

Empowering Measurement Users at ESnet

Dan Doyle <daldoyle@es.net>

ESnet

Internet2 TechEx 2023

Stardust

Network Measurement and Analysis for ESnet

Extensible / Open Architecture

NSF NetSAGE project derived Approach

- Integrate where we can, innovate where it makes a difference.
 - ◆ metadata and viz
- Loose coupling to avoid lock in

Authenticated access methods for many user groups.

Dashboards, Indexed APIs and “Raw”

- Grafana user editable dashboards
- Elasticsearch Query API access
- Kafka feeds
- RBAC with 2 Factor

Multi Datasource

Low and High Cardinality

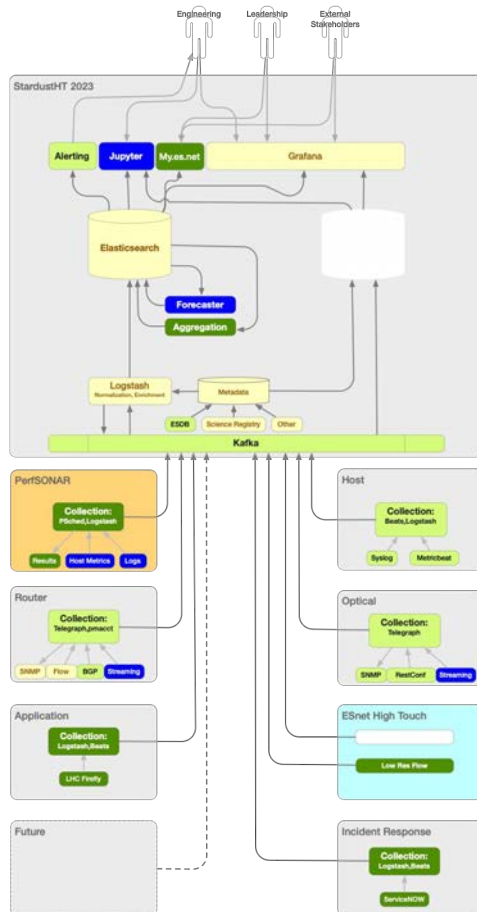
- Network Traffic Flows
- Interface Usage
- Optical Line System Performance
- perfSONAR
- High Touch measurements
- LHC Firefly measurements

Flexible aggregation

Variable time buckets and dimension reduction

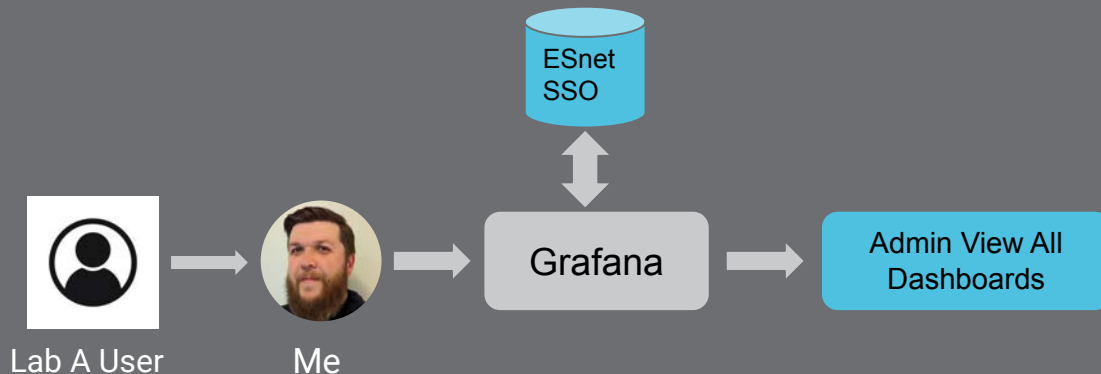
- Summarize in time
 - ◆ Hourly summary
- Summarize by dimension
 - ◆ All NERSC traffic
- Variable retention
 - ◆ hourly data for a year
 - ◆ 30 sec data for 90 days

