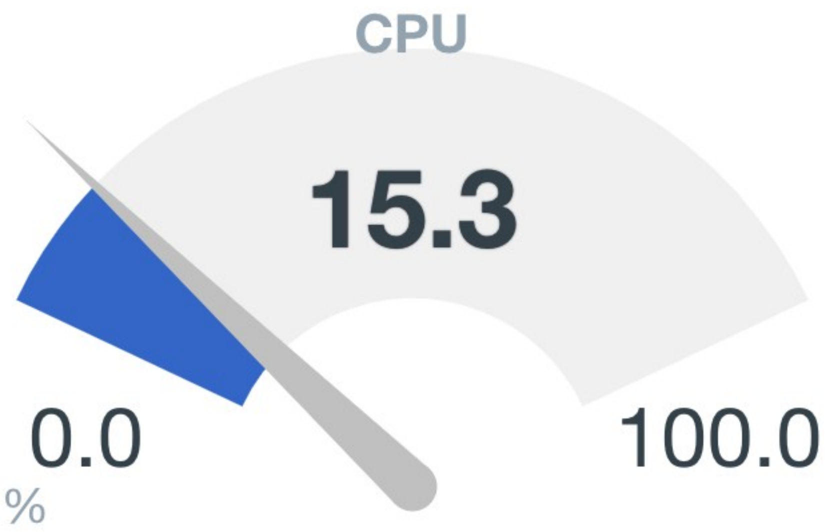
Network Monitoring and Management Tools

- Metrics collection
- Active measurement
- Netflow
- Logs
- Configuration management
- Alerting

Metrics

Metrics



Something that you *measure*

Metrics are always *numeric values*

Types of metric:

- Gauges *(e.g. available disk space, temperature)*
- Counters *(e.g. bytes received, total time spent working)*
    - *counters only ever INCREASE*

Data collection: SNMP

- Simple Network Management Protocol (v1, v2, v3)
- Widely implemented in network devices
- Also for servers, if you install an SNMP agent
- Counters: e.g. interface traffic, interface errors, …
  - Count of number of bytes sent/received since device booted
  - Monitoring software converts into rate (bits per second)
- Gauges: uptime, CPU/RAM utilization, temperature, fan status etc…
- Non-metric data: ARP and bridge tables, LLDP neighbors, …

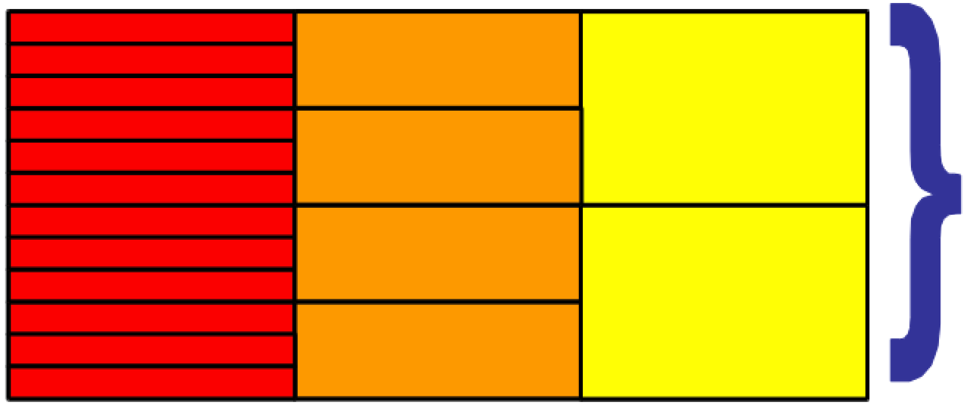
Metric storage: RRDtool

- Legacy storage format used by older tools like Cacti, Smokeping, and LibreNMS
- Optimized to use *fixed disk space*
  - Older data is stored at increasingly lower resolutions
  - But disk space is very cheap these days!
- Poor performance in terms of *disk I/O operations*

Metric storage: RRDtool

Recent data stored once every 5 minutes for the past 2 hours (1:24)

Old data averaged to one entry per day for the last 365 days (288:365)



--step 300

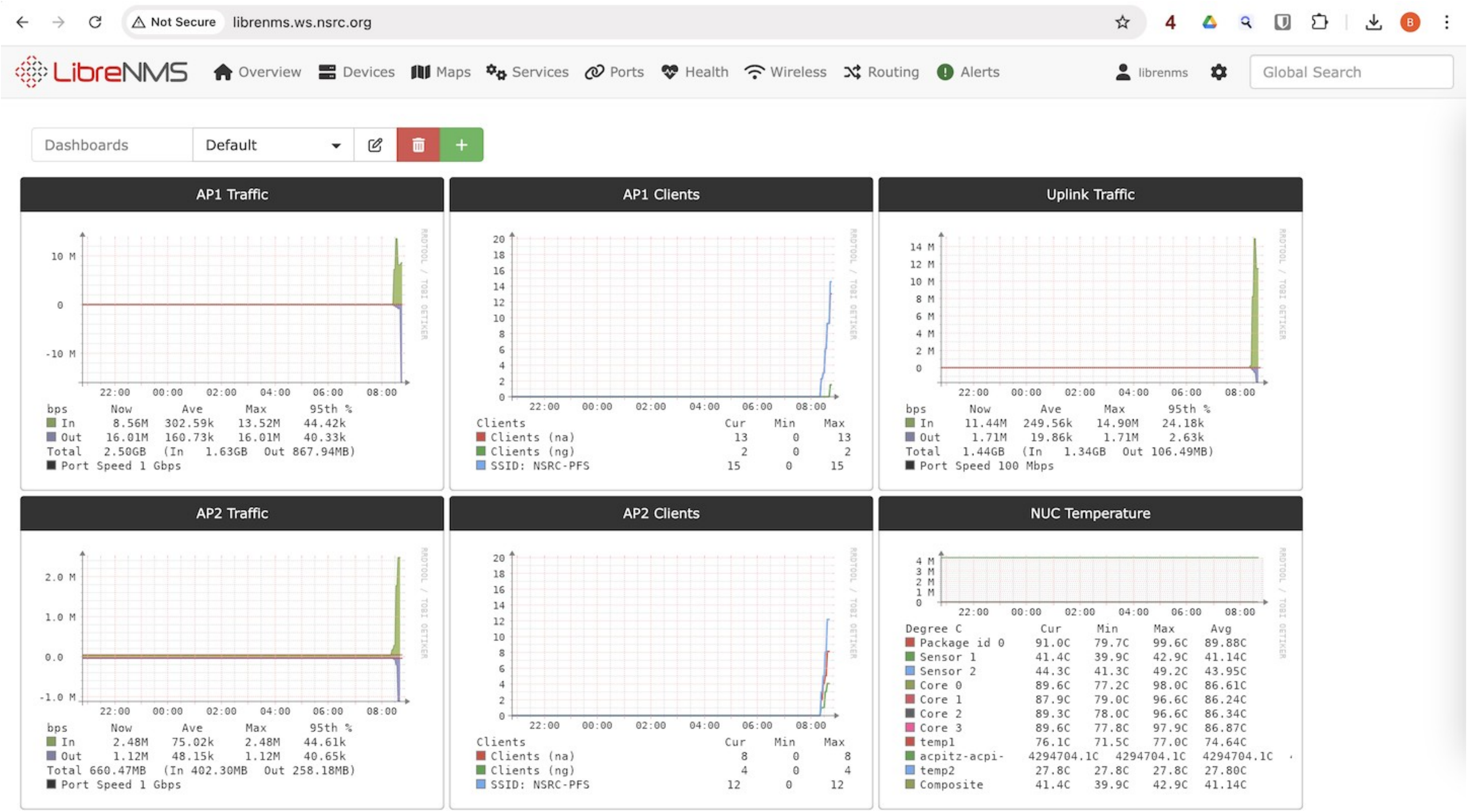
(5 minute input step size)

Medium length data averaged to one entry per half hour for the last 5 hours (6:10)

Software: LibreNMS (fork of Observium)

- "All-in-one" NMM platform, quick to deploy
- SNMP data collection
- Device inventory and discovery
- Topology discovery (LLDP/CDP)
- Auto-configuration of data collection for each device type
- Web interface
- Alerting

LibreNMS Demo



AP1 Traffic

```
bps         Now      Ave       Max      95th %
In        8.56M  302.59k    13.52M    44.42k
Out      16.01M  160.73k    16.01M    40.33k
Total    2.50GB  (In    1.63GB   Out 867.94MB)
Port Speed 1 Gbps
```

