Evaluation of existing bandwidth measurement tools for large scale deployment (Abstract)

M.A. Santcroos and H.A.J.R. Uijterwaal,
RIPE NCC New Projects Group,
Singel 258,
1016 AB Amsterdam,
The Netherlands.

I. Introduction

The RIPE NCC is a not-for-profit organization providing all services that its members have to organize as a group, even though they are competitors in other areas. The RIPE NCC membership consists of approximately 3,500 ISP's and telecommunication corporations and similar organizations, in its service region of Europe, former Soviet Union, North Africa and the Middle East. Since 1999, the services offered by the RIPE NCC include the Test Traffic Measurement (TTM) service [1].

The goal of the TTM service, is to do active end-to-end performance measurements between providers. In order to accomplish this, about 100 measurement probes or "test boxes" are installed in the network of the participating organizations. A "test box" (TB) consists of a PC running the FreeBSD operating system, a GPS system to synchronize the system clock and software to measure network properties such as delay (or latency), packet-loss, routing and jitter. The TTM service follows the applicable standards set forth by the IETF [2]. TTM supports both IPv4 and IPv6.

Over the last years, the RIPE NCC has been looking into expanding the measurements that can be done with the TB's. One candidate are bandwidth measurements, both raw capacity and available bandwidth. The reason for this is simple, ISP's spend large amounts of money buying bandwidth and want to know if the actual performance is what they expect, as well as measure available bandwidth in order to plan future capacity upgrades.

II. Requirements for a bandwidth measurement tool

As there are currently many tools to measure both capacity and available bandwidth on the market, we decided not to develop yet another implementation but rather evaluate existing tools and then add the most suitable tool to the TTM measurement program. This paper discusses the evaluation of existing tools inside our measurement network.

In order for a tool to be deployed in a large measurement network, it has to meet several requirements:

- First of all, it should produce reliable and consistent results. With a possible 100 active TB's and thus some 10,000 possible paths, it is impossible to calibrate or/and verify every single measurement.
- As measurements are done on production networks, the tool should be based on non-intrusive methods only.
- The bandwidth measurement tool should not interfere with other measurements on the TB.
- The tool should produce a result quickly. Most tools do not allow to measure bandwidth to 2 or more hosts in parallel. If one then wants to measure the available bandwidth to every possible destination, say, 3 times a day, one has $(24\times 60)/(3\times 100)$ or only 4 minutes for each measurement.

III. Results

For our evaluation, we have installed the following tools:

- Pathrate, a packet dispersion tool for measuring bottleneck capacity [5].
- Pathload, a tool for measuring bottleneck available bandwidth using self loading periodic stream [4].
- Pathchirp, another tool for measuring bottleneck available bandwidth using exponentially spaced packets [6].

We evaluated these tools in the following test setups:

- Two PC's, connected back-to-back with 100 Mbps Ethernet.
- Two PC's, connected through 3 routers with 10 Mbps in a test-network isolated from the regular Internet.
- Two PC's on the DataTAG network [8]. In this network, the make and model of all equipment is known.

For all tools and all networks, we find that the measured capacity is too low. For example, measuring the capacity of the first network several times in a row, we find results ranging from 87 to 97 Mbps for pathrate, where in these ideal circumstances, one would expect a number very close to 100 Mpbs.

Measuring the available bandwidth in two directions several times in a row, produces results that differ by as much as 25 % for pathchirp and 5 % for pathload. Also the capacity reported by pathrate differed by 5 % in both directions.

We then investigate the source of these differences:

IV. Hardware effects

The two PC's used for this experiment had different processor speeds (333 and 466 MHz respectively) as well as other small differences in the hardware. This might explain some of the differences. However, in a network of 100 or more measurement stations, it is impossible to guarantee that any 2 will be exactly the same and if small differences like this have such a large effect on the results, this makes the tools unsuitable for our application. In order to better understand the effects of the hardware, we will repeat these experiments with other machines and report on the results at the workshop.

V. Time stamping the packets

The algorithms used by the tools mentioned above all rely on time stamping the packets and are sensitive to small errors in those time stamps. As all tools run in user-space, time stamps are set at that level, before the packets are passed to or received from the operating system.

In another study [7], we tagged the Ethernet packets in the first setup (2 PC's, back to back) on the wire with a DAG-card [3]. These timestamps were compared against the timestamps set by the PC. This study showed both a constant offset as well as some jitter between the two.

The offset is caused by the processing overhead of the network stack, the jitter by other processes on the machines using CPU cycles.

Both can be addressed by moving the time-stamping of the packets to the kernel level of the operating system, where it can be less interrupted by other tasks. This will make the tools operating system dependent though. Most of these changes can be applied to any of the tools that we tested. Another approach would be would be to use Berkeley Packet Filter (BPF). Using the BPF one is able to do kernel level timestamps at the receiving end, while still being reasonably portable.

This is work in progress, in the presentation we will show more details. Other sources of errors will also be discussed in the presentation.

VI. Conclusion

Based on our measurements, we conclude that none of the evaluated tools is suitable for deployment on a large measurement network *at this time*. Reasons for this are running time and accuracy.

We will show how the tools can be modified in order to make the results more accurate and reliable, at the expense of making them more operating system dependent. We will also show how timestamps at various levels in the stack have various levels of improvement. People can then make the tradeoff themselves.

By increasing the accuracy of the timestamps we also hope to decrease the running time as the variance in the measurements will decrease and will make the statistics easier.

References

[1] New Measurements with the RIPE NCC Test Traffic Measurements Setup, M.Alves et al., In Proceedings of the Passive and Active Measurements Workshop 2002

[2] Framework for IP Performance Metrics, `http://www.ietf.org/rfc/rfc2330.txt`, V. Paxson, G. Almes, J. Mahdavi and M. Mathis

[3] DAG project, CS Dept., University of Waikato, New Zealand, `http://dag.cs.waikato.ac.nz/`

[4] Pathload: A measurement tool for end-to-end available bandwidth, Manish Jain and Constantinos Dovrolis, Proceedings of Passive and Active Measurements Workshop 2002

[5] Packet Dispersion Techniques and Capacity Estimation, Constantinos Dovrolis, Parameswaran Ramanathan and David Moore, IEEE/ACM Transactions in Networking, April 2003

[6] pathChirp: Efficient Available Bandwidth Estimation for Network Paths, Vinay Ribeiro, Rudolf Riedi, Richard Baraniuk, Jiri Navratil and Les Cottrell, Proceedings of Passive and Active Measurements Workshop 2003

[7] Analysis of delays between two test boxes, unpublished, TU Delft, RIPE NCC, October 2003

[8] The DataTAG project, see `http://datatag.web.cern.ch/datatag/`.