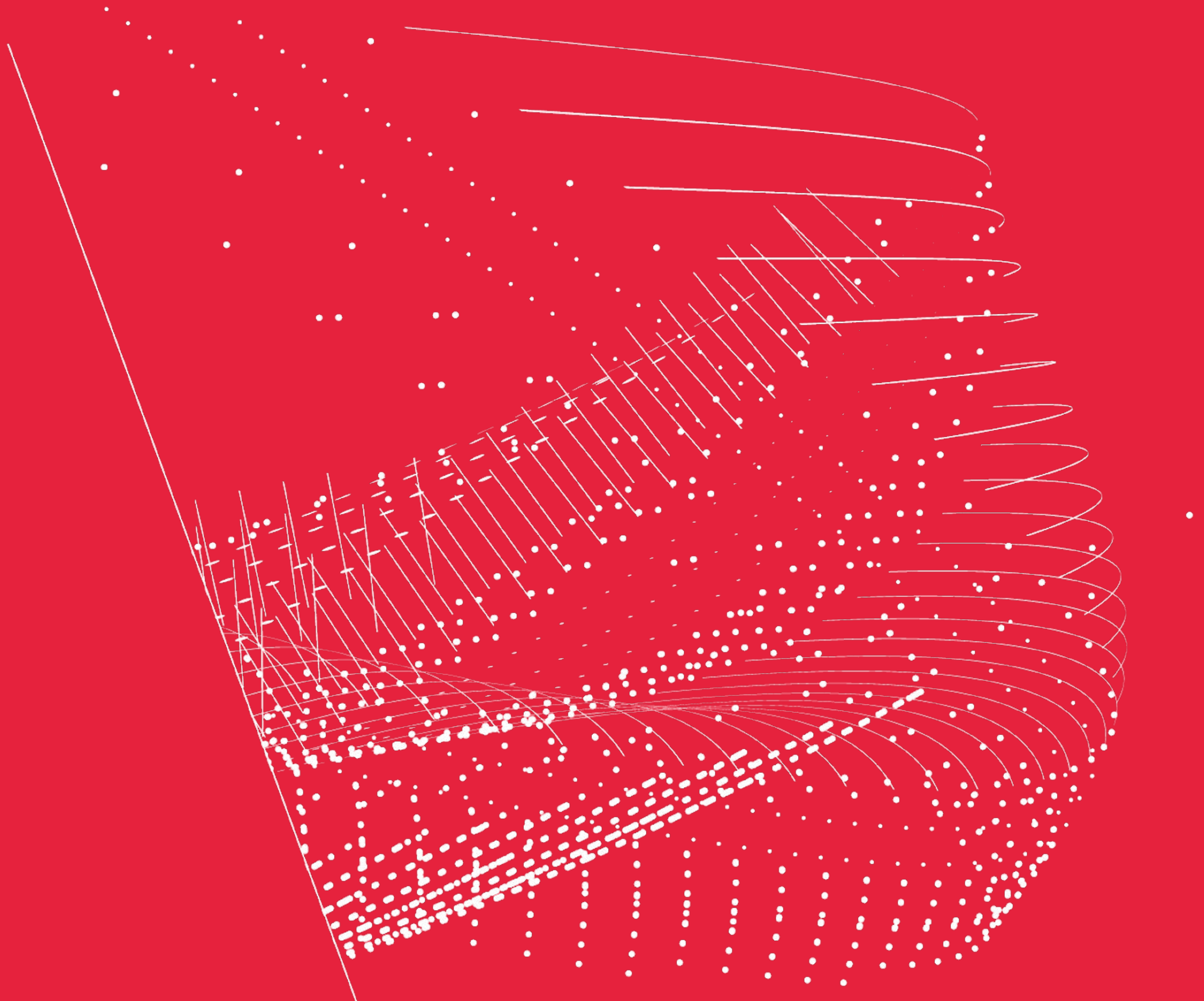
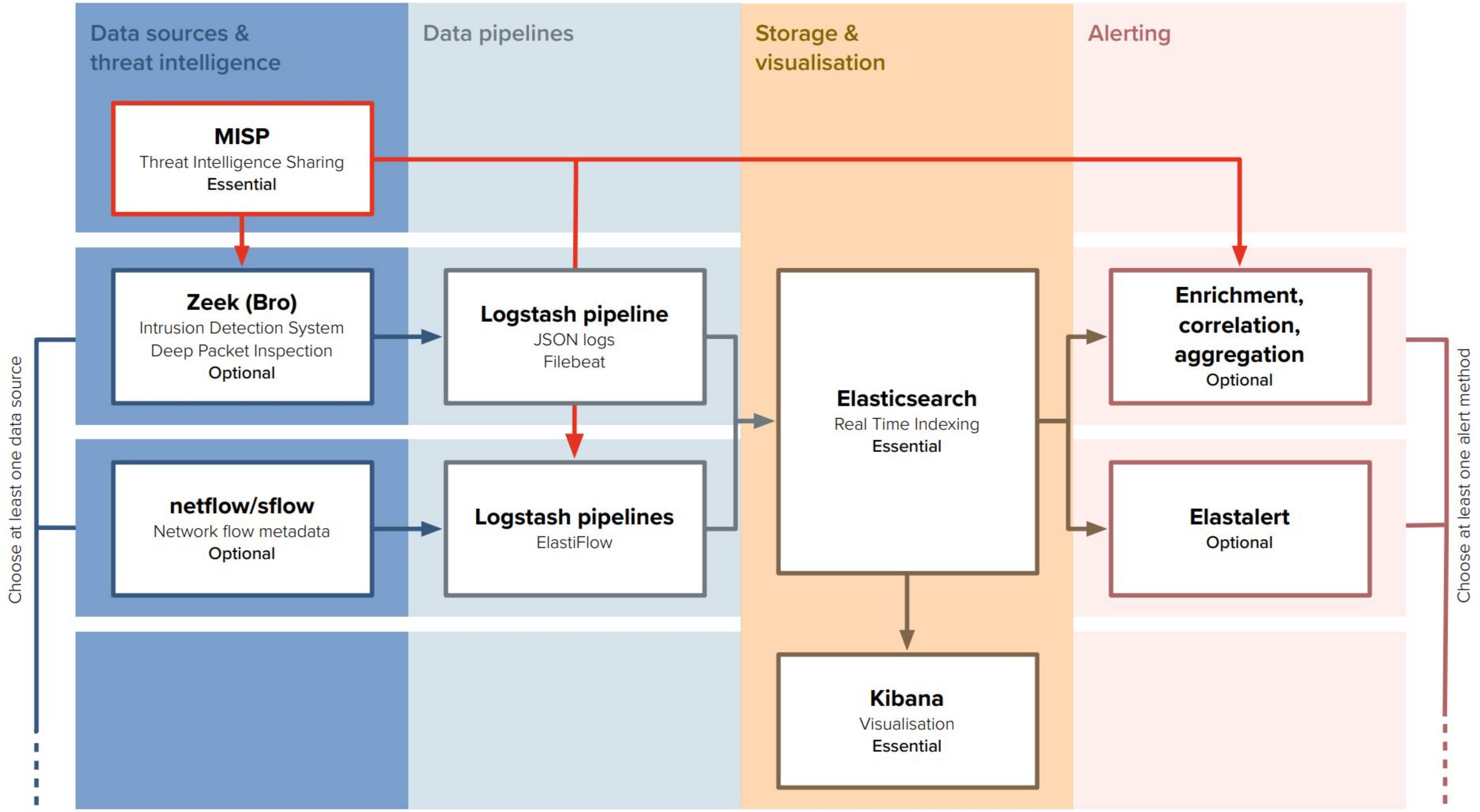