AP1 Clients

```
Clients                      Cur    Min    Max
Clients (na)                  13      0     13
Clients (ng)                   2      0      2
SSID: NSRC-PFS                15      0     15
```

Uplink Traffic

```
bps         Now      Ave       Max      95th %
In       11.44M  249.56k    14.90M    24.18k
Out       1.71M   19.86k     1.71M     2.63k
Total    1.44GB  (In    1.34GB   Out 106.49MB)
Port Speed 100 Mbps
```

AP2 Traffic

```
bps         Now      Ave       Max      95th %
In        2.48M   75.02k     2.48M    44.61k
Out       1.12M   48.15k     1.12M    40.65k
Total 660.47MB   (In 402.30MB   Out 258.18MB)
Port Speed 1 Gbps
```

AP2 Clients

```
Clients                      Cur    Min    Max
Clients (na)                   8      0      8
Clients (ng)                   4      0      4
SSID: NSRC-PFS                12      0     12
```

NUC Temperature

```
Degree C              Cur        Min        Max        Avg
Package id 0         91.0C      79.7C      99.6C      89.88C
Sensor 1             41.4C      39.9C      42.9C      41.14C
Sensor 2             44.3C      41.3C      49.2C      43.95C
Core 0               89.6C      77.2C      98.0C      86.61C
Core 1               87.9C      79.0C      96.6C      86.24C
Core 2               89.3C      78.0C      96.6C      86.34C
Core 3               89.6C      77.8C      97.9C      86.87C
temp1                76.1C      71.5C      77.0C      74.64C
acpitz-acpi-   4294704.1C 4294704.1C 4294704.1C
temp2                27.8C      27.8C      27.8C      27.80C
Composite            41.4C      39.9C      42.9C      41.14C
```

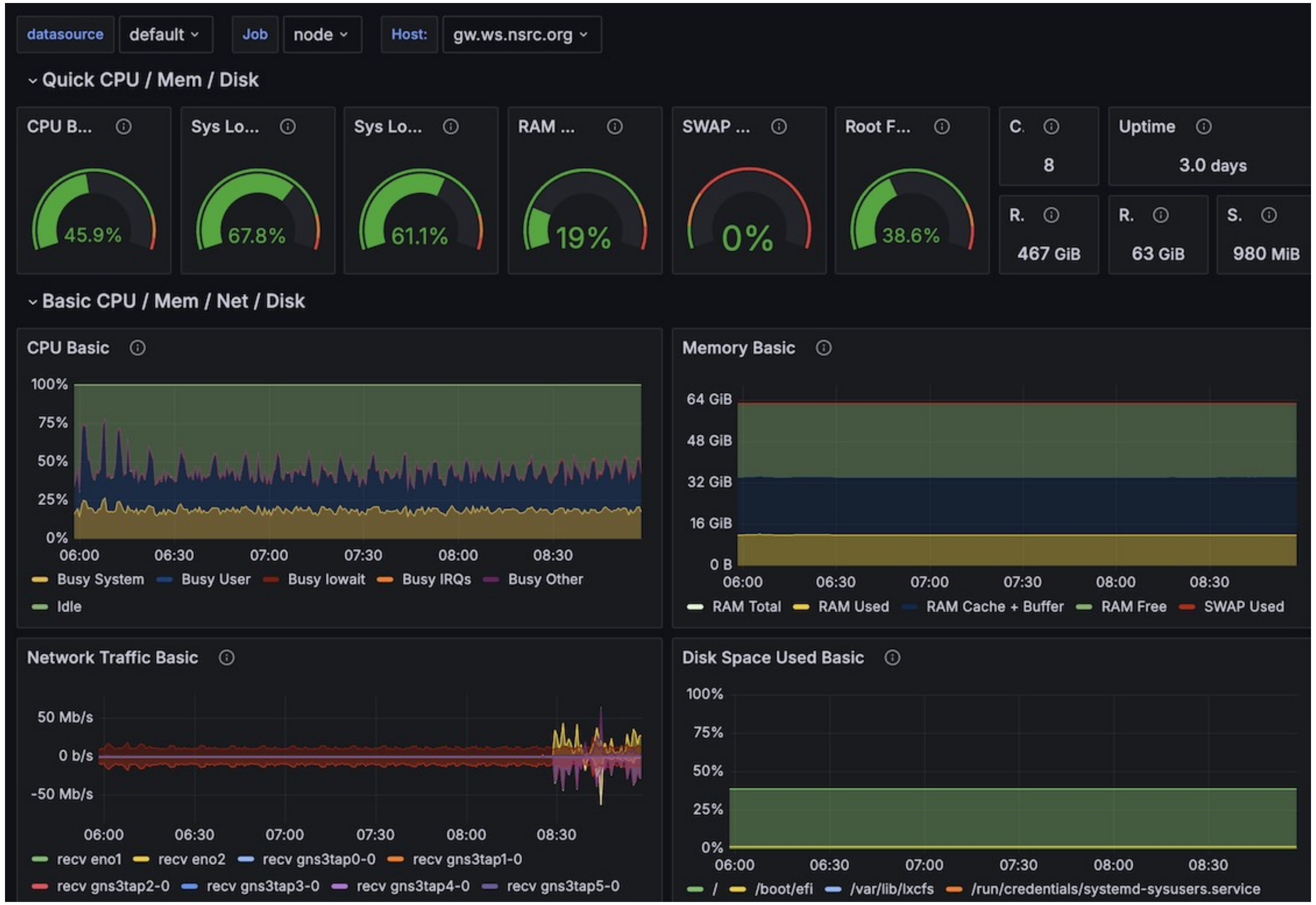
Limitations of LibreNMS

- Not efficient, primarily due to RRDtool
  - Creates large numbers of RRD files (graphs) per device
  - Scaling to large numbers of devices requires powerful hardware and/or sharding across multiple servers
- Many features not well documented

Alternative: Prometheus + Grafana

- Data collection using simple http protocol to scrape "exporters"
  - snmp_exporter for network devices
  - node_exporter for Linux/Unix systems
  - easy to write your own exporters to instrument any application
- Highly efficient time series database
  - Scales to millions of time series, highly performant
  - Unlike RRDtool, does not discard data (except configured retention time)
- Powerful query language (PromQL) used for graphing and alerting
- Clean separation of collection, storage, visualization, alerting

Demo: Prometheus + Grafana



datasource: default

Job: node

Host: gw.ws.nsrc.org

Limitations of Prometheus ecosystem

- Not a Network Management System
  - It's a generic metric collection system
  - "Kit of parts" that you assemble yourself
  - No device discovery or automatic device inventory
  - Tricky to set up for SNMP, beyond the supplied sample MIBs
- Steep learning curve (it's worthwhile!)
- Metrics only, no logs

Active Measurement

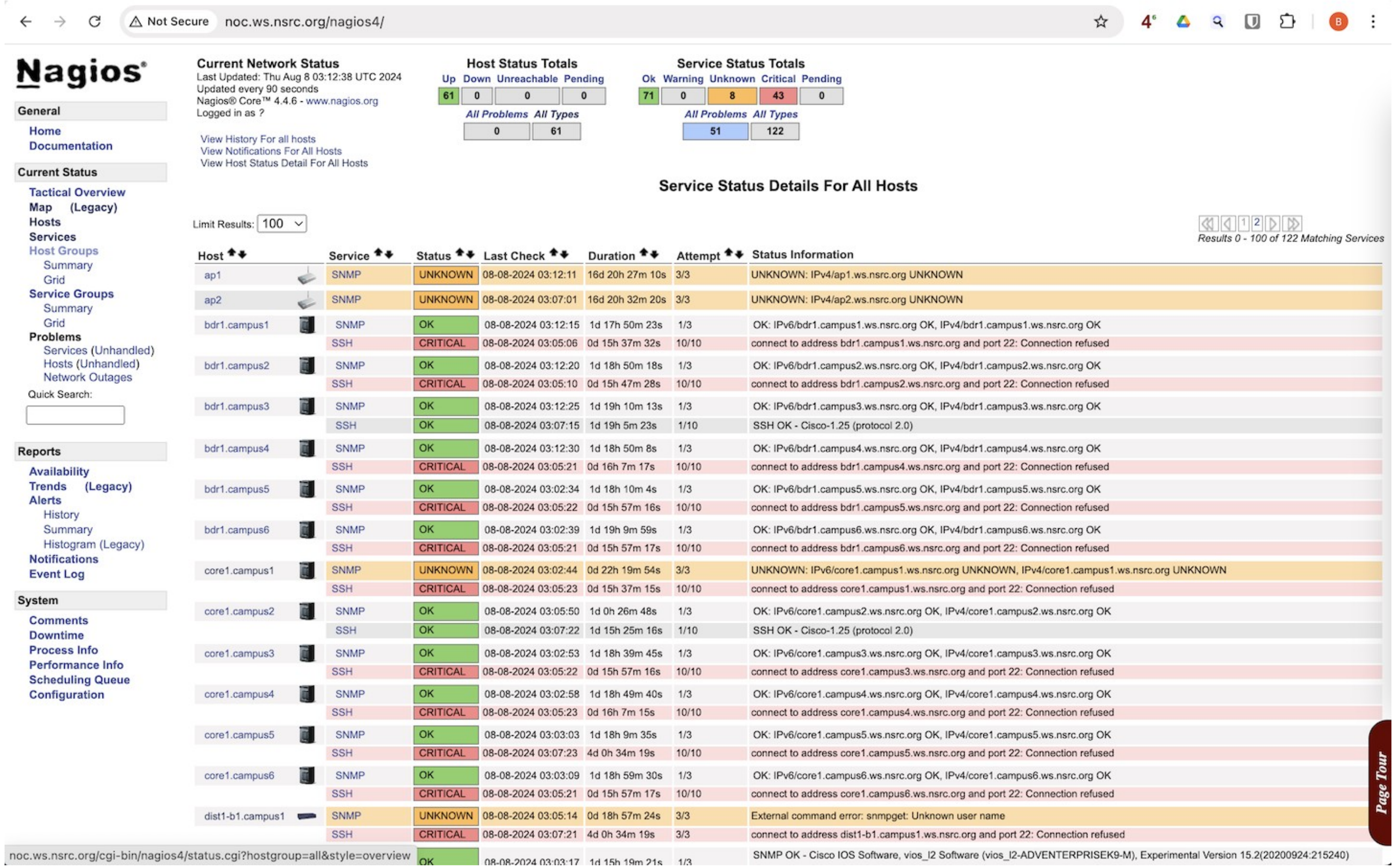
Active measurement (probing)

