Experiences with

on the Janet network

Tim Chown (Jisc)
Duncan Rand, Raul Lopes, Chris Walker (Jisc)

LHCOPN-LHCONE #53, Beijing, 10 October 2024

Using perfSONAR on the Janet network

Thoughts and experience of our use of perfSONAR

Jisc and the Janet network

Jisc network performance test facilities

perfSONAR support

Communities, including GridPP

perfSONAR value

Performance tuning with perfSONAR

Jisc and the Janet network

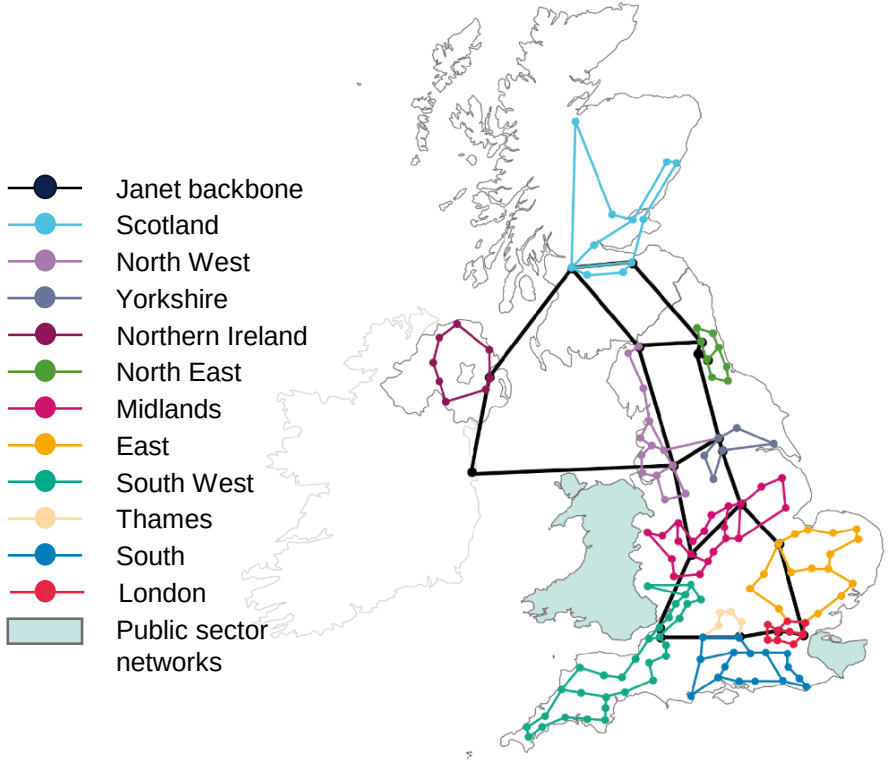
The UK National Research and Education Network (NREN)

Jisc operates the Janet network

- Connects around 160 universities, 100’s of FE colleges, plus research institutes and other organisations
- Backbone links up to 800Gbit/s
- Largely long-lease dark fibre (9,000km) and circuits built on Openreach services

Part of the global R&E network infrastructure

- Wider R&E connectivity via 400Gbit/s circuit to GÉANT



Supporting our member communities

Optimising members' use of their Janet connectivity

We have many science communities who need to move large volumes of data within and beyond Janet. GridPP is the largest example.

Implicit requirement to tune the network and end systems to optimise performance

- Jisc advises sites to follow 'Science DMZ' principles
- https://fasterdata.es.net/science-dmz/

Requirements mostly focused on throughput, but there are latency examples too

It's thus very important to Jisc to provide network performance test facilities and tools for our members

- Persistent monitoring is particularly useful

We also assist members in using the tools, performance diagnosis, capacity testing, identifying bottlenecks, etc.