Monitoring 100Gbit in 2U*

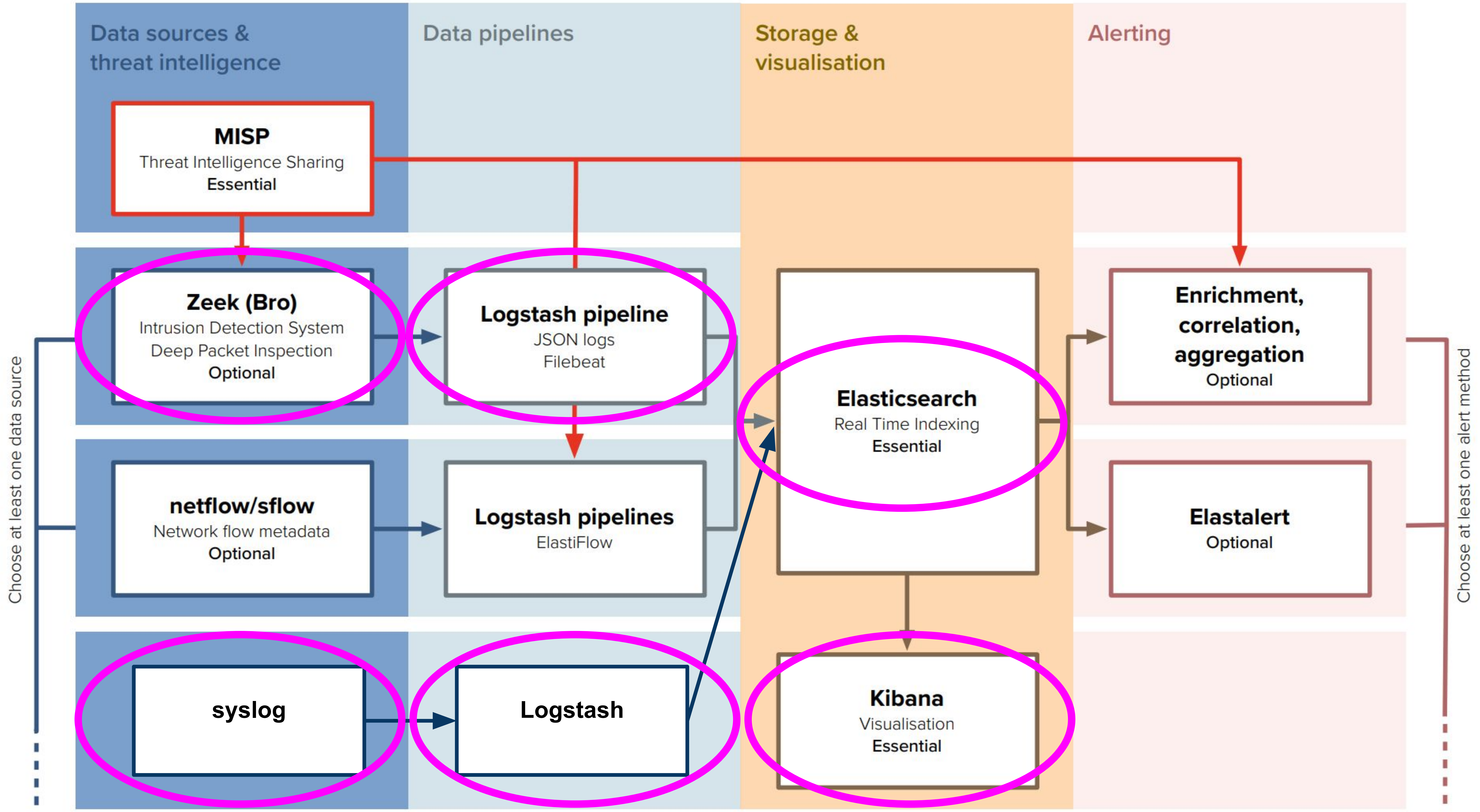
Jouke Roorda
PDP Group
Nikhef

Prelude: WLCG SOC WG Reference Model



D Crooks, et al. DOI 10.22323/1.351.0010

WLCG SOC WG Reference Model, edited



D Crooks, et al. DOI 10.22323/1.351.0010

Things we tried

IBM POWER8 S822L

Strange problems

More threads ➜ more NIC drops

Less threads ➜ more Zeek drops

One 100GE NIC?

