Waiting for Anonymity: Understanding Delays in the Tor Overlay

Prithula Dhungel[†], Moritz Steiner[‡],
Ivica Rimac[‡], Volker Hilt[‡], Keith W. Ross[†]
[†]Polytechnic Institute of NYU, Brooklyn, NY 11201
[‡]Bell Labs/Alcatel-Lucent, Holmdel, NJ, 07733

***Abstract*—Although Tor is the most widely used overlay for providing anonymity services, its users often experience very high delays. Because much of Tor usage is for Web applications, which are sensitive to latency, it is critical to reduce delays in Tor. To take an important step in this direction, we seek an in-depth understanding of delays in Tor. By taking snapshots of the entire Tor network within a short time window, we are able to study the delay distribution of the entire router population. We also monitor delays introduced by individual Tor routers over extended periods of time. Our results indicate that apart from delays introduced by routers, overlay network latency also plays a significant role in delays in Tor. We have also observed that at any time, there exist huge differences in the delays introduced by different routers. Our results reveal key performance characteristics of Tor system behavior and provide valuable insights for improving the Tor performance.**

I. Introduction

Although Tor [1] [2] is the most widely used overlay for providing anonymity services, its users often experience very high delays [3]. Because much of Tor usage is for Web applications, which are sensitive to latency, it is critical to reduce delays in Tor.

In this paper, we seek an in-depth understanding of the delays in Tor, which is a pre-requisite for addressing Tor's poor delay performance. Specifically, we address important questions like: (i) Are the delays in Tor mainly due to delays introduced by Tor routers as a result of heavy Tor traffic, or due to the extra latency each packet has to go through when hopping around multiple Tor routers across the globe? (ii) How much delay does each packet experience in Tor? (iii) Do delays differ significantly across routers? (iv) Does a router's delay significantly vary over different time-scales?

We perform a detailed measurement study of delays in the Tor network. By taking snapshots of the entire Tor network within a short time window, we are able to study the delay distribution of the entire router population. Moreover, we monitor the delays of individual routers over extended periods of time. Our measurements reveal that there are huge differences in the delays introduced by different routers. Our study is also the first to comprehensively analyze the relative contributions of overlay latency and router delays in the overall slowness of Tor. Our results indicate that apart from delays introduced by routers, overlay latency also plays a significant role in delays in Tor. This finding should facilitate the analysis of solution space for reducing Tor delays, e.g., modifying Tor's path selection algorithm to prefer nearby routers or modifying the internal operation of the Tor routers.

II. Related Work

Apart from studies related to improving its performance [4] [5] [6] [7] [8], there have also been measurement studies related to Tor. [9] presents a performance measurement of the Tor hidden service functionality, measuring the times required for different steps in the process of accessing a hidden service. McCoy et al. [10] performed a measurement study concluding that the web traffic makes up most of the connections in Tor. To the best of our knowledge, this paper is the first in-depth measurement study of delays in the entire Tor network. Our study is also the first to comprehensively analyze the relative contributions of overlay latency and router delays to the overall slowness of Tor. Measuring and evaluating delays in Tor is complementary to earlier throughput based measurement studies.

III. Overview of Tor

To obtain anonymity, a user installs the Tor application called a Tor onion proxy (OP). The onion proxy selects 3 user-operated servers, called onion routers (ORs), to make a circuit from the OP through the ORs. The first, second, and third routers are respectively known as the entry, middle, and exit routers. Each application packet is multiply encrypted and routed through these ORs. Each OR peels off a single layer of encryption from the packet and forwards it to the next OR in the circuit. Finally, the exit router peels off the final layer of encryption and forwards the packet to the actual destination for the packet. The response packet from the destination to the OP is then routed via the same 3 routers in the opposite direction. In this manner, each OR in the circuit knows only the OR before and after it in the circuit. Therefore, the communication between the OP and the destination server is anonymous unless the entry and exit routers collude.

ORs that agree to be exit routers are referred to as "exit" routers, they also specify what kind of traffic is allowed to exit via them in the form of well defined exit policies. The centralized authorities in Tor – the directory servers (DS) – keep track of the status of the ORs; for building circuits, the OPs download the list of available ORs from the DS. Each OR reports to the DSs the peak throughput it has observed for itself in the last 24 hours, called the advertized bandwidth for the OR. When selecting routers for circuits, OPs select routers with higher advertized bandwidth with higher probabilities

compared to ones with lower bandwidths[1]. In order to limit the long-term traffic below a threshold specified by its operator, each OR uses a token-bucket approach to limit the number of bytes it relays per second. After the token bucket for a particular second has been emptied, no further cells are relayed before the token bucket is replenished at the beginning of next second.

