

perfsONAR

perfSONAR Host Hardware

ASTRON perfSONAR training

Antoine Delvaux, PSNC, antoine.delvaux@man.poznan.pl

Szymon Trocha, PSNC, szymon.trocha@man.poznan.pl

24-26 September 2018

This document is a result of work by the perfSONAR Project (<http://www.perfsonar.net>) and is licensed under CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>).



Outline

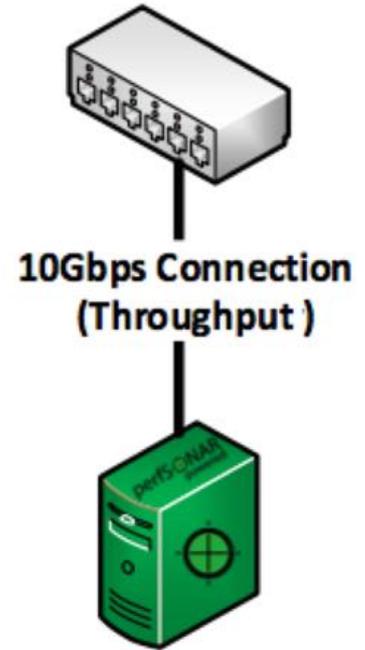
- Use Cases
- Hardware Selection
- Virtualization

Use Cases

- There are several deployment strategies for perfSONAR Hardware:
 - Bandwidth Only Testing
 - Latency Only Testing
 - Combined
 - Individual NIC for Bandwidth and Latency Testing
 - Shared NIC

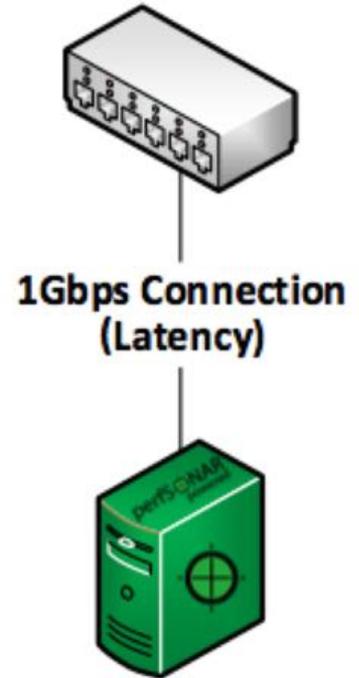
Bandwidth Use Case

- The bandwidth host is designed to saturate a network to gain a measure of achievable throughput (e.g. how much information can be sent, given current end-to-end conditions)
- Can test using TCP (will back off) or UDP (won't back off) – the end result is still the same
- Connectivity can be any size – typically you will want a host that matches the bottleneck of your network



Latency Use Case

- Tests are lightweight (e.g. smaller packets, less of them)
- Designed to measure things like jitter (variation in arrival times of data), packet loss due to congestion, and the time it takes to travel from source to destination
- Connection can be smaller – typically 100Mb or 1Gb connections will do fine. 10Gbps latency testing is not really necessary



Why Separate These?

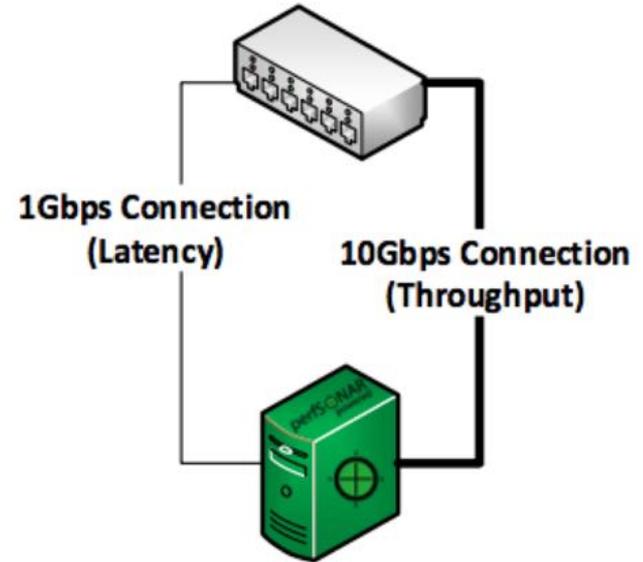
- Bandwidth testing is ‘heavy’ in that it is designed to fill the network as quickly as possible
 - E.g. the memory on the host, the queues on the NIC, the LAN, the WAN, etc.
 - Most throughput tests will cause loss, even if its temporal
- Latency testing is ‘light’ in that it wants to know if there is something that is perturbing the network
 - Congestion from other sources, a failing interface, etc.