Our two network performance team members are 50% seconded from universities

Jisc's network performance test facilities

Open to our members, and their collaborators, to use

Hosted in our Slough DC and one of our London PoPs

- Includes open 10G and 100G perfSONAR servers
- We offer a virtualised perfSONAR archive and Grafana mesh hosting

We also host

- 10G and 100G iperf and ethr servers (100G on request)
- 10G data transfer node (DTN) for application-oriented disk-to-disk tests (100G coming)

All facilities support IPv4 and IPv6, jumbo frames, option to use TCP-BBRv3

See https://www.jisc.ac.uk/guides/using-the-janet-network-performance-test-facilities

- Email netperf@jisc.ac.uk for any assistance or advice

Jisc

Network performance also includes latency

Not just about throughput testing

Networked music performance

Latency requirement < 30ms

NREN networks like Janet are well-tuned for low latency

Need to use high spec hardware

LoLa 2.0 software https://lola.conts.it/

perfSONAR is very useful

Also see https://timemap.geant.org/

https://www.youtube.com/watch?v=LK2WNyfLGlc

High-level perfSONAR advice

How do we advise sites deploy perfSONAR?

General advice is to deploy perfSONAR at your campus edge and/or alongside your local endpoint (typically storage)

If part of a test mesh for a community, sites can install a minimal ‘testpoint’ build and send all measurement results to the Jisc archive and see results on a Jisc hosted Grafana view

If a site wants to test with multiple collaborators and have more control it can run a full ‘toolkit’ install, archive data locally, and run its own Grafana views

See https://docs.perfsonar.net/install_options.html

We’ll help members whatever they choose to do

Jisc

Janet communities

Handling the perfSONAR 5.0 to 5.1 transition

There’s a significant change in the way results are presented and viewed in the newest 5.1 release

The old ‘classic’ maddash views are gone (unless you choose to run the older software)

We now have support for new, slicker, Grafana-based views

We’re keen to ensure the transition is a smooth one

The WLCG is in the process of updating its former maddash views

Jisc’s members are welcome to join our UK ‘test’ mesh to check their systems are operating correctly - running the desired measurements and archiving them - we then provide them with Grafana views

Jisc

Jisc’s UK test mesh

Hosted on Jisc’s virtual platforms

See here.

Sites run a psConfig script to join

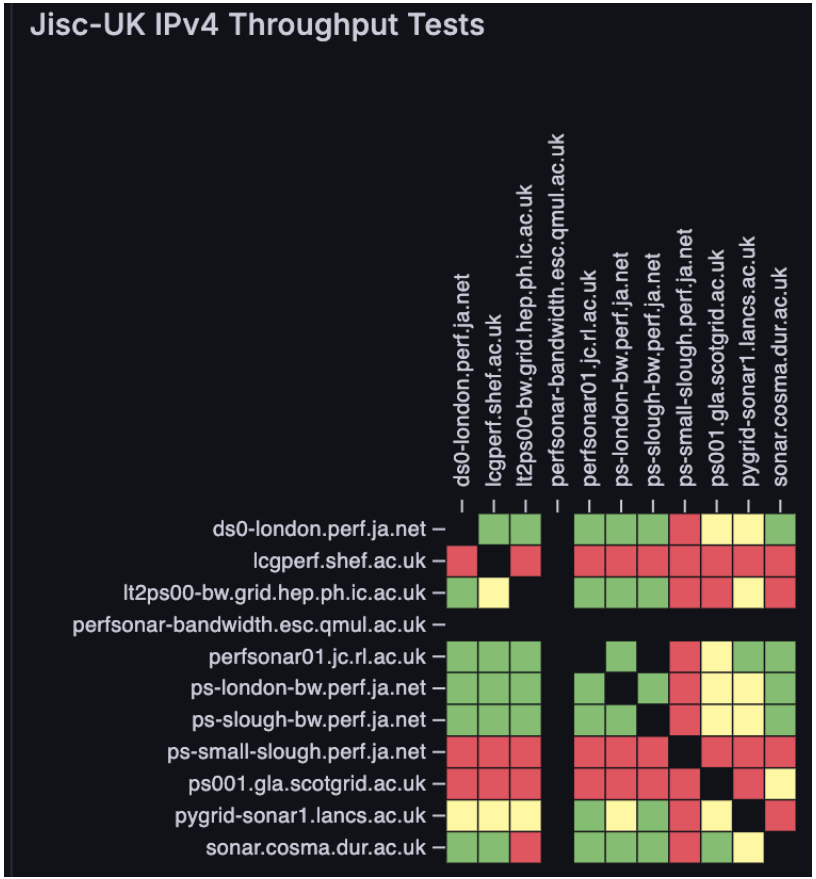
pscheduler ensures throughput tests are non-contending for all sites

