Using the AT&T Labs *PacketScope* for Internet Measurement, Design, and Performance Analysis

Nikos Anerousis, Ramon Caceres, Nick Duffield, Anja Feldmann, Albert Greenberg, Chuck Kalmanek, Partho Mishra, K.K. Ramakrishnan, Jennifer Rexford

Network and Distributed Systems Research Laboratory

AT&T Labs-Research, Florham Park, NJ

October 1997

Abstract. This note describes the AT&T Labs *PacketScope*, a high performance system for packet-level Internet measurement, which we have developed with the help of a great many others in AT&T Worldnet[sm] and AT&T Networking and Computer Systems. The NYC *PacketScope*, operational in June 1997, harvests about 10M packet headers per day on links between AT&T Worldnet[sm] and the external Internet. The Bridgeton, Missouri *PacketScope*, deployed in July 1997, can monitor a variety of traffic from the dial platform, web hosting and other servers, as well as gateway links to the external Internet. The data has a wide range of applications to AT&T Worldnet[sm] marketing, engineering, and service provisioning. One application is described here: to the design and performance tuning of the IP over ATM infrastructure for AT&T's ATM common backbone, which will provide the network platform for AT&T Worldnet[sm]. Finally, we offer some remarks on ongoing simulation and analytic work.

1. Introduction

In its rate of growth and its heterogeneity, the Internet is astonishing. In a few years, the traffic volume has switched from applications based on the TELNET and FTP protocols to World Wide Web applications based on the HTTP protocol. Traffic volumes may soon switch to other applications, for example to streaming multimedia services, such as those offered by *RealAudio* and *Vxtreme*. In this climate, how can we build an infrastructure for large commercial Internets at a level of economy, scale, flexibility, quality and reliability that provides competitive advantages? Basic system parameters such as the application and traffic mix are only roughly understood. Information about the rate of change in these parameters is largely anecdotal. Internet Service Providers are pressed to deploy facilities at large scale at an unprecedented pace. These forces lead to costs in overengineering new subsystems and in firefighting performance problems with older subsystems. A crucial part of the solution is to deploy measurement tools to understand the demands and the performance of the current systems.

In this note, we describe the AT&T Labs *PacketScope*, a tool for packet-level Internet measurement, which we have developed with the help of a great many others in AT&T Worldnet[sm] (WN) and AT&T Networking and Computer Systems. The NYC *PacketScope*, operational in June 1997, harvests about 10M packet headers per day on links between WN and the external Internet. The Bridgeton, Missouri *PacketScope*, deployed in July 1997, can monitor a variety of traffic from the dial platform, web hosting and other servers, as well as gateway links to the external Internet.

It is impossible today to do exhaustive measurement studies of large Internets, as were done for the earlier networks, such as the ARPANET [Kleinrock76]. Instead, network engineers have had to settle for probing the network at advantageous locations. Paxson [Paxson97] provides a taxonomy and a superb review of the state of the art: site studies, server studies, client studies, link studies, LAN studies, and end-

to-end studies. (See also [Claffy95].) In a large commercial Internet, such as WN, a related taxonomy presents itself: studies at the POPs (Points of Presence, where users access the network), access and aggregation routers, data centers, and network routers and switches. In all these locations, packet level data capture provides the simplest, most flexible, and most comprehensive tool for understanding application and network behavior. Systems related to *PacketScope*, presenting somewhat different cost and capability tradeoffs, have been built and deployed by network engineers at MCI [Apisdorf97] and Bellcore.

To amplify the value of these measurements, they must be joined with other data describing network usage and configuration: call record journals, registration data, facilities configuration, regional access options and pricing plans, SNMP query results, router MIB data, dialbot end-to-end logs, server and customer care logs. By joining these data sets we can advantageously attack crucial business problems, such as POP coverage and strategic planning, customer churn analysis, traffic matrix analysis, peering strategies, QoS network engineering, session characterization, internal server performance, web cache effectiveness, protocol behavior, router and switch performance. A coherent information engine is needed to store and query these diverse data sets. This is the role of the AT&T Worldnet[sm] Data Warehouse, an AT&T Worldnet[sm] and AT&T Labs *InfoLab* platform constructed this year. To provide security and privacy, the *PacketScope* and other links in the data recording chain obscure user and site identities.

