

NeTraWeb

- A Web-Based Traffic Flow Performance Meter

Magnus Brenning¹, Björn Olander¹, Ibrahim Orhan¹, Johan Wennberg¹, Thomas Lindh^{1,2}

¹School of Technology and Health, ²Laboratory for communication networks, School of Electrical Engineering KTH, Stockholm.

Abstract

This paper presents a web-based traffic flow performance meter. The NeTraWeb tool configures and automates the measurement activities, including storage and presentation of the main performance parameters.

1. Introduction

The tool presented in this paper, NeTraWeb, is a web-based traffic flow performance meter system. The measurement activities are configured, controlled and automated by a management server. The main traffic flow performance parameters e.g. packet loss, delays, delay variations and throughput, are presented in graphs and text. The name NeTraWeb is inherited from an extended version of the flow meter NeTraMet ([1], [2]).

The tool can typically be used by end users between hosts or by a network operator between measurement points in the network. Performance monitoring of delay and loss sensitive applications such as voice and multimedia interactive services between the end users or other measurements points in the network is the main application of the tool. The meter agents have also been used in a prototype implementation for monitoring of SIP-based communication, where the approach is to activate performance measurements between clients based on signalling messages to a SIP server [3].

Figure 1 shows a NeTraWeb server and meter agents in two access nodes. The meters could also be deployed in end user hosts or other points in the network where traffic can be observed, e.g. VPN access points or edge nodes.

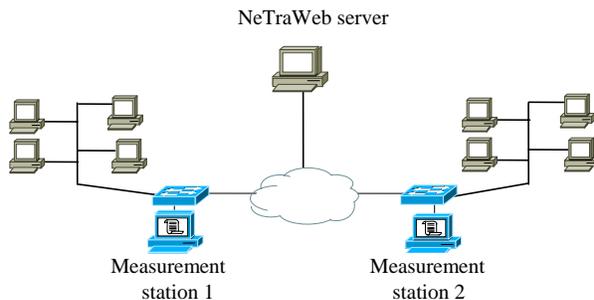


Figure 1. A NeTraWeb server and meter agents at two access points.

The method used by NeTraWeb is explained in Section 2. The functions and structure of the tool are described in Section 3, followed by some measurement results in Section 4 and the conclusions in Section 5.

2. Traffic Meters and Monitoring Packets

2.1. Combining active and passive methods

NeTraWeb uses light-weight traffic meters at the measurement points. An extended version of the traffic flow meter NeTraMet combines active and passive methods [1]. A traditional flow meter counts bytes and packets at a single point. However, to enable packet loss and delay metrics, NeTraWeb uses meters at two points that count the number of packets and bytes for the defined flows observed by the meters.

Monitoring packets are sent between the meters with periodic or random time intervals inserted between block of user packets (Figure 2). These packets can also be transmitted with a frequency that is adapted to the actual data rate of the monitored traffic flow, and thereby maintaining an approximately constant size of the monitoring block (Figure 2). When a meter detects a monitoring packet a timestamp and the current cumulative counter values of packets and bytes are stored in a flow file by each meter.

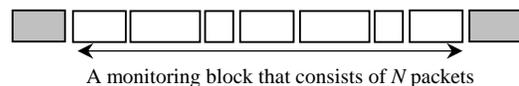


Figure 2. Two monitoring packets enclose a monitoring block that consists of N user packets on average.

The precision, accuracy and resolution of the metrics are determined by the size of the monitoring block, which is the distance between two consecutive monitoring packets (expressed in time intervals or in number of data packets). This is the main parameter of the monitoring method used to adjust the accuracy of the results.

The meter is controlled by a set of rules, using the simple ruleset language (srl), which define the flows and their attributes. A special companion program, NeMaC [4], manages the meter locally. The resulting flow files are then sent to the NeTraWeb server using an automated ssh function in Perl (Section 3).

The modifications to NeTraMet are included in the NeTraMet distribution from version 4.5b8 and later on. We have used version 5.1b11, in which round-trip time as well as one-way delays can be measured [3].

2.2. Measurement results

The method gives the following results per traffic flow (defined as a unique combination of destination and source addresses, destination and source ports, transport protocol and possibly other parameters).

Packet delays and their variation are estimated based on sampling. One-way delay can be measured if the meters have synchronized clocks; otherwise *round-trip delay* and its variation are measured. *Inter-arrival jitter* in both directions can reveal possible asymmetry in delays. The four timestamps used in the round-trip delay estimates are shown in Figure 3.

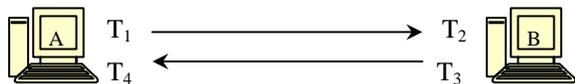


Figure 3. The four timestamps recorded for the monitoring packets.