The results are sent to our Jisc-hosted archive

Grafana dashboard shows results of tests from our central archive of results

You can click on any element of the mesh to view historical results over time

On the right, throughput is colour-coded; yellow is <5Gbit/s. *ps-small-slough* is all red as it’s only a 1G server (small form factor)



Example: drilling down to throughput over time

Here, ps-london to a RAL node

See here.

Nice example of good single stream TCP iperf throughput

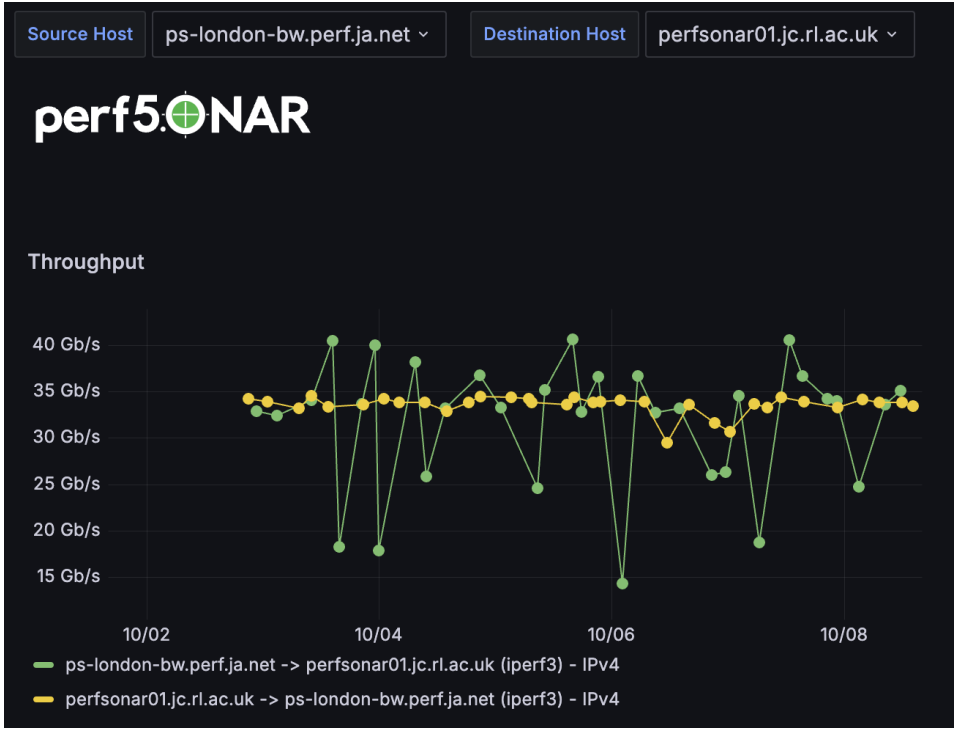
More consistent from RAL to Jisc-London than from Jisc-London to RAL

Would be interesting to explore why

May be down to specific server tuning, kernel version, real traffic competing, …

The iperf throughput test is 30 seconds TCP by default, so should be too disruptive to real traffic

