Inferring Router Statistics with IP Timestamps

Andrew D. Ferguson
Brown University
adf@cs.brown.edu

Rodrigo Fonseca
Brown University
rfonseca@cs.brown.edu

1. INTRODUCTION

This paper describes our initial experiences with using the IP timestamp option to infer router statistics such as traffic shape and CPU load. By deducing this information through the use of an IP option, we can gather these statistics without administrative access to the router under study. This creates the potential for researchers to gather basic statistics from routers in wide-area networks which support this option.

The IP timestamp option is an Internet packet option which requests that IP devices place a timestamp with millisecond-accuracy in the packet's header [1]. Three modes of the IP timestamp option are available: collect timestamps from each device in succession (space in the header is available for up to nine devices), collect the timestamp and IP address from each device in succession (up to four), or, alternatively, the sender may pre-specify up to four IP addresses from which a timestamp is requested. We make use of the third type. Although IP timestamps can be used with any type of IP packet, to date we have only explored their use with ICMP packets due to their convenient availability in the Linux `ping` command.

Our use of the IP timestamp option is motivated by its ability to selectively measure the delay on a link between two routers because we can request timestamps from the ingress and egress interfaces for both routers. This link delay consists of four parts: propagation delay, transmission delay, processing delay, and queuing delay [2]. The processing and queuing delays are dynamic properties of the router which we are seeking to infer by measuring the variation across repeated IP timestamp requests.

2. INITIAL EVALUATION

To characterize the relationship between timings provided by the IP timestamp option and router statistics, we have begun to evaluate their use in two controlled environments with different router models. Through the facility provided by the Wisconsin Advanced Internet Lab (WAIL),[1] we have explored the correlation between UDP traffic and IP timestamps produced by Cisco 3600-series routers. On our own campus network, we have identified a correlation between multicast traffic and the IP timestamps produced by the routing module of Cisco Catalyst 6500-series products.

2.1 Measuring UDP Traffic

For the Cisco 3600-series router, we found that the timestamps returned by the IP timestamp option are correlated with the rate of UDP traffic being carried. We used a 100 Mbps network with the following configuration: host T2 was connected to the first router interface, host U2 was connected to the second, and hosts T1 and U1 were connected to the third using a LAN.

We sent 1200 ICMP Echo Requests at a rate of one per second from host T1 to host T2 with IP timestamp requests for the router's interfaces, and an additional 1200 at the same time without IP options. After 60 seconds, host U1 began to send UDP traffic at a constant rate of 5 Mbps to host U2. A minute later, the flow of UDP traffic was reversed for 60 seconds. Then, U1 began again, but at the increased constant rate of 10 Mbps. The flow was similarly reversed after the first minute. This pattern continued at 25, 35, ..., 85 Mbps. Finally, the UDP traffic was stopped for the last minute.

While the Round-Trip Time (RTT) of the `ping` packets without options was unchanged by this increasing traffic, Figure 1 illustrates the effect upon the difference between the two timestamps from the router (ΔR). Using this experimental distribution of ΔR as a guide, we can provide a bound for an unknown rate of UDP traffic being carried by such a router by issuing a series of pings with IP timestamp options.

2.2 Detecting Multicast Traffic

Multicast traffic affects the IP Timestamps produced by a Cisco Catalyst 6509, even when all hosts in the multicast group are in separate subnets from the probe traffic.

[1] http://wail.cs.wisc.edu

ACM CoNEXT Student Workshop, November 30, Philadelphia, USA.

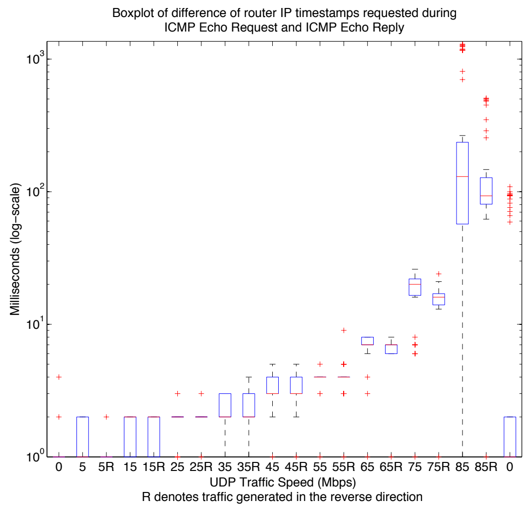

Figure 1: Boxplot of difference between router timestamps during increasing UDP traffic.