In Section 2, we outline the architecture of *PacketScope*, describing some implementation details and current deployments. In Section 3, we give one important application of the data: to the design and performance tuning of the IP over ATM infrastructure for AT&T's ATM common backbone, which will provide the network platform for WN. Finally, in Section 4, we offer some remarks on ongoing simulation and analytic work, and some remarks on the infrastructure needed for performance monitoring systems for large Internets.

2. PacketScope Architecture

Passive Link Access

The *PacketScope* system was designed to meet several simple requirements. The most fundamental is that monitors should be strictly passive, observing data flowing on the network without disturbing the network. To properly isolate the monitor from the network, we use one of two alternatives:

- *Amplifiers and a router, appropriately configured, between the network and the monitor*. At the transmission layer, the amplifiers prevent any electrical signal to flow into the network from the monitor. At the network layer, the router only allows traffic to flow from the network into the monitor. No traffic can flow into the network from the monitor environment.
- *A network interface that has the capability to ensure that no data can flow out of the monitor host*. In this case, the device driver and adapter are set up so that no transmits are allowed.

To tap the network passively, there are several alternatives. When the link is a multiaccess channel, such as FDDI, then attaching a station to the channel is all that is required. To monitor a point-to-point T3 link, we use a passive tap into the link. Normal operating procedures on T3 links already facilitate this: each link passes through a DSX-3 panel equipped with monitor jacks. These are routinely employed to extract a copy of line signal for testing purposes. For the packet monitor, for each direction of the T3 to be monitored, this copy is instead passed to an amplifier module, a standard piece of Central Office equipment. This serves both to amplify the signal to sufficient levels for retransmission to the packet monitor, and to physically isolate the production signal from the packet monitor. From the amplifier the signal is sent to the packet monitor, each copy being sent on a T3 (which is unidirectional, without the return path being used). Note that there is no requirement that the packet monitor be co-located with the backbone router to which the monitored T3 attaches. Indeed, for the deployment and testing of the first T3 packet monitor it was decided to house the monitor in a non-Central Office facility. The limitation of such

an approach is that when we want to monitor a different link, we have to add a tap to that link and connect it to the router that forwards data to the packet monitor.

When we go to an ATM environment, we expect to exploit the inherent multicast capability in switches to overcome the limitation of having to monitor on a link-by-link basis. To monitor point-to-point ATM links, an approach others have used is to have an optical splitter, much like the tap for a T1 or T3 link. While we believe the optical splitter can be made to be adequately reliable, we feel the complexity of having several splitters, one for each link, and possibly an optical switch to select one of them for monitoring, is avoidable. ATM switches have the capability to multicast traffic destined to an output port to additional output ports. Conventionally this is performed on a per-VC basis, by setting them up as multipoint VCs. Instead, we believe it is possible to set up all the VCs going to one or more of the output ports to be multicast to a single port designated as a monitoring port, simply through local actions at the switch. This is especially simple if the switch fabric is a multiaccess channel, such as a bus. This will allow for considerable flexibility in our ability to monitor many possible links on the ATM switch, with simple software control. When the capacity of a single monitor has been exceeded, then we can attach additional monitors on additional ports on the switch designated for monitoring. The primary assumption that we make, which we believe is reasonable, is that there is adequate capacity in the switching fabric to support the additional multicast traffic.

Packet Capture

A UNIX workstation is at the center of data collection. We have initially opted to use Digital Equipment Corporation's Alphastations for several reasons:

- We can use a standard UNIX environment, with associated `tcpdump` [Jacobson89] utilities to enable us to get started with monitoring rapidly.
- The path from the device driver to the packet filter utility in Digital UNIX has been enhanced to provide good performance under heavy receive load. The overload behavior is also graceful.
- The Alpha processors are fast. The primary task involved in packet monitoring at the moment is CPU intensive, and we would like, at least initially, to be assured that we can get as many packets traced as possible without loss. As we enhance our understanding, we may be able to gainfully use sampling techniques, and therefore tolerate a slower processor.