Stardust Design Evolution

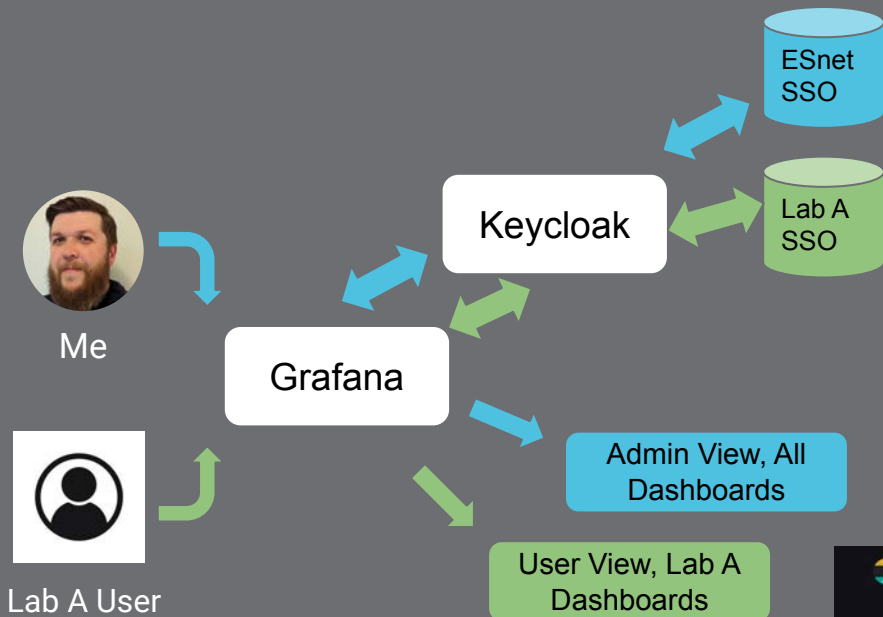


Setting the Stage

We collect tons of network data...
but we haven't have good
mechanisms to securely share
sensitive info with external users.



Adding External Users



Roles

Realm Roles

Default Roles

grafana



View all roles

Role Name	Composite	Description
grafana-admin	False	Admin permissions for Grafana Enterprise
grafana-editor	False	Editor permissions for Grafana Enterprise
grafana-viewer	False	Viewer permissions for Grafana Enterprise



Teams / Ames

Manage members and settings

Members

Settings

External group sync

External group sync

External Group ID

Ames



Data Sources / DEV Flow Ames

Type: Elasticsearch

Settings

Permissions

Insights

Permissions

Add a permission

Role

Permission

Admin

Edit

Team

Permission

Ames

Query

External Users

Dashboards
Create and manage dashboards to visualize your data

Browse Playlists Snapshots Library panels

Search for dashboards

Filter by tag Starred

- General
- Team AMES - Ames National Laboratory



Lab User

Dashboards
Create and manage dashboards to visualize your data

Browse Playlists Snapshots Library panels

Search for dashboards

Filter by tag Starred

- General
- Confab Demo
- EngageMap (internal beta)
- EngageMap - Dev
- Portal Query Notes
- Stardust BMP
- Stardust Flow
- Stardust High Touch Demo
- Stardust Internals



Admin User

Internal Observability

Who dunnit?

```
resources {
  "type": [
    "dashboard"
  ],
  "id": [
    537
  ]
}

result {
  "responseBody": [
    "{\\\"id\\\":537,\\\"message\\\":\\\"Dashboard Ed - Distribution of Bits per flow deleted\\\",\\\"title\\\":\\\"Ed - Distribution of Bits per flow\\\"}"
  ],
  "statusCode": [
    200
  ],
  "size": [
    120
  ],
  "statusType": [
    "success"
  ]
}

user {
  "authTokenId": [
    3397
  ],
  "isAnonymous": [
    false
  ],
  "orgRole": [
    "Admin"
  ],
  "name": [
    "Katrina Turner"
  ]
}
```

Visualize Events



Syslog events, Logstash



Starflakes

What if you wanted to extract data and do your own, arbitrarily complex analysis?