- Perform active tests across the network
  - ping tests
  - service tests (e.g. HTTP, DNS)
- Availability: test whether service is "up" or "down"
- Performance: measure response time
- Store, visualize, alert
- SLA reporting

Software: Nagios

- Main focus on "up/down" availability and alerting
- Configured by plain text files
- Tests done via running "plugins" which are easy to write
- Historical storage in plain text files
- More sophisticated derivatives available e.g. check_mk, omd

Demo: Nagios



Current Network Status
Last Updated: Thu Aug 8 03:12:38 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as ?

View History For all hosts
View Notifications For All Hosts
View Host Status Detail For All Hosts

Host Status Totals

Up	Down	Unreachable	Pending
61	0	0	0

All Problems	All Types
0	61

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
71	0	8	43	0

All Problems	All Types
51	122

Service Status Details For All Hosts

1 2

Results 0 - 100 of 122 Matching Services

Host	Service	Status	Last Check	Duration	Attempt	Status Information
ap1	SNMP	UNKNOWN	08-08-2024 03:12:11	16d 20h 27m 10s	3/3	UNKNOWN: IPv4/ap1.ws.nsrc.org UNKNOWN
ap2	SNMP	UNKNOWN	08-08-2024 03:07:01	16d 20h 32m 20s	3/3	UNKNOWN: IPv4/ap2.ws.nsrc.org UNKNOWN
bdr1.campus1	SNMP	OK	08-08-2024 03:12:15	1d 17h 50m 23s	1/3	OK: IPv6/bdr1.campus1.ws.nsrc.org OK, IPv4/bdr1.campus1.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:05:06	0d 15h 37m 32s	10/10	connect to address bdr1.campus1.ws.nsrc.org and port 22: Connection refused
bdr1.campus2	SNMP	OK	08-08-2024 03:12:20	1d 18h 50m 18s	1/3	OK: IPv6/bdr1.campus2.ws.nsrc.org OK, IPv4/bdr1.campus2.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:05:10	0d 15h 47m 28s	10/10	connect to address bdr1.campus2.ws.nsrc.org and port 22: Connection refused
bdr1.campus3	SNMP	OK	08-08-2024 03:12:25	1d 19h 10m 13s	1/3	OK: IPv6/bdr1.campus3.ws.nsrc.org OK, IPv4/bdr1.campus3.ws.nsrc.org OK
	SSH	OK	08-08-2024 03:07:15	1d 19h 5m 23s	1/10	SSH OK - Cisco-1.25 (protocol 2.0)
bdr1.campus4	SNMP	OK	08-08-2024 03:12:30	1d 18h 50m 8s	1/3	OK: IPv6/bdr1.campus4.ws.nsrc.org OK, IPv4/bdr1.campus4.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:05:21	0d 16h 7m 17s	10/10	connect to address bdr1.campus4.ws.nsrc.org and port 22: Connection refused
bdr1.campus5	SNMP	OK	08-08-2024 03:02:34	1d 18h 10m 4s	1/3	OK: IPv6/bdr1.campus5.ws.nsrc.org OK, IPv4/bdr1.campus5.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:05:22	0d 15h 57m 16s	10/10	connect to address bdr1.campus5.ws.nsrc.org and port 22: Connection refused
bdr1.campus6	SNMP	OK	08-08-2024 03:02:39	1d 19h 9m 59s	1/3	OK: IPv6/bdr1.campus6.ws.nsrc.org OK, IPv4/bdr1.campus6.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:05:21	0d 15h 57m 17s	10/10	connect to address bdr1.campus6.ws.nsrc.org and port 22: Connection refused
core1.campus1	SNMP	UNKNOWN	08-08-2024 03:02:44	0d 22h 19m 54s	3/3	UNKNOWN: IPv6/core1.campus1.ws.nsrc.org UNKNOWN, IPv4/core1.campus1.ws.nsrc.org UNKNOWN
	SSH	CRITICAL	08-08-2024 03:05:23	0d 15h 37m 15s	10/10	connect to address core1.campus1.ws.nsrc.org and port 22: Connection refused
core1.campus2	SNMP	OK	08-08-2024 03:05:50	1d 0h 26m 48s	1/3	OK: IPv6/core1.campus2.ws.nsrc.org OK, IPv4/core1.campus2.ws.nsrc.org OK
	SSH	OK	08-08-2024 03:07:22	1d 15h 25m 16s	1/10	SSH OK - Cisco-1.25 (protocol 2.0)
core1.campus3	SNMP	OK	08-08-2024 03:02:53	1d 18h 39m 45s	1/3	OK: IPv6/core1.campus3.ws.nsrc.org OK, IPv4/core1.campus3.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:05:22	0d 15h 57m 16s	10/10	connect to address core1.campus3.ws.nsrc.org and port 22: Connection refused
core1.campus4	SNMP	OK	08-08-2024 03:02:58	1d 18h 49m 40s	1/3	OK: IPv6/core1.campus4.ws.nsrc.org OK, IPv4/core1.campus4.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:05:23	0d 16h 7m 15s	10/10	connect to address core1.campus4.ws.nsrc.org and port 22: Connection refused
core1.campus5	SNMP	OK	08-08-2024 03:03:03	1d 18h 9m 35s	1/3	OK: IPv6/core1.campus5.ws.nsrc.org OK, IPv4/core1.campus5.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:07:23	4d 0h 34m 19s	10/10	connect to address core1.campus5.ws.nsrc.org and port 22: Connection refused
core1.campus6	SNMP	OK	08-08-2024 03:03:09	1d 18h 59m 30s	1/3	OK: IPv6/core1.campus6.ws.nsrc.org OK, IPv4/core1.campus6.ws.nsrc.org OK
	SSH	CRITICAL	08-08-2024 03:05:21	0d 15h 57m 17s	10/10	connect to address core1.campus6.ws.nsrc.org and port 22: Connection refused
dist1-b1.campus1	SNMP	UNKNOWN	08-08-2024 03:05:14	0d 18h 57m 24s	3/3	External command error: snmpget: Unknown user name
	SSH	CRITICAL	08-08-2024 03:07:21	4d 0h 34m 19s	3/3	connect to address dist1-b1.campus1.ws.nsrc.org and port 22: Connection refused
		OK	08-08-2024 03:03:17	1d 15h 19m 21s	1/3	SNMP OK - Cisco IOS Software, vios_l2 Software (vios_l2-ADVENTERPRISEK9-M), Experimental Version 15.2(20200924:215240)