Bear in mind typical WLCG traffic profiles



perfSONAR host info data

New - and useful - in 5.1

See here for ps-london example

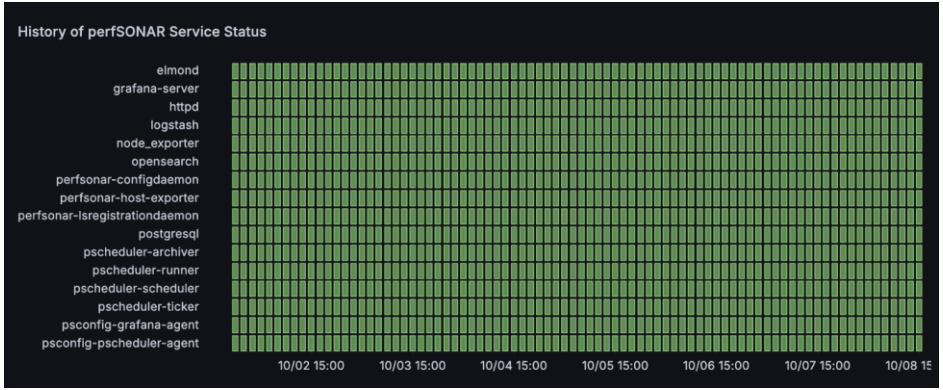
Host data is held for 1 week by default

Useful to troubleshoot a perfSONAR server, e.g., to check status of perfSONAR processes over time

But also

- CPU
- Free memory, disk
- Network utilisation, etc

And all tuning parameters, CCAs, etc can be viewed and checked



UK example: our GridPP community

The UK part of the WLCG

A collaboration of UK institutes providing data-intensive distributed computing resources for the UK High Energy Physics community

RAL is the UK Tier-1, with 2x100G LHCOPN and 400G general IP/LHCONE to Janet

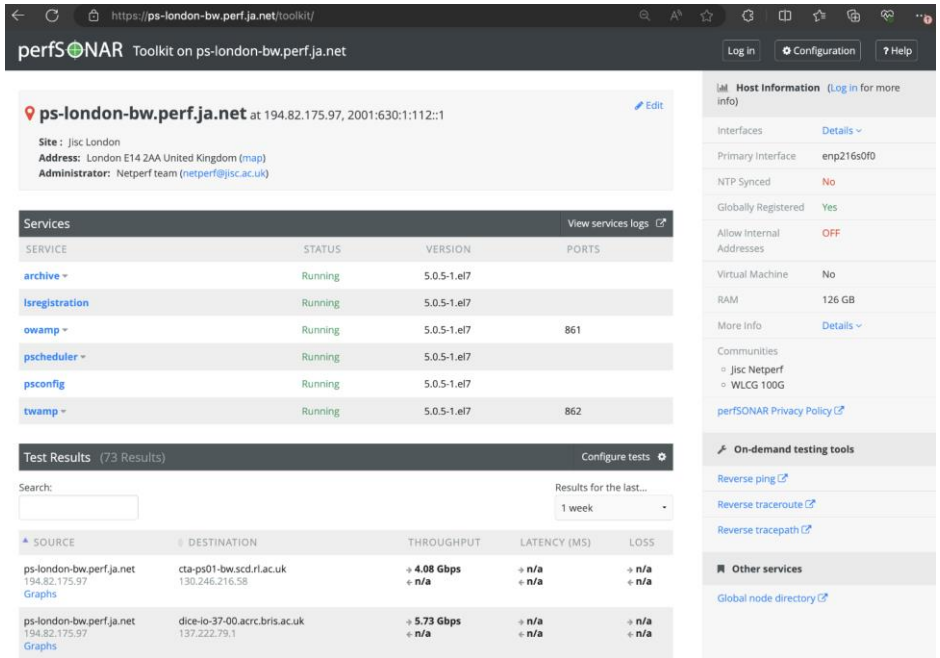
Many Tier-2s, at least four with 100G to Janet

