Quantifying measurement quality and load distribution in Tor

Andre Greubel
University of Wuerzburg
Wuerzburg, Germany
andre.greubel@uni-wuerzburg.de

Steffen Pohl
University of Tuebingen
Tuebingen, Germany
steffen.g.pohl@web.de

Samuel Kounev
University of Wuerzburg
Wuerzburg, Germany
samuel.kounev@uni-wuerzburg.de

ABSTRACT

Tor is a widely used anonymization network. Traffic is routed over different relay nodes to conceal the communication partners. However, if a single relay handles too much traffic, de-anonymization attacks are possible. The Tor Load Balancing Mechanism (TLBM) is responsible for balanced and secure load distribution. It must verify that relays cannot attract more traffic than they should by lying about their self-reported bandwidth. This work shows that the current bandwidth measurement method used for bandwidth verification is not suitable to verify the bandwidth of many relays. Most importantly, multiple measurements of high-bandwidth relays are uncorrelated to each other. Furthermore, we analyze the current load distribution in Tor. We show that the current load distribution reduces the resources necessary for several large-scale de-anonymization attacks by more than 80%. Additionally, as Tor favors fast relays during path selection, verifiable relays only handle a small fraction of Tor's traffic. More precisely, we show that only 7.21% of all paths consist of entry and exit relays verifiable by measurements. We discuss these results' security implications and argue that future TLBM research should focus at least as much on secure load distribution as on high traffic performance.

CCS CONCEPTS

• **Security and privacy** → **Pseudonymity, anonymity and untraceability**; • **Networks** → *Network privacy and anonymity*.

KEYWORDS

Privacy, Tor Network, Bandwidth Measurement, Load Distribution

ACM Reference Format:
Andre Greubel, Steffen Pohl, and Samuel Kounev. 2020. Quantifying measurement quality and load distribution in Tor. In *Annual Computer Security Applications Conference (ACSAC 2020), December 7–11, 2020, Austin, USA.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3427228.3427238

1 INTRODUCTION

In times where the freedom of the press reached a new low around the world, protecting sources by concealing communication to journalists is more important than ever [24]. A common way to conceal this information is the Tor network, a low-latency anonymous communication network designed to conceal the partners of a communication stream over the internet. It is designed to provide a reasonable trade-off between anonymity, usability, and efficiency [8]. In Tor, clients can build privacy-preserving paths (*circuits*) consisting of relay nodes (*relays*) that forward encrypted TCP packets to other relays in a circuit or destination server on the internet. This way, only the communication from the client to Tor and from the last relay to the destination can be observed.

As the available bandwidth of each relay is limited, the *Tor Load Balancing Mechanism* (TLBM) has to ensure that no relay attracts more traffic than it can handle. Additionally, de-anonymization attacks are possible if a single relay attracts much traffic (cf. Section 5). Hence, a balanced distribution of traffic to relays is crucial to Tor's security.In the current TLBM, some entities act as *Bandwidth Authorities* (BAs), measuring the available bandwidth of relays. Afterward, the BAs use these measurements and the relays' self-reported bandwidth to calculate a *consensus bandwidth* for each relay. By default, clients select relays for their circuits proportionally to this value. This process can also be interpreted as a *verification* of the self-reported bandwidth by the BAs: It should detect relays offering less bandwidth than they advertise. Most importantly, it should ensure that a relay cannot increase its traffic by lying about its self-reported bandwidth.

The first goal of this paper is to evaluate the quality of the TLBM measurement process. To do so, we collected more than 1.1 million bandwidth measurements of relays. This data collection is necessary as Tor does not publish the raw measurement results of the BAs. Afterward, we perform a large-scale evaluation of this data and data published by the Tor foundation. The results show that the current design of the measurement script is based on false assumptions and, as such, not suitable to verify the bandwidth values of many relays in Tor. More precisely, we show:

(1) The measurement results are, for high-bandwidth relays, independent both from the self-reported bandwidth and prior measurements of this relay. As such, they are not a suitable metric to calculate the consensus bandwidth.
(2) The consensus bandwidth is, for high-bandwidth relays, primarily influenced by the (untrusted) self-reported bandwidth, rather than the measurement results.

Both results contradict the core assumptions of the current load balancing process in Tor. As the measurement quality and security guarantees of the TLBM are firmly connected, the second goal of this paper is to evaluate the resulting implications for the anonymity of Tor users. We do so by introducing three large-scale de-anonymization attacks targeting the TLBM that reveal (i) the client that created the circuit, (ii) the data transmitted by a circuit,

ACSAC 2020, December 7–11, 2020, Austin, USA

ACM ISBN 978-1-4503-8858-0/20/12.
https://doi.org/10.1145/3427228.3427238

or (iii) both. Additionally, we build more than 12 million circuits to analyze the load distribution in Tor. We show that:

(3) As of March 2019, only 7.21% of Tor's traffic is handled by circuits whose relays can be verified by measurements.
(4) Tor's load distribution reduces the resources necessary for these large-scale de-anonymization attacks by more than 80% and up to 96.9% for attack (ii).
(5) These issues did not improve after the adoption of the new measurement script in 2019.

The remainder of the paper is structured as follows: Section 2 provides background information on Tor and the TLBM. Section 3 introduces our terminology for several definitions for the bandwidth of a relay. Section 4 contains our evaluation of the quality of bandwidth measurements in Tor. Section 5 describes the theoretical implications of our results for the security of Tor and introduces the three large-scale de-anonymization attacks. Section 6 quantifies the reduction in attack resistance caused by the current load distribution and problems described earlier. Section 7 to 9 discuss results and related work and conclude this paper.

2 BACKGROUND

This section describes Tor, its topology distribution, and the load balancing mechanism (TLBM) in more detail.

2.1 The Tor network

Tor is designed to ensure that every relay only knows about its predecessor and successor. By default, three relays are used for each circuit (called a *three-hop* circuit). Only the first relay (the *entry* of the circuit) knows about the client, and only the last relay (the *exit* of the circuit) knows about the destination. As such, an attacker only able to eavesdrop on a single point-to-point communication cannot infer both the source and destination of the communication.

To build a circuit, knowledge about the available relays is needed. This information is collected by the *Directory Authorities* (DAs).

2.2 Topology Information Distribution

In order to join the Tor network, a new relay publishes a *Server Descriptor*, including information like its internet address and whether and how it can be used as an exit. As different relays have different bandwidth capacities (their *available* bandwidth), load balancing is necessary. To support this process, relays include self-reported bandwidth values in their descriptor. Most importantly, they include the *advertised* bandwidth, which is the (untrusted) claim of a relay about its available bandwidth. These descriptors are then collected by the DAs, which independently publish part of this information in a network document called a *vote*. Periodically, the DAs exchange their vote documents and then aggregate this information to another network document called the *consensus* (specified in [30], archived at [10]). Clients can download this consensus from the (publicly known) DAs or fetch a cached consensus elsewhere.

Currently, the votes are published and aggregated every hour. As DAs might become compromised, the consensus only becomes valid for at most three hours after more than half of the current nine DAs signed it. During aggregation, numerical data like bandwidth information is aggregated by using the (lower) median value. Other information is aggregated by a vote of majority.

Table 1: Different aspects of the bandwidth of a relay.

Category	Bandwidth Name	Notation
real-world	(1) used bandwidth	bw_{use}
	(2) available bandwidth	bw_{avl}
	(3) measured bandwidth	bw_{mes}
self-reported	(4) advertised bandwidth	bw_{adv}
	(5) burst bandwidth	bw_{burst}
	(6) observed bandwidth	bw_{obs}
trusted information	(7) published bandwidth	bw_{pub}
	(8) proposed bandwidth	bw_{pro}
	(9) conensus bandwidth	bw_{con}

2.3 The Tor Load Balancing Mechanism

In earlier versions of Tor, the DAs would simply trust the relays' advertised bandwidth values, as long as they are below a specific, publicly known limit ($10000 KB/s$). With the introduction of the *TorFlow* [22] project, a subset of the DAs also act as Bandwidth Authorities (BAs). These BAs perform bandwidth measurements and use these measurement results (the *measured* bandwidth) to adjust the relays' advertised bandwidth, resulting in a *proposed* bandwidth that they publish in their vote documents. The corresponding aggregated value published in the consensus is called the *consensus* bandwidth. By default, clients then select relays for their circuits proportionally to this value.

