LACHESIS: A Tool for Benchmarking Internet Service Providers

Jeff Sedayao & Kotaro Akita – Intel Corporation

ABSTRACT

Internet access is increasingly critical to organizations and individuals [1]. With the current boom in Internet Service Providers (ISPs), how does one judge one vendor from another? LACHESIS[1] is a tool that provides a way to benchmark ISPs. LACHESIS takes a list of prominent Internet "Landmarks" and determines the packet loss and network latency involved in reaching those landmarks. Throughput was rejected as a factor. Several studies indicate that network latency is a critical factor in World Wide Web (WWW) performance [2-5]. The default set of LACHESIS landmarks (landmarks used are customizable) includes the Domain Name Service (DNS) root servers, well known FTP servers, and popular WWW servers. LACHESIS is implemented as a PERL script wrapped around FPING. The LACHESIS tool encourages ISPs to have good interconnectivity with other ISPs. It also encourages ISPs to have plenty of capacity and not to drop packets. LACHESIS has the potential to swamp landmarks with ICMP packets (used by FPING), but this can be dealt with by filtering out ICMP from abusive hosts. ISPs can cheat by favoring ICMP packets. Future plans include a Winsock implementation so that individual SLIP/ PPP Internet subscribers can run their own benchmarks.

Introduction

Internet access is becoming more and more critical to more and more organizations and individuals. Ignoring events on the Internet can have serious consequences [1]. There is a boom in Internet service providers (ISPs), ranging from local phone companies (e.g., Pacific Bell or Ameritech), long distance companies (e.g., AT&T, MCI, and Sprint) or On-line services companies (e.g., Compuserve, Prodigy, and America On-line), and start-ups (e.g., Internex, PSI, and UUNET). But how does one judge one ISP from another? What metrics does one use? What tools are available for measuring "performance" from one another? LACHESIS is a tool that provides benchmarks for Internet service.

The first part of this paper describes the LACHESIS approach. The next section discusses how LACHESIS is implemented. Actual results are listed after that. This section focuses on the experiences and implications of LACHESIS. The paper concludes with a discussion of future work, and information on how to get LACHESIS.

The Lachesis Approach

LACHESIS's purpose is to measure the performance of an Internet Service Provider. Performance can mean many things. The time to transfer a file is one measure, while the "responsiveness" of an interactive remote login session is another. The time to call up a Web page is still another.

LACHESIS concentrates on two aspects of Internet performance – packet loss and network delay. If an ISP drops many packets, it will clearly take longer to transmit data or do remote operations because packets need to be retransmitted. Network delay is the time it takes for packets to go through a network. Studies show that World Wide Web traffic is particularly sensitive to delay [2-5]. Domain Name Service (DNS), a service critical to Internet applications, can also be negatively impacted by network latency. Many applications simply idle while waiting for DNS information. Adding network delay to DNS query times only makes things worse.

Why not concentrate on throughput? There are a number of reasons that packet loss and network delay are more critical than throughput. First, throughput will have an absolute upper limit determined by the size of the connection. Having a T3 (45 Megabit) Internet connection will yield vastly different results than from having a 14.4 Kilobit SLIP connection. Second, measuring throughput requires that you have a significant amount of data to move to or from some Internet system. It is not always possible to have this. Third, as mentioned above, applications like WWW are very sensitive to delay and do not use all of the available bandwidth.

[1] In Greek Mythology, LACHESIS was one of the Fates, the three goddesses who determine the string of life. KLOTHOS spun the string of life, LACHESIS measured it, and ATROPOS cut it. [12]

In the long run, as larger and larger amounts of bandwidth to the Internet become cheaper and cheaper, the costs of network delays become higher and higher. Consider the cost of 1 second of network delay. A 45 Megabit T3 connection will waste more potential bandwidth waiting for 1 second delay than will a 14.4 Kilobit connection. Protocols that do format and parameter negotiation are particularly vulnerable to network delays.