Round-trip delay is here defined as $T_4 - T_1 - (T_3 - T_2)$ and the inter-arrival jitter (at node B in Figure 3) as $T_2(n+1) - T_2(n) - (T_1(n+1) - T_1(n))$. The definition of inter-arrival jitter is in agreement with RTP/RTCP Internet standards used VoIP measurements [5]. In this case the monitoring packet client in node A sends a packet (of same type and size as the data flow) to node B, which returns the packet to node A. These metrics do not require synchronized clocks. Since GPS synchronization is expensive and sometimes difficult to install, round-trip delay and inter-arrival jitter estimates are an alternative.

The monitoring packets have a dual role. Firstly to provide samples of delays based on the timestamps in Figure 3, and secondly to trigger intermediate readings of the meters, and store these cumulative counter values of packets and bytes for the defined flows.

Packet loss is measured directly. Beside the long-term averages for the entire measurement periods, the distribution of losses per monitoring block is obtained. The *length of loss periods and loss-free periods* can be computed.

Throughput is measured directly. The utilized capacity for the entire measurement period and per monitoring block is obtained.

3. The NeTraWeb System

The system consists of the NeTraWeb server and two meter agents that reside on existing hosts or on dedicated measurement nodes. The *server* manages and automates the measurement activities based on the

information an operator provides via the web user interface.

- Attributes of the flow to be monitored. The flow is specified using a proper combination of the five attributes: source and destination IP addresses, source and destination ports and layer-4 protocol. The granularity of the flow and sub flows is determined by the number of attributes and their combination.
- The length of the measurement period.
- The rate of the monitoring packets generated between the meter agents: constant time intervals between the packets or a rate that is adapted to the data rate (throughput) of the monitored flows. Monitoring packets generated based on the actual data flow rate can be seen as an adaptive sampling design. This is done by polling the meter MIB for information on the number of user data packets sent in the defined flow [6]. Further issues on systematic and random sampling are discussed in [7].

The *meter agent* is implemented in a Perl program that communicates with the NeTraWeb server on the one hand, and with the local NeTraMet meter and NeMAC on the other hand. The NeTraMet meter uses libpcap to access the network interface.

Based on the information above NeTraWeb automates the measurement activities. This communication scheme between the server and the meter agents is described below.

3.1. The automated measurement activities

The NeTraWeb server creates a configuration file based on user input and sends it to the meter agent in station 1 (Figure 1), using an automated ssh module in Perl. The meter agent in station 1 uses the configuration file to start the NeTraMet meter, the meter manager NeMaC and the monitoring packet generator. Thereafter station 1 sends the configuration file to station 2, which repeats the same procedure as station 1, and sends an acknowledgement to station 1. Both meters and the monitoring packet generators are now active. When the measurement period has ended the meters and monitoring packet generators terminate, and the two resulting flow files are sent to the NeTraWeb server. Further details can be found in [6].

The flow file from meter 1 contains a list of measurement data store by the meter when a monitoring packet is detected: the monitoring packet ID, the current timestamp t (T_1 in Figure 3), the cumulative number of packets and bytes sent and received at time t , and the corresponding measurement data when the monitoring packet returns (T_4 in Figure 3). The flow

file from meter 2 consists of measurement data when timestamps T_2 and T_3 are recorded.

The flow files are parsed and stored in different tables in a Postgres DBMS (DataBase Management System). This is done with efficiency and scalability in mind. The DBMS makes it easier to handle multiple simultaneous users and presentations.

The parser is written in ANSI-C with the ecpg library for PostgreSQL support. This allows SQL syntax inside the C code. All web material is written in Hyper Text Markup Language, PHP (PHP: Hypertext Pre-processor) and a few Javascripts.

When the flow files have been imported from the meters to the NeTraWeb system, a user creates a new project and can choose among presentations of the following groups of performance parameters: packet loss, packet round-trip delays, round-trip delay variation, inter-arrival jitter and throughput. It is also possible to import other existing flow files from other sources to NeTraWeb for presentation.

There are two types of graphs; regular time plots and histograms. NeTraWeb uses the Chart PHP4 library to generate graphs. It is covered by GNU GPL (General Public License). The graphs are generated as .PNG pictures. One can change the number of bars and the space between the bars in all histograms. Beside the graphs the most common statistical metric are calculated.

Since the two meters do not start at exactly the same time the flow files have to be synchronized. The useful common part of the flow files are picked out using SQL queries.

The processing and presentation of performance parameter metrics are implemented as SQL queries. This flexible approach means that users can add new metrics, graphs and histograms, or remove existing ones. Further details are found in [8].

4. Measurement Results

For each measurement project NeTraWeb generates more than twenty different graphs. As an illustration some graphs from different measurements are shown in the Figure 4-10.