Each OP always maintains an ordered list of nodes called "guard nodes". When choosing the first hop of a circuit, it chooses a router randomly from among the first 3 usable guard nodes. Each "exit" or "non-exit" router gets flagged as a "guard" node by the DS if it has a high uptime and has an advertized bandwidth higher than the median of advertized bandwidths of all other ORs.

IV. Delay in the Tor Overlay

In this section, we seek to a gain a deep understanding of the delay contributions from the different elements in the path when a single application packet passes through a Tor circuit.

A. Experiment Setup

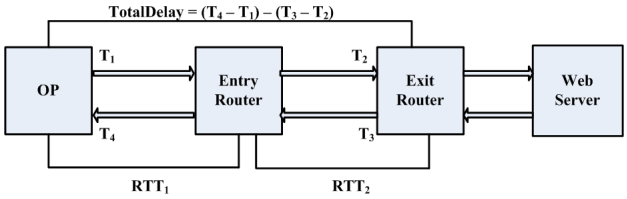

Fig. 1. Experiment Setup

As shown in Fig. 1, the experiment setup consists of four entities - an onion proxy (OP), a web server, an exit router, and an entry router[2]. The OP, web server, and exit router are kept fixed whereas the entry router is selected one-by-one from the current list of running routers in the Tor network. A single cell of 512 bytes (a TCP data packet which is part of an HTTP request) is sent from the OP to the web server via the 2-hop circuit made through the entry router and the exit router. As will be apparent later, using 2-hop instead of the default 3-hop circuits helps better decompose the delays faced by cells into various types of constituent delays.

Right before the cell is sent out from the OP, the current time (T_1) is noted. After travelling through the entry router, where it experiences processing and queuing delays, immediately after the cell arrives at the exit router, the time is noted (T_2). The cell again goes through processing and queuing delays in the exit router before it is received at the web server which then sends the response back to the exit router. Right before the response cell is sent out from the exit router, the time is again noted (T_3).

Finally, after going through new processing and queuing delays at the entry router, the response cell again arrives at the OP when the time (T_4)[3] is noted again. Immediately after this, two TCP SYN pings are sent - one from the OP to the entry router and the other from the exit router to the entry router. This procedure is repeated for different routers in the entry position, chosen one-by-one at random from the current list of running routers in Tor. The entire experiment was completed in a span of 40 minutes. We performed the experiment a number of times in a duration of 8 months between Aug. 2009 and Mar. 2010. In this section, we present results for the experiment we conducted on Mar. 23, 2010 which is a good representative of the results for all other experiments.

The round trip delay between the OP and exit router (excluding the queuing/processing delay in the exit router) is:

$$TotalDelay = (T_4 - T_1) - (T_3 - T_2) \quad (1)$$

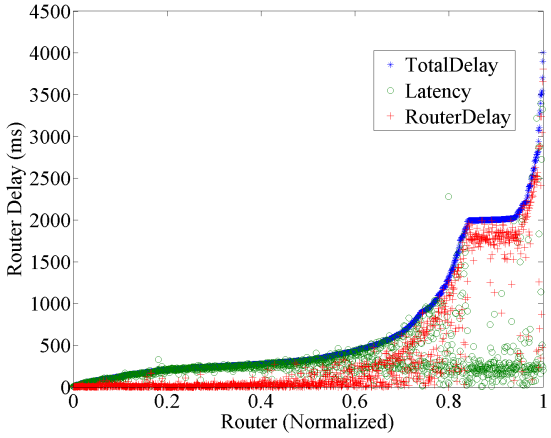

Fig. 2. Relative Contributions of Latency and Router Delay on the Total Delay (Data points sorted in ascending order of TotalDelay)