The workstation listens in promiscuous mode to all incoming traffic in order to collect packet trace data. The adapter and datalink driver are set to be in promiscuous mode. The `tcpdump` application instructs a packet filter in the UNIX kernel to discard all but the first few bytes of every packet (128 bytes are typical of our experiments). `tcpdump` then saves the captured packet headers to disk. We also have the option of having a separate program copy data to magnetic tape. Tape provides an order of magnitude more storage than disk arrays, albeit at slower speeds. We expect the tape to fall behind the disk during periods of high network load, but to catch up during more idle periods such as very late at night.

When there is a router between the monitored network and the monitor, then all of the data has to be forwarded on to the link on which the monitor sits, by appropriately configuring the router. The main pieces of equipment in the monitor cabinet are a Cisco 7505 router, on which the T3 from the amplifiers terminates, and the DEC Alphastation at which the packets are collected. The Alphastation connects to the monitor router with 100BaseT Ethernet. The router forwards all packets received on the T3 link onwards to the Alphastation, where they can be captured by the `tcpdump` program. However, for security purposes it is desirable not to have an IP path from the router to the Alphastation; furthermore, `tcpdump` does not require IP to be enabled at an interface that it monitors. So instead the following arrangement is employed. The router is configured with a static IP route of last resort which routes all packets to a fictitious IP address. This IP address lies in a subnet associated with the router interface to which the Alphastation attaches. A static ARP entry in the router associates the MAC address of the Alphastation's 100BaseT interface with the fictitious IP address. Thus all packets appearing at the monitor router's T3 interface

are forwarded on the interface to which the Alphastation attaches. Although IP is disabled at the Alphastation's 100BaseT interface, `tcpdump` recovers the packets at the datalink layer, which is running in promiscuous mode.

Management and Security

The monitors are manageable remotely, so that on-site presence of personnel to manage the monitoring equipment is not essential. A small Cisco router controls access to each monitor. The main access path is through a T1 line to AT&T Labs in Florham Park, NJ. There is also a POTS line connected to the router. It serves as backup in case the T1 is unavailable. Through either of these links, we can remotely start and stop data collection, reconfigure the packet filters, and perform many other control functions. We can also transfer programs and data in and out of the monitor.

At the same time, we protect the data collected on the monitor from unauthorized users using several techniques. First, we disallow logins to the monitor except from selected nodes in the AT&T Intranet. Second, we use the SSH (secure shell) software package to allow only authenticated, encrypted logins to the monitor. Third, dialup access is additionally protected with passwords and dialback.

Deployment

We have deployed two packet monitors inside WN, one in Bridgeton, Missouri, and one in New York City. The Bridgeton monitor taps into FDDI links internal to WN and is equipped with a magnetic tape stacker. The New York monitor taps into T3 links connecting WN to the rest of the Internet. Figure 1 shows the organization of our monitoring infrastructure.

