How to Measure the Killer Microsecond

Mia Primorac Edouard Bugnion Katerina Argyraki

EPFL, Switzerland

ABSTRACT

Datacenter-networking research requires tools to both generate traffic and accurately measure latency and throughput. While hardware-based tools have long existed commercially, they are very costly and lack flexibility. The recent development of kernel-bypass networking and advanced NIC features such as hardware timestamping have created new opportunities for accurate latency measurements using software tools. We directly compare the two approaches. We use hardware devices as the ground truth. We evaluate MoonGen software that uses NIC hardware timestamping and kernel-bypass-based T-Rex software, that relies on only on the CPU to timestamp traffic, as well as socket-based netperf.

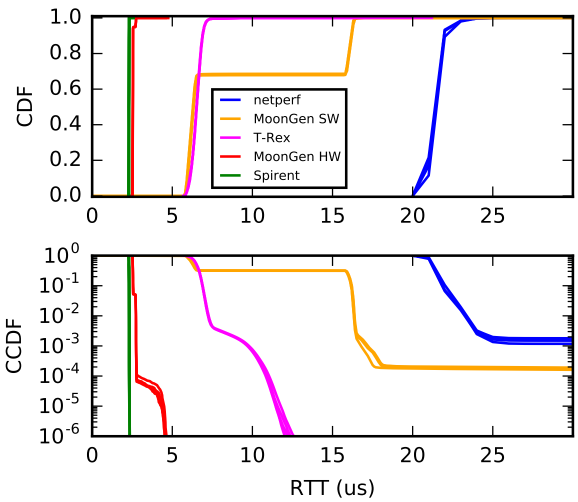

Figure 1: Latency measurements of a hardware switch using different tools.

1 METHODOLOGY AND RESULTS

Network researchers need tools to generate traffic and measure latency and throughput.

The industry has traditionally used hardware-based tools which provide accuracy, but neither flexibility nor low cost. Researchers, on the other hand, typically use software tools, which provide low cost and flexibility, but their accuracy is unclear, if not downright questionable [2].

With datacenter and cloud operators chasing the killer microsecond [1], the measurements at that scale can be trusted only if they are obtained with a tool that provides accuracy at the same scale.

We ask the following two questions: (1) *How close do state-of-the-art commodity solutions get to bridging the gap between hardware and software and providing accurate µs-scale tail latency measurements?* (2) *Are the measurements sufficiently accurate to study the latency distribution of software network functions?*

We answer the first question based on a simple observation: the latency of a constant-rate flow going through an ASIC-based switch is expected to be constant. We first use a proprietary hardware-based measuring device (Spirent) to confirm that it indeed hardly varies. We then use this measured latency as the ground truth and determine up to which percentile different software tools measure it correctly. We show our results in Figure 1.

We answer the second question by sending constant-rate flows through a software network function that simply forward packets. We use different hardware and software tools to observe the impact of operating system (OS) configurations on the network function's tail latency up to the 99.9999^{th} percentile. We quantify the mismatch between the hardware ground truth and other tools. For more details please refer to our paper [5].

We contribute the following results:

- The use of NIC-based hardware timestamps in [4] provides accurate readings up to the 99.99^{th} percentile.
- A tuned kernel-bypass-based solution such as T-Rex [3] introduces 5µs to 10µs overhead in readings.
- POSIX-based netperf should be avoided when measuring µs-scale latencies.
- Our study suggests that bidirectional hardware support is highly beneficial to accurately measure µs-scale latencies.

REFERENCES

[1] Luiz Barroso, Mike Marty, David Patterson, and Parthasarathy Ranganathan. Attack of the killer microseconds. *Commun. ACM*, 60(4):48–54, March 2017.

[2] Alessio Botta, Alberto Dainotti, and Antonio Pescapè. Do you trust your software-based traffic generator? *IEEE Communications Magazine*, 48(9):158–165, 2010.

[3] Cisco Systems. T-rex: Cisco's realistic traffic generator. https://trex-tgn.cisco.com. Accessed: 2017-03-01.

[4] Paul Emmerich, Sebastian Gallenmüller, Daniel Raumer, Florian Wohlfart, and Georg Carle. MoonGen: A Scriptable High-Speed Packet Generator. In *IMC*, pages 275–287, 2015.

[5] M. Primorac, E. Bugnion, and K. Argyraki. How to Measure the Killer Microsecond. ACM SIGCOMM KBNets'17, 2017.