An Architecture for Large-Scale Internet Measurement

Vern Paxson
Network Research Group
Lawrence Berkeley National Laboratory*
Berkeley, CA 94720
vern@ee.lbl.gov

Jamshid Mahdavi, Andrew Adams and Matt Mathis
Pittsburgh Supercomputing Center
Carnegie Mellon University
Pittsburgh, PA 15213
mahdavi@psc.edu, akadams@psc.edu, mathis@psc.edu

Abstract

Historically, the Internet has been woefully under-measured and under-instrumented. The problem is only getting worse with the network's ever-increasing size. We discuss the goals and requirements for building a "measurement infrastructure" for the Internet, in which a collection of measurement "platforms" cooperatively measure the properties of Internet paths and clouds by exchanging test traffic among themselves. The key emphasis of the architecture, which forms the underpinnings of the National Internet Measurement Infrastructure (NIMI) project, is on tackling problems related to *scale*. Consequently, the architecture emphasizes decentralized control of measurements; strong authentication and security; mechanisms for both maintaining tight administrative control over who can perform what measurements using which platforms, and delegation of some forms of measurement as a site's measurement policy permits; and simple configuration and maintenance of platforms.

1 Introduction

Historically, the Internet has been woefully under-measured and under-instrumented. The problem is only getting worse with the network's ever-increasing size. Several research efforts are tackling this problem by working on developing Internet "measurement infrastructures" [LJJMH+97, Al97, HGDL97]. Such an infrastructure consists of a collection of measurement "platforms," which are dedicated workstations sited at different locations around the network. The platforms cooperate with one another to measure the properties of Internet paths and clouds by exchanging test traffic among themselves.

There are several general uses for a measurement infrastructure: to diagnose performance problems in the interior of the network by probing the network's characteristics from different vantage points; to measure the properties of a wide range of network paths to facilitate research into how the network behaves and evolves [Pa97a, Pa97b]; and to assess the performance delivered by different Internet service providers (ISPs), by measuring what happens to traffic injected into one end of an ISP's network for delivery across the ISP's "cloud" to a remote point on the other end of the network. By systematizing the assessment of ISP performance, this last function holds considerable promise for spurring ISPs to optimize their networks.

We are currently developing the "National Internet Measurement Infrastructure" (NIMI). NIMI differs from the other infrastructure projects in that the emphasis with NIMI is on building a generalized architecture for scalable measurement infrastructure, rather than performing a specific set of measurements for specific analysis goals.[1] By "scalable" we mean that our hope is to devise an architecture that might one day see ubiquitous deployment throughout the Internet. However, even building a smaller-scale measurement infrastructure is difficult (as we discuss in § 2).

We note that the "end-to-end" nature of Internet performance measures (such as quantifying packet loss rates along the path from a data sender to its receiver) often entails analyzing the behavior of a chain of distinct ISPs, no single one of which can by itself fully monitor the traffic. Consequently, elements of the infrastructure must be able to cooperate with other, administratively diverse elements.

---

*This paper is to appear in *IEEE Communications*, 1998. This work has been supported through funding by National Science Foundation Grant No. NCR-9711091. V. Paxson's efforts were also supported by the Director, Office of Energy Research, Office of Computational and Technology Research, Mathematical, Information, and Computational Sciences Division of the United States Department of Energy under Contract No. DE-AC03-76SF00098.

[1] Indeed, if NIMI is fully successful, then it would be natural to build the other measurement infrastructures on top of it.

A natural question when considering deploying a large-scale infrastructure is then: who will own, control and maintain the platforms? One possible answer is that the infrastructure will belong to a single organization. However, we argue that the heterogeneity of the Internet makes this approach unlikely to work in practice, and, instead, if we hope to achieve widespread deployment, then the network operators themselves must decide to deploy measurement platforms. We see several important incentives for them to do so. First, the infrastructure will aid in detecting and debugging performance problems within their network and with their peers, enabling them to better operate their network. Second, we hope that eventually "service-level agreements" between the network operators and their customers will include provisions for on-going measurement so that the customers can verify that they really do obtain the service stated in their contracts. Third, while an operator might want to limit the access provided by their platforms for measuring their *own* network, we imagine that in some cases operators will be happy to make their platforms available for measuring the performance of their *competitors*. This last fits with a major goal of NIMI, namely to facilitate *public* access to Internet measurements. This does not mean *unlimited* access, which is politically infeasible, but instead *sufficient* access, i.e., access sufficient for debugging problems, studying overall Internet behavior, and, in some circumstances, assessing the service delivered by different ISPs.