Many 10GE NICs?

IBM POWER8 S822L w/ switch-based fanout

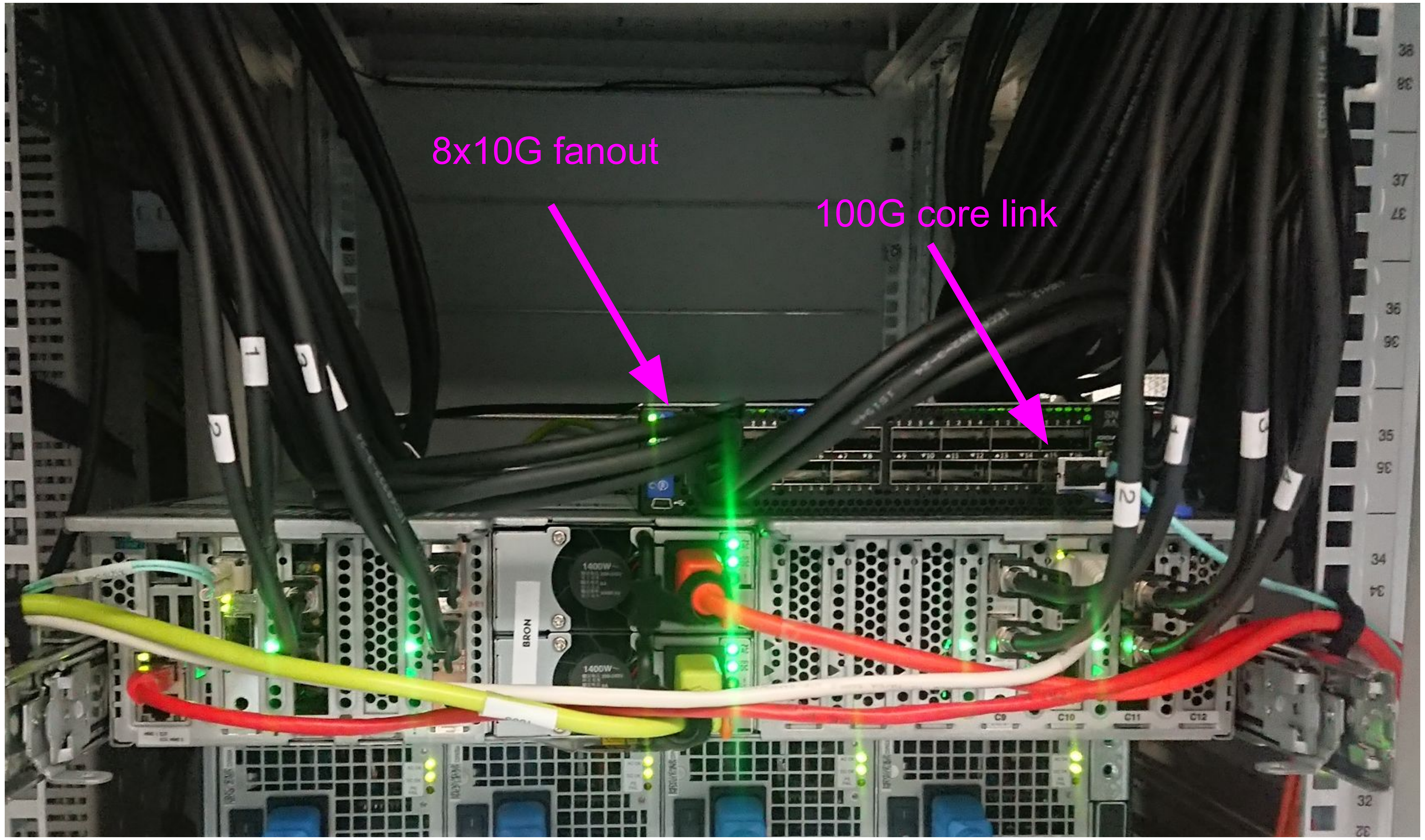
Flow based fanout

size(flow) > size(link)

