Justine Sherry - Previous Research

When average users connect to the Internet, they have little to no idea where their data is traveling, how long it will take to transmit, and who may be manipulating their traffic while it's en route. When network operators, researchers, and Internet-dependent businesses investigate the same issues, obtaining a meaningful view of the Internet is imperative: their enterprises are dependent on Internet functionality, and when packets become delayed, misrouted, or lost, their systems or businesses can fail. However, despite their resources, addressing these issues still proves difficult.

One way in which these issues are addressed is by issuing specially crafted packets to targeted destinations, and making inferences from the responses about different features of the web. While there are many types of probes used in these active measurements, such as pings, traceroute, and record-route enabled packets, the amount of data that can be gathered with such probes is still extremely limited. Seemingly simple data, like exactly how long it takes a packet to travel one-way across an arbitrary link, or whether or not two IP addresses are "aliases" belonging one single machine, remain difficult to measure.

Prespecified Timestamps

The emphasis of my work has been improving the types of measurements available for addressing these fundamental issues by including the the IP Prespecified Timestamp option in the toolbox of Internet measurement utilities. Prespecified Timestamps, an option built-in to every IP packet, allow the sender to specify up to four IP addresses from which to request timestamps. Each router along the path checks the first unstamped address field. If it owns that address, it will provide a timestamp in the form of milliseconds since midnight UTC.

While traditionally ignored for presumed limited coverage and inconsistent behavior, I have demonstrated that over 25% of routers will support timestamps and extensively documented the router-specific, distinct implementations of timestamp support. Furthermore, I argue that measurements gathered with this type of probe have several unique characteristics that set them apart from other tools, providing for new means to investigate basic issues that face effective Internet Measurement.

To this end, I have worked on several applications of prespecified timestamps. First, I have improved its usage within the Reverse Traceroute [1] system, which allows users to discover paths from a destination to their source, without control of the destination. Next, I developed arguments for confirming IP alias pairs using timestamp measurements. Currently, I am exploring using the timestamp literal values to measure one-way link latencies of individual links.

Reverse Traceroute

When I started my work with prespecified timestamps, they were already used in a limited fashion for Reverse Traceroute, a project being developed at my university. Reverse Traceroute seeks to identify reverse paths (destination to source), an issue that has remained difficult despite the ease of discovering forward paths (source to destination) since the invention of simple traceroute in the early 80s. Understanding the asymmetry of forward and reverse paths is extremely important to many people who provide web-based services; interacting with any client involves both receiving and sending traffic, and problems on either transmission mean a failure to connect to the client. Timestamps are used as a 'backup' step in Reverse Traceroute. When the primary method fails, a number of 'guess' hops are generated from the known topology[2], and timestamp probes are sent along the round-trip path requesting stamps from the destination, followed by the guessed IP address.

However, the rudimentary implementation was unsuccessful for several relatively common cases, thus limiting it's effectiveness. Using my knowledge of the various timestamp behaviors, I was able to make several recommendations to improve the Reverse Traceroute algorithm with regard to timestamps. Most significant among the limitations were unsolicited timestamps in easily identifiable cases, introducing the threat of false positives. Another limitation was that some destinations would fail to provide timestamps, limiting coverage despite the fact that the address on the path that we seeked to identify might still provide timestamp values. The new algorithm overcomes both of those limitations, and also includes other optimizations which decrease the number of required probes.

Alias Resolution

Making use of user-specified interface requests in timestamp probes, I looked towards IP alias resolution as a potential application for timestamps. IP alias resolution is a frequent problem for those who may have information about two different IP addresses, but don't know whether the addresses actually belongs to two different machines, or one machine with two addresses.

I discovered several distinct behaviors which routers demonstrated when requested for timestamps from two of their interfaces in the same probe. Subsequently, I came up with three distinct techniques for the identification of alias pairs, each of which addressed a different implementation of the timestamp option. These techniques depended on matching timestamp values to a common clock, evaluating the number and ordering of timestamps received, and verifying matching 'time to live' values for each alias pair. I then generated a new dataset of aliases, and both validated and compared against the leading alias resolution technique[3]. I identified thousands of aliases that were previously undiscovered. As the next generation of alias datasets are moving to combining several techniques [4], I believe that timestamps are an important step towards a comprehensive alias resolution system.

One-Way Delay

This set of work is just beginning. The eventual goal is to correctly identify the one-way latency of a single backbone link, by sending a timestamp request which traverses the link specifying timestamps from each machine on either end of the connection. Interesting challenges to the project are the variety of responses that routers provide when encountering a packet in transit, and independent clock skews for every routers. These challenges should be surmountable. As I demonstrated in both previous applications, distinct implementations of timestamp behaviors are documentable and working with each behavior requires an investigation of a limited set of cases. Independent clock skews can be solved for by issuing several probes along the link, and using algebraic manipulation to 'cancel out' skews in opposite directions.

Future Work

I will graduate in March, and the complete timestamp work will constitute my senior thesis. As the general timestamp data, alias resolution, and delay work remain unpublished, I hope to either develop them into a standalone paper, or integrate them into a related work that can make use of the new measurement capabilities provided by timestamp requests. Reverse Traceroute, now a complete and deployed system, is continuing to open up new avenues of research only just made possible by the potential to chart out previously invisible reverse paths.

References

[1] E. Katz-Bassett, H. V. Madhyastha, V. Adhikari, C. Scott, J. Sherry, P. van Wesep, A. Krishnamurthy, and T. Anderson, "Reverse traceroute." Under submission.

[2] H. V. Madhyastha, T. Isdal, M. Piatek, C. Dixon, T. Anderson, A. Krishnamurthy, and A. Venkataramani, "iPlane: An information plane for distributed services," in *OSDI*, 2006.

[3] A. Bender, R. Sherwood, and N. Spring, "Fixing Ally's growing pains with velocity modeling," in *IMC*, 2008.

[4] K. Keys, "Internet-scale IP alias resolution techniques," in *ACM SIGCOMM Computer Communication Review (CCR)*, 2009.