A Small Data Approach to Flow Analysis

Shannon Weyrick
Head of Technology, NetBox Labs
sweyrick@netboxlabs.com

https://orb.community/

netbox labs

ORB

modernizing network observability

https://orb.community/

Modernizing Open Source Network Observability

- Flexible and Modular Data Sources
- Agent Fleet Management
- Observability Pipelines with OpenTelemetry
- Dynamic Policy Orchestration
- Small Data

so what is small data?

The Data Conundrum

We *think* we want **all the data**

...because we think we may use it all *someday*

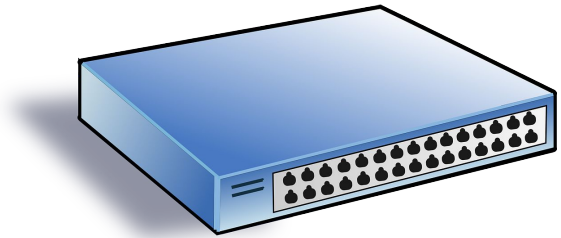
What we *actually* want are **targeted insights**

...to help us operate, debug, scale and protect our networks *today*

There is a price to pay for streaming raw data to a centralized solution

Case Study

Managed Authoritative DNS provider



- Global presence with 25 anycasted POPs
- Servicing over 100 billion DNS queries a day
- Over 70 million flow records per day
- 3.5 TB of storage needed for 30 days of flow history
- Custom data pipeline based on various OSS tools

Data Challenges

- Hard to maintain and to scale data pipelines
- Inability to make sense of all the data
- Slow dashboards
- Short retention times
- Significant ingestion and storage costs

