Reducing the Circuit Window Size in Tor

Karsten Loesing
karsten@torproject.org

Tor Tech Report 2009-09-002
September 20, 2009

1 Introduction

As one approach to make Tor faster, the circuit window size can be reduced from currently 1000 cells to only 101 cells. In theory, the effect is reduced latency for client connections, because the number of cells that are buffered in the Tor network is smaller. This report analyzes the effect in practice.

2 Measurement Setup

The measurement setup consists of 1 webserver serving a 40 KiB, a 50 KiB, and a 1 MiB file, 2 Tor exit relays, 2 Tor clients, and 2 SOCKS clients measuring download times. One of the exit relays is compiled with a reduced circuit window size, the other one is built without any code changes. In detail, the measurement setup is as follows:

- The *webserver* that serves the 40 KiB, 50 KiB, and 1 MiB test files is `www.freehaven.net`. The test files containing random data are available under `http://www.freehaven.net/~karsten/perf/.40kbfile`, `.50kbfile`, and `.1mbfile`.
- The *first exit relay* that is compiled with a reduced circuit window size is `echelon1` with fingerprint `5114 61B1 8F3A 5CA0 3655 8C9A 8FDD 22B3 D0A8 1EBB`. Its Tor version is 0.2.2.0-alpha-dev. It is configured to permit exiting using the default exit policy. It has been running as non-exit relay for a few days before starting these measurements.
- The *second exit relay* is using Tor version 0.2.2.0-alpha-dev, too, but compiled without code changes. Its nickname is `echelon2`, and its fingerprint is `3EED 5706 22FC C06D EF4D 4204 2DD8 47D8 9989 3972`. It is configured to use the default exit policy and was running as non-exit relay for a few days, too. Both exit relays are running on the same host located in Canada.