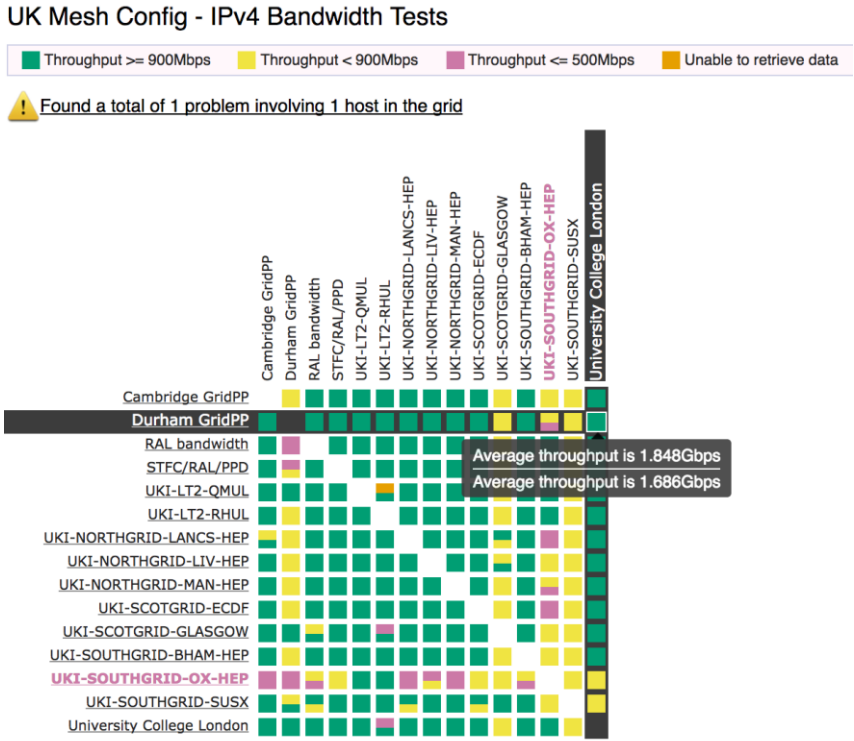
Some use of LHCONE (more is encouraged)

Janet backbone is up to 800G, our peering to GÉANT is 400G for R&E IP including LHCONE

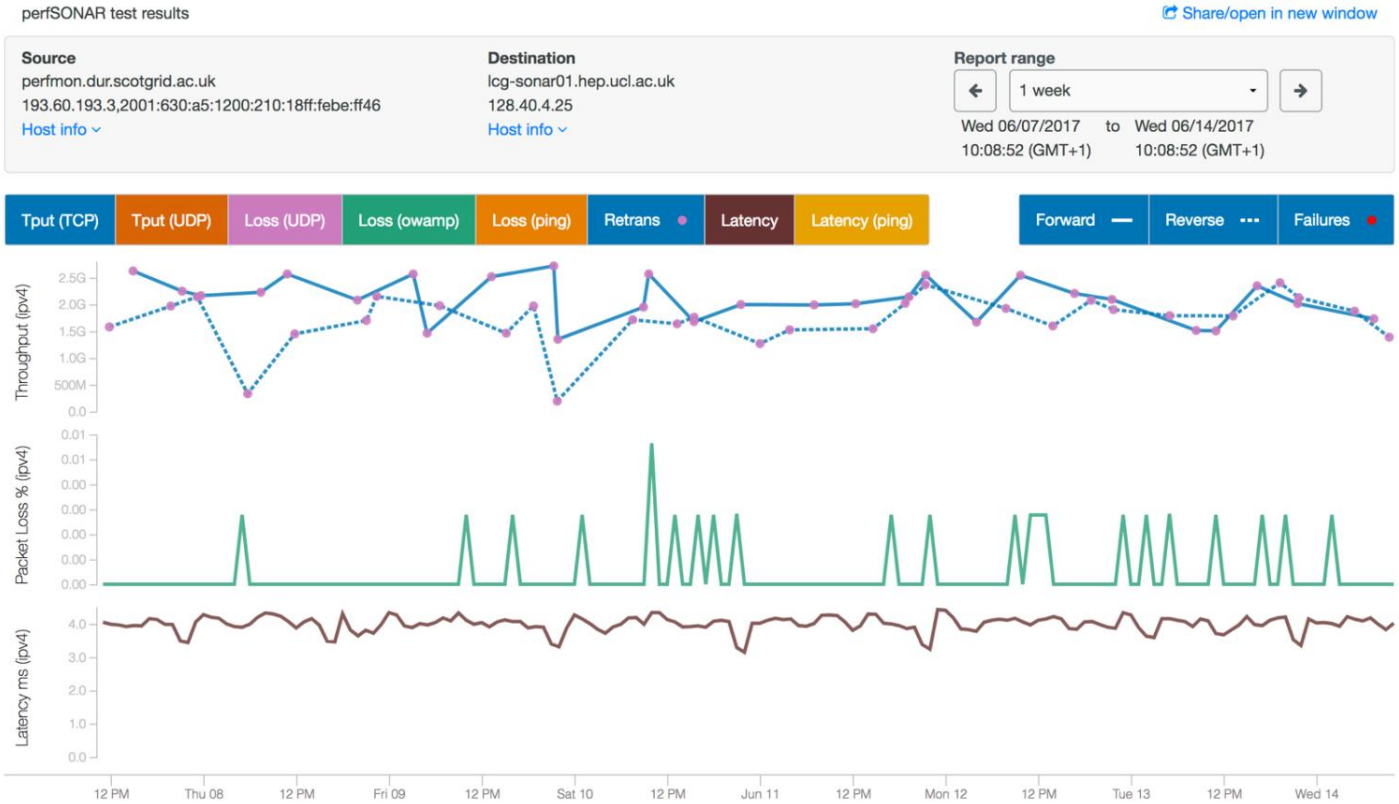
See BRIAN traffic plots here - total and LHCONE