For different entry routers, Fig. 2 shows the distribution of *TotalDelay* (as well as other delays to be described subsequently). The data points have been sorted in ascending order of $TotalDelay$. To improve the accuracy of results, for each router, 10 measurements were done back to back and the average values have been plotted. In the Fig. , delay values for 1426 different routers that could be successfully pinged at least 5 times have been plotted. It can be observed that 23% of circuits have a *TotalDelay* higher than 1 second, much higher than delays observed by packets in an un-anonymized setting. In actual Tor circuits consisting of 3 routers, the delays would be even higher. We argue that $TotalDelay$ can be decomposed into two parts - (i) Delay due to latency between OP and entry router plus the latency between entry router and exit router, and (ii) Queuing and processing delays in the entry router. We refer to the latter type of delay as the *RouterDelay*, and the former type of delay as the *latency*. In the following subsection, we investigate the relative contributions of latency and router delay on the $TotalDelay$ faced by each cell.

[1]Recently, the directory authorities have started reaching a consensus on the actual bandwidth they think each router is capable of providing, based on active measurements, referred to as consensus bandwidth [11].

[2]The OP and the exit router are running on the same university network

[3]For T_1 and T_4, time is noted right after the entire cell is written into the output buffer of the OP-OR connection and right after the entire cell is read from the input socket of the connection, respectively. For T_2 and T_3, time is noted right after the entire cell is read from the input socket of the OR-OR connection and right after the entire cell is written into the output socket, respectively.

B. Relative Contributions of Router Delay and Latency

We introduce RTT_1 and RTT_2, which denote the RTTs for the TCP SYN messages from OP to entry router and from exit router to entry router, respectively. The *latency* and *RouterDelay* observed by each cell are given by:

$$L_D = RTT_1 + RTT_2 \quad (2)$$

$$RouterDelay = TotalDelay - L_D \quad (3)$$

For each entry router, Fig. 2 also shows the relative contributions of router delay and latency in the $TotalDelay$ faced by a Tor cell. It can be observed that for most cases when $TotalDelay$ is high (> 1 sec), the router delay constitutes most of $TotalDelay$. Furthermore, it can be seen that delays introduced by different routers vary from a few milliseconds up to several seconds. Specifically, 61% of routers introduced router delays less than 100 ms whereas 18% of them had delays of 1 second or more.

We performed the same experiment 8 times within 24 hours. The shape of the curve was the same in all the rounds. There were 60 router IPs that were successfully contacted during all 8 rounds. 7% of these routers had consistently high delays throughout (> 1 sec), 17% had low delays throughout, and the rest of the routers had delays fluctuating from a few tens of milliseconds to a few seconds. This means a large fraction of the routers (76% here) have delays that dramatically fluctuate over a 1-day period. The possible reasons for such fluctuations are as follows: (i) The Tor router selection algorithm itself causes fluctuation in the amount of Tor traffic passing through any router. (ii) The machine the router is running on is running other applications and so there exist fluctuations in the network traffic through the router and/or fluctuations in the CPU load in the router (more on this in later sections).

Cells passing through any circuit that has one or more high-delay routers will face high round trip delays. Furthermore, even though the router delay seems to be the major contributor, 7.5% of the cases have latency of 450 ms or more. Therefore, we conclude that overlay latency can also play a significant role in the delays observed by Tor cells. *For all the snapshot experiments that we conducted from August 2009 through March 2010, the delay distributions observed for Fig. 2 (as well as Fig. 3 and 4 to be discussed subsequently) were very similar without drastic deviations on fractions of routers with high and low delays.*

In Fig. 3 we depict the delay distributions of "guard", "exit", and "non-exit" routers.The total number of routers of each type that were successfully contacted are indicated in the legend. Although the delay distributions of "exit" and "non-exit" routers do not differ significantly, a large fraction of "guard" routers have high delays. 40% of "guard" routers have delays of 1 second or more. Also, 33 out of 47 routers with delays higher than 2 seconds are "guard" routers.

V. Router Delay Analysis

In this section, we perform a more detailed analysis of router delays. Specifically, we check the variation in router delays over time. We also seek to understand the correlation, if any, of router delays with the corresponding router bandwidths.

