Introduction

Tor Metrics team report on the work done to complete Milestones 4 and 5.

Milestone 4: Tor daemon

4.1. Perform an analysis on reducing the amount of sensitive, potentially personally identifying data stored in memory of Tor relays and bridges or reported to the directory authorities.

We published a 26 page long Tor Technical Teport where we analyzed which possibly sensitive, potentially personally identifying data is stored in memory of Tor relays and bridges or reported to the directory authorities and made suggestions to reduce the collection and temporary storage of such data.

https://research.torproject.org/techreports/privacy-in-memory-2017-04-28.pdf

4.2. Reduce the amount of sensitive, potentially personally identifying data stored in memory of Tor relays and bridges by implementing at least one suggestion from the earlier analysis document.

4.3. Obfuscate data that gets reported by Tor relays and bridges to the directory authorities by implementing at least one suggestion from the earlier analysis document.

To achieve the activities above (4.2 and 4.3) we decided to implement both goals at the same time. Although, as described in detail below, we realized after a lot of work trying to create a first implementation of this solution, that the planned time or allocated resources for this work was estimated way lower than we now realize is actually required.

The idea was to apply differential privacy techniques by adding statistical noise to data stored in memory and later reported to the directory authorities. But we wanted to be sure that the resulting statistics will still be as useful as before, so we ran a simulation with archived Tor data.

The result of this test showed that we cannot go ahead and implement this noise without first working on statistical aggregation algorithms. The main reason is that when we did this implementation, user number estimates for any given country and day varied by a few hundred users.

This won't matter for countries with thousands or tens of thousands daily users including the top-10 countries by relay users, as shown in this image:

Country	Mean daily users
United States	416658 (20.70 %)
United Arab Emirates	244764 (12.16 %)
Russia	216307 (10.75 %)
Germany	172854 (8.59 %)
France	95197 (4.73 %)
United Kingdom	74236 (3.69 %)
Ukraine	73668 (3.66 %)
Canada	41165 (2.04 %)
Netherlands	40691 (2.02 %)
Italy	39977 (1.99 %)

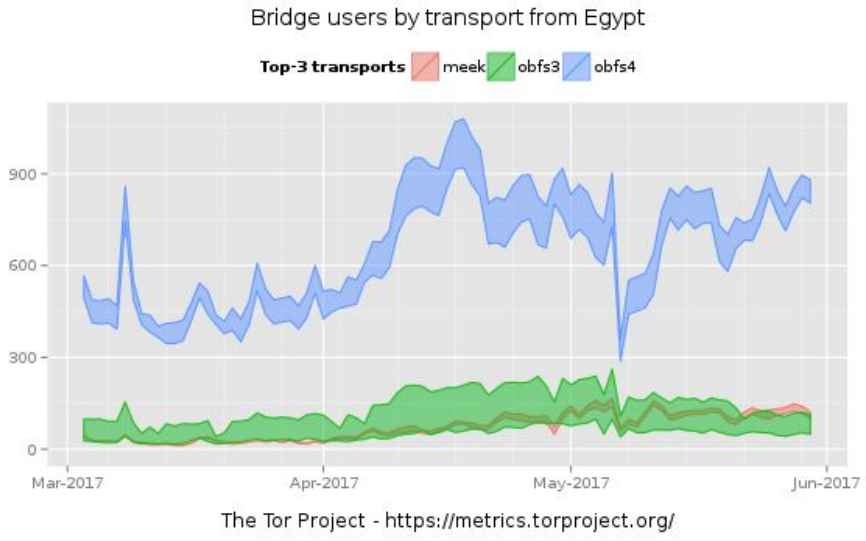
But it might affect the top-10 countries by bridge users:

Country	Mean daily users
United Arab Emirates	36949 (41.46 %)
Russia	7289 (8.18 %)
United States	6819 (7.65 %)
Turkey	3963 (4.45 %)
India	2892 (3.24 %)
Ukraine	2834 (3.18 %)
United Kingdom	2319 (2.60 %)
Germany	2091 (2.35 %)
Iran	2050 (2.30 %)
Belarus	1752 (1.97 %)