orb

feature and architecture overview

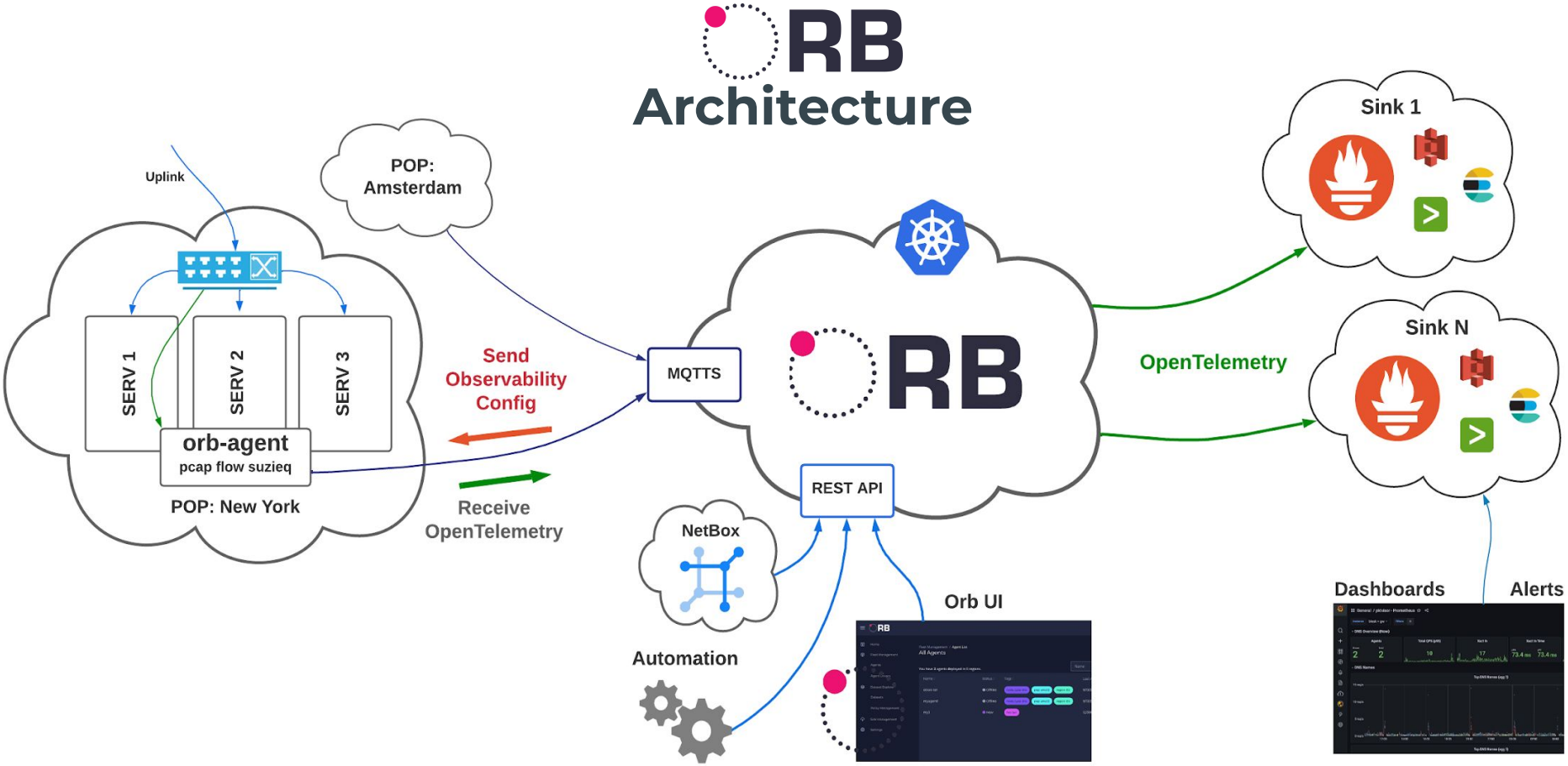
https://orb.community/

Orb Overview

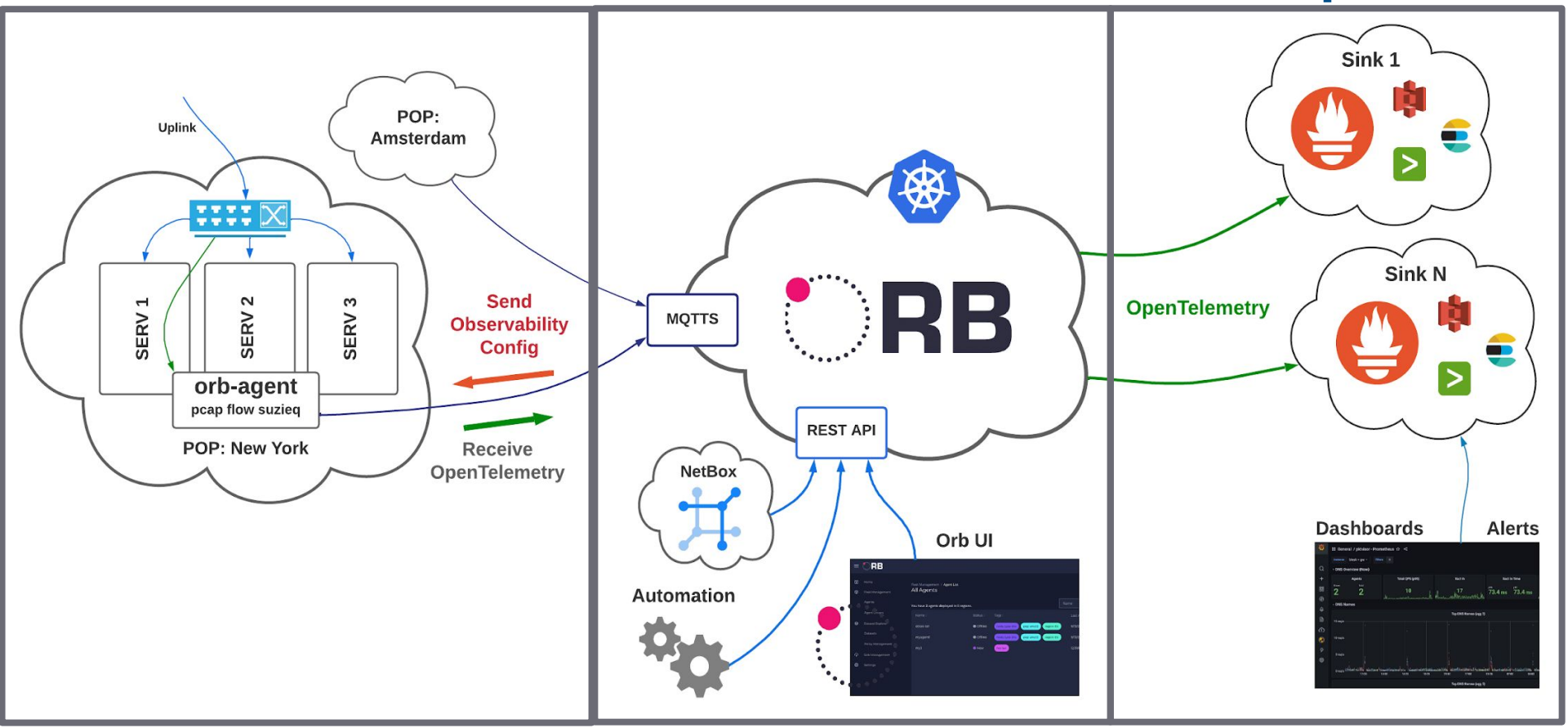
- Usability & Automation: Portal UI & REST API
- Fleet management: connect, organize, and manage agents
- Policy management: flexible recipes for analyzing data streams and devices
- Sink management: which databases can metrics and logs be sent to
- Configuration management: which groups of agents should be running which policies, updated in real time
- Pipelines with OpenTelemetry: metric and log output from all policies across all agents, exported to the proper databases and dashboards

