Building a Big Data platform with open source

RIPE 87
Rome, Italy
27 Nov - 1 Dec 2023

DE-CIX – Christian Petrasch
Product Owner Service Insights

DE-CIX

Why do we want to do this?

- Big benefit, especially for smaller customers (sometimes **blind without a provided solution**)
- We want to have deep, **observable insights** into our data with **state of the art technics**
- Benefit for us: Additonal tool for network planning
- Updates our old, limited metrics system

- The Internet is changing..

  .. and traffic is increasing

DE CIX

Datasources & challenges

- Get Telemetry data from Interconnection Platform Router
  - **appr. 10.000 values/5min**
  (Statistics like port or error counter which are available on the routers)

- Get IPFix data from Interconnection Platform Router
  - **appr. 300.000 pkts./s in peak**
  (BGP Flow data related to netflow data)

- Challenge of filtering, enriching, aggregating, analyzing and displaying that huge amount of data just in time
  (~5 GiB raw data/second)

Solution – a 3-steps approach

Start
the toolbox
shopping trip

The datahose & datalake

- Datalake **Clickhouse**
- Datahose - **Apache Kafka** message queue
- Both **blazing fast** – great scalability
- Broad usage in community
- Work together really well

DE-CIX IXPs

The collector & the battle of transport formats

- First approach: **JSON**
- Is too really big (~810 bytes/packet) …

  first lesson learned .. **slow parsing**
- **Binary** transport format will be a much better option

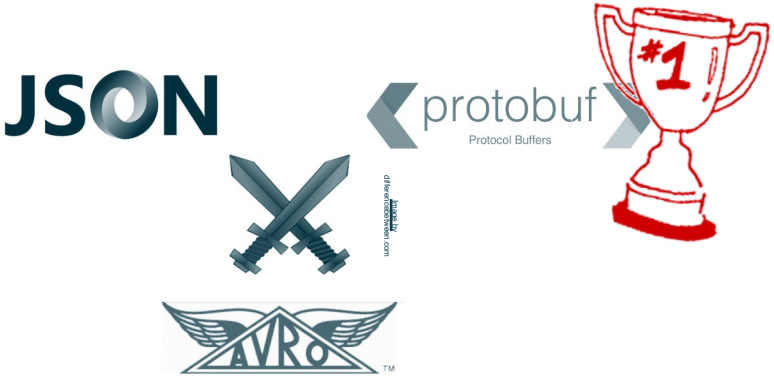
  ⇨ smaller packets ⇨ faster parsing

The collector & the battle of transport formats

- First approach: **JSON**
- Is too really big (~810 bytes/packet) …

  first lesson learned .. **slow parsing**
- **Binary** transport format will be a much better option

  ⇨ smaller packets ⇨ faster parsing



The collector & the battle of transport formats

- Only **goflow2** supports protobuf (at that time)
- *Finding*: Nokia has another IPFix template as other vendors
  (Remark: Interconnection Platform is built on top of Nokia Routers)

- Decision ⇨ **goflow2** (https://github.com/netsampler/goflow2)

Thanks to the lead developer –
Louis Poinsignion

Where to run the collector..?

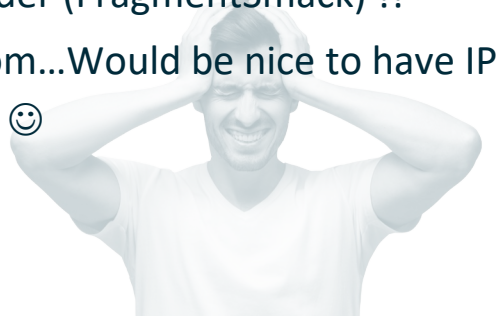
Idea: Running goflow2 collector in Kubernetes to scale

- But our infrastructure runs in cloud
- Sending IPFix UDP Packets to cloud results in **massive packet loss**.

  Why ?

