Measuring the Internet Using Public Traceroute Servers

Jingping Bi, Qi Wu, Zhongcheng Li
Institute of Computing Technology, Chinese Academy of Sciences
P.O. Box 2704, Beijing 100080, P.R.China
{jpingbi, abewu, zcli}@ict.ac.cn

1. Introduction

Probes are one of the most important resources in the Internet Measurement Infrastructure (IMI). Many of current IMIs have very limited probes, which put a brake on the representativeness of the data collected. We implement an IMI `WTracer`, which uses public Web-based Traceroute Servers (WTSs) around the world to collect routing data. The number of probes `WTracer` can use is about 15 times more than current IMIs, which largely increases the completeness of data collected

Another key problem to improve in IMIs is the promotion of measurement validity. In order to enhance the validity of data collection, `WTracer` provides abnormality detection and real-time feedback mechanism which helps to find out abnormalities in measurement. We conducted a large-scale measurement and concluded that the adoption of real-time feedback greatly increases the ratio of clean data to all results.

2. WTracer Architecture

Figure 1 illustrates `WTracer`'s architecture, in which 5 major components are included: stimulator, collector, compiler, filter and analyzer. ***Stimulator*** is responsible for requesting measurement and simulates web client to send measurement requests to WTS. ***Collector*** is used to collect data from WTSs. ***Compiler*** compiles traceroute output from WTSs. ***Filter*** mainly focuses on filtering results with selected rules. The use of filter makes it possible to analyze special kinds of data, which largely increases the power of analyzer. Besides, filter supports the addition of customized rules. ***Analyzer*** is used to analyze data chosen by filter. The function of analyzer includes: hop count analysis, routing stability analysis, topology analysis, end-to-end and hop-by-hop latency analysis and so on.

This work has been supported through funding by National Natural Science Foundation of China Grant No. 90104006 and Hi-Tech Research and Development Program of China (the 863 Project) Grant No. 2001AA112135 and 2001AA112091.

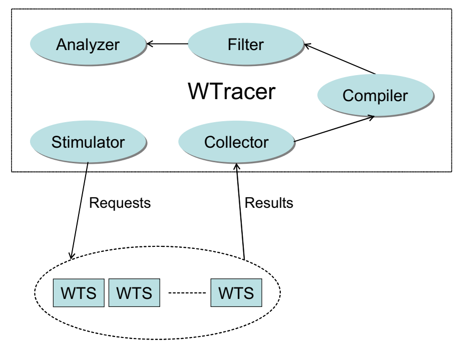

Figure 1. The architecture of WTracer

3. Abnormality Detection and Real-time Feedback

A long-time running IMI should provide enough feedbacks to users, which not only can report the current status of the results, but also can detect and deal with abnormalities in time. We classify the abnormalities in `WTracer` into 4 categories: WTS related abnormalities, `WTracer` related abnormalities, destination node related abnormalities and other abnormalities.

Many reasons can result in WTS related abnormalities. The first one is the determination of a valid WTS's address from a URL. First, if the IP address got by DNS query is an invalid address, then all connections to that address will result in timeout. Second, if a node with the IP address does not provide web access service, then all connections to the node's 80 port will be refused quickly. Third, if the web page pointed by the URL are removed or deleted, then all HTTP *get* message requesting that page will result in HTTP 404 error. Last, if the server does not provide traceroute service, then the stimulator will find the web page it obtains is not a measurement request page.

The insane of WTS also can prevent the measurement from proceeding. Fortunately, we can get the status of WTS from the measurement results. Both the halt of WTS and the crash of web daemon lead to a failing connection, but they have distinct behaviors: the halt results in no response to TCP SYN packet, and the crash causes RST

Proceedings of the 28th Annual IEEE International Conference on Local Computer Networks (LCN'03)

packet response to SYN immediately. The TCP timeout threshold is usually much larger than the latency from `WTracer` sending SYN to receiving RST (approximately the round-trip time). Thus we can use the persistent time of measurement to deduce the cause of the abnormalities in the case of connection failure.

