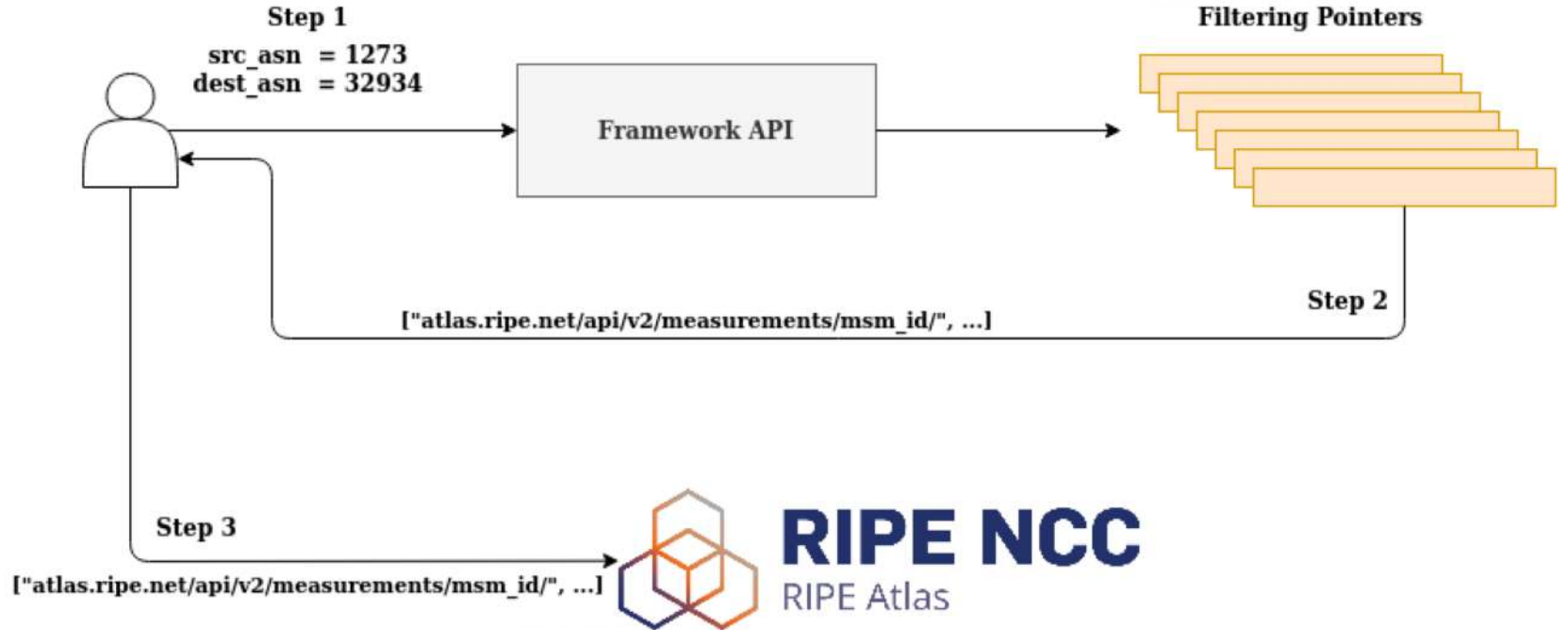
1. User issues an advanced query to our API. In our example sends a GET Request with parameters `src_asn = 1273` & `dest_asn = 32934`



2. API returns set of URL's pointing to RIPE Atlas.



3. User retrieves results from RIPE Atlas based on initial query.



Conclusions

- Built an open source prototype for filtering RIPE atlas data
- Preliminary results show that it can significantly accelerate advanced queries

Future Work

- Extend the framework API functionality (e.g. support all measurement types such as HTTP, DNS, e.t.c.).
- Improve Performance and Scalability.
- Fetch data automatically from RIPE Atlas Stream.