  MTU in cloud provider network is 1400 bytes – IPFix packets are 1460 bytes

- For DDoS protection reasons cloud provider **drops UDP fragments**, which are not in original order (FragmentSmack) !!
- For Nokia people in the room…Would be nice to have IPFIX export MTU size configure option! ☺



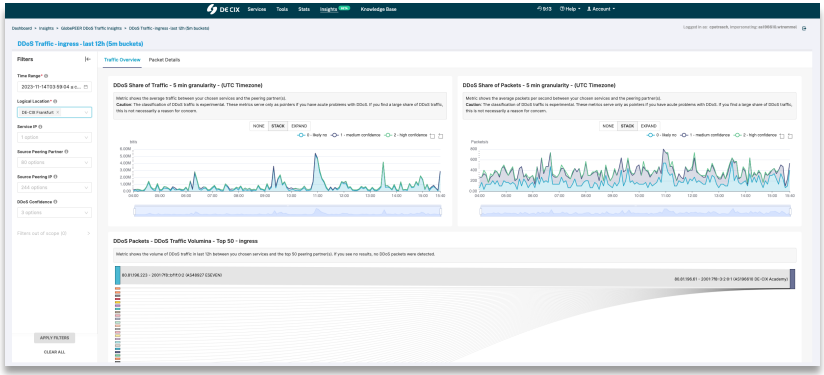
DECISION:

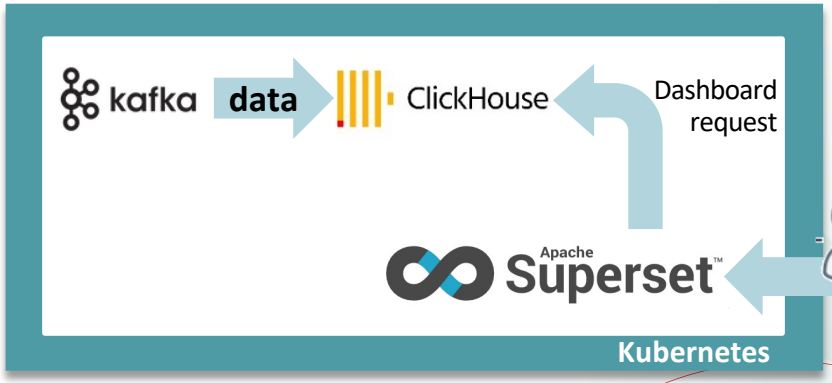
goflow2 has to run on premise ☹

second lesson learned !

The Front end

- Dashboards - Apache Superset
- **Really nice embedding framework function with authentication token**
- Row level security per customer for displayed data
- We contracted a pentest company - No critical findings
- One smaller finding was reported to community...and fixed meanwhile





DE CIX

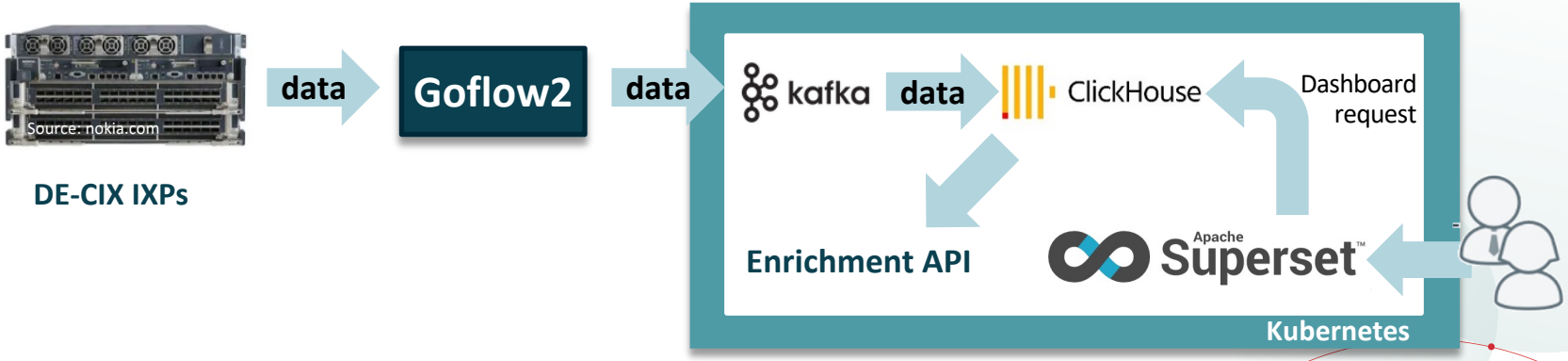
Shopping finished… Let’s start cooking

The Enrichment

Clickhouse Dictionaries

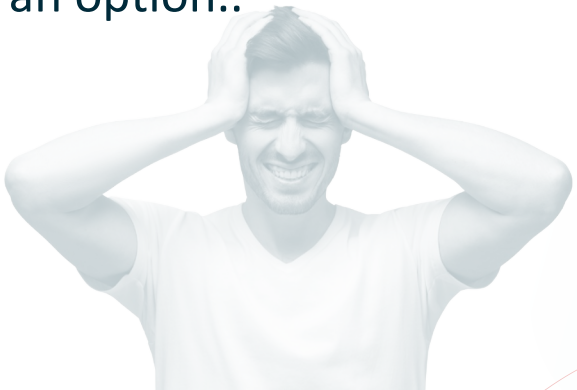
- Extremly fast because of in-memory ☺
- Dictionary is pulled periodically from Business Data source
- In our case data source is a proprietary web API, but databases and every http api are also possible