When measuring packet loss, we need some targets to measure packet loss. LACHESIS uses the concept of LANDMARKS. Landmarks are notable sites on the Internet. The default landmarks for LACHESIS are the root name servers, popular FTP servers, and popular WWW sites. There are landmarks from around the world to get a more complete picture of an ISP's connectivity. LACHESIS users can configure their own landmarks depending on their own usage patterns. This way, Internet users or organizations can pick an Internet vendor optimized to their particular usage patterns.

LACHESIS is implemented as a PERL [6] script wrapped around a modified version of Stanford University's FPING program. Packet loss and packet round trip times are generated from FPING. FPING uses the ICMP echo [7] to measure network latency. It has been pointed out that PING was not designed for measuring network performance and that different routers may handle ICMP packets in different ways [8]. While this is true, no other metric or protocol feature works with generic landmarks picked by a consumer.

How do we intend for LACHESIS to be used? We envision that organizations with current Internet access could run LACHESIS periodically against their favorite LANDMARKS. Data from these runs could be used to identify problem periods and get a feel for general performance through an Internet vendor. We also envision that when an organization is looking to procure Internet access, they could run LACHESIS from either the ISPs' local pops or at one of the ISPs' local customers. They could then evaluate the ISP from the resulting LACHESIS runs.

LACHESIS Implementation Notes

As mentioned above, LACHESIS is a PERL script wrapped around Stanford University's FPING program. LACHESIS is run periodically, and the packet loss, delay, and other statistics are logged, accompanied with a time stamp. To get a graphical representation of the data obtained by LACHESIS, a separate program was written to transform the data into World Wide Web [9] viewable graphs. The program GRAPHLACHESIS takes in the LACHESIS log file, parses the data, and calls upon another program which produces graphs. People can