Why Separate These?

- Because of the conflicting use case – running these at the same time is problematic
 - A heavy bandwidth test could cause loss in the latency testing.
 - This makes it challenging to figure out ‘where’ the loss is coming from; the host or the network
- If operating two machines isn’t possible, it is desirable to run these on a single host. There are two ways to do this:
 - Dual NICs
 - Single NIC, with isolated testing

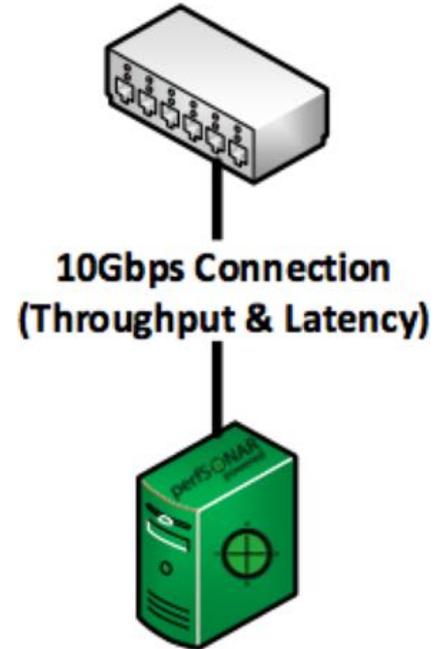
Dual NIC Testing Use Case

- Newer releases of the perfSONAR software facilitate the use of two interfaces
- Host-level routing manages the test traffic to each of the interfaces
- Bottlenecks are still possible:
 - If the host has a single CPU managing both sets of test traffic
 - If there is a memory bottleneck
 - If the NICs do not have an offload engine, they both will need to rely on the CPU to manage traffic flow internally



Single NIC/Dual Testing Use Case

- If the host has a single NIC, tests can be configured to share access
- The previous bottlenecks surrounding the NIC, CPU, and Memory are not as impactful (e.g. they will still be a problem, but impact both sets of tests equally, and one at a time)



Outline

- Use Cases
- Hardware Selection
- Virtualization

Hardware Selection

- Selecting hardware to do the job of measurement is not impossible
- Optimize for the use case of “memory to memory” testing, e.g. we don’t care about the disk subsystem
- Things that matter
 - CPU speed/number
 - Motherboard architecture
 - Memory availability
 - Peripheral interconnection
 - NIC card design + driver support

CPU/Motherboard/Memory

- Motherboard/CPU

- Intel Sandy Bridge or Ivy Bridge CPU architecture
 - Ivy Bridge is about 20% faster in practice
 - High clock rate better than high core count for measurement
 - Faster QPIC for communication between processors
- Multi-processor is waste given that cores are more and more common
- Motherboard/system possibilities:
 - SuperMicro motherboard X9DR3-F
 - Sample Dell Server (Poweredge r320-r720)
 - Sample HP Server (ProLiant DL380p gen8 High Performance model)

- Memory speed – faster is better

- We recommend at least 8GB of RAM for a test node (minimum to support the operating system and tools). More is better – especially for testing over larger distances and to multiple sites.



NIC

- There is a difference between 1G and 10G (or larger) testing
- As network speeds increase (e.g. requiring more packets to pass through interfaces per second) problems that are very nuanced become easier to see
 - Failing equipment with small ($< .01\%$) packet loss
 - CRC errors
 - Microbursts of congestion
- Consider these options when choosing a NIC speed

Small nodes

- Low-cost hardware (~200 Euro)
- Known limits
- Can become example measurement experimentation and training platform about network measurement