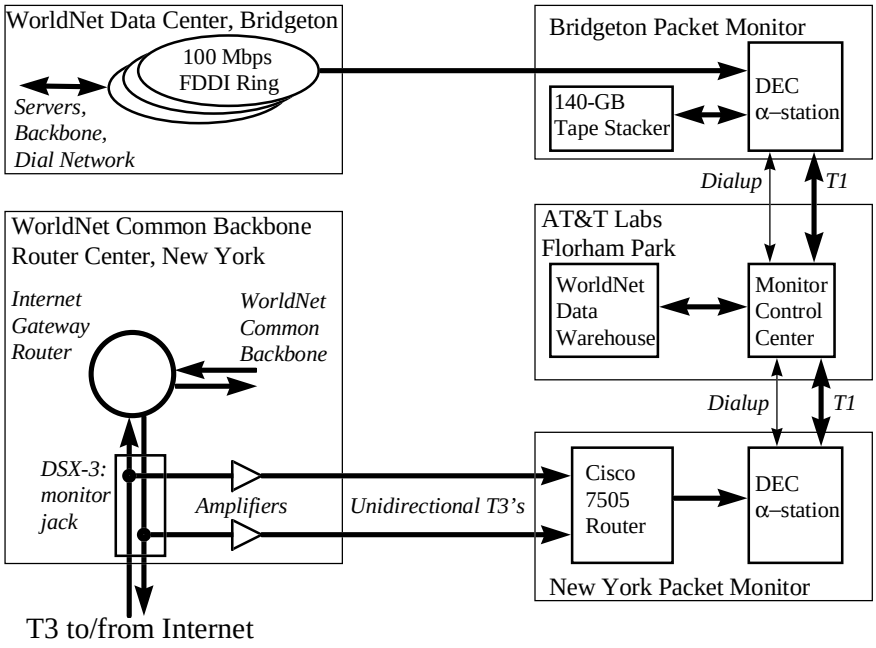

Figure 1. Current Deployment of Packet Monitoring Equipment

3. Application to Network Design and Analysis

The WorldNet packet-level traces facilitate a wide range of realistic performance evaluation work that can guide the design and provisioning of future network services. In collaboration with AT&T ATM Services, we are applying this extensive trace data to model various strategies for carrying connectionless Internet traffic on a circuit-switched backbone network [Feldmann97].

To efficiently transfer a large amount of diverse IP traffic over a high-speed backbone network, modern integrated networks require more efficient packet-switching techniques that can capitalize on recent advances in ATM switch hardware. Several promising approaches improve application performance and network utilization by creating dedicated "shortcut" connections for long-lived traffic flows, at the expense of the network overhead for establishing and maintaining these connections through the underlying switch fabric. The network can map packet-level IP traffic to ATM connections by detecting *flows* of packets that travel between the same end points. The ATM network can offer quality-of-service guarantees to these long-lived flows, such as large HTTP transfers and multimedia streams, by establishing dedicated "shortcut" virtual circuits between the ingress and egress routers, while forwarding other IP packets on "default" virtual paths. Although explicit short-cuts offer more predictable performance and better network utilization, creating and using these shortcuts consume processor and switch resources. In limiting these overheads, the network can control three tunable parameters: the *end-point address granularity* (e.g., port, host, subnet, net) for traffic in a single flow, the *timeout* for grouping these related packets into flows (e.g., 12 seconds, 1 minute, 5 minutes, etc.), and the *trigger* for creating a shortcut connection to carry the remaining packets in the flow (e.g., 5 packets, 10 packets, 50 packets, etc.). Through proper selection of these three parameters, ATM service providers can provision for the appropriate *proportion of traffic on short-cut routes* (quality of service), *short-cut set-up rate* (signaling load), and *number of simultaneous virtual circuits* (switch load).

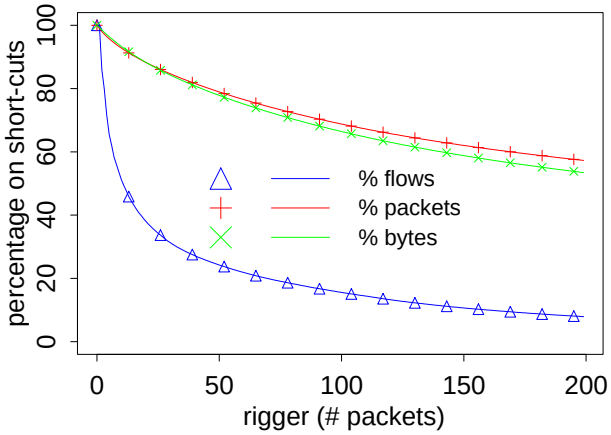

Figure 2. Percent Shortcut Traffic.

Drawing on a one-week *PacketScope* trace of World Wide Web transfers to WorldNet clients, Figure 2-4 plot these three metrics for a variety of packet triggers, a 1-minute timeout, and host-to-host end-point addresses. Defining flows as traffic between two end-point hosts, independent of the specific TCP or UDP port number, combines consecutive transfers between a Web server and a client into a single shortcut connection. Although the link carries a large number of short-lived flows, the majority of packets and bytes stem from the small number of long-lived flows, as shown in Figure 2. Hence, a 10-20 packet trigger can substantially reduce network overheads, while still permitting a large proportion of the traffic to reap the

quality-of-service benefits of shortcut connections. For example, the network could assign a guaranteed throughput to each shortcut to isolate traffic to and from different users.