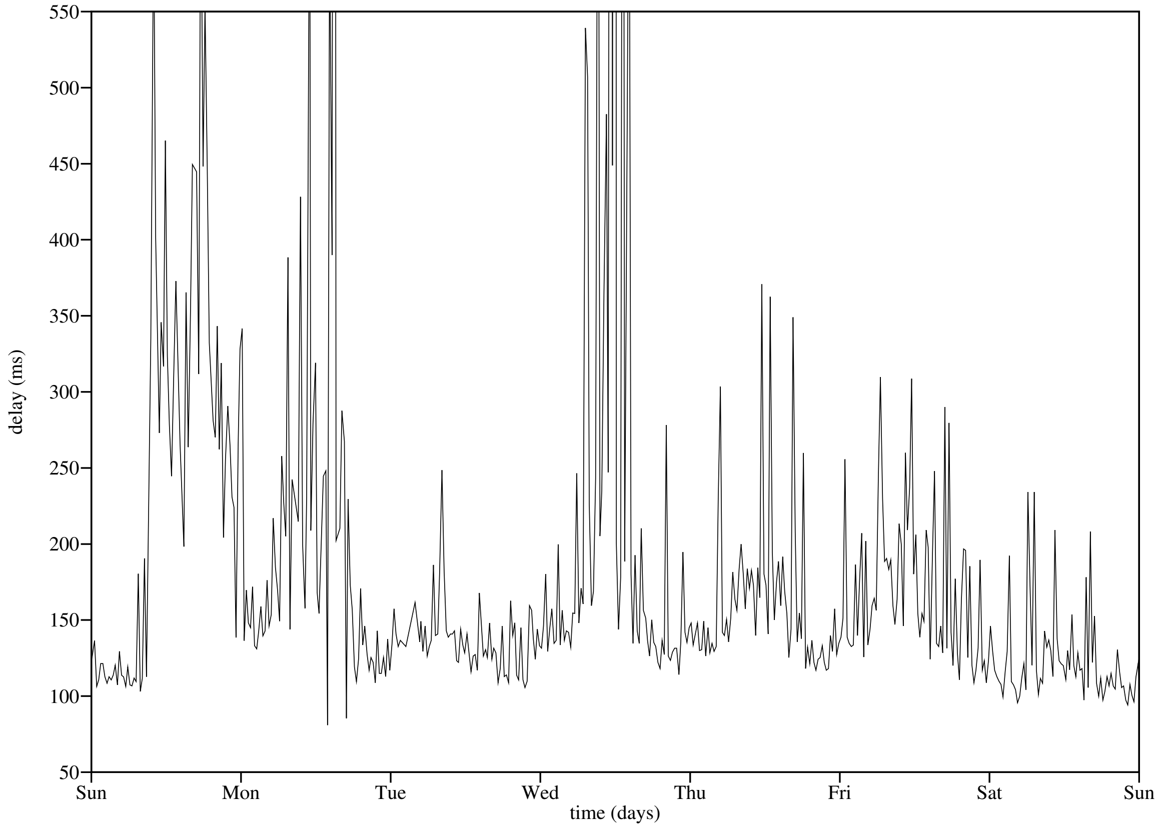

Figure 1: Delay through provider

now access LACHESIS data easily through their favorite Web browser. Nearly real-time analysis and monitoring of ISP performance has been made possible.

Each line of the log file shows when data was collected along with the values for five parameters: network delay, packet loss, number of hosts, hosts unreachable, and hosts unknown. For each of these dependent variables, GRAPHLACHESIS generates a file in the WWW's HTML [9] format. Each of those files are four graphs showing data for the current day, the current week, the previous week, and long term historical trend. Figure 1 shows a graph of delay through an ISP provider during Intel work week 27. Figure 2 is an example of graph of packet loss during that period.

GRAPHLACHESIS parses the log file data and creates individual files for each of the five parameters. The individual files are then fed to a program called WEBGRAPH. WEBGRAPH is a generic graphing package that reads in any single set of data in the form "x, y" and produces a graph that is automatically appended to a specified HTML document. GRAPHLACHESIS calls upon WEBGRAPH repeatedly, each time appending a graph to the appropriate WWW page.

WEBGRAPH was deliberately kept a separate program from GRAPHLACHESIS (as opposed to being a subroutine in GRAPHLACHESIS) because we wanted to have WEBGRAPH available as a stand-alone general purpose graphing package. WEBGRAPH allows the user to control many aspects of the output graph (such as the title, labels, axes ranges, tic marks, and plot style) all from the command line. This way, complete graph generation can be executed in one step. GRAPHLACHESIS takes advantage of this flexibility and demonstrates the usefulness of WEBGRAPH as a generic graphing package.

GRAPHLACHESIS and WEBGRAPH are both written in PERL. The former wraps around the latter, and the latter further wraps around two programs: GNUPLOT [10] and PPMTOGIF from the PBMPLUS package [11]. GNUPLOT plots the data and produces graphs in PBM (Portable Bit Map) format, and PPMTOGIF transforms the PBM graphs into GIF format, making it presentable to the WWW.

Results and Implications

We ran LACHESIS for a number of months against a single Internet vendor. LACHESIS proved useful in recording problems with Internet access. Figure 1 shows a graph of delay during a particularly bad week while figure 2 shows packet loss over this period. Our ISP was having problems with their