The possibility discussed above of an operator perhaps making their platforms publicly available for limited types of measurement is an example of the notion of "measurement policy." We use this term to mean defining the specifics of "who can perform what measurements using which platforms." With NIMI we do *not* attempt to define any particular measurement policies. Doing so, we believe, is hopeless, if our goal is widespread deployment in the Internet. Instead, we emphasize the development of general *mechanisms* for defining and enforcing a wide range of policies.

In the remainder of this paper we discuss the NIMI architecture in high-level terms: the requirements for a large-scale measurement infrastructure, and how different NIMI mechanisms combine to address these. We spend considerable time developing the requirements, as we believe these are crucial to understanding and shaping the architecture.

We confine our discussion to the architecture of the NIMI platforms themselves. We do not discuss here the specifics of how measurement is coordinated between multiple platforms, nor how analysis is conducted, except to note that we view the platforms as "measurement engines" whose proper job is to accurately conduct different types of measurements, and nothing beyond that.

We first discuss our experiences with a measurement infrastructure prototype (§ 2), as these experiences greatly aided us in formulating the set of goals and requirements we believe a large-scale infrastructure must meet (§ 3). In § 4 we give an overview of the NIMI architecture and discuss how it attempts to address these goals and requirements. In § 5 we briefly sketch the status of the NIMI implementation (discussed in greater detail in [AMMP98]), which is preliminary, and in § 6 offer thoughts on a number of challenging areas for future work.

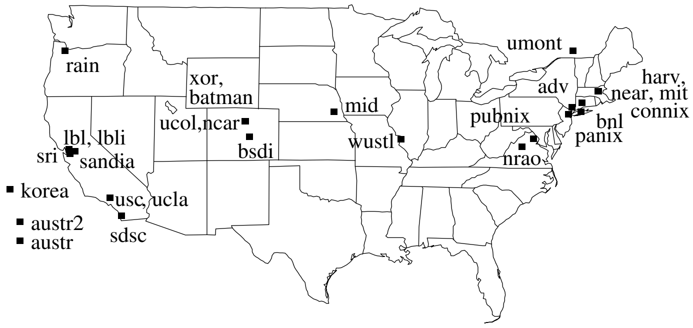

Figure 1: North American and Asian NPD sites

2 Experiences with a prototype

In this section we discuss experiences gained by creating a prototype of a measurement infrastructure, because these experiences highlight several problems that in turn shaped our development of the requirements and goals for building measurement infrastructure (discussed in § 3).

The original interest of one of us (Paxson) in the problem of developing a measurement infrastructure arose from the observation made by Danzig and colleagues that traffic mixes (i.e., which protocols comprise what fraction of the traffic) vary greatly from site to site [DJCME92]; and from the finding in our follow-on work that, in addition to mix, even the basic characteristics of TCP connections such as the distribution of their sizes or durations vary so much from site to site that we cannot soundly use the term "typical" to try to characterize Internet traffic [Pa94]. There is simply too much variation. Instead, we must attempt to soundly characterize the full range of behavior we might encounter at different points in the Internet.

Clearly we must measure Internet traffic properties from a large number of sites in order to capture the full range of behavior. With this in mind, when we embarked on a large-scale study of end-to-end Internet routing and packet dynamics, we developed a specialized measurement program, the Network Probe Daemon (NPD), which then—with the much-appreciated help of numerous volunteers—was deployed at more than 30 sites around the Internet, as shown in Figure 1. Each NPD functioned as a "measurement server" that would accept requests to either measure the route between the NPD host and a remote host using the `traceroute` utility [Ja89], or to source or sink a TCP bulk transfer and record the packets using `tcpdump` [JLM89].

A key property of the NPD measurement infrastructure is that it exhibits N^2 scaling: if the infrastructure consists of N sites, then we can measure $O(N^2)$ Internet paths between

Figure 2: N^2 mesh scaling: North American links measured by the NPD study

the sites. This scaling property means that a fairly modest (in terms of N) infrastructure can potentially observe a wide range of Internet behavior. Figure 2 shows the different links measured in the study. The N^2 scaling effect is readily apparent—a few dozen sites allowed us to study hundreds of paths through the network.

The results of the routing and packet dynamics studies are discussed in [Pa97a] and [Pa97b]. Here, our interest lies in the lessons learned from building and operating this prototype infrastructure.