https://orb.community/

RB Architecture



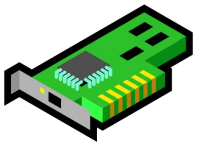
https://orb.community/



https://orb.community/

Orb Edge Agents

- Agent based system (outbound connect only)
- Tap into data streams from devices
  - flow (sFlow, Netflow, IPFIX)
  - packet analysis
  - SuzieQ
  - expandable via modules



sFlow

DNS

- Configured in real time with dynamic policies from control plane
- Fast streaming algorithms that analyze in-agent in real time
- Efficiently summarize important insights

quick start with orb

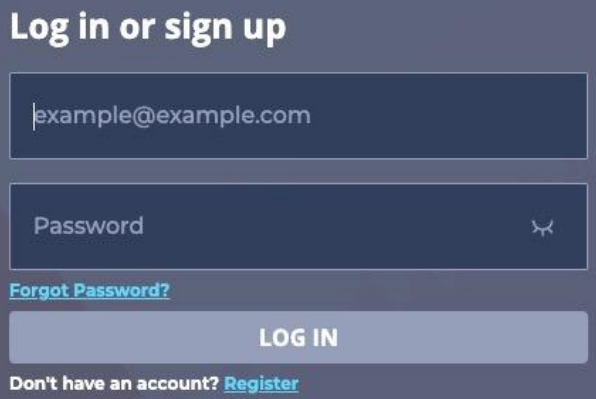
you can self-host, but use orb.live for free for the fastest start