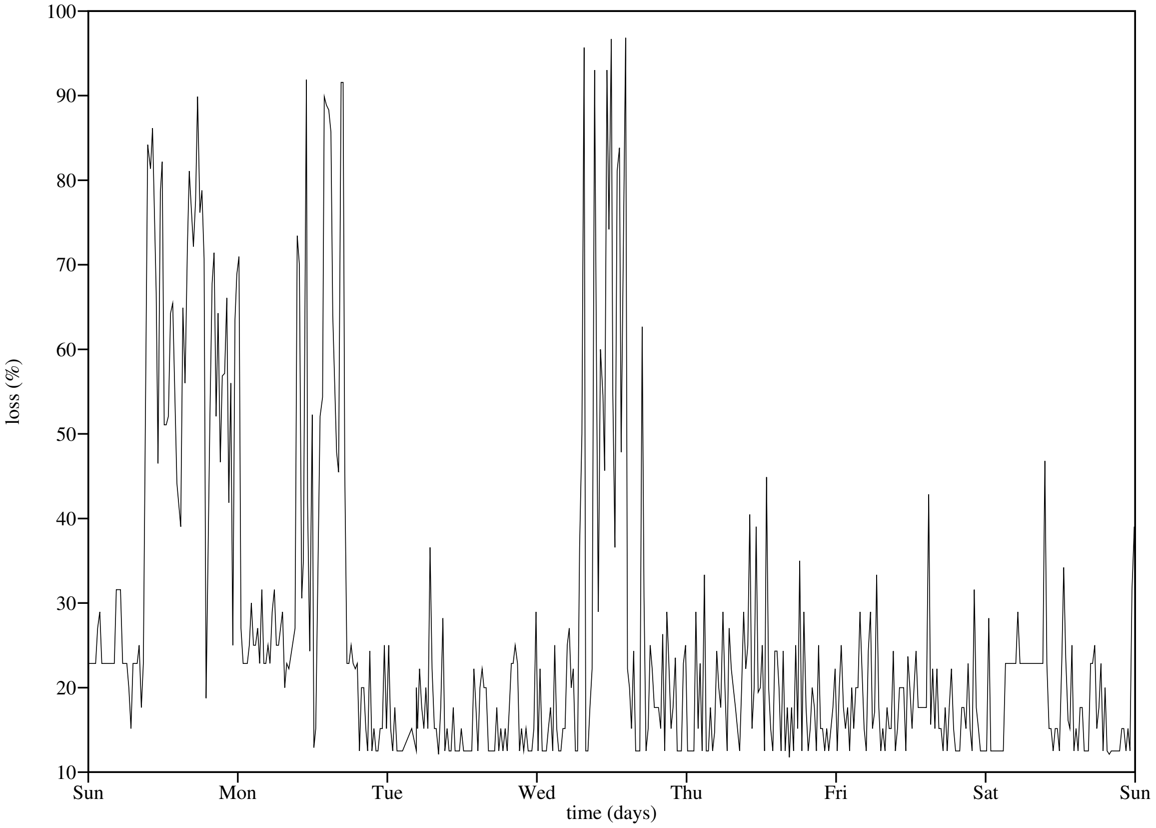

Figure 2: Packet Loss

backbone during this week. Mid-Monday and mid-Wednesday were particularly bad. Note the high delay for systems that could be reached, and the high packet loss (nearly 100%) during those periods. LACHESIS enables us to capture these periods of instability.

Sampling frequency needs to be selected carefully. One common ISP problem we encountered is having our default routes appear and disappear. Since we have multiple connections to the Internet, routes to the Internet and to Intel would be recomputed if our default route disappeared. It would take about 10 minutes for the routes to resolve inside and outside of the Internet. If the route reappeared, it would take another 10 minutes to resolve back again. During these 10 minute intervals, Internet connectivity would be lost. Since we ran LACHESIS every 20 minutes, we lost visibility into these route flaps. In order to catch events such as this, LACHESIS needs to run at least twice as frequently as the length of the event to be monitored.

One interesting suggestion was to run LACHESIS against our own Internet resources (such as the corporate WWW server). This would help us determine our own performance relative on the Internet as well as detect problems in our Internet connections and servers.

What does LACHESIS imply for Internet vendors? To get delays as low as possible, ISPs will need to be well interconnected to other ISPs. Landmarks that are on different ISPs will really bring this out. Vendors who route all of their Inter-ISP traffic through highly congested traffic exchange points will fare poorly using LACHESIS. ISPs must have low internal delay, and routes have to be sensible. ISPs who route traffic between neighboring states across North America and back will not fare well under LACHESIS. ISPs must not drop packets (because of overloaded routers, lines, etc.) because packet loss is measured. ISPs could cheat by letting ICMP packets go through at higher rate, but this is unlikely as it would affect other more important traffic on their networks.

LACHESIS poses a few problems. Will LANDMARKS be flooded by LACHESIS users? We don't think so. Abusers can easily be cut off with router access lists. Another way for LANDMARKS to handle this is to set up "sign posts." These sign posts would be special systems designated to respond to LACHESIS and other applications' pings.