Before 2019, the *speedracer* measurement script was used to calculate the proposed bandwidth value [22]. It divides the network into slices of relays with similar bandwidth and then repeatedly fetches a large file over two-hop circuits of relays in this slice. The average provided bandwidth compared to relays in the same slice (a value close to one) is determined as a *bandwidth factor*. The proposed bandwidth is calculated as the advertised bandwidth of a relay multiplied by the bandwidth factor.

During 2019, the *Simple Bandwidth Scanner* (sbws) replaced the speedracer measurement script [28]. However, the main reason for this change were issues like outdated dependencies of the former implementation [23], and the general approach did not change much: This script also builds random two-hop circuits between relays. However, different from speedracer, a random relay with at least twice as much bandwidth as the first one is chosen as the second hop. If there are none, this requirement is lowered. Hence, the bandwidth factor is now calculated relative to *all* relays. The proposed bandwidth is then calculated as before and additionally published in a *bandwidth file* (specified in [29], archived at [10]).

3 BANDWIDTH VALUES IN TOR

The bandwidth of a Tor relay can be defined in multiple ways. Unfortunately, prior research and Tor's documentation are not consistent in their terminology. This section explains the terminology used in this paper for the bandwidth definitions most relevant for the TLBM. A summary is provided in Table 1.

These definitions fall into three categories: The used, available, and measured bandwidths are real-world bandwidth information only obtainable by monitoring relays. The advertised, burst, and observed bandwidth values describe the (untrusted) self-reported

values published in the Server Descriptors. Lastly, the published, proposed, and consensus bandwidths are trusted information published in the network documents. However, "trusted" does not necessarily imply that these values are accurate.

(1) Used Bandwidth (bw_{use}) This value describes the sum of the bandwidth provided to all clients served by this relay in a time period. This metric is the best description of a relay's bandwidth, but it must not be published as it would de-anonymize traffic.

(2) Available Bandwidth (bw_{avl}) This value describes the maximum bandwidth this relay can provide, including both hardware (limited internet access speed) and software (maximum configured bandwidth) limitations.

(3) Measured Bandwidth (bw_{mes}) This value describes the bandwidth provided to a single client in a single circuit, e.g., during measurement. Values are obtained by using a particular client in a circuit and measuring the time necessary to fetch data over it.

(4) Advertised Bandwidth (bw_{adv}) This value describes the average bandwidth a relay is willing to provide over a long period.

(5) Burst Bandwidth (bw_{burst}) This value describes the maximum bandwidth a relay is willing to provide over short periods.

(6) Observed Bandwidth (bw_{obs}) This value is an estimation of a relay about the bandwidth provided to clients. This is currently roughly equivalent to the maximum bandwidth provided to a client in a ten-second period over the last five days.

(7) Published Bandwidth (bw_{pub}) This value describes the advertised bandwidth, as published in the vote document of a DA after being collected from the server descriptor. Values of the advertised bandwidth above $10000KB/s$ are published as $10000KB/s$.

(8) Proposed Bandwidth (bw_{pro}) This value represents a measurement script's output, as published in a vote document of a BA. It is equivalent to the advertised bandwidth, multiplied by the bandwidth factor. While this number is published as "measured" value in the vote documents, it is not equal to the raw measurement result, which we call the measured bandwidth.

(9) Consensus Bandwidth (bw_{con}). This value is defined as the lower median value of all proposed bandwidth values of the corresponding votes. By default, clients choose relays proportionally to this value while building circuits.

Unless explicitly stated otherwise, we will always present values in KB/s. Additionally to the absolute values, it sometimes is necessary to analyze the *relative position* of a relay's bandwidth value in the (ordered) list of all bandwidth values. These values are denoted as a number between zero and one: We write $rel_{mes} = 0$ for the relay with the lowest measured bandwidth and $rel_{con} = 1$ for the relay with the highest consensus bandwidth.

4 STUDY 1: MEASUREMENT QUALITY

Using these definitions, we can describe the central goal of the TLBM as follows: Ensure that the utilization values (used bandwidth per available bandwidth) of all relays are similar to each other. As the used bandwidth is determined by the consensus bandwidth in the selection process, this is the case when the consensus bandwidth is close to the available bandwidth. However, as self-reported values are untrusted, the consensus bandwidth calculation should rely as little as possible on the advertised bandwidth. Instead, in the current TLBM, the BAs collect the measured bandwidth of a relay to calculate the proposed bandwidth. This is done by adjusting the advertised bandwidth using the relative position of the measured bandwidth. Then, these proposed bandwidth values are aggregated to the consensus bandwidth. Overall, this process is based on two core assumptions:

(1) The relative position of the measured bandwidth is a reliable indicator of the relative position of the available bandwidth.
(2) The proposed bandwidth is primarily dependent on the measured bandwidth, rather than the advertised bandwidth.

In this first study, we show that both assumptions are wrong for high-bandwidth relays.

4.1 Approach

The current TLBM assumes that the measured bandwidth (bw_{mes}) is a reliable indicator for the available bandwidth (bw_{avl}). More precisely, the relative position of the measured bandwidth (rel_{mes}) is used to estimate the relative position of the available bandwidth (rel_{avl}). Hence, given a set of pairs (rel_{avl}, rel_{mes}), we expect the correlation between both dimensions to be very strong.

However, there is no way to determine the available bandwidth accurately. Hence, one cannot calculate this correlation directly. So instead, we assume the following:

(3) Changes to the relative position of the available bandwidth are small and occur only occasionally.

We will show why this assumption is reasonable in Section 4.4.2.

Under assumptions (1) and (3), we also expect changes to rel_{mes} to be small and occur only occasionally. To evaluate this, we group the bandwidth values and their relative position by the week they were acquired, resulting in a rel_{mes} value for each relay. Then, we calculate the same values for the following week's measurements and calculate $corr(rel_{mes})$, the Pearson correlation between the two sets. [1] This value is then used as **stability metric** of the measurements over time.

Under our assumptions, we expect $corr(rel_{mes})$ to be strong for any (sufficiently large) set of relays. Based on the guidelines of Cohen [4], we call a correlation strong, if $|corr| \geq 0.5$. At the very least, we expect some correlation (i.e., $|corr| \geq 0.1$) for the majority of weeks. Note that this expectation is extremely conservative: We only require the relative position of multiple measurement results to be weakly correlated to each other. This expectation is several orders of magnitude weaker than the (probably still reasonable) expectation that there is a causal connection between multiple measurement results. However, if this expectation does not conform to reality, we can be reasonably confident that the measured bandwidth is not a good indicator of the available bandwidth.

4.2 Methodology

Only the proposed (but not the measured) bandwidth is stored and published by the measurement scripts. Hence, we need to adapt them and actively collect measurements for such an analysis.

Like the speedracer and sbws measurement scripts, we use two-hop circuits for measurements. However, only relays in the top

[1] We also calculated $corr(bw_{mes})$, the direct correlation of the measured bandwidth values of both weeks. However, note that the bandwidth – contrary to its relative position – has no linear distribution and, as such, its linear dependency (correlation) cannot be used for a meaningful interpretation.

ten percent of all consensus bandwidth values in the most recent consensus were eligible as the second hop in our measurements – contrary to relays with a similar consensus bandwidth in speedracer. This choice decreases the probability of the second relay acting as a bottleneck and achieves higher quality. Because of this, measurements might seem more stable than they are.

The measurements themselves were collected by four servers repeatedly fetching a $1.6MB$ file until either at least eight seconds passed or 15 full fetches occurred. The average bandwidth provided during this process was considered to be the measured bandwidth of the first relay. Again, these requirements go beyond the known standard configuration of both the speedracer ($172KB$ file size) and sbws measurement scripts (minimum 5 iterations), which should lead to higher measurement quality. [2]

If no full fetch of the file finished after 4 seconds, a timeout occurred, and the measurement was stopped. Additionally, we verified whether the hash of the content received matched the expected value. If it did not, the measurement was dropped. All of these events were logged and happened only very rarely.

4.3 Results

We collected $n = 1,115,617$ measurements on $r = 10,592$ different relays available during the measurement period. Each relay was, on average, measured 105.33 times (median 126). Half of the relays were measured between $n_{25} = 23$ and $n_{75} = 175$ times.