https://orb.community/

step 1: sign-up for a free account at https://orb.live

ORB

An Open-Source dynamic network observability platform

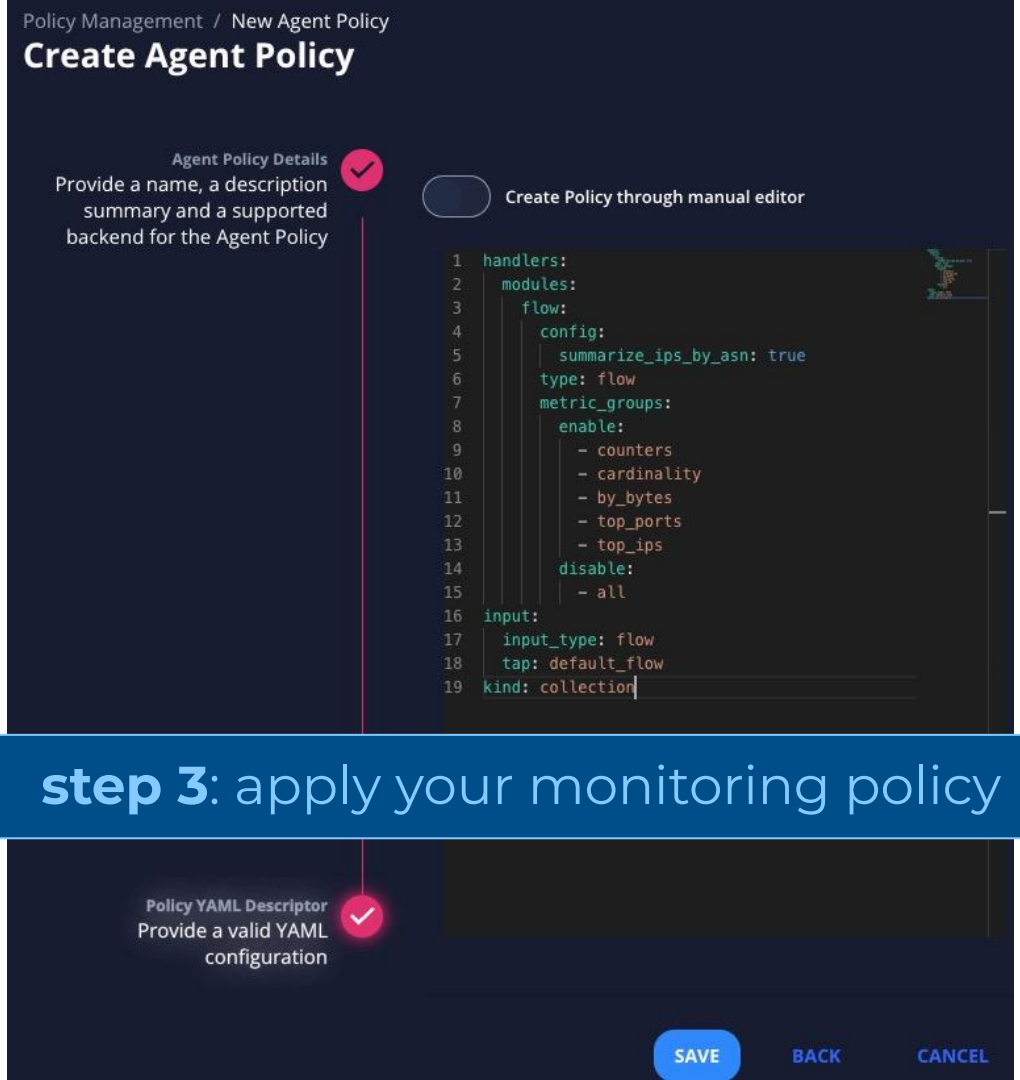


step 2: deploy your first agent

agent.yaml

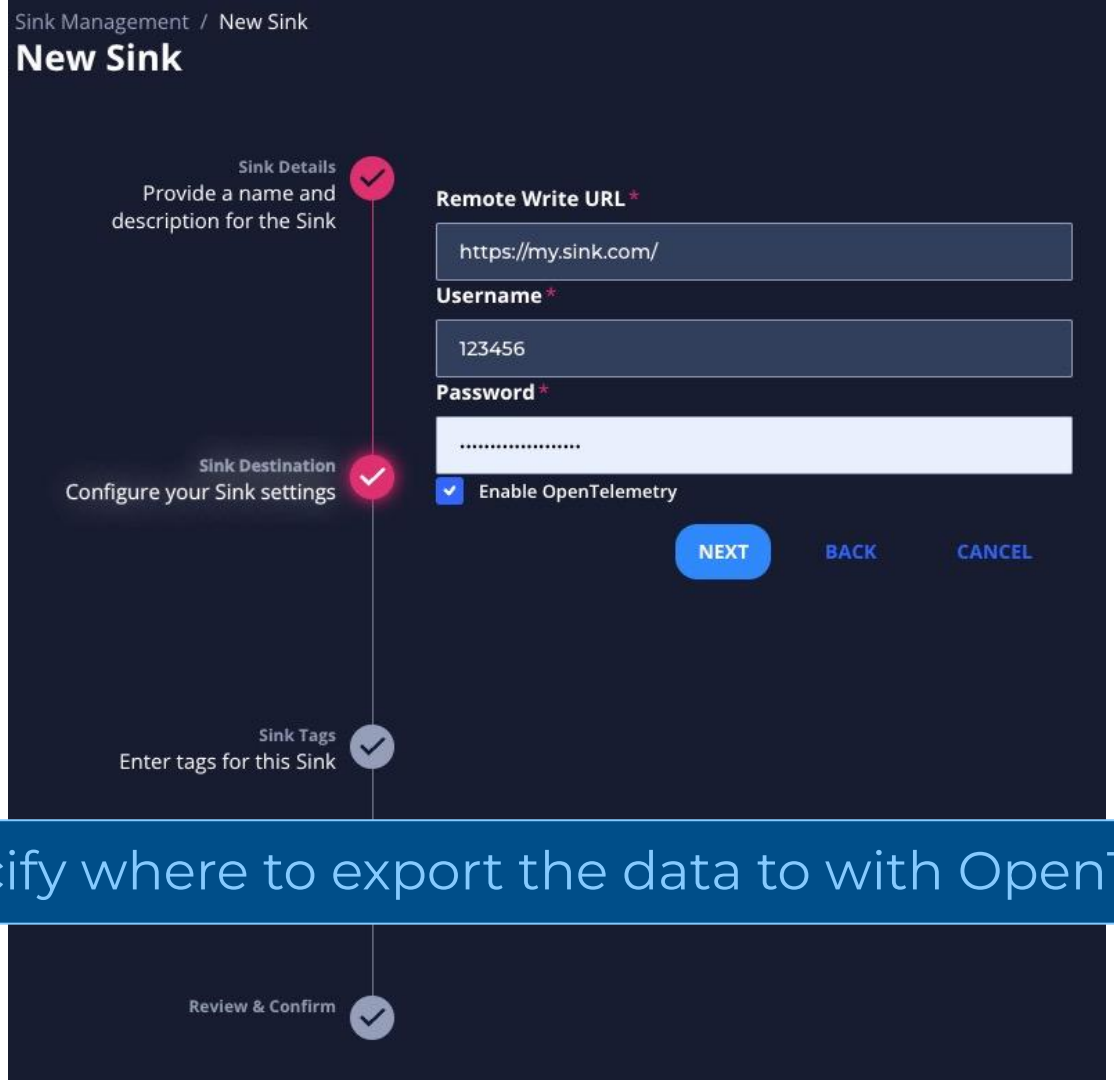
```
visor:
  taps:
    default_flow:
      input_type: flow
      config:
        port: 6343
        bind: 192.168.1.1
        flow_type: sflow
```

```
docker run -d --net=host \
-e ORB_CLOUD_MQTT_ID=a4315b19-1a6e-4ecb-9b87-9908c7b5c9cf \
-e ORB_CLOUD_MQTT_CHANNEL_ID=16bd1e66-dc05-442c-93ee-73a7cc6611ff \
-e ORB_CLOUD_MQTT_KEY=88463219-f869-43f6-925a-04b3790c1bca \
-v ${PWD}/agent.yaml:/opt/orb/agent.yaml \
ns1labs/orb-agent
```