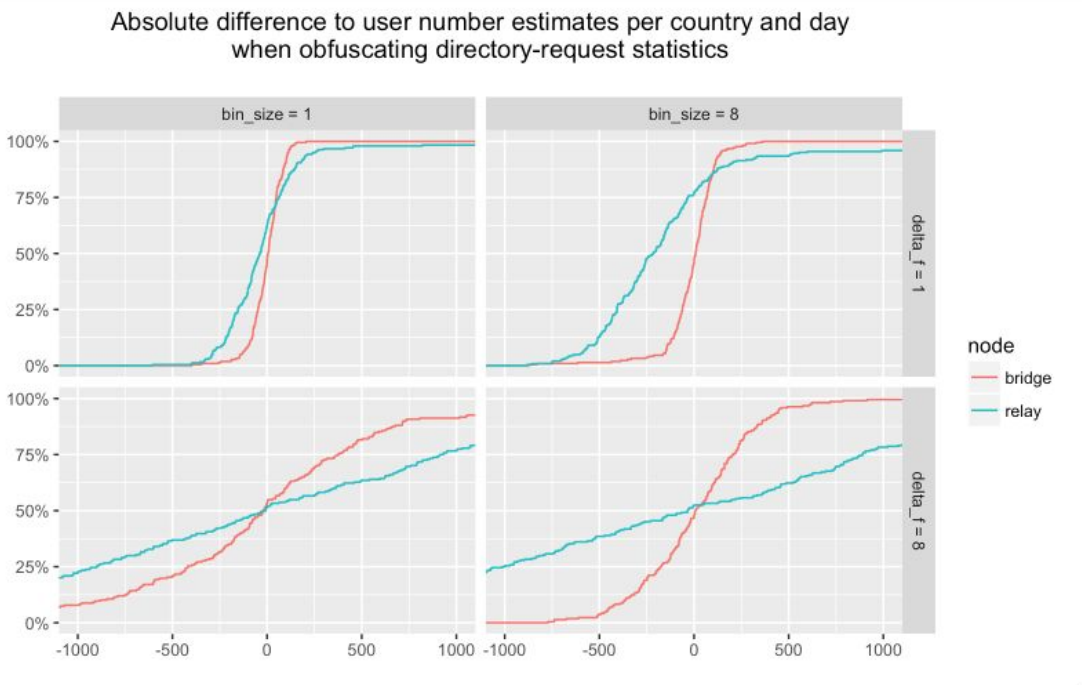
And it will quite certainly affect the top-10 countries by possible censorship events:

Country	Downturns	Upturns
Turkey	27	27
Egypt	14	14
Israel	11	11
Bosnia and Herzegovina	9	13
Macedonia	8	9
Taiwan	7	15
Republic of Moldova	7	7
Ethiopia	5	4
Lesotho	4	4
C??te d'Ivoire	4	3

For instance, Egypt is the second country with more censorship events, and the number of users who are using our circumvention tools is in the hundreds. Such implementation without first working on statistical aggregation algorithms will definitely affect this number.



We made the simulation source code and results available on a wiki page to document our attempts of implementing this solution: https://trac.torproject.org/projects/tor/wiki/org/teams/MetricsTeam/ObfuscationSimulationAnalysis



Even though we are not fully implementing this solution, this work is still very useful for us and

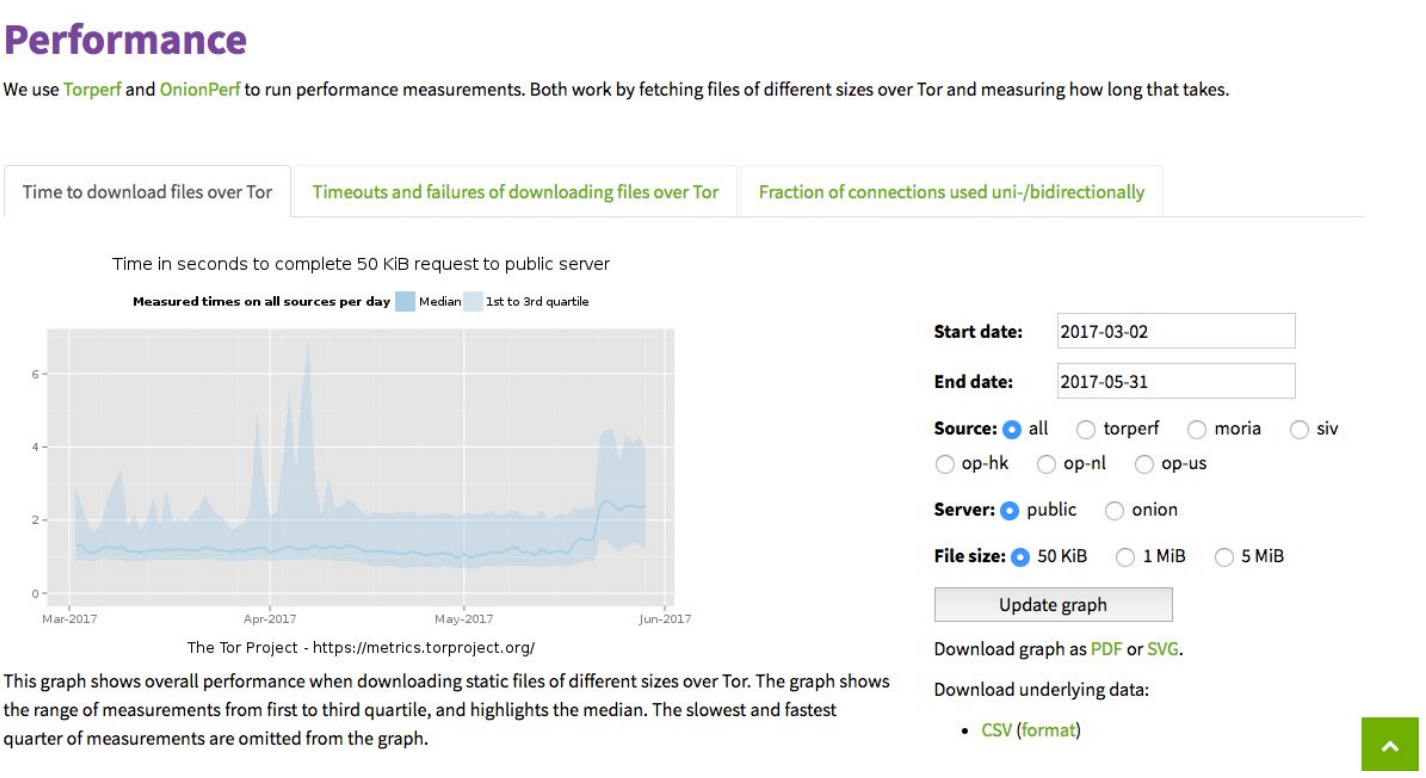
we decided to use all this work, the technical report together with these simulation results, as a solid foundation to start a new research and development project in the future.