An average relay provided around $404.69KB/s$ during our measurements. This value was calculated by calculating the median bw_{mes} value for each relay individually and then calculating the median of these values. The highest median bandwidth measured of a relay with more than 10 measurements was $1124.03KB/s$. The highest bandwidth measured was $4268.1KB/s$. However, during the measurement period, 3975 relays (29.3% of all known relays) have had observed bandwidth values higher than $4268.1KB/s$. This observation already suggests that high-bandwidth relays do not offer their full available bandwidth during measurements.

4.4 Analysis

In this section, we analyze the data we collected.

4.4.1 Data Verification. We used three single-core VMs and one Raspberry Pi 3B as measurement servers. Additional to our data collection, we made hourly verification measurements (without Tor) to ensure that resource limitation on the measures did not act as a bottleneck for the measurements. The average verification measurement resulted in a speed of around $25,000KB/s$ ($\sigma = 4,850KB/s$) for the VMs and $5,500KB/s$ ($\sigma = 900KB/s$) for the Raspberry Pi.

Overall, 135 (= 2.39% of the verification measurements) were below the maximum bandwidth measured, all of them by the Pi. No verification measurement was below the average result, but 25 (0.36% of all verification measurements) were below $2,500KB/s$. As only 505 measurement results (0.05% of all measurements) resulted in $2,500KB/s$ or more, we consider all four servers suitable for the measurements.

4.4.2 Stability of the Available Bandwidth. As the measurements try to estimate the available bandwidth, this ground truth must not change between multiple measurements. Otherwise, changes to the measurement results are to be expected. As less than 1% of our measurements resulted in more than $2,000KB/s$, we do not think that hardware limitations or changes influenced the available bandwidth for the vast majority of relays. [3] Furthermore, less than 0.1% of all relays re-configured their self-reported bandwidth per week in our evaluation period. Hence, we are confident that the available bandwidth is stable for the vast majority of relays.

4.4.3 Stability of the Measured Bandwidth. Based on our approach in Section 4.1, we calculate the values $corr(rel_{mes})$ for all consecutive weeks in our measurement period.

As additional pre-processing, relays were excluded for a pair of weeks if no measurement was performed in one of the two weeks. We also provide c, the number of relays that were measured in both weeks. If multiple measurement values were acquired in a single week, the median of these values was chosen. Due to this median aggregation, the correlation might seem higher than it is.

The full results are shown in Appendix A, Table 3. For every pair of consecutive weeks, around 7,000 relays were measured in both weeks, aligned to the estimated number of active relays published by TorMetrics [10]. [4] There is a strong correlation for the relative position of the measured bandwidth ($0.72 \leq corr(rel_{mes}) \leq 0.76$) but it is not close to 1.

4.4.4 Stability of further Bandwidth Values. We also calculated the corresponding values for the proposed and consensus bandwidth, denoted $corr(rel_{pro})$ and $corr(rel_{con})$.

As additional pre-processing, values consecutively published multiple times in the network documents were excluded, and only the first new proposed or consensus bandwidth value is recorded. This is necessary, as the network documents are published hourly. However, measurements are done less frequently by the BAs and, as such, often reported multiple times.

The full results are shown in Appendix A, Table 4. Both the proposed and consensus bandwidth are very stable over time with $corr(rel_{prop})$ and $corr(rel_{con})$ being close to 1. This is unexpected as the proposed and consensus bandwidth should be primarily based on the measured bandwidth that is less stable.

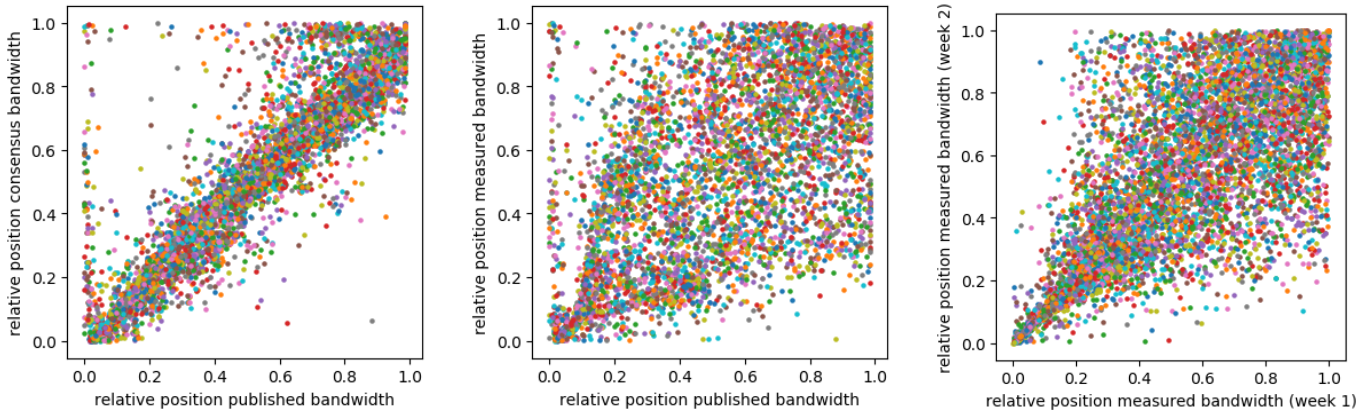
4.4.5 Dependency between Stability and Speed. As discussed in Section 4.3, high-bandwidth relays probably do not provide all of their bandwidth during measurements. The fraction of bandwidth offered during measurements is unknown and can influence the measurement result. Notably, if this fraction changes, the measurements are not reliable indicators of the available bandwidth anymore. Because of this, we analyze the dependency between the measured bandwidth of relays and their measurement stability.

To do so, we grouped the relays into four groups based on bandwidth. Then, we calculated the values described in Section 4.1 for each group individually. More precisely, we chose (300, 500, 700) as cut-off values for four different groups and inserted the relays into

[2] The speedracer measurement script does not require a minimum amount of measurements, and the default configuration of sbws does not define default destinations or file sizes. Neither approach defines a minimum measurement duration in seconds. Note that both decisions increase the resources demand on the measurer.

[3] A result of $2,000KB/s \approx 16Mbit/s$ is below the typical speed of a household internet access or the typical capacity of network components [5].

[4] It seems likely that the remaining $\approx 3,000$ relays (25% of relays) are precisely the 25% of relays measured less than 22 times (= on average, more than twice a week).



(a) The correlation between the published and consensus bandwidth: Very strong for slow and fast relays (corr=0.95).

(b) The correlation between the published and measured bandwidth: Strong for slow and negative for fast relays (corr=0.59).

(c) The correlation of multiple measurements over time: Strong for small and non-existent for fast relays (corr=0.73).

Figure 1: Visualization of some results of the first study. 1,115,617 measurements on 10,592 relays were collected.

the groups based on the first week's median measurement result. These values were a compromise between trying to have similarly sized groups while avoiding to split the large group of relays that provided around $400KB/s$. We also repeated this analysis using sightly different cut-offs and only three groups with similar results.

The full results are shown in Appendix A, Table 5. Only low-bandwidth relays ($bw_{mes} < 300$) provide high stability ($0.78 \leq corr(rel_{mes}) \leq 0.84$). The correlations in the bandwidth group $300 \leq bw_{prov} < 500$ are existent ($0.30 \leq corr(rel_{mes}) \leq 0.44$), but not strong. However, there is no correlation of the relative position of the measured bandwidth in the majority of weeks for both faster bandwidth groups ($0.02 \leq corr(rel_{mes}) \leq 0.15$).

These results show a high dependency between the measured bandwidth of relays and their measurement stability. Most importantly, only relays with a bandwidth of up to $300KB/s$ provide high stability.

4.4.6 Dependency between Bandwidth Aspects. Lastly, we analyze the alignment of the measured and published bandwidth to each other, as well as to the proposed and consensus bandwidth. Note that the published bandwidth is equal to the advertised bandwidth for the vast majority of relays claiming to offer less than $10000KB/s$. However, as the published bandwidth is stored and archived in the network documents, it is much easier to access.

Under Assumption (2) in Section 4.1, the proposed and consensus bandwidth should be aligned more closely to the measured, rather than the proposed, bandwidth. Furthermore, assuming that the measurements are accurate and that most real-world relays report honest values, the measured bandwidth should be aligned to the published bandwidth.

Again, we analyze the correlation between two bandwidth values ($corr_{bw}$) and their relative position ($corr_{rel}$). However, instead of using two consecutive bandwidth values, we use different bandwidth aspects: To analyze the dependency between the measured and the proposed bandwidth, we calculate $corr(rel_{mes}, rel_{pro})$, the correlation between the relative position of the measured bandwidth and the proposed bandwidth. As before, the median value is chosen if multiple values were collected, and values published consecutively multiple times in the network documents are ignored.