Switch drops

All NICs in different NUMAs

Still dropping packets



Let’s try something else

PoC: Lenovo SR655 w/ Connect-X 5 EN

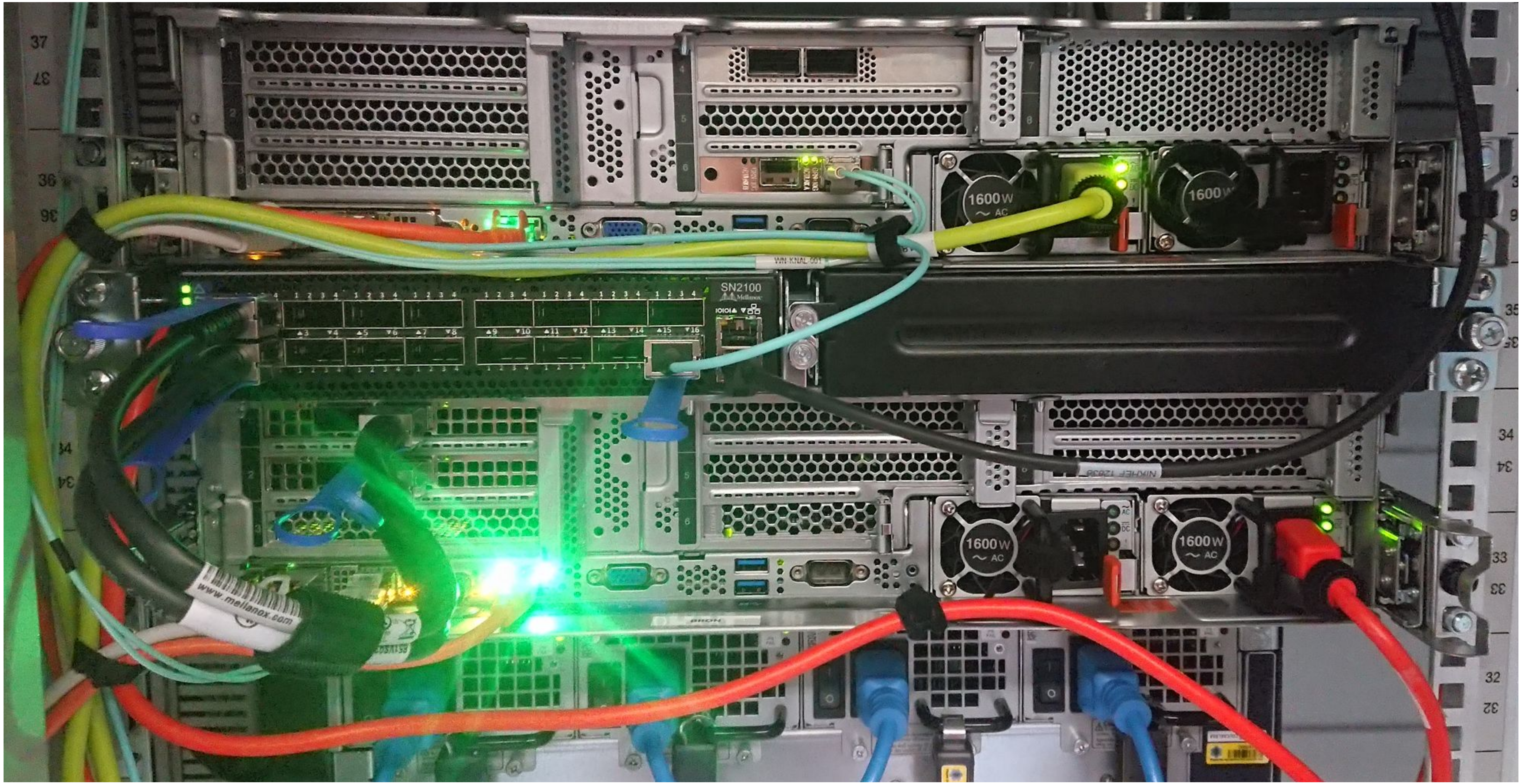
AMD EPYC 7702P 64-Core

SMT disabled

100Gb mirror from core

CentOS 8 + Zeek with 60 workers

Migration...