First, we found that running on-going "mesh" measurements between the NPD sites took an inordinate amount of effort simply to fix problems that inevitably cropped up. While the likelihood of a single site exhibiting a problem was low, once the number of sites becomes at all appreciably large (say, more than a dozen), the probability approaches 1 of at least one of them suffering from some sort of error (system crashed and failed to restart; daemon configuration accidentally overwritten; updates required to NPD or measurement software; packet filter misconfigured; disk space exhausted; clock running berserk; firewall changed to not allow the measurement traffic; account privileges revoked; daemon unable to obtain lock on log file). Since our goal was to sustain full mesh measurements, we needed to attend to these problems, a cumbersome process involving the exchange of a great deal of email, since we did not have direct administrative access to most of the NPD hosts.

These experiences suggest several steps in order to minimize the maintenance burden of an infrastructure: *(i)* keep the elements of the system homogeneous to the extent possible, as a number of the problems were caused by differences among the varying operating systems hosting the NPD software; *(ii)* build into each measurement point as much self-diagnosis as possible; *(iii)* facilitate remote access for administration such as automating software upgrades; *(iv)* design measurement procedures and analysis to accommodate inevitable outages.

A second lesson concerned how to coordinate measurement. In the NPD studies, a single workstation W in our office was used for all measurement scheduling: it ran through a (randomized) list of which measurements should be conducted at which times, and whenever the time arrived for a new measurement from NPD host A to host B, W would contact A and B to initiate the measurement. Clearly, such centralized coordination will fail when the infrastructure becomes too large.

However, it also suffers from a second serious shortcoming. If our goal is to conduct *unbiased* measurement of the characteristics of various paths, then a centralized approach for scheduling measurement fails: whenever W cannot connect to *either* of A or B, no measurement will be performed. It is quite possible that the problem preventing W from reaching A or B coincides with a problem along the path from A to B, i.e., a problem that we would have measured and included in our analysis, if only W could have contacted both A and B to arrange for the measurement. Thus, the centralized design introduces a form of *bias*: our measurements will be artificially skewed towards underreporting serious connectivity problems, because we will sometimes lose the opportunity to measure that the problem is occurring.

This problem of bias is actually less rooted in the use of a centralized scheduling agent, and more in the notion of scheduling *immediate* measurement. If W could instead instruct A and B to conduct a measurement at some point fairly far off in the *future*, then if W failed in its initial attempt to schedule the measurement, it could continue to try to do so. If it ever finally managed to contact both A and B, it would then still schedule the measurement for the original desired time, preserving the sampling properties designed into the original measurement schedule. With this approach, we introduce bias due to failure to schedule measurement only in the case of major, long-lived outages.

The upshot of this observation is to highlight the importance of incorporating the notion of a measurement schedule not only for use by the coordinating agent (W in our example), but also in the measurement platforms themselves (A and B), and that as soon as the agent decides on a future measurement, it should begin attempting to schedule it with the platforms.

Similarly, if we only support measurement-on-demand, then when we want to diagnose network problems, we will sometimes lose the ability to perform the measurements necessary for doing so. If, however, we schedule routine diagnostic measurements in advance, then we will have measurements of problems available for retrospective analysis.

A final lesson concerned what we thought at the time was a prudent mechanism to include in NPD, namely a limited lifetime for each measurement of 10 minutes, after which the NPD would automatically terminate the measurement under the presumption that something had gone wrong and action was required lest the NPD wedge. The problem with such a lifetime is that it again biases the measurement results by eliminating any measurements that happen to require more time than the lifetime—often losing opportunities to measure times of particularly poor network performance.

The measurement lifetime was motivated by the fact that the NPD was designed to only perform one measurement at a time. Consequently, if an NPD ever wedged, the NPD site would be lost to future measurement opportunities until it could be manually unwedged. A better way to deal with this problem is to eliminate the basic limitation: the measurement platforms should be able to conduct concurrent measurements. This functionality, however, also requires that the platforms can diagnose when two concurrent measurements will conflict (for example, in their access to common resources), and thus we must include resource locks associated with particular types of measurement.

Two other noteworthy features of the NPD prototype were its use of fairly strong authentication of measurement requests, based on shared private keys and a challenge/response protocol; and its built-in “self-test,” which would attempt to diagnose configuration problems. In retrospect, the authentication mechanism was only a slight burden (it required the development of a simple debugging mode for the coordination agent, one that first carried out the