The full results are shown in Appendix A, Table 6. The correlation between the measured and the published bandwidth is strong ($corr_{rel} = 0.59$). It is stronger for relays with a maximum published bandwidth of 300 ($corr_{rel} = .62$), but even negative for relays with a minimum bandwidth of 700 ($corr_{rel} = -.37$). Contrary to this, the published bandwidth correlates very highly with both the proposed ($corr_{rel} = .93$) and consensus ($corr_{rel} = .95$) bandwidth.

Overall, the measured bandwidth of high-bandwidth relays is neither aligned to the published, proposed, or consensus bandwidth. Instead, the proposed bandwidth of high-bandwidth relays is heavily influenced by the published bandwidth. This implies that Tor primarily uses the (untrusted) published bandwidth to calculate the consensus bandwidth.

4.5 Conclusion of the first study

The current measurement-based TLBM approach tries to estimate the available bandwidth of relays with measurements. This process assumes that the relative positions of the available bandwidth and the relative position of the measured bandwidth are aligned. It also assumes that the proposed bandwidth is primarily based on the measured bandwidth. However, we showed that, for high-bandwidth relays ($bw_{mes} \geq 500KB/s$):

(i) the relative positions of multiple measurements are uncorrelated in the majority of weeks of our evaluation period.
(ii) there is a negative correlation between the measured bandwidth with both the published and proposed bandwidth.
(iii) there is a very strong correlation between the published and proposed bandwidth.

The last two results show that high-bandwidth relays' proposed bandwidth primarily depends on (untrusted) self-reported values, rather than the measured bandwidth.

Furthermore, we showed that:

(iv) multiple measurements for relays with $300KB/s \leq bw_{mes} < 500KB/s$ are correlated, but not strongly.
(v) multiple measurement for relays with $bw_{mes} < 300KB/s$ are strongly correlated, but not as strongly as could be expected by a causal connection (i.e., the correlation is not close to 1).

We visualized some aspects of these results in Figure 1. For better readability, some relays with few measurements are omitted from the plots. All of these results contradict the assumptions used to design the TLBM.

5 IMPLICATIONS FOR TOR'S SECURITY

In the first study, we showed that the measured bandwidth of high-bandwidth relays is not a good indicator for the available bandwidth and that the bandwidth attracted by them is primarily based on (untrusted) self-reported values, rather than bandwidth measurements. In this section, we discuss the implications of these results on the anonymity guarantees of the Tor network. To do so, we first introduce attacks on Tor that the TLBM is designed to defend against. Then, we argue how the deficiencies of the measurement process undermine the security guarantees of the TLBM.

5.1 Attacker Model

Our attackers try to maximize the amount of information gained per invested resources (running or compromising relays). However, our attackers' attacks are non-targeted, i.e., they do not care whose traffic is de-anonymized. Instead, their goal is to attract as much traffic as possible to subsequently analyze what is sent by whom over Tor. Such a goal might reflect the approach of large-scale attackers like intelligence agencies trying to de-anonymize information protected with Tor.

5.1.1 Attacking Methods. As a first step, our attackers focus on collecting traffic sent over Tor by: (i) running relay nodes themselves and analyzing their traffic, and (ii) taking control of other relay nodes to analyze their traffic. Then they (iii) provide as much bandwidth to clients as their hardware allows, and (iv) try to exploit the TLBM to attract more traffic than they should. For example, they might let their relays lie about their bandwidth [1] or let them provide higher bandwidth during measurements only [31]. [5]

After collecting as much traffic as possible, the attackers then analyze the traffic with three different methods corresponding to three kinds of information they want to acquire. Each method will be associated with an attacker's name to make future references more readable,

5.1.2 Guard Surveillance (Gerald). This attacker wants to know if somebody at a certain internet access point uses the Tor network – without having access to information from Internet Service Providers. Hence, the attacker tries to be the first hop in the connection and thus sees which IP is making a request over Tor. Note that this attack does not leak the content of the requests. It also is rather complicated to perform as, by default, Tor chooses very few and seldom changing entry relays called *guards* and always uses one of them as the first hop in all circuits. [6]

5.1.3 Exit Surveillance (Elise). This attacker wants to know what kind of traffic users want to protect with Tor. While traffic is encrypted over the hops in the Tor network, the exit node can see the communication stream in the same way as the recipient server. Hence, when users are using protocols with no protection on higher layers (like SMTP for mails or HTTP for browsing without TLS), controlling the exit node allows an attacker to eavesdrop on communication. Prior research has shown that some relays use this approach to, among other things, steal credentials from Tor users [34]. However, Elise cannot see where the traffic originated from unless she can re-construct this information from the stream's content.

5.1.4 Stream Surveillance (Surija). This attacker has the goal of analyzing what is sent by whom over the Tor network. He does so by compromising the entry and exit of a circuit and then performing a powerful traffic correlation attack [14, 19, 27]. This attack leads to the full de-anonymization of a certain circuit of a Tor user if performed successfully. However, this attack should be very difficult as it requires a large number of compromised relays: An attacker able to compromise $x\%$ of entry and $y\%$ of exit relays is only expected to compromise $x\% * y\%$ of all circuits as both relays randomly chosen in the circuit have to be compromised for this attack to work.

5.2 TLBM Security Guarantees

The TLBM should ensure that these attacks are as difficult as possible. Most notably, it should guarantee that:

(1) The bandwidth of relays is *verified*: The fraction of traffic attracted by a relay should not exceed its fraction of available bandwidth in the whole network. Most importantly, it should be independent of untrusted values.
(2) The network is *balanced*: No single relay should attract enough traffic to perform these attacks.

Note that in a network with differently fast relays, trade-offs between these goals are inevitable: In a network with three relays having available bandwidth values of 10, 20, and 170, there is no distribution of traffic among them that satisfies both goals. Hence, there is room for reasonable disagreement about how these trade-offs should be designed. However, in our second study, we will show that the TLBM currently does not fulfill either of these goals to a reasonable degree. Based on these goals, we define the following terms from strongest to weakest:

- We call a relay *measurable* if there is a way to use external measurements to predict the bandwidth it can provide.
- We call a relay *verifiable* if there is a way to use external measurements to detect deviations between the bandwidth it claims to provide and the bandwidth it actually provides.
- We call a relay *unverifiable* if it is neither measurable nor verifiable.

The TLBM can detect the exploits introduced in Section 5.1.1 if the relays used to perform them are verifiable. However, it cannot uphold its security guarantees if too many relays are unverifiable.

[5] Note that, because of the last attack, relays should not be able to distinguish between measurements and regular usage. However, it is possible to detect measurements in both speedracer and sbws [31]. How to prevent this is an open research problem.

[6] The number of compromised circuits is independent of this method. It merely ensures that all or none of a client's circuits using a specific guard are compromised [2, 7, 9].

5.3 Verifying Relays with Measurements

Based on this classification, it is worth discussing the question: "(How) can external measurements be used to verify relays?". We argue that, for high-bandwidth relays, they cannot: The first question to ask is what the measured bandwidth actually describes. As relays may serve multiple clients at once, one reasonable answer might be: "The amount of bandwidth expected for a future circuit". However, as bandwidth measurements are just regular usage of this relay, this is equivalent to "future measurements of this relay". Our results show that this is not the case, as neither bw_{mes} nor rel_{mes} are not stable over time.

The most likely actual answer is that the measured bandwidth does not describe any property of the relay that is meaningful, i.e., that does not change more frequently than measurements are performed and published (currently: hourly). As such, above all else, the measured bandwidth cannot be used to verify the advertised or available bandwidth of high-bandwidth relays.

5.4 Summary of our Analysis

To prevent the large-scale de-anonymization attacks introduced in Section 5.1, the TLBM must verify the bandwidth of relays. That is, the TLBM must detect relays lying about their advertised bandwidth and ensure they do not attract a larger fraction of Tor's traffic than their fraction of available bandwidth in the network.

As the current measurements for high-bandwidth relays are neither aligned to each other nor (probably) to any meaningful ground truth, measurements cannot be used for this verification. However, bandwidth measurements are currently the only mechanism for bandwidth verification. Hence, the measurements' low quality decreases the ability of the TLBM to provide its security guarantees designed to defend against these attacks.