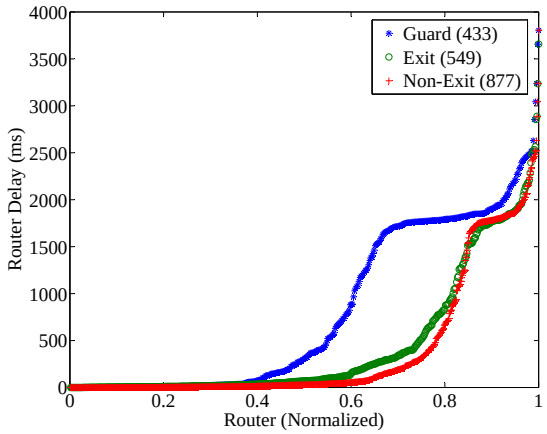

Fig. 3. Router Delay Distribution for "guard", "exit" and "non-exit" Routers

A. Correlation with Advertized Bandwidth

Fig. 4 shows the scatter plot of router delays and the corresponding advertized bandwidths on the left axis and consensus bandwidths on the right axis. There are 3 key observations: (i) 28 out of 29 routers (2% of all routers measured) with advertized bandwidths of 2 MB/sec or higher have delays mostly in the order of a few hundred milliseconds. Fig. 5 further shows that delays for a high bandwidth router (8 MB/sec; monitored over an extended period of time) are always in the order of only a few hundred milliseconds. These results are in agreement with the theoretical claim in [3] that the high bandwidth routers are selected with a lower probability compared to an optimal router selection strategy. (ii) 39 (2.7%) out of 47 routers with delay values more than 2 seconds have bandwidths equal to 150 KB/sec or less. This indicates that routers with highest delays are generally those with low bandwidths. (iii) *However, for majority of the routers (95%), there is very low correlation between the advertized bandwidth of a router and its delay.* The correlation between router delay and consensus bandwidth is also very low.

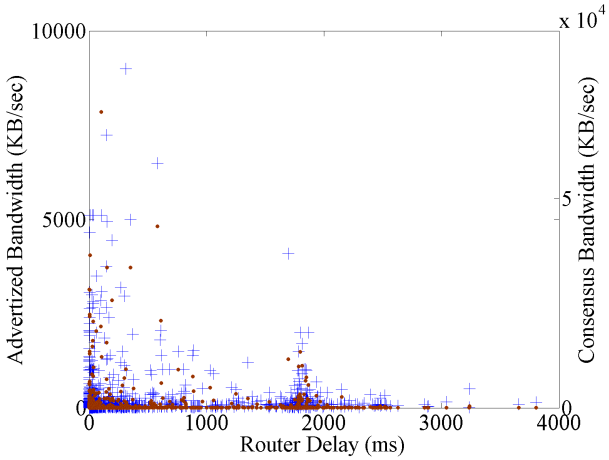

Fig. 4. Router Delay vs. Advertized Bandwidth and Consensus Bandwidth

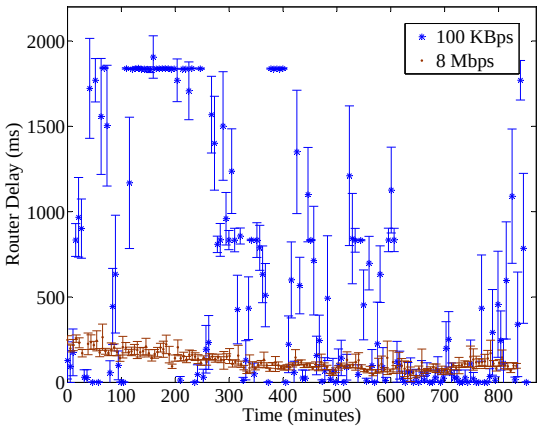

Fig. 5. Average Router Delay over Extended Time Period for High and Low Bandwidth Routers (95% confidence interval)

B. Variation in Router Delays

Fig. 5 also presents the delays for a low bandwidth router (100 KB/sec) over extended periods of time. The router was set up in our research lab network, conigured as a "non-exit" router. In order to avoid loading the router with other applications, there were no extra programs running on the router. In the figure, the average delay faced by 15 measurement cells every 5 minutes has been plotted. The delays in this case are very fluctuating. Since the router showed dramatic fluctuations in delay over time even with no extra interference, it is very likely that the Tor router-selection algorithm itself plays a role in the variation in delays across a given router.