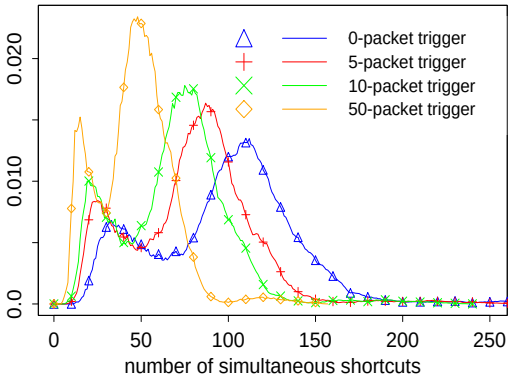

Figure 3. Number of Shortcuts.

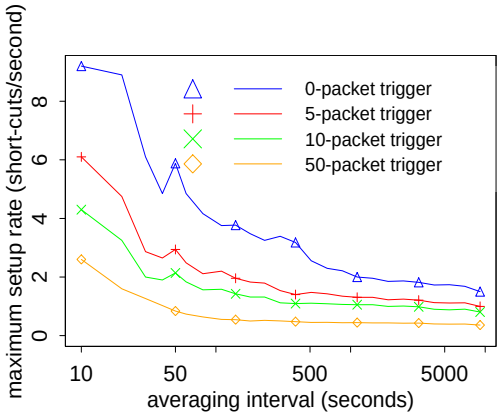

Figure 4. Shortcut Setup Rate.

The network overheads vary with the time of day, as shown in Figure 3, which plots the probability density function of the number of simultaneous shortcut connections. The two distinct peaks, for the light load in the early-morning hours and the heavier load during the rest of the day, suggest guidelines for selecting the trigger parameter to avoid exhausting the virtual-circuit capacity of the network switches. Larger packet triggers can substantially reduce the number of connections, particularly during the peak busy hours. In addition, compared with port-level flows, host-level address aggregation lowers overheads by a factor of *four* by combining consecutive and concurrent TCP sessions from the same server to the same client in a single shortcut connection [Feldmann97]. By aggregating traffic at the host level, the network can also isolate well-behaved clients from heavy users that open a large number of TCP sessions at the same time.

Finally, Figure 4 shows the maximum connection set-up rate on a variety of time scales, ranging from 10-10000 seconds. For example, the far left of each curve plots the set-up rate during the most heavily-loaded ten-second interval in the one-week trace; variations in flow arrival rates cause reductions in the set-up rate over larger time intervals. Effective triggering policies can substantially reduce the signaling load for establishing shortcut connections. The variation in set-up requests across the range of time scales can assist in striking a trade-off between the signaling capacity of the network and the delay in creating shortcut connections during times of peak load. For example, during short-term load fluctuations, the network could temporarily delay the creation of shortcuts and force packets to continue to follow the default path. This would allow the network to provision the signaling resources for a lower “worst-case” set-up rate.

These results, and other experiments with timeout, trigger, and end-point addressing parameters, can offer guidelines for designing and provisioning the AT&T Common Backbone. In this context, we are developing stochastic models of IP flows to develop analytic models for tuning shortcut policies.

4. Ongoing Work

The *PacketScope* monitor enables a diverse assortment of measurement-based models of communication networks, ranging from low-level analysis of TCP dynamics to high-level characterization of the WN traffic mix. By changing the data-collection mode, the monitors can vary the granularity of information in the packet traces. For example, instead of collecting the IP header of every packet, the *PacketScope*

can record the full HTTP header for web request and response messages, or even entire HTTP documents (or at least their checksums) to perform a detailed study of web caching services.

Strategic deployment of *PacketScopes* can provide realistic packet traces to drive web resource cache simulations and implementations, for a realistic evaluation of how new caching schemes would perform in large commercial Internets *today*. Several promising web caching mechanisms have been evaluated using local packet traces or server logs, which may not be representative of the traffic generated by a large commercial Internet. (See [Douglis97] and references therein.) In ongoing work, we are applying the data harvested from *PacketScope* to simulate these mechanisms on appropriate, realistic traffic. Moreover, for web resources whose rate of change limits cache performance, we can look one level deeper, and ask how large are successive changes (deltas) to the resources [Douglis97]. Preliminary results indicate that a combination of delta-encoding and data compression can provide remarkable improvements in response time for an important subset of HTTP content types [Mogul97].