GridPP mesh (perfSONAR 5.0, pre-Grafana)





Durham – UCL example (pre-Grafana)



perfSONAR users on Janet

`Examples of our users

Biggest example is GridPP

- The new Grafana-based mesh is being prepared
- May be hosted by WLCG and/or Jisc (on new VM platform)

Other science communities:

- SKA, Vera Rubin, …
- UK HPC facilities, HPC-SIG, …
- STFC internal deployment across multiple sites

Universities who provide Science DMZ for their science users

Small node perfSONAR use cases

- perfSONAR on a small form factor PC or RPi

Tests with HEAnet on latency driven by mutual interest in PTP

Jisc

perfSONAR latency

Examples

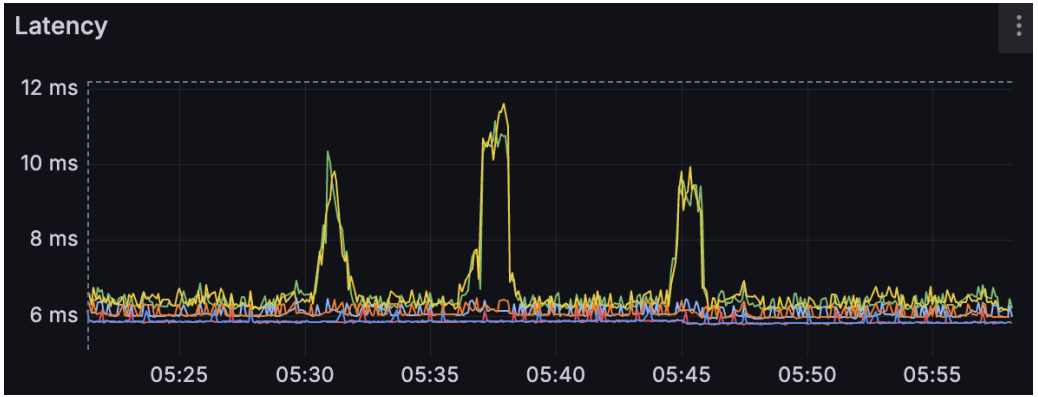
Top: HEAnet - Janet

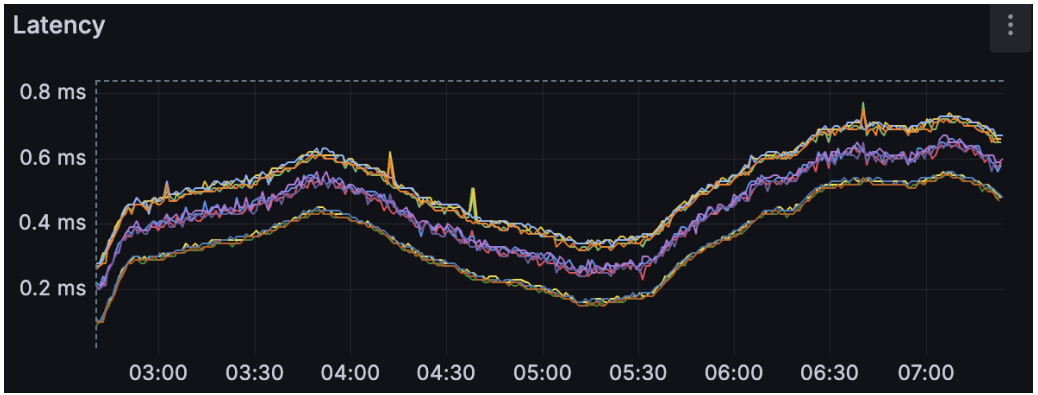
- Spikes imply queueing

Bottom: Two Janet servers (ps-london - ps-slough)

- Very low latency path
- Variation implies clock drift

With latest Linux *chronyd* and hardware time-stamping are beneficial - perfSONAR very useful to see the effect





Examples of perfSONAR value

Nuanced problem detection

Normal traffic appears fine but large science transfers performing poorly

- A campus firewall upgrade caused small packet loss

Intermittent low throughput between QUB and Hawaii

- Faulty optics on one link of a 6x100 LAG bundle impacted one in six transfers. Slow but steady increase in loss showed the degradation over time

These are examples that will not be picked up by general traffic volume monitoring - perfSONAR is able to spot them - though we could use better alarm reporting (the devs have started looking at ML/AI approaches)

Contributing to perfSONAR

Testing, feedback, continuous improvement

Jisc contributes to perfSONAR development via the GÉANT GN5-1 project, within WP6

Testing beta and new releases

Reporting issues, working with developers

Monthly calls with WLCG members (Shawn, Marian) and perfSONAR devs to discuss GridPP-related topics