If `WTracer` fails in connecting with the Internet, then obviously all measurement results will fall in connection failure. All WTSs halt may lead to the phenomenon, but it happens so scarcely that we can ignore it in a large scale measurement. Another possibility is the communication between `WTracer` and the outside world at HTTP layer is prohibited by a firewall. Anyhow, all results having the status of connection failure mean that `WTracer` cannot establish HTTP connection with the outside.

If the destination node is behide a firewall by which prohibits ICMP packets to pass, then all probing packets using ICMP protocol are discarded by the firewall, or the responses to the packets using UDP protocol are discarded. From the WTS's point of view, after the TTL value in the probing packet increases to a special value, it won't receive any response. And the persistent time T of the measurement will be:

$$T > (H - h) \times t \times p \quad (1)$$

H is the default maximum hop count for traceroute program, t is the default time which traceroute waits for each packet and p is the default number of packets traceroute sends for each hop. Variable h is the hop count from a WTS to a destination (that is, the number of routers a packet visits from the WTS to the destination plus 1). By default, H is 30, p is 3, t is 5, and h is set to 19 according to [1] [4]. Then T can be computed to be greater than 180 seconds which is a very large value compared with the persistent time of regular measurement results.

It is reasonable that a destination temporally loses the connection to the Internet. Several reasons such as halting, power off or router crashing can cause this. The measurements performed at this time have the same phenomenon as the destination being behide a firewall. But from the viewpoint of a longer time, these two still can be distinguished: the former has the phenomenon for some time, and the latter for all the time.

The network performance from `WTracer` to WTS can affect executing of measurements. If the network is congested, then measurement may stop at any step. We define a measurement as a *successful measurement* if the measurement result is successfully received and compiled. If the ratio of successful measurements to all measurements by a WTS is relatively low compared with other WTSs, then the path from `WTracer` to the WTS may be too bad to fit for a large scale measurement. Accordingly, we will eliminate the WTS from the WTS set according to the measurement requirements.

The feedback of `WTracer` is designed according to 4 categories of the abnormality detection. `WTracer` generates a small icon at the system tray of the task bar. After the mouse pointer stays on the icon for several hundred milliseconds, `WTracer` will report the summary of the current measurements. When `WTracer` receives a new measurement result, it will change the icon for a while according to the measurement. `WTracer` also provides more detailed feedback charts of the measurement results.

4. Measurements

We conducted our measurements with `WTracer` and 47 randomly chosen WTSs from Nov. 24, 2001 to Dec. 24, 2001 [3]. In the measurements, we get 85033 results, in which 5800 results were completely failed and 876 were incomplete (6.82% and 1.02%). Y.Zhang et al carried out similar measurements and they got 220551 results in which 11% of the results were completely failed and another 4% were incomplete [2]. The ratio of clean data to the total results in `WTracer` and Y.Zhang's measurements are 92.15% and 85% respectively, which concludes that the adoption of abnormality detection and real-time feedback mechanism enhances the efficiency of data collection.

5. Conclusion

We characterize the end-to-end routing measurement infrastructure `WTracer` and discuss the crucial technologies in this paper. Although `WTracer` is a traceroute server based Internet measurement infrastructure, it can be used in many areas, such as measurement and analysis of end-to-end routing behavior of the Internet, the estimation of the global Internet host distances, the measurement of hop count in the Internet, the discovery of Internet topology at both router level and AS level and so on.

References

[1] V. Paxson, "End-to-End Routing Behavior in the Internet", *IEEE/ACM Transactions on Networking*, 5(5), pp. 601-615, Oct.1997.

[2] Y. Zhang, V. Paxson and S. Shenker, "The Stationarity of Internet Path Properties: Routing, Loss, and Throughput", *ACIRI Technical Report*, May 2000.

[3] J. Bi, Q. Wu, and Z. Li, "Measurement and Analysis of Internet Delay Bottlenecks", *Chinese Journal of Computers*, 26(4), pp. 406-416, April 2003. (in Chinese)

[4] A. Fei, G. Pei, R. Liu, and L. Zhang, "Measurements on Delay and Hop-Count of the Internet", *Proceedings of Globecom'98*, Sydney, Australia.