6 STUDY II: LOAD DISTRIBUTION

In this second study, we analyze and quantify the distribution of traffic to relays in Tor. More precisely, we analyze:

(1) How much traffic is attracted by how many relays?
(2) Are there any relays that handle a disproportionate amount of traffic in Tor?
(3) How much traffic is attracted by unverifiable relays?

To perform our analysis, we use information about 12 million circuits we created to analyze the load distribution in Tor. Then, we estimate how the current load distribution simplifies the attacks explained in Section 5.1.

Furthermore, we estimate the amount of traffic attracted by unverifiable relays. This way, we can quantify the impact of the lacking bandwidth verification beyond the number of relays affected and show that the problem is bigger than one would initially expect.

6.1 Nomenclature and Implicit Assumptions

We call a relay and its streams and traffic *compromised* if an attacker has full control over it. In the event of both entry and exit being compromised, we call the circuit and its streams and traffic *fully compromised*.

We call a distributed network *centralized* if only a few available entities can be chosen for a specific task (like relaying traffic to other nodes). If there are many entities to be chosen in theory, we call the network *decentralized*. Additionally, we call a distributed network *re-centralized* if many entities can be chosen, but only a few of them are chosen in practice during the selection process. Note that a re-centralized Tor network enables an attacker to compromise more traffic with the same amount of invested resources (compromised relays) by focusing on popular relays.

To simplify writing, we assume an ideal world for an attacker in which (i) all attacks are always executed successfully, and (ii) any attack on a relay leads to full control over this relay. We also assume that (iii) all circuits handle the same amount of streams and traffic [7] and that (iv) all clients build the same amount of circuits. These assumptions allow us to simplify the statement "Elise can analyze the stream content of x% of Tor's circuits after successfully attacking x% of the exit relays in a way that gains her insight into the networking activity of that relay" to "Elise can listen to x% of Tor's exit traffic after attacking x% of the exit relays."

6.2 Data Collection

To analyze re-centralization, we collected information about the circuit-building behavior of the real-world Tor network. This data collection is necessary as multiple variables that cannot be (accurately) simulated influence this process. Examples include the assignment of special roles (e.g., guard and exit flags), the precise result of the bandwidth measurements, the relay life cycle [6], and relays entering or leaving the network.

More precisely, we used five clients to randomly build circuits in the real-world Tor network. We logged the circuit creation timestamp alongside with the relays in this circuit. During this process, we deactivated the selection of guards, as the Tor clients would otherwise choose from the same few entry relays in each circuit. Instead, they then choose their entry relays from all available relays with a guard flag.

We created and logged over 8.6 million circuits in March 2019. This data collected corresponds to more than 275,000 circuit creations per day. We consider this representative of the relay choosing behavior of the roughly 2,000,000 daily users in the Tor network [10]. On the other side, the average time of 16 minutes between the consecutive selection of a relay and the fact that we dropped each circuit immediately after creation should ensure that this process did not interfere with the capability of the selected relays to build further circuits. [8]

6.3 Analysing re-centralization in Tor

In order to analyze re-centralization, we analyze the frequency with which different relays are chosen in circuits.

6.3.1 Method. More precisely, we analyze how many relays y an attacker would need to compromise x% of all circuits at position z. Under the assumption stated in Section 6.1, this is equivalent to the number of relays needed to compromise a certain amount of Tor's traffic at a specified position. We calculated these values for

[7] There is a performance incentive to choose high-bandwidth circuits for streams that transmit a high amount of traffic. Hence, this assumption is likely to *underestimate* the traffic attracted by unverifiable high-bandwidth relays in Tor.
[8] Building a circuit and dropping it immediately afterward produces a negligible overhead on the selected relays, compared to using them.

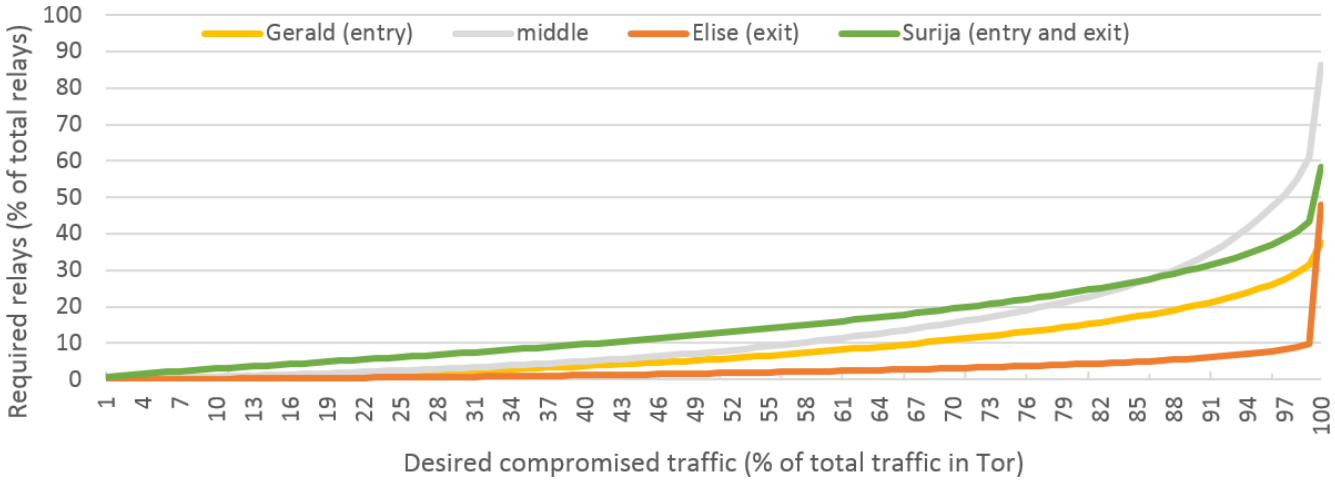


Figure 2: Influence of re-centralization on the required number of relays to perform various attacks on Tor (March 2019).

Table 2: Re-centralization in March 2019 and February 2020.

Pos (z)	Year	relays (y)						
entry	'19	3	44	136	403	941	1515	2790
	'20	3	41	135	404	955	1578	3003
middle	'19	3	53	182	554	1363	2450	6395
	'20	3	53	177	527	1253	2201	5951
exit	'19	1	11	42	124	267	435	3559
	'20	1	16	51	135	300	509	1683
stream	'19	47	220	454	920	1598	2267	4330
	'20	51	229	472	979	1727	2499	4304
traffic (x):		1%	10%	25%	50%	75%	90%	100%

the positions *only entry*, *only middle*, *only exit*, and *entry and exit*, corresponding to the attacks described in Section 5.1. Comparing these results to a selection process where ever relay is chosen with the same probability, we can quantify re-centralization in Tor. Note that $z = $ *all positions* is not as relevant as $z = $ *entry and exit* already enables the stream compromise attack of Surija. Also note that not all relays can be chosen for every position and that eavesdropping on a middle relay does not lead to new information. The corresponding numbers are only presented for comparison.

6.3.2 Results. From the 7,409 relays chosen in at least one circuit in our analysis, 2,790 (37.7%) acted as entry, 6,395 (86.3%) as middle, and 3,560 (48.0%) as exit; 2,020 (27.3%) acted as both entry and exit and 4,330 (58.6%) were eligible as at least entry or exit.

With an equally-likely selection, the number of relays required to compromise a certain percentage of traffic at a specific position scales linearly with the desired traffic for a single position. Table 2 and Figure 2 show the actual number of required relays for the specified positions z in the real Tor network.

Gerald needs to compromise 136 relays – 4.9% of all relays willing to act as an entry node – instead of 697 relays to control 25% of all entry relays chosen in circuits. In this scenario, 57.8% of clients are expected to choose at least one compromised entry node: If 25% of all entry relays are compromised, the attacker must choose a genuine relay three times (with a probability of 0.75^3) to have only genuine guards if they use the default of three guard relays.

Worse, Elise only needs to compromise a single relay (nicknamed "IPredator") to listen to 1.9% of the total exit traffic of Tor. If Elise wants to listen to 50% of the exit traffic, she needs to compromise 124 relays – a total of 3.5% of all relays willing to act as an exit node – instead of the 1,779 relays necessary in equal selection.