Recent example - we identified a memory leak which led to perfSONAR sub-processes being killed; a fix was applied to powstream and the memory footprint is now much more stable

Jisc

Aside: Using perfSONAR (pscheduler) to tune network configuration parameters

Running 3rd party pscheduler tests, varying the tuning parameters

Helps understand how to optimise network throughput

Still need good tuning in disk I/O etc

(again, see “Science DMZ” - https://fasterdata.es.net/science-dmz/)

What parameters does *pscheduler* support?

Examples for throughput tests

CCA: --congestion – Reno, CUBIC, H-TCP, BBR, etc

MTU: --mss (actually the TCP maximum segment size)

- Many WLCG sites run 9000 MTU, but many do not

TCP window size: --window-size – important for long fat pipes

Number of streams: --parallel

Pacing: --bandwidth

You can use any combination of the above

See https://docs.perfsonar.net/pscheduler_ref_tests_tools.html

Jisc

Configuring servers for tuning tests

Must ensure *pscheduler* can use a full range of parameters

Server set open for 3rd party testing

BBRv3 installed (not necessarily as the default CCA)

9000 MTU enabled

IPv6 enabled

Enhanced window/buffer size settings by default (e.g., using settings from FasterData)

Example: cubic vs BBR (Janet -> CERN)

CUBIC

$ pscheduler task throughput --dest pse01-gva.cern.ch --source ps-london-bw.perf.ja.net --congestion cubic

* Stream ID 5

Interval	Throughput	Retransmits	Current Window
0.0 - 1.0	2.15 Gbps	1470	4.78 MBytes
1.0 - 2.0	2.36 Gbps	0	5.10 MBytes
2.0 - 3.0	2.42 Gbps	0	5.38 MBytes
3.0 - 4.0	2.68 Gbps	0	5.61 MBytes
4.0 - 5.0	2.76 Gbps	0	5.81 MBytes
5.0 - 6.0	2.78 Gbps	0	5.97 MBytes
6.0 - 7.0	2.85 Gbps	0	6.10 MBytes
7.0 - 8.0	2.24 Gbps	895	3.10 MBytes
8.0 - 9.0	1.51 Gbps	0	3.25 MBytes
9.0 - 10.0	1.57 Gbps	0	3.37 MBytes