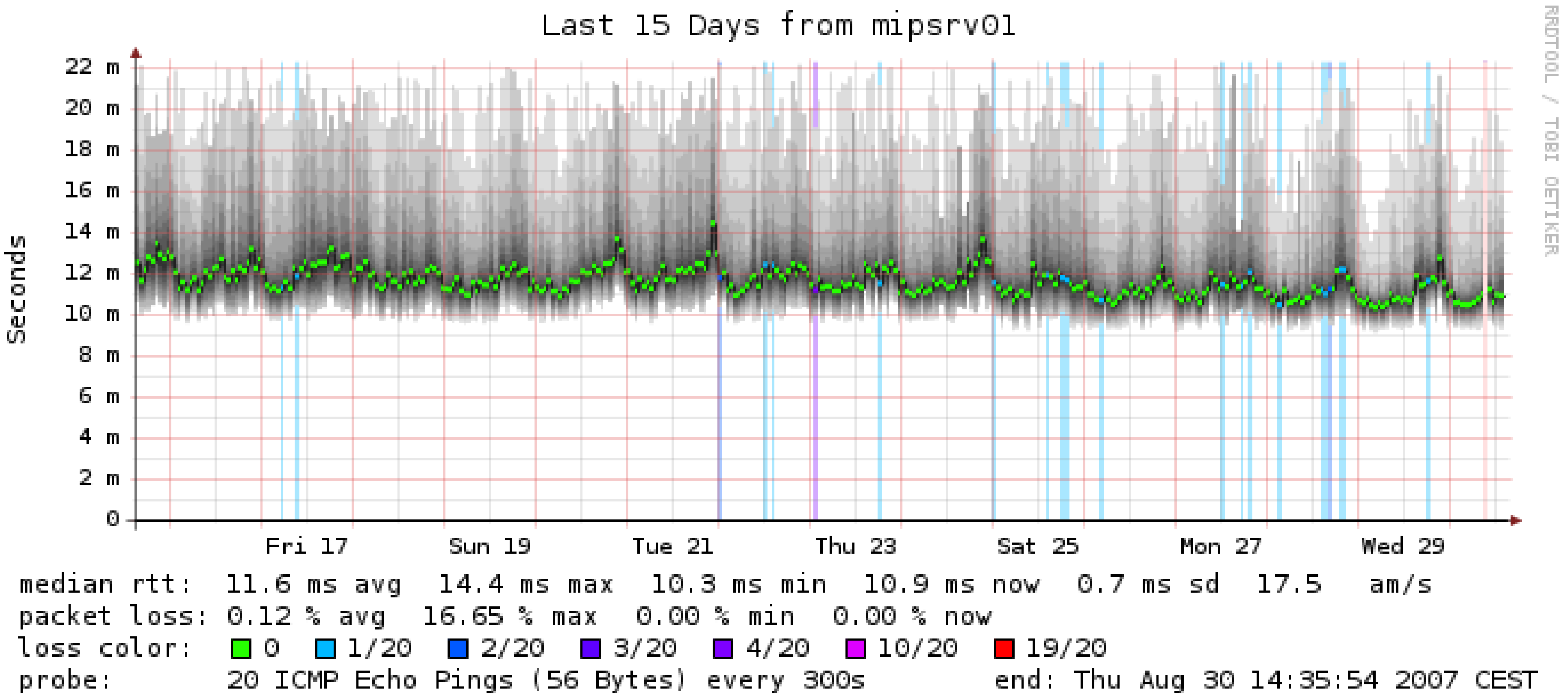
Limitations/Alternatives to Nagios

- Text file configuration
  - Pro: easy to backup and compare
  - Con: need to edit text files every time you add a device
- There are other tools in this space if you just want basic service availability checking, e.g. Uptime Kuma
  - and free services which will perform some tests from outside your network
- LibreNMS can also invoke Nagios plugins
  - Maybe that's sufficient for your needs?

Software: Smokeping

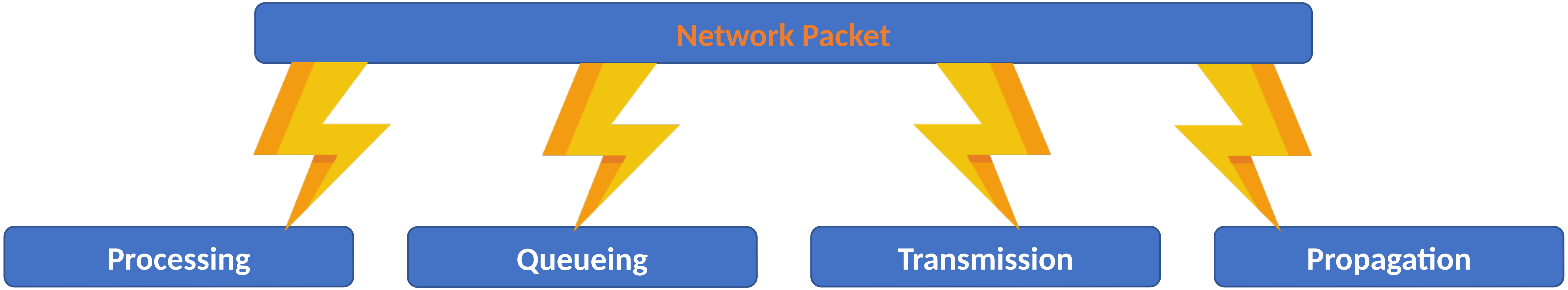
- Main focus on round-trip time and packet loss measurement
- Configured by plain text files
- Historical storage in RRD files
- Can also measure response times for DNS etc
  - Unfortunately the HTTP response time plugin (echoping) is unmaintained and has been removed

Demo: Smokeping



```
median rtt:   11.6 ms avg  14.4 ms max  10.3 ms min  10.9 ms now  0.7 ms sd  17.5   am/s
packet loss:  0.12 % avg  16.65 % max   0.00 % min   0.00 % now
loss color:   🟩 0   🟦 1/20  🟦 2/20  🟪 3/20  🟪 4/20  🟪 10/20  🟥 19/20
probe:        20 ICMP Echo Pings (56 Bytes) every 300s       end: Thu Aug 30 14:35:54 2007 CEST
```

What causes the variation in delay?



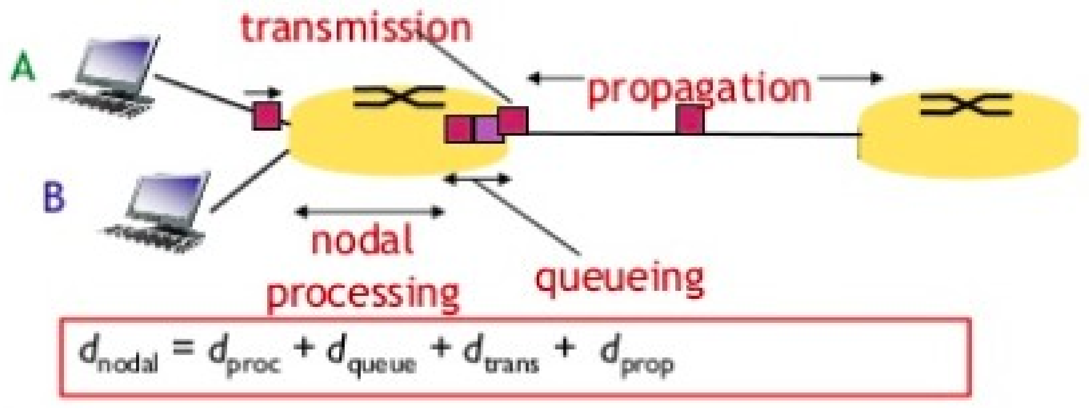
Network Packet

Processing

Queueing

Transmission

Propagation



$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

Courtesy https://www.slideshare.net/maamirfarooq/lec-4packet-delay-layered-architecture

What causes packet loss?