Surija needs to compromise 220 relays – 3.4% of all available relays – to compromise 10% of Tor's traffic. Note that in order to compromise all circuits at the entry *and* exit position, an attacker needs to compromise all relays serving at the entry *or* exit position. It is a coincidence that 220 is also 10% of all relays serving as both entry and exit. He would have needed more than four times more relays (995) to achieve the same result in equal-likely selection. [9]

Focusing on majorities, Elise needs to compromise 124 (1.7% of all relays) instead of 1,780 relays to compromise the majority of exit traffic in Tor. Gerald needs to compromise 108 (1.5% of all relays) instead of 585 relays to know about the majority of Tor users. [10] Surija needs to compromise 920 (12.4% of all relays) instead of 2437 relays to compromise the majority of Tor's traffic. [11]

6.3.3 Summary. Because of the current load distribution, an attacker can significantly reduce the resources necessary to perform various attacks by focusing on popular relays. The load distribution (as of March 2019) reduces the relays necessary to compromise the majority of traffic by 81.5% (Gerald), 93.0% (Elise), and 62.2% (Surija). Targeting less traffic, e.g., 10%, increases this reduction (Gerald: 85.6%, Elise: 96.9%, Surija: 77.9%).

[9] In equal likely selection, the percentage of compromised circuits is expected to be $c = \frac{r}{2,790} \cdot \frac{r}{3,560}$ when compromising $r \leq 2,020$ relays that acted as both entry and exit. This way, at most $c = 41.0\%$ of traffic can be compromised for $r = 2,020$.

[10] These 108 guards are chosen in 21% of all circuits. Hence, 50.7% of all Tor users will choose at least one of them with equal likely selection.

[11] If all 2,020 relays acting as both exit and entry are compromised and additionally x relays acting only as an entry, and y acting only as an exit, are compromised, the percentage of compromised circuits in equal likely selection is expected to be $c = \frac{2020+x}{2790} \cdot \frac{2020+y}{3,560}$. With $c \geq 50\%$, this equation has eight different solutions for $x + y = 417$, including $x = 205$ and $y = 212$. There are no solutions for $c \geq 50\%$ and $x + y \leq 416$.

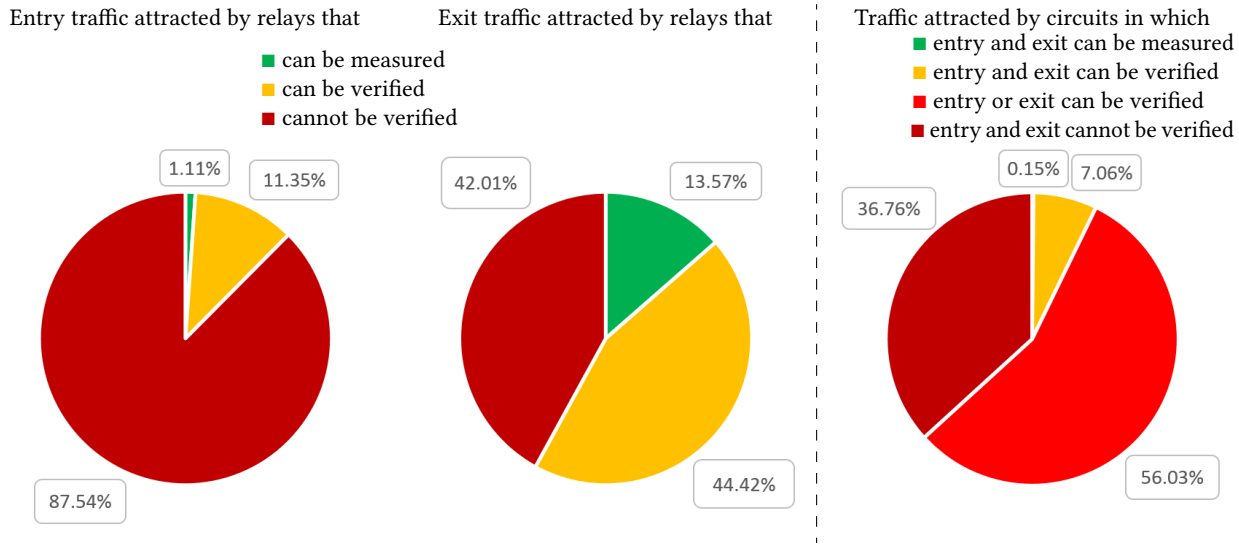


Figure 3: Influence of re-centralization on the distribution of traffic to relays that cannot be verified.

6.4 Impact of the new measurement script

The re-centralization study data was collected right before the first BA adopted the new sbws measurement script. Since we argued that the bandwidth of Tor relays is primarily based on self-reported values, and the new measurement script is very similar to the old one, it is unlikely that the new TLBM measurement method changed re-centralization in Tor. However, we want to analyze this assumption empirically.

6.4.1 Method. To do so, we repeated the data collection after the adoption of the new measurement method by all BAs. We built 3.4 million additional circuits in February 2020 and re-calculated the values described in section 6.3.

6.4.2 Results. The results of the 7,198 relays in this second analysis are also shown in Table 2. The first notable change is the slight increase in decentralization for exit traffic. This change is most likely caused by random fluctuation and the fact that the relay called "IPredator" (responsible for 1.9% of all exit traffic in the first analysis) left the Tor network in June 2019. The second notable change is the substantial decrease in the number of exit relays chosen at least once (from 3,559 down to 1,683). However, this might be an artifact of the smaller data set.

Overall, re-centralization has not changed much after the adoption of the new measurement script.

6.5 Quantifying the lack of measurements

In the previous section, we showed that Tor is quite re-centralized. This section analyzes just how much of Tor's traffic is handled by unverifiable relays.

6.5.1 Definitions. In Section 5, we argued that in order to be a meaningful concept, the measured bandwidth needs to at least be stable over time. Based on this, we state the following requirements to detect whether every entity in a set of relays is verifiable or measurable:

(1) Relays in a set are measurable if their relative bandwidth positions correlate strongly over multiple measurements.
(2) Relays in a set are verifiable if their relative bandwidth positions correlate over multiple measurements.

Based on these definitions and the results of the first study, we call relays with $bw_{mes} < 300KB/s$ measurable, relays with $300 \leq bw_{mes} < 500$ verifiable, and relays with $bw_{mes} \geq 500$ unverifiable. Note that these requirements are again lower than one could argue for. This is by design: We want to calculate a lower limit for the traffic attracted by circuits containing unverifiable relays.

6.5.2 Method. We group the circuits collected in March 2019 based on the measured bandwidth of the corresponding relay for the entry and exit position. We took the first re-centralization data set as it was collected simultaneously to the measurements necessary to choose the corresponding set. Additionally, we group the circuits into four categories:

(i) both the entry and the exit can be measured
(ii) both the entry and the exit can be verified
(iii) at least one (the entry or the exit) can be verified
(iv) both the entry and the exit cannot be verified.

The lowest applicable category was chosen: A circuit with an unverifiable entry and a measurable exit would be assigned to (iii).

6.5.3 Results. Based on our first study, 32.1% of all relays can be measured, and 28.3% can be verified. However, as shown in Figure 3, only 13.57% of all exit and 1.11% of all entry traffic passes through relays that can be measured. Additionally, only 7.21% of all circuits contain entry and exit relays that both can be verified.

Overall, re-centralization greatly increases the traffic attracted by relays or circuits unverifiable by measurements.

6.6 Conclusion of the second study

The TLBM greatly favors high-bandwidth relays during the selection of relays for circuits. This has severe consequences for the security of Tor. We showed that:

(i) Tor is highly re-centralized. For example, a single relay attracted 1.9% of Tor's exit traffic in March 2019.
(ii) The amount of re-centralization has not changed after the adoption of the new measurement method.
(iii) By focusing on popular relays, an attacker can reduce the resources necessary for most large-scale de-anonymization attacks by around 80% (up to 96.9%).
(iv) Due to re-centralization, only 7.21% of all traffic is attracted by circuits in which both the entry and exit node can be verified.

These results highlight that the insufficient measurement method impacts Tor far more significant than one would initially expect. In particular, only a small group of relays is responsible for handling a significant portion of Tor's traffic. In the majority of cases, it is not possible to verify the bandwidth of these relays by measurements.

7 DISCUSSION

In this section, we briefly discuss the implications of our results regarding Tor communitiy's current research questions.