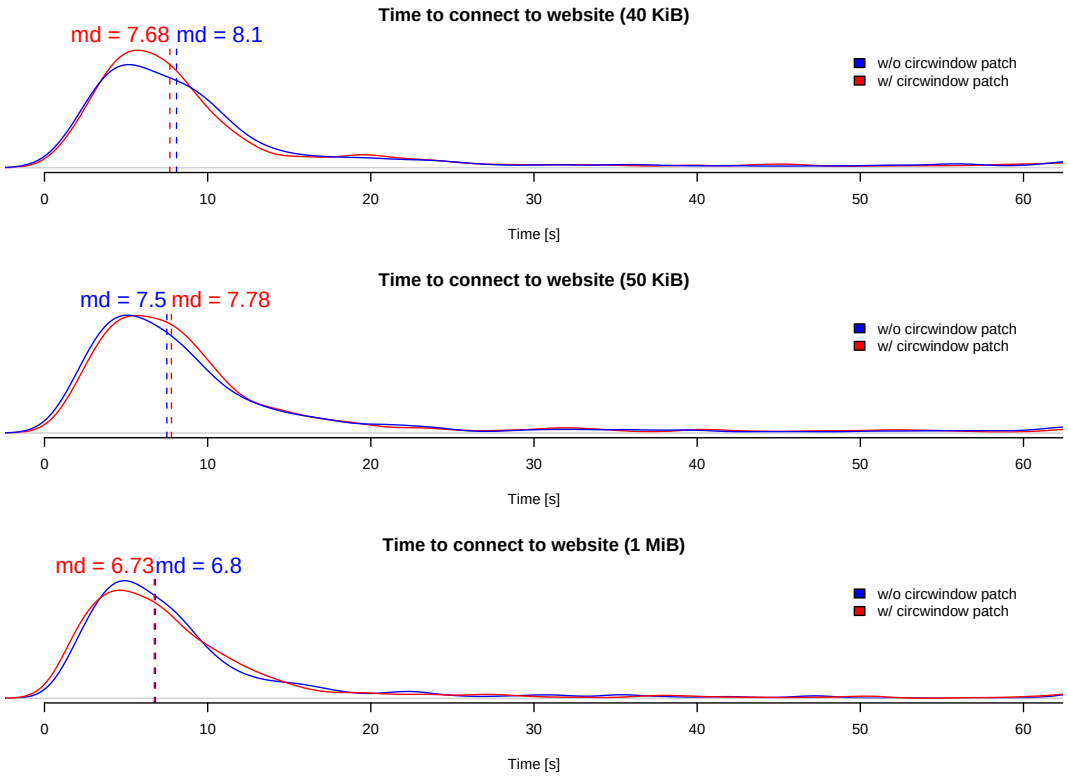

Figure 1: Kernel density estimates of times to connect to website

- The *two Tor clients* are configured to use no entry guards and to use circuits only with a maximum dirtiness of 1 minute. Both Tor clients are running version 0.2.2.0-alpha-dev. They are located on the same host as the two exit relays.

- The *two SOCKS clients* are the trivial SOCKS clients used for other `torperf` measurements, too. In contrast to previous measurements, the clients specify an exit relay using the `.exit` notation.

3 Results

Results include three data points: The first measured time is the time to connect to the website. This step requires the Tor client to create a circuit using the specified exit and send a `CONNECT` command to the exit relay. As soon as the exit relay replies with a `CONNECTED` cell, the connection to the website is established. Figure 1 shows kernel density estimates of the connection times.

The second measured time is the time between starting a request and receiving the first byte of the response. Figure 2 shows these times for the three requested file sizes.

The third measured time is the overall time to complete a request. This time include all steps from connecting to the website until having received all bytes of the response. Figure 3 shows the distributions of these times.

