# BMP in the SURF network

## Observing SURFinternet peerings using OpenBMP

Internet Measurements workshop 5/4/23, Joachim Opdenakker

# About me

- Joachim Opdenakker
- Network Engineer @ SURF since August 2022
  - Connectivity / peering
  - Service layer
  - Observability
  - Automation

# Agenda

- The SURF network
- How do we monitor
  - The past
  - The present
    - Demo of OpenBMP
    - Shortcomings
    - Use cases
  - The future

# SURF network

# SURF network

- Juniper based backbone
  - 1 x Juniper MX10008
  - 2 x Juniper MX2008
  - 397 x Juniper MX204
  - 22 x Juniper MX240
  - 28 x Juniper MX480
  - 8 x Juniper MX960

“SURF is one of the best connected networks in The Netherlands”

Rudolf Van Der Berg

# Connectivity
## Peering

- Unique AS numbers: 222
- Total peering sessions: 1158 (ipv4 and ipv6 counted separate)
- Internet Exchanges: 6 (540 Gbps capacity)
  - Capacity-wise we can run SURFinternet solely on IX’
- Router servers: 12 (each IX redundant)
- 3 Transit providers

# How do we monitor the SURF network?

# The past

# SNMP

# Interface stats with SNMP

- TX and RX counters
- Up/down state
- Error counters

# BGP monitoring with SNMP

- Established state or not (flap)
- BGP update counter
- Generic info like:
  - Local/remote AS
  - Local/remote IP
  - …

# The present

SURF

# BMP

BGP monitoring protocol

# BMP
## Recap

Thanks Luuk ;)

# BMP

## Open source implementations

- GoBMP https://github.com/sbezverk/gobmp
- Fernglas https://github.com/wobcom/fernglas
- OpenBMP https://github.com/OpenBMP
- Rotonda https://github.com/NLnetLabs/rotonda-store

# BMP

## Open source implementations: comparison

| Implementation | GoBMP | Fernglas | OpenBMP | Rotonda |
|---|---|---|---|---|
| Server/parser | Go | Rust | C++ | Rust |
| Expoter | Kafka<br>Stdout<br>File | In-memory | Kafka | Rotoro protocol |
| API | Kafka consumer* | Rest API | Kafka consumer to PostgreSQL | Rotonda-store/API/ CLI |
| Visualiser | ? | Webinterface | Grafana | ? |

*not included

SURF

# BMP
## OpenBMP

- At the time most interesting for SURF
  - Instant visualisation
- Kafka relatively robust to build on
  - Custom scripts to generate slack messages

# OpenBMP in practice

# OpenBMP
## Software

- Containers:
  - Server/parser
  - Kafka
  - Zookeeper
  - Java Kafka consumer (more on that later)
  - PostgreSQL
  - Cron-jobs (PostgreSQL scripts)

# OpenBMP
## Hardware

- One machine
  - 2 Cores
  - 4GB ram
  - 20GB Storage

I should not have put this on the slides.....

# OpenBMP
## Hardware

- Still one machine :)
  - 8 Cores
  - 16GB ram
  - 1TB Storage

# OpenBMP
## Kafka

- Multiple topics:
  - Collector
  - Router
  - Peer
  - ls_link
  - unicast_prefix
  - base_attribute

# OpenBMP

## PostgreSQL (Java Kafka Consumer)

- Kafka messages stored
- Cron scripts generate enriched tables/views
- Grafana displays nice graphs from PostgreSQL data

# Demo of the softwarestack

# Some shortcomings

# OpenBMP “shortcomings”

- Kafka messages are in TSV
  - JSON would be preferred for further parsing
- Logging of the different components
- Included PostgreSQL config is not tuned for our network size