7.1 The future of TLBM research

Much of the discussion about improving load balancing in Tor is focused on performance and the decentralization of measurements. The 2018 blog entry about open research topics in Tor [23] even uses *Network Performance: Load Balancing* as the headline for open problems in the TLBM. Aligned to this, only a single research question stated in this article is related to the quality of the TLBM. The aspects regarding security focus on a planned decentralization of the measurement process. Similarly, most research on the TLBM is focused on circuit performance. Based on our results, we argue that the current shortcomings in the quality and security of the TLBM deserve more attention than they currently do.

7.2 The case for decentralization

Research questions raised in the same article [23] include whether we should raise the minimum bandwidth required to be a guard node and whether there currently is high variance in bandwidth measurements. We can answer the last question as we showed that fast relays indeed have very high degrees of variance in (measurement) performance. More precisely, the bandwidths measured in consecutive measurements are uncorrelated to each other.

Regarding cut-offs: The currently implemented cut-off to be a guard relay amplifies the amount of traffic attracted by relays whose bandwidth cannot be verified by measurements: In our analysis, only 12.46% of the entry traffic is attracted by relays that can be verified. This is far less than the corresponding value for exit traffic (57.99%). Raising the cut-off to be a guard relay naturally increases the amount of traffic attracted by high-bandwidth relays (that cannot be verified) and, as such, also potentially increases the entry traffic of attackers lying about their bandwidth. Hence, we argue against increasing existing or introducing additional cut-offs.

Instead, it might be necessary to re-introduce upper bandwidth limits (like the $10,000 KB/s$ limit before TorFlow) to reduce re-centralization and, as such, the amount of traffic attracted by unverifiable relays. We roughly estimate that no relay should be allowed to handle more than 0.25% of Tors traffic at a specific position. While again somewhat arbitrary, this cut-off guarantees that the majority of Tors traffic is distributed to at least 200 relays. This change would have reduced the exit traffic attracted by the top 50 relays from 28.0% down to 12.5% in our March 2019 evaluation, significantly reducing Tor's re-centralization. However, additional research on how to pick these cut-offs would be necessary.

7.3 The case for a bandwidth curve

Lastly, we also argue for research into alternatives to cut-offs. As shown in this paper, Tor's overall load distribution is primarily based on the (untrusted) advertised bandwidth. Hence, as Tor is heavily re-centralized, the overall load distribution is defined by very few fast (and untrusted) relays. We argue that the load distribution should instead follow a curve hard-coded into Tor. This way, the trade-off between re-centralization and performance can be set by parameters of the DAs. Note that the design of such a curve is a tough question for future research. However, as long as we can only adjust the bandwidth for fast relays, rather than actually measure or verify it, we think it is more desirable that these trade-offs are deliberately chosen by Tor – rather than by untrusted relays.

8 RELATED WORK

In [16], the authors present ethical guidelines for data collection in Tor to which we complied. Approaches for improved path selection for better performance are explored in [20, 21, 25, 33]; an overview in [32] also includes approaches for improved anonymity.

The EigenSpeed project [26] proposes a decentralized TLBM based on a trusted subset of relays collecting metadata about communication. This approach was improved in the PeerFlow project [13] that is evaluated in [18]. The SmarTor project [12] mitigates the TLBM process from DAs to a blockchain-based smart contract. However, these approaches were never implemented in Tor.

The mTor project [15] tries to optimize client performance in Tor and limit the impact of bulk data transfer by routing bulk data about previously unused low-bandwidth relays. During their analysis, they also reported on the low utilization of low-bandwidth relays. However, they used a different metric: They simulated the number of unused relays in the system for a certain number of simultaneous Tor users. Another metric is used in [17]: Here, the number of times that each real-world router appears on a circuit together with an experiment router is used as an indicator for re-centralization. While these evaluations show some re-centralization, due to the lack of a large-scale evaluation of real-world behavior, they both cannot quantify this phenomenon.

A similar analysis of Tor's load distribution focusing on a longitudinal analysis over six months confirms a key finding: Few exit relays are far more likely to be selected during path selection than others [3]. Unfortunately, the authors do not include data on entry relays. Their longer time frame and smaller data set also cause some (expected) differences: They report on an even more significant relative popularity of some relays. Furthermore, they do not find the

large amount of barely used exit relays we identified. While their study is mostly descriptive, they provide additional insight into the dependency between internet subnets and relay popularity.

9 CONCLUSION AND FUTURE WORK

In this paper, we analyzed the measurement quality and load distribution in Tor. We showed that the currently used measurement method is not suitable to verify the bandwidth of relays offering more than $500KB/s$ as multiple measurements are uncorrelated to each other. Moreover, this problem is not limited to high-bandwidth relays: The measurement results of relays offering between $300KB/s$ and $500KB/s$ correlate only weakly with each other. Moreover, while the measurement results of relays offering less than $300KB/s$ correlate strongly with each other, this correlation is weaker than expected from a causal connection.

These shortcomings impact Tor's anonymity guarantees as the TLBM needs to verify the self-reported bandwidth values of relays to defend against several large-scale de-anonymization attacks. As of right now, despite the security guarantees of the TLBM, no verification of high-bandwidth relays is performed.

Lastly, we analyzed load distribution in Tor. We showed that the current load distribution reduces the resources necessary to perform several of the mentioned attacks by up to 96.9%. Most attacks require around 80% fewer resources if the attacker focuses on attacking popular relays. Furthermore, only 7.21% of all traffic passes through circuits containing verifiable entry and exit relays.

Based on these results, we argued to devote more attention to the security guarantees of the TLBM (rather than the resulting circuit performance). Ways to achieve this might contain the distribution of traffic to more relays and the re-introduction of upper bandwidth limits for relays. However, both steps would probably lead to decreased circuit performance.

The data collected during our research and the measurement script used to collect this data are available for verification and further analysis at our homepage [11].

In the future, additional qualitative research into how relays behave during measurements might lead to additional insight into the quality of the measurement process and more accurate groups of relays that can or cannot be measured. Overall, we are confident that our results give a good overview of the current state of load balancing in Tor. Lastly, we hope our results increase the attention devoted to improving the load balancing mechanism's quality and security, rather than only performance and decentralization.

ACKNOWLEDGMENTS

We like to thank our anonymous reviewers for their constructive and helpful feedback. Most importantly, we thank Eugene Vasserman for his time invested while shepherding this paper.

REFERENCES

[1] Kevin Bauer, Damon McCoy, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. 2007. Low-resource Routing Attacks Against Tor. In *Proceedings of the 2007 ACM Workshop on Privacy in Electronic Society.*

[2] Alex Biryukov, Ivan Pustogarov, and Ralf-Philipp Weinmann. 2013. Trawling for tor hidden services: Detection, measurement, deanonymization. In *2013 IEEE Symposium on Security and Privacy.*

[3] Tao Chen, Weiqi Cui, and Eric Chan-Tin. 2019. Measuring Tor Relay Popularity. In *Security and Privacy in Communication Networks.*

[4] Jacob Cohen. 1988. Statistical power analysis for the behavioral sciences New York. *NY: Academic* (1988).

[5] European court of auditors. 2018. Broadband in the EU Member States: despite progress, not all the Europe 2020 targets will be met. https://www.eca.europa.eu/en/Pages/DocItem.aspx?did=45796 (accessed 25.08.2020).

[6] Roger Dingledine. 2013. The lifecycle of a new relay. https://blog.torproject.org/lifecycle-new-relay (accessed 25.08.2020).

[7] Roger Dingledine, Nicholas Hopper, George Kadianakis, and Nick Mathewson. 2014. One fast guard for life (or 9 months). In *7th Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETs 2014).*

[8] Roger Dingledine, Nick Mathewson, and Paul Syverson. 2004. *Tor: The second-generation onion router.* Technical Report. Naval Research Lab Washington DC.

[9] Tariq Elahi, Kevin Bauer, Mashael AlSabah, Roger Dingledine, and Ian Goldberg. 2012. Changing of the Guards. In *Proceedings of the 2012 ACM Workshop on Privacy in the Electronic Society.*

[10] The Tor Foundationt. 2020. Tor Metrics. https://metrics.torproject.org/ (accessed 25.08.2020).

[11] André Greubel. 2020. Tor Bandwidth Measurement and Load Distribution Data. https://go.uniwue.de/20 (accessed 25.08.2020).