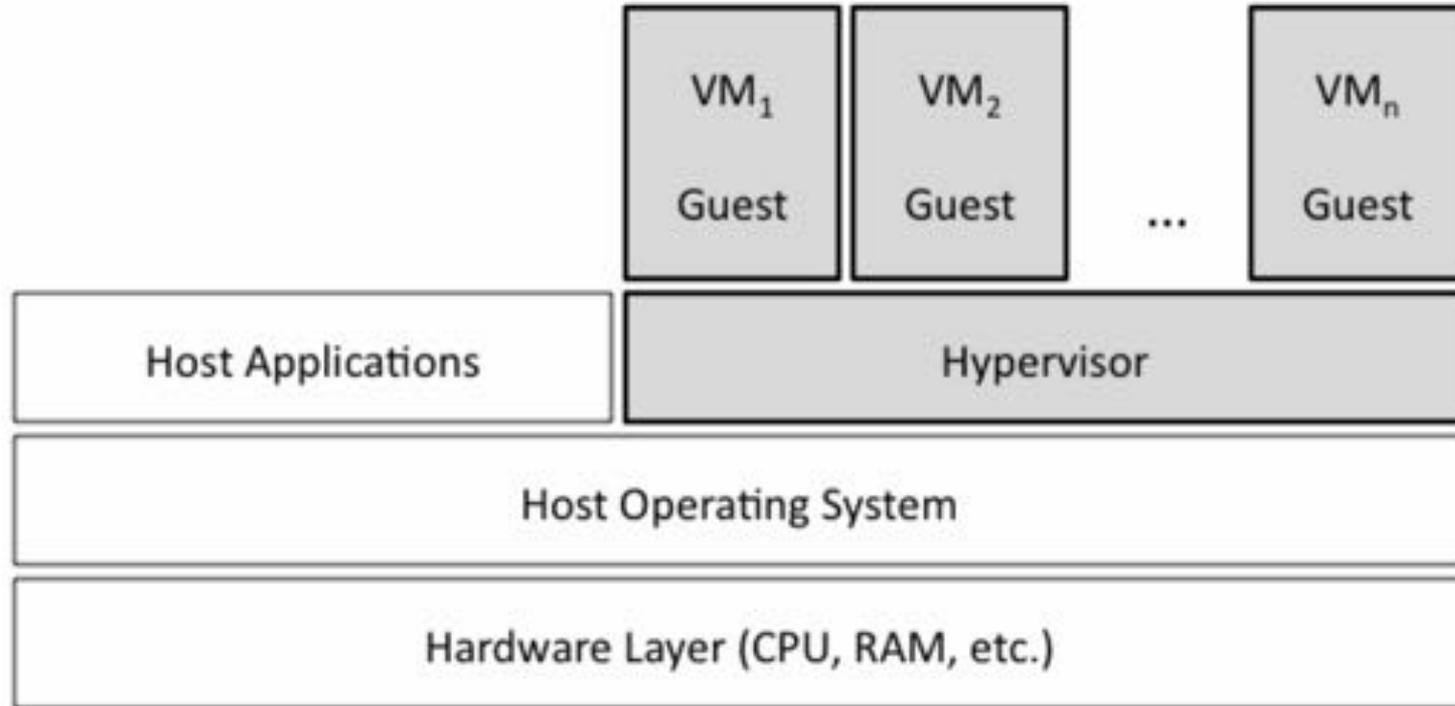
Hardware Suggestions

- The target is continually being moved and updated recommendations can be found at:
http://docs.perfsonar.net/install_hardware.html
- Additionally talk with others: perfsonar-user@internet2.edu
- Check out existing deployment examples:
http://docs.perfsonar.net/deployment_examples.html

Outline

- Use Cases
- Hardware Selection
- **Virtualization**

Virtualization Introduction



What Time is it?

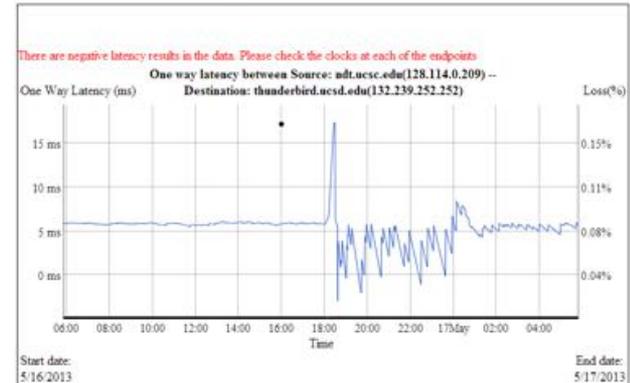
- Known complication: the ability to keep accurate time.
- perfSONAR uses NTP (network time protocol) which is designed to keep time monotonically increasing
 - Slows a fast clock, skips ahead a slow clock. Never 'reverses' time
- VM environments rely on the hypervisor to tell them what time is – this means time could skip forwards, or backwards.
 - IF NTP sees this, it turns off – this is normally catastrophic for measurement purposes (when do I start? When do I end?)
- Picture on right – jitter observed after a hypervisor adjusted the clock.