step 3: apply your monitoring policy

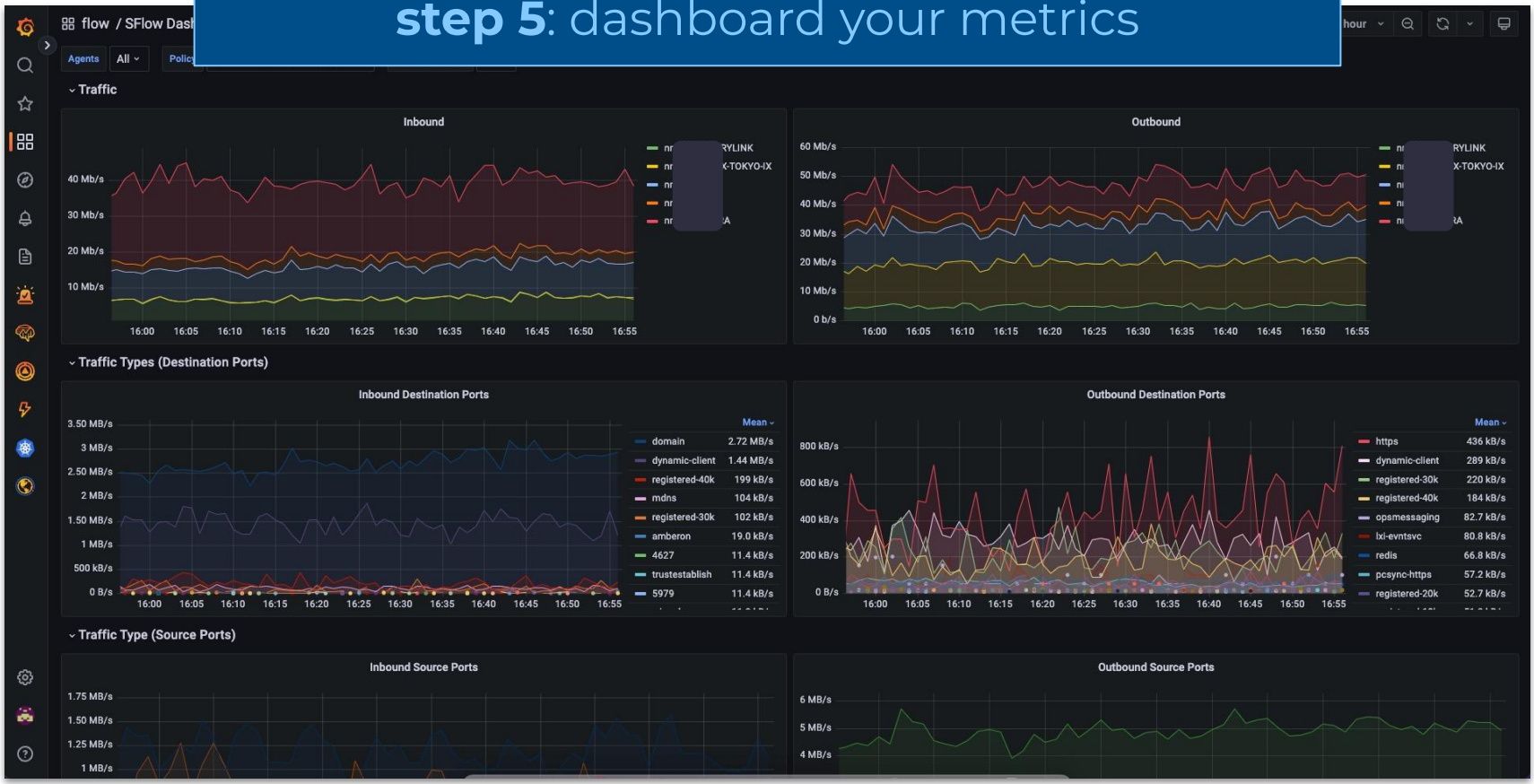
https://orb.community/



step 4: specify where to export the data to with OpenTelemetry

https://orb.community/

step 5: dashboard your metrics



flow analysis with orb

What is Network Flow Data?

Records that describe network connections, typically providing the following minimal set of information as a tuple:

- Source IP address
- Destination IP address
- IP Protocol
- Source port
- Destination port

Orb Policies for Flow

Filter

what data do you want to analyze

Enrich

how do you want to present the data

Summarize

how should the data be aggregated

Filter

Orb Agents act as flow collectors

Orb Policies allow you to analyze a subset of the received flows at the edge, enabling more tailored analysis

Different policies on the same agent can concurrently analyze different subsets of data and generate specific streams of metrics

```
filter:
  only_device_interfaces:
    1.1.1.1: [1,2,3]
    2.2.2.2: [2]
    3.3.3.3: [2,3]
```

Enrich

Orb Policies allow you to specify how the data should be presented, whether in raw form or relabeled for easier interpretation