Milestone 5: OnionPerf

5.1. Replace all existing Torperf instances gathering current Tor network performance measurements with OnionPerf instances.

We created three OnionPerf instances on virtual machines in Hong Kong, the Netherlands, and the U.S. Shortly after we retired the three Torperf instances running in the U.S. and in Sweden. We included the new data in our CollecTor service and in visualizations on the Tor Metrics website (sources op-hk, op-nl, and op-us):

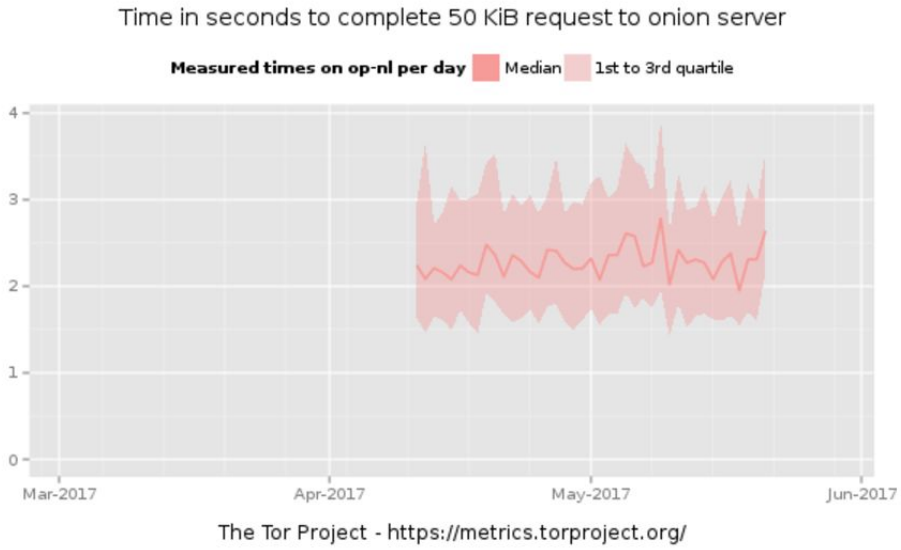
- https://metrics.torproject.org/collector.html#type-torperf
- https://metrics.torproject.org/torperf.html



5.2. Develop and deploy at least one more user model in addition to the current model.

With the three new OnionPerf instances we don't just measure performance to public servers but also to onion servers, which was not possible by default with Torperf. For example, the graph below shows how long it takes for the OnionPerf instance in the Netherlands to download a 50 KiB file over the Tor network from its own onion service:

https://metrics.torproject.org/torperf.html?source=op-nl&server=onion&filesize=50kb


Time in seconds to complete 50 KiB request to onion server
Measured times on op-nl per day
Median
1st to 3rd quartile
4
3
2
1
0
Mar-2017
Apr-2017
May-2017
Jun-2017
The Tor Project - https://metrics.torproject.org/