# Some use cases

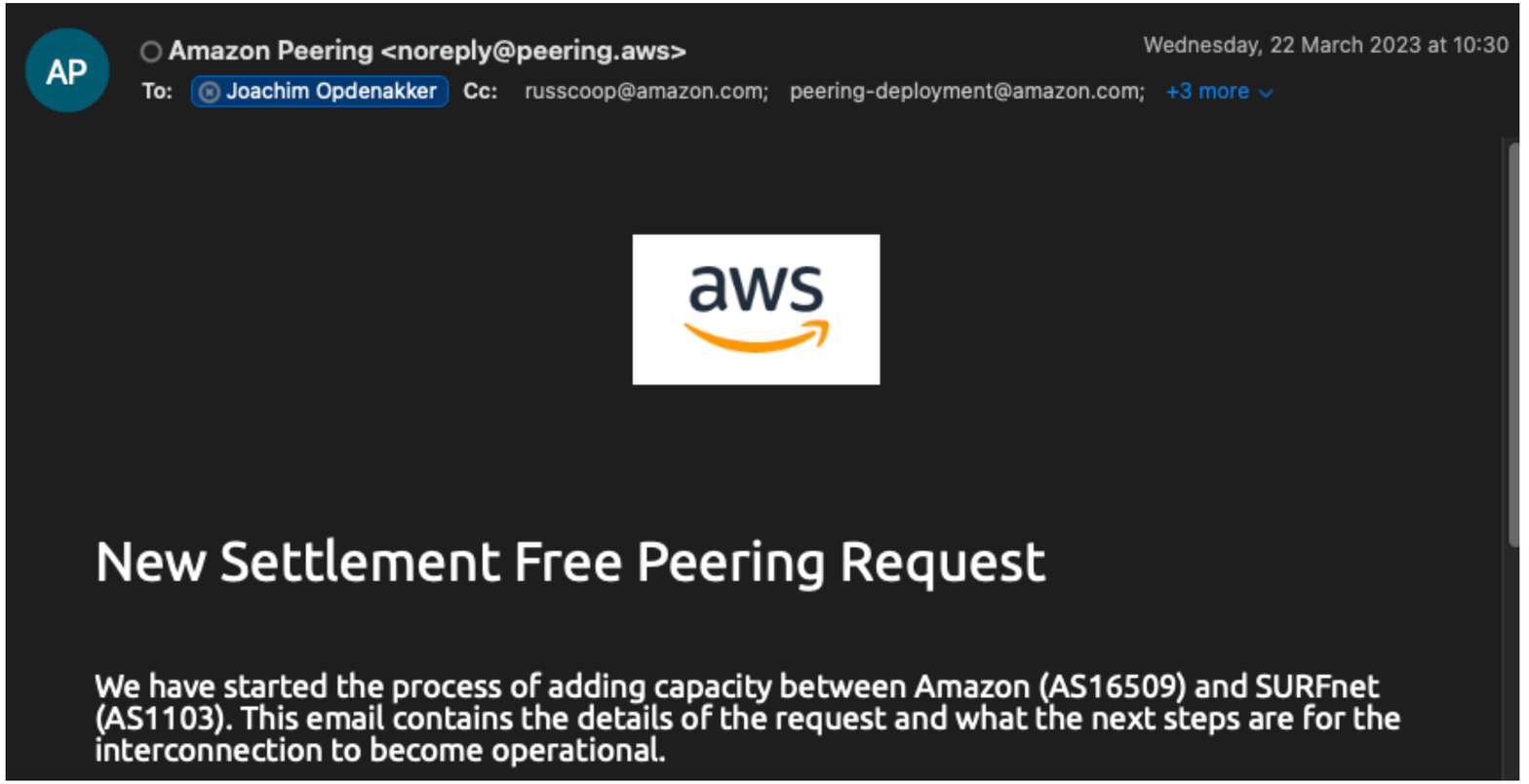
SURF

# Amazon incident

- Single 100G PNI
- Amazon withdraws all prefixes
- Traffic moves partly to IX, partly to transit

# Amazon incident

## What is the result?



# The future

# BMP
## Next Steps

- Deeper look at the Rotonda implementation
- Building a simple looking glass
- API integration with our automation stack
  - Enriched alarming
- Integration with streaming telemetry
  - Alarm correlation

SURF

# BMP
## Next Steps: details

- Enriched alarming, with the use of our Orchestrator
  - Relate a PeerUp/PeerDown notification to customer/internet peer
  - Relate PeerUp/PeerDown to interface state
  - Relate PeerUp/PeerDown to maintenance announcements
  - Relate Adj-RIB-in to traffic flow on interface
    - If routes are withdrawn, relate this to drop in traffic

# BMP

## Next Steps: example: Microsoft Peering

- Dual 100G PNI
- MS withdraws prefixes
- Route policy on our side wrong
- Traffic moves towards 20G LINX connection
- Conclusion:
  - Prevent further impact by correlating alarms from interfaces and BMP

# End

Questions?

Thank you!