- Queue overruns; and
- Transmission errors
- Both of these cause TCP to reduce speed drastically (response to congestion)
- The speed drop depends very much on round-trip-time
  - Nearby destinations not affected much; International destinations very strongly affected

Limitations of Smokeping

- Text file config (see Nagios)
- RRD storage
- Low resolution: default send 20 packets every 5 minutes
    - Won't detect packet loss < 5%
    - Won't detect outages during the other 4 minutes 40 seconds

Alternative: probing with Prometheus

- blackbox_exporter
  - measures ping, DNS, HTTP
- smokeping_exporter
- nrpe_exporter
  - talks to Nagios plugins
- Run a test script that outputs Prometheus metrics
- Makes sense if you're already in the Prometheus ecosystem

Software: Perfsonar

- Very sensitive packet loss and RTT measurement
- By default sends 10 packets per second = 36,000 packets per hour
  - Detect packet loss as low as 0.003% over an hour
- Also performs periodic TCP throughput "speed test"
- Tests run *between* perfsonar nodes
  - Option to separate the measurement endpoints and central data storage
- Tools to build a mesh configuration

Netflow

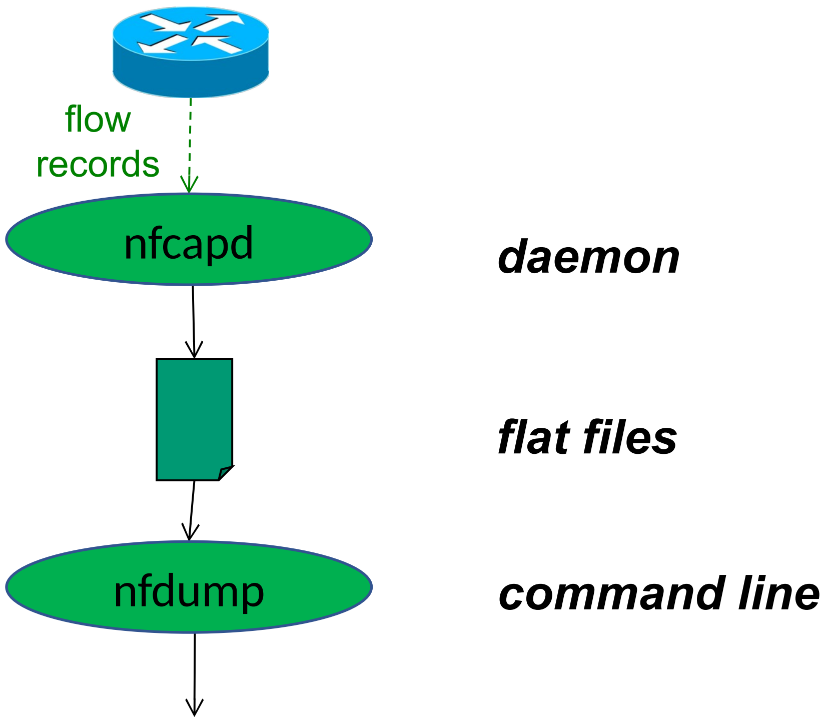
Netflow / IPFIX

- Records of individual packet flows
  - Collection of packets with same protocol, source IP, destination IP, source port and destination port
  - Record gives total number of bytes and total number of packets in the flow
- Flow records usually generated by router/firewall
- Sent via UDP to a flow collector
- Collector stores to disk, allows querying and visualization
- Very powerful, e.g. identify the "top talkers" on your network

Software: nfdump + nfsen

- Very resource efficient
  - Minimal disk I/O ops for received flows; even spinning hard drives are fine
  - No indexing
  - Flip side is that queries can be slow
- Old
  - nfdump still being actively maintained; nfsen barely so
- Not pretty

nfdump architecture



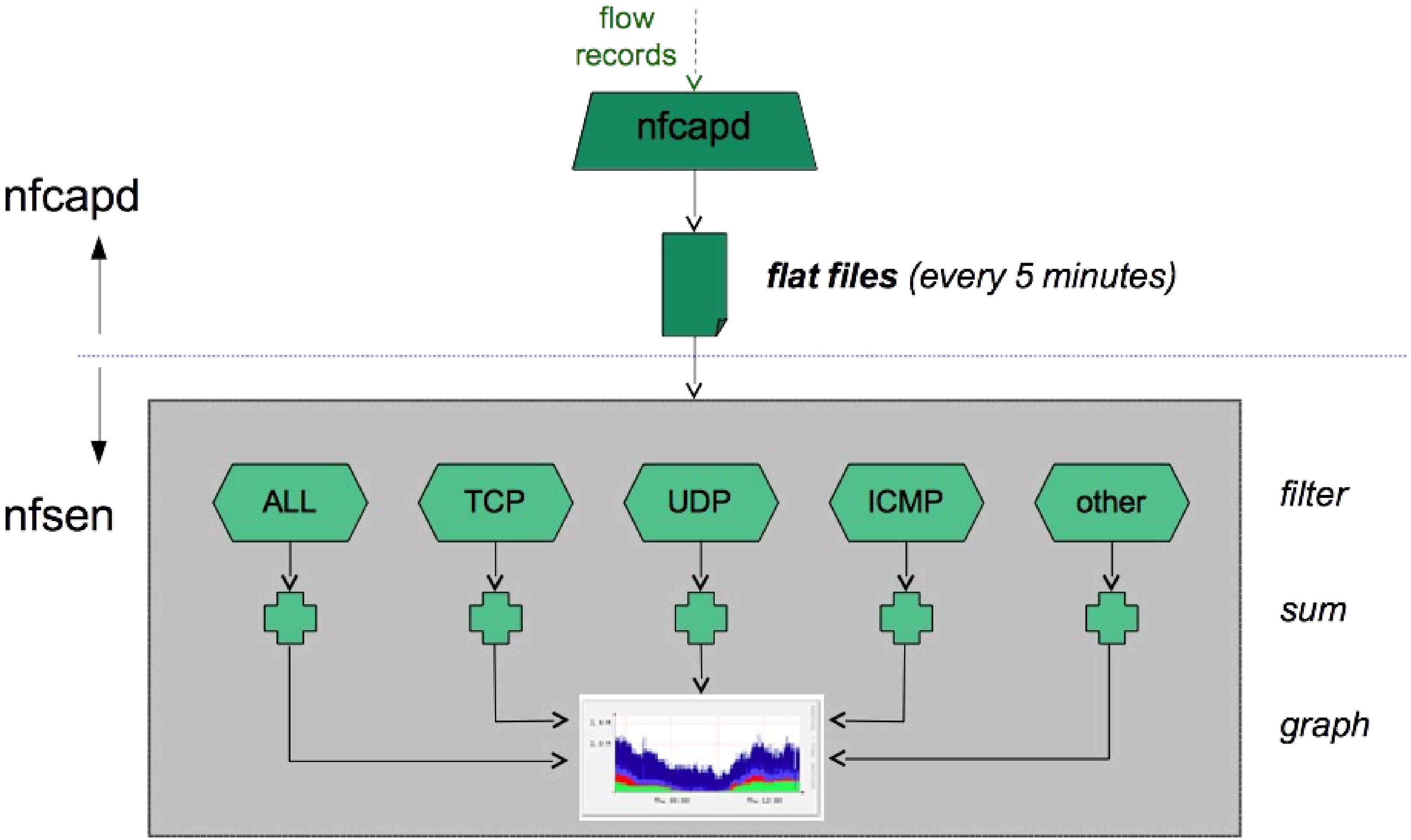
daemon

flat files

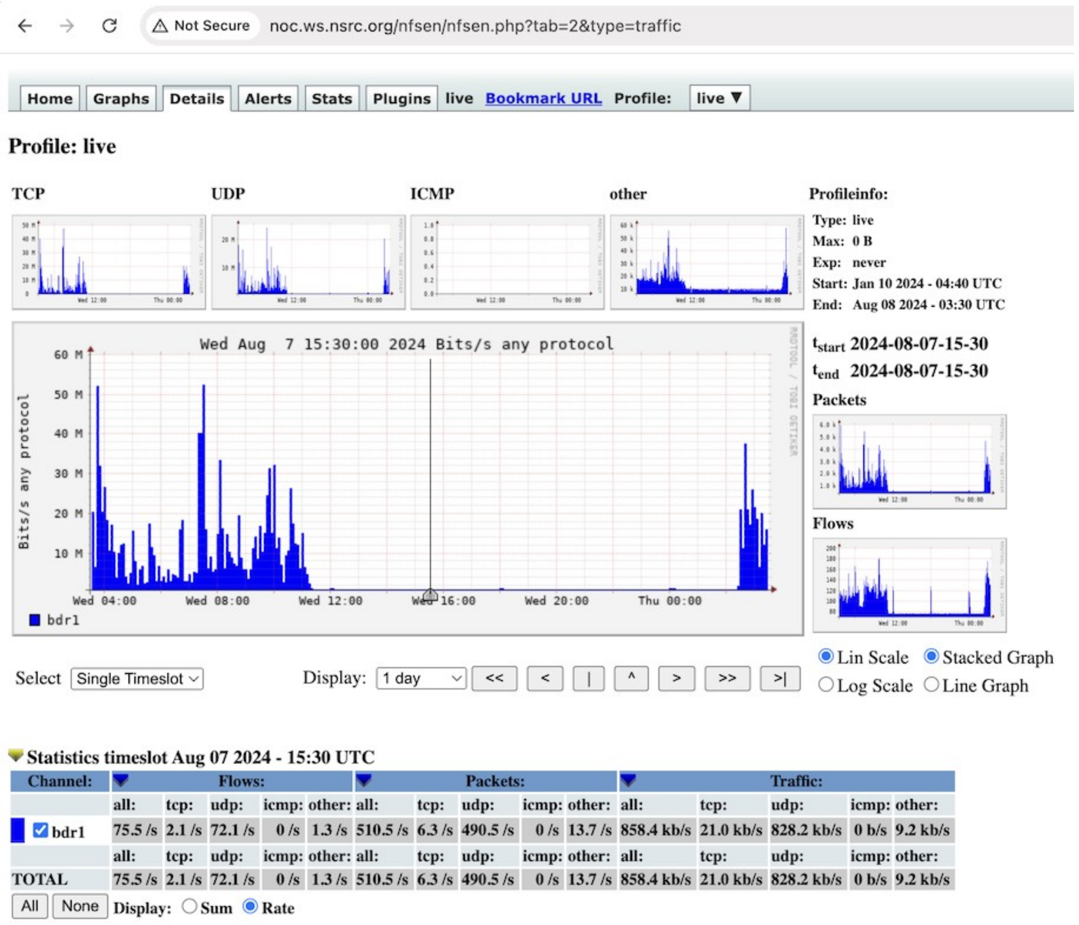
command line

```
Date flow start          Duration Proto      Src IP Addr:Port          Dst IP Addr:Port   Packets    Bytes Flows
2013-04-18 13:35:23.353  1482.000 UDP       10.10.0.119:55555 ->  190.83.150.177:54597      8683   445259     1
2013-04-18 13:35:23.353  1482.000 UDP    190.83.150.177:54597 ->     10.10.0.119:55555      8012   11.1 M     1
2013-04-18 13:48:21.353   704.000 TCP     196.38.180.96:6112  ->     10.10.0.119:62099        83    20326     1
2013-04-18 13:48:21.353   704.000 TCP       10.10.0.119:62099 ->   196.38.180.96:6112        105     5085     1
```