DE-CIX IXPs



The Aggregation

- **Huge pile of raw data**
- Can‘t store all data in raw granularity ⇨ need a solution
- **5 min, 1hr, 6hr bucket** aggregation
- Materialized Views could be an option..



The Materialized Views

- **Permanently** running query looking for **insert triggers** of a database table
  (A little bit like INOTIFY in Linux but for database tables)

- Really fast to add or manipulate data, running in memory

But…It **never** knows the data which it is working with..
Insert trigger only… (remember that ☺)

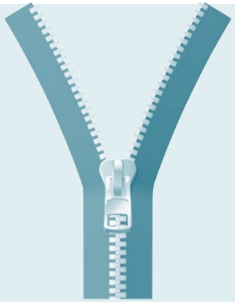
Database Schema matters…

Issues:

- **Heavy database load**, replication delays, tremendous slow queries
- **Strange/wrong data** in aggregated buckets

Root causes

- **Stacked MV** and replication with real high input data rates
- Material View for performance reasons: **insert only data if a specific bucket size is full** !!
- When aggregation is running it only knows that “partial data“ (insert trigger)

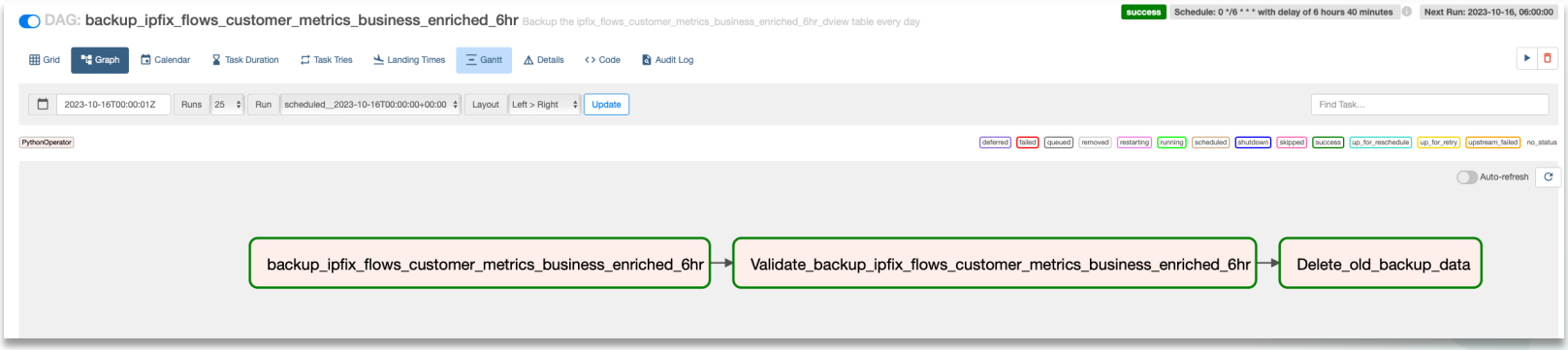


SOLUTION:

Apache Airflow

Airflow to the rescue

- Use a scheduler instead of MVs (**batch processing** instead)
- Write MV tasks as scheduled python/sql scripts.. ☺
- Flexible job control, catch-up failed jobs, dependent DAGs



Next step – integrate Anonymization of IPs (GDPR)