Working on building a Jupyter environment for researchers

- SQL interface to Elastic, lower barrier to entry
- Python, Pandas, Numpy, etc, the usual suspects for data analysis

A screenshot of a Jupyter Notebook interface. The top bar shows "File Edit View Run Kernel Tabs Settings Help". The notebook has two tabs: "Launcher" and "HelloStarFlakes.ipynb". The code cell contains a SQL query. The console output shows the results of the query, including a histogram and a table of data.

```
sql
1 SELECT
2 ingress_router,
3 ingress_router_ip,
4 egress_router_ip,
5 total_bytes
6 FROM (
7     SELECT
8         HISTOGRAM(@timestamp, INTERVAL
9         meta.router.name.keyword as ingress_router,
10        meta.router.ip as ingress_router_ip,
11        meta.bgp.next_hop as egress_router_ip,
12        sum(values.num_bits) / 8 as total_bytes
13 FROM "stardust_flow_edge*"
14 WHERE @timestamp >= '2023-08-2
15 AND meta.router.name.keyword = 'f
16 AND meta iface_in.intercloud = '
17        group by 1, 2, 3, 4

console
20 * ingress_router_ip: 1
21 * egress_router_ip: 13
22 SELECT
23 ingress_router,
24 ingress_router_ip,
25 egress_router_ip,
26 total_bytes
27 FROM (
28     SELECT
29         HISTOGRAM(
30         meta.router
31         meta.router_ip
32         meta.bgp.n
33         sum(values
34         FROM meta
35         WHERE "fti
36         AND meta.i
37         AND meta.l
38         GROUP BY 1
39         )
40 Max rows 5000
41 Total rows 2034
42 first row:
43 * ingress_router: fnal
44 * ingress_router_ip: 1
45 * egress_router_ip: 13
46 * total_bytes: 119300
47
```

Display the data in a sortable table

```
[5]: ui.display_data()
```

ingress_router	ingress_router_ip	egress_router_ip	total_bytes
fnalgcc-cr6	134.55.200.188	134.55.200.165	11,193,000
fnalgcc-cr6	134.55.200.188	134.55.200.166	133,428,000
fnalgcc-cr6	134.55.200.188	134.55.200.167	826,824,000
fnalgcc-cr6	134.55.200.188	134.55.200.168	8,911,000

Covering New Ground

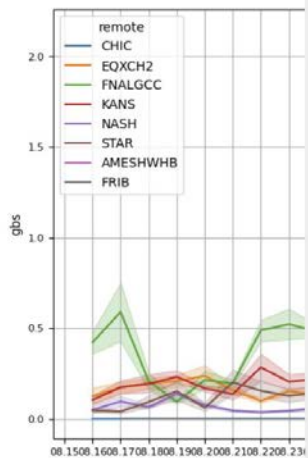
graph the t1 variable against

```
[19]: import matplotlib as mpl
import matplotlib.pyplot as plt
import seaborn as sns
import matplotlib.dates as mdates
import warnings
warnings.filterwarnings('ignore')

fig = plt.figure(figsize=(12,6))
axes = fig.add_subplot()

lp1 = sns.lineplot(x="t1", y="gbs", hue="
lp1.tick_params(labelsize=8)

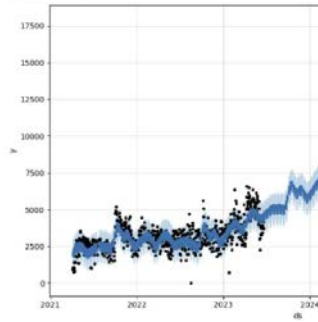
axes.grid(True)
axes.set_xticklabels(axes.get_xticklabels)
axes.xaxis.set_major_formatter(mdates.Date
axes.xaxis.set_major_locator(mdates.DayLo
```