Summary

Interval	Throughput	Retransmits	Receiver Throughput
0.0 - 10.0	2.34 Gbps	2365	2.31 Gbps

Example: CUBIC vs BBR (Janet -> CERN)

BBR

$ pscheduler task throughput --dest pse01-gva.cern.ch --source ps-london-bw.perf.ja.net --congestion bbr

* Stream ID 5

Interval	Throughput	Retransmits	Current Window
0.0 - 1.0	12.65 Gbps	0	63.88 MBytes
1.0 - 2.0	14.55 Gbps	0	63.55 MBytes
2.0 - 3.0	14.93 Gbps	0	63.88 MBytes
3.0 - 4.0	14.97 Gbps	0	64.54 MBytes
4.0 - 5.0	14.93 Gbps	0	64.47 MBytes
5.0 - 6.0	14.80 Gbps	0	63.37 MBytes
6.0 - 7.0	14.75 Gbps	0	65.12 MBytes
7.0 - 8.0	14.72 Gbps	0	63.08 MBytes
8.0 - 9.0	14.85 Gbps	0	63.23 MBytes
9.0 - 10.0	14.92 Gbps	0	63.86 MBytes

•

Summary

Interval	Throughput	Retransmits	Receiver Throughput
0.0 - 10.0	14.61 Gbps	0	14.58 Gbps

Or just change parameters and observe results…

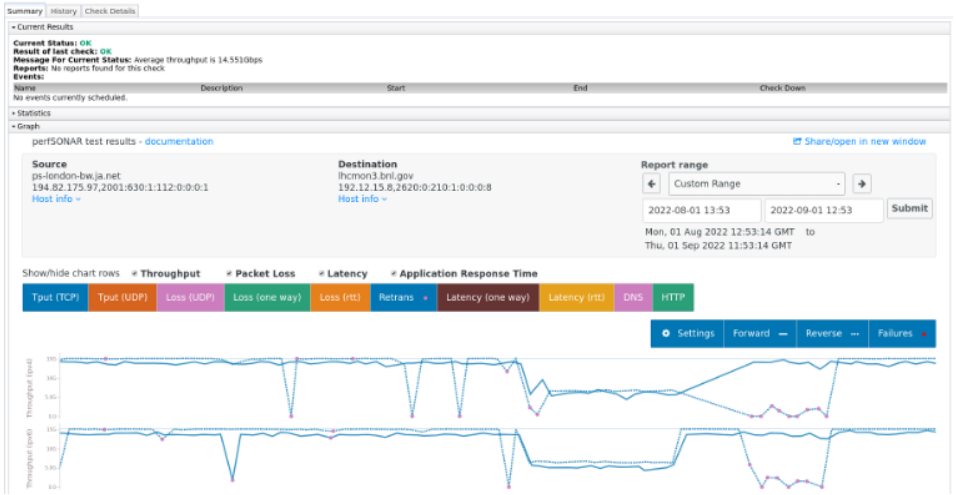
Here perfSONAR shows the impact of 1500 vs 9000 MTU

Here we changed the tuning between two perfSONAR servers and noted the plotted results over time

The MTU is dropped from 9000 to 1500 on Jisc London for tests to BNL (USA), then raised again

Throughput falls: 14Gbit/s to 6Gbit/s

The second dip on the reverse path is where we set the London pS node to default OS tuning



Summary

perfSONAR on Janet

A valuable tool to monitor network characteristics over time

- Key is to have history to study when incidents arise

Easy to install, various deployment models

Jisc is happy to support communities that want a turnkey tool

Many examples of problems being identified

Where other tools typically fail to do so

perfSONAR is just one component of performance analysis

- DC24 challenges lay in FTS scheduling and tokens
- WLCG traffic profile tends not to be big multi-Gbit/s flows

Jisc

Questions / discussion ?

Tim Chown

tim.chown@jisc.ac.uk

netperf@jisc.ac.uk

Tel 01325 822106

customerservices@jisc.ac.uk

jisc.ac.uk