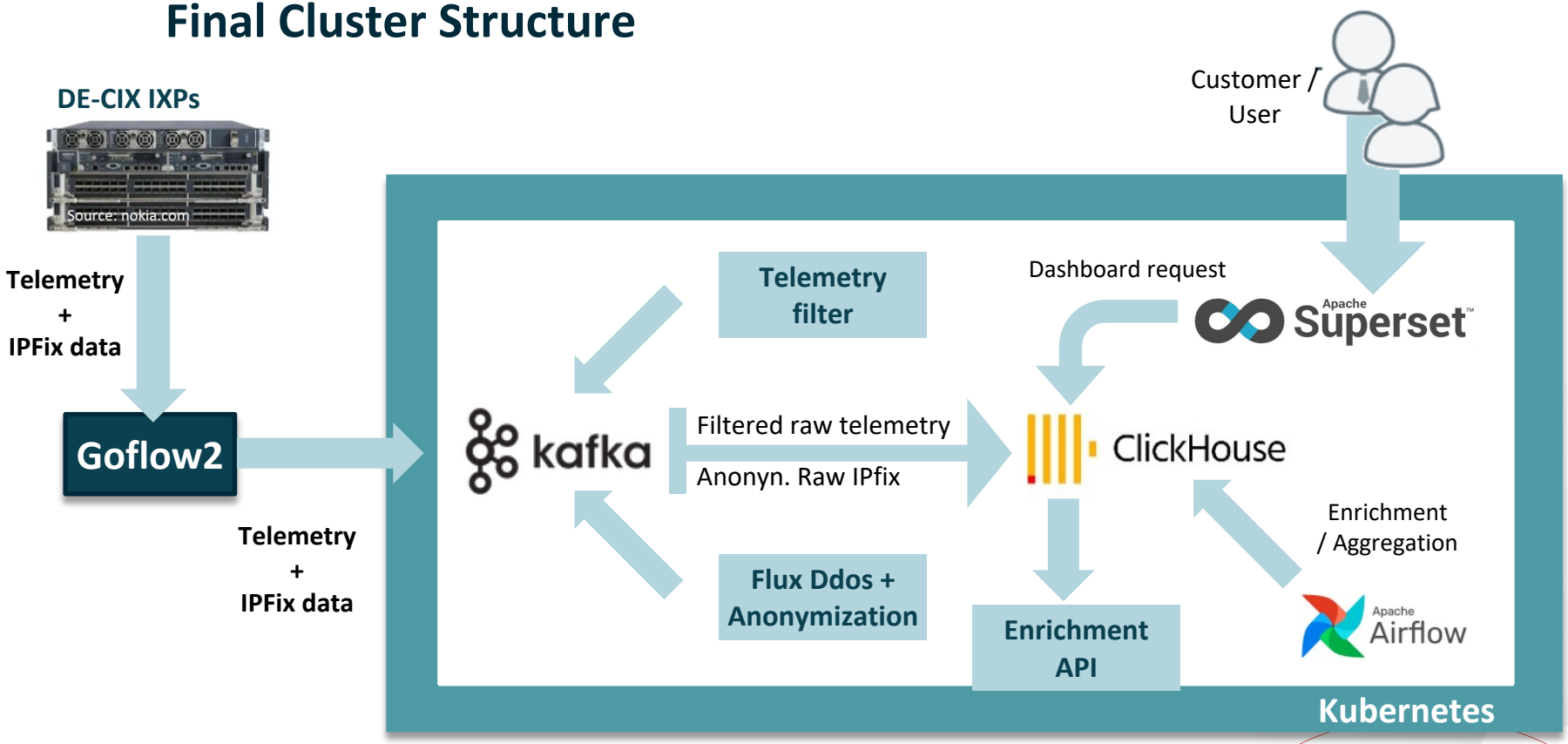
- **Flux - Proprietary tool for DDoS Statistics**
  (Algorithm Presentation @RIPE84, M.Wichtlhuber)
- Written in Rust, reading protobuf from Kafka, really fast
- Good Clickhouse integration
- Adding anonymization was easy..(M. Wichtlhuber said ☺)

Anonymizer has to developed on your own.
Maybe ours will be open source somewhere.