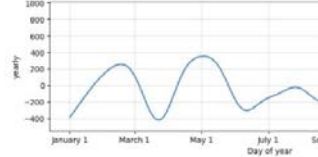
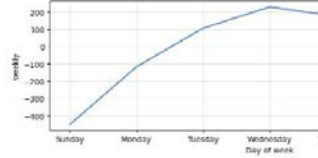
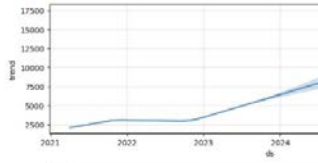
```
model = Prophet()
t = model.t[1:107] # prophet.forecaster._Prophet

# if we have daily data (in Frequency is 365)
# predict 3 years
YEARS_OUT = 3
future = model.make_future_data(frequency=365 * YEARS_OUT, freq='D')
forecast = model.predict(future)

fig, forecast = model.plot(forecast)
fig.show()
```



```
fig = model.plot_components(forecast)
fig.show()
```



Examples of using the Prophet library in python for forecasting and capacity planning of the network

Visualizing the Network

Current maps maintained by software developers, not network engineers.

Lots of hand tuning, games of telephone, brittle, etc.

Lack of consistent, structured “source of truth” for maps

The screenshot displays the TerraNova web application interface. The main map area shows a network of blue lines and nodes overlaid on a map of the United States. The interface includes a left sidebar with navigation options: Tools (Create New Dataset, Create New Map, Node SVG Builder), Libraries (Datasets: Core Geographical, Core Logical, Sites Geographical, All Datasets...; Maps: Portal Geographical, Portal Logical, Stardust Geographical, All Maps...; Node SVGs: ENet Logo, Star, Triangle, All Node SVGs...), and Documentation. The top right corner shows the map URL: `https://api.layer-source.example.net/api/v1/`. The right sidebar contains map configuration options: Map Background (Map Tiles), Map Background TileSet (ArcGIS Default Set), Map Political Boundaries (No Political Boundaries), Map Initial View Strategy (Static Center, No zoom on resize), and Supplemental Layers (Weather). The bottom section, titled "Layer 1 Portal Geographic - Core", provides detailed styling options for the network layer, including Dataset Options (Import Dataset: ENet Core, Dataset View: Geographic, Dataset Version: Dynamic, Layer Name: Override Layer Name), Node Style (Shape: Square, Color: Choose Color..., Highlight Color: Choose Color..., Size: 5), and Edge Style (Color: Choose Color..., Width: 5, Offset: 5).

TerraNova

Tools

- Create New Dataset
- Create New Map
- Node SVG Builder

Libraries

Datasets

- Core Geographical
- Core Logical
- Sites Geographical
- All Datasets...

Maps

- Portal Geographical
- Portal Logical
- Stardust Geographical
- All Maps...


Node SVGs

- ESnet Logo
- Star
- Triangle
- All Node SVGs...

Documentation

Settings

ESnet Core



Preview Type

Edge Graph

Realtime Layer URL:

<https://api.layer-source.example.net/api/v1/>

Snapshot Layer URL:

Query

Circuits where...

description is like Search

circuit state is like Search

57 Results

TerraNova

Tools

- Create New Dataset
- Create New Map
- Node SVG Builder

Libraries

Datasets

- Core Geographical
- Core Logical
- Sites Geographical
- All Datasets...

Maps

- Portal Geographical
- Portal Logical
- Stardust Geographical
- All Maps...


Node SVGs

- ESnet Logo
- Star
- Triangle
- All Node SVGs...

