Contribution ID: **66** Type: **not specified**

# Netbench –testing network devices with real-life traffic patterns

*Tuesday, 17 October 2017 14:40 (25 minutes)*

Network performance is key to the correct operation of any modern datacentre or campus infrastructure. Hence, it is crucial to ensure the devices employed in the network are carefully selected to meet the required needs.

The established benchmarking methodology [1,2] consists of various tests that create perfectly reproducible traffic patterns. This has the advantage of being able to consistently asses the performance differences between various devices, but comes at the disadvantage of always using known, pre-defined traffic patterns (frame sizes and traffic distribution) that do not stress the buffering capabilities of the devices to the same extent as real-life traffic would.

Netbench is a network-testing framework, based on commodity servers and NICs, that aims at overcoming the previously mentioned shortcoming. While not providing identical conditions for every test, netbench enables assessing the devices'behaviour when handling multiple TCP flows, which closely resembles real-life usage.

Furthermore, due to the prohibitive cost of specialized hardware equipment that implements RFC tests [1,2], few companies/organisations can afford a large scale test setup. The compromise that is often employed is to use two hardware tester ports and feed the same traffic to the device multiple times through loop-back cables (the so called "snake-test"). This test fully exercises the per-port throughput capabilities, but barely stresses the switching engine of the device in comparison to a full-mesh test [3]. The per-port price of a netbench test setup is significantly smaller than that of a testbed made using specialized hardware, especially if we take into account the fact that generic datacentre servers can be time-shared between netbench and day-to-day usage. Thus, a large-scale multi-port netbench setup is easily affordable, and enables organisations/companies to complement the snake test with benchmarks that stress test the switching fabric of network devices.

The presentation will cover the design of the netbench software platform and subsequently present results from a recent evaluation of high-end routers conducted by CERN.
Netbench has a layered architecture and uses standard technologies. At its core, it relies on iperf3 [4] as an engine to drive TCP flows between servers. The orchestration platform that sets up multiple iperf3 sessions is written in Python and relies on XML-RPC for fast provisioning of flows. Per-flow statistics are gathered into a PostgreSQL database, and the results visualisation is based on a Python REST API and a web page using JavaScript and the D3.js library for displaying graphs. Statistics are presented at different levels of detail allowing the human tester to quickly asses the overall state of a test from both per-node and per-pair (source-destination) statistics.
During its last call for tender for high-end routers, CERN has employed netbench for evaluating the behaviour of network devices when exposed to meshed TCP traffic. We will present results from several devices. Furthermore, during the evaluation it became apparent that, due to the temporary congestion caused by competing TCP flows, netbench provides a good estimation of the devices'buffering capabilities.

To summarize, we present netbench, a tool that allows provisioning TCP flows with various traffic distributions (pairs, partial and full-mesh). We consider netbench an essential complement to synthetic RFC tests [1][2], as it enables affordable, large-scale testing of network devices with traffic patterns that closely resemble real-life conditions.

[1] RFC 2544, Bradner, S. and McQuaid J., "Benchmarking Methodology for Network Interconnect Devices"
[2] RFC 2889, Mandeville, R. and Perser J., "Benchmarking Methodology for LAN Switching Devices"
[3] RFC 2285, Mandeville, R., "Benchmarking Terminology for LAN Switching Devices"
[4] iperf3 http://software.es.net/iperf/

## Desired length

20 minutes

**Primary authors:** STANCU, Stefan Nicolae (CERN); KRAJEWSKI, Adam Lukasz (CERN)

**Presenter:** STANCU, Stefan Nicolae (CERN)

**Session Classification:** Networking and security

**Track Classification:** Security & Networking