[12] André Greubel, Alexandra Dmitrienko, and Samuel Kounev. 2018. SmarTor: Smarter Tor with Smart Contracts. In *Proceedings of the 34th Annual Computer Security Applications Conference.*

[13] Aaron Johnson, Rob Jansen, Nicholas Hopper, Aaron Segal, and Paul Syverson. 2017. PeerFlow: Secure load balancing in Tor. In *Proceedings on Privacy Enhancing Technologies.*

[14] Aaron Johnson, Chris Wacek, Rob Jansen, Micah Sherr, and Paul Syverson. 2013. Users Get Routed: Traffic Correlation on Tor by Realistic Adversaries. In *Proceedings of the 2013 ACM Conference on Computer and Communications Security.*

[15] Lei Yang and Fengjun Li. 2015. mTor: A multipath Tor routing beyond bandwidth throttling. In *2015 IEEE Conference on Communications and Network Security.*

[16] Karsten Loesing, Steven J. Murdoch, and Roger Dingledine. 2010. A Case Study on Measuring Statistical Data in the Tor Anonymity Network. In *Financial Cryptography and Data Security.*

[17] Damon McCoy, Kevin Bauer, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. 2008. Shining Light in Dark Places: Understanding the Tor Network. In *Privacy Enhancing Technologies.*

[18] Asya Mitseva, Thomas Engel, and Andriy Panchenko. 2020. Analyzing PeerFlow. In *Sicherheit 2020.*

[19] Steven Murdoch and George Danezis. 2005. Low-cost traffic analysis of Tor. In *2005 IEEE Symposium on Security and Privacy.*

[20] Andriy Panchenko, Fabian Lanze, and Thomas Engel. 2012. Improving performance and anonymity in the Tor network. In *2012 IEEE 31st International Performance Computing and Communications Conference (IPCCC).*

[21] Andriy Panchenko, Lexi Pimenidis, and Johannes Renner. 2008. Performance Analysis of Anonymous Communication Channels Provided by Tor. In *2008 Third International Conference on Availability, Reliability and Security.*

[22] Mike Perry. 2009. Torflow: Tor network analysis. In *Proceedings of the Workshop on Hot Topics in Privacy Enhancing Technologies.*

[23] Mike Perry. 2018. Tor's Open Research Topics: 2018 Edition. https://blog.torproject.org/tors-open-research-topics-2018-edition (accessed 25.08.2020).

[24] Reporters without Borders. 2020. World press freedom index. (2020).

[25] Robin Snader and Nikita Borisov. 2008. A Tune-up for Tor: Improving Security and Performance in the Tor Network.. In *Proceedings of the Network and Distributed System Security Symposium.*

[26] Robin Snader and Nikita Borisov. 2009. EigenSpeed: secure peer-to-peer bandwidth evaluation.. In *Proceedings of the 8th international conference on Peer-to-peer systems.*

[27] Yixin Sun, Anne Edmundson, Laurent Vanbever, Oscar Li, Jennifer Rexford, Mung Chiang, and Prateek Mittal. 2015. RAPTOR: Routing Attacks on Privacy in Tor. In *24th Usenix Security Symposium.*

[28] The Tor Foundation. 2019. How Bandwidth Scanners Monitor The Tor Network. https://blog.torproject.org/how-bandwidth-scanners-monitor-tor-network (accessed 25.08.2020).

[29] The Tor Foundation. 2020. Tor Bandwidth file specification. https://gitweb.torproject.org/torspec.git/tree/bandwidth-file-spec.txt (accessed 25.08.2020).

[30] The Tor Foundation. 2020. Tor directory protocol, version 3. https://gitweb.torproject.org/torspec.git/tree/dir-spec.txt (accessed 25.08.2020).

[31] Fabrice Thill. 2014. *Hidden Service Tracking Detection and Bandwidth Cheating in Tor Anonymity Network.* Ph.D. Dissertation. University of Luxembourg.

[32] Chris Wacek, Henry Tan, Kevin S Bauer, and Micah Sherr. 2013. An Empirical Evaluation of Relay Selection in Tor.. In *Proceedings of the Network and Distributed System Security Symposium.*

[33] Tao Wang, Kevin Bauer, Clara Forero, and Ian Goldberg. 2012. Congestion-Aware Path Selection for Tor. In *Financial Cryptography and Data Security.*

[34] Philipp Winter, Richard Köwer, Martin Mulazzani, Markus Huber, Sebastian Schrittwieser, Stefan Lindskog, and Edgar Weippl. 2014. Spoiled onions: Exposing malicious Tor exit relays. In *Proceedings of the Privacy Enhancing Technologies.*

A FULL RESULTS

Table 3: Stability of measured bandwidth over time in February and March 2019. The measurement are stable but not as stable as expected by a causal dependency.

Bandwidth Aspect	start first week / end second week	02-01 / 02-14	02-08 / 02-21	02-15 / 02-28	02-22 / 03-07	03-01 / 03-14	03-08 / 03-21	03-15 / 03-28
bw_{mes}	c	7003	6888	6829	6844	6946	6967	6918
	$corr(bw_{mes})$	.65	.66	.69	.67	.67	.71	.74
	$corr(rel_{mes})$	.73	.73	.74	.72	.72	.76	.76

Table 4: Stability of the proposed and consensus bandwidth over time in February and March 2019. Both bandwidth aspects are very stable and indeed more stable than the measured bandwidth.

Bandwidth Aspect	start first week / end second week	02-01 / 02-14	02-08 / 02-21	02-15 / 02-28	02-22 / 03-07	03-01 / 03-14	03-08 / 03-21	03-15 / 03-28
bw_{pro}	c	7428	7180	7153	7172	7272	7289	7230
	$corr(bw_{pro})$	.97	.98	.97	.98	.98	.98	.98
	$corr(rel_{pro})$	.98	.99	.98	.99	.99	.99	.99
bw_{con}	c	7449	7185	7152	7220	7309	7325	7267
	$corr(bw_{con})$	.97	.98	.97	.98	.98	.98	.98
	$corr(rel_{con})$	.98	.98	.98	.98	.98	.98	.98

Table 5: Stability of the measured bandwidth, grouped by the measurement result, over time in February and March 2019. The size of a relay has a significant impact on the stability of the measurements: Faster relays are less stable.

Bandwidth Group	start first week / end second week	02-01 / 02-14	02-08 / 02-21	02-15 / 02-28	02-22 / 03-07	03-01 / 03-14	03-08 / 03-21	03-15 / 03-28
$bw_{mes} < 300$	c	2173	2261	2213	2126	2058	2144	1924
	$corr(bw_{mes})$	.64	.63	.60	.59	.55	.66	.60
	$corr(rel_{mes})$	.82	.81	.80	.78	.78	.83	.84
$300 \leq bw_{mes} < 500$	c	2211	2315	2153	2075	2225	2342	2152
	$corr(bw_{mes})$	.25	.28	.32	.30	.30	.34	.39
	$corr(rel_{mes})$	.30	.31	.35	.33	.35	.37	.44
$500 \leq bw_{mes} < 700$	c	1554	1595	1677	1748	1848	1881	2077
	$corr(bw_{mes})$	.09	.07	.05	.14	.08	.10	.12
	$corr(rel_{mes})$	.09	.07	.05	.15	.09	.11	.14
$700 \leq bw_{mes}$	c	1065	717	786	895	815	600	765
	$corr(bw_{mes})$	.09	.07	.17	.16	.02	.08	.11
	$corr(rel_{mes})$	.08	.05	.13	.08	.02	.06	.13

Table 6: Correlation between different bandwidth aspects. The TLBM assumes that the proposed bandwidth is primarily based on the measured bandwidth. For high-bandwidth relays, it correlates more strongly with the published bandwidth.

		full set		$bw < 300$		$300 \leq bw < 500$		$500 \leq bw < 700$		$700 < bw$	
Aspects 1	Aspect 2	$corr_{bw}$	$corr_{rel}$	$corr_{bw}$	$corr_{rel}$	$corr_{bw}$	$corr_{rel}$	$corr_{bw}$	$corr_{rel}$	$corr_{bw}$	$corr_{rel}$
bw_{mes}	bw_{pub}	.48	.59	.40	.62	-.01	.02	.27	.26	-.30	-.37
bw_{mes}	bw_{con}	.38	.66	.24	.74	-.04	.03	.27	.33	-.23	-.35
bw_{mes}	bw_{pro}	.39	.70	.25	.79	-.04	.04	.29	.36	-.22	-.32
bw_{pub}	bw_{pro}	.73	.93	-.08	.32	.21	.26	.05	.02	.68	.91
bw_{pub}	bw_{con}	.73	.95	-.01	.81	.18	.23	.05	.00	.68	.91