Producion: Lenovo SR655 w/ 2* Connect-X 6 Dx

AMD EPYC 7702P 64-Core

- SMT disabled

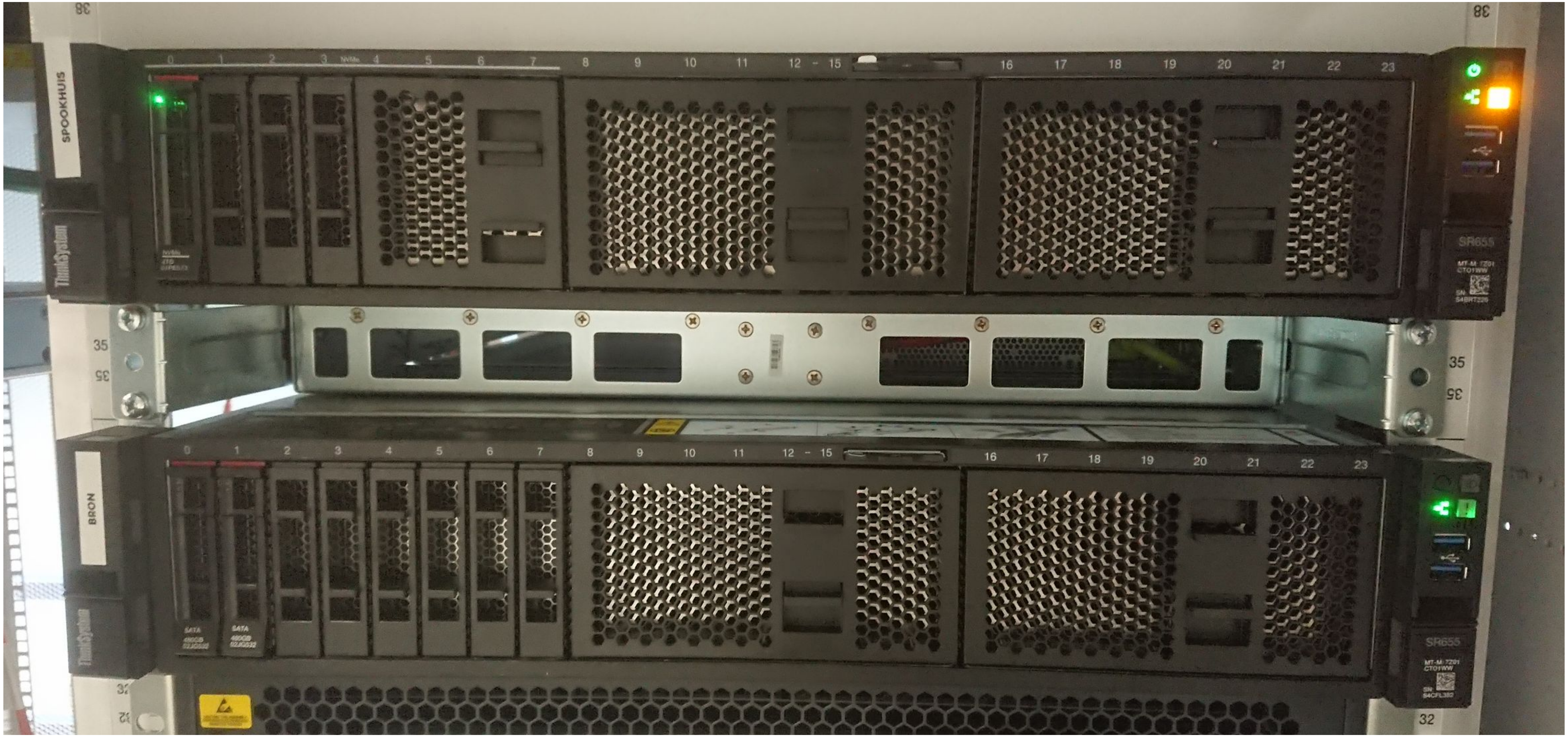
Zeek with 2*30 workers

Debian 10

100Gb mirror from core to SN2100

2*100G from SN2100 to NICs

Bonus picture



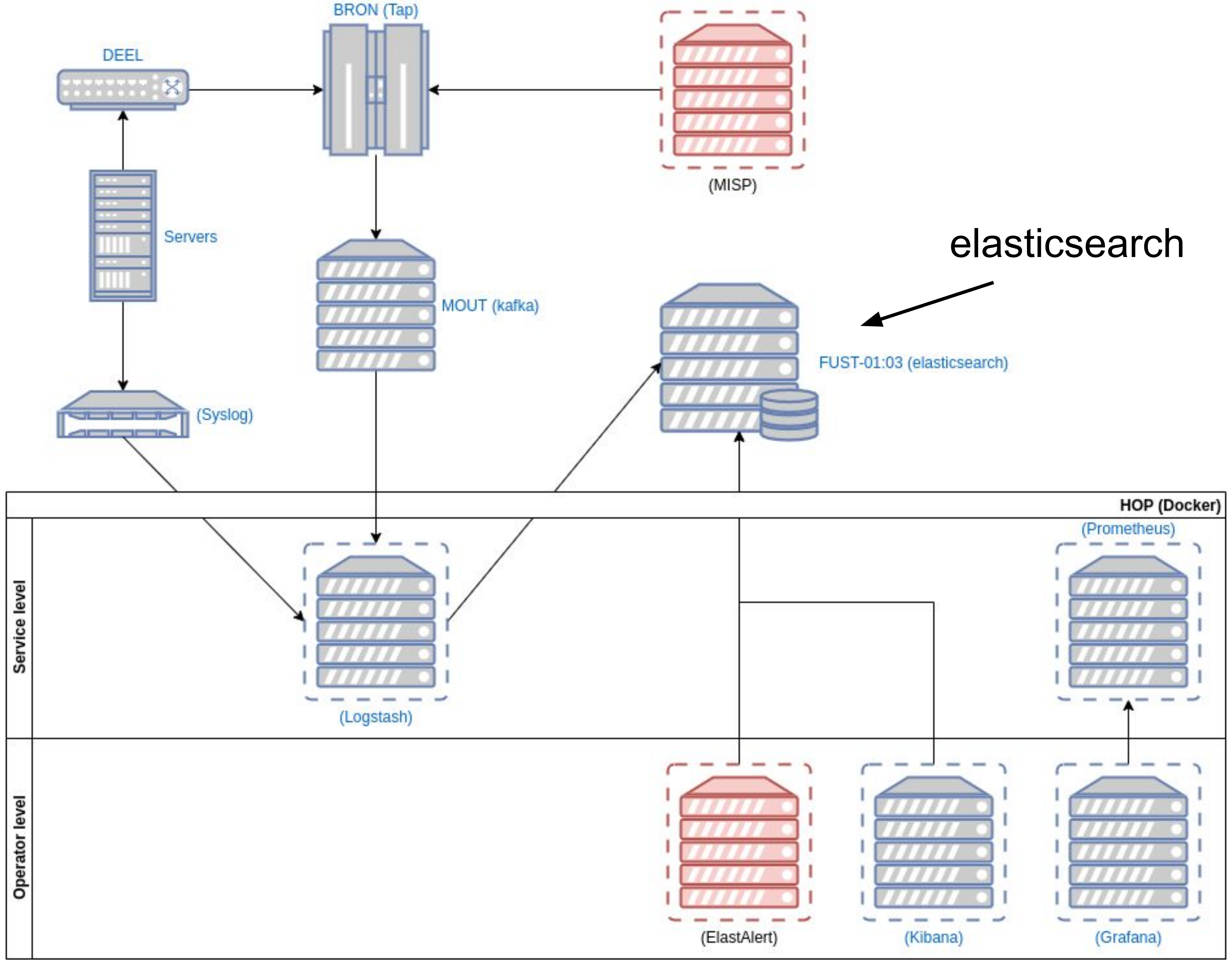
Reality is usually more nuanced

10Gb dedicated private network
5 (soon to be 7) ex-grid machines

Docker host (services)

elasticsearch

Kafka host



Core

SR655

elasticsearch

Attack traffic

TAP data rate
100 Gbps
95 Gbps
90 Gbps
85 Gbps
80 Gbps
75 Gbps
21:08:00 21:08:10 21:08:20 21:08:30 21:08:40 21:08:50 21:09:00 21:09:10 21:09:20 21:09:30 21:09:40 21:09:50 21:10:00 21:10:10 21:10:20 21:10:30 21:10:40 21:10:50 21:11:00 21:11:10 21:11:20 21:11:30 21:11:40 21:11:50 21:12:00 21:12:10 21:12:20 21:12:30 21:12:40 21:12:50 21:13:00 21:13:10 21:13:20 21:13:30 21:13:40 21:13:50 21:14:00
enp129s0f0
TAP data rate
94.4 Gbps
TAP packet rate
12 Mpps
11 Mpps
10 Mpps
9 Mpps
8 Mpps
enp129s0f0
TAP pack rate
11.0 Mpps
TAP drops
25 kpps
20 kpps
15 kpps
10 kpps
5 kpps
0 pps
enp129s0f0
TAP drops
2.835 kpps
TAP processorgebruik
40.0
37.5
35.0
32.5
30.0
mode!="idle"
CPU TAP
38
/64
TAP geheugengebruik
89.1 GiB
89.0 GiB
88.9 GiB
88.8 GiB
88.8 GiB
RAM
RAM TAP
88.9 GiB
/512GB