Source IPs ⇒ **Device Names**

Interface Indexes ⇒ **Interface Names**

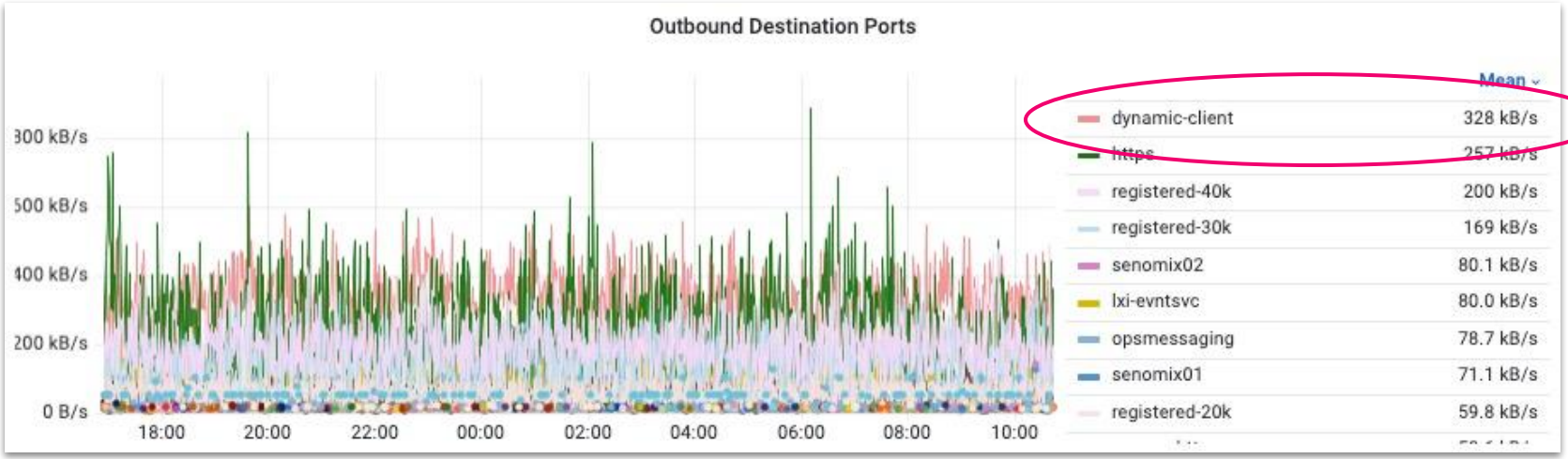
Port Numbers ⇒ **Service Names**

DSCP Values ⇒ **QoS Class Names**

Enrich & Summarize Ports

Based on IANA Service Name and Transport Protocol Port Number Registry

https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml

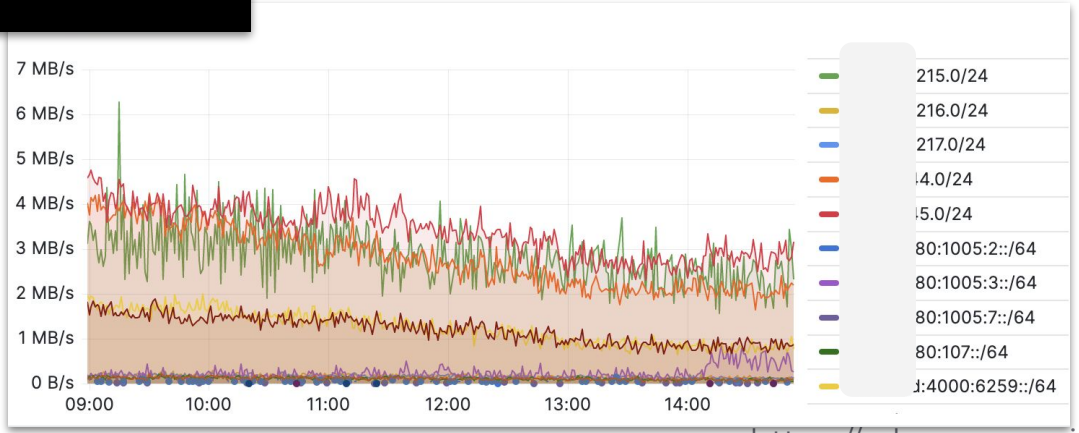


can be customized or upload your own

Summarize IPs by Subnet

Not all IPs are of interest, aggregates are sometimes more insightful

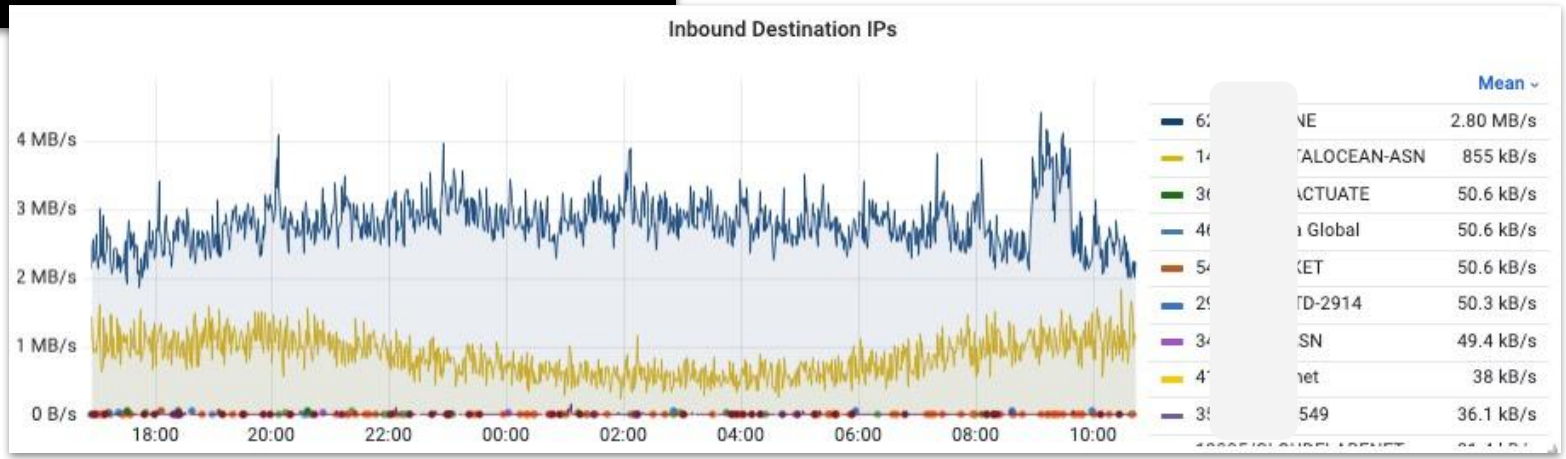
```
subnets_for_summarization:
  - 0.0.0.0/16
  - ::/64
exclude_ips_from_summarization:
  - 10.0.0.0/8
```



Summarize IPs by ASN

Higher level aggregates are sometimes all you need

```
summarize_ips_by_asn: true
exclude_asns_from_summarization:
 - 62597
```



Available Flow Metrics

Flow Metrics

flow_records_flows

flow_records_filtered

flow_(in|out)_(bytes|packets)

flow_(in|out)_(ipv4|ipv6)_(bytes|packets)

flow_(in|out)_(tcp|udp|other_l4)_(bytes|packets)

Cardinality Metrics

flow_cardinality_conversations

flow_cardinality_(src|dst)_ips_(in|out)

flow_cardinality_(src|dst)_ports_(in|out)

Top N Metrics

flow_top_(in|out)_(src|dst)_ips_(bytes|packets)

flow_top_(in|out)_(src|dst)_ports_(bytes|packets)

flow_top_(in|out)_(src|dst)_ip_ports_(bytes|packets)

flow_top_(in|out)_dscp_(bytes|packets)

flow_top_(in|out)_ecn_(bytes|packets)

flow_top_**asn**_(bytes|packets)

flow_top_**geo_loc**_(bytes|packets)

flow_top_conversations_(bytes|packets)

Small Data by the Numbers

For one of NS1's POP over a 24 hour period:

enabled metrics:

counters, cardinality, top talkers, top applications, by_bytes

flow records processed:

3.38 million flow records = ~200 MB

metric data points generated:

163k metric data points = ~250 KB

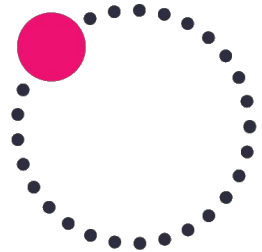
Roughly **3 orders of magnitude** difference

Use Cases for Small Data Flow Analysis

- Real time operational view and debugging of network traffic
- Thresholding and alerting using simplified time series data
- Integration with common observability data pipelines
- Cross domain dashboarding (for DevOps, for example)
- Cost savings for storage and processing
- Reliability and ease of use (versus legacy tools and pipelines)

wrapping it up

Remember This



- Orb is a network observability platform with exceptional support for flow analysis
- Orb promotes the small data paradigm with dynamic policy orchestration
- Orb integrates with modern observability stacks via OpenTelemetry
- Orb is free, open source software backed by NetBox Labs

What's Up Next

- Device discovery and configuration monitoring (SuzieQ, NAPALM)
- gNMI streaming telemetry
- Deeper NetBox integration to pair Observed State with Intended State
- Machine learning for automated insights and anomaly detection
- **What are your ideas?**

Where To Go From Here

- Learn: orb.community
- Join the community: netdev.chat
- Try Orb SaaS for free: orb.live
- Star the project: github.com/orb-community/orb
- Give us your feedback! We'd love to understand your use case

v2

thank you

questions?

netbox labs

https://orb.community/