The *PacketScope* monitors offer a unique opportunity to collect packet-level data at various points in the network, from the end-point user at a modem pool to large web sites in the web-hosting service. Comparative study of these datasets can help guide the placement of new network services, such as IP short-cutting and proxy caching, which can improve client performance and reduce network overheads. The diverse collection of measurements can also enable the development of new analytic models for traffic profiling, provisioning, and resource allocation for emerging network services.

Finally, some remarks are in order on the best options before us for performance monitoring systems for large Internets. Performance monitoring systems serve two sets of goals:

- *customer facing* goals, such as: underwriting service level guarantees; providing customer specific value added services; attracting new customers; and
- *network facing* goals, such as: ensuring DMOQ's (direct measures of quality) are met; diagnosing, troubleshooting, and isolating persistent network problems; providing planning and support for future differential services; and understanding the impact of future technology and user behavior for network evolution.

To meet these goals, we need an architecture describing how to deploy measurement tools within the network and at the network edge, to economically collect data on availability, route stability, packet loss, packet delay, and throughput. The tools fall into two categories:

- passive tools (such as the *PacketScope*), which inject no traffic in the network, and
- active tools (such as *ping*), which inject test traffic in the network.

It is our belief that *the measurement architecture must combine and correlate passive and active measurement*. In particular, we argue that a key part of the measurement infrastructure for the common backbone for AT&T Worldnet[sm] should be a deployment of measurement tools in the large router centers for:

- active measurement with minimal interference with other systems–from router center to router center, from router center to network interconnection points, and from router center down to the edges of the access/aggregation networks;
- passive measurement of traffic within the router center;
- passive measurement of traffic to remote servers co-located at large router centers;
- remote data harvesting and reduction.

It makes sense to use the "reference servers" deployed in the router centers to maintain views of the common backbone performance, continuously on the web, for internal use and for customer use. Measurement tools occupy the lowest level in the stack of tools needed for decision support in a performance monitoring system. To make sense of the data, the whole stack is needed:

- data analytic tools, diagnostic tests;
- data reduction, synthesis and warehousing;
- active monitoring tools (end to end, reference servers);
- passive monitoring tools (server and router logs, *PacketScope* data).

The AT&T Worldnet[sm] Data Warehouse (report in progress), an AT&T Labs *InfoLab* project, aims to provide the first two layers of the stack.

Acknowledgment: John Friedmann provided a tremendous amount of help and support. We are very grateful for the help of Sam Alexander, Eileen Carey Brown, Don Coffield, Ted Eckberg, Mohammed El-Sayed, Bob Fuller, Steven Gao, Peter Glasser, Sam Herr, Ray Hodge, Georg Karawas, Larry Koons, Jim Lynch, Ray Rossmann, Vikram Saksena, Fred Scherer.

References

[Apisorf97] J. Apisdorf, K. Claffy, K. Thompson, and R. Wilder, "OC3MON: Flexible, Affordable, High Performance Statistics Collection," http://www.nlanr.net/NA/Oc3mon.

[Claffy95] K. Claffy, H.-W. Braun, and G. Polyzos, "A Parameterizable Methodology for Internet Traffic Flow Profiling," IEEE JSAC, 13(8), pp. 1481-1494, October 1995.

[Douglis97] F. Douglis, A. Feldmann, B. Krishnamurthy, and J. Mogul, "Rate of Change and Other Metrics: A Live Study of the World Wide Web," to appear in USENIX Symposium on Internet Technologies and Systems, December 1997.

[Feldmann97], A. Feldmann, J. Rexford, and R. Caceres "Reducing Overheads in Flow-Switched Networks: An Empirical Study of Web Traffic," preprint, July 1997.

[Jacobson89] V. Jacobson, C. Leres, and S. McCanne, `tcpdump`, ftp://ftp.ee.lbl.gov, June 1989.

[Kleinrock76] L. Kleinrock, "Queueing Systems, Volume 2: Computer Applications," J. Wiley & Sons, 1976.

[Mogul97] J. Mogul, F. Douglis, A. Feldmann, and B. Krishnamurthy, "Potential Benefits of Delta-Encoding and Data Compression in HTTP," Proc. ACM SIGCOMM, pp. 181-194, September 1997.

[Paxson97] V. Paxson, "Measurements and Analysis of End to End Internet Dynamics," Ph.D. Dissertation, UC Berkeley, April 1997.