Documentation

Settings

ESnet Core



Preview Type

Geographic

Realtime Layer URL:

<https://api.layer-source.example.net/api/v1/>

Snapshot Layer URL:

<https://api.layer-source.example.net/api/v1/>

Query

Circuits where...

description is like Search interfaces... 112 results

circuit state is like Search interfaces... 57 results

57 Results

Moving Beyond ESnet

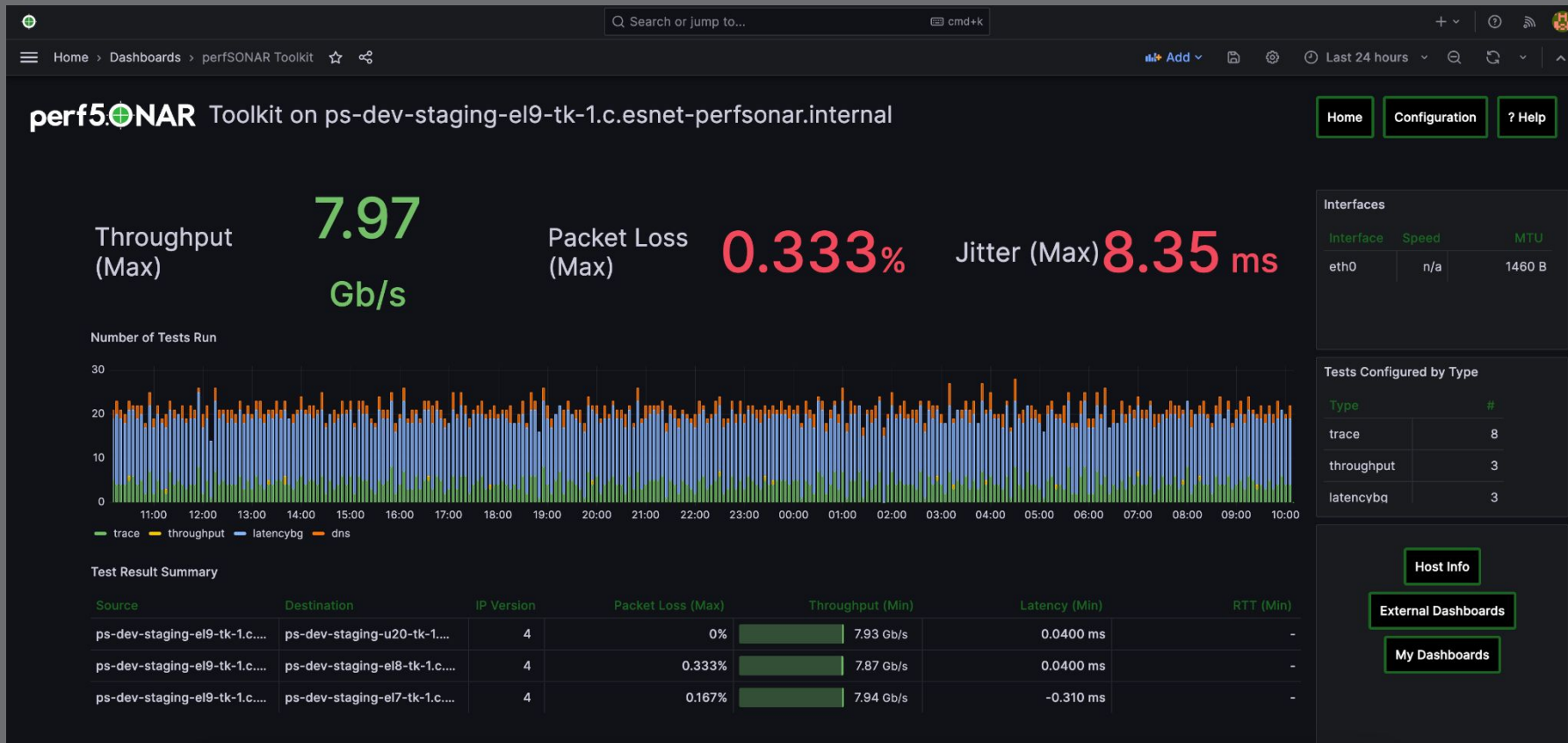
StellaNOVA

- A consortium for the Advancement of Network Observation, Visualization, and Analysis.
- Develop and socialize technical capabilities in the measurement space for R&E community
- Focus on developing tools, tactics, and techniques but will not offer measurement service