Real traffic

TAP data rate
80 Gbps
60 Gbps
40 Gbps
20 Gbps
0 bps
16:30
16:35
16:40
16:45
16:50
16:55
17:00
17:05
17:10
17:15
17:20
17:25
17:30
17:35
17:40
17:45
17:50
17:55
18:00
18:05
18:10
18:15
18:20
18:25
— enp129s0f0
TAP data rate
40.3 Gbps
TAP packet rate
8 Mpps
6 Mpps
4 Mpps
2 Mpps
0 pps
— enp129s0f0
TAP pack rate
3.524 Mpps
TAP drops
2.0 kpps
1.5 kpps
1.0 kpps
500 pps
0 pps
— enp129s0f0
TAP drops
124 pps
TAP processorgebruik
15
10
5
0
— mode!="idle"
CPU TAP
8.5
/64
TAP geheugengebruik
88.3 GiB
88.1 GiB
87.9 GiB
87.7 GiB
87.5 GiB
— RAM
RAM TAP
87.8 GiB
/512GB

More real traffic

TAP data rate
100 Gbps
75 Gbps
50 Gbps
25 Gbps
0 bps
10/11 16:00 10/11 18:00 10/11 20:00 10/11 22:00 10/12 00:00 10/12 02:00 10/12 04:00 10/12 06:00 10/12 08:00 10/12 10:00 10/12 12:00 10/12 14:00 10/12 16:00 10/12 18:00 10/12 20:00 10/12 22:00 10/13 00:00 10/13 02:00 10/13 04:00 10/13 06:00 10/13 08:00 10/13 10:00 10/13 12:00 10/13 14:00
enp129s0f0
TAP data rate
Last 5 seconds
64.8 Gbps
TAP packet rate
15 Mpps
10 Mpps
5 Mpps
0 pps
enp129s0f0
TAP pack rate
Last 5 seconds
5.905 Mpps
TAP drops
80 kpps
60 kpps
40 kpps
20 kpps
0 pps
enp129s0f0
TAP drops
Last 5 seconds
5.2 pps
TAP processorgebruik
30
20
10
0
mode!="idle"
CPU TAP
Last 5 seconds
15
/64
TAP geheugengebruik
93 GiB
91 GiB
89 GiB
88 GiB
RAM
RAM TAP
Last 5 seconds
88.3 GiB
/512GB