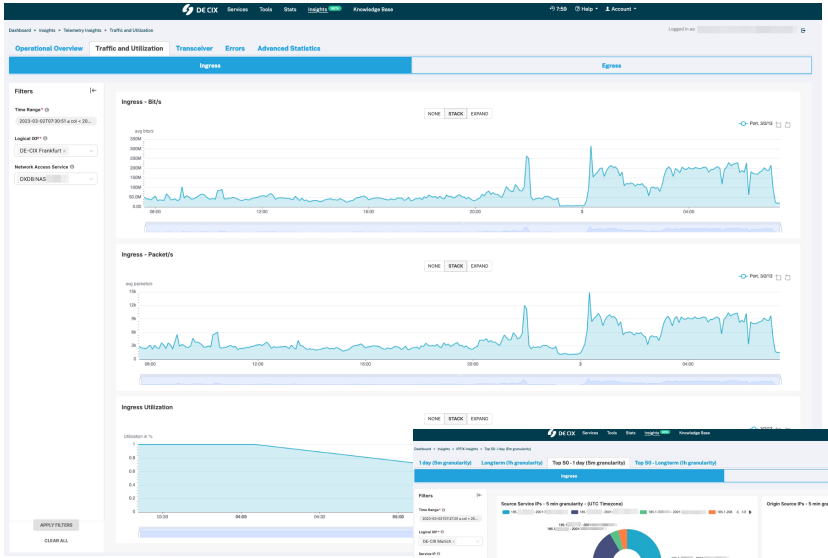
Implement Telemetry

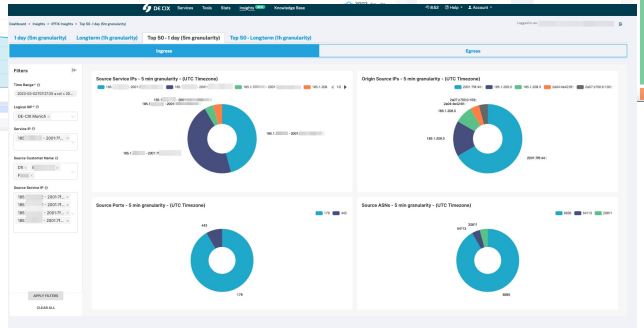
- **Get all data from routers** of Interconnection Platform by GNMI subscription/gRPC protocol
- Send it to Kafka by script (proprietary)
- **Remove the ones we don‘t need by telemetry filter** (proprietary)
- Enrich and aggregate it via Airflow
- Show it on dashboards

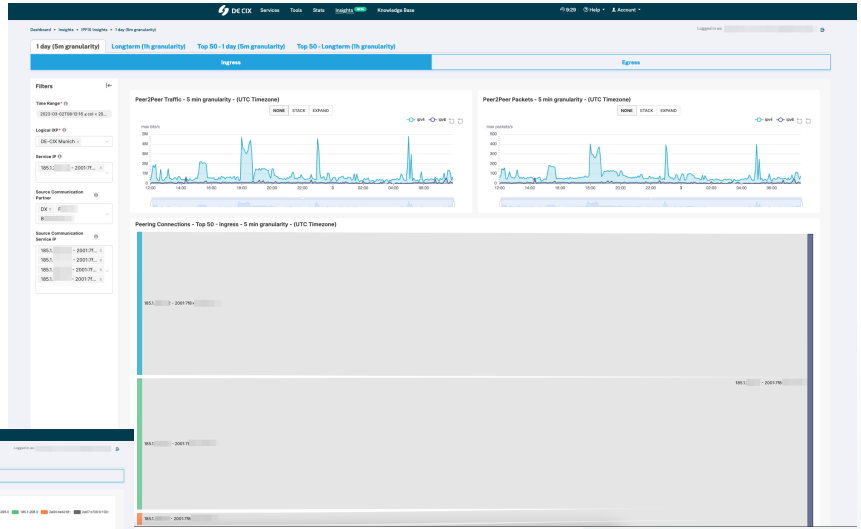
Final Cluster Structure



And that’s how it can look







With these great tools you can also build a system like this!

Our insights@de-cix

- Multiple insights for ingress and egress traffic
- Traffic metrics showing flow traffic with communication partners
- Top talker metrics
- Top 50 analytics pie charts
- Telemetry of your ports and services

- Transceiver RX/TX optical values
- Resellers can view metrics of their customers’ ports
- Statistics of DDoS Traffic
- Statistics of Cloud ROUTER telemetry

DE CIX

Q&A

- Or want to have a look in a live demo in a coffee break?
- Would love to get in touch
- Contact me: Christian.Petrasch@de-cix.net

RIPE 87
Rome, Italy
27 Nov - 1 Dec 2023

Thank you

Christian Petrasch

DE-CIX – Product Owner Service Insights

Christian.petrasch@de-cix.net

DE CIX

What we give back to the community

- We have a Clickhouse development support contract
- We did a Superset Embedding Framework pentest and gave the results to the community
- DE-CIX supports several open source projects like: Alice, peeringDB, IX-API…

Licensing of open source tools

- Apache Superset (Apache Licence - free to use)
- Apache Airflow (Apache Licence - free to use)
- Apache Kafka (Apache Licence - free to use)
- Apache Zookeeper (Apache Licence - free to use)
- Clickhouse (Apache Licence - free to use)
- PostgreSQL (PostgreSQL Licence - free to use)