nfdump/nfsen architecture



Demo: nfsen



Profile: live

Profileinfo:
Type: live
Max: 0 B
Exp: never
Start: Jan 10 2024 - 04:40 UTC
End: Aug 08 2024 - 03:30 UTC

t_{start} 2024-08-07-15-30
t_{end} 2024-08-07-15-30

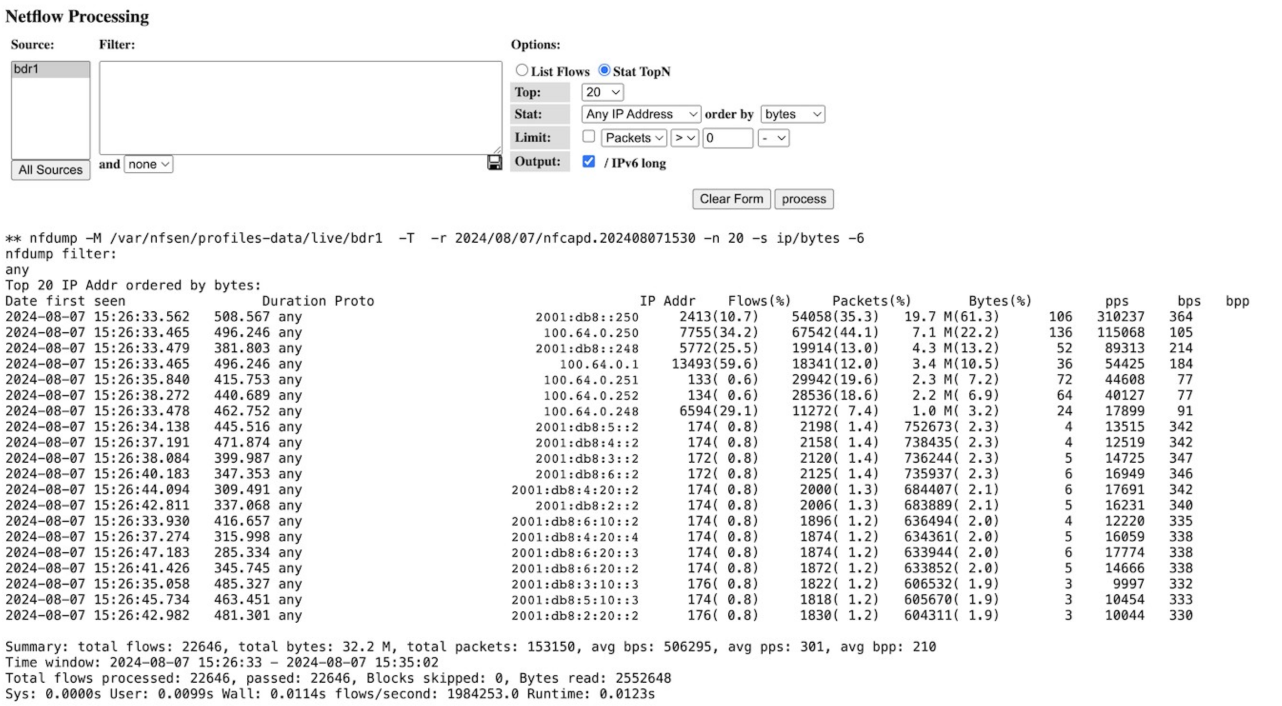
● Lin Scale ● Stacked Graph
○ Log Scale ○ Line Graph

Select: Single Timeslot
Display: 1 day

Statistics timeslot Aug 07 2024 - 15:30 UTC

Channel:	Flows:					Packets:					Traffic:				
	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:
☑ bdr1	75.5 /s	2.1 /s	72.1 /s	0 /s	1.3 /s	510.5 /s	6.3 /s	490.5 /s	0 /s	13.7 /s	858.4 kb/s	21.0 kb/s	828.2 kb/s	0 b/s	9.2 kb/s
	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:	all:	tcp:	udp:	icmp:	other:
TOTAL	75.5 /s	2.1 /s	72.1 /s	0 /s	1.3 /s	510.5 /s	6.3 /s	490.5 /s	0 /s	13.7 /s	858.4 kb/s	21.0 kb/s	828.2 kb/s	0 b/s	9.2 kb/s

All None Display: ○ Sum ● Rate



Netflow Processing

Source: bdr1
Filter:
and none

Options:
○ List Flows ● Stat TopN
Top: 20
Stat: Any IP Address order by bytes
Limit: □ Packets > 0 -
Output: ☑ / IPv6 long

Clear Form process

```
** nfdump -M /var/nfsen/profiles-data/live/bdr1  -T  -r 2024/08/07/nfcapd.202408071530 -n 20 -s ip/bytes -6
nfdump filter:
any
Top 20 IP Addr ordered by bytes:
Date first seen               Duration Proto                            IP Addr    Flows(%)     Packets(%)       Bytes(%)         pps      bps   bpp
2024-08-07 15:26:33.562   508.567 any                         2001:db8::250     2413(10.7)    54058(35.3)   19.7 M(61.3)     106   310237   364
2024-08-07 15:26:33.465   496.246 any                          100.64.0.250     7755(34.2)    67542(44.1)    7.1 M(22.2)     136   115068   105
2024-08-07 15:26:33.479   381.803 any                         2001:db8::248     5772(25.5)    19914(13.0)    4.3 M(13.2)      52    89313   214
2024-08-07 15:26:33.465   496.246 any                            100.64.0.1    13493(59.6)    18341(12.0)    3.4 M(10.5)      36    54425   184
2024-08-07 15:26:35.840   415.753 any                          100.64.0.251      133( 0.6)    29942(19.6)    2.3 M( 7.2)      72    44608    77
2024-08-07 15:26:38.272   440.689 any                          100.64.0.252      134( 0.6)    28536(18.6)    2.2 M( 6.9)      64    40127    77
2024-08-07 15:26:33.478   462.752 any                          100.64.0.248     6594(29.1)    11272( 7.4)    1.0 M( 3.2)      24    17899    91
2024-08-07 15:26:34.138   445.516 any                         2001:db8:5::2      174( 0.8)     2198( 1.4)    752673( 2.3)      4    13515   342
2024-08-07 15:26:37.191   471.874 any                         2001:db8:4::2      174( 0.8)     2158( 1.4)    738435( 2.3)      4    12519   342
2024-08-07 15:26:38.084   399.987 any                         2001:db8:3::2      172( 0.8)     2120( 1.4)    736244( 2.3)      5    14725   347
2024-08-07 15:26:40.183   347.353 any                         2001:db8:6::2      172( 0.8)     2125( 1.4)    735937( 2.3)      6    16949   346
2024-08-07 15:26:44.094   309.491 any                      2001:db8:4:20::2      174( 0.8)     2000( 1.3)    684407( 2.1)      6    17691   342
2024-08-07 15:26:42.811   337.068 any                         2001:db8:2::2      174( 0.8)     2006( 1.3)    683889( 2.1)      5    16231   340
2024-08-07 15:26:33.930   416.657 any                      2001:db8:6:10::2      174( 0.8)     1896( 1.2)    636494( 2.0)      4    12220   335
2024-08-07 15:26:37.274   315.998 any                      2001:db8:4:20::4      174( 0.8)     1874( 1.2)    634361( 2.0)      5    16059   338
2024-08-07 15:26:47.183   285.334 any                      2001:db8:6:20::3      174( 0.8)     1874( 1.2)    633944( 2.0)      6    17774   338
2024-08-07 15:26:41.426   345.745 any                      2001:db8:6:20::2      174( 0.8)     1872( 1.2)    633852( 2.0)      5    14666   338
2024-08-07 15:26:35.058   485.327 any                      2001:db8:3:10::3      176( 0.8)     1822( 1.2)    606532( 1.9)      3     9997   332
2024-08-07 15:26:45.734   463.451 any                      2001:db8:5:10::3      174( 0.8)     1818( 1.2)    605670( 1.9)      3    10454   333
2024-08-07 15:26:42.982   481.301 any                      2001:db8:2:20::2      176( 0.8)     1830( 1.2)    604311( 1.9)      3    10044   330

Summary: total flows: 22646, total bytes: 32.2 M, total packets: 153150, avg bps: 506295, avg pps: 301, avg bpp: 210
Time window: 2024-08-07 15:26:33 - 2024-08-07 15:35:02
Total flows processed: 22646, passed: 22646, Blocks skipped: 0, Bytes read: 2552648
Sys: 0.0000s User: 0.0099s Wall: 0.0114s flows/second: 1984253.0 Runtime: 0.0123s
```