Not yet done: drops

System still occasionally dropping

- 5pps LLDP?
- Reason for rest unclear

rx_discards_phy keeps rising

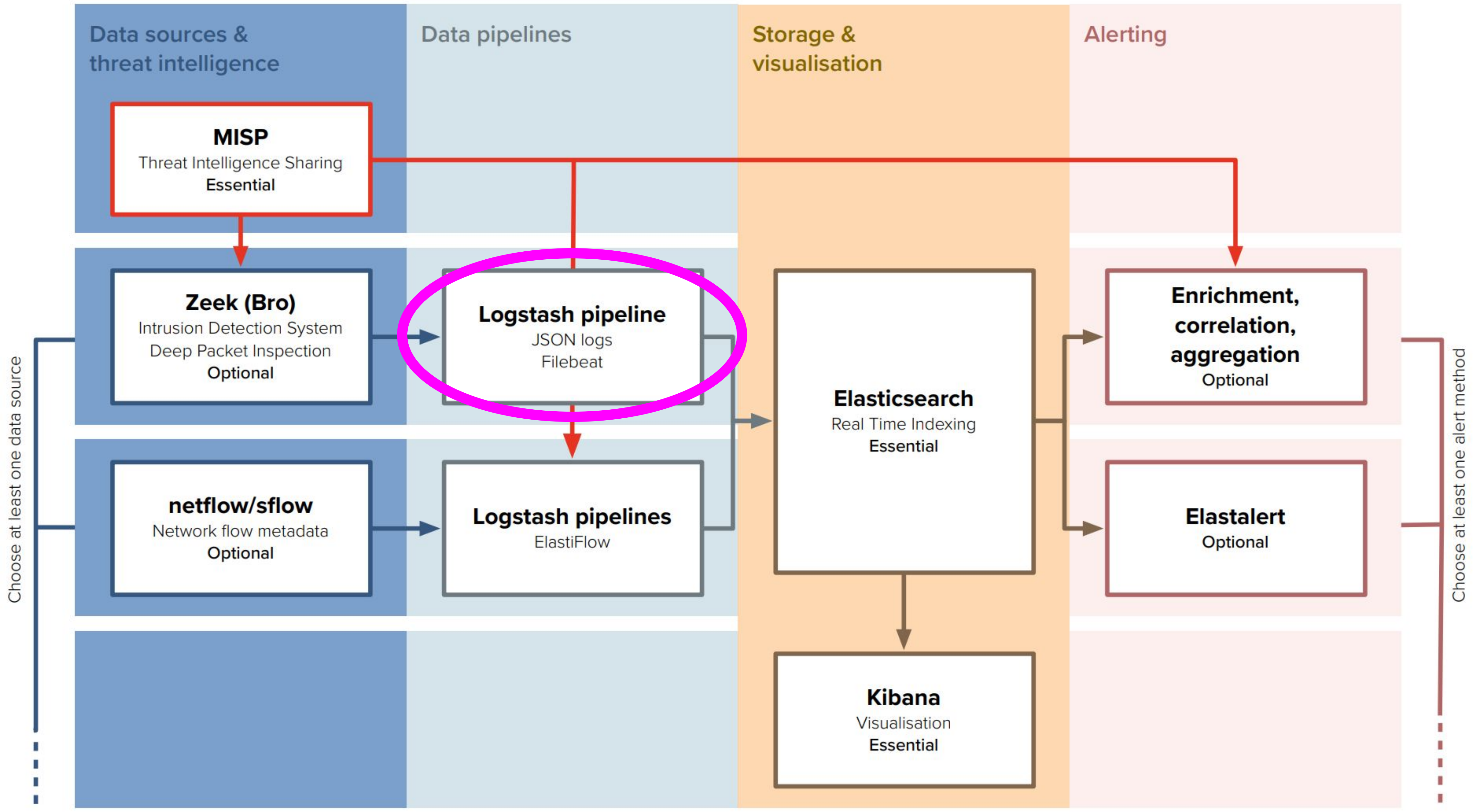
Low Coalescing gets rid of drops

- But also of traffic :)

Even with low traffic

TAP data rate
100 Gbps
75 Gbps
50 Gbps
25 Gbps
0 bps
10/11 16:00
10/11 18:00
10/11 20:00
10/11 22:00
10/12 00:00
10/12 02:00
10/12 04:00
10/12 06:00
10/12 08:00
10/12 10:00
10/12 12:00
10/12 14:00
10/12 16:00
10/12 18:00
10/12 20:00
10/12 22:00
10/13 00:00
10/13 02:00
10/13 04:00
10/13 06:00
10/13 08:00
10/13 10:00
10/13 12:00
10/13 14:00
enp129s0f0
TAP data rate
Last 5 seconds
64.8 Gbps
TAP packet rate
15 Mpps
10 Mpps
5 Mpps
0 pps
enp129s0f0
TAP pack rate
Last 5 seconds
5.905 Mpps
TAP drops
80 kpps
60 kpps
40 kpps
20 kpps
0 pps
enp129s0f0
TAP drops
Last 5 seconds
5.2 pps
TAP processorgebruik
30
20
10
0
mode!="idle"
CPU TAP
Last 5 seconds
15
/64
TAP geheugengebruik
93 GiB
91 GiB
89 GiB
88 GiB
RAM
RAM TAP
Last 5 seconds
88.3 GiB
/512GB

What we did not do: log files



Data sources & threat intelligence

Data pipelines

Storage & visualisation

Alerting

MISP
Threat Intelligence Sharing
Essential

Zeek (Bro)
Intrusion Detection System
Deep Packet Inspection
Optional

netflow/sflow
Network flow metadata
Optional

Logstash pipeline
JSON logs
Filebeat

Logstash pipelines
ElastiFlow

Elasticsearch
Real Time Indexing
Essential

Kibana
Visualisation
Essential

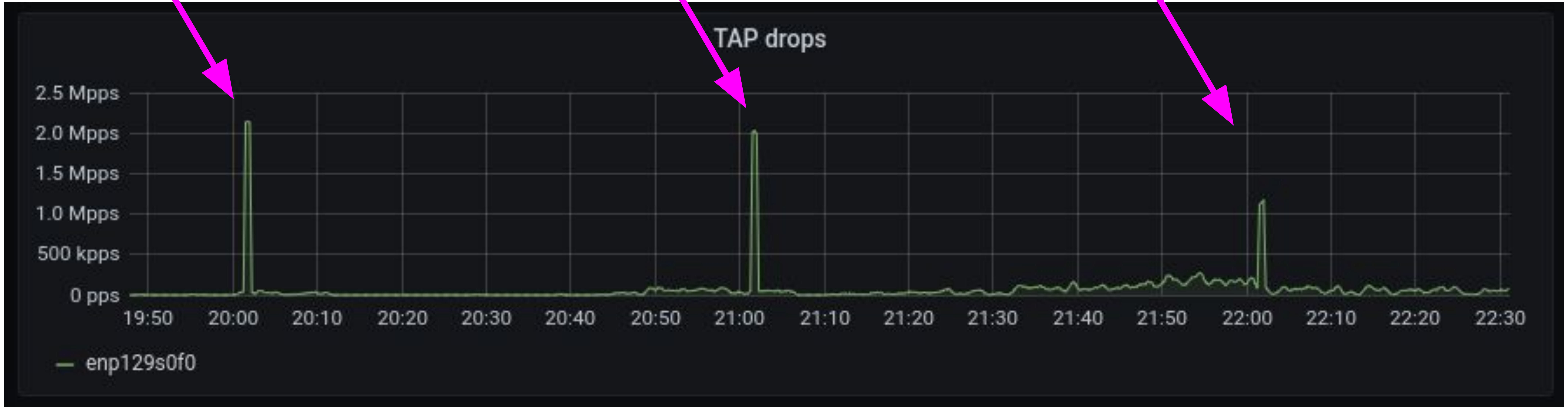
Enrichment, correlation, aggregation
Optional

Elastalert
Optional

Choose at least one data source

Choose at least one alert method

Writing Zeek logs to file
Significant drops during log rotate
Filebeat not even used yet