Another problem results from having sites or organizations with multiple Internet connections. If one ISP loses connectivity, LACHESIS starts measuring whatever ISP takes over. The solution to that is to have LACHESIS measuring take place from segments that only route through a particular Internet provider.

One very useful package to fall out of the LACHESIS work is the WEBGRAPH program. We will use this package for plotting all kinds of data and presenting that data on the Web. Using the Web makes LACHESIS information available to a very wide audience. Users running on Intel Architecture PCs to Unix workstations can see the data.

Conclusions and Next Steps

LACHESIS has proven to be a useful tool despite its simplicity. Measuring delay and packet loss is useful for evaluating and benchmarking Internet vendors. Data presented on the Web can reach a wide audience in almost real-time. LACHESIS currently runs on BSDI Unix and SunOS. Future plans include a WINSOCK version for Microsoft Windows (TM) is also being planned to enable individual Internet subscribers to benchmark their own providers. Another idea that we are considering is to apply statistical process control methodologies to LACHESIS data and graphs. We would then benchmark and contact ISPs when they are "out of control".

Acknowledgements

We would like to thank Darci Chapman for ideas about using LACHESIS to benchmark ourselves. Thanks should also go to Cindy Bickerstaff for some ideas on statistics.

Author Information

Jeff Sedayao received a B.S.E. in Computer Science from Princeton University in 1986 and a M.S. in Computer Science from the University of California at Berkeley in 1989. He has worked at Intel Corporation since 1986, spending most of his time running Intel's main internet gateway. Reach him at Intel Corporation; SC9-37; 2250 Mission College Blvd; Santa Clara, CA 95052-8119. Reach him electronically at sedayao@argus.intel.com.

Kotaro Akita is a Junior at Princeton University studying electrical engineering. He plans to concentrate in communication networks and also receive a certificate in Engineering & Management Systems upon graduation. This summer he worked at Intel Corporation for twelve weeks for Jeff Sedayao. Reach him at 1903 Hall #504; Princeton University; Princeton, NJ 08544. Reach him electronically at KAkita@Princeton.EDU.

References

[1] Suzanne Johnson. "Internet Affects the Corporation: Experiences from Eight Years of Connectivity." *Proceedings of INET 95*, Honolulu, June 1995.

[2] Venkata N. Padmanabhan and Jeffrey C. Mogul. "Improving HTTP Latency." *Proceedings of the Second International World Wide*

Web Conference, pages 995-1005, Chicago, October 1994.

[3] Simon Spero. “Analysis of HTTP Performance Problems.´ http://elanor.oit.unc.edu/http-prob.html.

[4] Jeff Sedayao. “World Wide Web Network Traffic Patterns.” *Proceedings of Spring COMPCON 95*. San Francisco, March 1995.

[5] Jeff Sedayao. “Mosaic will kill my Network!” *Proceedings of the Second International World Wide Web Conference*, pages 1029-1038, Chicago, October 1994.

[6] Larry Wall and Randal L. Schwartz. *Programming PERL*. O’Reilly & Associates, Inc., Sebastopol, CA, 1991.

[7] J. Postel. “Internet Control Message Protocol – DARPA Internet Program Protcol Specification,” *RFC 792*, USC/Information Sciences Institute, September 1981.

[8] Stan Barber. “Minutes from the April 1995 Danvers IETF (IP Provider Metrics BOF).”

[9] Tim Berners-Lee, R. Cailiau, A. Luotonen, H. Nielsen, and A. Secret. “The World Wide Web.” *Communications of the ACM*. 37(8):76-82. August 1994.

[10] Thomas Williams and Colin Kelley. *GNUPLOT*. http://www.cs.dartmouth.edu/gnuplot_info.html

[11] Jef Poskanzer. *PBMPLUS*. ftp://ftp.x.org/R5contrib/

[12] John Boswell and Dan Starer. *Five Rings, Six Crises, Seven Dwarfs, and 38 Ways To Win An Argument*. Penguin Books, New York, NY, 1990.