Some Netflow collector alternatives

- Filebeat + Elasticsearch* + Kibana*
  - Elasticsearch is very resource intensive, due to up-front indexing
  - You *must* use SSD, and you needs lots of RAM and CPU
  - Expect your data to expand in size by a factor of 10
- Elastiflow: was free, now commercial
  - free basic license available for up to 4,000 flows per second
  - but need to renew it annually – will it always remain free?
- ntop-ng: commercial, but free for R&E networks
  - real-time reporting + historical storage in Clickhouse database
- Akvorado: free, uses Kafka and Clickhouse, relatively new

** or Opensearch + Opensearch Dashboards*

Other ways to generate flow records

- Use a switch mirror port and a software flow monitor
    - softflowd, pfflowd: generate standard Netflow records
    - packetbeat
        - JSON for insertion into Elasticsearch etc
        - Can also decode content to a degree (e.g. DNS queries/responses)
- May be convenient place to run an IDS as well

Logs

Logs

Detailed records of *individual events*

Unstructured text, e.g. syslog:

```
2021-12-04 22:02:35 gw1 publickey accepted for user: oxidized
2021-12-04 22:02:35 gw1 user oxidized logged in from 10.12.255.40 via ssh
```

Structured, e.g. JSON:

```
{"@timestamp":"2021-12-04T22:08:37.694Z","type":"dns",
 "dns":{"question":{"type":"A","class":"IN","name":"nsrc.org"},
        "type":"answer","resolved_ip":["128.223.157.25"]}}
```

Other binary examples: Netflow records, SNMP traps, RADIUS accounting records

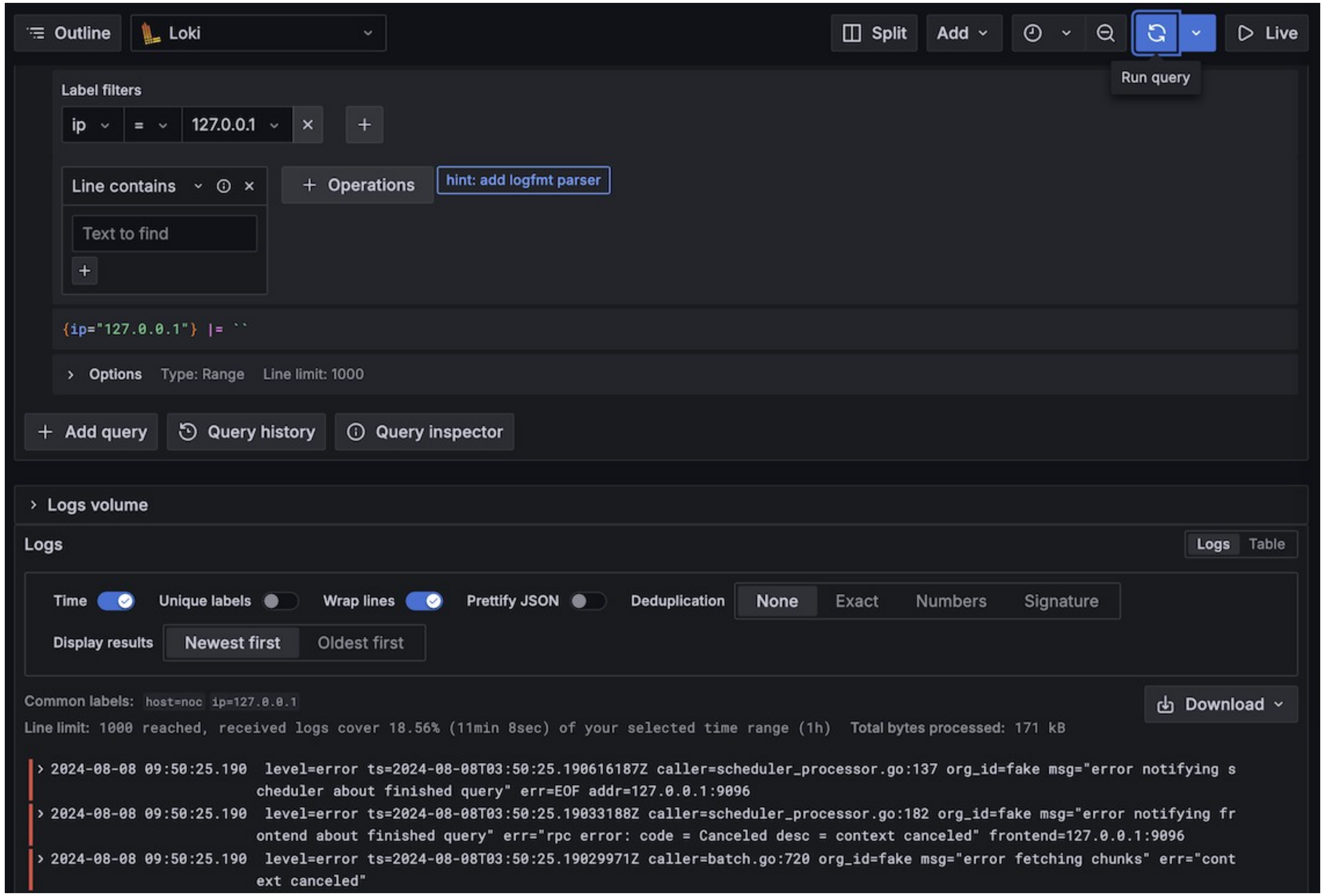
Logs compared to Metrics

- Logs are richer, more detailed, more granular
- Much larger volume generated
- Often required for debugging to know *exactly what happened and why*
- Metrics are good for spotting trends that prompt further investigation
- Authentication logs will tell you who has been using a given IP address at a given time

Syslog

- From network devices, Linux/Unix servers
- Traditionally sent over UDP, can also use TCP
- Software available to convert Windows events to syslog
- Various tools to capture and store the logs
  - rsyslog/syslog-ng, write to plain text files (grep to search)
  - log aggregators/pipelines: fluentd, filebeat, vector.dev, OTel collector, alloy
  - logstash + elasticsearch
  - loki + grafana; victoria-logs
  - also expensive commercial platforms (splunk, ...)