Alternative solution: Kafka

Part of Apache Metron

- Continuous streaming

Zeek ➜ Kafka ➜ Logstash ⇶ Elasticsearch

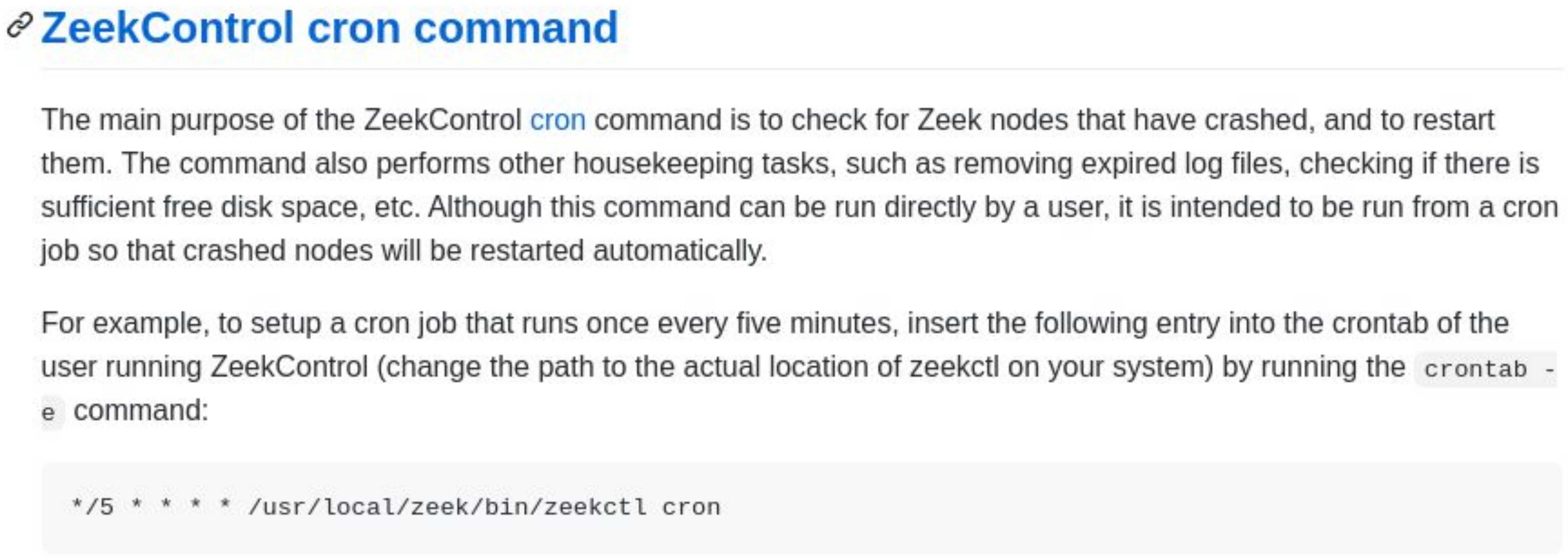
Added bonuses:

- Communication is JSON by default, i.e. no parsing needed
- Kafka buffers when Logstash goes down

What we did not do: zeekctl cron

Might be a side-effect of 1-box Zeek
Reason for impact is unclear

Alternative: to be determined
Systemd is used for now

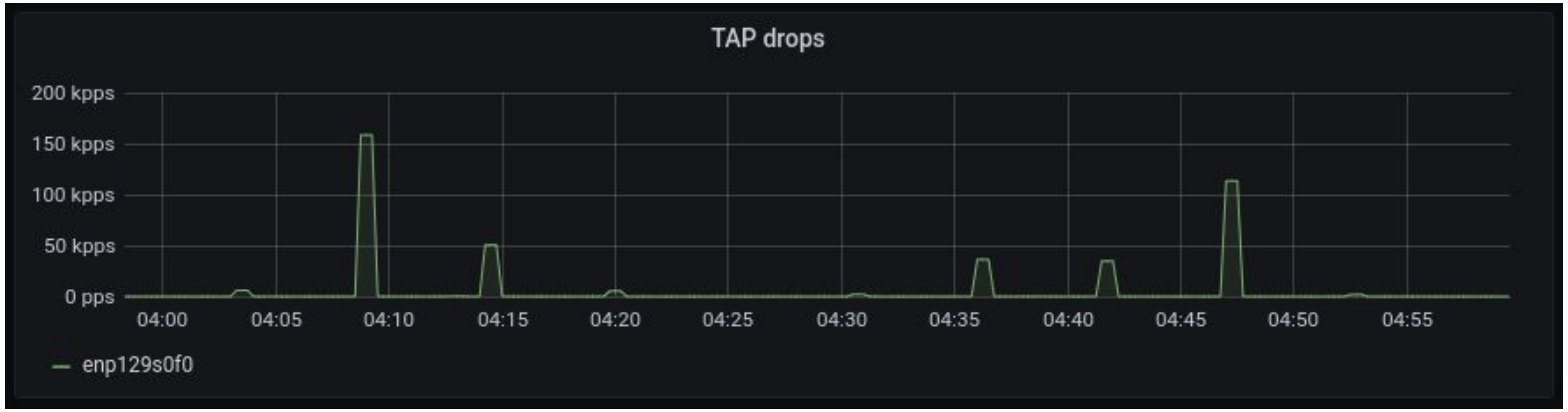


ZeekControl cron command

The main purpose of the ZeekControl cron command is to check for Zeek nodes that have crashed, and to restart them. The command also performs other housekeeping tasks, such as removing expired log files, checking if there is sufficient free disk space, etc. Although this command can be run directly by a user, it is intended to be run from a cron job so that crashed nodes will be restarted automatically.

For example, to setup a cron job that runs once every five minutes, insert the following entry into the crontab of the user running ZeekControl (change the path to the actual location of zeekctl on your system) by running the `crontab -e` command:

```
*/5 * * * * /usr/local/zeek/bin/zeekctl cron
```



Observations running our 100G SOC

ES data hungry: 2 more nodes

Flow size is more important than link size

Monitoring Zeek availability is not trivial

Next steps & future work

Additional node to monitor storage
- Zeek clustering?
- Separate nodes?

See how newer POWER performs

Core count vs clock performance

High availability

Extra ES nodes
- Same as current
- Already reserved

Upgrades?
- SSDs for ES nodes?

In conclusion

100G monitoring in 2U: you can, and this is how