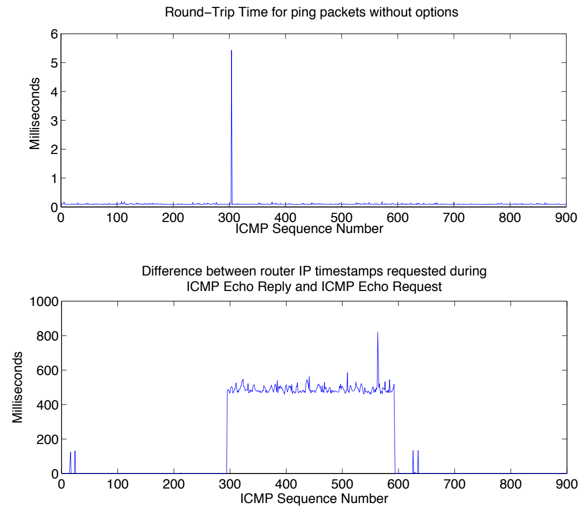

Figure 2: Delay of ping packets with options (below) and without options (above) traversing subnets D and E.

Figure 2 shows the result of an experiment focused on this router, which interconnects five subnets: A, B, C, D and E. We first constructed a multicast group with a single sender in subnet A, and 169 listeners across subnets A, B, and C. We then sent 900 ICMP Echo Requests at a rate of one per second from a host in subnet D to a host in subnet E. The minimum bandwidth on all paths was 1 Gbps.

Each `ping` requested an IP timestamp from the router as it traveled on the forward and reverse paths. We will denote these timestamps as $R1$ and $R2$ respectively. Between minutes 5:00 and 10:00 of the experiment, the multicast sender generated UDP multicast traffic at a constant rate of 8 Mbps using `iperf`.

By plotting the difference $R2 - R1$ against the ICMP sequence number, we can recover the timing of the generated multicast traffic, as shown in the lower half of Figure 2. The upper half shows the RTT for a simultaneous series of 900 pings from subnet D to subnet E which did *not* have any IP options set. As can be seen, `ping` packets without IP options were only affected at the start of multicast traffic.

We found it surprising that 8 Mbps of multicast traffic increased the processing time of packets with options by approximately 500 ms. During the experiment, the router reported no buffer misses, although the router CPU usage rose from 3% to 13%. Further experiments indicated that the IP Timestamp processing delay is related to the rate of multicast traffic by a threshold-type function; multicast traffic at a rate below 4 Mbps did not delay the IP Timestamp processing, whereas traffic above that rate always produced the approximately 500 ms delay.

3. SUMMARY AND FUTURE WORK

By using `ping` packets with the IP timestamp option, we can provide bounds for the rate of UDP traffic carried by Cisco 3600-series routers and potentially detect the start and finish of multicast traffic carried by 6500-series Catalysts. We plan to continue identifying scenarios in which the use of IP timestamps allows us to infer router or traffic statistics, with the goal of eventually performing measurements in wide-area IPv4 networks.

Measuring path latency is generally accomplished using `traceroute`, which relies upon ingress interfaces generating ICMP Time Exceeded messages. Using IP timestamp options, we have the potential to improve upon latency results from `traceroute` because we can also collect timings from egress interfaces. The use of IP timestamps also provides an alternative to `traceroute`'s ability to measure latency on known paths, which may prove useful in networks which filter ICMP Time Exceeded messages or with routers which ignore `traceroute`-style probes.

Understanding the cause of IP timestamp delays inside a single router is a continued focus of this project. In future work, we intend to separate the ingress and egress timestamps to better characterize each router. To analyze such timestamps, it will be necessary to correct for skew between the independent clocks of successive routers.

4. ACKNOWLEDGMENTS

The authors wish to thank Mike Blodgett at WAIL for supporting the Cisco 3600-series routers used in this experiment, and Jennifer Rexford for a helpful discussion during this work's preliminary stage.

5. REFERENCES

[1] RFC 791: Internet Protocol, September 1981.

[2] B.-Y. Choi, S. Moon, Z.-L. Zhang, K. Papagiannaki, and C. Diot. Analysis of Point-To-Point Packet Delay in an Operational Network. In *IEEE INFOCOM*, 2004.