perfSONAR

 Scale Y axis from 0 Show Reverse Direction Data

Graph Key (Src-Dst)

- Max delay
- Min delay
- Loss
- Third Quartile
- Median
- First Quartile



Functionality Comparison

- Pros:

- Ability to have many ecosystems (Windows, FreeBSD, Linux, etc.) invoked through a standard management layer
- Utilize resources ‘horizontally’ on the machine. E.g. most times a server sits idle if it has no task. By stacking multiple guest machines onto a single host, the probability of the resource being better utilized increases
- Ad-hoc, short time testing

- Cons:

- Limit is reached when machines require resources beyond what is available. Can ‘plan’ for this and design the system so its underutilized, or overprovision in the hopes that there will be no conflicts
- Because this is a shared resource, it won’t do one job very well.



Consolation Prize

- Virtualization can be useful:
 - Experimental setup
 - Testing virtual environments (e.g. cloud providers)
 - Non-latency/bandwidth sensitive testing (passive monitoring, etc.)
 - Smaller performance expectation versus the network
 - E.g. if you are supporting testing for 100s of 100MB connected laptops, a 1G or 10G server in a virtual machine is far greater than the bottleneck of performance

perfsONAR

Discovering perfSONAR

ASTRON perfSONAR training

Antoine Delvaux, PSNC, antoine.delvaux@man.poznan.pl

Szymon Trocha, szymon.trocha@man.poznan.pl

24-26 September 2018

This document is a result of work by the perfSONAR Project (<http://www.perfsonar.net>) and is licensed under CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>).



Administrative information

- The system needs this so that it can be located
 - Name
 - Contact
 - Location
 - Metadata

The screenshot shows the 'Administrative Information' configuration page in the perfSONAR interface. The page is titled 'perfsONAR Toolkit on psmall-poz1.man.poznan.pl' and includes navigation links for 'View public dashboard', 'Configuration', and 'Help'. The main content area is divided into several sections:

- Administrative Information:** Contains input fields for Organization Name, Administrator Name, Administrator Email, City, Country (a dropdown menu), State/Province, ZIP/Postal Code, Latitude, and Longitude. There is also an 'Autofill Lat/Long' button and a checkbox for 'I agree to the perfsONAR Privacy Policy'.
- Metadata:** Includes a 'Node Role' dropdown menu (set to 'Public'), a 'Node Access Policy' dropdown menu (set to 'Public'), and a text area for 'Access Policy Notes'.
- Communities:** Features a text input field for 'Select communities' and a link to 'Add a community'.

On the right side, there is a 'Resources' sidebar with links for 'Editing Host Information', 'Managing Communities', and 'Privacy Policy'. At the bottom of the form, a dark bar displays the message 'You've made changes that haven't been saved.' with 'Cancel' and 'Save' buttons.

Node discovery

- Once you complete the administrative info – your host will attempt to register with the “Lookup Service”
- This is a global directory that makes it easier to find perfSONAR nodes
- If your host has the admin info present, and isn't a private IP, it will do this automatically

Finding other instances

The screenshot shows the 'perfsONAR Lookup Service Directory' interface. On the left, there is a search bar and a list of services under the 'Browser' section, including items like 'jitterbuffer Server', 'BWCTL Server', 'ONVAMP Server', 'NCF Server', 'NPAO Server', 'Ping Responder', 'Traceroute Responder', 'NA', 'BWCTL MP', 'ONVAMP MP', 'hears', 'board', and 'dashboard'. Below the browser is a 'Communities' section and a 'Developer' section. The main area on the right is titled 'Service Information' and contains a table with columns: Service Name, Addresses, Geographic Location, Communities, Version, and Custom. Below this is a 'Host Information' table with columns: Host Name, Hardware, System Info, Toolkit Version, and Communities. The bottom right section is 'Service Map', which displays a world map with numerous red dots representing service instances across various continents.

<http://stats.es.net/ServicesDirectory/>

perfsONAR

Discovering perfSONAR

ASTRON perfSONAR training

Antoine Delvaux, PSNC, antoine.delvaux@man.poznan.pl

Szymon Trocha, PSNC, szymon.trocha@man.poznan.pl

24-26 September 2018

This document is a result of work by the perfSONAR Project (<http://www.perfsonar.net>) and is licensed under CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>).