C. Impact of Tor Token Buckets

Notice that a number of routers in Fig. 2 have delay values very close to 2 seconds. Also, Fig. 5 has a number of points around the 2 second mark for the low bandwidth router. The logs at our research lab network router indicated that for cases when the router delay observed was close to 2 seconds, the Tor cell travelling from OP to the research lab router and the response cell from the exit router to the research lab router had to each wait for almost a second for relay read/write tokens to be available before they could be read from or written to the corresponding *input/output* socket buffers.Note the narrow confidence intervals around the points close to the 2 seconds mark. This suggests that the router was handling a lot of Tor relay data at that point of time. The narrow confidence intervals for routers close to the 2 second mark in Fig. 2 (not shown here due to lack of space) also indicate that these routers were most likely handling large volumes of Tor data and therefore the cells in all 10 measurements for each router were blocked by empty token buckets. The fraction of such routers (11% in Fig. 2 with delay equal to 1.7 seconds or more and confidence interval equal to 400 ms or less) gives a lower bound for the fraction of routers overloaded with Tor traffic.

VI. Conclusion

To take an important step in improving the perceived delays in the Tor overlay, a thorough understanding of its delays is required. In this paper, we perform a detailed measurement study of delays in the entire Tor network. Our key findings are as follows: (i) Router delays are the principal contributors to delays in Tor. Some routers frequently introduce delays as high as a few seconds. At any instant of time, we observed 11% or more of the routers to be overloaded with Tor traffic. (ii) The router delay is not the only culprit. In almost 7.5% of circuits the overlay latency contributed more than 450 ms, which is much higher than delays in an un-anonymized setting. (iii) At any point in time, there exist huge differences in the delays introduced by different routers. (iv) In general, "guard" routers introduced higher delay values than "non-guard" routers. (v) Except for the routers with very high advertized bandwidths, there is no correlation between the delay introduced by a router and its advertized or consensus bandwidths. (vi) Except for the routers with very high advertized bandwidths, the delays for the routers dramatically fluctuate over time, ranging from a few milliseconds up to several seconds. This fluctuation is introduced by the Tor network itself and not due to the fluctuation in load from non-Tor applications that might be running in the Tor routers. (vii) In the current router design, the cells often sit waiting for relay tokens to be available in the next time slot, before they can be read from or written to TCP socket buffers. This phenomenon occurs frequently when the router is handling a large amount of Tor traffic.

Our findings should facilitate the analysis of solution space for reducing Tor delays. For example, modifying OPs to actively keep track of delays introduced by different routers and choosing routers with low delay values for circuits serving delay-sensitive applications can be one approach to improve perceived delays; in doing so, piggy-backing delay measurement messages with Tor protocol messages might be necessary to minimize overhead. Similarly, making the criterion for a router to be promoted into a "guard" to be less stringent might help by distributing loads across more "guards".

References

[1] "Tor," http://www.torproject.org/index.html.en.
[2] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The Second-Generation Onion Router," in *Proc. of the 13th USENIX Security Symposium*, 2004.
[3] R. Dingledine and S. Murdoch, "Why is Tor slow and what are we going to do about it," The Tor Project, Inc., Tech. Rep., 2009.
[4] L. Øverlier and P. Syverson, "Improving efficiency and simplicity of Tor circuit establishment and hidden services," in *Proc. of PETS*, 2007.
[5] R. Snader and N. Borisov, "A tune-up for Tor: Improving security and performance in the Tor network," in *Proc. of NDSS*, 2008.
[6] S. J. Murdoch and R. N. M. Watson, "Metrics for security and performance in low-latency anonymity networks," in *Proc. of PETS*, 2008.
[7] A. Panchenko and J. Renner, "Path selection metrics for performance-improved onion routing," in *IEEE/IPSJ International Symposium on Applications and the Internet*, 2009.
[8] J. Reardon and I. Goldberg, "Improving Tor Using a TCP-over-DTLS Tunnel," in *Proc. of 18th USENIX Security Symposium*, 2009.
[9] K. Loesing, W. Sandmann, C. Wilms, and G. Wirtz, "Performance Measurements and Statistics of Tor Hidden Services," in *Proc. of SAINT*, 2008.
[10] D. Mccoy, K. Bauer, D. Grunwald, T. Kohno, and D. Sicker, "Shining light in dark places: Understanding the Tor network," in *Proc. of PETS*, 2008.
[11] "Authorities vote for bandwidth offsets in consensus," https://git.torproject.org/checkout/tor/master/doc/spec/proposals/160-bandwidth-offset.txt.