Some results from a VoIP call between two wireless clients in a local area network are shown in Figure 4-8. The average round-trip delay is 0.62 ms, the maximum value 1.18 ms, the minimum value 0.2 ms, the 99th percentile 1.11 ms, the 97.5th percentile 1.06 ms, the 10th percentile 0.35 ms and the standard deviation 0.22 ms. Figure 4 and Figure 5 show the time plot and distribution of round-trip delays.

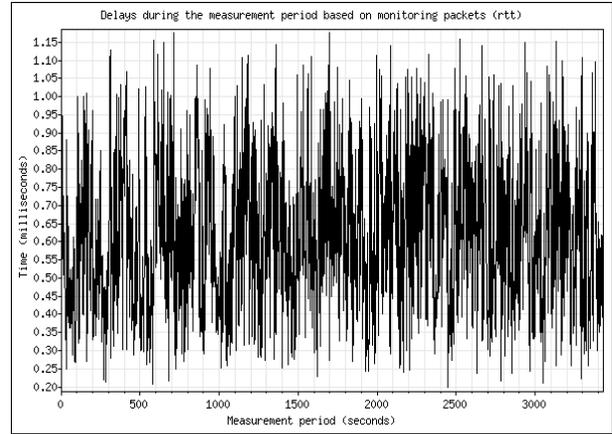


Figure 4. Round-trip delays (in milliseconds) for each monitoring packet during the measurement period.

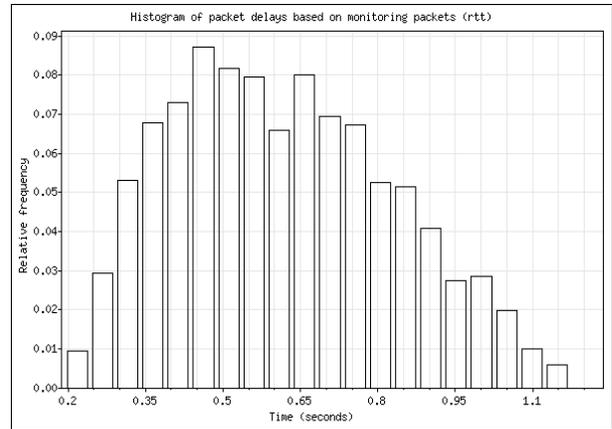


Figure 5. Histogram of the round-trip packet delays in Figure 4.

Figure 6 shows the packet loss ratio per monitoring block. The length of loss-free periods and the corresponding histogram can be seen in Figure 7 and Figure 8. The total loss ratio in this measurement is 0.014.

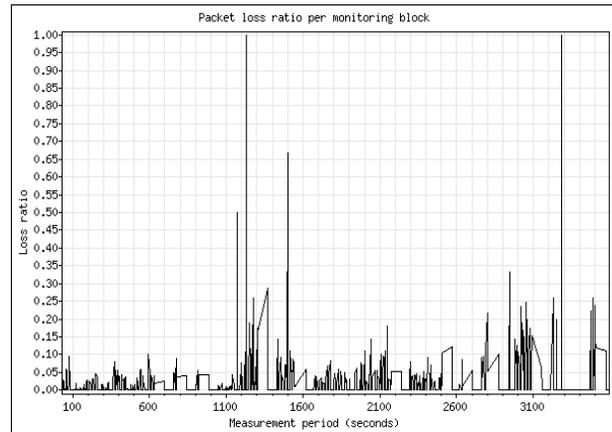


Figure 6. Packet loss ratio per monitoring block during the measurement period.

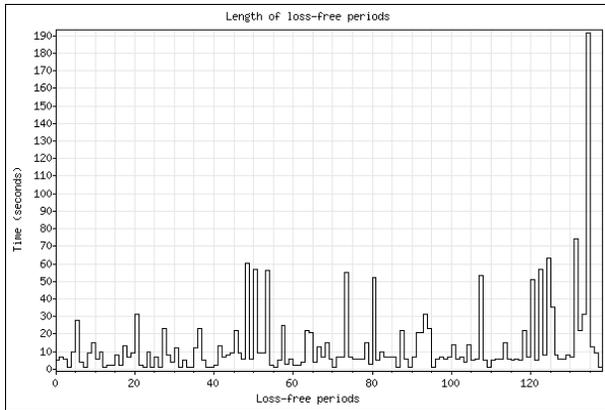


Figure 7. The length of loss-free periods in seconds.

