What we have learnt from developing and running ABwE

Jiri Navratil and Les Cottrell Stanford Linear Accelerator Center (SLAC)

The Internet is not one compact, centrally managed network. It is composed of many large network backbones operated by different Internet Service Providers (ISPs). Even though most large ISPs create their own monitoring systems that provide basic information for users, a casual scientific user has little chance to effectively use these results. Such publicly accessible systems contain mostly the traffic statistics for one ISP, have limited granularity, do not include neighborhoods and the edge connections, and typically the user has very limited time to study traffic diagrams and graphs. There is also no single source today that could tell the users if an end-to-end connection from A to B is reliable and suitably fast enough for a user's application. For example the connection from SLAC to the University of Manchester (UK) goes via 4 big ISPs (CALREN, Abilene, GEANT, JANet and Net North West), in addition to crossing the SLAC, the Stanford University and the Manchester University networks.

How can the user thus recognize, whether the path for an intended communication has enough bandwidth. In the past the users were content to use the Internet "ping" tool to see how long it took to transfer packets from A to B and packet loss rates. Nowadays, when users want to run data intensive applications that must transfer many GBytes of data before start processing or to run graphical visualizations with 3D rendering on remote data, they often first run iperf to evaluate the achievable throughput. This is typically the simplest practical way for users to relatively easily estimate if they can run such applications or not. However, iperf is a tool that can load the network with Gigabytes of dummy data. Network capacities have increased dramatically in the last few years. Many universities and scientific networks operate with Gigabit backbones. At these high speeds, using iperf is becoming increasingly problematic. Development of more effective tools that estimate the available bandwidth or an achievable throughput has become a hot subject of networking research in several projects.

One of the main tasks of SLAC's Computing Services is to provide physicists with the a reliable, predictable, high performance network infrastructure. An integral part of this is our monitoring activity. It provides information about the current status of the network from SLAC to many places around the world, and we have several monitoring systems in operation today. Typically users want to know the Round Trip Time, and the achievable throughput/bandwidth to partner sites. They also need to know whether the actually measured value is normal or not, and if this situation is likely to be stable for several hours. This requires our monitoring system to have an archive and be able to provide short and longer-term predictions. Using iperf and other network intensive tools (such as the bbcp, and GridFTP file transfer tools), we have such information, but only in limited sampling periods (with measurements being made at 60 - 90 minutes intervals). The reason for this limitation is evident – load/cost of the measurements with these tools.

Two years ago, there were no tools for bandwidth estimation that could be used in a continuous mode day-in, day-out and could quickly (within a minute) detect and report unusual changes in the available bandwidth or capacities of our connections. Besides detecting capability, such a tool must be fast, manageable and minimally intrusive on the network, since we want to use it on many paths simultaneously, and it must not overload

the monitoring host's entry point or any of our paths. In April 2002, we decided to create a new tool for quickly measuring bandwidth based on packet dispersion techniques. Packet-dispersion techniques have been described in many papers and recently there are several groups doing active research in this field. Each group is trying to find ways to create improved configurations of probing packets to gain better information for the analysis. Our method is based on the simplest way of probing. We are using only packet pairs with a fixed size and no (or very small) delay between packet pairs. We send several (typically 20) probes to one destination before evaluation. The evaluation of the observed packet pair delays is based on detailed technical analysis of the problems that we can expect to meet in the routers, queuing theory, and many experimental observations that we made during the development phase. Technical details have been described in the paper "ABwE: A Practical Approach to Available Bandwidth Estimation" presented on PAM'2003, April 6-8.2003, La Jolla, Ca.

We regularly compare our results with IEPM results (achievable throughput) measured using iperf/TCP with multiple parallel streams and we have very good agreement in 60% of our 30 testing paths. We are studying the remaining 40% from both sides and trying to unearth the causes for the poor matches. Some of these analysis and comparisons between Iperf and ABwE will be shown. During summer 2003, we tested ABwE against tools such as the Pathchirp method being developed at Rice University and Pathload and found a very good match. We will present our results from these measurements on an experimental path between SLAC and Chicago that consists from several Gigabit and OC-12 segments (to CERNs machines located at Starlight). During summer 2003, we started testing on the CAIDA Gigabit-testbed with the Smartbit cross-traffic simulator. Unfortunately, these results are not very convincing. Our current understanding is that the test bed is very specific and doesn't represent real Internet path with heavily aggregated traffic. The ABwE sends only few probing packets and so if the cross traffic is very low ABwE doesn't detect the changes. We will present some analysis concerning these problems and our new solution for elimination this problem.

Until now, we have mainly described ABwE as a bandwidth estimator. However, ABwE can be used in other directions. It gives very good results in the detection of anomalies and changes in network performance and so can be used very effectively for network troubleshooting or in an automated warning system. We can detect: automatic rerouting when some lines are broken; line upgrades or degradation of the bandwidth in distant segments of the Internet; and abnormal traffic on the paths. Also these examples will be presented and explained during our presentation.

The problem of measuring the time dispersions of incoming packets is not simple and each methods behaves differently in extreme situations. However, there are several common features and dangers valid for all these methods. If the measurement machine is not fast enough or an internal communication between NIC and operating system is under coalescent control then these methods fail. Another serious problem which occurs on some paths is packet reordering. Another problem is that high speed routers that use different forwarding strategies (not only FIFO) more often. Such routers also cause a high burstiness of a traffic as the result of concentration traffic from many lower speeds inputs to higher speed Gigabit output.