Demo: loki + grafana



Loki

Label filters

ip = 127.0.0.1

Line contains

```
{ip="127.0.0.1"} |= ``
```

Options Type: Range Line limit: 1000

Logs

Time: on · Unique labels: off · Wrap lines: on · Prettify JSON: off · Deduplication: None | Exact | Numbers | Signature (None)

Display results: Newest first | Oldest first (Newest first)

Common labels: host=noc ip=127.0.0.1

Line limit: 1000 reached, received logs cover 18.56% (11min 8sec) of your selected time range (1h) Total bytes processed: 171 kB

```
> 2024-08-08 09:50:25.190  level=error ts=2024-08-08T03:50:25.190616187Z caller=scheduler_processor.go:137 org_id=fake msg="error notifying s
                           cheduler about finished query" err=EOF addr=127.0.0.1:9096
> 2024-08-08 09:50:25.190  level=error ts=2024-08-08T03:50:25.19033188Z caller=scheduler_processor.go:182 org_id=fake msg="error notifying fr
                           ontend about finished query" err="rpc error: code = Canceled desc = context canceled" frontend=127.0.0.1:9096
> 2024-08-08 09:50:25.190  level=error ts=2024-08-08T03:50:25.19029971Z caller=batch.go:720 org_id=fake msg="error fetching chunks" err="cont
                           ext canceled"
```

UNIVERSITY OF OREGON

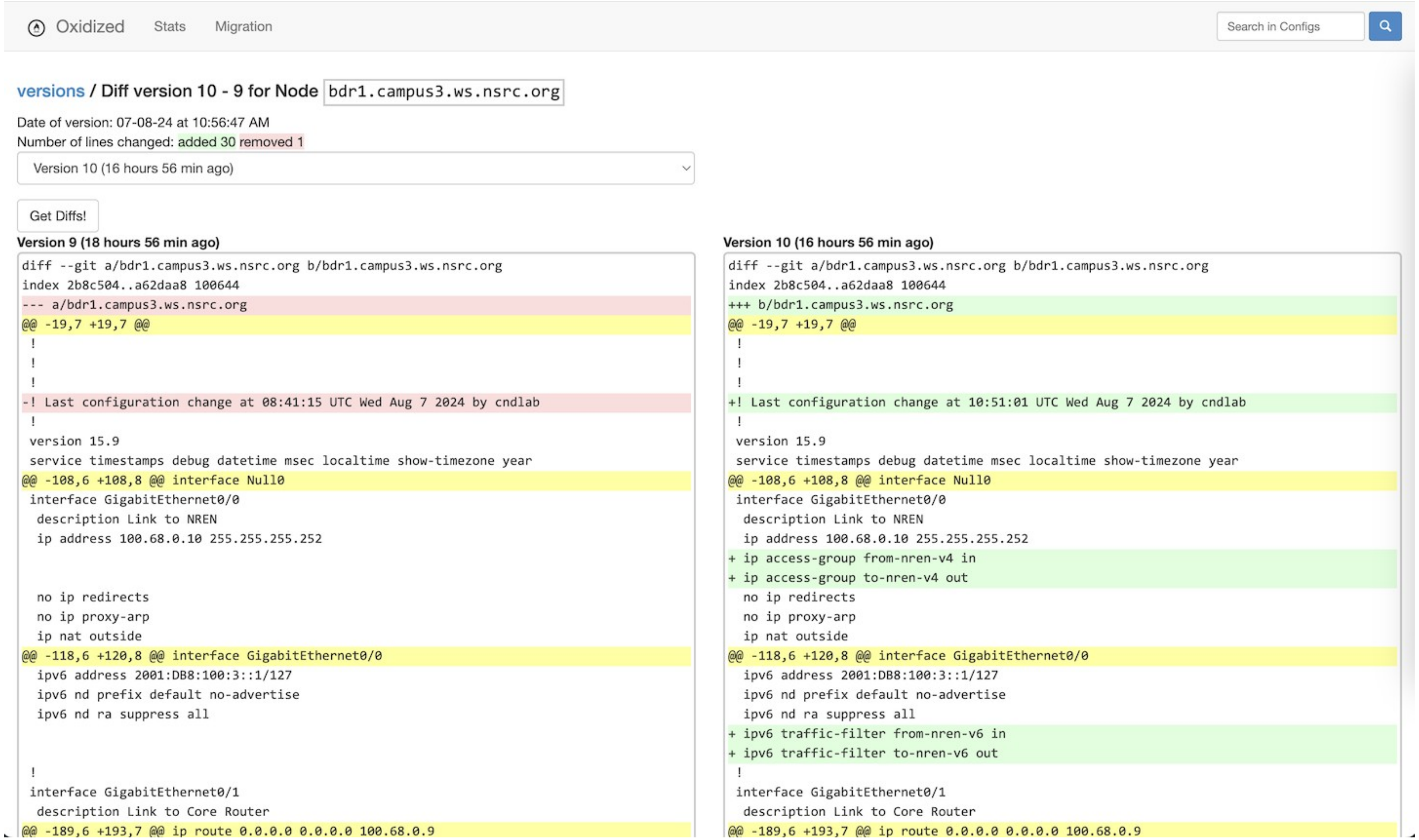
NSRC
Network Startup Resource Center

Configuration Management

Configuration backup

- Periodically connect to every network device and automatically download the configuration
- Store versions in a version control system
  - Configuration backups
- Compare with previous version, generate diffs
  - Send E-mail if there has been a change
- Software options: RANCID, Oxidized

Demo: Oxidized



versions / Diff version 10 - 9 for Node `bdr1.campus3.ws.nsrc.org`

Date of version: 07-08-24 at 10:56:47 AM
Number of lines changed: added 30 removed 1

Version 10 (16 hours 56 min ago)

Get Diffs!

Version 9 (18 hours 56 min ago)

```
diff --git a/bdr1.campus3.ws.nsrc.org b/bdr1.campus3.ws.nsrc.org
index 2b8c504..a62daa8 100644
--- a/bdr1.campus3.ws.nsrc.org
@@ -19,7 +19,7 @@
 !
 !
 !
-! Last configuration change at 08:41:15 UTC Wed Aug 7 2024 by cndlab
 !
 version 15.9
 service timestamps debug datetime msec localtime show-timezone year
@@ -108,6 +108,8 @@ interface Null0
 interface GigabitEthernet0/0
  description Link to NREN
  ip address 100.68.0.10 255.255.255.252


  no ip redirects
  no ip proxy-arp
  ip nat outside
@@ -118,6 +120,8 @@ interface GigabitEthernet0/0
  ipv6 address 2001:DB8:100:3::1/127
  ipv6 nd prefix default no-advertise
  ipv6 nd ra suppress all


 !
 interface GigabitEthernet0/1
  description Link to Core Router
@@ -189,6 +193,7 @@ ip route 0.0.0.0 0.0.0.0 100.68.0.9
```

Version 10 (16 hours 56 min ago)

```
diff --git a/bdr1.campus3.ws.nsrc.org b/bdr1.campus3.ws.nsrc.org
index 2b8c504..a62daa8 100644
+++ b/bdr1.campus3.ws.nsrc.org
@@ -19,7 +19,7 @@
 !
 !
 !
+! Last configuration change at 10:51:01 UTC Wed Aug 7 2024 by cndlab
 !
 version 15.9
 service timestamps debug datetime msec localtime show-timezone year
@@ -108,6 +108,8 @@ interface Null0
 interface GigabitEthernet0/0
  description Link to NREN
  ip address 100.68.0.10 255.255.255.252
+ ip access-group from-nren-v4 in
+ ip access-group to-nren-v4 out
  no ip redirects
  no ip proxy-arp
  ip nat outside
@@ -118,6 +120,8 @@ interface GigabitEthernet0/0
  ipv6 address 2001:DB8:100:3::1/127
  ipv6 nd prefix default no-advertise
  ipv6 nd ra suppress all
+ ipv6 traffic-filter from-nren-v6 in
+ ipv6 traffic-filter to-nren-v6 out
 !
 interface GigabitEthernet0/1
  description Link to Core Router
@@ -189,6 +193,7 @@ ip route 0.0.0.0 0.0.0.0 100.68.0.9
```

Alerting

Alerting

- Nagios and LibreNMS have alerting as core functionality
  - Smokeping can do it too
- Prometheus has Alertmanager (also karma/alerta dashboards)
  - Richness of PromQL allows for sophisticated alert conditions, e.g. "alert if rate of increase of disk space used predicts disk to be full in 24 hours"
- Grafana has its own alerting system
- Configuring alerts in all these is done differently
- Delivery options include E-mail, SMS, Slack, Telegram and commercial services like Pagerduty and VictorOps

Alerting

- Too many alerts are worse than too few alerts
  - "Alert fatigue"
  - Does this condition really require immediate attention?
  - Alerts should be urgent, important, actionable, and real
  - Less urgent conditions via summary E-mails, dashboards etc
- General principle: alert on symptoms, not causes
  - Alert on "web server not responding" more important than "database down"
  - These are the things that users care about
- Please read this "Philosophy on alerting": https://docs.google.com/document/d/199PqyG3UsyXlwieHaqbGiWVa8eMWi8zzAn0YfcApr8Q/