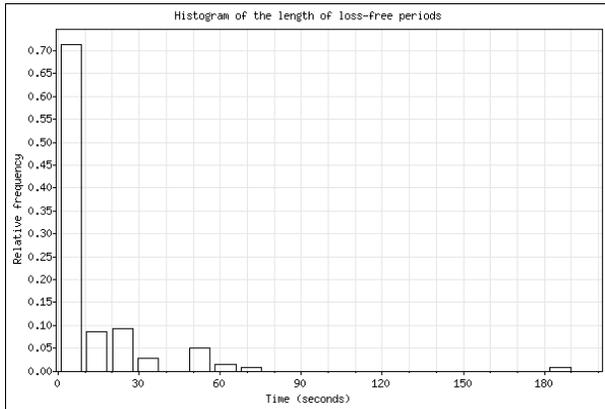


Figure 8. A histogram of the length of loss-free periods shown in Figure 7.

In addition to the graphs presented above, NeTraWeb creates graphs and histograms that cover: round-trip jitter (according to IETF and ITU-T), packet loss (in number of bytes, packets and loss ratio) per monitoring block or in time, the number of losses in loss periods, the length of loss and loss-free periods and throughput.

Inter-arrival jitter from a measurement between VoIP clients located at KTH in Haninge and BTH in Karlskrona can be seen in Figure 9 and Figure 10. The inter-arrival jitter is not symmetric. The variation in arrival times to client B is higher than in the opposite direction. Hence, asymmetric delay variations can be detected without having synchronized clocks using the four timestamps in Figure 3.

5. Conclusions

This paper has presented an open source web-based traffic flow performance meter to be used by end users or network operators. The NeTraWeb system uses a communication procedure between the server and the agents in order to automate the measurement activities. The results are stored in a database. The processing and presentation of performance parameter metrics are

implemented as SQL queries. This flexible approach means that it is easy to develop this tool further and adapt it and the users' requirements. A detailed evaluation of the tool will be part of future work.

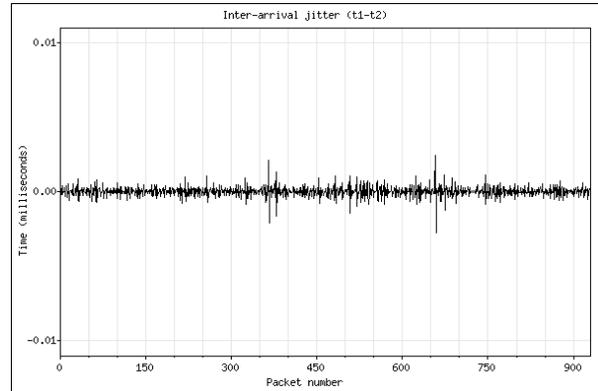


Figure 9. Inter-arrival jitter in the direction from client A to client B. The maximum value is 2.5 ms and the standard deviation is 0.36.

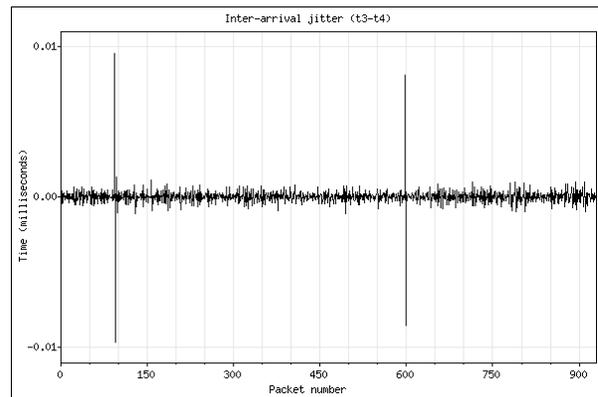


Figure 10. Inter-arrival jitter in the direction from client B to client A. The maximum value is 10.5 ms and the standard deviation is 0.83.

6. References

- [1] T. Lindh and N. Brownlee: "Integrating Active Methods and Flow Meters - an implementation using NeTraMet", Passive and Active Measurement workshop (PAM2003), San Diego, April 2003.
- [2] NeTraMet - a Network Traffic Flow Measurement Tool , <http://www.caida.org/tools/measurement/netramet/>
- [3] NeTraMet beta versions: <ftp://ftp.auckland.ac.nz/pub/iawg/NeTraMet/beta-versions/>
- [4] NeTraMet & NeMaC Reference Manual v4.3, <http://www2.auckland.ac.nz/net/Accounting/ntmref.pdf>
- [5] RTP: A Transport Protocol for Real-Time Applications, RFC 3550, July 2003.
- [6] M. Brenning, I. Orhan: "Performance measurements in IP networks: automation and rate adapted sampling", Master Thesis, Master program Computer networks, School of Technology and Health, KTH, Stockholm, September 2006.
- [7] T. Lindh: "Systematic sampling and cluster sampling of packet delays", Passive and Active Measurement workshop (PAM2006), Adelaide, March 2006.
- [8] B. Olander, J. Wennberg: "NeTraWeb Management System for Measurements of Performance Parameters", Master Thesis, Master program Computer networks, School of Technology and Health, KTH, Stockholm, September 2006.