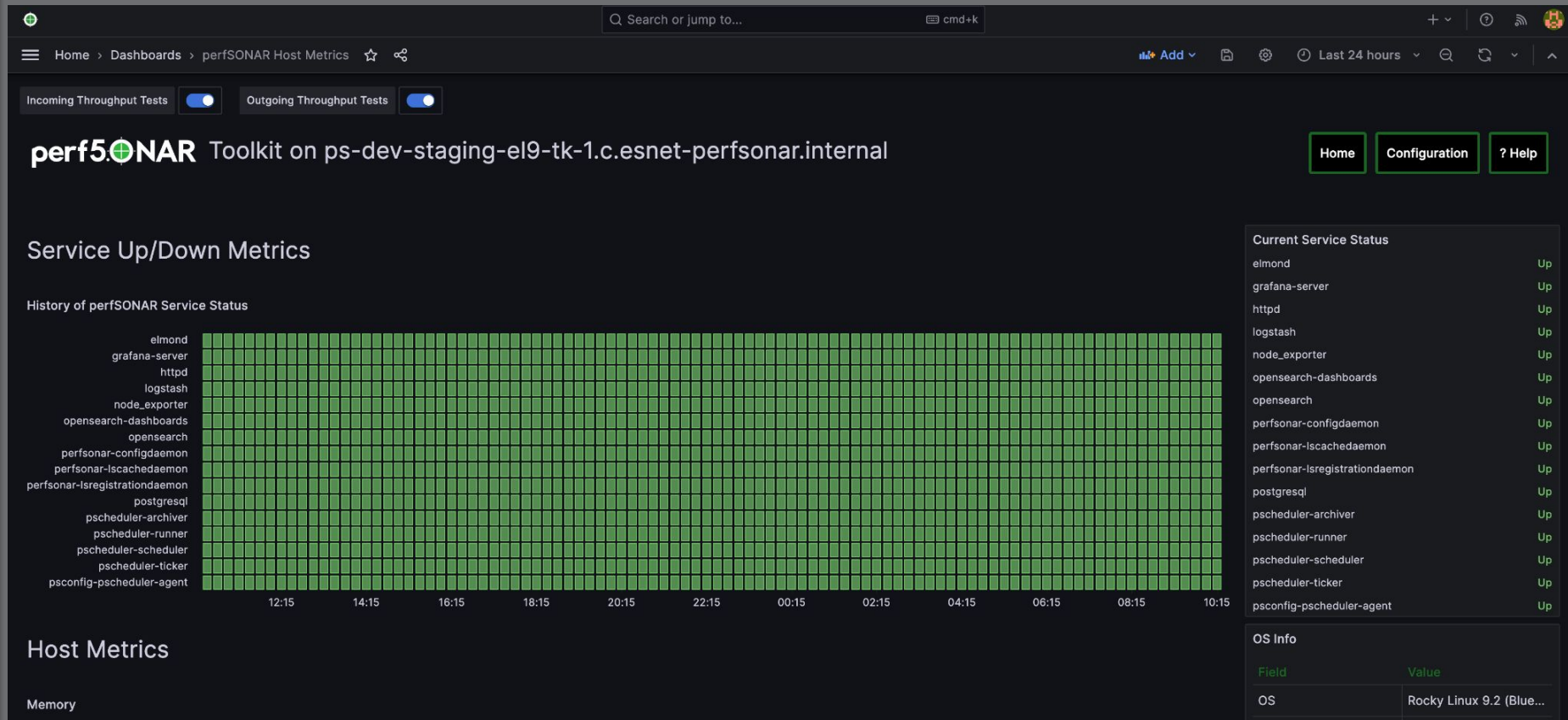
Open Sourcing

- Data ingest pipeline and index configurations made publicly example as reference - <https://github.com/esnet/stardust-snmp-pipeline>

perfSONAR flavored Stardust



perfSONAR flavored Stardust



Thank You

Questions, comments, tomatoes, etc all welcome.

Dan Doyle

daldoyle@es.net