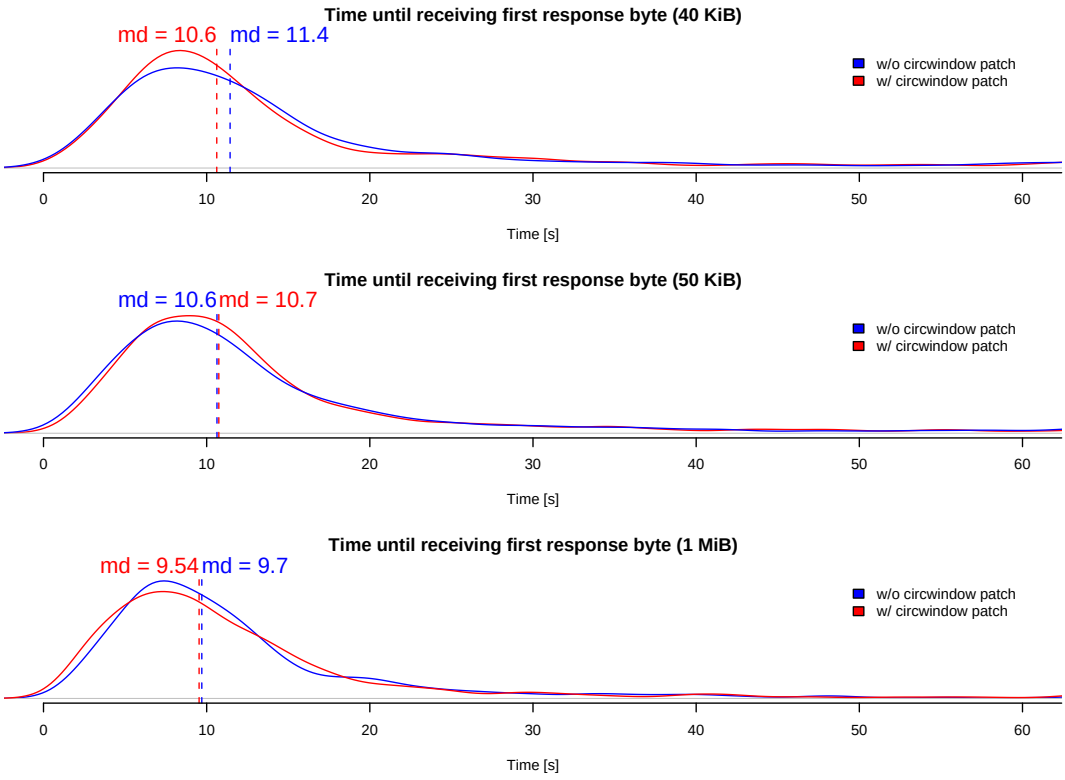

Figure 2: Kernel density estimates of times until receiving first response byte

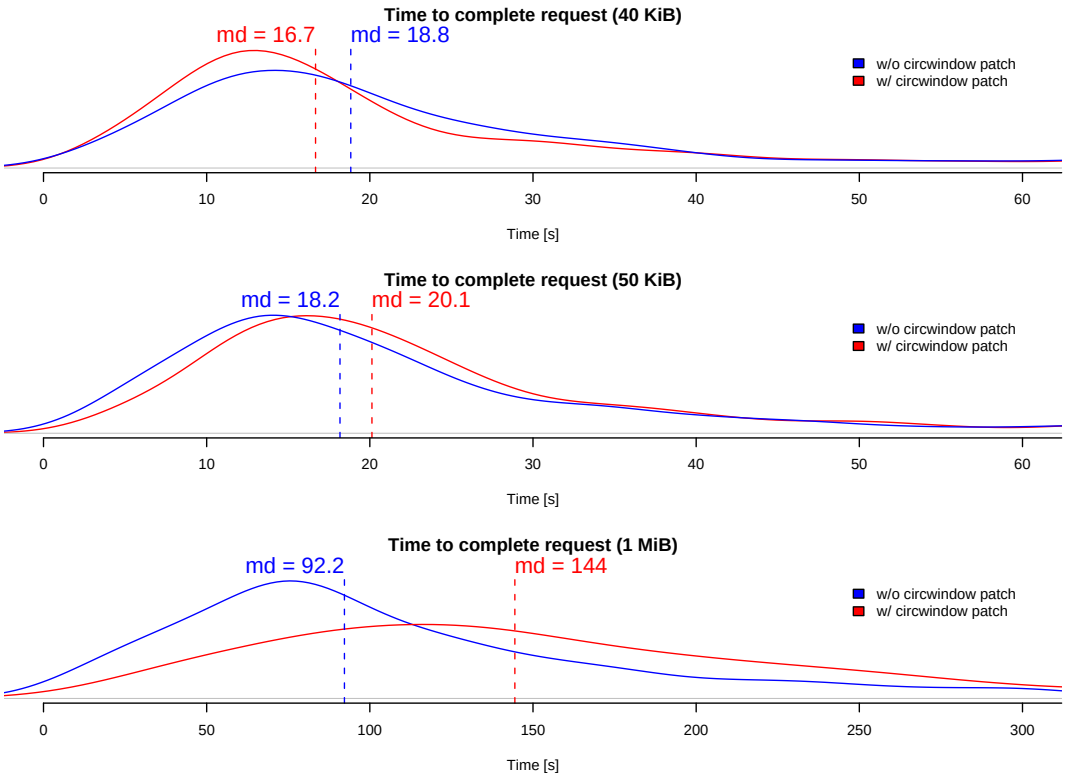

Figure 3: Kernel density estimates of times to complete request

4 Discussion

The results, especially the time to complete requests, indicate an improvement of the circuit window reduction to 101 cells for downloads of 40 KiB files at the cost of significant decreasing performance for 1 MiB files. A possible explanation is that 40 KiB downloads fit into one circuit window of 101 cells in contrast to 50 KiB and 1 MiB downloads. The latter two require additional round-trips for the 101 cells circuit window.

However, these are only early measurements of 1000+ downloads for each setting (or even 820 for 1 MiB downloads) that need to be confirmed in the future. Results might also differ as soon as a certain number of other relays in the network have upgraded to smaller circuit windows. As an alternative to changing the circuit window from 1000 to 101 cells in the code, the circuit window size could be included in the network status consensus. That way, the optimal circuit window size could be switched easily as soon as a certain fraction of relays has upgraded to use the included circuit window size.